LPD

INTERFACE DIAGNOSTIC

MD-11-DZLPJ-B

EP-DZLPJ-B-DL-A

OCT 1976

FICHE 1 OF 1

digital

Made In U.S.A.

```
.REM *
```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DZLPJ-B-D
PRODUCT NAME:	LPD11 INTERFACE DIAGNOSTIC TEST
DATE CREATED:	NOVEMBER 7, 1974
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	R. MOORE

1. ABSTRACT

THIS DIAGNOSTIC WILL EXERCISE ALL LOGIC FUNCTIONS
AND DATA CAPABILITIES OF THE LPD11 CONTROLLER.
THE PROGRAM SHOULD BE STARTED AT LOCATION 200 AND
WILL TYPE OUT THE PROGRAM NAME AND REQUEST INPUT OF
VECTOR ADDRESS, REGISTER ADDRESS, AND INTERFACE MODE SETTING.

THE PROGRAM CONSISTS OF FIVE (5) TEST PARTS:
LOGIC TESTS, DATA TESTS, MAINTENANCE MODE TESTS,
LINE COUNT TEST AND THE DATA OUTPUT TEST.
THE LOGIC AND DATA TESTS ARE PERFORMED SEQUENTIALY;
(EXCEPT LOGIC TEST 22 WHICH IS ENTERED VIA SWITCH TEN).
THE MAINTENANCE MODE, LINE COUNT TEST AND DATA
OUTPUT TEST ARE ENTERED THROUGH THE CONSOLE SWITCHES.

THE PROGRAM IS DESIGNED TO PROVIDE THE OPERATOR WITH AS MUCH
FLEXABILITY AS POSSIBLE THROUGH THE USE OF THE CONSOLE
SWITCH REGISTER. USE OF THE SWITCHES PROVIDES FOR CONTROL
OF ERROR PRINT, STOP ON ERROR, ITERATION OF DATA PATTERNS,
REPEAT LOOP, ERROR STATUS BIT TESTING, MAINTENANCE MODE
ENTRY, MAINTENANCE MODE INTERRUPT RECOGNITION, LINE
COUNT TEST ENTRY AND DATA OUTPUT TEST ENTRY.

2. REQUIREMENTS (EQUIPMENT)

A. PDP-11 FAIMILY
B. CONSOLE TELETYPE
C. LPD11 CONTROLLER
D. LPC01 INTERFACE - REQUIRED FOR PART 1 - TEST 22
E. PHOTO-COMPOSITION MACHINE OR LPC01 - REQUIRED FOR PART 5
F. PDP-11/45 AND KW11-L LINE CLOCK
   ARE REQUIRED FOR THE LINE COUNT TEST.

3. LOADING PROCEDURE

A. USE STANDARD PROCEDURE FOR LOADING BINARY TAPES

4. STARTING PROCEDURE

A. LOAD AND START 200 TO ENTER NEW VECTOR AND REGISTER
   ADDRESSES VIA TTY; SET CONSOLE SWITCHES BEFORE TYPING 'G'.

B. LOAD AND START 204 TO BYPASS ENTRY OF NEW ADDRESSES;
   SET CONSOLE SWITCHES BEFORE TYPING 'G'.

C. DUE TO FLOATABLE VECTOR AND REGISTER USAGE, A START
   AT 200 MUST BE DONE ON THE INITIAL RUN. THEREAFTER
   204 MAY BE USED.

D. THE PROGRAM WILL TYPE "END OF TEST" UPON COMPLETION OF A SINGLE PASS AND MAY BE RESTARTED BY PRESSING THE CONTINUE SWITCH.

E. A POWER FAIL RESTART IS PROVIDED AND WILL TYPE A POWER FAIL MESSAGE BEFORE RESTARTING AT LOC 200(8).

5. CONSOLE SWITCH SETTINGS

A. THE CONSOLE SWITCHES MAY BE SET TO ANY CONFIGURATION AND ARE APPLICABLE TO ALL TESTS.
B. ANY SWITCH MAY BE CHANGED DYNAMICALLY.
C. IF NO SWITCHES ARE SET THE PROGRAM WILL RUN IN THE "NORMAL" MANNER. THE SWITCHES SHOULD BE SET FOR ANY VARIATION DESIRED.

SWITCHES:

SW 15: 1=NO ERROR PRINTS
0=PRINT ALL ERRORS

SW 14: 1=STOP ON ERROR
0=CONTINUE ON ERROR

SW 13: 1=LOOP MODE
0=SINGLE PASS

SW 12: 1=INHIBIT DATA ITERATIONS
0=DO NOT INHIBIT ITERATIONS

SW 11: 1=GO TO MAINTENANCE MODE
0=DO NOT ENTER MAINTENANCE MODE

SW 10: 1=DO ERROR BIT TEST
0=DO NOT DO ERROR BIT TEST

SW 9: 1=USE TESTER INTERRUPT FOR MAINTENANCE MODE
0=USE CONTROLLER INTERRUPT

SW 8: 1=ENTER LINE COUNT TEST (PDP-11/45 ONLY)
0=DO NOT ENTER LINE COUNT TEST

SW 6: 1=ENTER DATA OUTPUT TEST
0=DO NOT ENTER DATA OUTPUT TEST

SW 1: 1=IF LINE COUNT TEST PERFORMED ON 50HZ SYSTEM
0=IF LINE COUNT TEST PERFORMED ON 60HZ SYSTEM

SW 0: 1=SELECT TYPESET READER 0 FOR DATA INPUT (TEST 5)
0=SELECT KEYBOARD FOR DATA INPUT (TEST 5)

6. ERROR PRINTOUTS

A. THERE ARE THREE (3) TYPES OF ERROR PRINTOUTS:

1. ILLEGAL VECTOR OR REGISTER ENTRY
2. LOGIC TEST ERROR
3. DATA TEST ERROR

B. ANY ODD VECTOR OR REGISTER ADDRESS, OR OUT OF BOUNDS ENTRY (0-776 FOR VECTORS; 760000-777600 FOR REGISTERS) WILL RESULT IN AN ERROR PRINT AND REENTRY REQUEST. REGARDLESS OF CONSOLE SWITCH 15 SETTING.

C. ANY LOGIC ERROR WILL RESULT IN A PRINT OUT OF THE TEST NUMBER AND ENGLISH LANGUAGE DESCRIPTION OF THE ERROR.

D. ANY DATA ERROR WILL RESULT IN A PRINT OUT OF THE EXPECTED DATA AND THE RECIEVED DATA IN BIT FORMAT.

E. CONSOLE SWITCH 15 CONDITIONS ERROR PRINTOUTS.
BY SETTING SWITCH 15 TO A ONE, ALL ERROR MESSAGES ARE INHIBITED EXCEPT FOR ENTRY OF AN ILLEGAL VECTOR OR REGISTER ADDRESS.

F. CONSOLE SWITCH 14 CONDITIONS STOP/CONTINUE ON ERROR
IF A LOGIC TEST ERROR HALT IS ENCOUNTERED, THE PROGRAM MAY BE RESTARTED BY PRESSING THE CONTINUE SWITCH.
IF A DATA TEST ERROR HALT IS ENCOUNTERED, THE PROGRAM MAY BE CONTINUED BY PRESSING THE CONTINUE SWITCH.

7. TEST DESCRIPTIONS

A. PART 1: LOGIC TESTS

THIS GROUP OF 23(8) TESTS WILL CHECK ALL USABLE BITS OF BOTH THE TESTER AND CONTROLLER STATUS REGISTERS FOR SET AND RESET. THE ERROR BIT (15) IN THE CONTROLLER STATUS REGISTERS ONLY SETS WHEN THE INTERFACE (LPC01) IS POWERED DOWN OR 'OFF LINE' OR THE CONTROLLER (LPD) IS IN THE LOCAL TEST MODE WHICH REQUIRES MANNUAL INTERVENTION BY AN OPERATOR. CONSOLE SWITCH 10 CONDITIONS ENTRY TO THIS TEST. WHEN ENTERED, MESSAGES WILL BE TYPED REQUESTING THE ABOVE CONDITIONS. WHEN THE TEST IS SUCCESSFULLY COMPLETED, THE MESSAGE 'ERROR BIT OK' WILL BE TYPED AND THE TESTING WILL RESUME.

TEST0:	ASSURE THAT THE ERROR BIT(15) IS SET WHEN THE LPD HAS BEEN SWITCHED TO THE LOCAL TEST MODE.
TEST1:	ASSURE THAT ALL BITS OF THE CONTROLLER STATUS REGISTER EXCEPT READY CAN BE CLEARED.
TEST2:	ASSURE THAT ALL BITS OF THE TESTER STATJS REGISTER EXCEPT READY CAN BE CLEARED.
TEST3:	ASSURE THAT THE CONTROLLER READY BIT WAS CLEARED BY INIT.
TEST4:	ASSURE THAT THE TESTER READY BIT WAS CLEARED BY INIT.
TEST5:	ASSURE THAT THE CONTROLLER READY BIT REMAINS RESET WHEN A CHARACTER IS LOADED INTO THE CONTROLLER BUFFER.
TEST6:	ASSURE THAT THE TESTER READY BIT DOES SET UPON COMPLETION OF DATA TRANSFER FROM CONTROLLER BUFFER TO TESTER BUFFER.
TEST7:	ASSURE THAT THE GO BIT OF THE TESTER STATUS REGISTER RESETS IMMEDIATELY AFTER BEING SET.
TEST10:	ASSURE THAT THE TESTER READY BIT IS NOT RESET BY THE SETTING OF THE GO BIT.

TEST11: ASSURE THAT THE CONTROLLER READY BIT SETS AFTER THE GO BIT IS SET TO COMPLETE THE CYCLE.

TEST12: ASSURE THAT THE TESTER READY BIT IS NOT RESET AT THE END OF THE CYCLE.

TEST13: ASSURE THAT THE CONTROLLER READY BIT IS NOT RESET AT AT THE END OF THE CYCLE.

TEST14: ASSURE THAT THE TESTER INTERRUPT IS GENERATED AND RECOGNIZED AFTER COMPLETION OF A DATA TRANSFER CYCLE.

TEST15: ASSURE THAT THE CONTROLLER INTERRUPT IS GENERATED AND RECOGNIZED AT THE END OF A CYCLE.

TEST16: ASSURE THAT THE TESTER INTERRUPT WILL NOT BE RECOGNIZED WHEN THE INTERRUPT ENABLE BIT IS NOT SET.

TEST17: ASSURE THAT THE TESTER INTERRUPT WILL NOT BE RECOGNIZED AT A PRIORITY LEVEL ABOVE FOUR (4).

TEST20: ASSURE THAT THE CONTROLLER INTERRUPT IS NOT RECOGNIZED WHEN THE INTERRUPT ENABLE BIT IS NOT SET.

TEST21: ASSURE THAT THE CONTROLLER INTERRUPT IS NOT RECOGNIZED AT A PRIORITY LEVEL ABOVE FOUR (4).

TEST22: THIS TEST, ENTERED VIA CONSOLE SWITCH TEN (10), WILL CHECK THE ERROR STATUS BIT, BIT FIFTEEN (15), FOR THE PROPER STATE IN EACH MODE OF THE INTERFACE & CONTROLLER. OPERATOR INTERVENTION WILL BE REQUESTED ON THE TELETYPE TO SET THE ON/OFF LINE SWITCH AT THE INTERFACE (LPC01) AND THE LOCAL TEST MODE SWITCH ON THE M523 MODULE AT THE CONTROLLER (LPD).
1. ON LINE - NORMAL MODE: BIT 15 = 0
2. OFF LINE - NORMAL MODE: BIT 15 = 1
3. ON LINE - NORMAL MODE - POWER DOWN INTERFACE: BIT 15 = 1
4. ON LINE - TEST MODE: BIT 15 = 1

NOTE:
NORMAL MODE REFERS TO THE LOCAL TEST MODE SWITCH - OFF
TEST MODE REFERS TO THE LOCAL TEST MODE SWITCH - ON

TEST23: THIS TEST CHECKS THE IDENTIFICATION BITS OF THE LPD STATUS WORD AGAINST THE MEMORY LOCATION 'ID'. ANY 'ID' VALUE OTHER THAN 5 MUST BE LOADED INTO THE MEMORY LOCATION IDENTIFIED BY 'ID'.

B. PART 2: DATA TESTS

THIS GROUP OF 2 TESTS WILL PERFORM DATA CHECKS BY
WRAPPING DATA AROUND FROM THE CONTROLLER BUFFER TO THE
TESTER BUFFER AND COMPARING FOR ERRORS.

TEST 1: WRAP - PATTERNS OF ALL ZEROS, ALL ONES,
A ZERO WALKING FROM RIGHT TO LEFT IN A
FIELD OF ONES, AND A ONE WALKING FROM RIGHT TO LEFT IN
A FIELD OF ZEROS. THE DATA CAN BE OBSERVED IN THE
INDICATOR LIGHTS ON THE INTERFACE PANEL.

TEST 2: WRAP PATTERNS OF ALL POSSIBLE COMBINATIONS
OF THE 8 BIT CHARACTER (0 - 377).

THE PATTERNS IN TEST 1 ARE EACH REPEATED 512 TIMES
AND EACH SET OF ALL BIT COMBINATIONS IS REPEATED 6
TIMES IN TEST 2. THESE ITERATIONS MAY BE INHIBITED
VIA CONSOLE SWITCH 12.

C. PART 3: MAINTENANCE MODE

THIS GROUP, ENTERED VIA CONSOLE SWITCH 11, WILL
RUN CONTINUOUSLY AND SHOULD BE USED FOR DEBUGGING AND
DELAY ADJUSTMENTS. A FULL CYCLE OF DATA WRAP AND INTERRUPT
AT END OF CYCLE IS PERFORMED.
THE INTERRUPT RECOGNIZED, TESTER OR CONTROLLER, IS VARIABLE
DYNAMICALLY VIA CONSOLE SWITCH 9.
DATA IS TAKEN FROM CONSOLE SWITCHES 0-7 AND MAY
BE VARIED DYNAMICALLY. THE DATA IS COMPLEMENTED
ON EVERY OTHER PASS TO PROVIDE TRANSITIONS IN THE LOGIC.

D. PART 4: LINE COUNT TEST

THIS PART, ENTERED VIA CONSOLE SWITCH EIGHT (8),
WILL COUNT THE NUMER OF CHARACTERS TRANSMITTED IN
ONE SECOND AND DIVIDE THIS NUMBER BY 37 IN ORDER
TO PROVIDE A PRINTOUT OF THE NUMBER OF LINES PER
SECOND. CONSULT THE LPD11 MANUAL FOR LINES PER SECOND
INFORMATION.

THE TEST WILL ONLY RUN ON A PDP-11/45 WITH A KW11-L
LINE CLOCK.
SWITCH 1 MUST BE SET ON SYSTEMS CONNECTED TO A 50HZ
POWER SOURCE.
AS LONG AS SWITCH 8 IS SET TO A ONE, THE TEST WILL
CONTINUE TO RUN WITH A PRINTOUT APPROXIMATLY EVERY
SECOND.
CONSOLE SWITCH 13 (CONTINUOUS OR SINGLE PASS) WILL
HAVE FULL USE DURING THIS TEST; HOWEVER, NO OTHER
SWITCHES ARE USED.

E. PART 5: DATA OUTPUT TEST

THIS TEST, ENTERED VIA CONSOLE SWITCH SIX (6), IS
DESIGNED TO TAKE OCTAL VALUES (0-377) FROM THE KEYBOARD
OR TYPESET READER 0, STORE THEM IN A MEMORY BUFFER,
AND THEN TRANSMIT THE CONTENTS OF THIS BUFFER
OUT TO THE PHOTO-COMP MACHINE OR LPC01.
THE PROGRAM REQUESTS THAT THE CONTROLLER BE IN THE
NORMAL MODE AND THE PHOTO-COMP BE PLACED 'ON LINE'.
WHEN READY, THE OPERATOR RESPONDS BY TYPING A 'G'.
THE PROGRAM THEN LOOKS AT SW 0 FOR THE SELECTED INPUT
MODE (KEYBOARD OR TYPESET READER 0). IF THE READER IS
SELECTED (SW0=1), THE PAPER TAPE WILL BE READ UNTIL OUT
OF TAPE, AT WHICH TIME THE CONTROL CHARACTER 'T' IS EXPECTED.
NOTE; READER 0 MUST HAVE THE PAPER TAPE IN POSITION BEFORE
TYPING THE 'G' MENTIONED ABOVE.
IF THE KEYBOARD IS SELECTED (SW0=0), THEN THE PROGRAM TYPES
A LINE NUMBER AND WAITS FOR DATA.
THE PROGRAM WILL EXPECT OCTAL VALUES FROM 0 TO 377,
EACH TERMINATED WITH A CARRIAGE RETURN.
TYPING A "RUBOUT" WILL TYPE "??" BEFORE RETYPING THE
CURRENT LINE NUMBER.

THE OCTAL BUFFER WILL EXPAND TO 2048 BYTES BEFORE
A 'BUFFER FULL' MESSAGE IS TYPED. HOWEVER,
IF CORE EXISTS ABOVE 4K THE INPUT MAY CONTINUE TO
THAT LIMIT.
THE FOLLOWING KEYS CONTROL THE DATA TRANSMISSION
TO THE PHOTO-COMP MACHINE:

'T' -TRANSMIT- WILL TRANSMIT THE CONTENTS OF THE BUFFER
'S' -STOP- WILL STOP THE TRANSFER OF DATA ANYTIME
'C' -CONTINUE- WILL CONTINUE AT THE POINT WHERE 'S' WAS TYPED
'I' -INPUT- WILL TRANSFER CONTROL TO INPUT A NEW BUFFER

CONSOLE SWITCH 0, WHEN SET, WILL SELECT TYPESET READER 0*
INSTEAD OF THE KEYBOARD FOR DATA INPUT.
CONSOLE SWITCH 13 (CONTINUOUS OR SINGLE PASS) WILL
HAVE FULL USE DURING THIS TEST.
UNACCEPTABLE INPUTS WILL EITHER BE IGNORED OR
QUESTIONED BY THE RESPONSE '??'.

