

LPS11

TEST #2

MD-11-DZLPD-C

EP-DZLPD-C-DL-A

OCT 1976

COPYRIGHT ©1976

digital

FICHE 1 OF 1

Made in U.S.A.

.REM *

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZLPD-C
PRODUCT NAME: LABORATORY PERIPHERAL SYSTEM
DIAGNOSTIC TEST II
DATE: MAY 21, 1976
MAINTAINER: DIAGNOSTIC GROUP

FIRST PRINTING, 1973

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 DIGITAL'S 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) 1973, 1974, 1976 BY DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT

THIS DIAGNOSTIC TESTS AND EXERCISES THE "LPS". THE PROGRAM IS SELF-STARTING AND WHEN LOADED WILL TYPE OUT THE PROGRAM TITLE. A SENTANCE IS THEN TYPED GIVING THE LETTER DESIGNATORS TO BE TYPED TO RUN ANY ONE OF THE SEVEN (7) SEPERATE TESTS OF WHICH THIS PROGRAM IS COMPRISED. THE PROGRAM THEN TYPES A 'CR .' AND THEN WAITS IN A KEYBOARD MONITOR MODE FOR A LETTER TO BE TYPED. ALTHOUGH THESE TESTS MAY BE RUN IN ANY ORDER IT IS IMPERATIVE THAT THE 'LOGIC' TESTS ARE RUN FIRST AND PROVED FULLY OPERATIONAL. OVER THE PROGRAM AS POSSIBLE VIA THE TELETYPE. TYPING A 'C' (OBTAINED VIA TYPING THE 'CNTR' AND 'C' KEYS SIMULTANEOUSLY) WHILE RUNNING ANY TEST WILL ENABLE THE PROGRAM TO RETURN TO THE KEYBOARD MONITOR AND AWAIT A NEW LETTER DESIGNATOR TO BE TYPED. TYPING A 'A' WHILE IN MONITOR MODE WILL ENABLE THE LETTER DESIGNATORS TO BE RETYPED. IF RUNNING ON A NON-SWITCH REGISTER CPU, TYPING A 'CTRL G' WILL ALLOW THE CHANGING OF A SOFTWARE SWITCH REGISTER.

2. REQUIREMENTS (EQUIPMENT)

- A. PDP-11 COMPUTER WITH 8K OF MEMORY.
- B. TELETYPE
- C. LPS11 OPTION BOX WITH:
 - LPSKW REAL TIME CLOCK CONTROL AND/OR
 - LPSDR DIGITAL INPUT-OUTPUT CONTROL AND/OR
 - LPSVC POINT PLOT SCOPE CONTROLLPSDRA IS SUPPORTED BY A SEPERATE DIAGNOSTIC (MD-11-DZLPI)

3. LOADING PROCEDURE

- A. USE STANDARD PROCEDURE FOR LOADING BINARY TAPES.

4. STARTING PROCEDURE

THE PROGRAM IS SELF STARTING WITH A RESTART ADDRESS OF '174' OR RE-INITIALIZED AT ADDRESS '200'.
(FOR ADDITIONAL STARTING ADDRESSES REFER TO 11.)

5. CONSOLE SWITCH SETTINGS

- A. ALL SWITCHES SHOULD BE DOWN (0) WHEN THE PROGRAM IS STARTED.
- B. REFER TO THE INDIVIDUAL TEST DESCRIPTIONS FOR APPLICABLE CONSOLE SWITCH SETTINGS
- C. REFER TO 15. FOR SOFTWARE SWITCH REGISTER OPERATION.

* TYPE 'CARRIAGE RETURN' (CR) TO TERMINATE ALL INPUT DATA.

6. CLOCK LOGIC TEST

A. THE "CLOCK LOGIC TEST" IS DESIGNED TO TEST INDIVIDUAL BITS IN THE CONTROL AND STATUS REGISTERS, COUNT PRESET BUFFER AND COUNTER ALONG WITH PROPER OPERATION UNDER INTERRUPT CONTROL MODE CONTROL

B. STARTING SEQUENCE

1. TYPE 'A' TO RUN THE CLOCK LOGIC TEST.
2. THE PROGRAM WILL THEN EXECUTE THE CLOCK LOGIC TEST.

C. CONTROL SWITCHES

1. TYPING 'C' AT ANY TIME WILL ENABLE THE PROGRAM TO EXIT 'CLOCK LOGIC' TEST AND RETURN TO THE MONITOR.

2. CONTROL SWITCH FUNCTIONS

CONSOLE SW11=0	NORMAL RUN (2048 PASSES/TEST)
CONSOLE SW11=1	SUPPRESS SUBPROGRAM INTERACTIONS
CONSOLE SW13=0	PRINT ERROR MESSAGE
CONSOLE SW13=1	INHIBIT ERROR MESSAGE
CONSOLE SW14=0	INHIBIT SCOPE MODE
CONSOLE SW14=1	RUN SCOPE MODE
CONSOLE SW15=0	CONTINUE AFTER TYPING ERROR
CONSOLE SW15=1	HALT ON ERROR

D. ERRORS

ON ENCOUNTERING AN ERROR (DATA SWITCHES DOWN) THE ERROR ADDRESS AND THE CONTENTS OF THE CLOCK STATUS AND CLOCK PRESET BUFFER ARE TYPED OUT.

E. RESTRICTIONS

NO CONNECTIONS SHOULD BE MADE TO THE SCHMITT TRIGGER.

F. TEST TIME

IT TAKES APPROXIMATELY 60 SECONDS TO RUN THE CLOCK LOGIC TEST AND RING THE TELETYPE BELL.

7. DIGITAL INPUT-OUTPUT

A. THIS TEST IS DESIGNED TO TEST THE LPSDR <DIGITAL INPUT-OUTPUT> LOGIC FOR THIS TEST THE EXTERNAL JUMPER CABLE MUST BE INSTALLED TO TEST THE DATA INPUT/OUTPUT REGISTERS AND THE CONTROL SIGNALS. IF THIS EXTERNAL JUMPER CABLE IS NOT INSTALLED, ONLY A MINIMAL LOGIC TEST CAN BE PERFORMED.

B. STARTING SEQUENCE

1. TYPE 'B' TO RUN THE 'DIGITAL I/O LOGIC' TEST.
2. THE PROGRAM WILL THEN EXECUTE THE DIGITAL I/O LOGIC TEST.

C. CONTROL SWITCHES

1. TYPING 'C' WILL CAUSE THE PROGRAM TO EXIT THE 'DIGITAL I/O LOGIC' TEST AND RETURN TO THE MONITOR.

2. CONSOLE SWITCH FUNCTION

CONSOLE SW06=0*	EXTERNAL JUMPER CABLE CONNECTED
CONSOLE SW06=1*	EXTERNAL JUMPER CABLE NOT CONNECTED
	*= ACTIVATE SWITCH BEFORE SELECTING TEST
CONSOLE SW11=0	NORMAL RUN
CONSOLE SW11=1	SUPPRESS SUBPROGRAM INTERACTIONS
CONSOLE SW13=0	PRINT ERROR MESSAGES
CONSOLE SW13=1	INHIBIT ERROR MESSAGES
CONSOLE SW14=0	INHIBIT SCOPE MODE
CONSOLE SW14=1	SCOPE MODE
CONSOLE SW15=0	CONTINUE AFTER TYPING ERROR
CONSOLE SW15=1	HALT ON ERROR

D. LOGIC ERRORS

ON ENCOUNTERING AN ERROR (DATA SWITCHES DOWN) THE ERROR ADDRESS AND THE CONTENTS OF THE DIGITAL I/O STATUS, OUTPUT AND INPUT REGISTERS ARE TYPED OUT.

E. RESTRICTIONS

NONE

F. TEST TIME

IT TAKES APPROXIMATELY 30 SECONDS TO RUN THE DIGITAL I/O LOGIC TEST AND RING THE TELETYPE BELL.

B. POINT PLOT SCOPE LOGIC TEST

A. THIS TEST IS DESIGNED TO TEST THE LPSVC SCOPE CONTROL LOGIC
ALL USABLE BITS OF THE STATUS REGISTER ARE TESTED.

B. STARTING SEQUENCE

1. TYPE 'C' TO RUN THE SCOPE LOGIC TEST.
2. THE PROGRAM WILL THEN EXECUTE THE SCOPE LOGIC TEST.

C. CONTROL SWITCHES

1. TYPEING 'C' AT ANY TIME WILL ENABLE THE PROGRAM TO EXIT
AND RETURN TO THE MONITOR.

2. <u>CONSOLE SWITCHES</u>	<u>FUNCTION</u>
----------------------------	-----------------

CONSOLE SW05=0*	611/613 NOT CONNECTED
CONSOLE SW05=1*	611/613 CONNECTED
	*= ACTIVATE SWITCH BEFORE SELECTING TEST

D. LOGIC ERRORS

ON ENCOUNTERING AN ERROR (DATA SWITCHES DOWN) THE ERROR ADDRESS
AND THE CONTENTS OF THE VC STATUS, X AXIS AND Y AXIS
REGISTERS ARE TYPED OUT ON THE TELETYPE

E. RESTRICTIONS

IF 611/613 STORAGE SCOPE IS CONNECTED, IT MUST HAVE POWER ON.

F. TEST TIME

IT TAKES APPROXIMATELY 10 SECONDS TO RUN THE SCOPE LOGIC TEST.

9. POINT PLOT VISUAL DISPLAY TEST

A. THIS TEST IS DESIGNED TO AID IN THE ADJUSTING AND ALIGNMENT OF THE VR14/20 OR 611/613 SCOPE ON THE LPSVC DISPLAY CONTROL.

B. STARTING SEQUENCE

1. TYPE 'D' TO RUN THE VISUAL DISPLAY TEST.
2. THE PROGRAM WILL THEN EXECUTE THE VISUAL DISPLAY TEST.

C. CONTROL SWITCHES

1. TYPING 'C' AT ANY TIME WILL ENABLE THE PROGRAM TO EXIT AND RETURN TO THE MONITOR.

<u>CONSOLE SWITCHES</u>	<u>FUNCTION</u>
CONSOLE SW08=0	LOOP THRU DISPLAY TEST
CONSOLE SW08=1	SELECT TEST IN SW 00-02
CONSOLE SW04=0*	PLOT CHARACTERS IN FAST INTENSIFY MODE (VR14/20)
CONSOLE SW04=1*	PLOT CHARACTERS IN NORMAL INTENSIFY MODE (611/613)
	*= ACTIVATE SWITCH BEFORE SELECTING TEST
CONSOLE SW00-02=0	DISPLAY A HORIZONTAL LINE
CONSOLE SW00-02=1	DISPLAY A VERTICAL LINE
CONSOLE SW00-02=2	DISPLAY A SQUARE
CONSOLE SW00-02=3	DISPLAY A "X"
CONSOLE SW00-02=4	DISPLAY CHARACTER SET
CONSOLE SW00-02=5	DISPLAY CHANNEL TEST (VR14/VR20)
CONSOLE SW00-02=6	DISPLAY COLOR PATTERN (VR20)
CONSOLE SW00-02=7	DISPLAY ERASE AND PHOSPOR (611/613)

D. ERRORS

NO PROVISIONS ARE MADE FOR LOGIC ERRORS. THE ONLY ERRORS IN THIS TEST ARE CHECKED VISUALLY.

E. RESTRICTIONS

IF VR14/VR20, CHANNEL SWITCH MUST BE SET TO "1 & 2" POSITION.
 IF VR20, COLOR SWITCH MUST BE SET IN THE REMOTE POSITION.
 IF 611/613, POWER MUST BE APPLIED.

F. EXECUTION TIME

IT TAKES APPROXIMATELY 90 SECONDS TO THIS TEST.

10. VISUAL DISPLAY TEST DESCRIPTIONS

DISPLAY HORIZONTAL LINE

A HORIZONTAL LINE IS DISPLAYED ON THE SCOPE BY INITIALLY SETTING THE X AND Y DAC'S TO ZERO AND THEN INCREMENTING THE X VALUE WHILE HOLDING THE Y VALUE AT ZERO. THE POINTS ARE DISPLAYED USING THE DISPLAY INTERRUPT ENABLED.

DISPLAY VERTICAL LINE

A VERTICAL LINE IS DISPLAYED ON THE SCOPE IN THE SAME MANNER AS FOR A HORIZONTAL LINE EXCEPT NOW THE Y VALUE IS INCREMENTED WHILE HOLDING THE X VALUE AT ZERO.

DISPLAY SQUARE

A SQUARE IS DISPLAYED BY INITIALLY SETTING THE X AND Y VALUES TO NEGATIVE FULL SCALE, THEN X IS INCREMENTED TO POSITIVE FULL SCALE (BOTTOM LINE) THEN Y IS INCREMENTED TO POSITIVE FULL SCALE (RIGHT LINE) THEN X IS DECREMENTED TO NEGATIVE FULL SCALE (TOP LINE) AND FINALLY Y IS DECREMENTED TO NEGATIVE FULL SCALE (LEFT LINE). MODE 01 (INTENSIFY ON LOADING X) AND MODE 10 (INTENSIFY ON LOADING Y) ARE USED.

DISPLAY X

AN X IS DISPLAYED BY INITIALLY SETTING THE X AND Y VALUES TO NEGATIVE FULL SCALE AND THEN INCREMENTING BOTH TO POSITIVE FULL SCALE (LOWER LEFT TO UPPER RIGHT DIAGONAL) THEN X IS RESET TO NEGATIVE FULL SCALE, Y REMAINS AT POSITIVE FULL SCALE AND THEN X IS INCREMENTED WHILE Y IS DECREMENTED UNTIL BOTH REACH FULL SCALE AGAIN (UPPER LEFT TO LOWER RIGHT DIAGONAL). MODE 01 (INTENSIFY ON LOADING X) IS USED.

DISPLAY ALPHA-NUMERIC CHARACTER SET

THE ALPHABET AND NUMBERS 1 THRU 0 ARE DISPLAYED.

DISPLAY CHANNEL 1 AND CHANNEL 2

THE TEXT "CHANNEL 1" IS DISPLAYED ON CHANNEL 1 SWITCH POSITION. THE TEXT "CHANNEL 2" IS DISPLAYED ON CHANNEL 2 SWITCH POSITION. THE COMBINED MESSAGE WILL APPEAR IF THE CHANNEL SELECTOR SWITCH IS IN THE 1 & 2 POSITION.

DISPLAY COLOR PATTERN

THIS ROUTINE WILL DISPLAY A BOX AROUND THE OUTER DEGE OF THE SCREEN AND A SMALL "X" IS DISPLAYED IN THE CENTER. THIS PATTERN IS PLOTTED IN GREEN AND THEN IN RED. THE END RESULT IS THAT ALL DOTS (RED AND GREEN) CONVERGE AND THE PATTERN WILL APPEAR TO BE ORANGE IN COLOR. THIS TEST IS USED TO ADJUST THE COMPENSATION GAIN AMPLIFIER IN THE VR20 DISPLAY.

PHOSPHOR AND ERASE TEST

THIS ROUTINE WILL FIRST ERASE THE 611/613 SCREEN AND THEN START INT THE UPPER RIGHT CORNOR AND FILL THE SCREEN DOWNWARD. THIS TEST

11. ADDITIONAL STARTING ADDRESSES

INCLUDED IN THIS PROGRAM ARE SEVERAL 'MINI' TEST TO AID IN THE CHECKING OF THE UNIQUE NON-PROGRAMABLE HARDWARE TO THE LPS-11 OPTION BOX.

<u>SA</u>	<u>TEST DESCRIPTION</u>
204	MANUAL SA OF THE CLOCK LOGIC TEST
210	MANUAL SA OF THE DIGITAL I/O LOGIC TEST
214	MANUAL SA OF THE SCOPE LOGIC TEST
220	MANUAL SA OF THE VISUAL SCOPE TEST
224	MINI-TEST OF SCHMITT TRIGGER #1
230	MINI-TEST OF SCHMITT TRIGGER #2
234	MINI-TEST OF CLOCK OVERFLOW
240	MINI-TEST OF RELAYS

12. OPERATOR VARIABLE LOCATIONS

LOCATION 1000 CONTAINS THE LPS STARTING DEVICE ADRESS
LOCATION 1002 CONTAINS THE LPS STARTING DEVICE VECTOR
LOCATION 1004 CONTAINS THE LPS A TO D BR LEVEL
LOCATION 1006 CONTAINS THE LPS CLOCK BR LEVEL
LOCATION 1010 CONTAINS THE LPS DIGITAL I/O BR LEVEL
LOCATION 1012 CONTAINS THE LPS DISPLAY BR LEVEL
LOCATION 1014 CONTAINS THE DELAY CONSTANT FOR MEMORY AND/OR CPU SPEED
(1 IF AN 11/20, 2 IF AN 11/45 ETC.)
LOCATION 1016 CONTAINS THE TTY FILLER COUNT
LOCATION 1020 CONTAINS THE TTY FILLER CHARACTER
LOCATION 170 CONTAINS THE SOFTWARE SWITCH REGISTER VALUE
LOCATION 172 CONTAINS THE SOFTWARE DISPLAY REGISTER VALUE

13. MISC. INFORMATION

IF THE PROGRAM WAS LOADED BY ACT-11 OR DDP, THE CLOCK LOGIC, SCOPE LOGIC AND SCOPE VISUAL WILL BE RUN.
THIS PROGRAM DOES NOT SUPPORT THE "APT" HOOKS.

14. MINI-TEST DESCRIPTIONS

SCHMITT TRIGGER #1

THE PURPOSE OF THIS MINI-TEST IS TO QUICK VERIFY THE OPERATION OF SCHMITT TRIGGER #1 ON THE CLOCK LOGIC MODULE. THIS IS DONE BY THE OPERATOR ROTATING THE THRESHOLD KNOB FROM END TO END OR CHANGING THE SLOPE SWITCH BETWEEN + OR -. THE PROGRAM WILL RING THE TTY BELL AND UPDATE THE NUMBER IN THE LPS DISPLAY LEDS UPON EACH SCHMITT TRIGGER #1 FLAG. ROTATING SCHMITT TRIGGER #2 THRESHOLD KNOB OR SWITCH SHOULD NOT CAUSE SCHMITT TRIGGER #1 TO FIRE.

SCHMITT TRIGGER #2

THE PURPOSE OF THIS MINI-TEST IS TO QUICK VERIFY THE OPERATION OF SCHMITT TRIGGER #2 ON THE CLOCK LOGIC MODULE. THIS IS DONE BY THE OPERATOR ROTATING THE THRESHOLD KNOB FROM END TO END OR CHANGING THE SLOPE SWITCH BETWEEN + OR -. THE PROGRAM WILL RING THE TTY BELL AND UPDATE THE NUMBER IN THE LPS DISPLAY LEDS UPON EACH SCHMITT TRIGGER #2 FLAG. ROTATING SCHMITT TRIGGER #1 THRESHOLD KNOB OR SWITCH SHOULD NOT CAUSE SCHMITT TRIGGER #2 TO FIRE.

CLOCK OVERFLOW

THE PURPOSE OF THIS MINI-TEST IS TO VERIFY THE OUTPUT OF THE CLOCK OVERFLOW LOGIC TO THE FRONT PANNEL.. THE CLOCK IS ENABLED TO RUN AND OVERFLOW AT A FAST RATE.
THE OUTPUT MUST BE VERIFIED WITH THE USE OF AN OSCILLOSCOPE.

RELAY TEST

THE PURPOSE OF THIS MINI-TEST, IS TO ALLOW THE OPERATOR TO VERIFY THE PROPER OPERATION OF BOTH RELAYS. THIS IS ACCOMPLISHED BY SWITCHING THE RELAYS AT A SLOW RATE TO ALLOW THE OPERATOR TO CHECK THE CONTINUITY OF THE RELAY CONTACTS.

15. SOFTWARE SWITCH REGISTER OPERATION

THE PROGRAM SUPPORTS NON-SWITCH REGISTER CPU TYPES. THIS IS ACCOMPLISHED BY TYPING A "CTRL G". THE RESPONSE WILL REPORT THE OLD VALUE AND WAIT FOR A NEW VALUE. THE OPERATOR NOW INPUTS THE NEW VALUE AND TERMINATES IT WITH A "CR". IF THE OPERATOR TYPES A "CR" WITH NO INPUT, THE SOFTWARE SWITCH REGISTER IS SET TO 0. UPON TERMINAING, THE PROGRAM WILL RESUME THE APPROPATE TEST.

