FPDDR

IDENTIFICATION

PRODUCT CODE:	MAINDEC-12-DAFPC-A-D REPLACES: MAINDEC-12-DOPC-D
PRODUCT NAME:	FPP-12 ADDRESS TEST
DATE CREATED:	JULY 15, 1972
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	B. LAFLAMME/W. MANTER

1. ABSTRACT
****

THIS PROGRAM IS DESIGNED TO DETECT A FAULT IN THE FPP-12
MEMORY ADDRESSING HARDWARE, ALL OF AVAILABLE MEMORY IS FIRST
SET TO 0707 OR 7070, AT THE START OF EACH PASS THE MEMORY
CONSTANT IS COMPLIMENTED, THE FPP-12 THEN STORES ONE 36 BIT
WORD INTO 3 CONSECUTIVE CORE LOCATIONS, THE FPP READS BACK THIS
36 BIT WORD AND STORES IT INTO THE BASE TABLE THEN GOES INTO
PAUSE, THIS ALLOWS THE OPERATION OF BOTH THE LOAD AND STORE
FUNCTIONS TO BE CHECKED, WHILE THE FPP-12 IS IN PAUSE, THE
PDP CHECKS BOTH 36 BIT WORDS (THE WORD AT THE MEMORY ADDRESS
AND THE WORD IN THE BASE TABLE,) AT THE OPERATORS OPTION THE PDP WILL
THEN CHECK ALL UNUSED MEMORY TO SEE THAT NOTHING HAS CHANGED,
THE PDP THEN CLEARS THE FPP WORD AT THE MEMORY ADDRESS AND
INCREMENTS THE ADDRESS BY 1 MEMORY LOCATION AND REPEATS THE
PROCESS, WHEN ALL OF MEMORY IS CHECKED EXCEPT THE LOCATIONS
OCCUPIED BY THE PROGRAM, AT THE OPERATORS OPTION, THE PROGRAM
WILL RELOCATE ITSELF TO THE NEXT FIELD,

2. REQUIREMENTS

2.1 EQUIPMENT

A. AN FPP-12 FLOATING POINT PROCESSOR

B. A PDP-8 OR PDP-12 WITH AT LEAST 4K OF MEMORY

C. AN ASR33 OR ASR35 TELETYPE

2.2 STORAGE

THE PROGRAM IS LOADED INTO LOCATIONS 0000-2577
OF FIELD 0. THE PROGRAM USES ALL LOCATIONS OF EVERY FIELD
TESTED.

2.3 PRELIMINARY PROGRAMS

ALL PDP-8 OR PDP-12 PROCESSOR AND MEMORY DIAGNOSTICS
FPP-12 INSTRUCTION TESTS 2A, 2B, AND 2C.

3. LOADING PROCEDURE

LOAD THE PROGRAM WITH THE BIN LOADER, DIAL LOADER OR PS-8 LOADER.

4. STARTING PROCEDURE

1) START AT LOCATION 0020 IN FIELD 0.
   THE PROGRAM WILL HALT AT LOCATION 1666.
2) SET SR09-11 TO THE HIGHEST MEMORY FIELD TO BE TESTED.
3) PRESS THE CONTINUE KEY
   THE PROGRAM WILL HALT AGAIN @1672
4) SET THE SWITCH REGISTER AS DESIRED (SEE "SWITCH OPTIONS")
5) PRESS THE CONTINUE KEY TO START THE TEST.

5. RESTART PROCEDURE

A. DETERMINE BY THE INSTRUCTION FIELD LIGHTS WHICH FIELD THE
   PROGRAM IS RUNNING IN.

B. RESTART ACCORDING TO "4. STARTING PROCEDURE", BUT START
   AT LOCATION 0020 IN THE FIELD DESIGNATED BY THE
   INSTRUCTION FIELD LIGHTS.

6, OPERATION INSTRUCTIONS

6,1 NORMAL OPERATION

THE PROGRAM IS NORMALLY OPERATED WITH SWITCH REGISTER = 0000,

1) START 20
2) (HLTs) ← 8-mode — RSW ØØØX X = highest memory field to be tested
3)
4) I/O Reset
5 CONT.

6,2 COMPLETE MEMORY CHECK (SRØ7 & SRØ8)

IF SRØ8 = Ø THE PROGRAM WILL CHECK EVERY LOCATION IN MEMORY EACH TIME AN FPP-12 DATA ERROR OR BASE TABLE ERROR (SEE 9, ERRORS) IS DETECTED, IF THE FPP-12 IS STORING THE FAC INTO THE WRONG LOCATION, THIS WILL FIND THE LOCATION THAT THE FPP DID STORE INTO, THE DIFFERENCE BETWEEN THE FPP FAILING ADDRESS AND THE UNUSED MEMORY ADDRESS MAY POINT TO THE FAILING ADDRESS BIT,

IF SRØ7 = 1 ALL OF MEMORY WILL BE CHECKED EACH TIME THE FPP-12 DOES A STORE REGARDLESS OF THE SETTING OF SRØ8, THIS WILL HAPPEN WHETHER OR NOT THERE WAS AN ERROR DETECTED,

6,3 RELOCATION OPTION (SRØ3)

EVERY TIME THE FPP-12 COMPLETES STORING INTO ALL AVAIBLE MEMORY, THE PROGRAM WILL RELOCATE ITSELF INTO THE NEXT MEMORY FIELD UNDER TEST, IF SRØ3 = 1, THIS RELOCATION IS BYBASSED AND TEH PROGRAM RUNS ANOTHER PASS IN THE SAME MEMORY FIELD,

7.1 SWITCH OPTIONS (SUMMARY)

SEE "REF." PARAGRAPH FOR MORE DETAILED DESCRIPTION.

SWITCH	STATE	REF.	OPERATION
@0	0	9.3	HALT ON ERRORS
	1		BYPASS ERROR HALT
@1	0	9.1	TYPE ERRORS
	1		BYPASS ERROR TYPEOUT
@2	0	9.4	CONTINUE TEST AFTER AN ERROR
	1		LOOP ON ERROR
@3	0	6.3	RELOCATE THE PROGRAM TO THE NEXT FIELD
	1		RUN THE PROGRAM IN THE SAME FIELD.
@4	0	8.	TYPE END OF PASS INFORMATION
	1		SUPPRESS END OF PASS TYPEOUTS
@5	0	8.	CONTINUE TO NEXT PASS
	1		HALT AT END OF A PASS
@6	0	9.5	CONTINUE COMPARE AFTER AN ERROR
	1		START FPP AT THE NEXT ADDRESS AFTER AN ERROR
@7	0	6.2	CHECK FPP DATA ONLY (NORMAL MODE)
	1		CHECK ALL OF MEMORY (COMPLETE MODE)
@8	0	6.2	CHECK ALL OF MEMORY AFTER AN FPP DATA ERROR
	1		CHECK SR07
@9	0	9.7	OUTPUT ERROR INFO TO THE TTY
	1		OUTPUT ERROR INFO TO THE LINE PRINTER
10	0	NONE	PDP-12, 8I, 8E OR 8L LET FPP ACCESSES OVERLAP 4K MEMORY BOUNDRIES
	1		PDP-8 OR LINC-8 DO NOT LET FPP ACCESS OVER 4K MEMORY BOUNDRIES
11	0	9.6	OUTPUT COMPLETE ERROR INFORMATION
	1		OUTPUT SHORT FORM ERRORS

8, END OF PASS
    ****

EACH TIME THE PROGRAM RELOCATES TO FIELD Ø IS CONSIDERED THE
END OF 1 PASS THROUGH THE TEST', IF SRØ3 = 1, (NO RELOCATION)
THE END OF PASS IS WHEN THE FPP-12 STORES INTO THE LAST
LOCATION IN THE LAST FIELD UNDER TEST',
AT THE END OF EACH PASS A PASS COUNTER IS INCREMENTED',
EACH ERROR COUNTER IS CHECKED FOR AN OCCURENCE OF AN ERROR',
FOR EACH NON ZERO ERROR COUNTER THE NUMBER OF ERRORS
OF THAT TYPE (SEE "ERRORS") IS TYPED OUT, THE PASS
NUMBER IS ALSO TYPED, IF SRØ9 = 1 THESE MESSAGES ARE
OUTPUT TO THE LINE PRINTER, SETTING SRØ4 = 1 WILL
SUPPRESS THESE OUTPUTS, EACH TIME THE END OF PASS
INFORMATION IS OUTPUT (SRØ4=Ø) THE ERROR COUNTERS
ARE RESET TO ZERO', THIS MEANS THAT FOR A GIVEN END
OF PASS TYPEOUT, THE ERROR COUNTS ARE THE NUMBER OF
ERRORS DETECTED SINCE THE LAST TYPEOUT',
THE FORMAT OF THE TYPEOUT IS:

XXXX FPP DATA ERRORS
XXXX BASE TABLE ERRORS
XXXX UNUSED MEMORY ERRORS
END PASS XXXX

IF SRØ5 = 1 THE PROGRAM WILL HALT WHETHER THE TYPEOUT
OCCURS OR NOT',

IF SRØ5 = Ø THE END OF PASS HALT WILL BE BYPASSED',

9. ERRORS

9.1 ERROR TYPEOUTS (SR01)

IF SR01=1 TYPEOUTS WILL BE SUPPRESSED IF AN ERROR IS DETECTED.