*NOTE: TYPESET READER 0 BUS ADDRESSES OF 172600 & 172602
ARE USED BY THE PROGRAM. LOCATIONS 'PRS' & 'PRB' MAY BE
ALTERED TO SELECT ANOTHER READER.

8. LISTING

←

```
                        .TITLE LPD11 INTERFACE TEST - MAINDEC-11-DZLPJ-B-D

                        .ENABL ABS AMA

                        ;NOVEMBER 7, 1974
                        ;DIGITAL EQUIPMENT CORP. MAYNARD MASS.
                        ;PROGRAMMER R. MOORE

                        ;************************
                        ;CONSOLE SWTICH SETTINGS
                        ;
                        ;SW 15: 1=NO ERROR PRINT
                        ;       0=PRINT ALL ERRORS
                        ;
                        ;SW 14: 1=STOP ON ERROR
                        ;       0=CONTINUE
                        ;
                        ;SW 13: 1=CONTINUE MODE
                        ;       0=1 PASS ONLY
                        ;
                        ;SW 12: 1=INHIBIT ITERATIONS
                        ;       0=NORMAL TEST
                        ;
                        ;SW 11: 1=MAINTENANCE MODE
                        ;       0=NORMAL MODE
                        ;
                        ;SW 10: 1=DO ERROR BIT TEST
                        ;       0=NO ERROR BIT TEST
                        ;
                        ;SW  9: 1=USE TESTER INTERRUPT IN MAINTENANCE MODE
                        ;       0=USE CONTROLLER INTERRUPT
                        ;
                        ;SW  8: 1=ENTER LINE COUNT TEST (PDP-11/45 ONLY)
                        ;       0=DO NOT ENTER LINE COUNT TEST
                        ;
                        ;SW  6: 1=ENTER DATA OUTPUT TEST
                        ;       0=DO NOT ENTER DATA OUTPUT TEST
                        ;
                        ;SW  1: 1=LINE COUNT TEST PERFORMED ON 50HZ SYSTEMS
                        ;       0=LINE COUNT TEST PERFORMED ON 60HZ SYSTEMS
                        ;
                        ;SW  0: 1=SELECT TYPESET READER 0 FOR DATA INPUT (TEST 5)
                        ;       0=SELECT KEYBOARD FOR DATA INPUT (TEST 5)
                        ;************************
```

```
                                        ;REGISTER EQUIVS
                                        
            000000                      R0=%0
            000001                      R1=%1
            000002                      R2=%2
            000003                      R3=%3
            000004                      R4=%4
            000005                      R5=%5
            000006                      SP=%6
            000006                      R6=SP
            000007                      PC=%7
                                        
                                        ;TRAP CATCHERS 0-1000
                                        
            000000                      .=0
                                        .REPT 200
                                        .+2
                                        HALT
                                        .ENDR
                                        
                                        ;POWER FAIL VECTOR
                                        
            000024                      .=24
000024  006476                          PWRFAL
000026  000340                          340
                                        
                                        ;LINE CLOCK VECTOR
                                        
            000100                      .=100
000100  004372                          TINT
000102  000340                          340
```

```
                                        ;****************************************
                                        ;PROGRAM START LOCATIONS:
                                        ;
                                        ;200 = INPUT VECTOR AND REGISTER ADDRESSES AT START
                                        ;204 = USE PREVIOUSLY SET ADDRESSES
                                        ;
                                        ;VECTORS AND REGISTER ADDRESSES MUST BE
                                        ;INPUT ON THE INITIAL RUN.
                                        ;ON ALL OTHER RUNS, THE PREVIOUSLY SET
                                        ;ADDRESSES MAY BE USED.
                                        ;****************************************

            000200                      .=200
000200  000137  001206                  JMP     PBEG            ;GO TO START AND SET VECTOR / REGISTERS

            000204                      .=204
000204  000137  001200                  JMP     PSKIP           ;GO TO START BUT DO NOT SET ADDRESSES
```

```
                    001000                          .=1000

                                                    ;FLAGS AND COUNTERS

001000  000000      PVEC:   0                       ;VECTOR POINTER SAVE
001002  000000      PREG:   0                       ;REGISTER POINTER SAVE
001004  000000      PTN:    0                       ;TEST NUMBER STORAGE
001006  000000      PSPS:   0                       ;STACK POINTER SAVE LOC
001010  000000      PITA:   0                       ;INTERRUPT RETURN STORAGE
001012  000000      PGOF:   0                       ;GO FLAG
001014  000000      PTIM1:  0                       ;TIMER
001016  000000      PTIM2:  0                       ;GROSS TIMER
001020  000000      PIC:    0                       ;ITERATION CNTR
001022  000000      PCHAR:  0                       ;TEST CHAR TEMP STORAGE
001024  000000      TOB:    0                       ;TTY OUTPUT BUFFER STORAGE
001026  000000      TIB:    0                       ;TTY INPUT BUFFER STORAGE
001030  000000      ROTA:   0                       ;ROTATION FACTOR
001032  000000      PINTR:  0                       ;INTERRUPT RETURN POINTER
001034  000000      PET:    0                       ;ERROR BIT TEMP STORAGE
001036  000005      ID:     5                       ;CONTAINS 'ID' BITS FILLED BY
                                                    ;USER IF 'ID' NOT EQUAL TO 5

                                                    ;ADDRESS CONSTANTS

001040  172600      PRS:    172600                  ;TYPESET READER 0 CSR ADRS
001042  172602      PRB:    172602                  ;TYPESET READER 0 DBR ADRS
001044  177560      TKS:    177560                  ;TTY STATUS
001046  177562      TKB:    177562                  ;TTY BUFFER
001050  177564      TPS:    177564                  ;TTP STATUS
001052  177566      TPB:    177566                  ;TTP BUFFER
001054  177546      LKS:    177546                  ;LINE CLOCK REGISTER
001056  177570      SWR:    177570                  ;CONSOLE SWITCH REGISTER
001060  177776      PSW:    177776                  ;PROGRAM STATUS WORD
001062  000000      PCS:    0                       ;LPD STATUS
001064  000000      PCB:    0                       ;LPD BUFFER
001066  000000      PTS:    0                       ;TEST STATUS
001070  000000      PTB:    0                       ;TEST BUFFER
```

```
                001200                          .=1200

                                                ;*************************************
                                                ;PROGRAM START
                                                ;
                                                ;THIS IS THE HOUSE KEEPING ROUTINE
                                                ;WHICH CLEARS ALL COUNTERS AND FLAGS
                                                ;TYPES THE PROGRAM NAME, SETS UP
                                                ;REGISTERS AND VECTOR ADDRESSES, ASSURES
                                                ;THAT THE LPD11 INTERFACE IS IN THE PROPER
                                                ;MODE, AND CHECKS FOR MAINTENANCE MODE.
                                                ;*************************************

                                                ;CLEAR COUNTERS AND FLAGS

001200  012702  000001          PSKIP:  MOV     #1,R2           ;SET SKIP FLAG
001204  000401                  PBEG:   BR      PSTART          ;GO TO START
001206  005002                          CLR     R2              ;RESET SKIP FLAG
001210  012706  000500          PSTART: MOV     #500,SP         ;SET STACK POINTER
001214  012777  000340  177636          MOV     #340,@PSW       ;SET PSW
001222  000005                          RESET                   ;RESET AT START OR RESTART
001224  005000                          CLR     R0
001226  005300                          DEC     R0
001230  001376                          BNE     .-2             ;DELAY FOR RESET
001232  012701  000014                  MOV     #14,R1          ;R1=NUMBER OF LOCATIONS TO CLEAR
001236  012700  001034                  MOV     #PET,R0         ;R0=START ADDR
001242  005010                          CLR     @R0
001244  005040                  PCLR:   CLR     -(R0)           ;CLEAR ALL COUNTERS
001246  005301                          DEC     R1
001250  001375                          BNE     PCLR            ;CONTINUE UNTIL DONE
001252  005702                          TST     R2              ;SEE IF SHOULD GET ADDRS
001254  001143                          BNE     PSMC            ;IF NOT; BR

                                                ;PRINT PROGRAM NAME

001256  012704  006661                  MOV     #MSG1,R4
001262  004737  006130                  JSR     PC,TTOUT        ;TYPE PROGRAM NAME
001266  012704  006710                  MOV     #MSG2,R4

                                                ;GET NEW VECTOR ADDRESS

001272  004737  006130                  JSR     PC,TTOUT        ;TYPE VECTOR REQUEST
001276  012704  001000          PV1:    MOV     #PVEC,R4        ;R4=VECTOR STORAGE ADDR
001302  012703  000003                  MOV     #3,R3           ;R3=INPUT CHAR NUMBER
001306  012737  000003  001030          MOV     #3,ROTA         ;ROTA=ROTATION FACTOR
001314  012705  177770                  MOV     #177770,R5      ;R5=MASK
001320  004737  006226                  JSR     PC,READ         ;GO READ IN VECTOR
001324  032737  000001  001000          BIT     #1,PVEC         ;TEST FOR ODD ADDR
001332  001004                          BNE     PVE             ;IF ODD: BR
001334  022737  000776  001000          CMP     #776,PVEC       ;TEST FOR TOO HIGH
001342  101005                          BHI     PRG             ;IF OK: BR
001344  012704  006750          PVE:    MOV     #MSG3,R4
001350  004737  006130                  JSR     PC,TTOUT        ;TYPE ILLEGAL VECTOR
001354  000750                          BR      PV1             ;TRY AGAIN
                                        ;GET NEW REGISTER ADDRESS
```

```
001356  012704  007004          PRG:    MOV     #MSG4,R4
001362  004737  006130                  JSR     PC,TTOUT        ;TYPE REGISTER ADDR REQUEST
001366  012704  001002          PRG1:   MOV     #PREG,R4
001372  012703  000006                  MOV     #6,R3
001376  012737  000003  001030          MOV     #3,ROTA
001404  012705  177770                  MOV     #177770,R5
001410  004737  006226                  JSR     PC,READ         ;GO READ STARTING REGISTER ADDR
001414  032737  000001  001002          BIT     #1,PREG         ;TEST FOR ODD ADDR
001422  001010                          BNE     PREGE           ;IF ODD: BR
001424  022737  160000  001002          CMP     #160000,PREG    ;TEST FOR TOO LOW
001432  101004                          BHI     PREGE           ;IF TOO LOW: BR
001434  022737  177600  001002          CMP     #177600,PREG    ;TEST FOR TOO HIGH
001442  101005                          BHI     PSET            ;IF NOT TO HIGH: BR
001444  012704  007046          PREGE:  MOV     #MSG5,R4
001450  004737  006130                  JSR     PC,TTOUT        ;TYPE ILLEGAL REGISTER ADDR
001454  000744                          BR      PRG1            ;TRY AGAIN
001456  013700  001002          PSET:   MOV     PREG,R0         ;SET P. S. STATUS ADDR IN R0
001462  012701  001062                  MOV     #PCS,R1         ;R1=LPD STATUS ADDR
                                        .REPT 4
                                        MOV     R0,(R1)+        ;SET REGISTERS
                                        ADD     #2,R0
                                        .ENDR

                                        ;REQUEST ON LINE/LPD TEST MODE

001516  012704  007103          REQ:    MOV     #MSG6,R4
001522  004737  006130                  JSR     PC,TTOUT        ;REQUEST ON LINE/LPD TEST MODE
001526  012704  007222                  MOV     #MSG6A,R4
001532  004737  006130                  JSR     PC,TTOUT        ;LOCATE SWITCH
001536  012704  001012                  MOV     #PGOF,R4
001542  012703  000001                  MOV     #1,R3
001546  012737  000003  001030          MOV     #3,ROTA
001554  012705  177770                  MOV     #177770,R5      ;SET MASK
001560  004737  006225                  JSR     PC,READ         ;AWAIT GO COMMAND

                                        ;MAINTENANCE MODE CHECK

001564  032777  004000  177264  PSMC:   BIT     #4000,@SWR      ;LOOK FOR MAINTENANCE MODE BIT
001572  001402                          BEQ     PLCC            ;IF NOT: BR
001574  000137  004C10                  JMP     PSCOP           ;ELSE GO TO MAINTENANCE MODE

                                        ;LINE COUNT TEST ENTRY CHECK

001600  032777  000400  177250  PLCC:   BIT     #400,@SWR       ;SEE IF SHOULD ENTER LINE COUNT TEST
001606  001402                          BEQ     PSDC            ;IF NOT: BR
001610  000137  004236                  JMP     PLC             ;ELSE GO TO LINE COUNT TEST

                                        ;DATA OUTPUT TEST ENTRY CHECK

001614  032777  000100  177234  PSDC:   BIT     #100,@SWR       ;LOOK FOR DATA OUTPUT SW
001622  001402                          BEQ     PTO             ;BR IF NOT SELECTED
001624  000137  004500                  JMP     PSD             ;ELSE GO TO DATA OUTPUT TEST
```

```
                                        ;****************************************
                                        ;PART 1: LOGIC TESTS
                                        ;
                                        ;THIS PART CONSISTS OF 23(8) TESTS TO
                                        ;ASSURE THAT ALL FUNCTIONS OF EACH USABLE
                                        ;BIT IN BOTH THE TESTER AND THE CONTROLLER
                                        ;STATUS REGISTERS WILL SET AND RESET FROM
                                        ;PROPER CONDITIONS AND IN THE PROPER SEQUENCE.
                                        ;THE INTERRUPT SYSTEM OF BOTH THE TESTER
                                        ;AND CONTROLLER IS ALSO TESTED TO ASSURE
                                        ;THAT EACH WILL TRAP TO THE PROPER VECTOR
                                        ;AND PRIORITY LEVEL.
                                        ;****************************************

                                        ;LOGIC TEST 00: SEE THAT ERROR BIT SET WHEN LPD IN LOCAL TEST

001630  012737  030060  001004  PT0:    MOV     #30060,PTN      ;SAVE TEST NO.
001636  005777  177220                  TST     @PCS            ;LOOK FOR BIT
001642  100404                         BMI     PT.             ;BR IF OK
001644  012703  012444                  MOV     #MSG55,R3       ;SET ERROR MSG ADDR
001650  000137  005442                  JMP     PLE             ;GO TYPE ERROR

                                        ;LOGIC TEST 01: CLEAR CONT STATUS

001654  005077  177202          PT1:    CLR     @PCS            ;CLEAR CONT STATUS
001660  005077  177202                  CLR     @PTS            ;CLEAR TESTER STATUS
001664  012737  030460  001004          MOV     #30460,PTN      ;SAVE TEST NO.
001672  004737  006452                  JSR     PC,PRSTV        ;RESET VECTORS
001676  032777  077500  177156          BIT     #77500,@PCS     ;LOOK FOR CLEAR
001704  001404                         BEQ     PT1A1           ;IF OK: BR
001706  012703  007517                  MOV     #MSG13,R3       ;SET ERROR MSG ADDR
001712  000137  005442                  JMP     PLE             ;GO TYPE ERROR

                                        ;LOGIC TEST 02: CLEAR TESTER STATUS

001716  012737  031060  001004  PT1A1:  MOV     #31060,PTN      ;SAVE TEST NO.
001724  032777  077577  177134          BIT     #77577,@PTS     ;LOOK FOR CLEAR
001732  001404                         BEQ     PT1A1A          ;IF OK: BR
001734  012703  007462                  MOV     #MSG12,R3       ;SET ERROR MSG ADDR
001740  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
```

```
                                                ;LOGIC TEST 03: CONT READY RESET FROM INIT
                                                
001744  012737  031460  001004  PT1A1A: MOV     #31460,PTN      ;SAVE TEST NO.
001752  032777  000200  177102          BIT     #200,@PCS       ;LOOK FOR CONT READY RESET
001760  001404                          BEQ     PT1A1B          ;IF OK: BR
001762  012703  011014                  MOV     #MSG41,R3       ;SET ERROR MSG ADDR
001766  000137  005442                  JMP     PLE             ;GO TO ERROR ROUTINE
                                                
                                                ;LOGIC TEST 04: TESTER READY RESET FROM INIT
                                                
002000  012737  032060  001004  PT1A1B: MOV     #32060,PTN      ;SAVE TEST NO.
002006  032777  000200  177060          BIT     #200,@PTS       ;LOOK FOR TESTER READY RESET
002014  001404                          BEQ     PT1A2           ;IF OK: BR
002016  012703  011053                  MOV     #MSG42,R3       ;SET ERROR MSG ADDR
002022  000137  005442                  JMP     PLE             ;GO TO ERROR ROUTINE
                                                
                                                ;LOGIC TEST 05: CONT READY RESET FROM CHAR LOAD
                                                
002030  012737  032460  001004  PT1A2:  MOV     #32460,PTN      ;SAVE TEST NO.
002036  012777  000000  177030          MOV     #0,@PCB         ;CLEAR BUFFER
002044  032777  000200  177020          BIT     #200,@PCS       ;LOOK FOR READY
002052  001404                          BEQ     PT1A3           ;IF OK: BR
002054  012703  007560                  MOV     #MSG14,R3       ;SET ERROR MSG ADDR
002060  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
                                                
                                                ;LOGIC TEST 06: TESTER READY SET AFTER CHAR LOAD
                                                
002064  012737  033060  001004  PT1A3:  MOV     #33060,PTN      ;SAVE TEST NO.
002072  005037  001014          PT1A3A: CLR     PTIM1           ;SET TIMER
002076  005337  001014                  DEC     PTIM1           ;DELAY
002102  001375                          BNE     PT1A3A
002104  032777  000200  176764          BIT     #200,@PTS       ;LOOK FOR TESTER READY
002112  001004                          BNE     PT1A4           ;IF OK: BR
002114  012703  007632                  MOV     #MSG15,R3       ;SET ERROR MSG ADDR
002120  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
                                                
                                                ;LOGIC TEST 07: ASSURE GO BIT RESET
                                                
002124  012737  034060  001004  PT1A4:  MOV     #34060,PTN      ;SAVE TEST NO.
002132  005277  176740                  INC     @PTS            ;SET GO
002136  012737  000100  001014          MOV     #100,PTIM1      ;SET TIME
002144  005337  001014          PT1A:   DEC     PTIM1           ;DELAY
002150  001375                          BNE     PT1A
002152  032777  000001  176716          BIT     #1,@PTS         ;LOOK FOR GO RESET
002160  001404                          BEQ     PT1B1           ;IF OK: BR
002162  012703  007701                  MOV     #MSG16,R3       ;SET ERROR MSG ADDR
002166  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
```

```
                                                ;LOGIC TEST 10: TESTER READY NOT RESET BY GO

002162  012737  030061  001004  PT1B1:  MOV     #30061,PTN      ;SAVE TEST NO.
002170  032777  000200  176670          BIT     #200,@PTS       ;LOOK FOR TESTER READY NOT RESET
002176  001004                          BNE     PT1B2           ;IF OK: BR