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
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505

100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

000000
000001
000002
000003
000004
000005
000006
000007

```
.TITLE LPS DIAGNOSTIC TEST II MAINDEC-11-DZLPD-C
.ENABL ABS,AMA
.LIST ME
.NLIST MC,MD,CND

BIT15=100000
BIT14=40000
BIT13=20000
BIT12=10000
BIT11=4000
BIT10=2000
BIT9=1000
BIT8=400
BIT7=200
BIT6=100
BIT5=40
BIT4=20
BIT3=10
BIT2=4
BIT1=2
BIT0=1

;SWITCH REGISTER DEFINITIONS AND FUNCTIONS:
SW15=100000 ;=1, HALT ON ERROR
SW14=40000 ;=1, LOOP ON CURRENT TEST
SW13=20000 ;=1, SUPPRESS ERROR TYPEOUT
SW12=10000
SW11=4000 ;=1, SUPPRESS 'SUBPROGRAM' ITERATIONS
SW10=2000 ;=1, FORCE TYPEOUT (REPEATIBILITY)
SW09=1000 ;LPSVC SELECT TEST IN SR 0-2
SW08=400
SW07=200
SW06=100 ;=1, LPSDR JUMPER NOT CONNECTED
SW05=40
SW04=20
SW03=10
SW02=4
SW01=2
SW00=1

;LPSVC VISUAL PATTERN
;LPSVC VISUAL PATTERN
;LPSVC VISUAL PATTERN

;REGISTER DEFINITIONS
R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
SP=%6
PC=%7
```

```

506 ;LOAD TRAP CATCHER INTO LOC'S 0-1000
507      000024 012352      =24
508      000026 000340      PWRFAL      ;POWER FAIL HANDLER
509      000026 000340      340
510      000060 011734      =60
511      000062 000340      XTTYIN      ;TELEPRINTER KEYBOARD ROUTINE
512      000062 000340      340
513      000030 012452      =30
514      000032 000340      EMTSRV      ;EMT TRAP, EMT DISPATCH SERVICE
515      000034 013016      340
516      000036 000340      LOGERR      ;TRAP TRAP, LOGIC ERROR TRAP
517      000036 000340      340
518      000046 002012      =46
519      000046 000052      LOGICAL
520      000052 000000      =52
521      000170 000000      0
522      000172 000000      =170
523      000172 000000      SOFTSW: 0      ;SOFTWARE SWITCH REGISTER VALUE
524      000172 000000      SOFTDI: 0
525      000174 000137 001352      JMP      MONITR      ;PROGRAM 'RESTART' ADDRESS
526      000200 000137 001120      JMP      INIT        ;INITIALIZATION ADDRESS
527      000204 000137 002054      JMP      CKTEST      ;MANUAL SA OF CLOCK TEST
528      000210 000137 006014      JMP      IOTEST      ;MANUAL SA OF DIGITAL I/O LOGIC TEST
529      000214 000137 010034      JMP      VCTEST      ;MANUAL SA OF SCOPE LOGIC TEST
530      000220 000137 014534      JMP      VISUAL      ;MANUAL SA OF SCOPE VISUAL TEST
531      000224 000137 017414      JMP      ST1         ;MINI-TEST SCHMITT TRIGGER #1
532      000230 000137 017430      JMP      ST2         ;MINI-TEST SCHMITT TRIGGER #2
533      000234 000137 017364      JMP      CKOVFL      ;MINI-TEST CLOCK OVERFLOW
534      000240 000137 017300      JMP      RELAY       ;MINI-TEST RELAY CONTACT
535
536      104400      ;TRAP EQUIVALENCE TABLE:
537      104000      ERROR=TRAP      ;LOGIC TEST ERROR ROUTINE
538      104001      PRINT=EMT      ;MESSAGE PRINTER ROUTINE
539      104002      SCOPE0=EMT+1      ;SCOPE SUBROUTINE (1)
540      104003      SCOPE=EMT+2      ;LOGIC TEST SCOPE SUBROUTINE (4000)
541      104004      SCOPE1=EMT+3      ;LOGIC TEST SCOPE SUBROUTINE (10)
542      104005      SPACE=EMT+4      ;TYPE 'N' SPACES
543      104006      PRTCT=EMT+5      ;OCTAL PRINT ROUTINE
544      104007      TTYIN=EMT+6      ;TELETYPE INPUT ROUTINE
545      104007      TSTTKS=EMT+7      ;SUBROUTINE TO TEST FOR KEYBOARD FLAG
546
547      001000 170400      =1000
548      001002 000340      LPSADD: 170400      ;LPS STARTING ADDRESS
549      001004 000300      LPSVCT: 340        ;LPS STARTING VECTOR
550      001006 000300      ADBRL: 300        ;A TO D BR LEVEL
551      001010 000200      CKBRL: 300        ;CLOCK BR LEVEL
552      001012 000200      DIOBRL: 200       ;DIGITAL I/O BR LEVEL
553      001014 000001      VCBRL: 200        ;SCOPE BR LEVEL
554      001016 000002      PDPDLY: 1         ;1 FOR 11/20 2 FOR 11/45
555      001020 000000      FILLS: 2          ;TTY FILLER COUNT
556      001022 177776      FILCHR: 0         ;TTY FILLER CHARACTER
557      001024 177560      PSW: 177776
558      001026 177562      TKS: 177560
559      001030 177564      TKB: 177562
560      001032 177566      TPS: 177564
561      001034 177570      TPB: 177566
      SWR: 177570      ;OR LOC. 170 IF RUNNING WITH NO SWITCH REGISTER

```

562	001036	177570	DISPLA:	177570	
563	001040	000000	PASSCT:	0	
564	001042	004000	ICOUNT:	4000	
565					
566					
567					
568	001044	170400	ADCS:	170400	;A TO D STATUS/CONTROL REGISTER
569	001046	170402	ADDBR:	170402	;A TO D CONVERTED VALUE <READ ONLY>
570					;A TO D LED DISPLAY LIGHTS <WRITE ONLY>
571					
572	001050	170404	CSR:	170404	;CLOCK STATUS/CONTROL REGISTER
573	001052	170406	CSB:	170406	;CLOCK PRESET BUFFER
574					
575	001054	170410	GRSTAT:	170410	;DIGITAL I/O STATUS/COMMAND REGISTER
576	001056	170412	GRDAI:	170412	;DIGITAL I/O INPUT REGISTER <READ ONLY>
577	001060	170414	GRDIO:	170414	;DIGITAL I/O OUTPUT REGISTER
578					
579	001062	170416	VCSTAT:	170416	;POINT PLOT STATUS REGISTER
580	001064	170420	VCXREG:	170420	;POINT PLOT X AXIS
581	001066	170422	VCYREG:	170422	;POINT PLOT Y AXIS
582	001070	170424	VCEXT:	170424	;EXTERNAL DAC REGISTER
583					
584	001072	170415	GRBHIO:	170415	;DIGITAL I/O OUTPUT REGISTER <HIGH BYTE>
585					
586					
587					
588	001074	000340	ADINT:	340	;A TO D INTERRUPT VECTOR
589	001076	000342	ADINT1:	342	
590					
591	001100	000344	CKV:	344	;CLOCK INTERRUPT VECTOR
592	001102	000346	CKVS:	346	
593					
594	001104	000350	GRIVA:	350	;DIGITAL INPUT INTERRUPT VECTOR
595	001106	000352	GRIVSA:	352	
596					
597	001110	000354	GRIVB:	354	;DIGITAL OUTPUT INTERRUPT VECTOR
598	001112	000356	GRIVSB:	356	
599					
600	001114	000360	VCIV:	360	;DISPLAY INTERRUPT VECTOR
601	001116	000362	VCIVS:	362	
602					

```

603                                     ;THIS ROUTINE IS EXECUTED ON LOADING THE PROGRAM
604
605 001120 013706 014444 INIT:  MOV  STACK,SP      ;INIT STACK POINTER=1000
606 001124 012777 000340 177670  MOV  #340,SPSW
607 001132 000005 RESET      ;CLEAR THE WORLD
608 001134 012700 001044 MOV  #ADCS,RO
609 001140 013701 001000 MOV  LPSADD,R1
610 001144 012702 000013 MOV  #13,R2
611 001150 010120 INIT1A: MOV  R1,(0)+
612 001152 062701 000002 ADD  #2,R1
613 001156 005302 DEC  R2
614 001160 001373 BNE  INIT1A
615 001162 013737 001060 001072 MOV  GRDIO,GRBHIO
616 001170 052737 000001 001072 BIS  #1,GRBHIO
617 001176 012700 001074 MOV  #ADINT,RO
618 001202 013701 001002 MOV  LPSVCT,R1
619 001206 012702 000012 MOV  #12,R2
620 001212 010120 INIT1B: MOV  R1,(0)+
621 001214 062701 000002 ADD  #2,R1
622 001220 005302 DEC  R2
623 001222 001373 BNE  INIT1B
624 001224 012737 001240 000004 MOV  #1$,SWR      ;LOAD TRAP RETURN
625 001232 005777 177576 TST  SWR        ;TEST IF SWITCH REGISTER
626 001236 000407 BR  2$         ;BR IF YES
627 001240 022626 1$:  CMP  (SP)+,(SP)+
628 001242 012737 000170 001034 MOV  #170,SWR   ;LOAD LOC. 170 INTO ADDRESS
629 001250 012737 000172 001036 MOV  #172,DISPLA
630 001256 005037 013642 2$:  CLR  NOLEDS
631 001262 012737 001342 000004 MOV  #INITIC,SWR
632 001270 005077 177552 CLR  SWR
633 001274 000240 NOP
634 001276 012737 000006 000004 MOV  #6,SWR
635 001304 005737 000042 TST  SWR
636 001310 001402 BEQ  3$
637 001312 000137 001400 JMP  INIT2
638 001316 012777 000100 177500 3$:  MOV  #100,SWR   ;ENABLE TTY INTERUPTS
639 001324 005037 014442 CLR  SWR
640 001330 104000 PRINT      ;CALL MESSAGE PRINTER VIA 'EMT'
641 001332 013644 TITLE      ;TYPE PROGRAM HEADER.
642 001334 104000 INITA:  PRINT      ;PRINT THE TEST CALL LETTERS.
643 001336 014165 MES4
644 001340 000417 BR  INIT2      ;GO AND AWAIT COMMAND.
645 001342 012737 177777 013642 INIT1C: MOV  #-1,NOLEDS
646 001350 000002 RTI
647                                     ;MONITOR SUBROUTINE. ENTER VIA 'IC' OR A RESTART AT LOCATION '200'.
648 001352 000005 MONITR: RESET      ;INITIALIZE ON ENTRY
649 001354 013706 014444 MOV  STACK,SP      ;RESET STACK POINTER
650 001360 012777 000340 177434 MOV  #340,SPSW
651 001366 012777 000100 177430 MOV  #100,SWR
652 001374 104000 PRINT      ;ENABLE TTY INTERRUPTS
653 001376 014353 CNTRLC      ;CALL MESSAGE PRINTER
654                                     ;TYPE 'IC'

```

```

00000000 001400 012737 012352 000024 INIT2: MOV      #PWRFAL, @#24      ;SET UP POWER FAIL
00000000 001406 012737 000006 000004      MOV      #6, @#4        ;SET UP BUSS ERROR
00000000 001410 005077 000006      CLR      @#6
00000000 001414 013777 001076 177446      MOV      @DINT1, @ADINT
00000000 001418 005077 177444      CLR      @ADINT1
00000000 001422 013777 001102 177440      MOV      @CKVS, @CKV
00000000 001426 005077 177436      CLR      @CKVS
00000000 001430 013777 001106 177432      MOV      @GRIVSA, @GRIVA
00000000 001434 005077 177430      CLR      @GRIVSA
00000000 001438 013777 001112 177424      MOV      @GRIVSB, @GRIVB
00000000 001442 005077 177422      CLR      @GRIVSB
00000000 001446 005037 013412      CLR      SCOPEF
00000000 001450 012737 001334 014446      MOV      @INITA, @VECTR ;SET UP 'A' VECTOR ADDRESS.
00000000 001454 004537 013524      JSR      RS, LED$
00000000 001458 000006      S
00000000 001462 005737 000042      TST      @#42
00000000 001466 001402      BEQ      .+6
00000000 001470 000137 001632      JMP      WHAT
00000000 001474 012777 000100 177274      MOV      #100, @TKS ;ENABLE KEYBOARD INTERRUPT
00000000 001478 104000      PRINT
00000000 001482 014412      DOT ;PRINT ' ' TO INDICATE MONITOR READY
00000000 001486 104006      TTYIN ;WAIT FOR TTY ENTRY
00000000 001490 042737 000040 012334      BIC      #BITS, INBUF ;ENABLE LOWER CASE
00000000 001494 122737 000101 012334      CMPB    #'A', INBUF ;TEST FOR 'A'
00000000 001498 001002      BNE      .+6
00000000 001502 000137 002054      JMP      CKTEST ;YES RUN 'CLOCK LOGIC TEST'
00000000 001506 122737 000102 012334      CMPB    #'B', INBUF ;TEST FOR 'B'
00000000 001510 001002      BNE      .+6 ;NOT 'B'
00000000 001514 000137 006014      JMP      IOTEST ;YES RUN 'I/O LOGIC TEST'
00000000 001518 122737 000103 012334      CMPB    #'C', INBUF ;TEST FOR 'C'
00000000 001522 001002      BNE      .+6 ;NOT 'C'
00000000 001526 000137 010034      JMP      VCTEST ;YES, RUN 'SCOPE LOGIC TEST'
00000000 001530 122737 000104 012334      CMPB    #'D', INBUF ;TEST FOR 'D'
00000000 001534 001002      BNE      .+6 ;NOT 'D'
00000000 001538 000137 014534      JMP      VISUAL ;YES, RUN 'VISUAL DISPLAY TEST'
00000000 001542 104000      PRINT ;ILLEGAL ENTRY
00000000 001546 014416      OMARK ;TYPE '?'
00000000 001550 000662      BR      INIT2 ;WAIT AGAIN

```

:EXECUTE ONLY IF LOCATION 42 IS NON ZERO

699	001632	005037	002024	WHAT:	CLR	NOCLK	:CLEAR NO CLOCK
700	001636	005037	002026		CLR	NODIO	:CLEAR NO DIGITAL I/O
701	001642	005037	002030		CLR	NOSCOPE	:CLEAR NO SCOPE
702	001646	012737	002032	000004	MOV	#WHAT1,2#4	
703	001654	005077	177170		CLR	ACSR	:TEST FOR CLOCK
704	001660	012737	002040	000004	MOV	#WHAT2,2#4	
705	001666	005077	177162		CLR	AGRSTAT	:TEST FOR DIGITAL I/O
706	001672	012737	002046	000004	MOV	#WHAT3,2#4	
707	001700	005077	177156		CLR	AVCSTAT	:TEST FOR SCOPE
708	001704	012737	000006	000004	MOV	#6,2#4	
709	001712	005737	002024		TST	NOCLK	:TEST FOR NO CLOCK
710	001716	001410			BEQ	WHATA	:BRANCH IF CLOCK
711	001720	005737	002026		TST	NODIO	:TEST FOR NO DIGITAL I/O
712	001724	001405			BEQ	WHATA	:BRANCH IF DIGITAL I/O
713	001726	005737	002030		TST	NOSCOPE	:TEST FOR NO SCOPE
714	001732	001402			BEQ	WHATA	:BRANCH IF SCOPE
715	001734	000000			HALT		:FATAL ERROR, NO SLAVE SYNC FROM ANY LPS-11 DEVICE
716	001736	000777			BR	.	:HANG HERE
717	001740	005737	002024	WHATA:	TST	NOCLK	:TEST FOR CLOCK
718	001744	001002			BNE	WHATA	:BRANCH IF NO CLOCK
719	001746	000137	002062		JMP	CTEST1	:TEST CLOCK
720	001752	005737	002026	WHATA:	TST	NODIO	:TEST FOR DIGITAL I/O
721	001756	000402			BR	WHATA	:DONT RUN LPS-11-DR IN CHAIN MODE (NEWER LPS-11-DR)
722	001760	000137	006022		JMP	ITEST1	:RUN DIGITAL I/O TEST
723	001764	005737	002030	WHATA:	TST	NOSCOPE	:TEST FOR SCOPE
724	001770	001004			BNE	WHATA	:BRANCH IF NO SCOPE
725	001772	000137	010042		JMP	VTEST1	:TEST SCOPE LOGIC
726	001776	000137	014542	WHATA:	JMP	VTEST2	:TEST VISUAL
727	002002	000005		WHATE:	RESET		
728	002004	000005			RESET		
729	002006	013700	000042		MOV	2#42,R0	
730	002012	004710		LOGICAL:	JSR	PC,(0)	
731	002014	000240			NOP		
732	002016	000240			NOP		
733	002020	000240			NOP		
734	002022	000703			BR	WHAT	
735	002024	000000		NOCLK:	0		
736	002026	000000		NODIO:	0		
737	002030	000000		NOSCOPE:	0		
738	002032	005137	002024	WHAT1:	COM	NOCLK	
739	002036	000002			RTI		
740	002040	005137	002026	WHAT2:	COM	NODIO	
741	002044	000002			RTI		
742	002046	005137	002030	WHAT3:	COM	NOSCOPE	
743	002052	000002			RTI		

74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92

002054 000005
002056 104000
002060 014030
002062 005037 001040
002066 013777 001040 176742
002074 013706 014444
002100 005077 176716
002104 012737 002132 013414
002112 052777 000100 176704

002120 005077 176724
002124 005077 176722

002130 104002
002132 012777 177777 176712
002140 022777 177777 176704
002146 001401
002150 104400

002152 104002
002154 012777 052525 176670
002162 022777 052525 176662
002170 001401
002172 104400

002174 104002
002176 012777 025252 176646
002204 022777 025252 176640
002212 001401
002214 104400

002216 104003
002220 012777 177777 176624
002226 000005
002230 005777 176616
002234 001401
002236 104400

002240 052777 000100 176556

```
*****
:      CLOCK LOGIC TEST
*****
CKTEST:  RESET
          PRINT
          MES2          ;IDENTIFY TEST
CTEST1:  CLR          PASSCT
BEGIN:   MOV          PASSCT,DISPLA
          MOV          STACK,SP
          CLR          @PSW
          MOV          @KWTO+2,RETURN ;SET UP RESTART OF PROGRAM
          BIS          @BIT6,@TKS

:TEST FOR NO BUSS ERRORS
          CLR          @CSR
          CLR          @CSB

:TEST THE COUNTER PRESET BUFFER
KWT0:    SCOPE
          MOV          #-1,@CSB      ;LOAD PRESET BUFFER
          CMP          #-1,@CSB
          BEQ          .+4           ;BRANCH IF EQUAL
          ERROR        ;ERROR, COUNTER PRESET FAILED TO LOAD

KWT2:    SCOPE
          MOV          @52525,@CSB   ;LOAD PRESET BUFFER
          CMP          @52525,@CSB
          BEQ          .+4           ;BRANCH IF EQUAL
          ERROR        ;ERROR, COUNTER PRESET FAILED TO LOAD

KWT3:    SCOPE
          MOV          @25252,@CSB   ;LOAD PRESET
          CMP          @25252,@CSB
          BEQ          .+4           ;BRANCH IF EQUAL
          ERROR        ;ERROR, COUNTER PRESET FAILED TO LOAD

:TEST INIT TO CLEAR COUNT PRESET BUFFER WHEN IT IS =-1
KWT4:    SCOPE1
          MOV          #-1,@CSB
          RESET
          TST          @CSB
          BEQ          .+4
          ERROR        ;ERROR, INIT FAILED TO CLEAR CSB

          BIS          @BIT6,@TKS
```

793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825

002246 104003
002250 012777
002256 022777
002264 001401
002266 104400

002270 104002
002272 012777
002300 022777
002306 001401
002310 104400

002312 104002
002314 012777
002322 022777
002330 001401
002332 104400

002334 104002
002336 012777
002344 022777
002352 001401
002354 104400

000001 176572
000001 176564

000002 176550
000002 176542

000004 176526
000004 176520

000010 176504
000010 176476

;TEST ENABLE COUNTER (BIT 0) CAN BE SET AND CLEARED

KWT5: SCOPE1
MOV #BIT0,@CSR
CMP #BIT0,@CSR
BEQ .+4
ERROR

;ERROR COUNTER ENABLE FAILED TO SET

;TEST RATE SELECT (BIT 1) MAY BE SET AND CLEARED

KWT6: SCOPE
MOV #2,@CSR
CMP #2,@CSR
BEQ .+4
ERROR

;ERROR, CSR NOT = 2

;TEST THAT RATE SELECT (BIT 2) MAY BE SET AND CLEARED

KWT7: SCOPE
MOV #4,@CSR
CMP #4,@CSR
BEQ .+4
ERROR

;ERROR, CSR NOT = 4

;TEST THAT RATE SELECT (BIT 3) MAY BE SET AND CLEARED

KWT8: SCOPE
MOV #10,@CSR
CMP #10,@CSR
BEQ .+4
ERROR

;ERROR, CSR NOT = 10

026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053
054
055
056
057
058

```

002356 104002
002360 012777 000100 176462
002366 022777 000100 176454
002374 001401
002376 104400

002400 104002
002402 012777 000400 176440
002410 022777 000400 176432
002416 001401
002420 104400

002422 104002
002424 012777 001000 176416
002432 022777 001000 176410
002440 001401
002442 104400

002444 104002
002446 012777 020000 176374
002454 022777 020000 176366
002462 001401
002464 104400
    
```

;TEST MODE INTERRUPT ENABLE (BIT 6) CAN BE SET AND CLEARED

```

KWT10: SCOPE
MOV #100,CSR
CMP #100,CSR
BEQ .+4
ERROR ;ERROR, CSR NOT = 100
    
```

;TEST MODE (BIT 8) CAN BE SET AND CLEARED

```

KWT11: SCOPE
MOV #400,CSR
CMP #400,CSR
BEQ .+4
ERROR ;ERROR, CSR NOT = 400
    
```

;TEST MODE (BIT 9) CAN BE SET AND CLEARED

```

KWT12: SCOPE
MOV #1000,CSR
CMP #1000,CSR
BEQ .+4
ERROR ;ERROR, CSR NOT = 1000
    
```

;TEST ST#1 START ENABLE (BIT 13) CAN BE SET AND CLEARED

```

KWT14: SCOPE
MOV #20000,CSR
CMP #20000,CSR
BEQ .+4
ERROR ;ERROR, CSR NOT = 20000
    
```

```

859
860 ;TEST ST#1 INTERRUPT ENABLE (BIT 14) CAN BE SET AND CLEARED
861
862 002466 104002 KWT15: SCOPE
863 002470 012777 040000 176352 MOV #40000,@CSR
864 002476 022777 040000 176344 CMP #40000,@CSR
865 002504 001401 BEQ .+4
866 002506 104400 ERROR ;ERROR, CSR NOT = 40000
867
868 ;TEST THAT THE DONE (BIT 7) CAN BE SET AND CLEARED
869
870 002510 104002 KWT16: SCOPE
871 002512 012777 000200 176330 MOV #BIT7,@CSR ;SET THE DONE BIT
872 002520 105777 176324 TSTB @CSR
873 002524 100401 BMI .+4
874 002526 104400 ERROR ;ERROR, CSR NOT = 200
875 002530 105777 176314 TSTB @CSR
876 002534 100401 BMI .+4
877 002536 104400 ERROR ;ERROR CSR NOT = 200
878 002540 005077 176304 CLR @CSR
879 002544 005777 176300 TST @CSR
880 002550 001401 BEQ .+4
881 002552 104400 ERROR ;ERROR CSR NOT = 0
882
883 ;TEST THAT THE ST FLAG (BIT 15) CAN BE SET AND CLEARED
884
885 002554 104002 KWT17: SCOPE
886 002556 012777 100000 176264 MOV #BIT15,@CSR ;SET BIT 15
887 002564 005777 176260 TST @CSR
888 002570 100401 BMI .+4
889 002572 104400 ERROR ;ERROR CSR NOT 100000
890 002574 005777 176250 TST @CSR
891 002600 100401 BMI .+4
892 002602 104400 ERROR ;ERROR CSR NOT 100000
893 002604 005077 176240 CLR @CSR
894 002610 005777 176234 TST @CSR
895 002614 001401 BEQ .+4
896 002616 104400 ERROR ;ERROR CSR NOT = 0
897
898 ;TEST THAT THE ST FLAG DOES NOT SET FROM THE OUTSIDE SOURCE
899
900 002620 104002 KWT18: SCOPE
901 002622 005077 176222 CLR @CSR
902 002626 005037 014506 CLR TEMP
903 002632 005777 176212 KWT18A: TST @CSR
904 002636 100001 BPL .+4
905 002640 104400 ERROR ;ERROR ST1 SET IN ERROR
906 002642 005237 014506 INC TEMP
907 002646 001371 BNE KWT18A

```



```

931 ;TEST THAT MODE 1 AND BIT 10 (MAINT. ST 2) SET BIT 7
932
933 002730 104002 KWT22: SCOPE
934 002732 005077 176112 CLR @CSR ;CLEAR STATUS
935 002736 005077 176110 CLR @CSB ;CLEAR PRESET
936 002742 012777 001000 176100 MOV #BIT9,@CSR ;LOAD MODE
937 002750 052777 002000 176072 BIS #BIT10,@CSR ;FIRE ST 2
938 002756 105777 176066 TSTB @CSR ;TEST BIT 7
939 002762 100401 BMI .+4
940 002764 104400 ERROR ;ERROR, ST 2 FAILED TO SET BIT 7
941
942 ;TEST THAT THE COUNTER CAN BE LOADED
943
944 002766 104002 KWT23: SCOPE
945 002770 005077 176054 CLR @CSR ;CLEAR STATUS
946 002774 012777 177777 176050 MOV #-1,@CSB ;LOAD PRESET AND COUNTER
947 003002 012777 001401 176040 MOV #1401,@CSR ;LOAD MODE AND ENABLE COUNT
948 003010 005077 176036 CLR @CSB ;SHOULD ONLY CLEAR PRESET
949 003014 052777 002000 176026 BIS #BIT10,@CSR ;FIRE ST
950 003022 022777 177777 176022 CMP #-1,@CSB ;TEST THE NUMBER
951 003030 001401 BEQ .+4
952 003032 104400 ERROR ;ERROR, COUNTER FAILED TO LOAD PROPERLY
953
954 ;TEST THAT THE COUNTER CAN BE LOADED
955
956 003034 104002 KWT24: SCOPE
957 003036 005077 176006 CLR @CSR ;CLEAR STATUS
958 003042 012777 125252 176002 MOV #125252,@CSB ;LOAD PRESET AND COUNTER
959 003050 012777 001401 175772 MOV #1401,@CSR ;LOAD MODE AND ENABLE COUNT
960 003056 005077 175770 CLR @CSB ;SHOULD ONLY CLEAR PRESET
961 003062 052777 002000 175760 BIS #BIT10,@CSR ;FIRE ST
962 003070 022777 125252 175754 CMP #125252,@CSB ;TEST THE NUMBER
963 003076 001401 BEQ .+4
964 003100 104400 ERROR ;ERROR, COUNTER FAILED TO LOAD PROPERLY
965
966 ;TEST THAT THE COUNTER CAN BE LOADED
967
968 003102 104002 KWT25: SCOPE
969 003104 005077 175740 CLR @CSR ;CLEAR STATUS
970 003110 012777 052525 175734 MOV #52525,@CSB ;LOAD PRESET AND COUNTER
971 003116 012777 001401 175724 MOV #1401,@CSR ;LOAD MODE AND ENABLE COUNT
972 003124 005077 175722 CLR @CSB ;SHOULD ONLY CLEAR PRESET
973 003130 052777 002000 175712 BIS #BIT10,@CSR ;FIRE ST
974 003136 022777 052525 175706 CMP #52525,@CSB ;TEST THE NUMBER
975 003144 001401 BEQ .+4
976 003146 104400 ERROR ;ERROR, COUNTER FAILED TO LOAD PROPERLY
977