IF SR01=0 THE TYPE OF ERROR, FAILING ADDRESS, AND ERONEOUS DATA WILL BE TYPED ON THE TTY. IF SR09=1, THE MESSAGE WILL BE OUTPUT TO THE LP08 OR LP12.
THE FORMAT OF THE ERROR TYPEOUT IS:

TYPE OF ERROR

ADDRESS	GOOD	BAD
X XXXX	XXXX	XXXX
	XXXX	XXXX
	XXXX	XXXX

THE "ADDRESS" IS THE FIELD AND ABSOULTE ADDRESS WHERE THE ERROR WAS DETECTED. (SEE INDIVIDUAL ERROR TYPES BELOW FOR THE MEANING OF EACH ADDRESS.)
"GOOD" IS WHAT THE 36 BIT CORRECT DATA WORD FOR THAT LOCATION SHOULD BE. THOUGH THE WORD IS ONE 36 BIT PPP=12 WORD, IT IS OUT-PUT AS THREE 12 BIT WORDS AS FOUND IN 3 CONSECUTIVE MEMORY LOCATIONS. ALL THREE 12 BIT WORDS ARE TYPED EVEN THOUGH ONLY 1 WORD MAY BE IN ERROR.

"BAD" IS THE DATA FOUND IN MEMORY AT "ADDRESS" AND THE NEXT 2 MEMORY LOCATIONS.

9,2 TYPES OF ERRORS

THERE ARE 3 TYPES OF ERRORS CHECKED FOR IN THE PROGRAM, HERE IS A DESCRIPTION OF EACH TYPE OF ERROR, THE HEADING IS WHAT WILL BE TYPED IN THE "TYPE OF ERROR " FIELD OF THE TYPEOUT,

9,2,1 ERROR IN FPP-12 DATA WORD

BAD DATA WAS FOUND IN MEMORY WHERE THE FPP-12 WAS SUPPOSED TO STORE ITS FAC, THE FAC SHOULD HAVE CONTAINED EITHER 5252 2525 5252 OR THE COMPLIMENT OF THIS, IF THE BAD DATA IS SIMILAR TO THIS EXCEPT FOR A FEW BITS THEN THE FPP-12 DROPPED OR PICKED-UP THESE BITS, IF THE BAD DATA IS 7070 OR 0707, THE FPP-12 DID NOT STORE INTO THIS LOCATION, IF THE PROGRAM IS ALLOWED TO CONTINUE WITH SR06-0 IT WILL CHECK ALL UNUSED MEMORY FOR AN ERROR, THIS SHOULD TELL IF THE FPP STORED INTO THE WRONG ADDRESS OR IF IT DID NOT STORE INTO ANY ADDRESS, ONE OTHER POSSIBLE FAILURE THAT COULD CAUSE THIS TYPE OF ERROR IS IF THE BAD DATA IS 5252 5252 5252 OR THE COMPLIMENT OF THIS, THIS WOULD OCCUR IF THE FPP STORED INTO THE CORRECT ADDRESS, BUT STORED THE WRONG PORTION OF THE FAC,

9,2,2 ERROR IN BASE TABLE

AFTER THE FPP STORED ITS FAC INTO MEMORY AT THE TEST ADDRESS IT LOADED THIS DATA BACK INTO THE FAC AND STORED IT INTO THE BASE TABLE. THIS WORD IN THE BASE TABLE WAS FOUND TO BE BAD. IF AN ERROR IN THE FPP-12 DATA WORD (SEE 9,2,1) AND NO ERROR IN THE BASE TABLE OCCURS THE FPP STORED THE CORRECT DATA INTO THE WRONG LOCATION. IT IS UNLIKELY THAT THIS ERROR WOULD OCCUR WITHOUT AN ERROR IN THE FPP-12 DATA WORD HOWEVER IT IS POSSIBLE. IF THIS IS THE CASE THE MOST LOGICAL SUSPECT WOULD THE "FLDA" INSTRUCTION WITH ADDRESS MODE 3 (SINGLE WORD INDIRECT).

THE ADDRESS IN THE TYPEOUT IS THE ADDRESS OF THE BASE TABLE WHERE THE DATA WAS STORED. IF NECESSARY, THE FPP-12 MEMORY ADDRESS CAN BE FOUND IN LOCATIONS 42 & 43 OF THE INSTRUCTION FIELD.

9,2,3 ERROR IN UNUSED MEMORY

BAD DATA WAS FOUND IN A MEMORY LOCATION UNUSED BY THE FPP-12. THE ADDRESS IN THE TYPEOUT IS ADDRESS WHERE THE FIRST ERROR WAS FOUND. THE NEXT 2 WORDS ARE ALSO TYPED IN CASE THIS IS THE FPP-12 DATA WORD. THE GOOD DATA WILL BE 7070 OR 0707. IF THE BAD DATA IS SIMILAR TO THE GOOD DATA THEN THE FPP-12 DATA BREAK MAY HAVE DISTURBED SOME UNUSED MEMORY LOCATIONS. THIS TYPE OF FAILURE IS PROBABLY IN THE MEMORY RATHER THAN IN THE FPP-12 THOUGH EXCESSIVE NOISE ON THE FPP-12 ADDRESS LINES MAY BE THE CAUSE. IF THE BAD DATA IS 5252 2525 5252 OR THE COMPLIMENT OF THIS, THE FPP STORED ITS FAC INTO THIS LOCATION INSTEAD OF THE CORRECT LOCATION. AN ERROR IN THE FPP-12 DATA WORD SHOULD HAVE BEEN TYPED BEFORE THIS ERROR. BY CHECKING THE DIFFERENCE IN ADDRESSES BETWEEN THIS TYPEOUT AND THE FPP DATA ERROR, A BAD ADDRESS BIT CAN BE ISOLATED

9,3 ERROR HALT (SR00)

IF SR00=0 THE PROGRAM WILL HALT AFTER THE TYPEOUT, AT THIS TIME THE FPP=12 IS IN PAUSE, THE FPC CONTAINS 0030 IN BITS 3=14 AND THE INSTRUCTION FIELD IN BITS 0=2, THE OP ADDRESS CONTAINS 0051 IN BITS 3=14 AND THE INSTRUCTION FIELD IN BITS 0=2, THE FAC CONTAINS THE DATA, THIS SHOULD BE 5252 2525 5252 OR THE COMPLIMENT,

9,4 LOOP ON ERROR (SR02)

WHEN AN ERROR IS DETECTED, THE FPP=12 IS IN PAUSE, IF SR02=1 THE PROGRAM ENTERS A LOOP WHICH FORCES THE FPP TO DO CONTINUOUS STORES AND LOADS AT THE BAD ADDRESS, THIS IS DONE BY STARTING THE FPP EVERY TIME IT ENTERS PAUSE, IN THIS LOOP THE OP ADDRESS CAN BE CHECKED WITH A SCOPE, THIS LOOP WILL CONTINUE UNTILL SR02 IS RESET,

9,5 CONTINUE CHECK (SR06)

WHEN AN ERROR OCCURS IF SR06=1 THE PROGRAM WILL BYPASS CHECKING FOR MORE ERRORS IN THE LAST FPP OPERATION, THE PROGRAM WILL CAUSE THE FPP=12 TO STORE INTO THE NEXT MEMORY LOCATION AND CHECK FOR ERRORS THERE, THE REASON FOR THIS OPTION IS THAT THERE COULD BE 3 TYPES OF ERRORS TYPED FOR EACH FPP STORE, THE OPERATOR MAY ONLY WANT THE FIRST DETECTED ERROR TO BE TYPED AND BYPASS FURTHER TYPEOUTS,

9,6 SHORT TYPEOUT (SR11)

IF SR11 = 1 THE ERROR TYPEOUT WILL CONSIST OF THE ADDRESS LINE ONLY, THIS BYPASSES THE 2 HEADER LINES AND THE 2 DATA ONLY LINES,

9,7 LINE PRINTER (SR09)

IF SR09 = 1 THE PROGRAM WILL CHECK WHICH LINE PRINTER (LP08 OR LP12) IS ON THE SYSTEM AND OUTPUT ALL MESSAGES TO THE PRINTER,