002200  012703  007731                  MOV     #MSG17,R3       ;SET ERROR MSG ADDR
002204  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG

                                                ;LOGIC TEST 11: CONT  READY SET FROM GO

002210  012737  030461  001004  PT1B2:  MOV     #30461,PTN      ;SAVE TEST NO.
002216  005037  001014                  CLR     PTIM1           ;SET TIMER
002222  005337  001014          PT1B2A: DEC     PTIM1           ;DELAY
002226  001375                          BNE     PT1B2A
002230  032777  000200  176624          BIT     #200,@PCS       ;LOOK FOR CONT READY SET
002236  001004                          BNE     PT1B3           ;IF OK: BR
002240  012703  007767                  MOV     #MSG20,R3       ;SET ERROR MSG ADDR
002244  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG

                                                ;LOGIC TEST 12: TESTER READY AT END OF CYCLE

002250  012737  032601  001004  PT1B3:  MOV     #32601,PTN      ;SAVE TEST NO.
002256  012777  000000  176600          MOV     #0,@PCB         ;LOAD CHAR
002264  005037  001014                  CLR     PTIM1           ;SET TIME
002270  005337  001014          PT1B:   DEC     PTIM1           ;DELAY
002274  001375                          BNE     PT1B
002276  032777  000200  176562          BIT     #200,@PTS       ;LOOK FOR TESTER READY
002304  001004                          BNE     PT1C            ;IF OK: BR
002306  012703  010032                  MOV     #MSG21,R3       ;SET ERROR MSG ADDR
002312  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG

                                                ;LOGIC TEST 13: CONT READY RESET AT END OF CYCLE

002316  012737  031461  001004  PT1C:   MOV     #31461,PTN      ;SAVE TEST NO.
002324  005337  001014                  DEC     PTIM1
002330  001372                          BNE     PT1C            ;DELAY
002332  032777  000200  176522          BIT     #200,@PCS       ;LOOK FOR CONT READY
002340  001404                          BEQ     PT1E            ;IF OK: BR
002342  012703  010101                  MOV     #MSG22,R3       ;SET ERROR MSG ADDR
002346  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
```

```
                                                ;LOGIC TEST 14: TESTER INTERRUPT
                
002352  012737  031461  001004  PT1E:   MOV     #31461,PTN      ;SAVE TEST NO.
002360  005077  176502                  CLR     @PTS            ;CLEAR TESTER
002364  005077  176472                  CLR     @PCS            ;CLEAR CONTROLLER
002370  013700  001000                  MOV     PVEC,R0         ;R0 = STARTING VECTOR
002374  022020                          CMP     (R0)+,(R0)+     ;POINT TO TESTER VECTOR
002376  012720  006362                  MOV     #PINT,(R0)+     ;SET VECTOR ADDR
002402  012720  000340                  MOV     #340,(R0)+      ;SET PSW
002406  052777  000100  176452          BIS     #100,@PTS       ;SET TESTER INTERRUPT ENABLE
002414  005037  001014                  CLR     PTIM1           ;SET TIME
002420  012737  000010  001016          MOV     #10,PTIM2       ;SET TIME
002426  012737  002472  001032          MOV     #PT1FA,PINTR    ;SET RETURN
002434  005077  176420                  CLR     @PSW            ;SET PSW TO LOWEST PRIORITY
002440  012777  000000  176416          MOV     #0,@PCB         ;LOAD CHAR
002446  005337  001014          PT1F:   DEC     PTIM1           ;DELAY
002452  001375                          BNE     PT1F
002454  005337  001016                  DEC     PTIM2           ;DELAY
002460  001372                          BNE     PT1F
002462  012703  010150                  MOV     #MSG24,R3       ;SET ERROR MSG ADDR
002466  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
                
                                                ;LOGIC TEST 15: CONT INTERRUPT
                
002472  012737  032461  001004  PT1FA:  MOV     #32461,PTN      ;SAVE TEST NO.
002500  005077  176362                  CLR     @PTS            ;CLEAR TESTER INT ENABLE
002504  005037  001014                  CLR     PTIM1
002510  005337  001014          PT1FB:  DEC     PTIM1
002514  001375                          BNE     PT1FB           ;AWAIT COMPLETE FROM TESTER INTERRUPT
002516  005337  001014          PT1FC:  DEC     PTIM1
002522  001375                          BNE     PT1FC
002524  004737  005452                  JSR     PC,PRSTV        ;RESET VECTORS
002530  013700  001000                  MOV     PVEC,R0         ;R0 = STARTING VECTOR
002534  012720  006362                  MOV     #PINT,(R0)+     ;SET VECTOR ADDR
002540  012720  000340                  MOV     #340,(R0)+      ;SET PSW
002544  052777  000100  176310          BIS     #100,@PCS       ;SET CONT INTERRUPT ENABLE
002552  012737  000010  001016          MOV     #10,PTIM2       ;SET TIME
002560  012737  002622  001032          MOV     #PT1H,PINTR     ;SET RETURN
002566  005077  176266                  CLR     @PSW            ;SET PSW TO LOWEST PRIORITY
002572  005277  176270                  INC     @PTS            ;SET GO
002576  005337  001014          PT1G:   DEC     PTIM1
002602  001375                          BNE     PT1G            ;DELAY
002604  005337  001016                  DEC     PTIM2
002610  001372                          BNE     PT1G            ;DELAY
002612  012703  010211                  MOV     #MSG25,R3       ;SET ERROR MSG ADDR
002616  000137  005442                  JMP     PLE             ;GO TYPE ERROR MSG
```

```
 793                                                    ;LOGIC TEST 16: TESTER INT ENABLE OFF
 794
 795  002622  012737  033061  001004  PT1H:   MOV     #33061,PTN      ;SAVE TEST NO.
 796  002630  005077  176226                  CLR     @PCS
 797  002634  005077  176226                  CLR     @PTS
 798  002640  005005                          CLR     R5              ;CLEAR FLAG
 799  002642  012737  002754  001010          MOV     #PT1M,PITA      ;SET POINTER
 800  002650  012704  002744                  MOV     #PT1L,R4        ;SET RETURN ADDRESS
 801  002654  005003                          CLR     R3
 802  002656  013700  001000                  MOV     PVEC,R0         ;R0=STARTING VECTOR ADDRESS
 803  002662  022020                          CMP     (R0)+,(R0)+     ;POINT TO TESTER VECTOR
 804  002664  012720  006350          PT1HA:  MOV     #PINTS,(R0)+    ;SET VECTOR TO SERVICE ROUTINE
 805  002670  012720  000340                  MOV     #340,(R0)+      ;SET PSW TO PRIORITY 7
 806  002674  010437  001032          PT1HB:  MOV     R4,PINTR        ;SET INTERRUPT RETURN
 807  002700  010377  176154                  MOV     R3,@PSW         ;SET PSW
 808  002704  005037  001014                  CLR     PTIM1           ;CLEAR TIMER
 809  002710  012777  000000  176146          MOV     #0,@PCB         ;LOAD CHAR
 810  002716  005337  001014          PT1J:   DEC     PTIM1
 811  002722  001375                          BNE     PT1J            ;DELAY
 812  002724  005337  001014          PT1K:   DEC     PTIM1
 813  002730  001375                          BNE     PT1K            ;DELAY
 814  002732  012777  000340  176120          MOV     #340,@PSW       ;RESET PSW TO PRIORITY 7
 815  002740  000177  176044                  JMP     @PITA           ;CONTINUE TO NEXT TEST
 816  002744  012703  010467          PT1L:   MOV     #MSG34,R3       ;TYPE ERROR
 817  002750  000137  005442                  JMP     PLE
 818
 819                                                    ;LOGIC TEST 17: TESTER INT AT HIGHER LEVEL
 820
 821  002754  012737  033461  001004  PT1M:   MOV     #33461,PTN      ;SAVE TEST NO.
 822  002762  005077  176102                  CLR     @PTB
 823  002766  005077  176074                  CLR     @PTS
 824  002772  005077  176064                  CLR     @PCS
 825  002776  005705                          TST     R5              ;SEE IF TESTER COMPLETE
 826  003000  001025                          BNE     PT1P            ;IF SO:BR
 827  003002  005205                          INC     R5              ;SET FLAG
 828  003004  005337  001014          PT1M1:  DEC     PTIM1
 829  003010  001375                          BNE     PT1M1           ;DELAY
 830  003012  005077  176046                  CLR     @PCB            ;LOAD CHAR
 831  003016  052777  000100  176042          BIS     #100,@PTS       ;SET TESTER INTERRUPT ENABLE
 832  003024  012777  000200  176026          MOV     #200,@PSW       ;PSW SETTING = PRIORITY 4
 833  003032  012737  003044  001032          MOV     #PT1N,PINTR     ;LOAD RETURN ADDRESS
 834  003040  000137  002716                  JMP     PT1J            ;DELAY FOR INTERRUPT
 835  003044  012703  010544          PT1N:   MOV     #MSG35,R3
 836  003050  000137  005442                  JMP     PLE             ;GO TO ERROR ROUTINE
 837
```

```
                                                ;LOGIC TEST 20: CNT INT ENABLE OFF

003054  012737  030062  001004  PT1P:   MOV     #30062,PTN      ;SAVE TEST NO.
003062  005077  176002                  CLR     @PTB
003066  005077  175774                  CLR     @PTS
003072  005077  175764                  CLR     @PCS
003076  005005                         CLR     R5              ;CLEAR FLAG
003100  013700  001000                  MOV     PVEC,R0         ;R0=VECTOR ADDRESS
003104  005003                         CLR     R3              ;R3=PSW
003106  012704  003154                  MOV     #PT1R,R4        ;R4=INTERRUPT RETURN ADDR
003112  012737  003164  001010          MOV     #PT1S,PITA      ;SET CONTINUE ADDR
003120  012720  006350                  MOV     #PINTS,(R0)+    ;SET VECTOR ADDR
003124  012720  000340                  MOV     #340,(R0)+      ;SET PSW
003130  010437  001032          PT1Q:   MOV     R4,PINTR        ;SET RETURN ADDR
003134  010377  175720                  MOV     R3,@PSW         ;SET PSW
003140  005037  001014                  CLR     PTIM1           ;SET TIMER
003144  005277  175716                  INC     @PTS            ;SET GO
003150  000137  002716                  JMP     PT1J            ;TEST CONT INTERRUPT
003154  012703  010617          PT1R:   MOV     #MSG36,R3
003160  000137  005442                  JMP     PLE             ;GO TO ERROR ROUTINE

                                                ;LOGIC TEST 21: CONT INT AT HIGHER LEVEL

003164  012737  030462  001004  PT1S:   MOV     #30462,PTN      ;SAVE TEST NO.
003172  005077  175672                  CLR     @PTB
003176  005077  175664                  CLR     @PTS
003202  005077  175654                  CLR     @PCS
003206  005705                         TST     R5              ;SEE IF DONE BOTH TESTS
003210  001016                         BNE     PEBT            ;IF SO:BR
003212  005205                         INC     R5              ;SET FLAG
003214  052777  000100  175640          BIS     #100,@PCS       ;SET CONTROLLER INTERRUPT ENABLE
003222  012703  000200                  MOV     #200,R3         ;SET PSW
003226  012704  003236                  MOV     #PT1T,R4        ;SET INTERRUPT RETURN ADDR
003232  000137  003130                  JMP     PT1Q            ;TEST CONTROLLER
003236  012703  010700          PT1T:   MOV     #MSG37,R3
003242  000137  005442                  JMP     PLE             ;GO TO ERROR ROUTINE

                                                ;LOGIC TEST22: ERROR BIT SET AND RESET

003246  012737  031062  001004  PEBT:   MOV     #31062,PTN      ;SAVE TEST NUMBER
003254  004737  006452                  JSR     PC,PRSTV        ;RESET VECTORS.
003260  032777  002000  175570          BIT     #2000,@SWR      ;SEE IF ERROR BIT TEST
003266  001002                         BNE     PEBTO           ;IF SO: BR
003270  000137  003464                  JMP     PIDBT           ;ELSE GO TO 'ID' TEST
003274  005037  001034          PEBTO:  CLR     PET             ;TEST FOR RESET
003300  012704  011141                  MOV     #MSG44,R4       ;REQUEST ON LINE - NORMAL MODE
003304  004737  006130                  JSR     PC,TTOUT
003310  012704  007222                  MOV     #MSG6A,R4       ;LOCATE SWITCH 8
003314  004737  003402                  JSR     PC,PEBT1
003320  012737  100000  001034          MOV     #100000,PET     ;TEST FOR SET
003326  012704  011310                  MOV     #MSG45,R4       ;REQUEST OFF LINE - NORMAL MODE
003332  004737  003402                  JSR     PC,PEBT1
003336  012704  011464                  MOV     #MSG46,R4       ;REQUEST ON LINE - NORMAL MODE - PWR DOWN
003342  004737  003402                  JSR     PC,PEBT1
003346  012704  007103                  MOV     #MSG6,R4        ;REQUEST ON LINE - LPD TEST MODE
003352  004737  006130                  JSR     PC,TTOUT
```

```
  894  003356  012704  011674                    MOV     #MSG46A,R4      ;REQUST POWER ON
  895  003362  004737  003402                    JSR     PC,PEBT1
  896  003366  012704  011764                    MOV     #MSG47,R4
  897  003372  004737  006130                    JSR     PC,TTOUT        ;TYPE ERROR BIT OK
  898  003376  000137  003464                    JMP     PIDBT           ;GO TO 'ID' TEST
  899  003402  004737  006130          PEBT1:    JSR     PC,TTOUT        ;TYPE MSG
  900  003406  013704  001012                    MOV     PGOF,R4
  901  003412  012703  000001                    MOV     #1,R3
  902  003416  012737  000003  001030           MOV     #3,ROTA
  903  003424  012705  177770                    MOV     #177770,R5
  904  003430  004737  006226                    JSR     PC,READ         ;AWAIT GO REPLY
  905  003434  017700  175422          PEBT2:    MOV     @PCS,R0         ;SAVE CONTROLLER STATUS
  906  003440  042700  077777                    BIC     #77777,R0       ;MASK ERROR BIT
  907  003444  020037  001034                    CMP     R0,PET          ;SEE IF ERROR BIT IN PROPER STATE
  908  003450  001404                            BEQ     PEBT3           ;IF SO: BR
  909  003452  012703  010256                    MOV     #MSG26,R3
  910  003456  000137  005442                    JMP     PLE             ;GO TYPE ERROR MSG
  911  003462  000207                  PEBT3:    RTS     PC
  912
  913                                            ;LOGIC TEST 23: READ 'ID' BITS
  914
  915  003464  012737  031462  001004  PIDBT:    MOV     #31462,PTN      ;SAVE TEST NUMBER
  916  003472  017703  175364                    MOV     @PCS,R3         ;GET STATUS
  917  003476  042703  177770                    BIC     #177770,R3      ;GET RID OF THE READY & ERROR BITS
  918  003502  023703  001036                    CMP     ID,R3           ;LOOK AT 'ID' BITS
  919  003506  001404                            BEQ     PDAT            ;IF OK: GO TO DATA TEST
  920  003510  012703  007411                    MOV     #MSG11,R3       ;SET ERROR MSG ADRS
  921  003514  000137  005442                    JMP     PLE             ;GO TO ERROR ROUTINE
```

```
                                        ;*************************************
                                        ;PART 2: DATA TESTS
                                        ;
                                        ;THIS PART CONSISTS OF TWO TESTS TO
                                        ;ASSURE THAT ALL COMBINATIONS OF DATA
                                        ;CAN BE WRAPPED AROUND FROM CONTROLLER
                                        ;BUFFER TO TESTER BUFFER.
                                        ;TEST 1 WRAPS ALL ONES, ALL ZEROS, A
                                        ;WALKING ONE IN A FIELD OF ZEROS, AND
                                        ;A WALKING ZERO IN A FIELD OF ONES.
                                        ;TEST 2 WRAPS ALL POSSIBLE BIT COMBINATIONS
                                        ;OF THE 8 BIT CHARACTER.
                                        ;EACH TEST PATTERN IN TEST ONE WILL BE
                                        ;REPEATED 512 TIMES; EACH ALL BIT COMBINATION
                                        ;IN TEST TWO WILL BE REPEATED 6 TIMES.
                                        ;TO PREVENT REPETITION OF OF PATTERNS
                                        ;USE CONSOLE SWITCH 12.
                                        ;*************************************


                                        ;DATA TEST 1: ONES AND ZEROS WALK

003520  012701  006612          PDAT:   MOV     #PPATS,R1       ;R1=START OF DATA TABLE
003524  012737  001000  001020  PDAT0:  MOV     #1000,PIC
003532  005077  175324                  CLR     @PCS
003536  005077  175324                  CLR     @PTS
003542  005077  175322                  CLR     @PTB
003546  005037  001022                  CLR     PCHAR
003552  022701  006634                  CMP     #PPATE,R1       ;LOOK FOR END OF DATA TABLE
003556  001437                          BEQ     PDAT4           ;IF SO:BR
003560  112137  001022                  MOVB    (R1)+,PCHAR     ;SET CHAR
003564  113777  001022  175272  PDAT1:  MOVB    PCHAR,@PCB      ;SET BUFFER
003572  005037  001014                  CLR     PTIM1           ;SET TIMER
003576  032777  000200  175262  PDAT2:  BIT     #200,@PTS       ;AWAIT TESTER READY
003604  001005                          BNE     PDAT2A          ;IF HAVE: BR
003606  005337  001014                  DEC     PTIM1           ;DELAY
003612  001371                          BNE     PDAT2
003614  000137  003720                  JMP     PDAT7A          ;TYPE ERROR
003620  117700  175244          PDAT2A: MOVB    @PTB,R0         ;R0=INPUT CHAR
003624  120037  001022                  CMPB    R0,PCHAR        ;COMPARE INPUT/OUTPUT
003630  001402                          BEQ     PDAT3           ;IF OK: BR
003632  004737  005530                  JSR     PC,PDE          ;GO TYPE ERROR
003636  032777  010000  175212  PDAT3:  BIT     #10000,@SWR     ;LOOK FOR INHIBIT ITERATIONS
003644  001327                          BNE     PDAT0           ;IF SO: BR
003646  005337  001020                  DEC     PIC             ;DECREMENT ITERATION COUNTER
003652  001344                          BNE     PDAT1           ;IF NOT DONE: BR
003654  000723                          BR      PDAT0           ;ELSE GET NEXT PATTERN
```

```
                                                ;DATA TEST 2: ALL CHARACTERS
                                                
003656  012737  000006  001020  PDAT4:  MOV     #6,PIC          ;SET ITERATION CNTR
003664  005037  001022          PDAT5:  CLR     PCHAR
003670  013777  001022  175166  PDAT6:  MOV     PCHAR,@PCB      ;SET BUFFER
003676  005037  001014                  CLR     PTIM1           ;CLEAR TIMER
003702  032777  000200  175156  PDAT7:  BIT     #200,@PTS       ;LOOK FOR READY BIT  SET
003710  001011                          BNE     PDAT7B          ;IF OK: BR
003712  005337  001014                  DEC     PTIM1
003716  001371                          BNE     PDAT7           ;AWAIT READY
003720  012704  010426          PDAT7A: MOV     #MSG33,R4
003724  004737  006130                  JSR     PC,TTOUT        ;TYPE NO READY IN DATA TEST
003730  000137  005702                  JMP     PEND            ;GO TO END ROUTINE
003734  117700  175130          PDAT7B: MOVB    @PTB,R0         ;R0=INPUT CHAR
003740  120037  001022                  CMPB    R0,PCHAR        ;TEST CHAR
003744  001402                          BEQ     PDAT10          ;IF OK: BR
003746  004737  005530                  JSR     PC,PDE          ;GO TYPE ERROR
003752  005237  001022          PDAT10: INC     PCHAR           ;SET NEW CHAR
003756  032737  000400  001022          BIT     #400,PCHAR      ;SEE IF DONE ALL
003764  001741                          BEQ     PDAT6           ;IF NOT: BR
003766  032777  010000  175062          BIT     #10000,@SWR     ;LOOK FOR INHIBIT ITERATIONS
003774  001003                          BNE     PDAT11          ;IF SO: BR
003776  005337  001020                  DEC     PIC             ;LOOK FOR COMPLETE ITERATIONS
004002  001330                          BNE     PDAT5           ;IF NOT: BR
004004  000137  005702          PDAT11: JMP     PEND            ;GO TO END ROUTINE
```

```
                                        ;************************************
                                        ;PART 3: MAINTENANCE MODE
                                        ;
                                        ;THE MAINTENANCE MODE, ENTERED FROM
                                        ;START VIA CONSOLE SWITCH 11, IS TO
                                        ;PROVIDE A MEANS FOR DEBUGGING AND