```

```

978
979
980
981
982
983
984
985 003150 005037 014506
986 003154 104002
987 003156 005077 175666
988 003162 005077 175664
989 003166 012777 001002 175654
990 003174 052777 002000 175646
991 003202 027737 175644 014506
992 003210 001402
993 003212 104400
994 003214 000416
995 003216 105777 175626
996 003222 100402
997 003224 104400
998 003226 000411
999 003230 052777 004000 175612
1000 003236 042777 000200 175604
1001 003244 005237 014506
1002 003250 001351
1003

;TEST THAT THE COUNTER COUNTS UP
;USE MAINT COUNT
;USE MAINT ST2
;USE MODE 2
;USE RATE 1 <1MHZ>

KWT26: CLR TEMP
SCOPE
CLR @CSR
CLR @CSB
MOV #1002,@CSR
KWT26A: BIS #BIT10,@CSR ;MAINT ST2
CMP @CSB,TEMP
BEQ .+6
ERROR ;MAINT COUNT FAILED
BR KWT30
TSTB @CSR ;TEST DONE FLAG
BMI 1$
ERROR ;ERROR, CKBF TO BF FAILED TO SET DONE
BR KWT30
BIS #BIT11,@CSR ;MAINT COUNT
BIC #BIT7,@CSR ;CLEAR BIT 7
INC TEMP
BNE KWT26A ;BRANCH IF NOT FULL COUNT

```

1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056

;TEST THAT THE COUNTER COUNTS UP
;MAKE SURE THAT THE CLOCK SELECTION LOGIC WORKS
;USE MAINT COUNT <100KHZ>
;USE MAINT ST2
;USE MODE 2
;RATE #100KHZ

KWT30: SCOPE1
CLR @CSR ;CLEAR CLOCK STATUS
CLR @CSB ;CLEAR PRESET
MOV #1004,@CSR ;LOAD STATUS
BIS #BIT10,@CSR ;MAINT ST2
TST @CSB ;TEST PRESET
BEQ .+4
ERROR ;ERROR PRESET INCREMENTED IN ERROR
MOV #10,COUNT ;SET UP A COUNTER
KWT30A: BIS #BIT11,@CSR ;GENERATE MAINT COUNT
DEC COUNT
BNE KWT30A ;BR
BIS #BIT10,@CSR ;MAINT ST2
CMP #1,@CSB ;
BEQ .+4
ERROR ;ERROR, CLOCK PRESET BUFFER
;COUNTED IN ERROR, FAULT IS PROBILY IN THE
;CLOCK DOWN COUNT OR RATE SELECTION LOGIC

;TEST THAT THE COUNTER COUNTS UP
;MAKE SURE THAT THE CLOCK SELECTION LOGIC WORKS
;USE MAINT COUNT <10KHZ>
;USE MAINT ST2
;USE MODE 2
;RATE #10KHZ

KWT31: SCOPE1
CLR @CSR ;CLEAR CLOCK STATUS
CLR @CSB ;CLEAR PRESET
MOV #1006,@CSR ;LOAD STATUS
BIS #BIT10,@CSR ;MAINT ST2
TST @CSB ;TEST PRESET
BEQ .+4
ERROR ;ERROR PRESET INCREMENTED IN ERROR
MOV #100,COUNT ;SET UP A COUNTER
KWT31A: BIS #BIT11,@CSR ;GENERATE MAINT COUNT
DEC COUNT
BNE KWT31A ;BR
BIS #BIT10,@CSR ;
CMP #1,@CSB ;
BEQ .+4
ERROR ;ERROR, CLOCK PRESET BUFFER
;COUNTED IN ERROR, FAULT IS PROBILY IN THE
;CLOCK DOWN COUNT OR RATE SELECTION LOGIC

```

1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108

```

```

;TEST THAT THE COUNTER COUNTS UP
;MAKE SURE THAT THE CLOCK SELECTION LOGIC WORKS
;USE MAINT COUNT <1KHZ>
;USE MAINT ST2
;USE MODE 2
;RATE #1KHZ

KWT32: SCOPE1
CLR @CSR ;CLEAR CLOCK STATUS
CLR @CSB ;CLEAR PRESET
MOV #1010,@CSR ;LOAD STATUS
BIS #BIT10,@CSR ;MAINT ST2
TST @CSB ;TEST PRESET
BEQ .+4
ERROR ;ERROR PRESET INCREMENTED IN ERROR
MOV #1000.,COUNT ;SET UP A COUNTER
KWT32A: BIS #BIT11,@CSR ;GENERATE MAINT COUNT
DEC COUNT
BNE KWT32A ;BR
BIS #BIT10,@CSR
CMP #1,@CSB
BEQ .+4
ERROR ;ERROR, CLOCK PRESET BUFFER
;COUNTED IN ERROR, FAULT IS PROBILY IN THE
;CLOCK DOWN COUNT OR RATE SELECTION LOGIC

```

```

;TEST THAT THE COUNTER COUNTS UP
;MAKE SURE THAT THE CLOCK SELECTION LOGIC WORKS
;USE MAINT COUNT <100HZ>
;USE MAINT ST2
;USE MODE 2
;RATE #100HZ

KWT33: SCOPE1
CLR @CSR ;CLEAR CLOCK STATUS
CLR @CSB ;CLEAR PRESET
MOV #1012,@CSR ;LOAD STATUS
BIS #BIT10,@CSR ;MAINT ST2
TST @CSB ;TEST PRESET
BEQ .+4
ERROR ;ERROR PRESET INCREMENTED IN ERROR
MOV #10000.,COUNT ;SET UP A COUNTER
KWT33A: BIS #BIT11,@CSR ;GENERATE MAINT COUNT
DEC COUNT
BNE KWT33A ;BR
BIS #BIT10,@CSR
CMP #1,@CSB
BEQ .+4
ERROR ;ERROR, CLOCK PRESET BUFFER
;COUNTED IN ERROR, FAULT IS PROBILY IN THE
;CLOCK DOWN COUNT OR RATE SELECTION LOGIC

```

```

1109
1110 ;TEST THAT THE COUNTER COUNTS UP
1111 ;MAKE SURE THAT THE CLOCK SELECTION LOGIC WORKS
1112 ;USE MODE 2
1113 ;USE MAINT ST1
1114 ;RATE #EXTURNAL
1115
1116 003652 005037 014506          CLR      TEMP
1117 003656 104003          KWT34:  SCOPE1
1118 003660 005077 175164          CLR      @CSR
1119 003664 005077 175162          CLR      @CSB
1120 003670 012777 001014 175152  MOV      #1014,@CSR      ;MODE 2 EXTURNAL CLOCK RATE
1121 003676 052777 002000 175144  KWT34A: BIS      #BIT10,@CSR  ;MAINT ST 2
1122 003704 027737 175142 014506  CMP      @CSB,TEMP      ;COMPARE BUFFER TO PRESET
1123 003712 001402          BEQ      .+6
1124 003714 104400          ERROR
1125 003716 000406          BR      KWT35           ;EXTURNAL CLOCDAILED TO COUNT
1126 003720 052777 010000 175122  BIS      #BIT12,@CSR    ; THE COUNTER, CHECK THE RATE SELECTION
1127 003726 005237 014506          INC      TEMP          ;MAINT ST1
1128 003732 001361          BNE      KWT34A
1129
1130 ;TEST EXTURNAL INTERVAL FROM ZERO BASE (MODE 3)
1131
1132 003734 104003          KWT35:  SCOPE1
1133 003736 005077 175106          CLR      @CSR
1134 003742 012777 025252 175102  MOV      #25252,@CSB
1135 003750 012777 001402 175072  MOV      #1402,@CSR
1136 003756 052777 002000 175064  BIS      #BIT10,@CSR
1137 003764 022777 025252 175060  CMP      #25252,@CSB
1138 003772 001401          BEQ      .+4
1139 003774 104400          ERROR
1140 ;ERROR, MODE 3 FAILED TO LOAD CLOCK
1141 ;PRESET BUFFER
1141 003776 052777 002000 175044  BIS      #BIT10,@CSR
1142 004004 022777 000000 175040  CMP      #0,@CSB
1143 004012 001401          BEQ      .+4
1144 004014 104400          ERROR
1144 ;ERROR, MODE 3 FAILED TO ZERO CLOCK COUNTER

```

```

1145
1146 ;TEST THAT RESET CLEARS RATE SELECT BITS
1147
1148 004016 104002 KWT36: SCOPE1
1149 004020 012777 000016 175022 MOV #BIT3!BIT2!BIT1,@CSR ;SET MODE BITS
1150 004026 000005 RESET
1151 004030 005777 175014 TST @CSR
1152 004034 001401 BEQ .+4
1153 004036 104400 ERROR ;ERROR, RESET FAILED TO CLEAR RATE BITS
1154
1155 ;TEST THAT RESET CLEARS MODE SELECT BITS
1156
1157 004040 104003 KWT37: SCOPE1
1158 004042 012777 001400 175000 MOV #BIT9!BIT8,@CSR ;SET MODE BITS
1159 004050 000005 RESET
1160 004052 005777 174772 TST @CSR
1161 004056 001401 BEQ .+4
1162 004060 104400 ERROR ;ERROR, RESET FAILED TO CLEAR MODE BITS
1163
1164 ;TEST THAT RESET CLEARS ST1 FLAG,ST1 INTERRUPT ENABLE AND ST1 START ENABLE
1165
1166 004062 104003 KWT38: SCOPE1
1167 004064 012777 160000 174756 MOV #BIT15!BIT14!BIT13,@CSR ;SET ST1 FLAG,INT ENABLE AND ST1 START ENABLE
1168 004072 000005 RESET
1169 004074 005777 174750 TST @CSR
1170 004100 001401 BEQ .+4
1171 004102 104400 ERROR ;ERROR, RESET FAILED TO CLEAR ST1 LOGIC
1172
1173 ;TEST THAT RESET CLEARS DONE FLAG AND DONE INTERRUPT ENABLE
1174
1175 004104 104003 KWT39: SCOPE1
1176 004106 012777 000300 174734 MOV #BIT7!BIT6,@CSR ;SET DONE FLAG AND INTERRUPT ENABLE
1177 004114 000005 RESET
1178 004116 005777 174726 TST @CSR
1179 004122 001401 BEQ .+4
1180 004124 104400 ERROR ;ERROR, RESET FAILED TO CLEAR DONE FLAG OR DONE
1181
1182 ;TEST THAT RESET CLEARS COUNTER ENABLE
1183
1184 004126 104003 KWT40: SCOPE1
1185 004130 012777 000001 174712 MOV #BIT0,@CSR ;LOAD COUNTER ENABLE
1186 004136 000005 RESET
1187 004140 005777 174704 TST @CSR ;TEST STATUS
1188 004144 001401 BEQ .+4
1189 004146 104400 ERROR ;ERROR, RESET FAILED TO CLEAR COUNTER ENABLE
1190
1191
1192 004150 052777 000100 174646 BIS #BIT6,@TKS

```

:TEST CLOCK TO COUNT UP AT ALL FREQUENCIES (MODE 0)
:TEST THAT CLOCK ENABLE DOES CLEAR ON DONE FLAG
:1MHZ

11193
11194
11195
11196
11197
11198
11199
11200
11201
11202
11203
11204
11205
11206
11207
11208
11209
11210
11211
11212
11213
11214
11215
11216
11217
11218
11219
11220
11221
11222
11223
11224
11225
11226
11227
11228
11229
11230
11231
11232
11233
11234
11235
11236
11237
11238
11239
11240
11241
11242
11243
11244
11245
11246
11247
11248
11249
11250
11251
11252
11253
11254
11255
11256
11257
11258
11259
11260
11261
11262
11263
11264
11265
11266
11267
11268
11269
11270
11271
11272
11273
11274
11275
11276
11277
11278
11279
11280
11281
11282
11283
11284
11285
11286
11287
11288
11289
11290
11291
11292
11293
11294
11295
11296
11297
11298
11299
11300

004156 104003
004160 012737 000003 006004
004166 004737 004470
004172 105777 174652
004178 100401
004184 104400
004190 032777 000001 174640
004196 001401
004202 104400

KWT41: SCOPE1
MOV #3,RATE ;SELECT MODE 0, 1MHZ., GO
JSR PC,UPCNT
TSTB 3CSR
BMI .+4
ERROR ;ERROR, 1MHZ FAILED TO COUNT
BIT #BIT0,3CSR
BEQ .+4
ERROR ;ERROR, MODE 0 FAILED TO CLEAR COUNTER ENABLE

:TEST CLOCK TO COUNT UP AT ALL FREQUENCIES
:100 KHZ

004214 104003
004218 012737 000005 006004
004224 004737 004470
004230 105777 174614
004236 100401
004242 104400
004248 032777 000001 174602
004254 001401
004260 104400

KWT42: SCOPE1
MOV #5,RATE ;SELECT MODE 0, 100KHZ., GO
JSR PC,UPCNT
TSTB 3CSR
BMI .+4
ERROR ;ERROR, 100KHZ. FAILED TO COUNT
BIT #BIT0,3CSR
BEQ .+4
ERROR ;ERROR MODE 0 FAILED TO CLEAR COUNT ENABLE

:10 KHZ

004272 104003
004276 012737 000007 006004
004282 004737 004470
004288 105777 174556
004294 100401
004300 104400

KWT43: SCOPE1
MOV #7,RATE ;SELECT MODE 0, 10KHZ., GO
JSR PC,UPCNT
TSTB 3CSR
BMI .+4
ERROR ;ERROR, 10 KHZ. FAILED TO COUNT

:TEST CLOCK TO COUNT UP AT ALL FREQUENCIES
:1KHZ

004326 104003
004330 012737 000011 006004
004336 004737 004470
004342 105777 174532
004348 100401
004354 104400

KWT44: SCOPE1
MOV #11,RATE ;SELECT MODE 0, 1 KHZ, GO
JSR PC,UPCNT
TSTB 3CSR
BMI .+4
ERROR ;ERROR, 1KHZ FAILED TO COUNT

1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286

:TEST CLOCK TO COUNT UP AT ALL FREQUENCIES
:TEST 100 HZ

004333 104003
004335 012737 000013 006004
004337 004737 004470
004339 105777 174506
004341 100401
004343 104400

KWT45: SCOPE1
MOV #13,RATE ;SELECT MODE 0, 100 HZ, GO
JSR PC,UPCNT
TSTB @CSR
BMI .+4
ERROR ;ERROR, 100HZ FAILED TO COUNT

:TEST CLOCK TO COUNT UP AT ALL FREQUENCIES
:LINE FREQ.

004346 104003
004350 012737 000017 006004
004356 004737 004470
004362 105777 174462
004366 100401
004370 104400

KWT46: SCOPE1
MOV #17,RATE ;SELECT MODE 0, LINE FREQ, GO
JSR PC,UPCNT
TSTB @CSR
BMI .+4
ERROR ;ERROR, LINE FREQUENCY FAILED TO COUNT

:REPEAT T12 WITH MODE 1
:TEST THAT CLOCK ENABLE DOES NOT CLEAR ON DONE
: 1 MHZ.

004372 104003
004374 012737 000403 006004
004402 004737 004470
004406 105777 174436
004412 100401
004414 104400

KWT47: SCOPE1
MOV #403,RATE ;SELECT MODE 1, 1MHZ., GO
JSR PC,UPCNT
TSTB @CSR
BMI .+4
ERROR ;ERROR, 1 MHZ. FAILED TO COUNT
; MODE 1

004416 032777 000001 174424
004424 001001
004426 104400

BIT #BIT0,@CSR
BNE .+4
ERROR ;MODE 1 CLEARED COUNTER ENABLE IN ERROR

: 100KHZ.

004430 104003
004432 012737 000405 006004
004440 004737 004470
004444 105777 174400
004450 100401
004452 104400
004454 032777 000001 174366
004462 001001
004464 104400
004466 000425

KWT48: SCOPE1
MOV #405,RATE ;MODE 1 100KHZ GO
JSR PC,UPCNT
TSTB @CSR
BMI .+4
ERROR ;ERROR, 100KHZ FAILED TO COUNT MODE 1
BIT #BIT0,@CSR
BNE .+4
ERROR ;MODE 1 CLEARED COUNTER ENABLE IN ERROR
BR KWT49

12987
12988
12989
12990
12991
12992
12993
12994
12995
12996
12997
12998
12999
13000
13001
13002
13003
13004
13005
13006
13007
13008
13009
13010
13011
13012
13013
13014
13015
13016
13017
13018
13019

004470 005077 174354
004474 012777 177776 174350
004502 013777 006004 174340
004510 005037 014456
004514 105777 174330
004520 100407
004522 027777 174322 174320
004530 062737 000001 014456
004536 001366
004540 000207

004542 104003
004544 000005
004546 042737 177437 001006
004554 001001
004556 000000
004560 022737 000340 001006
004566 001001
004570 000000

004572 013737 001006 014514
004600 162737 000040 014514
004606 013737 001006 014516
004614 013737 001006 014520
004622 062737 000040 014520

:SUBROUTINE TO LOAD -2 INTO THE CLOCK PRESET REGISTER
: AND LOAD CLOCK RATE INTO CLOCK STATUS REGISTER
: AND START THE CLOCK. WAIT A PERIOD OF TIME THEN EXIT

UPCNT: CLR @CSR ;CLEAR CLOCK STATUS
MOV #-2,@CSB ;MOVE -2 INT PRESET
MOV RATE,@CSR ;LOAD RATE AND ENABLE CLOCK
CLR DELAY ;DELAY
UPCNTA: TSTB @CSR
BMI UPCNTB
CMP @CSR,@CSR
ADD #1,DELAY
BNE UPCNTA
UPCNTB: RTS ;EXIT

:PRE-INTERRUPT SETUP

KWT49: SCOPE1
RESET
BIC #177437,CKBRL
BNE .+4 ;ERROR, BR LEVEL INDICATED FOR CLOCK WAS 0
HALT
CMP #340,CKBRL
BNE .+4 ;ERROR BR LEVEL FOR CLOCK WAS 7
HALT

MOV CKBRL,BRLEV1
SUB #40,BRLEV1
MOV CKBRL,BRLEV2
MOV CKBRL,BRLEV3
ADD #40,BRLEV3

```

1320
1321 ;TEST THAT THE SCHMITT TRIGGER INTERRUPTS AT LEVEL INDICATED -1
1322
1323 004630 104003 KWT50: SCOPE1
1324 004632 000005 RESET
1325 004634 012777 000340 174160 MOV #340, @PSW
1326 004642 005077 174202 CLR @CSR
1327 004646 012777 004712 174224 MOV #KWT50A, @CKV ;SET UP INTERRUPT RETURN VECTOR
1328 004654 012777 040000 174166 MOV #BIT14, @CSR ;ENABLE INTERRUPT
1329 004662 012700 000010 MOV #10, R0
1330 004666 013777 014514 174126 MOV BRLEV1, @PSW ;LOWER PRIOR.
1331 004674 052777 010000 174146 BIS #BIT12, @CSR
1332 004702 005300 DEC R0
1333 004704 001376 BNE .-2
1334 004706 104400 ERROR ;ERROR, CLOCK INTERRUPT FAILED TO OCCUR.
1335 004710 000415 BR KWT51 ;CHECK FOR PROPER BR LEVEL
1336 ;SHOULD BE LEVEL 6 (300)
1337
1338 ;SUBTEST, TEST THAT IF PRIORITY IS LOWERED AGAIN
1339 ;NO INTERRUPT SHOULD OCCUR
1340
1341 004712 022626 KWT50A: CMP (SP)+, (SP)+
1342 004714 012777 004740 174156 MOV #KWT50B, @CKV ;RESET THE VECTOR
1343 004722 012700 000010 MOV #10, R0
1344 004726 005077 174070 CLR @PSW
1345 004732 005300 DEC R0
1346 004734 001376 BNE .-2 ;DELAY
1347 004736 000402 BR KWT51
1348 004740 022626 KWT50B: CMP (SP)+, (SP)+
1349 004742 104400 ERROR ;ERROR, INT DONE B FAILED TO CLEAR INT REQ B

```

1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388

004744 104003
004746 000005
004750 012777 000340 174044
004756 012777 005042 174114
004764 012777 000340 174110
004772 005077 174052
004776 012777 177774 174046
005004 012777 000102 174036
005012 012700 000200
005016 013777 014514 173776
005024 052777 000001 174016
005032 005300
005034 001376
005036 104400
005040 000401

005042 022626

005044 104003
005046 012777 000340 173746
005054 005077 173770
005060 013777 001102 174012
005066 012777 000000 174006
005074 012777 040000 173746
005102 012700 000010
005106 052777 010000 173734
005114 005300
005116 001376
005120 012700 000100
005124 000005
005126 005077 173670
005132 005300
005134 001376

;TEST THAT THE CLOCK INTERRUPTS AT LEVEL INDICATED -1 (MODE 0)

KWT51: SCOPE1
RESET
MOV #340, @PSW
MOV #KWT51A, @CKV ;LOAD VECTOR
MOV #340, @CKVS
CLR @CSR
MOV #-4, @CSB ;LOAD PRESET
MOV #BIT6!BIT1, @CSR ;LOAD RATE AND INT ENABLE
MOV #200, RO ;SET UP DELAY
MOV BRLEV1, @PSW
BIS #BIT0, @CSR ;CLOCK
DEC RO
BNE -2 ;DELAY
ERROR ;ERROR, CLOCK FAILED TO INTERRUPT
BR KWT52