IF THERE IS A PRINTER ERROR OR NO PRINTER AVAILABLE, THE OUTPUT MESSAGE WILL BE LOST,

```
                /COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754


                /FPP-12 ADDRESS TEST 2



                /       FADDR2          V71-06567
                /       BILL LAFLAMME / WALTER MANTER
```

```
                /       SWITCH SETTINGS
                /
                /       SR00    0       HALT ON ERROR
                /               1       BYPASS ERROR HALT
                /
                /       SR01    0       TYPE ERRORS
                /               1       BYPASS ERROR TYPEOUT
                /
                /       SR02    0       CONTINUE AFTER AN ERROR
                /               1       LOOP ON ERROR
                /
                /       SR03    0       RELOCATE PROGRAM TO NEXT FIELD
                /               1       RUN PROGRAM IN THE SAME FIELD
                /
                /       SR04    0       TYPE END OF PASS INFORMATION
                /               1       SUPPRESS END OF PASS TYPEOUTS
                /
                /       SR05    0       CONTINUE TO NEXT PASS
                /               1       HALT AT END OF PASS
                /
                /       SR06    0       CONTINUE COMPARE AFTER ERROR
                /               1       START FPP AT NEXT WORD AFTER ERROR
                /
                /       SR07    0       CHECK SR08 (NORMAL MODE)
                /               1       CHECK ALL OF MEMORY (COMPLETE MODE)
                /
                /       SR08    0       CHECK UNUSED MEMORY TO FIND THE
                /                       FPP DATA AFTER AN ERROR IS DETECTED
                /               1       CHECK FPP DATA ONLY
                /
                /       SR09    0       OUTPUT ERRORS TO THE TTY
                /               1       OUTPUT ERRORS TO THE LP08 OR LP12
                /
                /       SR10    0       NOT A PDP-8 OR LINC-8 COMPUTER
                /               1       A PDP-8 OR LINC-8 COMPUTER
                /
                /       SR11    0       OUTPUT COMPLETE ERROR TYPEOUT
                /               1       TYPE ERROR ADDRESS LINE ONLY
```

```
                        
0020    MODE=   20
                        
                        
                / FPP-12 INSTRUCTIONS
                        
0001    FPAUSE= 0001
0000    FLDA=   0000
6000    FSTA=   6000
1030    JA=     1030
0002    FCLA=   0002
                        
                        
                / FPP-12 IOT'S
                        
6553    FPCOM=  6553
6555    FPST=   6555
6556    FPRST=  6556
6552    FPICL=  6552
                        
                        
                / LP08 IOT'S
                        
6663    LSR=    6663
6661    LSF=    6661
6666    LPC=    6666
                        
                        
                / LP12 IOT'S
                        
6651    LSE=    6651
6664    LPR=    6664
6652    LCF=    6652
6654    LLB=    6654
6661    LSD=    6661
                        
                        
                / IOT'S FOR BOTH LINE PRINTERS USED TO CHECK WHICH PRINTER TO USE
                        
6662    SFLG=   6662            /SET PRINTER FLAG FOR TEST
6661    CFLG=   6661            /CHECK WHICH PRINTER IS AVAILABLE
                        
```

```
                                PMODE

          0000                  *0

0000      0000      ZBLOCK      20

          0020                  *20

0020      4777'                 JMS       CLEAR           /NORMAL START
0021      5062                  JMP       PDPT

0022      4776'                 JMS       SE22            /RELOCATE PROGRAM FROM SR
0023      5062                  JMP       PDPT

0024      4775'                 JMS       SE24            /GET DATA FROM SWITCHES
0025      5062                  JMP       PDPT

0026      5774'                 JMP       RESTOR          /RESTORE "BIN" AND JMP TO 7777

0027      0002      K2,         0002
```

```
                        *30

                /FPP-12 INSTRUCTIONS

                /THE FPP-12 GETS DATA FROM DATA1 AND STORES IT INTO
                /THE MEMORY LOCATION IN AFLD AND APNTR AND THE NEXT 2 LOCATIONS
                /THEN CLEARS THE FAC AND LOADS THE DATA BACK INTO THE FAC FROM
                /THE MEMORY ADDRESS.  IT THEN STORES THE DATA INTO DATA 4 TO
                /BE CHECKED.

0030  0001    FPP,    FPAUSE                  /WAIT FOR PDP
0031  0201            FLDA    201             /LOAD DATA WORD FROM BASE TABLE
0032  6600            FSTA    600             /STORE DATA AT ADDRESS IN BASE TABLE
0033  0002            FCLA                    /CLEAR THE FAC
0034  0600            FLDA    600             /LOAD DATA BACK FROM MEMORY
0035  6202            FSTA    202             /STORE DATA IN BASE TABLE
0036  1030    FRJA,   JA                      /FIELD BITS WILL BE CHANGED BY THE
0037  0030            FPP                     /PDP EACH TIME THE PROGRAM IS MOVED
0040  7070    DATA,   7070                    /DATA TO FILL MEMORY WITH
0041  0000    BASE,   0                       /BASE TABLE
0042  0000    AFLD,   0                       /FIELD BITS FOR FSTA AND FLDA
0043  0000    APNTR,  0                       /12 BIT ADDR FOR FSTA AND FLDA
0044  5252    DATA1,  5252                    /DATA THE FPP WILL
0045  2525    DATA2,  2525                    /STORE INTO MEMORY
0046  5252    DATA3,  5252
0047  0000    DATA4,  0                       /DATA THE FPP READ BACK
0050  0000            0                       /FROM MEMORY
0051  0000            0
0052  0000    APT,    0                       /ACTIVE PARAMETER TABLE
0053  0030    FPC,    FPP
0054  0000            0
0055  0041    P0,     BASE                    /BASE TABLE POINTER
0056  0000            0
0057  0000            0
0060  0000            0
0061  0000            0
                /
                /TEST SR-10
                /IF SET FLAG LOCATION LAP
                /THE HOST CP IS A PDP-8 OR LINC-8
                /INHIBIT OVERLAPPING OF 4K BOUNDRIES BY FPP INSTRUCTIONS
                /
0062  7604    POPT,   LAS             /READ SW REG
0063  0027            AND     K2      /EXAMINE SR 10
0064  7640            SZA CLA         /IS IT SET ?
0065  7040            CMA             /YES
0066  3121            DCA     LAP     /CONDITION LAP MEMORY 4K BOUNDRY FLAG
0067  5773'           JMP     START
```

```
                0070            *70     /CONSTANTS AND TEMPORARY REGISTERS

0070    0000    CPNTR,  0               /ADDRESS POINTERS
0071    0000    PPNTR,  0
0072    0000    EPNTR,  0
0073    0000    TPNTR,  0
0074    0000    TADDR,  0
0075    0000    WCNT,   0               /COUNTERS
0076    0000    ECNT,   0
0077    0000    ETCNT,  0               /ERROR TYPEOUT COUNTER
0100    0000    ECNT1,  0               /ERROR COUNT 1
0101    0000    ECNT2,  0               /ERROR COUNT 2
0102    0000    ECNT3,  0               /ERROR COUNT 3
0103    0000    PCT,    0               /PASS COUNT
0104    0000    LIMIT,  0               /UPPER FIELD LIMIT
0105    0000    FSW,    0               /SWITCH
0106    0000    T1,     0
0107    0215    T215,   215
0110    0212    T212,   212
0111    0077    T77,    77
0112    7740    TM40,   -40
0113    0100    T100,   100
0114    0240    T240,   240
0115    0306    CRLF,   0306
0116    0000    EBUF,   0
0117    0000            0
0120    0000            0
0121    0000    LAP,    0               /OVERLAP MEMORY 4K BOUNDRY FLAG

                0200            PAGE
```

```
                        /SET ALL OF MEMORY TO A CONSTANT,
                        /THE CONSTANT WILL BE COMPLIMENTED
                        /EACH TIME THE PROGRAM IS MOVED TO A NEW FIELD
                        /IF SPECIAL ENTRY 0024 HAS NOT BEEN USED, THE
                        /CONSTANT WILL BE 0000 OR 7777,

0200  6552    START,  FPICL                   /CLEAR THE FPP-12
0201  1040            TAD     DATA            /GET DATA WORD
0202  7040            CMA                     /COMPLIMENT IT
0203  3040            DCA     DATA            /STORE NEW DATA WORD
0204  4206            JMS     CLCOR           /SET ALL AVAILABLE CORE TO A CONSTANT
0205  5251            JMP     STPPP           /START THE FPP-12

0206  0000    CLCOR,  0                       /SET MEMORY TO A CONSTANT
0207  4777'           JMS     PFLD            /SET DATA FIELD = PROGRAM FIELD
0210  4776'           JMS     GFLD            /GET PROGRAM FIELD BITS
0211  7650            SNA CLA                 /IS PROGRAM IN FIELD 0
0212  1172            TAD     CEND-3          /YES - BYPASS PROGRAM AREA
0213  3070            DCA     CPNTR           /SET POINTER TO STARTING ADDRESS
0214  1171            TAD     CCDF            /DATA FIELD 0 INSTRUCTION
0215  3216            DCA     CFLD            /SET CLEAR FIELD IOT
0216  6201    CFLD,   CDF     00              /CLEAR MEMORY FIELD POINTER