                                        ;TIME DELAY SETTING IN A REPETATIVE
                                        ;LOOP. A FULL OPERATION IS ATTEMPTED
                                        ;ON EACH LOOP WITH DATA CHARACTERS
                                        ;BEING TAKEN FROM CONSOLE SWITCHES
                                        ;0 - 7. THE DATA IS COMPIMENTED ON
                                        ;EVERY OTHER PASS IN ORDER TO PROVIDE
                                        ;TRANSITIONS THROUGH THE LOGIC
                                        ;DATA MAY ALSO BE CHANGED AT THE CONSOLE
                                        ;DYNAMICALLY.
                                        ;ONLY ONE OF THE INTERRUPTS WILL BE
                                        ;RECOGNIZED DEPENDING ON THE SETTING
                                        ;OF CONSOLE SWITCH 9 AND MAY BE CHANGED
                                        ;DYNAMICALLY.
                                        ;************************************

004010  017702  175042          PSCOP:  MOV     @SWR,R2         ;R2 = DATA FROM SWITCHES
004014  010203                          MOV     R2,R3
004016  127702  175034          PSCOP0: CMPB    @SWR,R2         ;SEE IF NEW DATA
004022  001372                          BNE     PSCOP           ;IF SO: BR
004024  012777  000340  175026          MOV     #340,@PSW       ;RESET PSW TO PROIRTIY 7
004032  005077  175030                  CLR     @PTS            ;CLEAR STATUS REGISTERS
004036  005077  175020                  CLR     @PCS
004042  004737  006452                  JSR     PC,PRSTV        ;RESET VECTORS
004046  032777  001000  175002          BIT     #1000,@SWR      ;SEE IF TESTER INTERUPT
004054  001012                          BNE     PSCOP1          ;IF SO: BR
004056  013700  001000                  MOV     PVEC,R0         ;R0 = STARTING VECTOR
004062  012720  006350                  MOV     #PINTS,(R0)+    ;SET VECTOR FOR MAINTENANCE MODE TRAP
004066  012720  000340                  MOV     #340,(R0)+
004072  052777  000100  174762          BIS     #100,@PCS       ;SET CONTROLLER INTERRUPT ENABLE
004100  000412                          BR      PSCOP2
004102  052777  000100  174756  PSCOP1: BIS     #100,@PTS       ;SET TESTER INTERRUPT ENABLE
004110  013700  001000                  MOV     PVEC,R0         ;R0 = STARTING VECTOR
004114  022020                          CMP     (R0)+,(R0)+     ;POINT TO TESTER VECTOR
004116  012720  006350                  MOV     #PINTS,(R0)+    ;SET VECTOR ADDR TO MAINTENANCE MODE TRAP
004122  012720  000340                  MOV     #340,(R0)+
004126  012737  004176  001032  PSCOP2: MOV     #PSCOP4,PINTR   ;SET RETURN ADDR
004134  012737  000010  001016          MOV     #10,PTIM2       ;SET DELAY
004142  005037  001014                  CLR     PTIM1           ;SET DELAY
004146  110377  174712                  MOVB    R3,@PCB         ;LOAD CHAR
004152  005277  174710                  INC     @PTS            ;SET GO BIT
004156  005077  174676                  CLR     @PSW            ;SET TO PRIORITY 0
```

```
1042  004162  005337  001014          PSCOP3: DEC     PTIM1
1043  004166  001375                          BNE     PSCOP3          ;DELAY FOR INTERRUPT
1044  004170  005337  001016                  DEC     PTIM2
1045  004174  001372                          BNE     PSCOP3
1046  004176  005103                  PSCOP4: COM     R3              ;COMPLEMENT DATA
1047  004200  032777  001000  174650          BIT     #1000,@SWR      ;SEE IF ON TESTER INTERRUPT
1048  004206  001703                          BEQ     PSCOPO          ;IF NOT: BR
1049  004210  005037  001014                  CLR     PTIM1
1050  004214  032777  000200  174644  PSCOP5: BIT     #200,@PTS       ;LOOK FOR TESTER READY
1051  004222  001275                          BNE     PSCOPO          ;IF SO: BR
1052  004224  005337  001014                  DEC     PTIM1
1053  004230  001371                          BNE     PSCOP5          ;DELAY FOR COMPLETE  FROM  TESTER
1054  004232  000137  004016                  JMP     PSCOPO          ;CONTINUE
```

```
                                        ;****************************************
                                        ;PART 4: LINE COUNT TEST
                                        ;
                                        ;THE LINE COUNT TEST, ENTERED VIA
                                        ;CONSOLE SWITCH 8, WILL PROVIDE A
                                        ;PRINTOUT OF THE DATA TRANSFER RATE
                                        ;IN LINES PER SECOND.  EACH PASS
                                        ;OF THE LINE COUNT TEST WILL COUNT
                                        ;THE NUMBER OF 37 CHARACTER GROUPS
                                        ;AND PRINT OUT THE AMOUNT IN
                                        ;DECIMAL FORM.
                                        ;SW  1 SET IF A 50HZ PDP-11/45
                                        ;SW  8 SET TO ENTER AND REMAIN IN THIS TEST
                                        ;SW 13 SET FOR CONTINUOUS LOOPING OF THIS TEST
                                        ;NO OTHER SWITCHES HAVE EFFECT.
                                        ;****************************************

004236  005077  174624        PLC:    CLR     @PTS            ;CLEAR TESTER STATUS
004242  005077  174616                CLR     @PCB            ;CLR TESTER FLAG
004246  005077  174610                CLR     @PCS            ;CLEAR CONTROLLER STATUS
004252  005003                        CLR     R3              ;CLR CHARACTER COUNTER
004254  012737  000074  004476        MOV     #74,TIME        ;SET TIME FOR 1 SECOND @ 60HZ
004262  032777  000002  174566        BIT     #2,@SWR         ;SEE IF 50HZ
004270  001403                        BEQ     1$              ;BR IF NOT
004272  012737  000062  004476        MOV     #62,TIME        ;SET TIME FOR 1 SECOND @ 50HZ
004300  013700  001000        1$:     MOV     PVEC,R0         ;GET VECTOR ADRS
004304  062700  000004                ADD     #4,R0           ;POINT TO TESTER VECTOR ADRS
004310  012720  004400                MOV     #DINT,(R0)+     ;SET UP TESTER INT SERVICE ADRS
004314  012720  000340                MOV     #340,(R0)+      ;SET PSW TO LEVEL 7 WHEM INTERRUPTED
004320  013700  004476                MOV     TIME,R0         ;SET UP 1 SECOND TIMER VALUE
004324  005077  174524                CLR     @LKS            ;CLR LINE CLOCK
004330  032777  000200  174516  2$:    BIT     #200,@LKS       ;SYNC ON LINE FREQ
004336  001774                        BEQ     2$              ;WAIT FOR CLOCK TICK
004340  012777  000100  174506        MOV     #100,@LKS       ;CLR TICK AND ENABLE CLOCK INT
004346  052777  000100  174512        BIS     #100,@PTS       ;SET TESTER INT ENABLE
004354  005277  174506                INC     @PTS            ;SET GO BIT (TESTER)
004360  005077  174474                CLR     @PSW            ;ALLOW ALL INTERRUPTS
004364  000001                3$:     WAIT                    ;AWAIT INTERRUPTS
004366  000137  004364                JMP     3$              ;RETURNS HERE AFTER INTERRUPT

                                        ;CLOCK TIMER INTERRUPT HANDLER

004372  005300                TINT:   DEC     R0              ;SEE IF DONE 1 SECOND
004374  001407                        BEQ     PLCP            ;IF SO: BR
004376  000002                        RTI                     ;ELSE RETURN

                                        ;DATA INTERRUPT HANDLER

004400  005203                DINT:   INC     R3              ;BUMP CHARACTER COUNTER
004402  010377  174456                MOV     R3,@PCB         ;CLR TESTER FLAG & XMIT CHAR
004406  005277  174454                INC     @PTS            ;SET GO
004412  000002                        RTI                     ;RETURN
```

```
                                                        ;LINE COUNT PRINT ROUTINE
                                                        ;
004414  022626                        PLCP:   CMP     (R6)+,(R6)+     ;FIX STACK SINCE NO RETURN
004416  005077  174432                        CLR     @LKS            ;CLEAR CLOCK
004422  005077  174440                        CLR     @PTS            ;CLEAR TESTER
004426  005077  174430                        CLR     @PCS            ;CLEAR CONTROLLER
004432  005002                        CLR     R2              ;CLR HIGH ORDER WORD OF 32 BIT DIVIDEND
004434  071237  004474                        DIV     LINE,R2         ;DIVIDE R2 & R3 BY CHARACTER PER LINE VALUE
004440  010203                        MOV     R2,R3           ;SET UP FOR DECPRT ROUTINE
004442  012704  011114                        MOV     #MSG43,R4
004446  004737  006130                        JSR     PC,TTOUT        ;PRINT HEADER
004452  004737  005730                        JSR     PC,DECPRT       ;PRINT LINE AMOUNT IN DECIMAL
004456  032777  020000  174372                BIT     #20000,@SWR     ;SEE IF CONTINUOUS MODE
004464  001001                        BNE     PLCG            ;IF SO: BR
004466  000000                        HALT
004470  000137  001600              PLCG:   JMP     PLCC            ;CONTINUE
                                                        ;
004474  000045                        LINE:   45                      ;DIVISOR FOR 37 CHARACTER PER LINE
004476  000000                        TIME:   0                       ;TIMER VALUE FOR 1 SECOND AT 50 OR 60 HZ
```

```
                                        ;*****************************************
                                        ;PART 5: DATA OUTPUT TEST
                                        ;
                                        ;TAKES THE CONTENTS OF 'OCTBUF' PREVIOUSLY
                                        ;LOADED VIA THE TTY OR TYPESET READER 0
                                        ;AND SENDS THIS DATA TO THE PHOTO-COMP MACHINE.
                                        ;THE FOLLOWING KEYS ON THE KEYBOARD ARE USED TO
                                        ;CONTROL THE TRANSMISSION OF DATA TO THE PHOTO-
                                        ;COMP MACHINE:
                                        ;'T'-TRANSMIT THE CONTENTS OF THE BUFFER PREVIOUSLY TYPED.
                                        ;'S'-STOP THE TRANSFER OF DATA.
                                        ;'C'-CONTINUE THE TRANSFER WHERE INTERRUPTED.
                                        ;'I'-INPUT A NEW DATA BUFFER.
                                        ;*****************************************

1141  004500  012704  012003           PSD:    MOV     #MSG50,R4       ;SET UP MSG ADRS
1142  004504  004737  006130                   JSR     PC,TTOUT        ;GO IDENTIFY TEST & INSTRUCTIONS
1143  004510  012705  012000           SETBUF: MOV     #OCTBUF,R5      ;SET UP R5 WITH STARTING ADRS OF 'OCTBUF'
1144  004514  012737  000001  005420           MOV     #1,LINCNT       ;SET UP 'LINCNT' TO 1
1145  004522  037727  174330  000001           BIT     @SWR,#1         ;IS INPUT TO BE FROM READER?
1146  004530  001445                           BEQ     NXLIN           ;BR IF INPUT FROM TTY
1147
1148                                  ;THIS CODE GETS THE INPUT DATA FROM THE TYPESET READER 0
1149
1150  004532  005377  174302           RDRLK1: INC     @PRS            ;ADVANCE READER
1151  004536  105777  174302           RDRLK:  TSTB    @TKS            ;LOOK FOR A KEY
1152  004542  100021                           BPL     2$              ;BR IF NO KEY
1153  004544  017737  174276  005422           MOV     @TKB,TEMP       ;GET AND SAVE COMMAND
1154  004552  023727  005422  000311           CMP     TEMP,#311       ;SEE IF A NEW BUFFER REQUESTED
1155  004560  001003                           BNE     1$              ;BR IF NOT
1156  004562  004737  005372                   JSR     PC,TYPCMD       ;GO TYPE 'I'
1157  004566  000750                           BR      SETBUF          ;BR TO GET NEW DATA
1158  004570  023727  005422  000324   1$:     CMP     TEMP,#324       ;SEE IF XMIT REQUESTED
1159  004576  001003                           BNE     2$              ;BR IF NOT
1160  004600  004737  005372                   JSR     PC,TYPCMD       ;GO TYPE 'T'
1161  004604  000557                           BR      XFER1           ;GO XMIT DATA
1162  004606  105777  174226           2$:     TSTB    @PRS            ;IS READER DONE?
1163  004612  100351                           BPL     RDRLK           ;BR IF NOT
1164  004614  117725  174222                   MOVB    @PRB,(R5)+      ;STORE DATA IN BUFFER
1165  004620  005237  005420                   INC     LINCNT          ;RECORD THIS DATA
1166  004624  022705  020000                   CMP     #20000,R5       ;ARE WE AT THE END OF MEM?
1167  004630  001340                           BNE     RDRLK1          ;BR IF NOT
1168  004632  012704  012334                   MOV     #MSG52,R4       ;SET UP BUFFER FULL MSG
1169  004636  004737  006130                   JSR     PC,TTOUT        ;GO TYPE MSG
1170  004642  000733                           BR      RDRLK1          ;SHOULD RESPOND WITH A 'T'
```

```
1171                                                    ;THIS CODE CONTROLS LINE NOS. ON KEYBOARD INPUT
1172
1173  004644  012704  012442          NXLIN:  MOV     #MSG54,R4       ;SET UP FOR CR/LF TYPE
1174  004650  004737  006130                  JSR     PC,TTOUT        ;GO SET MARGIN
1175  004654  012702  005434                  MOV     #TEMBUF,R2      ;SET UP TEMP DIGIT COUNTER
1176  004660  012737  000004  005416          MOV     #4,CCNT         ;SET UP DIGIT COUNT
1177  004666  013703  005420                  MOV     LINCNT,R3       ;GET CURRENT LINE COUNT
1178  004672  004737  005730                  JSR     PC,DECPRT       ;GO TYPE IT
1179  004676  012737  000240  001024          MOV     #240,TOB        ;SET UP FOR 'SPACE'
1180  004704  004737  006210                  JSR     PC,TOG          ;GO TYPE IT
1181
1182                                                    ;GET CHARACTER FROM KEYBOARD
1183
1184  004710  105777  174130          TTI:    TSTB    @TKS            ;LOOK FOR CHARACTER
1185  004714  100375                          BPL     TTI             ;WAIT FOR CHARACTER
1186  004716  017737  174124  001024          MOV     @TKB,TOB        ;GET CHAR AND SAVE AT 'TOB'
1187  004724  023727  001024  000260          CMP     TOB,#260        ;SEE IF LESS THAN ASCII #260
1188  004732  100424                          BMI     CNTRCK          ;BR IF SO
1189  004734  023727  001024  000270          CMP     TOB,#270        ;SEE IF GREATER THAN ASCII #267
1190  004742  100057                          BPL     ENDCK           ;BR IF SO
1191  004744  004737  006210                  JSR     PC,TOG          ;GO TYPE DIGIT
1192  004750  005337  005416                  DEC     CCNT            ;SEE IF OVER 3 INPUT DIGITS
1193  004754  001406                          BEQ     TYPER           ;BR IF SO
1194  004756  042737  177770  001024          BIC     #177770,TOB     ;GET RID OF ASCII
1195  004764  013722  001024                  MOV     TOB,(R2)+       ;SAVE DIGIT IN 'OCTBUF'
1196  004770  000747                          BR      TTI     ;GO GET NEXT DIGIT
```

```
1197                                                    ;THIS CODE REPORTS AN INPROPER INPUT RESPONSE
1198
1199  004772  012704  012331          TYPER:  MOV     #MSG51,R4       ;GO SET UP FOR ??
1200  004776  004737  006130                  JSR     PC,TTOUT        ;GO TYPE IT
1201  005002  000720                          BR      NXLIN           ;GO GET SAME LINE AGAIN
1202
1203                                                    ;THIS CODE ASSEMBLES THE 3 DIGIT ASCII INTO
1204                                                    ;3 DIGIT OCTAL AND STORES IT IN THE 'OCTBUF'
1205
1206  005004  023727  001024  000215  CNTRCK: CMP     TOB,#215        ;IS IT THE LINE TERMINATOR?
1207  005012  001367                          BNE     TYPER           ;BR IF NOT
1208  005014  014237  005422                  MOV     -(R2),TEMP      ;PUT LSD IN TEMP
1209  005020  006342                          ASL     -(R2)           ;ADJUST FOR MIDDLE DIGIT
1210  005022  006312                          ASL     (R2)            ;
1211  005024  006312                          ASL     (R2)            ;
1212  005026  061237  005422                  ADD     (R2),TEMP       ;ADD MIDDLE DIGIT TO TEMP
1213  005032  022742  000004                  CMP     #4,-(R2)        ;SEE IF MSD IS LARGER THAN 3
1214  005036  003755                          BLE     TYPER           ;BR IF NUMBER 400 OR OVER
1215  005040  000312                          SWAB    (R2)            ;MAKE MSD EASIER TO SHIFT
1216  005042  006212                          ASR     (R2)            ;ADJUST FOR OVER SHIFT
1217  005044  006212                          ASR     (R2)            ;
1218  005046  061237  005422                  ADD     (R2),TEMP       ;ADD MSD TO 'TEMP'
1219  005052  113725  005422                  MOVB    TEMP,(R5)+      ;STORE OCTAL VALUE TYPED
1220  005056  020527  020000                  CMP     R5,#20000       ;SEE IF BUFFER FULL
1221  005062  001004                          BNE     BUFOK           ;BR IF NOT
1222  005064  012704  012334                  MOV     #MSG52,R4       ;SET UP BUFFER FULL MSG
1223  005070  004737  006130                  JSR     PC,TTOUT        ;GO TYPE IT
1224  005074  005237  005420          BUFOK:  INC     LINCNT          ;ADVANCE TO NEXT LINE
1225  005100  000661                          BR      NXLIN           ;GO GET NEXT LINE
1226
1227                                                    ;THIS CODE LOOKS FOR THE KEYS 'T' OR 'I' DURING THE
1228                                                    ;OCTAL INPUT ROUTINE AND CHECKS THAT IT FOLLOWED A 'CR'
1229
1230  005102  023727  001024  000311  ENDCK:  CMP     TOB,#311        ;SEE IF CALLING FOR NEW DATA
1231  005110  001004                          BNE     1$              ;BR IF NOT
1232  005112  004737  006210                  JSR     PC,TOG          ;INDICATE 'INPUT'
1233  005116  000137  004510                  JMP     SETBUF          ;GO GET NEW DATA
1234  005122  023727  001024  000324  1$:     CMP     TOB,#324        ;SEE IF CALLING FOR XMIT
1235  005130  001320                          BNE     TYPER           ;BR IF NOT
1236  005132  022702  005434                  CMP     #TEMBUF,R2      ;MAKE SURE THAT LAST LINE WAS TERMINATED
1237  005136  001315                          BNE     TYPER           ;BR IF NOT
1238  005140  004737  006210                  JSR     PC,TOG          ;INDICATE TRANSMIT
```

```
                                                ;THIS CODE TRANSMITS THE CONTENTS OF 'OCTBUF'
                                                ;OUT TO THE PHOTO-COMP MACHINE CONTINUOUSLY
                                                ;UNTIL A CONTROL CHARACTER IS TYPED.
                                                ;SWITCH 13 CONTROLS: LOOP OR SINGLE PASS

005144  005077  173712          XFER1:  CLR     @PCS            ;MAKE SURE INT ENABLE IS OFF
005150  012705  014000          XFER:   MOV     #OCTBUF,R5      ;SET UP STARTING ADRS OF DATA BUFFER
005154  013702  005420                  MOV     LINCNT,R2       ;SET UP TOTAL NUMBER OF OUTPUT BYTES
005160  005302                  NXFER:  DEC     R2              ;MAINTAIN XFER COUNT
005162  001006                          BNE     1$              ;BR IF NOT END OF BUFFER
005164  032777  020000  173664          BIT     #20000,@SWR     ;SEE IF LOOP SWITCH SET
005172  001366                          BNE     XFER            ;BR IF SET
005174  000000                          HALT                    ;COMPLETED BUFFER OUTPUT TO PHOTO-COMP
005176  000764                          BR      XFER            ;GO XFER BUFFER AGAIN
005200  005777  173656          1$:     TST     @PCS            ;MAKE SURE PHOTO-COMP ON LINE
005204  100007                          BPL     2$              ;BR IF SO
005206  012704  012372                  MOV     #MSG53,R4       ;SET UP ADRS OF MSG
005212  004737  006130                  JSR     PC,TTOUT        ;GO TYPE PHOTO-COMP OFF LINE
005216  000000                          HALT                    ;ERROR- PHOTO-COMP BECAME OFF LINE
005220  000137  005200                  JMP     1$              ;CHECK IT AGAIN
005224  105777  173632          2$:     TSTB    @PCS            ;LOOK FOR READY INDICATION