KWT51A: CMP (SP)+, (SP)+

;TEST THAT RESET CLEARS INT REQ B

KWT52: SCOPE1
MOV #340, @PSW
CLR @CSR ;CLEAR STATUS
MOV @CKVS, @CKV ;LOAD VECTOR
MOV #0, @CKVS
MOV #BIT14, @CSR ;LOAD INT ENABLE
MOV #10, RO
BIS #BIT12, @CSR ;FIRE ST
DEC RO
BNE -2 ;DELAY
MOV #100, RO
RESET
CLR @PSW
DEC RO
BNE -2

```

1389                                     ;TEST THAT THE CLOCK DOES NOT INTERRUPT AT LEVEL INDICATED
1390
1391 005136 104003                                     KWT53: SCOPE1
1392 005140 012777 000340 173654                   MOV #340, @PSW ;SET PROCESSOR PRIORITY 5
1393 005146 012777 005230 173724                   MOV #KWT53A, @CKV ;SET UP INTERRUPT RETURN STATUS
1394 005154 012777 000340 173720                   MOV #340, @CKVS
1395 005162 005077 173662                               CLR @CSR
1396 005166 012777 177774 173656                   MOV #-4, @CSB
1397 005174 012777 000102 173646                   MOV #102, @CSR ;ENABLE INTERRUPT
1398 005202 013777 014516 173612                   MOV BRLEV2, @PSW
1399 005210 052777 000001 173632                   BIS #BIT0, @CSR
1400 005216 012700 000100                               MOV #100, @R0
1401 005222 005300                               DEC @R0
1402 005224 001376                               BNE .-2
1403 005226 000401                               BR .+4
1404 005230 104400                                     KWT53A: ERROR ;ERROR, INTERRUPT OCCUR IN ERROR AT LEVEL INDICA
1405 005232 000005                                     RESET
1406
1407                                     ;TEST THAT THE CLOCK DOES NOT INTERRUPT AT LEVEL INDICATED +1
1408
1409 005234 104003                                     KWT54: SCOPE1
1410 005236 012777 000340 173556                   MOV #340, @PSW ;SET PROCESSOR PRIORITY 6
1411 005244 012777 005326 173626                   MOV #KWT54A, @CKV ;SET UP INTERRUPT RETURN
1412 005252 012777 000340 173622                   MOV #340, @CKVS
1413 005260 005077 173564                               CLR @CSR
1414 005264 012777 177774 173560                   MOV #-4, @CSB
1415 005272 012777 000102 173550                   MOV #102, @CSR ;INTERRUPT ENABLE
1416 005300 013777 014520 173514                   MOV BRLEV3, @PSW
1417 005306 052777 000001 173534                   BIS #BIT0, @CSR
1418 005314 012700 000100                               MOV #100, @R0
1419 005320 005300                               DEC @R0
1420 005322 001376                               BNE .-2 ;DELAY LONG ENOUGH TO ALLOW INTERRUPT
1421 005324 000401                               BR .+4
1422 005326 104400                                     KWT54A: ERROR ;ERROR, INT SHOULDN'T HAVE OCCURRED
1423 005330 005077 173514                               CLR @CSR ;STOP CLOCK
1424 005334 013777 001102 173536                   MOV @CKVS, @CKV
1425 005342 005077 173534                               CLR @CKVS
1426 005346 000005                               RESET
1427 005350 005077 173446                               CLR @PSW
1428 005354 052777 000100 173442                   BIS #BIT6, @TKS

```

```

1429
1430 ;TEST 1MHZ REPEATABILITY
1431
1432 005362 104003
1433 005364 004537 005604 KWT55: SCOPE1
1434 005370 000002 JSR R5, REPEAT ;TEST REPEATIBILITY
1435 005372 000002 ;CLOCK RATE
1436 005374 000001 ;CLOCK DEV.
1437 005376 000036 ;MIN. COUNT
1438 005400 104400 ;DELAY
1439 005402 003401 ;ERROR, FAILED TO REACH MIN. COUNT
1440 005404 104400 ;ERROR, CLOCK REPEATIBILITY >2
1441
1442 ;TEST 100 KHZ REPEATABILITY
1443
1444 005406 104003
1445 005410 004537 005604 KWT56: SCOPE1
1446 005414 000004 JSR R5, REPEAT ;TEST REPEATIBILITY
1447 005416 000002 4 ;CLOCK RATE
1448 005420 000001 2 ;CLOCK DEV.
1449 005422 000454 1 ;MIN. COUNT
1450 005424 104400 300. ;DELAY
1451 005426 003401 ;ERROR, FAILED TO REACH MIN. COUNT
1452 005430 104400 ;ERROR, CLOCK REPEATIBILITY >2
1453
1454 ;TEST 10 KHZ REPEATABILITY
1455
1456 005432 104003
1457 005434 004537 005604 KWT57A: SCOPE1
1458 005440 000006 JSR R5, REPEAT ;CLOCK RATE
1459 005442 000002 6 ;CLOCK DEV.
1460 005444 000001 2 ;MIN. COUNT
1461 005446 005670 1 ;DELAY
1462 005450 104400 3000. ;ERROR, FAILED TO REACH MIN. COUNT
1463 005452 003401 ;ERROR, CLOCK REPEATIBILITY >2
1464 005454 104400 ;ERROR, CLOCK REPEATIBILITY >2
1465
1466 ;TEST 1KHZ REPEATABILITY
1467
1468 005456 104003
1469 005460 004537 005604 KWT60: SCOPE1
1470 005464 000010 JSR R5, REPEAT ;CLOCK RATE
1471 005466 000002 10 ;CLOCK DEV.
1472 005470 000001 2 ;MIN. COUNT
1473 005472 072460 1 ;DELAY
1474 005474 104400 30000. ;ERROR, FAILED TO REACH MIN. COUNT
1475 005476 003401 ;ERROR, CLOCK REPEATIBILITY >2
1476 005500 104400 ;ERROR, CLOCK REPEATIBILITY >2
1477

```

```

1478
1479
1480
1481 005502 104003
1482 005504 004537 005604
1483 005510 000012
1484 005512 000002
1485 005514 000001
1486 005516 177777
1487 005520 104400
1488 005522 003401
1489 005524 104400
1490
1491
1492
1493 005526 104003
1494 005530 004537 005604
1495 005534 000016
1496 005536 000001
1497 005540 000001
1498 005542 000000
1499 005544 104400
1500 005546 003401
1501 005550 104400
1502
1503
1504
1505 005552 104003
1506 005554 000005
1507 005556 005737 000042
1508 005562 001402
1509 005564 000137 001752
1510
1511 005570 004737 013472
1512 005574 005237 001040
1513 005600 000137 002066

```

;TEST 100HZ REPEATIBILITY

```

KWT61: SCOPE1
        JSR      RS,REPEAT
        12
        2
        1
        -1
        ERROR
        BLE      .+4
        ERROR

```

```

;
;CLOCK RATE
;CLOCK DEV
;MIN. COUNT
;DELAY
;ERROR, FAILED TO REACH MIN. COUNT
;ERROR CLOCK REPEATABILITY >2

```

;TEST LINE REPEATABILITY

```

KWT62: SCOPE1
        JSR      RS,REPEAT
        16
        1
        1
        0
        ERROR
        BLE      .+4
        ERROR

```

```

;
;CLOCK RATE
;CLOCK DEV
;MIN. COUNT
;DELAY
;ERROR, FAILED TO REACH MIN COUNT
;ERROR, CLOCK REPEATABILITY >+1

```

;BELL ON PASS COMPLETE

```

TSTEND: SCOPE1
        RESET
        TST      @#42
        BEQ     HERE
        JMP     WHATB

```

;LOGICAL END OF THE TEST

```

HERE:   JSR      PC,BELL
        INC     PASSCT
        JMP     BEGIN

```

;REPORT END OF PASS

```

1514
1515      ;SUBROUTINE TO TEST THE CLOCK REPEATIBILITY
1516      ;FIRST CLEAR CLOCK STATUS AND PRESET BUFFER
1517      ; THEN ENABLE THE CLOCK TO COUNT AT A RATE.
1518      ; DECREMENT RO FOR SOME PERIOD OF TIME, WHEN RO = 0
1519      ; FIRE THE ST #2 AND CAUSE THE COUNTER TO LOAD THE PRESET REGISTER
1520      ; SAVE THE PRESET VALUE AND REPEAT THIS OPERATION AGAIN
1521      ; THEN COMPARE THE FIRST TIMED VALUE TO THE SECOND TIMED VALUE
1522      ; <MACHINE AND MEMORY TIMING NOT IMPORTANT>
1523      ; TO BE WITHIN THE VALUE SPECIFIED BY LOCATION CNTDEV
1524      ; IF GREATER THAN EXPECTED IT IS AN ERROR.
1525      ; ALSO TEST THAT THE COUNTER HAS REACHED A MIN. COUNT
1526
1527 005604 005077 173240 REPEAT: CLR      @CSR      ;STOP THE CLOCK
1528 005610 005077 173236      CLR      @CSB      ;CLEAR THE BUFFER
1529 005614 012537 006004      MOV      (R5)+,RATE  ;SET UP RATE
1530 005620 012537 006006      MOV      (R5)+,CNTDEV ;SET UP CNT. DEV
1531 005624 012537 006010      MOV      (R5)+,MINCNT  ;SET UP MIN COUNT
1532 005630 012537 006012      MOV      (R5)+,CKDLY  ;SAVE DELAY
1533 005634 004737 005724      JSR      PC,10$      ;DUMMY - TO CHARGE THE "CACHE"
1534 005640 004737 005724      JSR      PC,10$      ;ENABLE THE CLOCK
1535 005644 010037 006000      MOV      RO,$GDDAT   ;SAVE 1ST RESULTS
1536 005650 004737 005724      JSR      PC,10$      ;ENABLE THE CLOCK
1537 005654 010037 006002      MOV      RO,$BDDAT   ;SAVE THE 2ND RESULT
1538 005660 013700 006000      MOV      $GDDAT,RO   ;GET 1ST RESULT
1539 005664 163700 006002      SUB      $BDDAT,RO   ;SUBTRACT SECOND RESULT
1540 005670 100001
1541 005672 005400      BPL      3$
1542 005674 023737 006002 006010 3$:  NEG      RO
1543 005702 002004      CMP      $BDDAT,MINCNT ;COMPARE TO MIN. COUNT
1544 005704 013737 006010 014506      BGE      4$           ;BRANCK IF GREATER
1545 005712 000205      MOV      MINCNT,TEMP  ;LOAD TEMP FOR TYPE-OUT
1546 005714 005725      RTS      R5
1547 005716 020037 006006      TST      (R5)+        ;UPDATE THE STACK
1548 005722 000205      CMP      RO,CNTDEV    ;COMPARE TO DEVEATION
1549
1550 005724 013700 006012      MOV      CKDLY,RO     ;LOAD DELAY COUNT
1551 005730 052737 003000 006004      BIS      #BIT10!BIT9,RATE ;ENABLE ST MODE
1552 005736 013777 006004 173104      MOV      RATE,@CSR   ;LOAD RATE
1553 005744 005277 173100      INC      @CSR        ;ENABLE THE CLOCK
1554 005750 005300      1$:  DEC      RO          ;DELAY
1555 005752 001376      BNE      1$
1556 005754 052777 002000 173066      BIS      #BIT10,@CSR  ;MAINT ST FIRE
1557 005762 017700 173064      MOV      @CSB,RO     ;READ THE RESULT
1558 005766 005077 173056      CLR      @CSR        ;STOP CLOCK
1559 005772 005077 173054      CLR      @CSB        ;CLEAR BUFFER
1560 005776 000207      RTS      PC          ;EXIT
1561
1562 006000 000000      $GDDAT: 0
1563 006002 000000      $BDDAT: 0
1564 006004 000000      RATE: 0             ;CLOCK RATE
1565 006006 000000      CNTDEV: 0          ;CLOCK DEV.
1566 006010 000000      MINCNT: 0         ;MIN. COUNT
1567 006012 000000      CKDLY: 0

```

```

1568
1569
1570 ;*****
1571 ;DIGITAL INPUT-OUTPUT LOGIC TEST
1572 ;*****
1573 006014 000005 IOTEST: RESET
1574 006016 104000 PRINT
1575 006020 014055 MES3 ;IDENTIFY TEST
1576 006022 005037 001040 ITEST1: CLR PASSCT
1577 006026 012737 006076 013414 IOTSTA: MOV #DRT0+2, RETURN
1578 006034 013777 001040 172774 MOV PASSCT, @DISPLA
1579 006042 013706 014444 MOV STACK, SP
1580 006046 005077 172750 CLR @PSW
1581 006052 052777 000100 172744 BIS #BIT6, @TKS
1582
1583 ;TEST FOR NO BUSS ERRORS
1584
1585 006060 005077 172770 CLR @GRSTAT
1586 006064 005077 172766 CLR @GRDAI
1587 006070 005077 172764 CLR @GRDIO
1588
1589 006074 104003 DRT0: SCOPE1
1590 006076 012777 177777 172754 MOV #-1, @GRDIO ;ALL ONES TO REGISTER
1591 006104 022777 177777 172746 CMP #-1, @GRDIO
1592 006112 001401 BEQ .+4
1593 006114 104400 ERROR ;REG WILL NOT HOLD ONES
1594
1595 006116 104002 DRT1: SCOPE
1596 006120 012777 177777 172732 MOV #-1, @GRDIO
1597 006126 000005 RESET ;SET DATA TO ALL ONES
1598 006130 005777 172724 TST @GRDIO ;SHOULD CLEAR REGISTER
1599 006134 001401 BEQ .+4
1600 006136 104400 ERROR ;REG FAILED TO CLEAR
1601
1602 006140 052777 000100 172656 DRT2: BIS #BIT6, @TKS
1603 006146 104003 SCOPE1
1604 006150 012777 052525 172702 MOV #52525, @GRDIO
1605 006156 022777 052525 172674 CMP #52525, @GRDIO
1606 006164 001401 BEQ .+4
1607 006166 104400 ERROR ;DATA NOT=52525
1608
1609 006170 104002 DRT3: SCOPE
1610 006172 012777 125252 172660 MOV #125252, @GRDIO
1611 006200 022777 125252 172652 CMP #125252, @GRDIO
1612 006206 001401 BEQ .+4
1613 006210 104400 ERROR ;DATA NOT=125252
1614

```

1615	006212	104003			DRT4:	SCOPE1		
1616	006214	012777	177777	172636		MOV	#-1,@GRDIO	
1617	006222	105077	172632			CLRB	@GRDIO	;CLEAR LOW BYTE
1618	006226	017737	172626	014506		MOV	@GRDIO,TEMP	
1619	006234	022737	177400	014506		CMP	#177400,TEMP	
1620	006242	001401				BEQ	+.4	
1621	006244	104400				ERROR		;BYTE LOW FAILED TO CLEAR
1622								
1623	006246	104002			DRT5:	SCOPE		
1624	006250	012777	177777	172602		MOV	#-1,@GRDIO	
1625	006256	105077	172610			CLRB	@GRBHIO	;CLEAR HIGH BYTE
1626	006252	017737	172572	014506		MOV	@GRDIO,TEMP	
1627	006270	022737	000377	014506		CMP	#377,TEMP	
1628	006276	001401				BEQ	+.4	
1629	006300	104400				ERROR		;HIGH BYTE CLEAR FAILED
1630								
1631	006302	005037	014506			CLR	TEMP	
1632	006306	104002			DRT6:	SCOPE		
1633	006310	113777	014506	172542	DRT6A:	MOVW	TEMP,@GRDIO	;LOAD THE OUTPUT
1634	006316	123777	014506	172534		CMPB	TEMP,@GRDIO	
1635	006324	001402				BEQ	+.6	;BRANCH IF EQUAL
1636	006326	104400				ERROR		;ERROR, LOW BYTE HAS BAD DATA
1637	006330	000405				BR	DRT7	
1638	006332	105237	014506			INCB	TEMP	
1639	006336	001364				BNE	DRT6A	
1640								
1641	006340	005037	014506			CLR	TEMP	
1642	006344	104002			DRT7:	SCOPE		
1643	006346	113777	014506	172516	DRT7A:	MOVW	TEMP,@GRBHIO	;LOAD THE HIGH BYTE
1644	006354	123777	014506	172510		CMPB	TEMP,@GRBHIO	
1645	006362	001402				BEQ	+.6	;BRANCH IF EQUAL
1646	006364	104400				ERROR		;ERROR, HIGH BYTE IN ERROR
1647	006366	000403				BR	DRT8	
1648	006370	105237	014506			INCB	TEMP	
1649	006374	001364				BNE	DRT7A	

```

1650
1651 ;RELAY #1 TEST
1652
1653 006376 104003 DRT8: SCOPE1
1654 006400 012777 000001 172446 MOV #BIT0, @GRSTAT ;SET BIT 0
1655 006406 022777 000001 172440 CMP #BIT0, @GRSTAT ;TEST IT
1656 006414 001401 BEQ .+4
1657 006416 104400 ERROR ;ERROR, RELAY 1 FAILED TO SET
1658
1659 ;RELAY #2 TEST
1660
1661 006420 104002 DRT9: SCOPE
1662 006422 012777 000400 172424 MOV #BIT8, @GRSTAT ;SET RELAY 2
1663 006430 022777 000400 172416 CMP #BIT8, @GRSTAT
1664 006436 001401 BEQ .+4
1665 006440 104400 ERROR ;ERROR, RELAY 2 FAILED TO SET
1666
1667 ;TEST OUTPUT DATA ACCEPT FLAG
1668
1669 006442 104002 DRT10: SCOPE
1670 006444 012777 100000 172402 MOV #BIT15, @GRSTAT ;SET BIT 15
1671 006452 022777 100000 172374 CMP #BIT15, @GRSTAT
1672 006460 001401 BEQ .+4
1673 006462 104400 ERROR ;ERROR, BIT 15 FAILED TO SET
1674
1675 ;TEST OUTPUT INTERRUPT ENABLE
1676
1677 006464 104002 DRT11: SCOPE
1678 006466 012777 000340 172326 MOV #340, @PSW ;RAISE PRIORITY
1679 006474 012777 040000 172352 MOV #BIT14, @GRSTAT ;LOAD BIT 14
1680 006502 022777 040000 172344 CMP #BIT14, @GRSTAT
1681 006510 001401 BEQ .+4
1682 006512 104400 ERROR ;ERROR BIT 14 FAILED TO SET
1683

```

```

1684
1685 ;TEST INPUT DATA READY FLAG
1686
1687 006514 104002 DRT12: SCOPE
1688 006516 012777 000200 172330 MOV #BIT7, @GRSTAT ;SET BIT 7
1689 006524 022777 000200 172322 CMP #BIT7, @GRSTAT
1690 006532 001401 BEQ .+4
1691 006534 104400 ERROR ;ERROR, BIT 7 FAILED TO SET
1692
1693 ;TEST INPUT INTERRUPT ENABLE
1694
1695 006536 104002 DRT13: SCOPE
1696 006540 012777 000340 172254 MOV #340, @PSW
1697 006546 012777 000100 172300 MOV #BIT6, @GRSTAT ;SET BIT 6
1698 006554 022777 000100 172272 CMP #BIT6, @GRSTAT
1699 006562 001401 BEQ .+4
1700 006564 104400 ERROR ;ERROR, BIT 6 FAILED TO SET
1701
1702 ;TEST THAT RESET CLEARS THE DIGITAL STATUS REGISTER
1703
1704 006566 104002 DRT14: SCOPE
1705 006570 012777 000340 172224 MOV #340, @PSW
1706 006576 012777 140701 172250 MOV #BIT15!BIT14!BIT8!BIT7!BIT6!BIT0, @GRSTAT
1707 006604 000005 RESET
1708 006606 005777 172242 TST @GRSTAT
1709 006612 001401 BEQ .+4
1710 006614 104400 ERROR ;ERROR, RESET FAILED TO CLEAR DIGITAL STATUS REG
1711
1712 006616 005077 172200 CLR @PSW
1713 006622 052777 000100 172174 BIS #BIT6, @TKS

```

```

:TEST:EXTERNAL TRANSFERS - CABLE MUST BE CONNECTED
DRT15: SCOPE1
006630 104003          BIT      #BIT6,3SWR      :TEST SWITCH BIT
006632 032777 000100 172174 BEQ      .+6        :BRANCH IF DOWN
006640 001402          JMP      DRT21      :BYPASS SOME TEST USING THE EXTERNAL CABLE
006642 000137 007044          MOV      #0,3GRDIO  :LOAD THE OUTPUT
006646 012777 000000 172204 CMP      #0,3GRDAI  :READ THE INPUT
006654 022777 000000 172174 BEQ      .+4        :BRANCH IF EQUAL
006662 001401          ERROR     :ERROR, INPUT DID NOT EQUAL THE OUTPUT REG.
006664 104400

006666 104002          DRT16: SCOPE1
006670 012777 177777 172162 MOV      #-1,3GRDIO  :LOAD THE OUTPUT
006672 022777 177777 172162 CMP      #-1,3GRDAI  :READ THE INPUT
006704 001401          BEQ      .+4        :BRANCH IF EQUAL
006706 104400          ERROR     :ERROR, INPUT DID NOT EQUAL THE OUTPUT.
:IS THIS REALLY A LPS-11-DR ?? OR IS IT AN
:LPS-11-DRA ?? IF DRA USE MD-11-DZLPI TEST

006710 104002          DRT17: SCOPE1
006712 012777 052525 172140 MOV      #52525,3GRDIO :LOAD THE OUTPUT
006720 022777 052525 172130 CMP      #52525,3GRDAI :READ THE INPUT
006726 001401          BEQ      .+4        :BRANCH IF EQUAL
006730 104400          ERROR     :ERROR, INPUT DID NOT EQUAL OUTPUT

006732 104002          DRT18: SCOPE1
006734 012777 025252 172116 MOV      #25252,3GRDIO :LOAD THE OUTPUT
006742 022777 025252 172106 CMP      #25252,3GRDAI :READ THE INPUT
006750 001401          BEQ      .+4        :BRANCH IF EQUAL
006752 104400          ERROR     :ERROR, INPUT DID NOT EQUAL OUTPUT

:TEST THAT WHEN OUTPUTTING THE INPUT DATA READY FLAG SETS
DRT19: SCOPE1
006754 104003          CLR      3GRSTAT
006756 005077 172072          MOV      #0,3GRDIO  :OUTPUT 0
006762 012777 000000 172070 CMP      #0,#0      :DELAY
006770 022727 000000 000000 TSTB    3GRSTAT    :TEST BIT 7
006776 105777 172052          BMI     .+4
007002 100401          ERROR     :INPUT DATA READY FLAG FAILED TO SET
007004 104400

:TEST THAT WHEN THE INPUT BUFFER IS READ THE OUTPUT FLAG IS SET
DRT20: SCOPE1
007006 104002          CLR      3GRSTAT
007010 005077 172040          MOV      #0,3GRDIO
007014 012777 000000 172036 CMP      #0,#0
007022 022727 000000 000000 MOV      3GRDAI,R0  :READ INPUT
007030 017700 172022          TST     3GRSTAT    :TEST BIT 15
007034 005777 172014          BMI     .+4
007040 100401          ERROR     :INPUT DATA READY FLAG FAILED TO SET
007042 104400

```

```

1764 :TEST THAT THE DIGITAL I/O DOES NOT INTERRUPT
1765 :SET INPUT-OUTPUT FLAGS BUT DO NOT ENABLE INTERRUPTS
1766
1767 007044 104002 DRT21: SCOPE
1768 007046 000000 RESET
1769 007050 012777 000340 171744 MOV #340,3PSW :RAISE PRIORITY
1770 007056 005077 171772 CLR 3GRSTAT
1771 007062 012777 007132 172014 MOV #DRT21A,3GRIVA :SET UP INTERRUPT INPUT VECTOR
1772 007070 012777 007136 172012 MOV #DRT21B,3GRIVB :SET UP INTERRUPT OUTPUT VECTOR
1773 007076 005077 171720 CLR 3PSW
1774 007102 012777 000000 171750 MOV #0,3GRDIO :OUTPUT
1775 007110 017700 171742 MOV 3GRDAI,RO :INPUT
1776 007114 000240 NOP
1777 007116 000240 NOP
1778 007120 000240 NOP
1779 007122 000240 NOP
1780 007124 005077 171724 CLR 3GRSTAT :CLEAR STATUS
1781 007130 000406 BR DRT22
1782
1783 007132 104400 DRT21A: ERROR :ERROR, DIGITAL INPUT INTERRUPTED
1784 007134 000002 RTI
1785
1786 007136 104400 DRT21B: ERROR :ERROR, DIGITAL OUTPUT INTERRUPTED
1787 007140 000002 RTI