0217  1040            TAD     DATA            /GET DATA
0220  3470            DCA I   CPNTR           /STORE IN MEMORY
0221  2070            ISZ     CPNTR           /INCREMENT POINTER
0222  5216            JMP     CFLD            /STORE NEXT MEMORY LOCATION
0223  4777'           JMS     PFLD            /END OF FIELD - RESTORE DATA FIELD
0224  1216            TAD     CFLD            /GET OLD FIELD CDF
0225  0170            AND     C70             /EXTRACT FIELD BITS
0226  7112            CLL RTR                 /MOVE FIELD BITS TO AC 9-11
0227  7010            RAR
0230  3106            DCA     T1              /SAVE FIELD BITS
0231  1106            TAD     T1              /RESTORE AC
0232  7041            CIA
0233  1104            TAD     LIMIT           /COMPARE WITH UPPER FIELD LIMIT,
0234  7750            SNA SPA CLA             /FINISHED?
0235  5606            JMP I   CLCOR           /YES - RETURN
0236  2106            ISZ     T1              /INCREMENT FIELD BITS TO NEXT FIELD
0237  4776'           JMS     GFLD            /GET PROGRAM FIELD BITS
0240  7041            CIA
0241  1106            TAD     T1              /COMPARE WITH NEXT FIELD
0242  7650            SNA CLA                 /GOING TO CLEAR PROGRAM FIELD?
0243  1167            TAD     CEND            /YES - BYPASS PROGRAM AREA
0244  3070            DCA     CPNTR           /SET STARTING ADDRESS IN FIELD
0245  1216            TAD     CFLD            /GET OLD FIELD CDF
0246  1166            TAD     C10             /INCREMENT FIELD BITS
0247  3216            DCA     CFLD            /STORE NEXT FIELD CDF
0250  5216            JMP     CFLD            /CLEAR NEXT FIELD
```

```
                        /SET UP THE APT AND START THE FPP-12
                        /SET THE FIELD BITS IN APT0 TO THE PROGRAM FIELD.
                        /SET THE FIELD BITS OF THE "JA" INSTRUCTION TO THE PROGRAM FIELD.
                        /COMPLIMENT THE FPP DATA.

0251  4777'  STFPP,  JMS     PFLD            /SET DATA FLD = INST FLD
0252  6552            FPICL                   /CLEAR THE FPP-12
0253  7200            CLA
0254  6224            RIF                     /GET INSTRUCTION FIELD BITS
0255  7106            RTL CLL
0256  7106            RTL CLL                 /MOVE 6 BITS LEFT
0257  7106            RTL CLL
0260  6224            RIF                     /GET FIELD BITS AGAIN
0261  7112            RTR CLL                 /FIELD BITS ARE NOW IN
0262  7110            RAR CLL                 /BITS 3-5 AND 9-11
0263  3052            DCA     APT             /SET APT FIELD BITS
0264  1165            TAD     CFPP            /GET ADDRESS OF FPP INSTRUCTIONS
0265  3053            DCA     FPC             /SET FPC IN THE APT
0266  3042            DCA     AFLD            /ZERO FIELD BITS IN BASE TABLE
0267  3043            DCA     APNTR           /ZERO ADDRESS IN BASE TABLE
0270  1044            TAD     DATA1           /COMPLIMENT THE DATA
0271  7040            CMA
0272  3044            DCA     DATA1           /FIRST DATA WORD (EXPONENT)
0273  1045            TAD     DATA2
0274  7040            CMA
0275  3045            DCA     DATA2           /SECOND DATA WORD (MSW)
0276  1046            TAD     DATA3
0277  7040            CMA
0300  3046            DCA     DATA3           /THIRD DATA WORD (LSW)
0301  4776'           JMS     GFLD            /GET INST FIELD BITS
0302  1164            TAD     CJA             /ADD FLD BITS TO JA INSTRUCTIONS
0303  3036            DCA     FRJA            /MODIFY FPP-12 PROGRAM FIELD BITS
0304  4776'           JMS     GFLD            /GET PROGRAM FIELD BITS
0305  6553            FPCOM                   /SET FPP-12 COMMAND REGISTER
0306  7200            CLA
0307  1163            TAD     CAPT            /GET APT ADDRESS
0310  6555            FPST                    /START THE FPP-12
0311  7402            HLT                     /FPP-12 DID NOT START
0312  6556            FPRST                   /GET FPP-12 STATUS REGISTER
0313  7012            RTR                     /BIT 10 TO THE LINK
0314  7620            SNL CLA                 /FPP-12 IN PAUSE?
0315  5312            JMP     .-3             /NO-WAIT
0316  5317            JMP     RUN             /YES-RUN TEST
```

```
                        /CHECK THE FPP-12 FIELD AND ADDRESS POINTERS AND
                        /CAUSE THE FPP-12 TO STORE 1 FLOATING POINT WORD
                        /AND PAUSE.
                        
0317  4776'   RUN,     JMS     GFLD            /GET PROGRAM FIELD BITS
0320  7041             CIA
0321  1042             TAD     AFLD            /COMPARE WITH FPP FIELD BITS
0322  7650             SNA CLA                 /MATCH?
0323  1167             TAD     CEND            /YES - BYPASS PROGRAM AREA
0324  3043             DCA     APNTR           /STORE FPP STARTING ADDRESS
0325  1042             TAD     AFLD            /GET FPP FIELD BITS
0326  7104             CLL RAL                 /MOVE TO AC 6-8
0327  7006             RTL
0330  1171             TAD     CCDF            /MAKE CDF THE SAME AS THE FPP FIELD
0331  3775'            DCA     DFLD            /MODIFY COMPARE ROUTINE
0332  6555   FSTEP,    FPST                    /STEP FPP THROUGH THE PROGRAM ONCE
0333  7000             NOP
0334  6556             FPRST                   /GET FPP STATUS
0335  7012             RTR                     /MOVE PAUSE BIT TO LINK
0336  7620             SNL CLA                 /IS THE FPP IN PAUSE
0337  5334             JMP     .-3             /NO - WAIT FOR FPP PAUSE
0340  5774'            JMP     CMPR            /YES - CHECK THE RESULTS
                        /
                        /IF LAP FLAG IS SET
                        /INHIBITING OVERLAPPING OF 4K MEMORY BOUNDRIES
                        /
0341  0000   INCLAP,    0
0342  1121             TAD     LAP
0343  7650             SNA CLA
0344  5351             JMP     .+5
0345  1043             TAD     APNTR
0346  1356             TAD     K3
0347  7650             SNA CLA
0350  5353             JMP     .+3
0351  2043             ISZ     APNTR
0352  5741             JMP I   INCLAP
0353  2341             ISZ     INCLAP
0354  5741             JMP I   INCLAP
0355  7402             HLT                     /SHOULD NEVER GET HERE

0356  0003   K3,       0003
0374  0400
0375  0413
0376  1623
0377  1647
      0400             PAGE
```

```
                        /COMPARE THE FPP-12 DATA WORD THAT WAS STORED INTO THE
                        /ADDRESS IN AFLD AND APNTR AND ALSO THE DATA WORD STORED
                        /INTO DATA 4 WITH THE CORRECT DATA,
                        /THE CORRECT FPP DATA IS IN DATA1,

0400  1043    CMPR,   TAD     APNTR           /GET FPP ADDRESS
0401  3070            DCA     CPNTR           /SET COMPARE POINTER
0402  1162            TAD     CDATA1          /GET ADDRESS OF GOOD DATA
0403  3074            DCA     TADDR           /STORE IN TEMP ADDRESS REGISTER
0404  1161            TAD     CDATA3          /GET ADDRESS OF BASE TABLE DATA
0405  3010            DCA     10              /STORE IN AUTO INDEX REGISTER
0406  1160            TAD     C-3             /GET WORD COUNT
0407  3075            DCA     WCNT            /SET WORD COUNTER FOR 3 WORDS
0410  6224            RIF                     /GET PROGRAM FIELD BITS
0411  1171            TAD     CCDF            /CREATE A RESTORE CDF
0412  3215            DCA     RFLD            /UPDATE THE PROGRAM
0413  6201    DFLD,   CDF                     /CHANGE TO FPP DATA FIELD
0414  1470            TAD I   CPNTR           /GET FPP DATA WORD FROM MEMORY
0415  6201    RFLD,   CDF                     /RESTORE DATA FIELD
0416  7041            CIA
0417  1474            TAD I   TADDR           /COMPARE WITH GOOD DATA
0420  7640            SZA CLA                 /IS FPP MEMORY DATA OK?
0421  4777'           JMS     ERROR1          /NO - REPORT ERROR
0422  1410            TAD I   10              /GET FPP BASE TABLE DATA
0423  7041            CIA
0424  1474            TAD I   TADDR           /COMPARE WITH GOOD DATA
0425  7640            SZA CLA                 /IS FPP BASE TABLE DATA OK?
0426  4776'           JMS     ERROR2          /NO - REPORT ERROR
0427  2075            ISZ     WCNT            /END OF CHECK?
0430  7610            SKP CLA                 /NO - CONTINUE
0431  5253            JMP     CEND            /YES - RESTORE ANY PROGRAM CHANGES
0432  2074            ISZ     TADDR
0433  2070            ISZ     CPNTR           /INCREMENT POINTER
0434  5213            JMP     DFLD            /CHECK NEXT WORD
0435  1213            TAD     DFLD            /FIELD OVERFLOW - MODIFY CDF
0436  0170            AND     C70             /GET OLD FIELD BITS
0437  7110            CLL RAR                 /MOVE TO AC 9-11
0440  7012            RTR
0441  7041            CIA
0442  1104            TAD     LIMIT           /COMPARE WITH LIMIT
0443  7650            SNA CLA                 /WAS THAT THE LAST FIELD?
0444  5255            JMP     LIM7            /YES - CHECK IF LIMIT IS FIELD 7