005230  100005                          BPL     3$              ;BR IF NOT READY
005232  112577  173626                  MOVB    (R5)+,@PCB      ;SEND OCTAL VALUE TO PHOTO-COMP
005236  005037  005424                  CLR     STOP            ;CLR BUM KEY INDICATION
005242  000746                          BR      NXFER           ;BR FOR NEXT VALUE
005244  105777  173574          3$:     TSTB    @TKS            ;SEE IF KEYBOARD HAS BEEN STRUCK
005250  100365                          BPL     2$              ;BR IF NOT
005252  017737  173570  005422  4$:     MOV     @TKB,TEMP       ;SAVE KEY TYPED

                                                ;THIS CODE CHECKS FOR TTY KEYS 'S','C','T'&'I'
                                                ;FOR I/O CONTROL DURING OUTPUT TO PHOTO-COMP

005260  023727  005422  000324          CMP     TEMP,#324       ;SEE IF AN 'T' HAS BEEN TYPED
005266  001003                          BNE     5$              ;BR IF NOT
005270  004737  005372                  JSR     PC,TYPCMD       ;GO TYPE COMMAND LETTER
005274  000725                          BR      XFER            ;XMIT BUFFER STARTING AT LINE #1
005276  023727  005422  000311  5$:     CMP     TEMP,#311       ;SEE IF AN 'I' HAS BEEN TYPED
005304  001004                          BNE     6$              ;BR IF NOT
005306  004737  005372                  JSR     PC,TYPCMD       ;GO TYPE COMMAND LETTER
005312  000137  004510                  JMP     SETBUF          ;GO GET NEW DATA
005316  023727  005422  000303  6$:     CMP     TEMP,#303       ;SEE IF AN 'C' HAS BEEN TYPED
005324  001003                          BNE     7$              ;BR IF NOT
005326  004737  005372                  JSR     PC,TYPCMD       ;GO TYPE COMMAND LETTER
005332  000734                          BR      2$              ;GO CONTINUE WHERE INTERRUPED
005334  023727  005422  000323  7$:     CMP     TEMP,#323       ;SEE IF AN 'S' HAS BEEN TYPED
005342  001403                          BEQ     8$              ;BR IF SO
005344  005737  005424                  TST     STOP            ;SEE IF IN THE STOP MODE
005350  001725                          BEQ     2$              ;RETURN TO WAIT LOOP IF BUM KEY
005352  004737  005372          8$:     JSR     PC,TYPCMD       ;GO TYPE COMMAND LETTER
005356  005237  005424                  INC     STOP            ;RECORD GOOD STOP KEY
005362  105777  173456          9$:     TSTB    @TKS            ;LOOK FOR COMMAND
005366  100375                          BPL     9$              ;WAIT FOR COMMAND
005370  000730                          BR      4$              ;GO SEE WHAT COMMAND
```

```
1292                                                    ;THIS ROUTINE TYPES THE COMMAND RECEIVED DURING TRANSMIT
1293
1294  005372  012704  012442          TYPCMD: MOV     #MSG54,R4       ;SET UP FOR CR/LF TYPE
1295  005376  004737  006130                  JSR     PC,TTOUT        ;GO SET MARGIN
1296  005402  013737  005422  001024          MOV     TEMP,TOB        ;SET UP FOR COMMAND ECHO
1297  005410  004737  006210                  JSR     PC,TOG          ;GO ECHO
1298  005414  000207                          RTS     PC              ;RETURN
1299
1300                                                    ;CORE LOCATIONS USED BY THE DATA OUTPUT TEST
1301
1302  005416  000000                  CCNT:   0               ;MAINTAINS COUNT OF INPUT DIGITS
1303  005420  000000                  LINCNT: 0               ;CONTAINS NUMBER OF LINES TYPED
1304  005422  000000                  TEMP:   0               ;INPUT OCTAL DATA ASSEMBLED HERE
1305  005424  000000                  STOP:   0               ;IGNORS BAD KEY IF ZERO
1306  005426  000000                  TEMBOS: 0               ;SPACE FOR LEADING ZEROS
1307  005430  000000                          0               ;
1308  005432  000000                          0               ;
1309  005434  000000                  TEMBUF: 0               ;START OF ASCII DIGIT BUFFER
1310  005436  000000                          0               ;
1311  005440  000000                          0               ;
```

```
1312                                    ;*************************************
1313                                    ;LOGIC TEST ERROR ROUTINE   .
1314                                    ;
1315                                    ;THIS ROUTINE IS USED TO DISPLAY THE
1316                                    ;TEST NUMBER AND ERROR CONDITION FOUND
1317                                    ;DURING PART 1.
1318                                    ;ERROR PRINTING IS CONDTIONED BY
1319                                    ;CONSOLE SWITCH 15, CONSOLE SWITCH 14
1320                                    ;CONDITIONS STOP OR CONTINUE ON ERROR.
1321                                    ;DUE TO TIMING AND SEQUENCE CONSIDERATIONS
1322                                    ;ANY LOGIC ERROR WILL RESULT IN A RESTART
1323                                    ;OF THE LOGIC TEST.
1324                                    ;*************************************
1325
1326  005442  032777  100000  173406  PLE:    BIT     #100000,@SWR    ;SEE IF NO PRINT
1327  005450  001012                          BNE     PCONTL          ;IF SO: BR
1328  005452  012704  006636                  MOV     #MSG0,R4        ;R4= TEST NUMBER MSG ADRS
1329  005456  013764  001004  000020          MOV     PTN,+20(R4)     ;SET TEST NUMBER
1330  005464  004737  006130                  JSR     PC,TTOUT        ;GO TYPE TEST NUMBER
1331  005470  010304                          MOV     R3,R4           ;R4= ERROR MSG ADRS
1332  005472  004737  006130                  JSR     PC,TTOUT        ;PRINT ERROR
1333  005476  032777  040000  173352  PCONTL: BIT     #40000,@SWR     ;SEE IF HALT ON ERROR
1334  005504  001401                          BEQ     PGOL            ;IF NOT: BR
1335  005506  000000                          HALT                    ;LOGIC TEST ERROR
1336  005510  032777  002000  173340  PGOL:   BIT     #2000,@SWR      ;SEE IF ERROR BIT TEST
1337  005516  001402                          BEQ     PGO             ;IF NOT: BR
1338  005520  000137  003434                  JMP     PEBT2           ;RETRY ERROR BIT TEST
1339  005524  000137  001200          PGO:    JMP     PSKIP           ;RESTART
1340
1341                                    ;*************************************
1342                                    ;DATA TEST ERROR ROUTINE
1343                                    ;
1344                                    ;THIS ROUTINE IS USED TO DISPLAY ANY
1345                                    ;PART TWO ERROR.
1346                                    ;ANY BAD DATA CHARACTER WILL BE PRINTED
1347                                    ;ALONG WITH THE EXPECTD DATA AS EIGHT
1348                                    ;BITS OF ONES AND ZEROS.
1349                                    ;ERROR PRINTING IS CONDITIONED BY CONSOLE
1350                                    ;SWITCH 15. CONSOLE SWITCH 14 CONDITIONS
1351                                    ;STOP OR CONTINUE ON ERROR. A DATA ERROR
1352                                    ;WILL RESULT IN CONTINUATION AT THE NEXT
1353                                    ;PATTERN OR ITERATION.
1354                                    ;*************************************
1355
1356  005530  032777  100000  173320  PDE:    BIT     #100000,@SWR    ;SEE IF NO PRINT
1357  005536  001053                          BNE     PCONTD          ;IF NO PRINT: BR
1358  005540  012704  010344                  MOV     #MSG27,R4       ;SET ERROR MSG ADDR
1359  005544  004737  006130                  JSR     PC,TTOUT
1360  005550  012704  010361                  MOV     #MSG30,R4
1361  005554  004737  006130                  JSR     PC,TTOUT        ;TYPE EXPECTED DATA HEADER
1362  005560  005003                          CLR     R3
1363  005562  013704  001022                  MOV     PCHAR,R4        ;R4 = EXPT CHAR
1364  005566  012705  000010          PDE1:   MOV     #10,R5          ;R5 = NUMBER OF BITS TO TYPE
1365  005572  105777  173252          PDE2:   TSTB    @TPS
1366  005576  100375                          BPL     PDE2            ;AWAIT TTY READY
1367  005600  132704  000200                  BITB    #200,R4         ;SEE IF ONE OR ZERO
```

```
1368  005604  001404                          BEQ     PDE3            ;IF ZERO: BR
1369  005606  012777  000061  173236          MOV     #061,@TPB       ;TYPE A ONE
1370  005614  000403                          BR      PDE4
1371  005616  012777  000060  173226  PDE3:   MOV     #060,@TPB       ;TYPE A ZERO
1372  005624  006104                  PDE4:   ROL     R4              ;GET NEXT BIT
1373  005626  005305                          DEC     R5              ;SEE IF DONE ALL
1374  005630  001360                          BNE     PDE2            ;IF NOT: BR
1375  005632  005703                          TST     R3              ;LOOK FOR RECIEVED DONE FLAG
1376  005634  001010                          BNE     PDE5            ;IF NOT: BR
1377  005636  012704  010402                  MOV     #MSG31,R4
1378  005642  004737  006130                  JSR     PC,TTOUT        ;TYPE RECIEVED DATA HEADER
1379  005646  005203                          INC     R3              ;SET FLAG
1380  005650  010004                          MOV     R0,R4           ;SET RECIEVED CHAR IN R4
1381  005652  000137  005566                  JMP     PDE1            ;DO RECIEVED
1382  005656  012704  010423          PDE5:   MOV     #MSG32,R4
1383  005662  004737  006130                  JSR     PC,TTOUT        ;TYPE CR/LF
1384  005666  032777  040000  173162  PCONTD: BIT     #40000,@SWR     ;SEE IF HALT ON ERROR
1385  005674  001401                          BEQ     PGOD            ;IF NO: BR
1386  005676  000000                          HALT
1387  005700  000207                  PGOD:   RTS     PC              ;RETURN
1388
1389
1390                                          ;*************************************
1391                                          ;PROGRAM END ROUTINE
1392                                          ;
1393                                          ;THIS ROUTINE IS USED TO PRINT OUT
1394                                          ;"END OF TEST" AND THEN HALT AT
1395                                          ;THE END OF A SINGLE PASS OR TO
1396                                          ;RESTART WITH NO PRINT OUT DEPENDING
1397                                          ;ON CONSOLE SWITCH 13.
1398                                          ;*************************************
1399
1400  005702  032777  020000  173146  PEND:   BIT     #20000,@SWR     ;LOOK FOR CONTUOUS MODE
1401  005710  001005                          BNE     PLOOP           ;IF SO: BR
1402  005712  012704  007347                  MOV     #MSG7,R4
1403  005716  004737  006130                  JSR     PC,TTOUT        ;PRINT END OF TEST
1404  005722  000000                          HALT
1405  005724  000137  001200          PLOOP:  JMP     PSKIP           ;LOOP
1406
1407
1408                                          ;*************************************
1409                                          ;DECIMAL PINT ROUTINE
1410                                          ;
1411                                          ;PRINT THE CONTENTS OF R3 IN DECIMAL
1412                                          ;*************************************
1413
1414  005730  012737  177773  006110  DECPRT: MOV     #-5,DIGCNT
1415  005736  012737  006116  006114          MOV     #DECPNT+2,DECPNT
1416  005744  005037  006112                  CLR     ZERO
1417  005750  012737  177777  006106  TYPT1:  MOV     #-1,DIGIT
1418  005756  005237  006106          TYPT2:  INC     DIGIT
1419  005762  167703  000126                  SUB     @DECPNT,R3
1420  005766  100373                          BPL     TYPT2
1421  005770  067703  000120                  ADD     @DECPNT,R3
1422  005774  004737  006020                  JSR     PC,DECOUT
1423  006000  005237  006110                  INC     DIGCNT
```

```
1424  006004  001001                          BNE     TYPT3
1425  006006  000207                          RTS     PC
1426  006010  062737  000002  006114  TYPT3:  ADD     #2,DECPNT
1427  006016  000754                          BR      TYPT1
1428  006020  005737  006106          DECOUT: TST     DIGIT
1429  006024  001010                          BNE     DEC1
1430  006026  022737  177777  006110          CMP     #-1,DIGCNT
1431  006034  001404                          BEQ     DEC1
1432  006036  013737  006112  006106          MOV     ZERO,DIGIT
1433  006044  000406                          BR      DEC2
1434  006046  012737  000260  006112  DEC1:   MOV     #260,ZERO
1435  006054  052737  000260  006106          BIS     #260,DIGIT
1436  006062  005737  006106          DEC2:   TST     DIGIT
1437  006066  001406                          BEQ     DEC3
1438  006070  105777  172754                  TSTB    @TPS
1439  006074  100372                          BPL     DEC2
1440  006076  013777  006106  172746          MOV     DIGIT,@TPB
1441  006104  000207                  DEC3:   RTS     PC
1442
1443  006106  000000                  DIGIT:  0
1444  006110  000000                  DIGCNT: 0
1445  006112  000000                  ZERO:   0
1446  006114  006116                  DECPNT: .+2
1447  006116  023420                          10000.
1448  006120  001750                          1000.
1449  006122  000144                          100.
1450  006124  000012                          10.
1451  006126  000001                          1.
1452
1453
1454                                          ;*************************************
1455                                          ;TTY OUTPUT ROUTINE
1456                                          ;
1457                                          ;ROUTINE TO OUTPUT ANY MESSAGE
1458                                          ;*************************************
1459
1460  006130  112437  001024          TTOUT:  MOVB    (R4)+,TOB       ;TOB = OUTPUT CHAR
1461  006134  122737  000043  001024          CMPB    #43,TOB         ;LOOK FOR TERMINATOR
1462  006142  001430                          BEQ     TEX             ;EXIT
1463  006144  122737  000045  001024          CMPB    #45,TOB         ;LOOK FOR CR/LF
1464  006152  001403                          BEQ     TCRLF           ;IF SO: BR
1465  006154  004737  006210                  JSR     PC,TOG          ;GO TYPE CHAR
1466  006160  000763                          BR      TTOUT           ;CONTINUE
1467  006162  112737  000015  001024  TCRLF:  MOVB    #15,TOB         ;TOB = CR
1468  006170  004737  006210                  JSR     PC,TOG          ;GO TYPE CR
1469  006174  112737  000012  001024          MOVB    #12,TOB         ;TOB = LF
1470  006202  004737  006210                  JSR     PC,TOG          ;GO TYPE LF
1471  006206  000750                          BR      TTOUT           ;CONTINUE
1472  006210  105777  172634          TOG:    TSTB    @TPS            ;LOOK FOR DONE
1473  006214  100375                          BPL     TOG             ;AWAIT FLAG
1474  006216  113777  001024  172626          MOVB    TOB,@TPB        ;ECHO CHAR
1475  006224  000207                  TEX:    RTS     PC              ;EXIT
1476
1477
1478                                          ;*************************************
1479                                          ;TTY INPUT ROUTINE
```

```
1480                                                    ;
1481                                                    ;THIS ROUTINE IS USED TO READ IN THE
1482                                                    ;VECTOR AND REGISTER SETTINGS.
1483                                                    ;************************************
1484
1485  006226  005014                        READ:   CLR     @R4             ;CLEAR FOR INPUT
1486  006230  004737  006276                READ1:  JSR     PC,TTIN         ;GO READ KEYBOARD AND ECHO
1487  006234  022737  000215  001026                CMP     #215,TIB        ;LOOK FOR CR
1488  006242  001414                               BEQ     REX             ;IF SO: BR
1489  006244  013701  001030                       MOV     ROTA,R1         ;SET ROTATION FACTOR
1490  006250  000241                        ROT:    CLC
1491  006252  006114                               ROL     @R4             ;ROTATE INPUT
1492  006254  005301                               DEC     R1              ;DECREMENT POSITION FACTOR
1493  006256  001374                               BNE     ROT             ;DO ALL ROTATIONS
1494  006260  040537  001026                       BIC     R5,TIB          ;STRIP ASCII
1495  006264  053714  001026                       BIS     TIB,@R4         ;SET NEW DIGIT
1496  006270  005303                               DEC     R3              ;DECREMENT CHAR CNTR
1497  006272  001356                               BNE     READ1           ;CONTINUE
1498  006274  000207                        REX:    RTS     PC              ;EXIT
1499
1500                                                    ;GET CHAR FROM KEYBOARD AND ECHO
1501
1502  006276  005077  172542                TTIN:   CLR     @TKS
1503  006302  005077  172540                        CLR     @TKB
1504  006306  005037  001026                        CLR     TIB
1505  006312  005277  172526                        INC     @TKS            ;SET READER GO BIT
1506  006316  105777  172522                TTIN1:  TSTB    @TKS            ;AWAIT DONE
1507  006322  100375                               BPL     TTIN1
1508  006324  017737  172516  001026                MOV     @TKB,TIB        ;TIB = INPUT CHAR
1509  006332  105777  172512                TTIN2:  TSTB    @TPS            ;AWAIT PUNCH READY
1510  006336  100375                               BPL     TTIN2
1511  006340  113777  001026  172504                MOVB    TIB,@TPB        ;SET ECHO
1512  006346  000207                               RTS     PC              ;EXIT
1513
1514
1515                                                    ;************************************
1516                                                    ;MAINTENANCE MODE INTERRUPT ROUTINE
1517                                                    ;
1518                                                    ;THIS ROUTINE IS USED TO RESET THE
1519                                                    ;STACK POINTER AND TO RESET VECTORS
1520                                                    ;TO HALTS THEN RETURN.
1521                                                    ;************************************
1522
1523  006350  032625                        PINTS:  BIT     (SP)+,(SP)+     ;RESET STACK POINTER
1524  006352  004737  006452                        JSR     PC,PRSTV        ;RESET VECTORS
1525  006356  000177  172450                        JMP     @PINTR          ;RETURN
```

```
1526                                    ;***********************************
1527                                    ;LOGIC TEST INTERRUPT ROUTINE
1528                                    ;
1529                                    ;THIS ROUTINE IS USED TO RESET THE
1530                                    ;THE STACK POINTER AND TO ASSURE THAT
1531                                    ;THE INTERRUPT DOES INDEED CLEAR.
1532                                    ;IF THE INTERRUPT SHOULD NOT CLEAR
1533                                    ;IT WILL BE FLAGGED AS AN ERROR.
1534                                    ;IF THE INTERRUPT IS CLEARED, THEN
1535                                    ;THE VECTORS WILL BE RESET AND A
1536                                    ;RETURN EXCUTED.
1537                                    ;***********************************
1538
1539  006362  022626          PINT:   CMP     (SP)+,(SP)+     ;RESET STACK POINTER
1540  006364  005077  172476         CLR     @PTS            ;RESET INTERRUPT ENABLE IN TESTER
1541  006370  005077  172466         CLR     @PCS            ;RESET CONT INTERRUPT ENABLE
1542  006374  024040                 CMP     -(R0),-(R0)     ;POINT TO PROPER VECTOR
1543  006376  012720  006442         MOV     #PINTE,(R0)+    ;SET TO ERROR VECTOR
1544  006402  012720  000340         MOV     #340,(R0)+
1545  006406  005077  172446         CLR     @PSW            ;SET PSW TO LOWEST PRIORITY
1546  006412  005037  001014         CLR     PTIM1           ;SET TIME
1547  006416  005337  001014 PINT1:  DEC     PTIM1
1548  006422  001375                 BNE     PINT1           ;AWAIT ANY ERROR INTERRUPTS