```

```

17898
17899
17900
17901
17902
17903 007142 005077 171706
17904 007146 104003
17905 007150 000005
17906 007152 005077 171676
17907 007156 012777 000340 171636
17908 007164 012777 007232 171712
17909 007172 012777 000340 171706
17910 007200 013777 001112 171702
18000 007206 012777 000000 171676
18001 007214 005077 171602
18002 007220 052777 000040 171626
18003 007226 000240
18004 007230 000000
18005
18006
18007
18008
18009 007232 013777 001106 171644
18010 007240 005077 171642
18011 007244 005077 171552
18012 007250 000240
18013 007252 000240
18014
18015
18016
18017
18018
18019 007254 005077 171574
18020 007260 104003
18021 007262 000005
18022 007264 005077 171564
18023 007270 012777 000340 171524
18024 007276 013777 001106 171600
18025 007304 012777 000000 171574
18026 007312 012777 007344 171570
18027 007320 012777 000340 171564
18028 007326 005077 171470
18029 007332 052777 020000 171514
18030 007340 000240
18031 007342 000000
18032
18033
18034
18035
18036 007344 013777 001112 171536
18037 007352 005077 171534
18038 007356 005077 171440
18039 007362 000240
18040 007364 000240
18041

```

:TEST THAT THE INPUT CAN INTERRUPT
:USE MAINT. BIT

```

DRT22: CLR      3GRSTAT
        SCOPE1
        RESET
        CLR      3GRSTAT           :CLEAR STATUS
        MOV      340,3PSW         :RAISE PRIOR
        MOV      3DRT22A,3GRIVA   :LOAD RETURN VECTOR
        MOV      340,3GRIVSA
        MOV      GRIVSB,3GRIVB    :LOAD OUTPUT VECTOR
        MOV      30,3GRIVSB
        CLR      3PSW
        BIS      3BITS,3GRSTAT    :MAINT. INT C
        NOP
        HALT                      :ERROR, INPUT FAILED TO INTERRUPT

```

:SUB-TEST, TEST THAT IF PSW IS LOWERED AGAIN NO INTERRUPT WILL OCCUR
: IF INTERRUPT OCCURS 'INT DONE C' FAILED TO CLEAR INT C FLOP

```

DRT22A: MOV     GRIVSA,3GRIVA     :LOAD INPUT VECTOR
        CLR     3GRIVSA
        CLR     3PSW              :LOWER PRIOR.
        NOP
        NOP

```

:TEST THAT THE OUTPUT DOES INTERRUPT
:USE MAINT. BIT

```

DRT25: CLR      3GRSTAT           :CLEAR STATUS
        SCOPE1
        RESET
        CLR      3GRSTAT           :CLEAR STATUS
        MOV      340,3PSW
        MOV      GRIVSA,3GRIVA    :LOAD INPUT VECTOR
        MOV      30,3GRIVSA
        MOV      3DRT25A,3GRIVB  :LOAD OUTPUT VECTOR
        MOV      340,3GRIVSB
        CLR      3PSW
        BIS      3BIT13,3GRSTAT   :MAINT. INTERRUPT
        NOP
        HALT                      :OUTPUT FAILED TO INTERRUPT

```

:SUB-TEST, TEST THAT IF PSW IS LOWERED AGAIN, NO INTERRUPT WILL OCCUR
: IF IT DOES, 'INT DONE D HIGH' FAILED TO CLEAR 'INT D' FLOP

```

DRT25A: MOV     GRIVSB,3GRIVB    :LOAD OUTPUT VECTOR
        CLR     3GRIVSB
        CLR     3PSW
        NOP
        NOP

```

```

1842
1843 007366 104003
1844 007370 042737 177437 001010
1845 007376 001001
1846 007400 000000
1847 007402 022737 000340 001010
1848 007410 001001
1849 007412 000000
1850 007414 013737 001010 014514
1851 007422 162737 000040 014514
1852 007430 013737 001010 014516
1853 007436 013737 001010 014520
1854 007444 062737 000040 014520
1855
1856
1857 007452 104002
1858 007454 000005
1859 007456 012777 000340 171336
1860 007464 005077 171364
1861 007470 012777 007532 171406
1862 007476 012777 007532 171404
1863 007504 013777 014514 171310
1864 007512 052777 020040 171334
1865 007520 000240
1866 007522 000240
1867 007524 000240
1868 007526 000240
1869 007530 104400
1870
1871 007532 022626
1872
1873
1874 007534 104003
1875 007536 000005
1876 007540 032777 000100 171266
1877 007546 001402
1878 007550 000137 007654
1879 007554 012777 000340 171240
1880 007562 005077 171266
1881 007566 012777 007652 171310
1882 007574 012777 007652 171306
1883 007602 013777 014514 171212
1884 007610 012777 040100 171236
1885 007616 012777 000000 171234
1886 007624 017700 171226
1887 007630 000240
1888 007632 000240
1889 007634 000240
1890 007636 042777 040100 171210
1891 007644 000240
1892 007646 104400
1893 007650 000401
1894 007652 022626
1895
1896
1897

```

```

:PRE INTERRUPT SETUP
DRT29: SCOPE1
      BIC      #177437,DIOBRL
      BNE     .+4
      HALT
      CMP     #340,DIOBRL
      BNE     .+4
      HALT
      MOV     DIOBRL, BRLEV1
      SUB     #40, BRLEV1
      MOV     DIOBRL, BRLEV2
      MOV     DIOBRL, BRLEV3
      ADD     #40, BRLEV3
;BR LEVEL INDICATED FOR DIGITAL I/O WAS 0
;BR LEVEL INDICATED FOR DIGITAL I/O WAS 7

```

```

:TEST FOR INTERRUPT FROM DIGITAL I/O ON LEVEL INDICATED -1
DRT29: SCOPE
      RESET
      MOV     #340, JPSW
      CLR     JGRSTAT
      MOV     #DRT29A, JGRIVA
      MOV     #DRT29A, JGRIVB
      MOV     BRLEV1, JPSW
      BIS     #BIT13!BITS, JGRSTAT
      NOP
      NOP
      NOP
      NOP
      ERROR
;RAISE PSW
;CLEAR ENABLES
;SET UP VECTORS
;LOWER PSW
;GENERATE MAINT. INTERRUPT
;ERROR, NO INTERRUPT FROM DIGITAL I/O

```

```

DRT29A: CMP (SP)+, (SP)+

```

```

:TEST FOR INTERRUPT FROM DIGITAL I/O ON LEVEL INDICATED -1
DRT29B: SCOPE1
      RESET
      BIT     #BIT6, JPSW
      BEQ    .+6
      JMP    DRT30
      MOV     #340, JPSW
      CLR     JGRSTAT
      MOV     #DRT29C, JGRIVA
      MOV     #DRT29C, JGRIVB
      MOV     BRLEV1, JPSW
      MOV     #BIT14!BIT6, JGRSTAT
      MOV     #0, JGRDIO
      MOV     JGRDAI, RO
      NOP
      NOP
      NOP
      BIC     #BIT14!BIT6, JGRSTAT
      NOP
      ERROR
      BR     DRT30
DRT29C: CMP (SP)+, (SP)+
;TEST SWITCH
;BYPASS THIS TEST
;IN CASE OF INTERRUPT
;OUTPUT
;INPUT
;LET INTERRUPT OCCUR
;ERROR, NO DIGITAL I/O INTERRUPT OCCURED

```

```

:TEST FOR NO INTERRUPT FROM DIGITAL I/O ON LEVEL INDICATED

```

```

1898 007654 104003          DRT30: SCOPE1
1899 007656 000005          RESET
1900 007660 012777 000340 171134  MOV      #340, JPSW
1901 007666 005077 171162          CLR      JGRSTAT      ;CLEAR STATUS
1902 007672 012777 007752 171204  MOV      #DRT30A, JGRIVA ;IN CASE OF INTERRUPT
1903 007700 012777 007752 171202  MOV      #DRT30A, JGRIVB ;
1904 007706 013777 014516 171106  MOV      BRLEV2, JPSW
1905 007714 012777 040100 171132  MOV      #BIT14!BIT6, JGRSTAT ;LOAD INT. ENABLE
1906 007722 012777 000000 171130  MOV      #0, JGRDIO      ;OUTPUT
1907 007730 017700 171122          MOV      JGRDAI, RO      ;INPUT
1908 007734 000240          NOP
1909 007736 000240          NOP
1910 007740 000240          NOP
1911 007742 042777 040100 171104  BIC      #BIT14!BIT6, JGRSTAT ;DON'T LEAVE IT SET
1912 007750 000401          BR      .+4              ;WITH NO INTERRUPT, BRANCH OVER HALT
1913 007752 104400          DRT30A: ERROR           ;ERROR, INTERRUPT OCCURED AT BR INDICATED
1914
1915 007754 104003          SCOPE1
1916 007756 000005          RESET
1917 007760 013777 001106 171116  MOV      GRIVSA, JGRIVA  ;FOR FALSE INTERRUPT
1918 007766 005077 171114          CLR      JGRIVSA
1919 007772 013777 001112 171110  MOV      GRIVSB, JGRIVB
1920 010000 005077 171106          CLR      JGRIVSB
1921
1922          ;LOGICAL END OF 'DIGITAL I/O LOGIC TEST'
1923
1924 010004 000005          RESET
1925 010006 005737 000042          TST     #42
1926 010012 001402          BEQ     .+6
1927 010014 000137 001764          JMP     WHATC
1928 010020 004737 013472          JSR     7, BELL          ;REPORT END OF PASS
1929 010024 005237 001040          INC     PASSCT
1930 010030 000137 006026          JMP     IOTSTA

```

```

1931 :*****
1932 : LPSVC POINT PLOT SCOPE CONTROL
1933 :*****
1934
1935 010034 000005 VCTEST: RESET
1936 010036 104000 PRINT
1937 010040 014110 MESS ;IDENTIFY TEST
1938 010042 005037 001040 VTEST1: CLR PASSCT
1939
1940 010046 013706 014444 VCTSTO: MOV STACK,SP
1941 010052 013777 001040 170756 MOV PASSCT,DISPLA
1942 010050 012737 010116 013414 MOV #VCTO+2,RETURN
1943 010066 005077 170730 CLR APSW
1944 010072 052777 000100 170724 BIS #BIT6,ATKS
1945
1946 ;TEST FOR NO BUSS ERRORS
1947
1948 010100 005077 170756 CLR SVCSTAT
1949 010104 005077 170754 CLR SVCXREG
1950 010110 005077 170752 CLR SVCYREG
1951
1952 ;TEST THAT RESET SETS READY BIT
1953
1954 010114 104002 VCTO: SCOPE
1955 010116 000005 RESET
1956 010120 105777 170736 TSTB SVCSTAT ;TEST READY
1957 010124 100401 BMI .+4
1958 010126 104400 ERROR ;ERROR, VCSTAT NOT = 200
1959
1960 ;TEST THAT FAST INTENSIFY (BIT 1) CAN BE SET AND CLEARED
1961
1962 010130 104003 VCT1: SCOPE1
1963 010132 052777 000100 170664 BIS #BIT6,ATKS
1964 010140 012777 000002 170714 MOV #BIT1,SVCSTAT ;LOAD DISPLAY STATUS
1965 010146 022777 000202 170706 CMP #BIT7!BIT1,SVCSTAT ;TEST STATUS
1966 010154 001401 BEQ .+4
1967 010156 104400 ERROR ;ERROR, VC STATUS NOT = 202
1968
1969 ;TEST THAT MODE (BIT 2) CAN BE SET AND CLEARED
1970
1971 010160 104002 VCT2: SCOPE
1972 010162 012777 000004 170672 MOV #BIT2,SVCSTAT ;LOAD DISPLAY STATUS
1973 010170 022777 000204 170664 CMP #BIT7!BIT2,SVCSTAT ;TEST STATUS
1974 010176 001401 BEQ .+4
1975 010200 104400 ERROR ;ERROR, VC STATUS NOT = 204
1976
1977 ;TEST THAT MODE (BIT 3) CAN BE SET AND CLEARED
1978
1979 010202 104002 VCT3: SCOPE
1980 010204 012777 000010 170650 MOV #BIT3,SVCSTAT ;LOAD DISPLAY STATUS
1981 010212 022777 000210 170642 CMP #BIT7!BIT3,SVCSTAT ;TEST STATUS
1982 010220 001401 BEQ .+4
1983 010222 104400 ERROR ;ERROR, VC STATUS NOT = 210
1984

```

```

1985
1986 ;TEST THAT EXT DEL (BIT 4) CAN BE SET AND CLEARED
1987
1988 VCT4: SCOPE
1989 010224 104002 MOV #BIT4,AVCSTAT ;LOAD DISPLAY STATUS
1990 010226 012777 000020 170626 CMP #BIT7!BIT4,AVCSTAT ;TEST STATUS
1991 010234 022777 000220 170620 BEQ .+4
1992 010242 001401 ;ERROR, VC STATUS NOT = 220
1993 010244 104400 ERROR
1994 ;TEST THAT INTERRUPT ENABLE (BIT 6) CAN BE SET AND CLEARED
1995
1996 VCT5: SCOPE
1997 010246 104002 MOV #BIT6,AVCSTAT ;LOAD DISPLAY STATUS
1998 010250 012777 000100 170604 CMP #BIT7!BIT6,AVCSTAT ;TEST STATUS
1999 010256 022777 000300 170576 BEQ .+4
2000 010264 001401 ;ERROR, VC STATUS NOT = 300
2001 010266 104400 ERROR
2002 ;TEST THAT CHANNEL (BIT 9) CAN BE SET AND CLEARED
2003
2004 VCT6: SCOPE
2005 010270 104002 MOV #BIT9,AVCSTAT ;LOAD DISPLAY STATUS
2006 010272 012777 001000 170562 CMP #BIT9!BIT7,AVCSTAT ;TEST STATUS
2007 010300 022777 001200 170554 BEQ .+4
2008 010306 001401 ;ERROR, VC STATUS NOT = 1200
2009 010310 104400 ERROR
2010 ;TEST THAT STORE (BIT 10) CAN BE SET AND CLEARED
2011
2012 VCT7: SCOPE
2013 010312 104002 MOV #BIT10,AVCSTAT ;LOAD DISPLAY STATUS
2014 010314 012777 002000 170540 CMP #BIT10!BIT7,AVCSTAT ;TEST STATUS
2015 010322 022777 002200 170532 BEQ .+4
2016 010330 001401 ;ERROR, VC STATUS NOT = 2200
2017 010332 104400 ERROR
2018 ;TEST THAT WRITE THRU (BIT 11) CAN BE SET AND CLEARED
2019
2020 VCT8: SCOPE
2021 010334 104002 MOV #BIT11,AVCSTAT ;LOAD DISPLAY STATUS
2022 010336 012777 004000 170516 CMP #BIT11!BIT7,AVCSTAT ;TEST STATUS
2023 010344 022777 004200 170510 BEQ .+4
2024 010352 001401 ;ERROR, VC STATUS NOT = 4200
2025 010354 104400 ERROR

```

```

2026
2027 ;TEST THAT THE X REGISTER (BITS 0-11) CAN BE SET AND CLEARED
2028
2029 010356 104002 VCT11: SCOPE
2030 010360 012777 002525 170476 MOV #2525,AVCXREG ;LOAD X REGISTER
2031 010366 022777 002525 170470 CMP #2525,AVCXREG
2032 010374 001401 BEQ .+4
2033 010376 104400 ERROR ;ERROR, VC X REGISTER NOT = 2525
2034
2035 010400 104002 VCT12: SCOPE
2036 010402 012777 005252 170454 MOV #5252,AVCXREG ;LOAD X REGISTER
2037 010410 022777 005252 170446 CMP #5252,AVCXREG
2038 010416 001401 BEQ .+4
2039 010420 104400 ERROR ;ERROR, VC X REGISTER NOT = 5252
2040
2041 ;TEST THAT THE Y REGISTER (BITS 0-11) CAN BE SET AND CLEARED
2042
2043 010422 104002 VCT15: SCOPE
2044 010424 012777 002525 170434 MOV #2525,AVCYREG ;LOAD Y REGISTER
2045 010432 022777 002525 170426 CMP #2525,AVCYREG
2046 010440 001401 BEQ .+4
2047 010442 104400 ERROR ;ERROR, VC Y REGISTER NOT = 2525
2048
2049 010444 104002 VCT16: SCOPE
2050 010446 012777 005252 170412 MOV #5252,AVCYREG ;LOAD Y REGISTER
2051 010454 022777 005252 170404 CMP #5252,AVCYREG
2052 010462 001401 BEQ .+4
2053 010464 104400 ERROR ;ERROR, VC Y REGISTER NOT = 5252
2054
2055 ;TEST THAT THE X-Y REGISTER CAN HOLD DIFFERENT DATA
2056
2057 010466 104002 VCT17: SCOPE
2058 010470 012777 001234 170366 MOV #1234,AVCXREG ;LOAD X REGISTER
2059 010476 012777 004321 170362 MOV #4321,AVCYREG ;LOAD Y REGISTER
2060 010504 022777 001234 170352 CMP #1234,AVCXREG
2061 010512 001401 BEQ .+4
2062 010514 104400 ERROR ;ERROR, SELECT X REGISTER INCORRECTLY
2063
2064 010516 022777 004321 170342 CMP #4321,AVCYREG
2065 010524 001401 BEQ .+4
2066 010526 104400 ERROR ;ERROR, SELECTED Y REGISTER INCORRECTLY
2067

```

```

2068
2069 ;TEST THAT WHEN INTENSIFY BIT IS SET THAT THE READY BIT CLEARS
2070 ; AND THEN SETS AFTER A DELAY
2071
2072 010530 104002 SCOPE
2073 010532 012700 000100 MOV #100,RO
2074 010536 012777 000001 170316 MOV #BIT0,AVCSTAT ;INTENSIFY
2075 010544 105777 170312 TSTB AVCSTAT ;TEST READY
2076 010550 100001 BPL .+4 ;IS IT CLEARED ?
2077 010552 104400 ERROR ;READY FAILED TO CLEAR
2078 010554 105777 170302 VCTST2: TSTB AVCSTAT ;TEST READY
2079 010560 100403 BMI VCTST3 ;SET EXIT
2080 010562 005300 DEC RO ;DELAY
2081 010564 001373 BNE VCTST2
2082 010566 104400 ERROR ;READY FAILED TO SET AFTER A DELAY
2083
2084 ;TEST THAT WHEN COLOR IS CHANGED READY CLEARS AND THEN SETS AFTER A DELAY
2085
2086 010570 104003 VCTST3: SCOPE1
2087 010572 012700 000400 MOV #400,RO ;SET UP A DELAY
2088 010576 012777 000400 170256 MOV #BIT8,AVCSTAT ;CHANGE TO RED
2089 010604 032777 000400 170250 BIT #BIT8,AVCSTAT ;TEST COLOR BIT
2090 010612 001001 BNE .+4 ;IS IT SET ?
2091 010614 104400 ERROR ;ERROR, COLOR BIT FAILED TO SET
2092 010616 105777 170240 TSTB AVCSTAT ;TEST READY
2093 010622 100001 BPL .+4 ;IS IT CLEARED ?
2094 010624 104400 ERROR ;ERROR, VC STATUS READY FAILED TO CLEAR
2095 010626 105777 170230 VCTST4: TSTB AVCSTAT ;TEST READY
2096 010632 100403 BMI VCTST5 ;IS IT SET ?
2097 010634 005300 DEC RO ;NO, DELAY
2098 010636 001373 BNE VCTST4
2099 010640 104400 ERROR ;ERROR, READY FAILED TO SET AFTER A COLOR CHANGE
2100
2101 ;TEST THAT WHEN COLOR IS CHANGED READY CLEARS AND THEN SETS AFTER A DELAY
2102
2103 010642 104001 VCTST5: SCOPE0
2104 010644 012700 020000 MOV #20000,RO ;SET UP A DELAY
2105 010650 012777 000000 170204 MOV #0,AVCSTAT ;CHANGE TO GREEN
2106 010656 032777 000400 170176 BIT #BIT8,AVCSTAT ;TEST COLOR BIT
2107 010664 001401 BEQ .+4
2108 010666 104400 ERROR ;ERROR, COLOR BIT FAILED TO CLEAR
2109 010670 105777 170166 TSTB AVCSTAT ;TEST READY
2110 010674 100001 BPL .+4 ;DID IT CLEAR
2111 010676 104400 ERROR ;ERROR, READY FAILED TO CLEAR
2112 010700 105777 170156 VCTST6: TSTB AVCSTAT ;TEST READY
2113 010704 100403 BMI VCTST7
2114 010706 005300 DEC RO
2115 010710 001373 BNE VCTST6
2116 010712 104400 ERROR ;ERROR, READY FAILED TO SET AFTER A COLOR CHANGE

```

```

2117
2118          ;TEST THAT MODE 1 (INTENSIFY ON X)
2119          ;CLEARS THE READY FLAG AND THEN SETS IT
2120
2121 010714 104001          VCTST7: SCOPED
2122 010716 012700 000100          MOV      #100,RO          ;SET UP DELAY
2123 010722 012777 000004 170132          MOV      #BIT2,@VCSTAT  ;LOAD MODE 1
2124 010730 105777 170126          TSTB    @VCSTAT        ;TEST READY
2125 010734 100401          BMI      .+4
2126 010736 104400          ERROR          ;ERROR, MODE 1 SHOULD NOT CLEAR READY UNTIL X IS
2127
2128 010740 005077 170120          CLR      @VCXREG        ;ADDRESS X AXIS
2129 010744 105777 170112          TSTB    @VCSTAT        ;TEST READY
2130 010750 100001          BPL      .+4
2131 010752 104400          ERROR          ;ERROR, MODE 1 LOAD X FAILED TO CLEAR READY FLAG
2132
2133 010754 105777 170102          VCTST8: TSTB    @VCSTAT  ;TEST READY
2134 010760 100403          BMI      VCTST9        ;SET, NEXT TEST
2135 010762 005300          DEC      RO            ;DELAY
2136 010764 001373          BNE      VCTST8        ;TEST READY AGAIN
2137 010766 104400          ERROR          ;ERROR, READY FAILED TO SET
2138          ; AFTER MODE 1 OPERATION
2139
2140          ;TEST THAT MODE 2 (INTENSIFY ON Y)
2141          ;CLEARS THE READY FLAG AND THEN SETS IT
2142
2143 010770 104003          VCTST9: SCOPED
2144 010772 012700 000100          MOV      #100,RO          ;SET UP DELAY
2145 010776 012777 000010 170056          MOV      #BIT3,@VCSTAT  ;LOAD MODE 2
2146 011004 105777 170052          TSTB    @VCSTAT        ;TEST READY
2147 011010 100401          BMI      .+4
2148 011012 104400          ERROR          ;ERROR, MODE 2 SHOULD NOT CLEAR READY UNTIL Y IS
2149
2150 011014 005077 170046          CLR      @VCYREG        ;ADDRESS Y AXIS
2151 011020 105777 170036          TSTB    @VCSTAT        ;TEST READY
2152 011024 100001          BPL      .+4
2153 011026 104400          ERROR          ;ERROR, MODE 2 LOAD Y FAILED TO CLEAR READY FLAG
2154
2155 011030 105777 170026          VCTS10: TSTB    @VCSTAT  ;TEST READY
2156 011034 100403          BMI      VCTS11        ;SET, NEXT TEST
2157 011036 005300          DEC      RO            ;DELAY
2158 011040 001373          BNE      VCTS10        ;TEST READY AGAIN
2159 011042 104400          ERROR          ;ERROR, READY FAILED TO SET
2160          ; AFTER MODE 2 OPERATION
2161