0445  7240            STA                     /AC=7777
0446  3105            DCA     FSW             /SET FIELD CHANGE SWITCH
0447  1213            TAD     DFLD            /GET FPP DATA CDF
0450  1166            TAD     C10             /INCREMENT FIELD BITS
0451  3213            DCA     DFLD            /STORE NEW CDF
0452  5213            JMP     DFLD            /CHECK NEW WORD
0453  4266    CEND,   JMS     RESCDF          /RESTORE DFLD CDF
0454  5303            JMP     CKMEM           /CHECK UNUSED MEMORY
```

```
                        /IF THE UPPER LIMIT WAS FIELD 7, THE FIRST LOCATIONS OF
                        /FIELD 0 WILL BE CHECKED TO SEE THAT THE FPP-12 CAN
                        /RAP-AROUND FROM FIELD 7 TO FIELD 0.

0455  1104    LIM7,   TAD     LIMIT           /GET LIMIT (SR 9-11)
0456  1137            TAD     [-7             /IS LIMIT=7 (32K OF MEMORY)
0457  7640            SZA CLA                 /IF YES CHECK RAP AROUND
0460  5253            JMP     CEND            /NO - END OF CHECK
0461  7240            STA
0462  3105            DCA     FSW             /SET FIELD CHANGE SWITCH
0463  1171            TAD     [CDF            /GET FIELD 0 CDF
0464  3213            DCA     DFLD            /UPDATE PROGRAM
0465  5213            JMP     DFLD            /CHECK WORDS IN FIELD 0

                        /IF THE CDF INSTRUCTION IN THE COMPARE ROUTINE WAS MODIFIED
                        /BECAUSE THE FPP-12 OVERLAPPED A FIELD, THEN IT MUST BE
                        /RESTORED TO ITS ORIGINAL FIELD.

0466  0000    RESCDF, 0                       /RESTORE DFLD CDF
0467  1105            TAD     FSW             /GET FIELD CHANGE SWITCH
0470  7650            SNA CLA                 /WAS DFLD MODIFIED?
0471  5666            JMP I   RESCDF          /NO - RETURN
0472  3105            DCA     FSW             /YES - RESET FIELD CHANGE SWITCH
0473  1213            TAD     DFLD            /GET MODIFIED CDF
0474  0170            AND     [70             /EXTRACT FIELD BITS
0475  7450            SNA                     /ARE FIELD BITS = 0?
0476  1156            TAD     [100            /YES - 100 IS FOR THIS SPECIAL CASE
0477  1155            TAD     [-10            /SUBTRACT 1 FROM FIELD BITS
0500  1171            TAD     [CDF            /MAKE NEW CDF
0501  3213            DCA     DFLD            /RESTORE DFLD CDF INSTRUCTION
0502  5666            JMP I   RESCDF          /RETURN
```

```
                        /CHECK ALL UNUSED MEMORY TO SEE IF ANY LOCATIONS HAVE BEEN CHANGED.
                        /THE FPP DATA WORD IS CLEARED TO THE MEMORY CONSTANT
                        /BEFORE CHECKING MEMORY,  THIS IS SO THAT AN ERROR WILL NOT
                        /BE DETECTED AT THE FPP DATA ADDRESS.

0503  7604    CKMEM,  LAS                     /GET SWITCH REGISTER
0504  0154            AND     [MODE           /EXTRACT FAST MODE SWITCH
0505  7650            SNA CLA                 /FAST MODE?
0506  5775'           JMP     CMPEND          /YES - BYPASS MEMORY CHECK
0507  1155            TAD     [-10            /GET ERROR COUNT
0510  3077            DCA     ETCNT           /SET COUNTER TO TYPE 10 ERRORS ONLY
0511  4774'           JMS     FPPZ            /CLEAR THE FPP DATA FROM MEMORY
0512  4773'           JMS     GFLD            /GET FIELD BITS IN AC 9-11
0513  7650            SNA CLA                 /IS PROGRAM IN FIELD 0?
0514  1167            TAD     [END            /YES - BYPASS PROGRAM AREA
0515  3070            DCA     CPNTR           /SET COMPARE POINTER TO FIRST WORD
0516  1171            TAD     [CDF            /CDF 00 - START IN FIELD 0
0517  3320            DCA     CMFLD           /SET COMPARE MEMFLD CDF
0520  6201    CMFLD,  CDF                     /CHANGE TO COMPARE DATA FIELD
0521  1470            TAD I   CPNTR           /GET DATA
0522  7041            CIA
0523  1040            TAD     DATA            /COMPARE WITH CORRECT DATA
0524  7640            SZA CLA                 /IS DATA CORRECT?
0525  4772'           JMS     ERROR3          /NO - UNUSED MEMORY ERROR
0526  2070            ISZ     CPNTR           /INCREMENT COMPARE ADDRESS
0527  5320            JMP     CMFLD           /GET NEXT WORD
0530  4771'           JMS     PFLD            /END OF FIELD
0531  1320            TAD     CMFLD           /GET DATA FIELD CDF
0532  0170            AND     [70             /EXTRACT FIELD BITS
0533  7112            CLL RTR                 /MOVE TO BITS 9-11
0534  7010            RAR
0535  3106            DCA     T1              /SAVE FIELD BITS
0536  1106            TAD     T1              /RESTORE AC
0537  7041            CIA
0540  1104            TAD     LIMIT           /COMPARE WITH LIMIT (SR 9-11)
0541  7750            SPA SNA CLA             /WAS THAT THE LAST FIELD?
0542  5775'           JMP     CMPEND          /YES - END OF CHECK
0543  2106            ISZ     T1              /INC DATA FIELD BITS
0544  4773'           JMS     GFLD            /GET PROGRAM FIELD BITS
0545  7041            CIA
0546  1106            TAD     T1              /COMPARE WITH NEXT FIELD
0547  7650            SNA CLA                 /IS PROG FIELD TO BE CHECKED?
0550  1167            TAD     [END            /YES - BYPASS PROGRAM AREA
0551  3070            DCA     CPNTR           /SET POINTER TO FIRST WORD IN FIELD
0552  1320            TAD     CMFLD           /GET DATA FIELD CDF
0553  1166            TAD     [10             /INCREMENT FIELD BITS
0554  3320            DCA     CMFLD           /STORE NEW CDF INSTRUCTION
0555  5320            JMP     CMFLD           /CHECK NEXT FIELD
0571  1647
0572  1131
0573  1623
0574  0614
0575  0600
0576  1054
```

```
478     0577  1026
              0600            PAGE
```

```
                        /CLEAR FPP WORDS AND INC ADDRESS, THE FPP DATA IN
                        /MEMORY MUST BE RESTORED TO THE SAME DATA THAT IS IN THE
                        /REST OF MEMORY BEFORE CONTINUING TO THE NEXT FPP OPERATION.

0600  4214    CMPEND, JMS      FPPZ            /CLEAR FPP DATA
0601  4777'           JMS      PFLD            /SET DATA FIELD = PROGRAM FIELD
0602  4763           JMS I    INCLA           /INCREMENT FPP ADDRESS
0603  5776'          JMP      FSTEP           /NOT END - CHECK NEXT WORD
0604  1042           TAD      AFLD            /GET FPP FIELD BITS
0605  7041           CIA
0606  1104           TAD      LIMIT           /COMPARE WITH UPPER LIMIT
0607  7750           SNA SPA  CLA             /MORE FIELDS?
0610  5245           JMP      CKREL           /NO - RELOCATE PROGRAM TO NEXT FIELD
0611  2042           ISZ      AFLD            /YES - INC FPP FIELD BITS
0612  5775'          JMP      RUN             /CONTINUE TO NEXT ADDRESS
0613  7402           HLT                      /PROGRAM SHOULD NEVER GET HERE
                                               /RELOAD THE PROGRAM

0614  0000    FPPZ,   0                       /CLEAR THE FPP DATA WORDS
0615  7200           CLA
0616  1043           TAD      APNTR           /GET FPP ADDRESS
0617  3070           DCA      CPNTR           /STORE IN CHECK POINTER
0620  1042           TAD      AFLD            /GET FPP FIELD BITS
0621  7106           CLL RTL                  /MOVE TO BITS 6-8
0622  7004           RAL
0623  1171           TAD      [CDF            /CREATE CORRECT CDF INSTRUCTION
0624  3227           DCA      CKFLD           /MODIFY PROGRAM
0625  1160           TAD      [-3             /GET WORD COUNT
0626  3075           DCA      WCNT
0627  6201    CKFLD,  CDF
0630  1040           TAD      DATA            /GET DATA WORD
0631  3470           DCA I    CPNTR           /CLEAR LAST FPP WORD
0632  2075           ISZ      WCNT            /INCREMENT WORD COUNT
0633  7610           SKP CLA                  /CONTINUE
0634  5614           JMP I    FPPZ            /END OF CLEAR
0635  2070           ISZ      CPNTR           /INCREMENT ADDRESS
0636  5227           JMP      CKFLD           /CLEAR NEXT WORD
0637  1227           TAD      CKFLD           /FIELD OVERFLOW - MODIFY CDF
0640  1166           TAD      [10             /INCREMENT FIELD BITS
0641  0170           AND      [70             /KEEP FIELD BITS ONLY
0642  1171           TAD      [CDF            /MAKE NEW CDF INSTRUCTION
0643  3227           DCA      CKFLD           /MODIFY PROGRAM
0644  5227           JMP      CKFLD           /CLEAR NEXT WORD
```

```
                        /CHECK IF PROGRAM IS TO BE RELOCATED (SR03 = 0)

0645  7604    CKREL,  LAS                     /GET THE SWITCH REGISTER
0646  0153            AND     [400            /EXTRACT SR03
0647  7640            SZA CLA                 /SR03 = 0?
0650  5263            JMP     EPASS           /NO = END OF PASS
0651  4774'           JMS     RELO            /YES = RELOCATE PROGRAM
0652  1773'           TAD     NFLD            /GET NEXT FIELD CDF
0653  0170            AND     [70             /EXTRACT FIELD BITS
0654  7450            SNA                     /IS NEXT FIELD 0?
0655  5263            JMP     EPASS           /YES = END OF PASS