1549  006424  012777  000340  172426 MOV     #340,@PSW       ;RESET PSW
1550  006432  004737  006452         JSR     PC,PRSTV        ;RESET VECTORS
1551  006436  000177  172370         JMP     @PINTR          ;RETURN
1552  006442  012704  007365 PINTE:  MOV     #MSG10,R4
1553  006446  000137  005442         JMP     PLE             ;GO TO ERROR ROUTINE
1554
1555
1556                                    ;***********************************
1557                                    ;RESET VECTOR ROUTINE
1558                                    ;
1559                                    ;THIS ROUTINE IS USED TO RESET BOTH
1560                                    ;CONTROLLER AND TESTER VECTORS TO
1561                                    ;HALT TRAPS.
1562                                    ;***********************************
1563
1564  006452  013700  001000 PRSTV:  MOV     PVEC,R0         ;R0 = CONT VECTOR
1565  006456  010001                 MOV     R0,R1
1566  006460  005721                 TST     (R1)+           ;POINT TO CONT HALT LOC
1567  006462  010120                 MOV     R1,(R0)+        ;SET CONT VECTOR ADDR
1568  006464  005020                 CLR     (R0)+           ;SET CONT HALT
1569  006466  022121                 CMP     (R1)+,(R1)+     ;POINT TO TESTER HALT
1570  006470  010120                 MOV     R1,(R0)+        ;SET TESTER VECTOR ADDR
1571  006472  005020                 CLR     (R0)+           ;SET TESTER HALT
1572  006474  000207                 RTS     PC              ;RETURN
```

```
1573                                            ;***********************************
1574                                            ;POWER FAIL ROUTINE
1575                                            ;
1576                                            ;THIS ROUTINE IS USED TO SAVE ALL
1577                                            ;PROCESSOR REGISTERS AND POINT TO
1578                                            ;THE POWER UP ROUTINE BEFORE
1579                                            ;HALTING.
1580                                            ;***********************************
1581
1582  006476  010046                    PWRFAL: MOV     R0,-(SP)
1583  006500  010146                            MOV     R1,-(SP)
1584  006502  010246                            MOV     R2,-(SP)
1585  006504  010346                            MOV     R3,-(SP)
1586  006506  010446                            MOV     R4,-(SP)
1587  006510  010546                            MOV     R5,-(SP)
1588  006512  013746  000024                    MOV     24,-(SP)
1589  006516  010637  001006                    MOV     SP,PSPS         ;SAVE STACK POINTER
1590  006522  012737  006532  000024            MOV     #PWRUP,24
1591  006530  000000                            HALT
1592
1593                                            ;***********************************
1594                                            ;POWER UP ROUTINE
1595                                            ;
1596                                            ;THIS ROUTINE IS USED TO RESTORE
1597                                            ;ALL PROCESSOR REGISTERS AND TYPE
1598                                            ;OUT A POWER FAIL MESSAGE BEFORE
1599                                            ;RESTARTING.
1600                                            ;***********************************
1601
1602  006532  012777  000340  172320    PWRUP:  MOV     #340,@PSW
1603  006540  013706  001006                    MOV     PSPS,SP
1604  006544  012637  000024                    MOV     (SP)+,24
1605  006550  012605                            MOV     (SP)+,R5
1606  006552  012604                            MOV     (SP)+,R4
1607  006554  012603                            MOV     (SP)+,R3
1608  006556  012602                            MOV     (SP)+,R2
1609  006560  012601                            MOV     (SP)+,R1
1610  006562  012600                            MOV     (SP)+,R0
1611  006564  005005                            CLR     R5
1612  006566  005305                            DEC     R5
1613  006570  001376                            BNE     .-2
1614  006572  012704  010757                    MOV     #MSG40,R4       ;TYPE POWER FAIL MESSAGE
1615  006576  004737  006130                    JSR     PC,TTOUT
1616  006602  005305                            DEC     R5
1617  006604  001376                            BNE     .-2
1618  006606  000137  001206                    JMP     PBEG
1619
```

```
1620
1621                                        ;************************
1622                                        ;DATA PATTERNS
1623                                        ;************************
1624
1625                                        .EVEN
1626  006612   000  377  376  PPATS:  .BYTE   0,377,376,375,373,367,357,337,277,177
1627  006615   375  373  367
1628  006620   357  337  277
1629  006623   177
1630  006624   001  002  004          .BYTE   1,2,4,10,20,40,100,200
1631  006627   010  020  040
1632  006632   100  200
1633  006634   377               PPATE:  .BYTE   377
1634           006636                    .EVEN
1635
1636
1637                                        ;************************
1638                                        ;MESSAGE TABLE
1639                                        ;************************
1640
1641  006636  046045  043517  041511  MSG0:   .ASCII  /%LOGIC TEST NO. 00#/
1642  006644  052040  051505  020124
1643  006652  047516  020056  030060
1644  006660     043
1645
1646  006661     045  050114  030504  MSG1:   .ASCII  /%LPD11 INTERFACE TEST%#/
1647  006666  020061  047111  042524
1648  006674  043122  041501  020105
1649  006702  042524  052123  021445
1650
1651  006710  042445  052116  051105  MSG2:   .ASCII  /%ENTER STARTING VECTOR ADDRESS%#/
1652  006716  051440  040524  052122
1653  006724  047111  020107  042526
1654  006732  052103  051117  040440
1655  006740  042104  042522  051523
1656  006746  021445
1657
1658  006750  044445  046114  043505  MSG3:   .ASCII  /%ILLEGAL VECTOR: TRY AGAIN%#/
1659  006756  046101  053040  041505
1660  006764  047524  035122  052040
1661  006772  054522  040440  040507
1662  007000  047111  021445
1663
1664  007004  042445  052116  051105  MSG4:   .ASCII  /%ENTER STARTING REGISTER ADDRESS%#/
1665  007012  051440  040524  052122
1666  007020  047111  020107  042522
1667  007026  044507  052123  051105
1668  007034  040440  042104  042522
1669  007042  051523  021445
1670
1671  007046  044445  046114  043505  MSG5:   .ASCII  /%ILLEGAL ADDRESS: TRY AGAIN%#/
1672  007054  046101  040440  042104
1673  007062  042522  051523  020072
1674  007070  051124  020131  043501
1675  007076  044501  022516     043
```

```
007103      045 042523 020124 MSG6:   .ASCII  /%SET THE LPC01 TO 'ON LINE'%/
007110   044124 020105 050114
007116   030103 020061 047524
007124   023440 047117 046040
007132   047111 023505    045
007137      120 052125 051440         .ASCII  /PUT SWITCH 8 OF SWITCH GROUP 2 TO THE ON POSITION%@/
007144   044527 041524 020110
007152   020070 043117 051440
007160   044527 041524 020110
007166   051107 052517 020120
007174   020062 047524 052040
007202   042510 047440 020116
007210   047520 044523 044524
007216   047117 021445
007222   044124 020105 053523 MSG6A:  .ASCII  THE SWITCH IS LOCATED ON THE M523 MODULE AT THE LPD LOGIC%/
007230   052111 044103 044440
007236   020123 047514 040503
007244   042524 020104 047117
007252   052040 042510 046440
007260   031065 020063 047515
007266   052504 042514 040440
007274   020124 044124 020105
007302   050114 020104 047514
007310   044507 022503
007314   054524 042520 023440         .ASCII  /TYPE 'G' WHEN READY TO GO%@/
007322   023507 053440 042510
007330   020116 042522 042101
007336   020131 047524 043440
007344   022517    043

007347      045 047105 020104 MSG7:   .ASCII  /%END OF TEST%@/
007354   043117 052040 051505
007362   022524    043

007365      045 047111 042524 MSG10:  .ASCII  /%INTERRUPT HANG UP@%
007372   051122 050125 020124
007400   040510 043516 052440
007406   021520    045

007411      045 047111 042524 MSG11:  .ASCII  /%INTERFACE 'ID' CODE NOT READ CORRECTLY%@/
007416   043122 041501 020105
007424   044447 023504 041440
007432   042117 020105 047516
007440   020124 042522 042101
007446   041440 051117 042522
007454   052103 054514 021445

007462   041445 047101 047516 MSG12:  .ASCII  /%CANNOT CLEAR TESTER STATUS%@/
007470   020124 046103 040505
007476   020122 042524 052123
007504   051105 051440 040524
007512   052524 022523    043

007517      045 040503 047116 MSG13:  .ASCII  /%CANNOT CLEAR CONTROLLER STATUS%@
007524   052117 041440 042514
```

```
 1732  007532  051101  041440  047117
 1733  007540  051124  046117  042514
 1734  007546  020122  052123  052101
 1735  007554  051525  021445
 1736
 1737  007560  041445  047117  051124  MSG14:  .ASCII  %CONTROLLER READY NOT RESET BY CHAR LOAD%/
 1738  007566  046117  042514  020122
 1739  007574  042522  042101  020131
 1740  007602  047516  020124  042522
 1741  007610  042523  020124  054502
 1742  007616  041440  040510  020123
 1743  007624  047514  042101  021445
 1744
 1745  007632  052045  051505  042524  MSG15:  .ASCII  %TESTER READY NOT SET AFTER CHAR LOAD%/
 1746  007640  020122  042522  042101
 1747  007646  020131  047516  020124
 1748  007654  042523  020124  043101
 1749  007662  042524  020122  044103
 1750  007670  051101  046040  040517
 1751  007676  022504     043
 1752
 1753  007701     045  047507  041040  MSG16:  .ASCII  /%GO BIT WILL NOT RESET%/
 1754  007706  052111  053440  046111
 1755  007714  020114  047516  020124
 1756  007722  042522  042523  022524
 1757  007730     043
 1758
 1759  007731     045  042524  052123  MSG17:  .ASCII  /%TESTER READY RESETS FROM GO%/
 1760  007736  051105  051040  040505
 1761  007744  054504  051040  051505
 1762  007752  052105  020123  051106
 1763  007760  046517  043440  022517
 1764  007766     043
 1765
 1766  007767     045  047503  052116  MSG20:  .ASCII  /%CONTROLLER READY NOT SET FROM GO%/
 1767  007774  047522  046114  051105
 1768  010002  051040  040505  054504
 1769  010010  047040  052117  051440
 1770  010016  052105  043040  047522
 1771  010024  020115  047507  021445
 1772
 1773  010032  052045  051505  042524  MSG21:  .ASCII  /%TESTER READY NOT SET AT END OF CYCLE%/
 1774  010040  020122  042522  042101
 1775  010046  020131  047516  020124
 1776  010054  042523  020124  052101
 1777  010062  042440  042116  047440
 1778  010070  020106  054503  046103
 1779  010076  022505     043
 1780
 1781  010101     045  047503  052116  MSG22:  .ASCII  /%CONTROLLER READY SET AT END OF CYCLE%/
 1782  010106  047522  046114  051105
 1783  010114  051040  040505  054504
 1784  010122  051440  052105  040440
 1785  010130  020124  047105  020104
 1786  010136  043117  041440  041531
 1787  010144  042514  021445
```

```
1788
1789  010150  047045  020117  047111  MSG24:  .ASCII  /?NO INTERRUPT FROM TESTER READY%#/
1790  010156  042524  051122  050125
1791  010164  020124  051106  046517
1792  010172  052040  051505  042524
1793  010200  020122  042522  042101
1794  010206  022531     043
1795
1796  010211     045  047516  044440  MSG25:  .ASCII  /%NO INTERRUPT FROM CONTROLLER READY%#/
1797  010216  052116  051105  052522
1798  010224  052120  043040  047522
1799  010232  020115  047503  052116
1800  010240  047522  046114  051105
1801  010246  051040  040505  054504
1802  010254  021445
1803
1804  010256  042445  051122  051117  MSG26:  .ASCII  /%ERROR BIT WILL NOT GO TO PROPER STATE%/
1805  010264  041040  052111  053440
1806  010272  046111  020114  047516
1807  010300  020124  047507  052040
1808  010306  020117  051120  050117
1809  010314  051105  051440  040524
1810  010322  042524     045
1811  010325     106  051117  052040          .ASCII  /FOR THIS MODE%#.
1812  010332  044510  020123  047515
1813  010340  042504  021445
1814
1815  010344  042045  052101  020101  MSG27:  .ASCII  /%DATA ERROR%#.
1816  010352  051105  047522  022522
1817  010360     043
1818
1819  010361     045  054105  042520  MSG30:  .ASCII   %EXPECTED DATA: #/
1820  010366  052103  042105  042040
1821  010374  052101  035101  021440
1822
1823  010402  051045  041505  044505  MSG31:  .ASCII  /%RECEIVED DATA: #/
1824  010410  042526  020104  040504
1825  010416  040524  020072     043
1826
1827  010423     045  021445          MSG32:  .ASCII  /%%#/
1828
1829
1830  010426  047045  020117  042522  MSG33:  .ASCII  /%NO READY FLAG DURING DATA TEST%#/
1831  010434  042101  020131  046106
1832  010442  043501  042040  051125
1833  010450  047111  020107  040504
1834  010456  040524  052040  051505
1835  010464  022524     043
1836
1837  010467     045  042524  052123  MSG34:  .ASCII  /%TESTER INTERRUPTS WITH NO INTERRUPT ENABLE%#/
1838  010474  051105  044440  052116
1839  010502  051105  052522  052120
1840  010510  020123  044527  044124
1841  010516  047040  020117  047111
1842  010524  042524  051122  050125
1843  010532  020124  047105  041101
```

```
 1944  010540  042514  021445
 1945
 1946  010544  052045  051505  042524  MSG35:  .ASCII  %TESTER INTERRUPTS AT HIGHER THAN LEVEL 4%@/
 1947  010552  020123  047111  042524
 1948  010560  051132  050125  051524
 1949  010566  040440  020124  044510
 1950  010574  044107  051105  052040
 1951  010602  040510  020116  042514
 1952  010610  042526  020114  022464
 1953  010616     043
 1954
 1955  010617     045  047503  052116  MSG36:  .ASCII  /%CONTROLLER INTERRUPTS WITH NO INTERRUPT ENABLE%@/
 1956  010624  047522  046114  051105
 1957  010632  044440  052116  051105
 1958  010640  052522  052120  020123
 1959  010646  044527  044124  047040
 1960  010654  020117  047111  042524
 1961  010662  051122  050125  020124
 1962  010670  047105  041101  042514
 1963  010676  021445
 1964
 1965  010700  041445  047117  051124  MSG37:  .ASCII  /%CONTROLLER INTERRUPTS AT HIGHER THAN LEVEL 4%@/
 1966  010706  046117  042514  020122
 1967  010714  047111  042524  051122
 1968  010722  050125  051524  040440
 1969  010730  020124  044510  044107
 1970  010736  051105  052040  040510
 1971  010744  020116  042514  042526
 1972  010752  020114  022464     043
 1973
 1974  010757     045  042522  052123  MSG40:  .ASCII  /%RESTART FROM POWER FAILURE%@/
 1975  010764  051101  020124  051106
 1976  010772  046517  050040  053517
 1977  011000  051105  043040  044501
 1978  011006  052514  042522  021445
 1979
 1980  011014  041445  047117  020124  MSG41:  .ASCII  /%CONT READY NOT RESET BY INIT%@/
 1981  011022  042522  042101  020131
 1982  011030  047516  020124  042522
 1983  011036  042523  020124  054502
 1984  011044  044440  044516  022524
 1985  011052     043
 1986
 1987  011053     045  042524  052123  MSG42:  .ASCII  /%TESTER READY NOT RESET BY INIT%@/
 1988  011060  051105  051040  040505
 1989  011066  054504  047040  052117
 1990  011074  051040  051505  052105
 1991  011102  041040  020131  047111
 1992  011110  052111  021445
 1993
 1994  011114  046045  047111  051505  MSG43:  .ASCII  /%LINES PER SECOND: %@/
 1995  011122  050040  051105  051440
 1996  011130  041505  047117  035104
 1997  011136  022440     043
 1998
 1999  011141     045  051105  047522  MSG44:  .ASCII  /%ERROR STATUS BIT TEST%/
```

```
 1900  011146  020122  052123  052101
 1901  011154  051525  041040  052111
 1902  011162  052040  051505  022524
 1903  011170  051445  052105  052040          .ASCII  /'SET THE LPC01 TO 'ON LINE'%/
 1904  011176  042510  046040  041520
 1905  011204  030460  052040  020117
 1906  011212  047447  020116  044514
 1907  011220  042516  022447
 1908  011224  052520  020124  053523          .ASCII  /PUT SWITCH 9 OF SWITCH GROUP 2 TO THE OFF POSITION%8/
 1909  011232  052111  044103  034040
 1910  011240  047440  020106  053523
 1911  011246  052111  044103  043440
 1912  011254  047522  050125  031040
 1913  011262  052040  020117  044124
 1914  011270  020105  043117  020106
 1915  011276  047520  044523  044524
 1916  011304  047117  021445
 1917
 1918  011310  051445  052105  052040  MSG45:  .ASCII  /%SET THE LPC01 TO 'OFF LINE'%/
 1919  011316  042510  046040  041520
 1920  011324  030460  052040  020117
 1921  011332  047447  043106  046040
 1922  011340  047111  023505     045
 1923  011345     120  052125  051440          .ASCII  /PUT SWITCH 8 OF SWITCH GROUP 2 TO THE OFF POSITION%/
 1924  011352  044527  041524  020110
 1925  011360  020070  043117  051440
 1926  011366  044527  041524  020110
 1927  011374  051107  052517  020120
 1928  011402  020062  047524  052040
 1929  011410  042510  047440  043106
 1930  011416  050040  051517  052111
 1931  011424  047511  022516
 1932  011430  054524  042520  023440          .ASCII  /TYPE 'G' WHEN READY TO GO %8/
 1933  011436  023507  053440  042510
 1934  011444  020116  042522  042101
 1935  011452  020131  047524  043440
 1936  011460  020117  021445
 1937  011464  051445  052105  052040  MSG46:  .ASCII  /%SET THE LPC01 TO 'ON LINE'%/
 1938  011472  042510  046040  041520
 1939  011500  030460  052040  020117
 1940  011506  047447  020116  044514
 1941  011514  042516  022447
 1942  011520  052520  020124  053523          .ASCII  /PUT SWITCH 8 OF SWITCH GROUP 2 TO THE OFF POSITION%/
 1943  011526  052111  044103  034040
 1944  011534  047440  020106  053523
 1945  011542  052111  044103  043440
 1946  011550  047522  050125  031040
 1947  011556  052040  020117  044124
 1948  011564  020105  043117  020106
 1949  011572  047520  044523  044524
 1950  011600  047117     045
 1951  011603     123  044527  041524          .ASCII  /SWITCH POWER OFF AT THE LPC01%/
 1952  011610  020110  047520  042527
 1953  011616  020122  043117  020106
 1954  011624  052101  052040  042510
 1955  011632  046040  041520  030460
```

```
 1956  011640     045
 1957  011641     124  050131  020105          .ASCII  /TYPE 'G' WHEN READY TO GO%@/
 1958  011646  043447  020047  044127
 1959  011654  047105  051040  040505
 1960  011662  054504  052040  020117
 1961  011670  047507  021445
 1962
 1963  011674  053523  052111  044103  MSG46A: .ASCII  /SWITCH POWER ON AT THE LPC01%/
 1964  011702  050040  053517  051105
 1965  011710  047440  020116  052101
 1966  011716  052040  042510  046040
 1967  011724  041520  030460     045
 1968  011731     124  050131  020105          .ASCII  /TYPE 'G' WHEN READY TO GO%@/
 1969  011736  043447  020047  044127
 1970  011744  047105  051040  040505
 1971  011752  054504  052040  020117
 1972  011760  047507  021445
 1973
 1974
 1975  011764  042445  051122  051117  MSG47:  .ASCII  /%ERROR BIT OK%@/
 1976  011772  041040  052111  047440
 1977  012000  022513     043
 1978
 1979  012003     045  040504  040524  MSG50:  .ASCII  /%DATA OUTPUT TEST%/
 1980  012010  047440  052125  052520
 1981  012016  020124  042524  052123
 1982  012024     045
 1983
 1984  012025     045  040515  042513          .ASCII  /%MAKE SURE THE PHOTO-COMP DEVICE IS 'ON LINE'%/
 1985  012032  051440  051125  020105
 1986  012040  044124  020105  044120
 1987  012046  052117  026517  047503
 1988  012054  050115  042040  053105
 1989  012062  041511  020105  051511
 1990  012070  023440  047117  046040
 1991  012076  047111  023505     045
 1992  012103     120  052125  051440          .ASCII  /PUT SWITCH 8 OF SWITCH GROUP 2 TO THE OFF POSITION%/
 1993  012110  044527  041524  020110
 1994  012116  020070  043117  051440
 1995  012124  044527  041524  020110
 1996  012132  051107  052517  020120
 1997  012140  020062  047524  052040
 1998  012146  042510  047440  043106
 1999  012154  050040  051517  052111
 2000  012162  047511  022516
 2001  012166  044124  020105  053523          .ASCII  /THE SWITCH IS LOCATED ON THE M523 MODULE IN THE LPD LOGIC%/
 2002  012174  052111  044103  044440
 2003  012202  020123  047514  040503
 2004  012210  042524  020104  047117
 2005  012216  052040  042510  046440
 2006  012224  031065  020063  047515
 2007  012232  052504  042514  044440
 2008  012240  020116  044124  020105
 2009  012246  050114  020104  047514
 2010  012254  044507  022503
 2011  012260  044127  047105  051040          .ASCII  /WHEN READY, ENTER OCTAL VALUES: (0-377)%@/
```

```
2012  012266  040505  054504  020054
2013  012274  047105  042524  020122
2014  012302  041517  040524  020114
2015  012310  040526  052514  051505
2016  012316  020072  030050  031455