```

```

2162          ;TEST THAT WHEN ERASE BIT IS SET THAT THE READY BIT
2163          ;CLEARS AND THEN SETS AFTER A DELAY.
2164
2165 011044 104003          VCTS11: SCOPE1
2166 011046 032777 000040 167760 BIT      #BITS, @SWR
2167 011054 001424          BEQ      AVCT12
2168 011056 012700 000002          MOV      #2, RO
2169 011062 005037 014506          CLR      TEMP
2170 011066 052777 012000 167766 BIS      #BIT12!BIT10, @VCSTAT ;CLEAR DELAY
2171 011074 105777 167762          TSTB    @VCSTAT ;SET BIT 12
2172 011100 100001          BPL      .+4 ;TEST THAT READY CLEARS
2173 011102 104400          ERROR ;ERROR, READY FAILED TO RESET
2174          ; UPON SETTING ERASE BIT
2175
2176 011104 105777 167752          TST11A: TSTB    @VCSTAT ;TEST FOR READY
2177 011110 100406          BMI      AVCT12 ;BRANCH IF SET
2178 011112 005337 014506          DEC      TEMP ;DELAY
2179 011116 001372          BNE      TST11A ;BRANCH IF NOT READY
2180 011120 005300          DEC      RO ;DECREMENT COUNTER
2181 011122 001370          BNE      TST11A ;BRANCH IF NOT COMPLETED
2182 011124 104400          ERROR ;ERROR, ERASE CLEARED READY AND FAILED
2183          ; TO SET READY AFTER A DELAY
2184
2185          ;TEST THAT THERE IS A TIME DIFFERENCE BETWEEN FAST INTENSIFY
2186          ; AND NOT FAST INTENSIFY
2187
2188 011126 104001          AVCT12: SCOPED
2189 011130 005000          CLR      RO ;CLEAR RO
2190 011132 012777 000001 167722 MOV      #BIT0, @VCSTAT ;START DISPLAY NOT FAST INTEN.
2191 011140 105777 167716          VCTS12: TSTB    @VCSTAT ;TEST READY
2192 011144 100403          BMI      VCTS13 ;IT IS SET
2193 011146 005200          INC      RO ;INCREMENT RO
2194 011150 001373          BNE      VCTS12 ;TEST AGAIN
2195 011152 104400          ERROR ;ERROR, READY FAILED TO SET
2196
2197 011154 005001          VCTS13: CLR      R1 ;CLEAR R1
2198 011156 012777 000002 167676 MOV      #BIT1, @VCSTAT ;FAST INTEN.
2199 011164 052777 000001 167670 BIS      #BIT0, @VCSTAT ;START DISPLAY
2200 011172 105777 167664          VCTS14: TSTB    @VCSTAT ;TEST READY
2201 011176 100403          BMI      VCTS15 ;IS IT SET
2202 011200 005201          INC      R1 ;NO, INCREMENT R1
2203 011202 001373          BNE      VCTS14 ;TEST AGAIN
2204 011204 104400          ERROR ;ERROR, READY FAILED TO SET
2205          ; WHEN IN FAST INTENSIFY
2206 011206 160001          VCTS15: SUB      RO, R1
2207 011210 042701 000007          BIC      #7, R1
2208 011214 001001          BNE      .+4 ;ARE THEY EQUAL ?
2209 011216 104400          ERROR ;ERROR, NO TIME DIFFERENCE BETWEEN FAST INTENSIF

```

```

2210
2211           ;PRE-INTERRUPT SETUP
2212
2213 011220 042737 177437 001012      BIC      #177437,VCBRL      ;MASK TO PSW
2214 011226 001001                    BNE      .+4
2215 011230 000000                    HALT
2216 011232 022737 000340 001012      CMP      #340,VCBRL      ;LOCATION VCBRL CONTAINED A BR LEVEL 0
2217 011240 001001                    BNE      .+4      ;TEST FOR BR 7
2218 011242 000000                    HALT      ;LOCATION VCBRL CONTAINS BR LEVEL 7
2219
2220 011244 013737 001012 014514      MOV      VCBRL,BRLEV1      ;SET UP BR LEVELS
2221 011252 162737 000040 014514      SUB      #40,BRLEV1      ; -1
2222 011260 013737 001012 014516      MOV      VCBRL,BRLEV2      ; 0
2223 011266 013737 001012 014520      MOV      VCBRL,BRLEV3      ; +1
2224 011274 062737 000040 014520      ADD      #40,BRLEV3
2225
2226           ;TEST THAT THE LPSVC DOES NOT INTERRUPT
2227
2228 011302 104003                    SCOPE1
2229 011304 012777 000340 167510      MOV      #340,@PSW
2230 011312 013706 014444                    MOV      STACK,SP      ;SET UP THE STACK POINTER
2231 011316 000005                    RESET
2232 011320 012777 011354 167566      MOV      #1$,@VCIV      ;SET VECTOR
2233 011326 005077 167470                    CLR      @PSW
2234 011332 012777 000100 167522      MOV      #BIT6,@VCSTAT      ;SET INTERRUPT ENABLE
2235 011340 005000                    CLR      RO
2236 011342 005300                    DEC      RO
2237 011344 001376                    BNE      -2
2238 011346 005077 167510      CLR      @VCSTAT      ;CLEAR INTERRUPT ENABLE
2239 011352 000401                    BR      VCTS20      ;NEXT TEST
2240
2241 011354 104400           1$:      ERROR      ;UNEXPECTED DISPLAY INTERRUPT
    
```

```

2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
;TEST THAT THE DISPLAY DOES INTERRUPT A LEVEL INDICATED -1
VCTS20: SCOPE1
RESET
MOV #340, APSW
MOV STACK, SP ;SET UP STACK
MOV #VCRET1, AVCIV ;SET INTERRUPT VECTOR
MOV #400, RO ;SET UP DELAY
MOV BRLEV1, APSW
MOV #BIT6!BIT0, AVCSTAT ;START DISPLAY
DEC RO ;DELAY
BNE -2
CLR AVCSTAT ;DO NOT LET INTERRUPT ENABLE SET
ERROR ;DISPLAY FAILED TO INTERRUPT

;TEST THAT THE DISPLAY DOES NOT INTERRUPT A LEVEL INDICATED
VCRET1: SCOPE1
RESET
MOV #340, APSW
MOV STACK, SP ;SET UP STACK
MOV #VCRET2, AVCIV ;SET INTERRUPT VECTOR
MOV #400, RO ;SET UP DELAY
MOV BRLEV2, APSW
MOV #BIT6!BIT0, AVCSTAT ;START DISPLAY
DEC RO ;DELAY
BNE -2
BR OPS5
VCRET2: CLR AVCSTAT ;DO NOT LET INTERRUPT ENABLE SET
ERROR ;DISPLAY INTERRUPTED IN ERROR
OPS5: MOV VCIVS, AVCIV ;RESTORE INTERRUPT VECTOR
RESET
    
```

```

;TEST THAT RESET CLEARS MODE 0,2 ,FAST INTEN. AND EXT DELAY
011554 104003 SCOPE1
011556 012777 MOV #BIT4!BIT3!BIT2!BIT1,3VCSTAT
011558 000005 RESET
011560 017700 MOV 3VCSTAT,R0 ;READ STATUS
011562 042700 BIC #BIT7,R0
011564 005700 TST R0
011566 001401 BEQ .+4
011568 104400 ERROR ;ERROR, RESET FAILED TO CLEAR VC STATUS REG

;TEST THAT RESET CLEARS INTERRUPT ENABLE, CHANNEL, STORE, WRITE THRU
011554 104003 SCOPE1
011556 012777 MOV #BIT11!BIT10!BIT9!BIT6,3VCSTAT
011558 000005 RESET
011560 017700 MOV 3VCSTAT,R0 ;READ STATUS
011562 042700 BIC #BIT7,R0
011564 005700 TST R0
011566 001401 BEQ .+4
011568 104400 ERROR ;ERROR, RESET FAILED TO CLEAR VC STATUS

;TEST THAT RESET CLEARS X REGISTER
011604 104003 SCOPE1
011606 012777 MOV #-1,3VCXREG
011608 000005 RESET
011610 005777 TST 3VCXREG
011612 001401 BEQ .+4
011614 104400 ERROR ;ERROR, RESET FAILED TO CLEAR VC X REGISTER

;TEST THAT RESET CLEARS Y REGISTER
011626 104003 SCOPE1
011630 012777 MOV #-1,3VCYREG
011634 000005 RESET
011638 005777 TST 3VCYREG
011642 001401 BEQ .+4
011646 104400 ERROR ;ERROR, RESET FAILED TO CLEAR VC Y REGISTER

011650 104003 SCOPE1
011654 000005 RESET
011658 005737 TST #42
011662 001402 BEQ .+6
011666 000137 JMP WHATD
011670 004737 JSR 7,BELL ;LOGICAL END OF THIS TEST
011674 005237 INC PASSCT
011678 000137 JMP VCTSTO

```

000000
000001
000002
000003
000004
000005
000006
000007
000008
000009
000010
000011
000012
000013
000014
000015
000016
000017
000018
000019
000020
000021
000022
000023
000024
000025
000026
000027
000028
000029
000030
000031
000032
000033
000034
000035
000036
000037
000038
000039
000040
000041
000042
000043
000044
000045
000046
000047
000048
000049
000050
000051
000052
000053
000054
000055
000056
000057
000058
000059
000060
000061
000062
000063
000064
000065
000066
000067
000068
000069
000070
000071
000072
000073
000074
000075
000076
000077
000078
000079
000080

```

011702 105777 167122
011706 100375
011710 012777 000240 167114
011716 005337 011732
011722 003367
011724 005037 011732
011730 000002
011732 000000

011734 010046
011736 010146
011740 010246
011742 010346
011744 010446
011746 010546
011750 012704 012334
011754 042777 000100 167042
011762 005037 014452
011766 005037 012334
011772 005037 012336
011776 005037 012340
012002 005037 012342
012006 005037 012344
012012 005037 012346
012016 005037 012350
012022 105777 166776
012026 100375
012030 017701 166772
012034 042701 177600
012040 120127 000060
012044 100420
012046 122701 000173
012052 100415
012054 010124
012056 005237 014452
012062 022737 000006 014452
012070 100516
012072 105777 166732
012076 100375
012100 110177 166726
012104 000746

012106 122701 000001
012112 001013
012114 104000
012116 014357
012120 012605
    
```

```

:SUBROUTINE TO ISSUE N SPACES
:N IS ONE PLUS VALUE CONTAINED IN SPACEX
:SPACEX IS CLEARED WITHIN THE SUBROUTINE, SO THAT A CALL ON
:SPACE WITHOUT LOADING SPACEX ISSUES ONLY ONE SPACE

XSPACE: TSTB  JTPS          ;WAIT FOR TTY READY
        SPL   -4
        MOV   #240, JTPB   ;OUTPUT A SPACE
        DEC   SPACEX      ;DECREMENT COUNT
        BGT   XSPACE      ;LOOP IF NOT DONE
        CLR   SPACEX      ;RESET COUNT TO ZERO
        RTI
SPACEX: 0
        ;RETURN

:KEYBOARD SERVICE ROUTINE

XTTYIN: MOV    R0, -(SP)
        MOV    R1, -(SP)
        MOV    R2, -(SP)
        MOV    R3, -(SP)
        MOV    R4, -(SP)
        MOV    R5, -(SP)
NEWIN:  MOV    #INBUF, R4   ;SETUP CHARACTER BUFFER
        BIC    #BIT6, JTKS
        CLR   CHRCNT      ;CLEAR CHARACTER COUNTER
        CLR   INBUF
        CLR   INBUF+2
        CLR   INBUF+4
        CLR   INBUF+6
        CLR   INBUF+10
        CLR   INBUF+12
        CLR   INBUF+14
INPUTA: TSTB  JTKS        ;CHARACTER READY?
        BPL   INPUTA      ;NO, WAIT IT OUT
        MOV   JTKB, R1     ;SAVE CHARACTER
        BIC   #177600, R1 ;STRIP PARITY BIT
        CMPB  R1, #60      ;IS IT A SPECIAL CHARACTER
        BMI   SPCHR       ;YES, TEST IT
        CMPB  #173, R1
        BMI   SPCHR
INPUTB: MOV   R1, (R4)+    ;SAVE CHARACTER
        INC   CHRCNT      ;INCREMENT THE CHARACTER COUNT.
        CMP   #6, CHRCNT
        BMI   SPCHR5
OUTPTA: TSTB  JTPS        ;ECHO CHARACTER
        BPL   OUTPTA
        MOVB  R1, JTPB
        BR    INPUTA      ;WAIT FOR NEXT CHARACTER

:SUBROUTINE TO TEST FOR SPECIAL CHARACTERS : 'A', 'C', 'G', 'CR', '.', ' ' OR 'RUBOUT'
SPCHR:  CMPB  #1, R1      ;CHAR. = 'A'
        BNE  SPCHR1      ;NO, NOT 'A'
        PRINT
        CNTRLA
        MOV   (SP)+, R5   ;RESTORE 'SP'
    
```

```

012122 012604 MOV (SP)+,R4
012124 012603 MOV (SP)+,R3
012126 012602 MOV (SP)+,R2
012130 012601 MOV (SP)+,R1
012132 012600 MOV (SP)+,R0
012134 022626 CMP (SP)+,(SP)+
012136 000177 002304 JMP JAVECTR ;YES, EXIT VIA 'IA' VECTOR ADDRESS.
012142 122701 000003 SPCHR1: CMPB #3,R1 ;CHAR. = 'IC'
012146 001002 BNE +6 ;NO, NOT 'IC'
012150 000137 001352 JMP MONITR ;YES, EXIT TO MONITOR
012154 122701 000177 CMPB #177,R1 ;CHAR. = 'RUBOUT'
012160 001011 BNE SPCHR3 ;IGNORE CHAR. & EXIT
012162 005737 014452 TST CHRCNT ;IS RUBOUT LEGAL?
012166 001715 BEQ INPUTA ;NO, IGNORE IT
012170 005337 014452 DEC CHRCNT
012174 012701 000134 MOV #134,R1 ;TYPE '\ ' TO INDICATE RUBOUT
012200 005744 TST -(R4) ;POP OFF LAST CHARACTER
012202 000733 BR OUTPTA ;WAIT FOR NEXT CHARACTER
012204 122701 000054 SPCHR3: CMPB #54,R1 ;TEST FOR ' '
012210 001721 BEQ INPUTB ;LEGAL CHAR., SAVE IT
012212 122701 000015 SPCHR4: CMPB #15,R1 ;=TO 'CARRIAGE RETURN' TO TERMINATE?
012216 001011 BNE IS ;NO, CONTINUE
012220 104000 PRINT ;YES, TYPE 'CR-LF'
012222 014411 CALF
4$: MOV (SP)+,R5
MOV (SP)+,R4
MOV (SP)+,R3
MOV (SP)+,R2
MOV (SP)+,R1
MOV (SP)+,R0
RTI ;EXIT
1$: CMPB #7,R1 ;TEST IF CTRL G
SPCHR5: BNE SPCHR5 ;BR IF NOT
PRINT CNTRLG ;REPORT OLD
PRTCT ;ASK FOR NEW
SOFTSW PRINT NEWSWR TTYIN ;ASK FOR NEW
CLR R1 ;CLEAR NEW
012270 012700 012334 MOV #INBUF,R0 ;LOAD NEW POINTER
012274 005710 2$: TST (R0) ;TEST FOR TERM
012276 001410 BEQ 3$ ;BR IF TERM
012300 012002 MOV (R0)+,R2 ;GET A VALUE
012302 042702 177770 BIC #177770,R2 ;MASK
012306 006301 ASL R1
012310 006301 ASL R1
012312 006301 ASL R1
012314 060201 ADD R2,R1
012316 000766 BR 2$
012320 010137 000170 3$: MOV R1,SOFTSW ;SAVE SWITCH VALUE
012324 000737 BR 4$
012326 104000 SPCHR5: PRINT ;OTHERWISE TYPE '?'
012330 014416 QMARK

```

012332 000606
012334 000000

012352

012352 010046
012354 010146
012356 010246
012358 010346
012360 010446
012362 010546
012364 010646
012366 010746
012368 010846
012370 010946
012372 011046
012374 011146
012376 011246
012404 000000

012406 012777
012414 013706
012422 012637
012430 012605
012438 012604
012446 012603
012454 012602
012462 012601
012470 012600
012478 000005
012486 040000
012494 014421
012502 000137

000024
014450
012406 000024

000340 166406
014450
000024

001352

```
BR NEWIN
INBUF: 0
: WAIT FOR NEW ENTRY
: CHARACTER STORAGE BUFFER

:POWER FAIL HANDLER
PWRFAL: MOV R0,-(SP)
MOV R1,-(SP)
MOV R2,-(SP)
MOV R3,-(SP)
MOV R4,-(SP)
MOV R5,-(SP)
MOV 24,-(SP)
MOV SP,PROC
MOV #PWRUP,24
HALT

:POWER UP HANDLER
PWRUP: MOV #340,SPSW
MOV PROC,SP
MOV (SP)+,R24
MOV (SP)+,R5
MOV (SP)+,R4
MOV (SP)+,R3
MOV (SP)+,R2
MOV (SP)+,R1
MOV (SP)+,R0
RESET
PRINT
MESG!
JMP MONITR
```

012452
012454
012460
012464
012466
012470
012472
012474
012500
012504
012510

012512
012514
012516
012520
012522
012524
012526
012530
012532

011646
162716 000002
017616 000000
005716
001001
000000
006316
042716 177001
062716 012512
017616 000000
000136

012534
013456
013322
013416
011702
012662
011734
013004
013456

:EMT DISPATCH SERVICE ROUTINE
:ARGUMENT OF EMT IS EXTRACTED AND USED AS OFFSET TO OBTAIN POINTER
:TO THE SELECTED SUBROUTINE.

EMTSRV: MOV (SP), -(SP) ;GET PC FOR TO RETURN
SUB #2, (SP) ;PC OF EMT
MOV 3(SP), (SP) ;GET EMT
TST (SP) ;IS EMT VALID?
BNE EMTOK
HALT ;INVALID EMT
EMTOK: ASL (SP) ;MULTIPLY EMT ARG BY '2'
BIC #177001, (SP) ;CLEAR UNWANTED BITS
ADD #EMTTAB, (SP) ;POINTER TO SUBROUTINE ADDRESS
MOV 3(SP), (SP) ;SUBROUTINE ADDRESS
JMP 3(SP)+ ;GO TO SUBROUTINE

:EMT DISPATCH TABLE

EMTTAB: TYPMES ;MESSAGE PRINT ROUTINE
SCOPEI ;SCOPE ROUTINE
SCOPEC ;LOGIC TEST SCOPE ROUTINE
SCOPEH ;LOGIC TEST SCOPE LOOP (10)
XSPACE ;SUBROUTINE TO TYPE SPACES
OCTPRT ;OCTAL PRINT ROUTINE
XTTYIN ;TELEPRINTER SERVICE ROUTINE
TKSFLG ;SUBROUTINE TO TEST FOR KEYBOARD FLAG
SCOPEI ;SCOPE ROUTINE

```

2500
2501
2502
2503
2504 012534 000240
2505 012536 000240
2506 012540 017605 000000
2507 012544 062716 000002
2508 012550 105777 166254
2509 012554 100375
2510 012556 122715 000100
2511 012562 001001
2512 012564 000002
2513 012566 122715 000045
2514 012572 001403
2515 012574 112577 166232
2516 012600 000763
2517 012602 012777 000015 166222
2518 012610 105777 166214
2519 012614 100375
2520 012616 012777 000012 166206
2521 012624 013737 001016 012660
2522 012632 105777 166172
2523 012636 100375
2524 012640 113777 001020 166164
2525 012646 005337 012660
2526 012652 100367
2527 012654 105725
2528 012656 000734
2529 012660 000000
2530
2531
2532
2533
2534 012662 000240
2535 012664 000240
2536 012666 017605 000000
2537 012672 062716 000002
2538 012676 012737 000006 012776
2539 012704 012737 000376 013002
2540 012712 000401
2541 012714 006115
2542 012716 006115
2543 012720 006115
2544 012722 111537 013000
2545 012726 143737 013002 013000
2546 012734 052737 000260 013000
2547 012742 132777 000200 166060
2548 012750 100374
2549 012752 113777 013000 166052
2550 012760 012737 000370 013002
2551 012766 005337 012776
2552 012772 001350
2553 012774 000002
2554 012776 000000
2555 013000 000000
    
```

```

:MESSAGE PRINT ROUTINE, ENTERED VIA EMT DISPATCH HANDLER.
:ROUTINE PICKS UP CONTENTS OF THE 'PC' AND USES THIS AS
:THE ADDRESS OF MESSAGE TO BE TYPED.

TYPMES: NOP
        NOP
        MOV      @ (SP), R5      ;GET THE MESSAGE ADDRESS FROM START
        ADD     #2, (SP)        ;SET UP STACK TO EXIT
TYPERA: TSTB   @TPS
        BPL     TYPERA          ;WAIT FOR TTY DONE
        CMPB   #100, (R5)       ;TEST FOR 'a'
        BNE    TYPER1          ;BRANCH IF NO EQUAL
        RTI                               ;OTHERWISE EXIT
TYPER1: CMPB   #45, (R5)        ;TEST FOR '%'
        BEQ    TYPECL          ;IF = TYPE 'CR-LF'
TYPER2: MOVB   (R5)+, @TPB      ;OUTPUT CHAR.
        BR     TYPERA
TYPECL: MOV    #15, @TPB       ;TYPE 'CR'
        TSTB   @TPS
        BPL    -4
        MOV    #12, @TPB
        MOV    FILL5, 25        ;LOAD COUNTER
1$:     TSTB   @TPS           ;TEST FLAG
        BPL    1$
        MOVB   FILCHR, @TPB    ;LOAD FILLER CHARACTER
        DEC    25              ;DONE
        TSTB   (R5)+          ;INCREMENT BUFFER
        BR     TYPERA
2$:     0

:SUBROUTINE TO TYPEOUT A '6' DIGIT OCTAL NO. THE 'PC' CONTAINS
:THE ADDRESS OF 'WORD' TO BE TYPED

OCTPRT: NOP
        NOP
        MOV    @ (SP), R5      ;THE ADDRESS OF WORD TO BE TYPED
        ADD   #2, (SP)        ;SET UP STACK TO EXIT
        MOV   #6, 10$
        MOV   #376, MASK      ;MASK FOR FIRST BIT
        BR   +4
1$:     ROL   (R5)
        ROL   (R5)
        ROL   (R5)
        MOVB (R5), 11$
        BICB MASK, 11$
        BIS  #260, 11$
        BITB #200, @TPS
        BPL  -6                ;WAIT FOR PRINTER READY
        MOVB 11$, @TPB        ;PRINT CHAR.
        MOV  #370, MASK      ;MASK FOR NEXT '5' DIGITS
        DEC  10$
        BNE  1$
        RTI
10$:    0
11$:    0
    
```