0656  1152    ENDREL, TAD     [CIF            /MAKE CORRECT CIF INSTRUCTION
0657  3260            DCA     .+1             /MODIFY PROGRAM
0660  6202            CIF                     /CHANGE TO NEXT INST FIELD
0661  5662            JMP I   .+1             /RUN TEST IN NEXT FIELD
0662  0200            START
```

```
                        /END OF A PASS,
                        /THIS ROUTINE IS ONLY ENTERED IF THE PROGRAM IS CONTINUOUSLY
                        /RELOCATING (SR00 = 0) AND THE PROGRAM IS BEING RELOCATED
                        /TO FIELD 0,  THIS IS CONSIDERED THE END OF ONE PASS THROUGH
                        /ALL PROGRAM FIELDS,  IF THE PROGRAM IS IN FIELD 0 AND ONLY
                        /FIELD 0 IS BEING TESTED, EACH PASS OF THE FPP-12 THROUGH
                        /THE FIELD IS ONE PROGRAM PASS,

0663  6202  EPASS,  CIF      00              /GO TO FIELD 0
0664  5551          JMP I    [,+1            /CONTINUE IN FIELD 0
0665  6201          CDF      00              /DATA FIELD 0
0666  7604          LAS                      /GET THE SWITCH REGISTER
0667  0173          AND      [200            /EXTRACT SR04
0670  7640          SZA CLA                  /TYPE END OF PASS INFORMATION ?
0671  5336          JMP      EP5             /NO = BYPASS TYPEOUT
0672  4772'         JMS      EFLAG           /YES = GET ERROR FLAG
0673  7650          SNA CLA                  /ANY ERRORS IN THIS PASS?
0674  5330          JMP      EP4             /NO=BYPASS ERROR COUNT
0675  4771'         JMS      TYPE            /TYPE "PASS XXXX
0676  2026          LIN2                     /TEXT ADDRESS
0677  1100  EP1,    TAD      ECNT1           /GET ERROR FLAG
0700  7650          SNA CLA                  /ANY FPP ADDRESS ERRORS?
0701  5310          JMP      EP2             /NO
0702  4770'         JMS      ASC             /CONVERT ERROR COUNT TO ASCII
0703  0100          ECNT1                    /ERROR COUNT
0704  2032          PET1+1                   /TEXT ADDRESS
0705  4771'         JMS      TYPE            /TYPE NO. OF ERRORS
0706  2031          PET1                     /TEXT ADDRESS
0707  5310          JMP      EP2             /CHECK ERROR COUNT 2


0710  1101  EP2,    TAD      ECNT2           /GET ERROR FLAG
0711  7650          SNA CLA                  /ANY BASE TABLE ERRORS?
0712  5320          JMP      EP3             /NO
0713  4770'         JMS      ASC             /CONVERT ERROR COUNT TO ASCII
0714  0101          ECNT2                    /ADDRESS OF ERROR COUNT
0715  2046          PET2+1                   /TEXT ADDRESS
0716  4771'         JMS      TYPE            /TYPE ERROR COUNT
0717  2045          PET2                     /ADDRESS OF TEXT STRING
0720  1102  EP3,    TAD      ECNT3           /GET LAST ERROR COUNT
0721  7650          SNA CLA                  /ANY UNUSED MEMORY ERRORS ?
0722  5330          JMP      EP4             /NO = BYPASS TYPEOUT
0723  4770'         JMS      ASC             /CONVERT ERROR COUNT TO ASCII
0724  0102          ECNT3                    /ERROR COUNT
0725  2063          PET3+1                   /TEXT ADDRESS
0726  4771'         JMS      TYPE            /TYPE ERROR COUNT
0727  2062          PET3                     /TEXT ADDRESS
```

```
0730  4770'  EP4,   JMS     ASC             /CONVERT PASS COUNT TO ASCII
0731  0103         PCT                     /PASS COUNTER
0732  2106         EPTV                    /TEXT ADDRESS
0733  4771'        JMS     TYPE            /TYPE "END PASS XXXX"
0734  2100         EPT                     /TEXT ADDRESS
0735  4767'        JMS     CLRCT           /ZERO ERROR COUNTERS
0736  2103  EP5,   ISZ     PCT             /INCREMENT PASS COUNTER
0737  7000         NOP
0740  7604         LAS                     /GET SWITCH REGISTER
0741  0156         AND     C100            /EXTRACT SR05
0742  7640         SZA CLA                 /IS SR05 SET?
0743  7402         HLT                     /NO = END OF PASS HALT
0744  5766'        JMP     START           /RUN NEXT PASS


                   /RESTORE THE LAST PAGE IN FIELD 0 AND
                   /JUMP TO THE BINARY LOADER
                   /THIS ROUTINE IS ENTERED BY THE OPERATOR AT THE CONSOLE.
                   /IT CAN IETHER BE ENTERED HERE OR BY STARTING AT LOCATION 0026.

0745  7200  RESTOR, CLA
0746  1150         TAD     C7600           /GET ADDRESS OF LAST PAGE
0747  3070         DCA     CPNTR           /SAVE IN POINTER
0750  1147         TAD     CSBUF-1         /ADDRESS OF SAVE BUFFER
0751  3010         DCA     10              /SAVE IN AUTO INDEX REGISTER
0752  4777'        JMS     PFLD            /SET DATA FIELD=PROG FIELD
0753  1410         TAD I   10              /GET WORD FROM BUFFER
0754  6201         CDF     00              /STORING IN FIELD 0
0755  3470         DCA I   CPNTR           /STORE WORD IN LAST PAGE
0756  2070         ISZ     CPNTR           /INCREMENT POINTER
0757  5352         JMP     .-5             /MOVE NEXT WORD
0760  6202         CIF     00              /FINISHED=GOING TO FIELD 0
0761  5762         JMP I   .+1             /JUMP TO BINARY LOADER
0762  7777         7777

0763  0341  INCLA,  INCLAP

0766  0200
0767  1631
0770  1400
0771  1444
0772  1637
0773  1017
0774  1000
0775  0317
0776  0332
0777  1647
      1000         PAGE
```

```
0645
0646
0647
0648                    /THIS ROUTINE WILL RELOCATE THE PROGRAM TO THE NEXT MEMORY
0649                    /FIELD TO BE USED.  IF THE PROGRAM IS IN FIELD 0 AND ONLY
0650                    /FIELD 0 IS BEING TESTED, THE PROGRAM GOES THROUGH THE MOTIONS
0651                    /OF RELOCATING ANYWAY.  THIS IS DONE TO SAVE THE NECESSARY
0652                    /CORE TO CHECK FOE THIS CONDITION.
0653
0654  1000  0000  RELO, 0
0655  1001  4777'       JMS     PFLD            /SET PROGRAM DATA FIELD
0656  1002  4776'       JMS     GFLD            /GET FIELD BITS IN BITS 9-11
0657  1003  7041        CIA
0658  1004  1104        TAD     LIMIT           /COMPARE WITH UPPER LIMIT
0659  1005  7750        SPA SNA CLA             /LAST FIELD?
0660  1006  5211        JMP     .+3             /YES-RELOCATE TO FIELD 0
0661  1007  6224        RIF                     /GET THIS FIELD BITS
0662  1010  1166        TAD     C10             /INCREMENT TO NEXT FIELD
0663  1011  1171  RE22, TAD     CCDF            /CREATE CDF INSTRUCTION
0664  1012  3217        DCA     NFLD            /SET NEW FIELD CDF
0665  1013  3071        DCA     PPNTR           /SET PROGRAM POINTER TO 0
0666  1014  1146        TAD     C-END           /SET PROGRAM WORD COUNT
0667  1015  3075        DCA     WCNT            /SET WORD COUNTER
0668  1016  1471  MOVEP, TAD I   PPNTR          /GET PROGRAM FROM THIS FIELD
0669  1017  6201  NFLD, CDF     00              /CHANGE TO NEW DATA FIELD
0670  1020  3471        DCA I   PPNTR           /STORE PROGRAM IN NEXT FIELD
0671  1021  4777'       JMS     PFLD            /SET DATA FLD = PROG FLD
0672  1022  2071        ISZ     PPNTR           /INCREMENT PROGRAM POINTER
0673  1023  2075        ISZ     WCNT            /INCREMENT WORD COUNT
0674  1024  5216        JMP     MOVEP           /GET NEXT WORD
0675  1025  5600        JMP I   RELO            /FINISHED
0676
0677
0678                    /FPP DATA ERROR IN MEMORY.  THE DATA IN THE ADDRESS WHERE
0679                    /THE FPP SHOULD HAVE STORED WAS FOUND TO BE BAD.
0680
0681  1026  0000  ERROR1, 0                     /FPP ADDRESS ERROR
0682  1027  2100        ISZ     ECNT1           /INCREMENT ERROR COUNT
0683  1030  5233        JMP     .+3             /NO OVERFLOW
0684  1031  7240        STA                     /7777 IS MAX ERROR COUNT
0685  1032  3100        DCA     ECNT1           /RESTORE COUNT AFTER OVERFLOW
0686  1033  1043        TAD     APNTR           /GET ADDRESS OF BAD DATA
0687  1034  3072        DCA     EPNTR           /STORE IN ERROR POINTER
0688  1035  4775'       JMS     RESCDF          /RESTORE DFLD CDF IF IT WAS CHANGED
0689  1036  1774'       TAD     DFLD            /GET BAD DATA CDF
0690  1037  4773'       JMS     ERROR           /REPORT ERROR
0691  1040  0044        DATA1                   /ADDRESS OF GOOD DATA
0692  1041  1674        ETXT1                   /ADDRESS OF ERROR TEXT
0693  1042  5772'       JMP     CMPEND          /SR06 = 1 START FPP AT NEXT WORD
0694  1043  1474  BTCK, TAD I   TADDR           /CONTINUE BASE TABLE CHECK
0695  1044  7041        CIA
0696  1045  1410        TAD I   10              /GET NEXT WORD FROM BASE TABLE
0697  1046  7640        SZA CLA                 /IS BASE TABLE DATA OK?
0698  1047  4254        JMS     ERROR2          /BASE TABLE DATA ERROR
0699  1050  2074        ISZ     TADDR           /INCREMENT TEST ADDRESS
```

```
700     1051  2075            ISZ     WCNT            /END OF CHECK?
701     1052  5243            JMP     BTCK            /NO-CHECK NEXT WORD
702     1053  5272            JMP     END12           /GO TO END TO CHECK SR
703
```

```
704
705                     /ERROR IN THE BASE TABLE.  THE DATA IN DATA4 IN THE BASE