2017  012324  033467  022451     043
2018
2019  012331     077  021477          MSG51:  .ASCII  /??*/
2020
2021  012334  047445  052103  046101  MSG52:  .ASCII  /%OCTAL OUTPUT BUFFER IS FULL%*/
2022  012342  047440  052125  052520
2023  012350  020124  052502  043106
2024  012356  051105  044440  020123
2025  012364  052506  046114  021445
2026
2027  012372  050045  047510  047524  MSG53:  .ASCII  /%PHOTO-COMP OR INTERFACE NOT 'ON LINE'%*/
2028  012400  041455  046517  020120
2029  012406  051117  044440  052116
2030  012414  051105  040506  042503
2031  012422  047040  052117  023440
2032  012430  047117  046040  047111
2033  012436  023505  021445
2034
2035  012442  021445          MSG54:  .ASCII  /%*/
2036
2037  012444  042445  051122  051117  MSG55:  .ASCII  /%ERROR BIT NOT SET WITH LPD SWITCHED TO LOCAL TEST%*/
2038  012452  041040  052111  047040
2039  012460  052117  051440  052105
2040  012466  053440  052111  020110
2041  012474  050114  020104  053523
2042  012502  052111  044103  042105
2043  012510  052040  020117  047514
2044  012516  040503  020114  042524
2045  012524  052123  021445
2046
2047                          ;BUFFER THAT CONTAINS OCTAL OUTPUT CHARACTERS
2048                          ;FOR THE DATA OUTPUT TEST
2049
2050          014000          .=14000
2051
2052  014000  000000  OCTBUF: 0              ;STARTING ADRS OF OCTAL BUFFER
2053                                          ;EXPANDS TO 28k
2054
2055          000001          .END
```

```
BUFOK   005074          1221    1224#
CCNT    005416          1176*   1192*   1302#
CNTRCK  005004          1188    1206#
DECOUT  006020          1422    1428#
DECPNT  006114          1415*   1419    1421    1426*   1446#
DECPRT  005730          1118    1178    1414#
DEC1    006046          1429    1431    1434#
DEC2    006062          1433    1436#   1439
DEC3    006104          1437    1441#
DIGCNT  006110          1414*   1423*   1430    1444#
DIGIT   006106          1417*   1418*   1428    1432*   1435*   1436    1440    1443#
DINT    004400          1082    1103#
ENDCK   005102          1190    1230#
ID      001036           494#    918
LINCNT  005420          1144*   1165*   1177    1224*   1246    1303#
LINE    004474          1114    1124#
LKS     001054           505#   1085*   1086    1088*   1110*
MSG0    006636          1328    1641#
MSG1    006661           547    1646#
MSG10   007365          1552    1711#
MSG11   007411           920    1716#
MSG12   007462           658    1724#
MSG13   007517           650    1730#
MSG14   007560           682    1737#
MSG15   007632           693    1745#
MSG16   007701           705    1753#
MSG17   007731           712    1759#
MSG2    006710           549    1651#
MSG20   007767           723    1766#
MSG21   010032           735    1773#
MSG22   010101           745    1781#
MSG24   010150           766    1789#
MSG25   010211           791    1796#
MSG26   010256           909    1804#
MSG27   010344          1358    1815#
MSG3    006750           563    1658#
MSG30   010361          1360    1819#
MSG31   010402          1377    1823#
MSG32   010423          1382    1827#
MSG33   010426           979    1830#
MSG34   010467           816    1837#
MSG35   010544           835    1846#
MSG36   010617           856    1855#
MSG37   010700           872    1865#
MSG4    007004           568    1664#
MSG40   010757          1614    1874#
MSG41   011014           665    1880#
MSG42   011053           673    1887#
MSG43   011114          1116    1894#
MSG44   011141           883    1899#
MSG45   011310           888    1918#
MSG46   011464           890    1937#
MSG46A  011674           894    1963#
MSG47   011764           896    1975#
MSG5    007046           581    1671#
MSG50   012003          1141    1979#
```

```
MSG51   012331          1199    2019*
MSG52   012334          1168    1222    2021*
MSG53   012372          1255    2027*
MSG54   012442          1173    1294    2035*
MSG55   012444           639    2037*
MSG6    007103           593     892    1677*
MSG6A   007222           595     885    1691*
MSG7    007347          1402    1707*
NXFER   005160          1247*   1263
NXLIN   004644          1146    1173*   1201    1225
OCTBUF  014000          1143    1245    2052*
PBEG    001206           467     529*   1618
PC      =%000007         430*    548*    553*    558*    564*    569*    574*    582*    594*    596*    601*    647*    779*
                         878*    884*    886*    889*    891*    893*    895*    897*    899*    904*    911*    963*    980*
                         985*   1023*   1117*   1118*   1142*   1156*   1160*   1169*   1174*   1178*   1180*   1191*   1200*
                        1223*   1232*   1238*   1256*   1273*   1277*   1281*   1287*   1295*   1297*   1298*   1330*   1332*
                        1359*   1361*   1378*   1383*   1387*   1403*   1422*   1425*   1441*   1465*   1468*   1470*   1475*
                        1486*   1498*   1512*   1524*   1550*   1572*   1615*
PCB     001064           509*    679*    729*    761*    809*    830*    953*    973*   1039*   1073*   1104*   1261*
PCHAR   001022           488*    949*    952*    953     961     972*    973     983     986*    987     1363
PCLR    001244           539*    541
PCONTD  005666          1357    1384*
PCONTL  005476          1327    1333*
PCS     001062           508*    585     637     644*    648     663     680     721     743     751*    782*    796*    824*
                         843*    864*    868*    905     916     946*   1022*   1029*   1074*   1112*   1244*   1253    1259
                        1541*
PDAT    003520           919     944*
PDATO   003524           945*    965     968
PDAT1   003564           953*    967
PDAT10  003752           984     986*
PDAT11  004004           990     993*
PDAT2   003576           955*    958
PDAT2A  003620           956     960*
PDAT3   003636           962     964*
PDAT4   003656           951     971*
PDAT5   003664           972*    992
PDAT6   003670           973*    988
PDAT7   003702           975*    978
PDAT7A  003720           959     979*
PDAT7B  003734           976     982*
PDE     005530           963     985    1356*
PDE1    005566          1364*   1381
PDE2    005572          1365*   1366    1374
PDE3    005616          1368    1371*
PDE4    005624          1370    1372*
PDE5    005656          1376    1382*
PEBT    003246           866     877*
PEBTO   003274           880     882*
PEBT1   003402           886     889     891     895     899*
PEBT2   003434           905*   1338
PEBT3   003462           908     911*
PEND    005702           981     993    1400*
PET     001034           493*    537     882*    887*    907
PGO     005524          1337    1339*
PGOD    005700          1385    1387*
PGOF    001012           484*    597     900
```

```
PGOL    005510          1334    1336#
PIC     001020           487#    945*    966*    971*    991*
PIDBT   003464           881     898     915#
PINT    006362           754     780    1539#
PINTE   006442          1543    1552#
PINTR   001032           492#    759*    784*    806*    833*    851*   1036*   1525    1551
PINTS   006350           804     849    1027    1034    1523#
PINTI   006416          1547#   1548
PITA    001010           483#    799*    815     848*
PLC     004236           613    1072#
PLCC    001600           606     611#   1122
PLCG    004470          1120    1122#
PLCP    004414          1098    1109#
PLE     005442           640     651     659     666     674     683     694     706     713     724     736     746     767
                         792     817     836     857     873     910     921    1326#   1553
PLOOP   005724          1401    1405#
PPATE   006634           950    1633#
PPATS   006612           944    1626#
PRB     001042           500#   1164
PREG    001002           480#    570     575     577     579     584
PREGE   001444           576     578     581#
PRG     001356           562     568#
PRG1    001366           570#    583
PRS     001040           499#   1150*   1162
PRSTV   006452           647     778     878    1023    1524    1550    1564#
PSCOP   004010           607    1016#   1019
PSCOP0  004016          1018#   1048    1051    1054
PSCOP1  004102          1025    1031#
PSCOP2  004126          1030    1036#
PSCOP3  004162          1042#   1043    1045
PSCOP4  004176          1036    1046#
PSCOP5  004214          1050#   1053
PSD     004500           619    1141#
PSDC    001614           612     617#
PSET    001456           580     584#
PSKIP   001200           470     527#   1339    1405
PSMC    001564           543     605#
PSPS    001006           482#   1589*   1603
PSTART  001210           528     530#
PSW     001060           507#    531*    760*    785*    807*    814*    832*    852*   1020*   1041*   1091*   1545*   1549*
                        1602#
PTB     001070           511#    822*    841*    862*    948*    960     982
PTIM1   001014           485#    688*    689*    700*    701*    718*    719*    730*    731*    741*    757*    762*    773#
                         774*    776*    787*    808*    810*    812*    828*    853*    954*    957*    974*    977*   1038#
                        1042*   1049*   1052*   1546*   1547*
PTIM2   001016           486#    758*    764*    783*    789*   1037*   1044*
PTN     001004           481#    636*    646*    655*    662*    670*    678*    687*    698*    709*    717*    728*    740#
                         749*    771*    795*    821*    840*    861*    877*    915*   1329
PTS     001066           510#    645*    656     671     691     699*    703     710     733     750*    756*    772*    786*
                         797*    823*    831*    842*    854*    863*    947*    955     975    1021*   1031*   1040*   1050
                        1072*   1089*   1090*   1105*   1111*   1540*
PTO     001630           618     636#
PT1     001654           638     644#
PT1A    002134           701#    702
PT1A1   001716           649     655#
PT1A1A  001744           657     662#
```

```
PT1A1B  001772            664     670*
PT1A2   002020            672     678*
PT1A3   002054            681     687*
PT1A3A  002066            689*    690
PT1A4   002114            692     698*
PT1B    002270            731*    732
PT1B1   002162            704     709*
PT1B2   002210            711     717*
PT1B2A  002222            719*    720
PT1B3   002250            722     728*
PT1C    002316            734     740*    742
PT1E    002352            744     749*
PT1F    002446            762*    763     765
PT1FA   002472            759     771*
PT1FB   002510            774*    775
PT1FC   002516            776*    777
PT1G    002576            787*    788     790
PT1H    002622            784     795*
PT1HA   002664            804*
PT1HB   002674            806*
PT1J    002716            810*    811     834     855
PT1K    002724            812*    813
PT1L    002744            800     816*
PT1M    002754            799     821*
PT1M1   003004            828*    829
PT1N    003044            833     835*
PT1P    003054            826     840*
PT1Q    003130            851*    871
PT1R    003154            847     856*
PT1S    003164            848     861*
PT1T    003236            870     872*
PVE     001344            560     563*
PVEC    001000            479*    554     559     561     752     779     802     845     1026    1032    1080    1564
PV1     001276            554*    565
PWRFAL  006476            443     1582*
PWRUP   006532           1590     1602*
RDRLK   004536           1151*    1163
RDRLK1  004532           1150*    1167    1170
READ    006226            558     574     601     904     1485*
READ1   006230           1486*    1497
REQ     001516            593*
RE).    006274           1488     1498*
RQT     006250           1490*    1493
RQTA    001030            491*    556*    572*    599*    902*    1489
R0      =%000000          422*    533*    534*    537*    538*    539*    584*    590*    752*    753     754*    755*    779*
                          780*    781*    802*    803     804*    805*    845*    849*    850*    905*    906*    907     960*
                          961     982*    983     1026*   1027*   1028*   1032*   1033    1034*   1035*   1080*   1081*   1082*
                         1083*   1084*   1097*   1380    1542    1543*   1544*   1564*   1565    1567*   1568*   1570*   1571*
                         1582    1610*
R1      =%000001          423*    536*    540*    585*    590*    944*    950     952     1489*   1492*   1565*   1566    1567
                         1569    1570    1583    1609*
R2      =%000002          424*    527*    529*    542     1016*   1017    1018    1113*   1114*   1115    1175*   1195*   1208
                         1209*   1210*   1211*   1212    1213    1215*   1216*   1217*   1218    1236    1246*   1247*   1584
                         1608*
R3      =%000003          425*    555*    571*    598*    639*    650*    658*    665*    673*    682*    693*    705*    712*
                          723*    735*    745*    766*    791*    801*    807     816*    935*    946*    852     956*    969*
```

```
                        872*  901*  909*  916*  917*  918   920*  1017*  1039  1046*  1075*  1103*  1104
                       1115*  1177* 1331   1362*  1375  1379*  1419*  1421*  1496*  1585   1607*
R4      =%000004        426*   547*  549*   554*  563*  568*   570*   581*   593*   595*   597*   800*   806
                        847*   851   870*   883*  885*  888*   890*   892*   894*   896*   900*   979*  1116*
                       1141*  1168*  1173*  1199*  1222*  1255*  1294*  1328*  1329*  1331*  1358*  1360*  1363*
                       1367   1372*  1377*  1380*  1382*  1402*  1460   1485*  1491*  1495*  1552*  1586   1606*
                       1614*
R5      =%000005        427*   557*  573*   600*  798*  825    827*   844*   865    867*   903*   1143*  1164*
                       1166   1219*  1220   1245*  1261  1364*  1373*  1494   1587   1605*  1611*  1612*  1616*
R6      =%000006        429*   1109
SETBUF  004510         1143*  1157   1233   1278
SP      =%000006        428*   429   530*   1523  1539  1582*  1583*  1584*  1585*  1586*  1587*  1588*  1589
                       1603*  1604   1605   1606  1607  1608   1609   1610
STOP    005424         1262*  1285   1288*  1305*
SWR     001056          506*   605   611    617   879   964    989    1016   1018   1024   1047   1077   1119
                       1145   1249   1326   1333  1336  1356   1384   1400
TCRLF   006162         1464   1467*
TEMBUF  005434         1175   1236   1309*
TEMBOS  005426         1306*
TEMP    005422         1153*  1154   1158   1208*  1212*  1218*  1219   1266*  1271   1275   1279   1293   1296
                       1304*
TEX     006224         1462   1475*
TIB     001026          490*  1487   1494*  1495  1504*  1508*  1511
TIME    004476         1076*  1079*  1084   1125*
TINT    004372          449   1097*
TKB     001046          502*  1153   1186   1266  1503*  1508
TKS     001044          501*  1151   1184   1264  1289   1502*  1505*  1506
TOB     001024          489*  1179*  1186*  1187  1189   1194*  1195   1206   1230   1234   1296*  1460*  1461
                       1463   1467*  1469*  1474
TOG     006210         1180   1191   1232   1238  1297   1465   1468   1470   1472*  1473
TPB     001052          504*  1369*  1371*  1440*  1474*  1511*
TPS     001050          503*  1365   1438   1472  1509
TTI     004710         1184*  1185   1196
TTIN    006276         1486   1502*
TTIN1   006316         1506*  1507
TTIN2   006332         1509*  1510
TTOUT   006130          548    553    564    569   582   594    596    884    893    897    899    980   1117
                       1142   1169   1174   1200  1223  1256   1295   1330   1332   1359   1361   1378  1383
                       1403   1460*  1466   1471  1615
TYPCMD  005372         1156   1160   1273   1277  1281  1287   1294*
TYPER   004772         1193   1199*  1207   1214  1235  1237
TYPT1   005750         1417*  1427
TYPT2   005756         1418*  1420
TYPT3   006010         1424   1426*
XFER    005150         1245*  1250   1252   1274
XFER1   005144         1161   1244*
ZERO    006112         1416*  1432   1434*  1445*
.       = 014002        434*   439    442*   448*  466*   469*   475*   512*   535    1446   1613   1617  1634*
                       2050*
```

```
ADD      590  1081  1212  1218  1421  1426
ASL     1209  1210  1211
ASR     1216  1217
BEQ      606   612   618   649   657   664   672   681   704   744   908   919   951   962   984
         988  1049  1078  1087  1098  1146  1193  1284  1286  1334  1337  1368  1385  1431  1437
        1462  1464  1488
BHI      562   578   580
BIC      906   917  1194  1494
BIS      756   782   831   868  1029  1031  1089  1435  1495
BIT      559   575   605   611   617   648   656   663   671   680   691   703   710   721   733
         743   879   955   964   975   987   989  1024  1047  1050  1077  1086  1119  1145  1249
        1326  1333  1336  1356  1384  1400  1523
BITB    1367
BLE     1214
BMI      638  1188
BNE      535   541   543   560   576   690   692   702   711   720   722   732   734   742   763
         765   775   777   788   790   811   813   826   829   866   880   956   958   965   967
         976   978   990   992  1019  1025  1043  1045  1051  1053  1120  1155  1159  1167  1207
        1221  1231  1235  1237  1248  1250  1272  1276  1280  1327  1357  1374  1376  1401  1424
        1429  1493  1497  1548  1613  1617
BPL     1152  1163  1185  1190  1254  1260  1265  1290  1366  1420  1439  1473  1507  1510
BR       528   565   583   968  1030  1157  1161  1170  1196  1201  1225  1252  1263  1274  1282
        1291  1370  1427  1433  1466  1471
CLC     1490
CLR      529   533   538   539   644   645   688   718   730   750   751   757   760   772   773
         785   796   797   798   801   808   822   823   824   830   841   842   843   844   846
         853   862   863   864   882   946   947   948   949   954   972   974  1021  1022  1038
        1041  1049  1072  1073  1074  1075  1085  1091  1110  1111  1112  1113  1244  1262  1362
        1416  1485  1502  1503  1504  1540  1541  1545  1546  1568  1571  1611
CMP      561   577   579   753   803   907   918   950  1033  1109  1154  1158  1166  1187  1189
        1206  1213  1220  1230  1234  1236  1271  1275  1279  1283  1430  1487  1539  1542  1569
CMPB     961   983  1018  1461  1463
COM     1046
DEC      534   540   689   701   719   731   741   762   764   774   776   787   789   810   812
         828   957   966   977   991  1042  1044  1052  1097  1192  1247  1373  1492  1496  1547
        1612  1616
DIV     1114
HALT     439  1121  1251  1257  1335  1386  1404  1591
INC      699   786   827   854   867   986  1040  1090  1103  1105  1150  1165  1224  1288  1379
        1418  1423  1505
JMP      467   470   607   613   619   640   651   659   666   674   683   694   706   713   724
         736   746   767   792   815   817   834   836   855   857   871   873   881   898   910
         921   959   981   993  1054  1093  1122  1233  1258  1278  1338  1339  1391  1405  1525
        1551  1553  1618
JSR      548   553   558   564   569   574   582   594   596   601   647   778   878   884   886
         889   891   893   895   897   899   904   963   980   985  1023  1117  1118  1142  1156
        1160  1169  1174  1178  1180  1191  1200  1223  1232  1238  1256  1273  1277  1281  1287
        1295  1297  1330  1332  1359  1361  1378  1383  1403  1422  1465  1468  1470  1486  1524
        1550  1615
MOV      527   530   531   536   537   547   549   554   555   556   557   563   568   570   571
         572   573   581   584   585   590   593   595   597   598   599   600   636   639   646
         650   655   658   662   665   670   673   678   679   682   687   693   698   700   705
         709   712   717   723   728   729   735   740   745   749   752   754   755   758   759
         761   766   771   779   780   781   783   784   791   795   799   800   802   804   805
         806   807   809   814   816   821   832   833   835   840   845   847   848   849   850
         851   852   856   861   869   870   872   877   883   885   887   888   890   892   894
```