2556	013002	000376		
2557				
2558	013004	105777	166014	
2559	013010	100001		
2560	013012	104006		
2561	013014	000002		
2562				
2563				
2564	013016	104007		
2565	013020	037727	166010	020000
2566	013026	001125		
2567	013030	012737	013314	000004
2568	013036	011637	014462	
2569	013042	162737	000002	014462
2570	013050	004537	013524	
2571	013054	014462		
2572	013056	017737	165766	014470
2573	013064	017737	165762	014472
2574	013072	017737	165756	014474
2575	013100	017737	165754	014476
2576	013106	017737	165744	014500
2577	013114	017737	165742	014464
2578	013122	017737	165736	014466
2579	013130	017737	165732	014504
2580	013136	013737	014506	014502
2581	013144	012737	000006	000004
2582	013152	005737	014442	
2583	013156	001006		
2584	013160	104000		
2585	013162	014411		
2586	013164	104000		
2587	013166	013723		
2588	013170	005237	014442	
2589	013174	104000		
2590	013176	014411		
2591	013200	104005		
2592	013202	014462		
2593	013204	104004		
2594	013206	104005		
2595	013210	014470		
2596	013212	104004		
2597	013214	104005		
2598	013216	014472		
2599	013220	104004		
2600	013222	104005		
2601	013224	014474		
2602	013226	104004		
2603	013230	104005		
2604	013232	014476		
2605	013234	104004		
2606	013236	104005		
2607	013240	014500		
2608	013242	104004		
2609	013244	104005		
2610	013246	014464		
2611	013250	104004		

```

MASK: 376
: SUBROUTINE TO TEST FOR THE KEYBOARD FLAG BEING SET
↑KSFLG: TSTB  JTKS      ; FLAG SET?
          BPL   .+4      ; NO, EXIT
          TTYIN ; YES, INQUIRE
          RTI

: ENTERED WITH SYSTEM TRAP CALL (ERROR)
LOGERR: TSTTKS      ; TEST FOR KEYBOARD INTERRUPT
          BIT  @SWR, #20000 ; TEST FOR INHIBIT PRINT OUT
          BNE  CK      ; INHIBIT, CHECK FOR HALT
          MOV  @LGERR2, @#4 ; SET UP FOR BUSS ERROR
          MOV  (SP), KSTOR3 ; PC OF FAILING ROUTINE
          SUB  #2, KSTOR3
          JSR  5, LEDS
          MOV  @CSR, KSTOR6 ; SAVE CLOCK STATUS
          MOV  @CSB, KSTOR7 ; SAVE CLOCK PRESET
          MOV  @GRSTAT, KSTOR8 ; SAVE I/O STATUS
          MOV  @GRDIO, KSTOR9 ; SAVE I/O OUTPUT
          MOV  @GRDAI, KSTR10 ; SAVE I/O INPUT
          MOV  @VCSTAT, KSTOR4 ; SAVE VC STATUS
          MOV  @VCXREG, KSTOR5 ; SAVE X AXIS
          MOV  @VCYREG, KSTR12 ; SAVE Y AXIS
          MOV  TEMP, KSTR11 ; SAVE TEMP
          MOV  #6, @#4 ; RESET BUSS ERROR
          TST  PRINT1
          BNE  LGERR1
          PRINT
          CRLF
          PRINT
          MES1
          INC  PRINT1
LGERR1: PRINT1 ; OUTPUT CARRIAGE RETURN AND LINE FEED
          CRLF
          PRTCT
          KSTOR3
          SPACE
          PRTCT
          KSTOR6
          SPACE
          PRTCT
          KSTOR7
          SPACE
          PRTCT
          KSTOR8
          SPACE
          PRTCT
          KSTOR9
          SPACE
          PRTCT
          KSTR10
          SPACE
          PRTCT
          KSTOR4
          SPACE

```

```

2612 013252 104005          PRTOCT
2613 013254 014466          KSTORS
2614 013256 104004          SPACE
2615 013260 104005          PRTOCT
2616 013262 014504          KSTR12
2617 013264 104004          SPACE
2618 013266 104005          PRTOCT
2619 013270 014502          KSTR11
2620 013272 104004          SPACE
2621 013274 105777 165530      TSTB      @TPS
2622 013300 100375          BPL      -4
2623 013302 005777 165526      CK:      TST      @SWR      ;CHECK SR FOR HALT SWITCH
2624 013306 100001          SPL      .+4      ;BRANCH IF NOT SET
2625 013310 000000          HALT      ;HALT ON ERROR UP
2626 013312 000002          RTI      ;RETURN TO MAIN LINE
2627 013314 062716 000002      LGERR2:  ADD      #2, (SP)
2628 013320 000002          RTI
2629
2630
2631      ;SCOPE AND/OR ITERATION LOOP FOR SOME TEST 4000 TIMES
2632
2633 013322 104007          SCOPEC:  TSTTKS      ;TEST FOR KEYBOARD INIT
2634 013324 032777 040000 165502      BIT      #40000, @SWR      ;TEST SR FOR SCOPE
2635 013332 001015          BNE      SCOPEB      ;YES, SCOPE
2636 013334 032777 004000 165472      BIT      #4000, @SWR      ;NO-TEST FOR ITERATION
2637 013342 001016          BNE      SCOPEG      ;INHIBIT ITERATION
2638 013344 005737 001040      TST      PASSCT      ;TEST IF FIRST PASS
2639 013350 001413          BEQ      SCOPEG      ;BR IF FIRST PASS -- QUICKK PASS
2640 013352 023737 013412 001042      CMP      SCOPEF, ICOUNT ;COMPARE CURRENT COUNT TO MAX NUMBER
2641 013360 001407          SCOPEJ:  BEQ      SCOPEG      ;EXIT-DONE
2642 013362 005237 013412          INC      SCOPEF      ;INCREMENT COUNT
2643 013366 022606          SCOPEB:  CMP      (6)+, SP      ;REPOSITION STACK
2644 013370 012677 165426      MOV      (6)+, @PSW      ;RESTORE PREVIOUS PROCESSOR STATUS
2645 013374 000177 000014          JMP      @RETURN      ;REPEAT TEST
2646 013400 005037 013412          SCOPEG:  CLR      SCOPEF      ;CLEAR COUNT
2647 013404 011637 013414          MOV      @SP, RETURN      ;SAVE SCOPE RETURN POINTER
2648 013410 000002          RTI      ;RETURN INLINE-NEXT TEST
2649 013412 000000          SCOPEF:  0          ;COUNT LOCATION FOR ITERATION LOOP
2650 013414 002130          RETURN:  KWTO      ;ADDRESS OF LAST TEST
2651
2652      ;SCOPE AND/OR INTERATION LOOP FOR SOME TESTS 10 TIMES
2653
2654 013416 104007          SCOPEH:  TSTTKS
2655 013420 032777 040000 165406      BIT      #40000, @SWR
2656 013426 001357          BNE      SCOPEB
2657 013430 032777 004000 165376      BIT      #4000, @SWR
2658 013436 001360          BNE      SCOPEG
2659 013440 005737 001040      TST      PASSCT      ;TEST IF FIRST PASS
2660 013444 001755          BEQ      SCOPEG      ;BR IF YES
2661 013446 023727 013412 000010      CMP      SCOPEF, #10
2662 013454 000741          BR      SCOPEJ
2663
2664      ;SCOPE LOOP FOR SOME TEST 1 TIME
2665
2666 013456 104007          SCOPEI:  TSTTKS      ;TEST KEYBOARD
2667 013460 032777 040000 165346      BIT      #40000, @SWR

```

```

2668 013466 001337          BNE      SCOPEB
2669 013470 000743          BR       SCOPEG
2670
2671 013472 104000          BELL:    PRINT
2672 013474 013504          ENDPAS
2673 013476 104000          BELLA:   PRINT
2674 013500 013521          RING
2675 013502 000207          RTS     PC      ;EXIT
2676
2677 013504 042445 042116 050040 ENDPAS:  .ASCII  '%END PASS  @'
2678 013512 051501 020123 020040
2679 013520      100
2680 013521      007      007      100 RING:   .BYTE  7,7,100
2681          .EVEN
2682
2683
2684          ;LOAD THE LPS DISPLAY LIGHTS
2685
2686 013524 012737 000006 013632 LEDS:   MOV     #6,CNTLED
2687 013532 005037 013640          CLR     LEDSV3
2688 013536 013537 013634          MOV     @5+,LEDSV1
2689 013542 005737 013642          TST     NOLEDS
2690 013546 001401          BEQ     LEDSA
2691 013550 000205          RTS
2692 013552 013737 013634 013636 LEDSA:  MOV     LEDSV1,LEDSV2
2693 013560 042737 177770 013636          BIC     #177770,LEDSV2
2694 013566 113737 013640 013637          MOVB   LEDSV3,LEDSV2+1
2695 013574 013777 013636 165244          MOV     LEDSV2,@ADDBR
2696 013602 006237 013634          ASR     LEDSV1
2697 013606 006237 013634          ASR     LEDSV1
2698 013612 006237 013634          ASR     LEDSV1
2699 013616 005237 013640          INC     LEDSV3
2700 013622 005337 013632          DEC     CNTLED
2701 013626 001351          BNE     LEDSA
2702 013630 000205          RTS     5
2703
2704 013632 000006          CNTLED: 6
2705 013634 000000          LEDSV1: 0
2706 013636 000000          LEDSV2: 0
2707 013640 000000          LEDSV3: 0
2708 013642 000000          NOLEDS: 0
2709

```

Line	Code 1	Code 2	Code 3	Code 4	Code 5	Message
2710						
2711						:MESSAGES
2712	013644	022445	046045	051520		TITLE: .ASCII '%LPS DIAGNOSTIC TEST II (MAINDEC-11-DZLPD-C)'
2713	013652	042040	040511	047107		
2714	013660	051517	044524	020103		
2715	013666	042524	052123	044440		
2716	013674	020111	046450	044501		
2717	013702	042116	041505	030455		
2718	013710	026461	055104	050114		
2719	013716	026504	024503	100		
2720						
2721	013723	040	050040	020103	MES1:	.ASCII " PC "
2722	013730	020040				
2723	013732	045503	052123	052101		.ASCII "CKSTAT "
2724	013740	040				
2725	013741	103	041113	043125		.ASCII "CKBUFF "
2726	013746	020106				
2727	013750	047511	052123	052101		.ASCII "IOSTAT "
2728	013756	040				
2729	013757	111	026517	052517		.ASCII "IO-OUT "
2730	013764	020124				
2731	013766	044440	026517	047111		.ASCII " IO-IN "
2732	013774	040				
2733	013775	126	051503	040524		.ASCII "VCSTAT "
2734	014002	020124				
2735	014004	041526	040530	051530		.ASCII "VCXAXS "
2736	014012	040				
2737	014013	126	054503	054101		.ASCII "VCYAXS "
2738	014020	020123				
2739	014022	052040	046505	040120		.ASCII " TEMP "
2740						
2741	014030	021045	046103	041517	MES2:	.ASCII '"CLOCK LOGIC TEST"'
2742	014036	020113	047514	044507		
2743	014044	020103	042524	052123		
2744	014052	022442	100			
2745	014055	045	042042	043511	MES3:	.ASCII '"DIGITAL I/O LOGIC TEST"'
2746	014062	052111	046101	044440		
2747	014070	047457	046040	043517		
2748	014076	041511	052040	051505		
2749	014104	021124	040045			
2750	014110	021045	041523	050117	MES5:	.ASCII '"SCOPE LOGIC TEST"'
2751	014116	020105	047514	044507		
2752	014124	020103	042524	052123		
2753	014132	022442	100			
2754	014135	045	053042	051511	MES6:	.ASCII '"VISUAL DISPLAY TEST"'
2755	014142	040525	020114	044504		
2756	014150	050123	040514	020131		
2757	014156	042524	052123	022442		
2758	014164	100				
2759	014165	045	054524	042520	MES4:	.ASCII '"TYPE LETTER ' ' TO RUN DESIRED TEST:'"
2760	014172	046040	052105	042524		
2761	014200	020122	020047	020047		
2762	014206	047524	051040	047125		
2763	014214	042040	051505	051111		
2764	014222	042105	052040	051505		
2765	014230	035124	045			

2766	014233	047	023501	041475		.ASCII	""A'=CLOCK LOGIC%"
2767	014240	047514	045503	046040			
2768	014246	043517	041511	045			
2769	014253	047	023502	044475		.ASCII	""B'=I/O LOGIC%"
2770	014260	047457	046040	043517			
2771	014266	041511	045				
2772	014271	047	023503	051475		.ASCII	""C'=SCOPE CONTROL LOGIC%"
2773	014276	047503	042520	041440			
2774	014304	047117	051124	046117			
2775	014312	046040	043517	041511			
2776	014320	045					
2777	014321	047	023504	053075		.ASCII	""D'=VISUAL SCOPE DISPLAY%@"
2778	014326	051511	040525	020114			
2779	014334	041523	050117	020105			
2780	014342	044504	050123	040514			
2781	014350	022531	100				
2782							
2783	014353	136	022503	100	CNTRLC:	.ASCII	'tC%@"
2784							
2785	014357	136	040101		CNTRLA:	.ASCII	'tA@"
2786	014362	043536	047445	042114	CNTRLG:	.ASCII	'tG%OLD SWR = @"
2787	014370	051440	051127	036440			
2788	014376	040040					
2789	014400	020040	042516	020127	NEWSWR:	.ASCII	' NEW = @"
2790	014406	020075	100				
2791							
2792	014411	045	100		CRLF:	.ASCII	'%@"
2793							
2794	014413	045	040056		DOT:	.ASCII	'%.@"
2795							
2796	014416	020077	100		QMARK:	.ASCII	'? @"
2797	014421	045	047520	042527	MES21:	.ASCII	'%POWER FAILURE @"
2798	014426	020122	040506	046111			
2799	014434	051125	020105	100			
2800							
2801		014442				.EVEN	
2802							
2803							

			;ADDRESS AND CONSTANTS TABLE	
2804			PRINT1:	0
2805			STACK:	1000
2806	014442	000000	AVECTR:	INITA
2807	014444	001000	PROC:	0
2808	014446	001334	CHRCNT:	0
2809	014450	000000	COUNT:	0
2810	014452	000000	DELAY:	0
2811	014454	000000	KSTOR1:	0
2812	014456	000000	KSTOR3:	0
2813	014460	000000	KSTOR4:	0
2814	014462	000000	KSTOR5:	0
2815	014464	000000	KSTOR6:	0
2816	014466	000000	KSTOR7:	0
2817	014470	000000	KSTOR8:	0
2818	014472	000000	KSTOR9:	0
2819	014474	000000	KSTR10:	0
2820	014476	000000	KSTR11:	0
2821	014500	000000	KSTR12:	0
2822	014502	000000	TEMP:	0
2823	014504	000000	TEMP1:	0
2824	014506	000000	TEMP2:	0
2825	014510	000000	BRLEV1:	0
2826	014512	000000	BRLEV2:	0
2827	014514	000000	BRLEV3:	0
2828	014516	000000	LOW:	0
2829	014520	000000	HIGH:	0
2830	014522	000000	INCR:	20
2831	014524	000000	TIMSV:	0
2832	014526	000020	TICKS:	0
2833	014530	000000		
2834	014532	000000		

; INITIAL SP. ADDRESS
 ; 'TA' VECTOR ADDRESS
 ; TEMP STORAGE FOR 'PSW'
 ; TEMP STORAGE
 ; TEMP STORAGE

; PERMANENT STORAGE
 ; PERMANENT STORAGE
 ; PERMANENT STORAGE

; TEMPORARY STORAGE
 ; TEMPORARY STORAGE

```

2835
2836
2837
2838
2839
2840 014534 000005          VISUAL: RESET
2841 014536 104000          PRINT
2842 014540 014135          MESS
2843 014542 005037 001040  VTEST2: CLR      PASSCT
2844 014546 013706 014444  VSUALO: MOV      STACK,SP      ;LOAD THE STACK POINTER
2845 014552 013777 001040 164256  MOV      PASSCT,DISPLA ;LOAD PASS COUNT
2846 014550 012737 014626 013414  MOV      #PICO+2,RETURN ;LOAD RETURN ADDRESS
2847 014566 005077 164230      CLR      APSW
2848 014572 042737 000002 015564  BIC      #BIT1,MODE
2849 014600 032777 000020 164220  BIT      #BIT4,JSWR
2850 014606 001003
2851 014610 052737 000002 015564  BNE      VSLOA
2852 014616 052777 000100 164200  VSLOA: BIS      #BIT1,MODE
2853
2854
2855 014624 013700 001064      ;DISPLAY HORIZONTAL LINE USING INTERRUPT, NON STORE DISPLAY.
2856 014630 013701 001066  PICO: MOV      VCXREG,R0
2857 014634 004737 014656      MOV      VCYREG,R1
2858
2859
2860
2861 014640 013700 001066      ;DISPLAY A VERTICAL LINE
2862 014644 013701 001064  PIC1: MOV      VCYREG,R0
2863 014650 004737 014656      MOV      VCXREG,R1
2864 014654 000433          JSR      PC,PBB
2865
2866 014656 005077 164200          BR      PIC3
2867 014662 013704 001062          PBB: CLR      AVCSTAT
2868 014666 012737 000700 014532  MOV      VCSTAT,R4
2869 014674 004737 017254          MOV      #700,TICKS
2870 014700 012703 007760          JSR      PC,CHTIME ;CHECK TIMER
2871 014704 013702 014526          PB: MOV      #7760,R3 ;SET HIGH LIMIT
2872 014710 012711 004000          PD: MOV      INCR,R2 ;INITIALIZE INCREMENTS BETWEEN POINTS
2873 014714 012710 000000          PEEA: MOV     #4000,(1)
2874 014720 060210          PE: MOV      #0,(0) ;INCREMENT
2875 014722 005214          INC      R2,(0) ;INTENSIFY
2876 014724 105714          TSTB     (4)
2877 014726 100376          BPL      -2
2878 014730 021003          CMP      (0),R3 ;DONE ALL POINTS?
2879 014732 001372          BNE      PE ;NO
2880 014734 004737 017142          JSR      PC,TIMER
2881 014740 000763          BR      PEEA
2882 014742 000207          RTS     PC
2883

```



```

015150 012737 000000 014522
015156 012737 007770 014524
015164 005077 163672
015170 012737 000400 014532
015176 004737 017254
015202 013701 001064
015206 013702 001066
015212 013703 001062
015216 013700 014524
015222 013704 014526
015232 013712 014522
015232 011211
015234 012713 000004
015240 012700 000377
015244 105713
015246 100376
015250 060412
015252 060411
015254 005300
015256 001372
015260 105713
015262 100376
015264 013712 014524
015270 013711 014522
015274 012700 000377
015280 105713
015282 100376
015284 160412
015286 060411
015290 005300
015292 001372
015294 004737 017142
015296 000742

```

```

:PLOT AN X WITH NON STORE DISPLAY
PIC4:  MOV #0,LOW
      MOV #7770,HIGH
      CLR @VCSTAT
      MOV #400,TICKS
      JSR PC,CHTIME      :CHECK TIME
PIC4B: MOV VCXREG,R1
      MOV VCYREG,R2
      MOV VCSTAT,R3
      MOV HIGH,R0
      MOV INCR,R4
P4:    MOV LOW,(2)
      MOV (2),(1)

:PLOT LINE BEGINNING IN LOWER LEFT CORNER
      MOV #4,(3)      :ENABLE INTENSIFY ON LOADING X
      MOV #377,R0
P4A:   TSTB (3)
      BPL .-2
      ADD R4,(2)      :+4 TO Y
      ADD R4,(1)      :+4 TO X
      DEC R0
      BNE P4A         :NO
      TSTB (3)
      BPL .-2
:PLOT LINE BEGINNING IN UPPER LEFT CORNER
      MOV HIGH,(2)
      MOV LOW,(1)
      MOV #377,R0
P4B:   TSTB (3)
      BPL .-2
      SUB R4,(2)      :-4 TO Y
      ADD R4,(1)      :+4 TO X
      DEC R0
      BNE P4B         :NO
      JSR PC,TIMER
      BR P4

```

```

2971
2972
2973 015322 012737 000140 014532 :PLOT NON STORE CHARACTER SET
2974 015330 004737 017254 PIC6: MOV #140,TICKS
2975 015334 012737 170000 015570 PIC6A: JSR PC,CHTIME ;CHECK TIME
2976 015342 012737 004000 015566 MOV #170000,XPOS
2977 015350 005077 163506 MOV #4000,YPOS
2978 015354 004737 015372 CLR JVCSTAT
2979 015360 004737 017142 JSR PC,PIC6B
2980 015364 000763 JSR PC,TIMER
2981 015366 000137 016070 BR PIC6A
2982 015372 012737 177734 015572 PIC6B: JMP PIC7
2983 015400 013705 001062 MOV #-36,CHRCOL ;CHARACTERS PER ROW
2984 015404 012702 015576 MOV VCSTAT,R5
2985 015410 004737 015424 GEN1: MOV #A,R2
2986 015414 005237 015572 JSR PC,CHAR
2987 015420 001373 INC CHRCOL
2988 015422 000207 BNE GEN1
2989
2990
2991 015424 013737 015566 015574 :PLOT CHARACTER
2992 015432 042715 000016 CHAR: MOV YPOS,YPT
2993 015436 013777 015570 163420 BIC #16,(5)
2994 015444 013777 015566 163414 MOV XPOS,JVCXREG
2995 015452 105777 163404 MOV YPOS,JVCYREG
2996 015456 100375 CHAR4: TSTB JVCSTAT
2997 015460 053715 015564 BPL CHAR4
2998 015464 012704 000020 BIS MODE,(5) ;ENABLE FAST INTENSIFY ON LOADING Y
2999 015470 012700 177773 MOV #20,R4
3000 015474 012701 177771 CHAR1: MOV #-5,R0 ;INITIALIZE COLDMN COUNT
3001 015500 112203 MOVB #-7,R1 ;INITIALIZE ROW COUNT
3002 015502 106103 CHAR2: ROLB (2)+,R3 ;PUT CHARACTER POINTS IN R3
3003 015504 100006 BPL CHAR3 ;NO
3004 015506 013777 015570 163350 MOV XPOS,JVCXREG
3005 015514 013777 015566 163344 MOV YPOS,JVCYREG
3006 015522 105777 163334 CHAR3: TSTB JVCSTAT
3007 015526 100375 BPL CHAR3
3008 015530 060437 015566 ADD R4,YPOS
3009 015534 005201 INC R1 ;+1 TO ROW
3010 015536 001361 BNE CHAR2 ;FINISH ROW
3011 015540 013737 015574 015566 MOV YPT,YPOS ;REINITIALIZE ROW FOR NEXT COLUMN
3012 015546 060437 015570 ADD R4,XPOS
3013 015552 005200 INC R0 ;+1 TO COLUMN COUNT
3014 015554 001347 BNE CHAR1
3015 015556 060437 015570 ADD R4,XPOS
3016 015562 000207 RTS PC ;EXIT
3017
3018 015564 000010 MODE: 10
3019 015566 000000 YPOS: 0
3020 015570 000000 XPOS: 0 ;CONTAINS Y POSITION AT ANY GIVEN TIME
3021 015572 000000 CHRCOL: 0 ;CONTAINS X POSITION AT ANY GIVEN TIME
3022 015574 000000 YPT: 0