706                     /TABLE WAS FOUND TO BE BAD.
707
708     1054  0000      ERROR2, 0                       /FPP BASE TABLE ERROR
709     1055  2101              ISZ     ECNT2           /INCREMENT ERROR COUNT
710     1056  5261              JMP     .+3             /NO OVERFLOW
711     1057  7240              STA                     /7777 IS MAX ERROR COUNT
712     1060  3101              DCA     ECNT2           /RESTORE COUNT AFTER OVERFLOW
713     1061  1145              TAD     [DATA2          /GET ADDRESS OF BAD DATA
714     1062  3072              DCA     EPNTR           /STORE IN ERROR POINTER
715     1063  4775'             JMS     RESCDF          /RESTORE DFLD CDF IF IT WAS CHANGED
716     1064  1171              TAD     [CDF            /GET CDF INSTRUCTION
717     1065  6224              RIF                     /ADD PROGRAM FIELD BITS
718     1066  4773'             JMS     ERROR           /REPORT ERROR
719     1067  0044              DATA1                   /ADDRESS OF GOOD DATA
720     1070  1713              ETXT2                   /ADDRESS OF ERROR TEXT
721     1071  5772'             JMP     CMPEND          /SR06 = 1 START FPP AT NEXT WORD
722     1072  7604    END12,    LAS                     /GET SWITCH REGISTER
723     1073  0166              AND     [10             /EXTRACT SR08
724     1074  7640              SZA CLA                 /IS SR08 = 1 ?
725     1075  5771'             JMP     CKMEM           /YES = CHECK SR07
726     1076  5770'             JMP     CKMEM+4         /NO = FIND THE FPP DATA
727
728
```

```
                        /SPECIAL ENTRY 24', ENTERED BY STARTING THE PDP AT
                        /LOCATION 0024', THIS ROUTINE ALLOWS THE OPERATOR TO
                        /SELECT HIS OWN MEMORY CONSTANT AND FPP DATA VIA THE
                        /SWITCH REGISTER.

1077  0000    SE24,   0                       /SPECIAL ENTRY 24
1100  7602            HLT CLA                 /SET SR TO MEMORY DATA
1101  7604            LAS                     /GET SWITCH REG
1102  3040            DCA     DATA            /STORE IN MEMORY DATA
1103  7602            HLT CLA                 /SET SR TO FPP EXPONENT
1104  7604            LAS                     /GET SWITCH REG
1105  3044            DCA     DATA1           /STORE DATA
1106  7602            HLT CLA                 /SET SR TO FPP MSW
1107  7604            LAS
1110  3045            DCA     DATA2           /STORE DATA
1111  7602            HLT CLA                 /SET SR TO FPP LSW
1112  7604            LAS
1113  3046            DCA     DATA3           /STORE DATA
1114  7602            HLT CLA                 /SET SR TO RUN
1115  5677            JMP I   SE24            /START PROGRAM


                        /SPECIAL ENTRY 22', ENTERED BY STARTING THE PDP AT
                        /LOCATION 0022', THIS ROUTINE ALLOWS THE OPERATOR TO
                        /RELOCATE THE PROGRAM TO ANY DESIRED FIELD BEFORE
                        /RUNNING THE TEST'.

1116  0000    SE22,   0                       /SPECIAL ENTRY 22
1117  7602            HLT CLA                 /SET SR9-11 TO FIELD
1120  1176            TAD     [SE22           /GET ADDRESS OF THIS ROUTINE
1121  3200            DCA     RELO            /SET RELOCATER RETURN ADDRESS
1122  7604            LAS                     /GET SWITCH REGISTER
1123  0144            AND     [7              /DELETE BITS 0-8
1124  7104            RAL CLL                 /MOVE TO BITS 6-8
1125  7006            RTL
1126  5211            JMP     RE22            /RELOCATE PROGRAM
1127  7602            HLT CLA                 /SET SWITCH REG FOR RUNNING
1130  5020            JMP     20              /START PROGRAM
```

```
                                /UNUSED MEMORY ERROR = THE MEMORY CONSTANT WAS WRONG
                                /IN A LOCATION NOT USED BY THE FPP-12.

1131  0000  ERROR3, 0                        /UNUSED MEMORY ERROR
1132  4777'        JMS     PFLD
1133  7604         LAS                      /GET SWITCH REGISTER
1134  7104         RAL CLL                  /SR01 TO SIGN
1135  7710         SPA CLA                  /TYPE ERRORS ?
1136  5344         JMP     .+6              /NO = BYPASS TYPEOUT COUNTER
1137  2077         ISZ     ETCNT            /HAVE 7 ERRORS BEEN TYPED ?
1140  5344         JMP     .+4              /NO = TYPE ERROR MESSAGE
1141  4767'        JMS     TYPE             /YES = TYPE 2 BLANK LINES
1142  2026         LIN2                     /TEXT ADDRESS
1143  5772'        JMP     CMPEND           /BYPASS FURTHER CHECKING
1144  2102         ISZ     ECNT3            /INCREMENT ERROR COUNT
1145  5350         JMP     .+3              /NO OVERFLOW
1146  7240         STA                      /7777 IS MAX ERROR COUNT
1147  3102         DCA     ECNT3            /RESTORE COUNT IF OVERFLOW
1150  1070         TAD     CPNTR            /GET ADDRESS OF BAD DATA
1151  3072         DCA     EPNTR            /STORE IN ERROR POINTER
1152  1766'        TAD     CMFLD            /GET BAD DATA FIELD CDF
1153  4773'        JMS     ERROR            /REPORT ERROR
1154  2375         END-3                    /ADDRESS OF GOOD DATA
1155  1727         ETXT3                    /ADDRESS OF ERROR TEXT
1156  7610         SKP CLA                  /NO MORE CHECKING
1157  5731         JMP I   ERROR3           /CONTINUE COMPARE
1160  4765'        JMS     CLCOR            /CLEAR MEMORY ERRORS
1161  5772'        JMP     CMPEND           /END OF CHECK

1165  0206
1166  0520
1167  1444
1170  0507
1171  0503
1172  0600
1173  1200
1174  0413
1175  0466
1176  1623
1177  1647
      1200         PAGE
```

```
                        /THIS ROUTINE HANDLES THE ERROR TYPEOUT, ERROR HALT,
                        /AND LOOP ON ERROR CONDITIONS.  IT IS ENTERED FROM ONE
                        /OF THE 3 INDIVIDUAL ERROR ROUTINES.  WHEN ENTERED, THE AC
                        /CONTAINES A "CDF" INSTRUCTION FOR THE FIELD THAT THE BAD
                        /DATA IS IN.  LOCATION "EPNTR" CONTAINES THE ADDRESS OF THE BAD
                        /DATA.  THE LOCATION FOLLOWING THE "JMS" TO THIS ROUTINE CONTAINS
                        /THE ADDRESS OF THE GOOD DATA.

1200  0000  ERROR,  0                        /REPORT ERRORS
1201  3231          DCA     EFLD             /STORE CDF IN PROGRAM
1202  1600          TAD I   ERROR            /GET GOOD DATA ADDRESS
1203  3271          DCA     EGD              /STORE IN CONVERT ROUTINE
1204  2200          ISZ     ERROR            /INCREMENT RETURN
1205  1600          TAD I   ERROR            /GET TEXT ADDRESS
1206  3265          DCA     ETXT             /STORE IN TEXT POINTER
1207  2200          ISZ     ERROR            /INCREMENT RETURN
1210  7604          LAS                      /GET SWITCH REGISTER
1211  7104          RAL CLL                  /BIT 1 TO THE SIGN
1212  7710          SPA CLA                  /TYPE ERRORS?
1213  5320          JMP     ENDET            /NO-BYPASS TYPEOUT
1214  1231          TAD     EFLD             /GET BAD DATA FIELD CDF
1215  0170          AND     [70              /EXTRACT FIELD BITS
1216  7104          CLL RAL                  /MOVE TO AC 3-5
1217  7006          RTL
1220  1143          TAD     [6040            /CONVERT TO ASCII
1221  3777'         DCA     EFLDT            /STORE IN ERROR FIELD TEXT
1222  4776'         JMS     ASC              /CONVERT ADDRESS TO ASCII
1223  0072          EPNTR                    /ADDRESS POINTER
1224  1747          EADT                     /ERROR ADDRESS TEXT
1225  1160          TAD     [-3              /GET WORD COUNT
1226  3076          DCA     ECNT             /SET WORD COUNTER
1227  1142          TAD     [EBUF-1          /GET ERROR BUFFER ADDRESS
1230  3012          DCA     12               /STORE IN AUTO INDEX REG
1231  6201  EFLD,   CDF                      /CHANGE TO BAD DATA FIELD
1232  1472          TAD I   EPNTR            /GET BAD DATA
1233  4775'         JMS     PFLD             /RESTORE DATA FIELD
1234  3412          DCA I   12               /STORE DATA IN ERROR BUFFER
1235  2076          ISZ     ECNT             /END OF DATA?
1236  7610          SKP CLA                  /NO-CONTINUE
1237  5252          JMP     ECONV            /YES-CONVERT TO ASCII
1240  2072          ISZ     EPNTR            /INC BAD DATA POINTER
1241  5231          JMP     EFLD             /GET NEXT WORD
1242  1231          TAD     EFLD             /FIELD OVERFLOW-GET CDF
1243  0170          AND     [70              /EXTRACT FIELD BITS
1244  1141          TAD     [-70             /CHECK IF FIELD 7
1245  7440          SZA                      /IF FIELD 7-GO TO FIELD 0
1246  1156          TAD     [100             /NOT FIELD 7-ADD 1 TO FIELD BITS
1247  1171          TAD     [CDF             /MAKE NO CDF INSTRUCTION
1250  3231          DCA     EFLD             /UPDATE PROGRAM
1251  5231          JMP     EFLD             /MOVE NEXT WORD
```

```
870
871    1252  7240    ECONV,  STA                     /AC = -1