BOS

```
DZ11 INTERFACE TEST - MAINDEC-11-DZLPJ-B-D      MACY11 27(732)  08-SEP-76  09:50  PAGE 55
DZLPJB.P11        CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

             896     900     901     902     903     905     909     915     916     920     944     945     971     973     979
            1016    1017    1020    1026    1027    1028    1032    1034    1035    1036    1037    1076    1079    1080    1082
            1083    1084    1089    1104    1115    1116    1141    1143    1144    1153    1168    1173    1175    1176    1177
            1179    1186    1195    1199    1208    1222    1245    1246    1255    1266    1294    1296    1328    1329    1331
            1358    1360    1363    1364    1369    1371    1377    1380    1382    1402    1414    1415    1417    1432    1434
            1440    1489    1508    1543    1544    1549    1552    1564    1565    1567    1570    1592    1583    1584    1595
            1596    1597    1588    1559    1590    1602    1603    1604    1605    1606    1607    1608    1609    1610    1614
MOVB         952     953     960     992    1039    1164    1219    1261    1460    1467    1469    1474    1511
RESET        532
ROL         1372    1491
RTI         1099    1106
RTS          911    1298    1387    1425    1441    1475    1498    1512    1572
SUB         1419
SWAB        1215
TST          542     637     925     965    1253    1285    1375    1428    1436    1566
TSTB        1151    1162    1184    1259    1264    1299    1365    1439    1472    1506    1509
WAIT        1092
.ASCII      1641    1646    1651    1658    1664    1671    1677    1682    1691    1701    1707    1711    1716    1724    1730
            1737    1745    1753    1759    1766    1773    1781    1799    1796    1804    1811    1515    1913    1823    1827
            1830    1937    1846    1855    1965    1874    1890    1897    1894    1899    1903    1909    1919    1323    1932
            1937    1942    1951    1957    1963    1969    1975    1979    1994    1992    2001    2011    2019    2021    2027
            2035    2037
.BYTE       1626    1630    1633
.ENABL       375
.END        2055
.EVEN       1625    1634
.LIST          1     373
.NLIST         1     373
.PAGE       1107
.REM           1
.REPT        435     596
.TITLE       373



 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *DZLPJB,DZLPJB.SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DZLPJB.P11
 RUN-TIME: 5 11 2 SECONDS
 RUN-TIME RATIO: 42/21=2.0
 CORE USED:  8K  (16 PAGES)
```

C05

Spooler ruuing 7 Seconds, 226 KCS, 192 disk reads, 3 disk writes, 53 pages