```

3023						
3024	015576	176	021	021	A:	.BYTE 176,21,21,21,176
3025	015601	021	176			
3026	015603	177	111	111	B:	.BYTE 177,111,111,111,66
3027	015606	111	056			
3028	015610	076	101	101	C:	.BYTE 76,101,101,101,42
3029	015613	101	042			
3030	015615	177	101	101	D:	.BYTE 177,101,101,101,76
3031	015620	101	076			
3032	015622	177	111	111	E:	.BYTE 177,111,111,111,101
3033	015625	111	101			
3034	015627	177	011	011	F:	.BYTE 177,11,11,11,1
3035	015632	011	001			
3036	015634	076	101	121	G:	.BYTE 76,101,121,121,62
3037	015637	121	062			
3038	015641	177	010	010	H:	.BYTE 177,10,10,10,177
3039	015644	010	177			
3040	015646	000	101	177	I:	.BYTE 0,101,177,101,0
3041	015651	101	000			
3042	015653	060	100	100	J:	.BYTE 60,100,100,100,77
3043	015656	100	077			
3044	015660	177	010	024	K:	.BYTE 177,10,24,42,101
3045	015663	042	101			
3046	015665	177	100	100	L:	.BYTE 177,100,100,100,100
3047	015670	100	100			
3048	015672	177	004	010	M:	.BYTE 177,4,10,4,177
3049	015675	004	177			
3050	015677	177	004	010	N:	.BYTE 177,4,10,20,177
3051	015702	020	177			
3052	015704	076	101	101	O:	.BYTE 76,101,101,101,76
3053	015707	101	076			
3054	015711	177	011	011	P:	.BYTE 177,11,11,11,6
3055	015714	011	006			
3056	015716	076	101	121	Q:	.BYTE 76,101,121,141,176
3057	015721	141	176			
3058	015723	177	011	031	R:	.BYTE 177,11,31,51,106
3059	015726	051	106			
3060	015730	046	111	111	S:	.BYTE 46,111,111,111,62
3061	015733	111	062			
3062	015735	001	001	177	T:	.BYTE 1,1,177,1,1
3063	015740	001	001			
3064	015742	077	100	100	U:	.BYTE 77,100,100,100,77
3065	015745	100	077			
3066	015747	037	040	100	V:	.BYTE 37,40,100,40,37
3067	015752	040	037			
3068	015754	177	020	010	W:	.BYTE 177,20,10,20,177
3069	015757	020	177			
3070	015761	143	024	010	X:	.BYTE 143,24,10,24,143
3071	015764	024	143			
3072	015766	003	004	170	Y:	.BYTE 3,4,170,4,3
3073	015771	004	003			
3074	015773	141	121	111	Z:	.BYTE 141,121,111,105,103
3075	015776	105	103			
3076	016000	000	102	177	N1:	.BYTE 0,102,177,100,0
3077	016003	100	000			
3078	016005	142	121	111	N2:	.BYTE 142,121,111,105,102

3079	016010	105	102			
3080	016012	042	101	111	N3:	.BYTE 42,101,111,111,66
3081	016015	111	066			
3082	016017	030	024	022	N4:	.BYTE 30,24,22,177,20
3083	016022	177	020			
3084	016024	047	105	105	N5:	.BYTE 47,105,105,105,71
3085	016027	105	071			
3086	016031	076	111	111	N6:	.BYTE 76,111,111,111,62
3087	016034	111	062			
3088	016036	101	041	021	N7:	.BYTE 101,41,21,11,7
3089	016041	011	007			
3090	016043	066	111	111	N8:	.BYTE 66,111,111,111,66
3091	016046	111	066			
3092	016050	046	111	111	N9:	.BYTE 46,111,111,111,76
3093	016053	111	076			
3094	016055	076	121	111	N0:	.BYTE 76,121,111,105,76
3095	016060	105	076			
3096	016062	000	000	000	SPACEA:	.BYTE 0,0,0,0,0
3097	016065	000	000			
3098						
3099	016070					.EVEN

3100					:CHANNEL 1 CHANNEL 2	
3101	016070	012737	000300	014532	PIC7: MOV	#300,TICKS ;SET UP A TIMER
3102	016076	004737	017254		JSR	PC,CHTIME ;CHECK TIMER
3103	016102	013705	001062		MOV	VCSTAT,R5 ;SET UP R5 FOR STATUS REGISTER POINTER
3104	016106	012777	000000	162746	PIC7AA: MOV	#0,AVCSTAT ;SET UP SCOPE CONTROL
3105	016114	012737	003000	015570	MOV	#3000,XPOS ;LOAD X POSITION
3106	016122	012737	005000	015566	MOV	#5000,YPOS ;LOAD Y POSITION
3107	016130	012737	000011	016264	MOV	#9,P7CNT ;SAVE THE NUMBER OF CHARACTERS
3108	016136	012737	016270	016266	MOV	#CH1,P7PNT ;SAVE CHANNEL 1 POINTER
3109	016144	017702	000116		PIC7A: MOV	AP7PNT,R2 ;MOVE MESSAGE POINTER INTO R2 FOR DISPLAY ROUTINE
3110	016150	004737	015424		JSR	PC,CHAR ;DISPLAY A CHARACTER
3111	016154	062737	000002	016266	ADD	#2,P7PNT ;ADD 2 TO THE MESSAGE POINTER
3112	016162	005337	016264		DEC	P7CNT ;DECREMENT CHARACTER COUNT
3113	016166	001366			BNE	PIC7A ;NOT FINISHED WITH ALL CHARACTERS
3114	016170	012777	001000	162664	MOV	#1000,AVCSTAT
3115	016176	012737	003000	015570	MOV	#3000,XPOS ;SET UP X POS FOR CHANNEL 2
3116	016204	012737	003000	015566	MOV	#3000,YPOS ;SET UP Y
3117	016212	012737	000011	016264	MOV	#9,P7CNT ;SET UP CHARACTER COUNT
3118	016220	012737	016312	016266	MOV	#CH2,P7PNT ;SET UP CHANNEL 2 POINTER
3119	016226	017702	000034		PIC7B: MOV	AP7PNT,R2 ;SET UP
3120	016232	004737	015424		JSR	PC,CHAR ;DISPLAY A CHARACTER
3121	016236	062737	000002	016266	ADD	#2,P7PNT ;ADD 2 TO THE POINTER
3122	016244	005337	016264		DEC	P7CNT ;DECREMENT COUNT
3123	016250	001366			BNE	PIC7B ;NOT FINISHED
3124	016252	004737	017142		JSR	PC,TIMER ;CHECK THE RUNTIME OF THIS ROUTINE
3125	016256	000713			BR	PIC7AA ;NOT FINISHED
3126	016260	000137	016334		JMP	PIC100 ;FINISHED, NEXT TEST
3127	016264	000000			P7CNT: O	
3128	016266	000000			P7PNT: O	
3129	016270	015610			CH1: O	
3130	016272	015641				
3131	016274	015576				
3132	016276	015677				
3133	016300	015677				
3134	016302	015622				
3135	016304	015665				
3136	016306	016062				
3137	016310	016000				
3138	016312	015610			CH2: C	
3139	016314	015641				
3140	016316	015576				
3141	016320	015677				
3142	016322	015677				
3143	016324	015622				
3144	016326	015665				
3145	016330	016062				
3146	016332	016005				
3147						

```

3148
3149
3150          ;COLOR PINCUSHION-COLORS SHOULD OVERLAY TO BECOME ORANGE
3151          ;DRAW A BOX AROUND THE SCREEN AND A "X" IN THE CENTER
3152          ;DRAW IN GREEN THEN IN RED
3153
3154 016334 005005          PIC100: CLR      R5          ;SET UP FOR COLOR PATTERN
3155 016336 012737 000240 014532 MOV      #240,TICKS
3156 016344 004737 017254          JSR      PC,CHTIME          ;CHECK TIMER
3157 016350 013701 001064          PIC10:  MOV      VCXREG,R1
3158 016354 013702 001066          MOV      VCYREG,R2
3159 016360 013703 001062          MOV      VCSTAT,R3
3160 016364 012704 000100          MOV      #100,R4
3161 016370 012777 000000 162466 PIC10A: MOV      #0,VCXREG
3162 016376 012777 000000 162462 MOV      #0,VCYREG
3163 016404 105713          TSTB     (3)          ;TEST CONTROL FOR READY
3164 016406 100376          BPL      -2
3165          ;BOTTOM LINE
3166 016410 012713 000004          MOV      #4,(3)          ;SET UP MODE
3167 016414 050513          BIS      R5,(3)          ;SET UP COLOR BIT
3168 016416 012700 000077          MOV      #77,R0
3169 016422 105713          PIC10B: TSTB     (3)          ;CHANGE IN COLOR, WAIT FOR DONE
3170 016424 100376          BPL      -2
3171 016426 060411          ADD      R4,(1)          ;ADD R4 TO XPOS
3172 016430 005300          DEC      R0
3173 016432 001373          BNE      PIC10B          ;NO
3174 016434 105713          TSTB     (3)          ;WAIT FOR LAST DOT
3175 016436 100376          BPL      -2
3176          ;RIGHT LINE
3177 016440 112713 000010          MOVB     #10,(3)          ;CHANGE MODE
3178 016444 050513          BIS      R5,(3)          ;COLOR BIT
3179 016446 012700 000077          MOV      #77,R0
3180 016452 105713          PIC10C: TSTB     (3)          ;WAIT
3181 016454 100376          BPL      -2
3182 016456 060412          ADD      R4,(2)          ;ADD R4 TO YPOS
3183 016460 005300          DEC      R0
3184 016462 001373          BNE      PIC10C          ;NO
3185 016464 105713          TSTB     (3)          ;WAIT FOR LAST DOT
3186 016466 100376          BPL      -2
3187          ;TOP LINE
3188 016470 112713 000004          MOVB     #4,(3)          ;CHANGE MODE
3189 016474 050513          BIS      R5,(3)          ;COLOR BIT
3190 016476 012700 000077          MOV      #77,R0
3191 016502 105713          PIC10D: TSTB     (3)          ;READY
3192 016504 100376          BPL      -2
3193 016506 160411          SUB      R4,(1)          ;SUB R4 FROM XPOS
3194 016510 005300          DEC      R0
3195 016512 001373          BNE      PIC10D          ;NO
3196 016514 105713          TSTB     (3)          ;WAIT FOR LAST DOT
3197 016516 100376          BPL      -2
3198          ;LEFT LINE
3199 016520 112713 000010          MOVB     #10,(3)          ;CHANGE MODE
3200 016524 050513          BIS      R5,(3)          ;COLOR BIT
3201 016526 012700 000077          MOV      #77,R0
3202 016532 105713          PIC10E: TSTB     (3)          ;READY
3203 016534 100376          BPL      -2

```



```

3251
3252
3253 ;PHOSPHOR AND ERASE TEST
3254 016712 012737 000002 014532 PIC12: MOV #2,TICKS
3255 016720 004737 017254 JSR PC,CHTIME ;CHECK TIMER
3256 016724 004737 016776 PIC12A: JSR PC,CLRVC ;ERASE THE SCREEN
3257 016730 004737 017042 JSR PC,LOADVC ;LOAD THE SCREEN
3258 016734 004737 017142 JSR PC,TIMER ;CHECK THE TIME
3259 016740 000771 BR PIC12A
3260 016742 004737 016776 JSR PC,CLRVC
3261
3262 016746 000005 PIC20: RESET
3263 016750 005737 000042 TST #42
3264 016754 001402 BEQ .+6
3265 016756 000137 002002 JMP WHATE
3266 016762 004737 013472 JSR PC,BELL ;REPORT END OF PASS
3267 016766 005237 001040 INC PASSCT
3268 016772 000137 014546 JMP VSUALQ
3269
3270 016776 012777 002000 162056 CLRVC: MOV #BIT10,AVCSTAT ;ENABLE STORE
3271 017004 052777 010000 162050 BIS #BIT12,AVCSTAT ;ERASE THE SCREEN
3272 017012 012700 000020 MOV #20,R0 ;SET UP DELAY
3273 017016 005001 CLR R1
3274
3275 017020 105777 162036 CLRVCA: TSTB AVCSTAT ;TEST FOR READY
3276 017024 100405 BMI CLRVCB ;BRANCH IF SET
3277 017026 005301 DEC R1 ;DELAY
3278 017030 001373 BNE CLRVCA
3279 017032 005300 DEC R0 ;DELAY
3280 017034 001371 BNE CLRVCA
3281 017036 104400 ERROR ;ERROR, ERASE FAILED TO SET READY AFTER A DELAY
3282
3283 017040 000207 CLRVCB: RTS PC
3284
3285 017042 005077 162014 014510 LOADVC: CLR AVCSTAT ;CLEAR STATUS
3286 017046 012737 007777 MOV #7777,TEMP1
3287 017054 013700 001062 MOV VCSTAT,R0
3288 017060 013701 001064 MOV VCXREG,R1
3289 017064 013702 001066 MOV VCYREG,R2
3290 017070 012710 002000 MOV #BIT10,(0) ;SET STORE MODE
3291 017074 013712 014510 MOV TEMP1,(2)
3292 017100 012711 007777 LODVCA: MOV #7777,(1)
3293 017104 000402 BR LODVCC
3294 017106 162711 000010 LODVCB: SUB #10,(1)
3295 017112 005210 LODVCC: INC (0)
3296 017114 000240 NOP
3297 017116 105710 TSTB (0)
3298 017120 100376 BPL .-2
3299 017122 022711 000007 CMP #7,(1)
3300 017126 001367 BNE LODVCB
3301 017130 104007 TSTTKS ;TEST FOR INPUT FLAG
3302 017132 162712 000003 SUB #3,(2)
3303 017136 001360 BNE LODVCA
3304 017140 000207 RTS PC

```

```

3305
3306
3307           ;TIMER ROUTINE
3308           ; ENTER VIA JSR PC,TIMER
3309
3310 017142 017737 161666 014530 TIMER:  MOV    @SWR,TIMSV
3311 017150 104007                TSTKS
3312 017152 032737 000400 014530 TIMERA: BIT    @BIT8,TIMSV
3313 017160 001006                BNE    TIMER2           ;BIT 8 SET ?
3314 017162 005337 014532                DEC    TICKS           ;NO, DECREMENT TICKS
3315 017166 001002                BNE    TIMER1
3316 017170 062716 000002                ADD    #2,(6)           ;ADD 2 TO STACK POINTER
3317 017174 000207                TIMER1: RTS    PC           ;RETURN
3318
3319           ; SWR 8=1 SELECT TEST TO LOCK ON
3320           ; SWR 2-0= TEST NUMBER
3321
3322 017176 042737 177770 014530 TIMER2: BIC    @177770,TIMSV
3323 017204 006337 014530                ASL    TIMSV
3324 017210 062737 017234 014530                ADD    @ROUTPT,TIMSV
3325 017216 017737 175306 014530                MOV    @TIMSV,TIMSV
3326 017224 013706 014444                MOV    STACK,SP
3327 017230 000177 175274                TIMER4: JMP    @TIMSV
3328
3329 017234 014624                ROUTPT: PICO           ;DISPLAY A HORIZONTAL LINE
3330 017236 014640                PIC1           ;DISPLAY A VERTICAL LINE
3331 017240 014744                PIC3           ;DISPLAY A SQUARE
3332 017242 015150                PIC4           ;DISPALY A "X"
3333 017244 015322                PIC6           ;DISPLAY CHARACTER SET
3334 017246 016070                PIC7           ;DISPLAY CHANNEL TEST
3335 017250 016334                PIC100        ;DISPLAY COLOR PATTERN
3336 017252 016712                PIC12        ;DISPLAY ERASE AND PHOSPHOR TEST
3337
3338 017254 013737 001014 014514 CHTIME: MOV    PDPDLY,BRLEVI
3339 017262 005337 014514 CHTMA:  DEC    BRLEVI
3340 017266 001403                BEQ    CHTMB
3341 017270 006337 014532                ASL    TICKS
3342 017274 000772                BR    CHTMA
3343 017276 000207                CHTMB:  RTS    PC

```

```

3344                                     ;SLOW RELAY SWITCH TEST
3345
3346 017300 013706 014444 RELAY: MOV STACK,SP ;LOAD THE STACK
3347 017304 052777 000401 161542 BIS #BIT8,BIT0,@GRSTAT ;LOAD RELAYS
3348 017312 004737 017332 JSR PC,DLY
3349 017316 042777 000401 161530 BIC #BIT8,BIT0,@GRSTAT ;CLEAR RELAYS
3350 017324 004737 017332 JSR PC,DLY
3351 017330 000763 BR RELAY
3352
3353 017332 013737 001014 014456 DLY: MOV PDPDLY,DELAY
3354 017340 005037 017362 CLR DELAY1
3355 017344 005237 017362 DLYA: INC DELAY1
3356 017350 001375 SNE DLYA
3357 017352 005337 014456 DEC DELAY
3358 017356 001372 BNE DLYA
3359 017360 000207 RTS PC
3360 017362 000000 DELAY1: 0
3361
3362                                     ;SCOPE OUTPUT OF CLOCK OVERFLOW
3363
3364 017364 013706 014444 CKOVFL: MOV STACK,SP
3365 017370 012777 177766 161454 MOV #-10,@CSB ;LOAD COUNTER PRESET
3366 017376 012777 000403 161444 MOV #403,@CSR ;LOAD RATE AND MODE
3367 017404 105777 161440 SOCOA: TSTB @CSR
3368 017410 100375 BPL SOCOA
3369 017412 000764 BR CKOVFL
3370
3371                                     ;TEST FOR SCHMITT TRIGGER #1 AND #2
3372
3373 017414 005037 014510 ST1: CLR TEMP1
3374 017420 042737 100000 017470 BIC #BIT15,STST
3375 017426 000406 BR STSA
3376
3377 017430 012737 001000 014510 ST2: MOV #1000,TEMP1 ;LOAD MODE
3378 017436 052737 100000 017470 BIS #BIT15,STST
3379 017444 013706 014444 STSA: MOV STACK,SP
3380 017450 005037 014462 CLR KSTOR3
3381 017454 013777 014510 161366 STSB: MOV TEMP1,@CSR ;LOAD STATUS
3382 017462 004537 013524 JSR 5,LEDS
3383 017466 014462 KSTOR3
3384
3385 017470 005777 161354 STST: TST @CSR ;TST OR TSTB
3386 017474 100375 BPL STST
3387 017476 005077 161346 CLR @CSR ;CLEAR BIT 15
3388 017502 005237 014462 INC KSTOR3
3389 017506 004737 013476 JSR PC,BELLA
3390 017512 000760 BR STSB
3391
3392
3393 001120 .END INIT

```


INCR	014526	2422	2438#						
INIT	0011200	2822#	2871	2897	2944				
INITA	001334	526#	605#	3293					
INIT1A	0011500	642#	668#	2808					
INIT1B	0012120	611#	614						
INIT1C	0013420	620#	622						
INIT2	0014000	631#	645#						
INIT3	0016200	637	644	656#	693				
INPUTA	0120200	691#							
INPUTB	0120500	2357#	2358	2372	2394				
TOTEST	006014	2365#	2400						
TOTESTA	006036	528	684	1573#					
TOTEST1	006020	1577#	1930						
K	0155500	720	1576#						
KSTOR1	0144600	3042#							
KSTOR3	0144620	3044#							
KSTOR4	014464	2813#							
KSTOR5	014466	2568#	2569*	2571	2592	2814#	3380*	3383	3388*
KSTOR6	014470	2577#	2610	2815#					
KSTOR7	014472	2578#	2613	2816#					
KSTOR8	014474	2572#	2595	2817#					
KSTOR9	014476	2573#	2598	2818#					
KSTR10	014478	2574#	2601	2819#					
KSTR11	014480	2575#	2604	2820#					
KSTR12	014482	2576#	2607	2821#					
KWT0	002130	2580#	2619	2822#					
KWT10	002356	2579#	2616	2823#					
KWT11	002400	755	765#	2850					
KWT12	002422	829#							
KWT14	002444	837#							
KWT15	002466	845#							
KWT16	002510	853#							
KWT17	002554	862#							
KWT18	002600	870#							
KWT18A	002632	885#							
KWT19	002650	900#							
KWT2	002152	903#	907						
KWT20	002674	912#							
KWT22	002730	771#							
KWT23	002766	922#							
KWT24	003034	933#							
KWT25	003102	944#							
KWT26	003154	956#							
KWT26A	003174	968#							
KWT3	002174	986#							
KWT30	003252	990#	1002						
KWT30A	003316	777#							
KWT31	003352	994	998	1012#					
KWT31A	003416	1021#	1023						
KWT32	003452	1038#							
KWT32A	003516	1047#	1049						
KWT33	003552	1065#							
KWT33A	003616	1074#	1076						
KWT34	003656	1091#							
		1100#	1102						
		1117#							

KWT34A	003576	1121#	1128						
KWT35	003734	1125	1132#						
KWT36	004016	1148#							
KWT37	004040	1157#							
KWT38	004062	1166#							
KWT39	004104	1175#							
KWT4	002216	785#							
KWT40	004126	1184#							
KWT41	004156	1197#							
KWT42	004214	1210#							
KWT43	004252	1222#							
KWT44	004276	1232#							
KWT45	004322	1242#							
KWT46	004346	1252#							
KWT47	004372	1263#							
KWT48	004430	1276#							
KWT49	004542	1285	1305#						
KWT5	002246	796#							
KWT50	004630	1323#							
KWT50A	004712	1327	1341#						
KWT50B	004740	1342	1348#						
KWT51	004744	1335	1347	1354#					
KWT51A	005042	1357	1370#						
KWT52	005044	1366	1374#						
KWT53	005136	1391#							
KWT53A	005230	1393	1404#						
KWT54	005234	1409#							
KWT54A	005326	1411	1422#						
KWT55	005362	1432#							
KWT56	005406	1444#							
KWT57A	005432	1456#							
KWT6	002270	804#							
KWT60	005456	1468#							
KWT61	005502	1481#							
KWT62	005526	1493#							
KWT7	002312	812#							
KWT8	002334	820#							
L	015665	3046#	3135	3144					
LEDS	013524	669	2570	2686#	3382				
LEDSA	013552	2690	2692#	2701					
LEDSV1	013634	2688#	2692	2696*	2697*	2699*	2705#		
LEDSV2	013636	2692#	2693*	2694*	2695	2706#			
LEDSV3	013640	2687#	2694	2699*	2707#				
LGERR1	013174	2583	2589#						
LGERR2	013314	2567	2627#						
LOADVC	017042	3257	3285#						
LODVCA	017100	3292#	3303						
LODVCB	017106	3294#	3300						
LODVCC	017112	3293	3295#						
LOGERR	013016	516	2564#						
LOGICA	002012	519	728#						
LOW	014522	2830#	2890*	2898	2899	2935*	2945	2961	
LPSADD	001000	547#	609						
LPSVCT	001002	548#	618						
M	015672	3048#							
MASK	013002	2539*	2545	2550*	2556#				

J07

LPS DIAGNOSTIC TEST II MAINDEC-11-DZLPD-C
DZLPDC.P11 04-APR-76 00:00

MACY11 27(1006) 16-SEP-76 21:04 PAGE 88
CROSS REFERENCE TABLE -- USER SYMBOLS

RUN-TIME: 9 23 3 SECONDS
RUN-TIME RATIO: 256/37=6.9
CORE USED: 10K (19 PAGES)