872    1253  3076            DCA     ECNT            /SET WORD COUNT
873    1254  1140            TAD     CEBUF           /GET BAD DATA ADDRESS
874    1255  3274            DCA     EBD             /SET BAD DATA POINTER
875    1256  7604            LAS                     /GET SWITCH REGISTER
876    1257  7010            RAR                     /MOVE AC 11 TO THE LINK
877    1260  7630            SZL CLA                 /SHORT TYPEOUT ?
878    1261  5270            JMP     EGD-1           /YES - BYPASS HEADER
879    1262  1160            TAD     C-3             /NO - SET WORD COUNT FOR 3 LINES
880    1263  3076            DCA     ECNT
881    1264  4774'           JMS     TYPE            /TYPE ERROR HEADER
882    1265  1674    ETXT,   ETXT1                   /TEXT ADDRESS
883    1266  4774'           JMS     TYPE
884    1267  1761            ETXT4
885    1270  4776'           JMS     ASC             /CONVERT GOOD DATA TO ASCII
886
887
888    1271  0044    EGD,    DATA1                   /DATA ADDRESS
889    1272  1753            EGDT                    /GOOD DATA TEXT ADDRESS
890    1273  4776'           JMS     ASC             /CONVERT BAD DATA TO ASCII
891    1274  0116    EBD,    EBUF                    /BAD DATA ADDRESS
892    1275  1756            EBDT                    /BAD DATA TEXT ADDRESS
893    1276  4774'           JMS     TYPE            /TYPE DATA
894    1277  1745            EDT                     /DATA TEXT ADDRESS
895    1300  1137            TAD     C4040           /ASCII SPACES
896    1301  3777'           DCA     EFLDT           /SPACES TO FIELD TEXT
897    1302  1137            TAD     C4040           /ASCII SPACES
898    1303  3773'           DCA     EADT            /SPACES TO ADDRESS TEXT
899    1304  1137            TAD     C4040           /ASCII SPACES
900    1305  3772'           DCA     EADT+1          /SPACES TO SECOND WORD OF TEXT
901    1306  2271            ISZ     EGD             /INC GOOD DATA ADDRESS
902    1307  2274            ISZ     EBD             /INC BAD DATA ADDRESS
903    1310  2076            ISZ     ECNT            /END OF TYPEOUT?
904    1311  5270            JMP     EGD-1           /NO-TYPE NEXT WORD
905    1312  5320            JMP     ENDET           /YES
906    1313  7604    ERET,   LAS                     /GET SWITCH REGISTER
907    1314  0136            AND     C40             /EXTRACT SR06
908    1315  7650            SNA CLA                 /CONTINUE COMPARE ?
909    1316  2200            ISZ     ERROR           /YES - INCREMENT RETURN
910    1317  5600            JMP I   ERROR           /ERROR EXIT
911
```

```
                                                    /GET SWITCH REGISTER
1320  7604  ENDET,  LAS
1321  7710          SPA CLA                         /HALT ON ERROR?
1322  5326          JMP     .+4                     /NO
1323  1072          TAD     EPNTR                   /YES-GET BAD DATA ADDRESS
1324  1135          TAD     C-2                     /RESTORE TOORIGINAL ADDRESS
1325  7402          HLT                             /ERROR HALT
1326  7604  ERLOOP, LAS                             /GET SWITCH REGISTER
1327  7106          RTL CLL                         /SR02 TO THE SIGN
1330  7700          SMA CLA                         /LOOP ON ERROR?
1331  5313          JMP     ERET                    /NO-RETURN
1332  6555          FPST                            /YES-START THE FPP-12
1333  7000          NOP
1334  6556          FPRST                           /GET FPP STATUS
1335  7112          CLL RTR                         /PAUSE BIT TO LINK
1336  7620          SNL CLA                         /IS FPP IN PAUSE?
1337  5334          JMP     .-3                     /NO-WAIT FOR PAUSE
1340  5326          JMP     ERLOOP                  /YES-CHECK SR02 AGAIN
1372  1750
1373  1747
1374  1444
1375  1647
1376  1400
1377  1746
      1400          PAGE
```

```
                        /CONVERT OCTAL WORD TO 6 BIT ASCII
                        /PACK 2 CHARACTERS PER WORD,

1420  0000  ASC,   0
1421  1600          TAD I   ASC             /GET ADDRESS OF DATA WORD
1422  3241          DCA     ASC4            /SAVE
1423  2200          ISZ     ASC             /INCREMENT RETURN
1424  1600          TAD I   ASC             /GET TEXT ADDRESS
1425  3240          DCA     ASC3            /SAVE TEXT ADDRESS
1426  2200          ISZ     ASC             /INCREMENT RETURN
1427  1243          TAD     ASC77           /GET MASK
1430  7040          CMA                     /LEFT HALF
1431  0641          AND I   ASC4            /EXTRACT LEFT HALF OF DATA
1432  7112          CLL RTR                 /MOVE TO RIGHT HALF
1433  7012          RTR
1434  7012          RTR
1435  4223          JMS     ASCB            /CONVERT LEFT HALF
1436  2240          ISZ     ASC3            /INCREMENT TEXT ADDRESS
1437  1243          TAD     ASC77           /GET MASK
1440  0641          AND I   ASC4            /EXTRACT RIGHT HALF OF DATA WORD
1441  4223          JMS     ASCB            /CONVERT RIGHT HALF
1442  5600          JMP I   ASC             /EXIT

1443  0000  ASCB,   0                       /CONVERT 2 OCTAL DIGITS
1444  3242          DCA     ASC5            /SAVE DATA
1445  1242          TAD     ASC5            /RESTORE DATA
1446  7006          RTL                     /MOVE DATA 1 DIGIT LEFT
1447  7004          RAL
1450  0236          AND     ASC1            /DELETE RIGHT DIGIT
1451  1242          TAD     ASC5            /GET CORRECT RIGHT DIGIT
1452  0236          AND     ASC1            /SAVE ONLY 2 CORRECT DIGITS
1453  1237          TAD     ASC2            /INSERT ASCII MODIFIER
1454  3640          DCA I   ASC3            /STORE CONVERTED DATA
1455  5623          JMP I   ASCB            /RETURN

1456  0707  ASC1,   0707
1457  6060  ASC2,   6060
1460  0000  ASC3,   0
1461  0000  ASC4,   0
1462  0000  ASC5,   0
1463  0077  ASC77,  77
```

```
                        /THIS ROUTINE UNPACKES 6 BIT PACKED ASCII CHARACTERS AND
                        /OUTPUTS THEM TO THE TELETYPE.  A WORD CONTAINING OCTAL
                        /7472 (<1) IS DECODED AS A CARRIAGE RETURN - LINE FEED.

1444  0000  TYPE,   0                       /TYPE 6 BIT PACKED ASCII
1445  7200          CLA
1446  1644          TAD I   TYPE            /GET TEXT ADDRESS
1447  3073          DCA     TPNTR           /STORE IN POINTER
1450  2244          ISZ     TYPE            /INCREMENT RETURN
1451  1473  TCRLF,  TAD I   TPNTR           /GET TEXT WORD
1452  1115          TAD     CRLF            /ADD CARRIAGE RETURN CONSTANT
1453  7640          SZA CLA                 /CARRIAGE RETURN (<1)?
1454  5262          JMP     .+6             /NO-CONVERT DATA
1455  1107          TAD     T215            /ASCII RETURN
1456  4320          JMS     TOUT            /TYPE IT
1457  1110          TAD     T212            /LINE FEED
1460  4320          JMS     TOUT            /TYPE IT
1461  5271          JMP     TRET            /RETURN FOR NEXT WORD
1462  1473          TAD I   TPNTR           /GET TEXT WORD
1463  7112          CLL RTR                 /MOVE LEFT CHARACTER
1464  7012          RTR                     /TO RIGHT HALF OF AC
1465  7012          RTR
1466  4273          JMS     TYPA            /CONVERT AND TYPE LEFT CHARACTER
1467  1473          TAD I   TPNTR           /GET TEXT WORD AGAIN
1470  4273          JMS     TYPA            /CONVERT AND TYPE RIGHT CHARACTER
1471  2073  TRET,   ISZ     TPNTR           /INC TEXT POINTER
1472  5251          JMP     TCRLF           /GET NEXT WORD

1473  0000  TYPA,   0                       /CONVERT AC 6-11 TO TRUE ASCII
1474  0111          AND     T77             /DELETE AC 0-5
1475  7450          SNA                     /END OF TEXT STRING?
1476  5337          JMP     PRT             /YES - PRINT LINE AND RETURN
1477  1112          TAD     TM40            /SUBTRACT 40
1500  7510          SPA                     /LESS THAN 40?
1501  1113          TAD     T100            /YES-300 SERIES CHAR-ADD 100
1502  1114          TAD     T240            /ADD ORIGINAL 40+200
1503  4320          JMS     TOUT            /TYPE CHARACTER
1504  5673          JMP I   TYPA            /GET NEXT CHARACTER

1505  0000  TTY,    0
1506  1336          TAD     TCHR            /GET CHARACTER
1507  6046          TLS                     /OUTPUT CHARACTER
1510  6041          TSF                     /WAIT FOR FLAG
1511  5310          JMP     .-1
1512  6042          TCF                     /CLEAR THE TTY FLAG
1513  7200          CLA
1514  6031          KSF                     /IS KEYBOARD FLAG SET?
1515  5705          JMP I   TTY             /NO-RETURN
1516  6032          KCC                     /YES-CLEAR FLAG
1517  5644          JMP I   TYPE            /BYPASS REMAINING TEXT
```

```
1036
1037    1520  0000    TOUT,   0                       /CHECK OUTPUT DEVICE
1038    1521  3336            DCA     TCHR            /SAVE CHARACTER
1039    1522  7604            LAS                     /GET SWITCH REGISTER
1040    1523  7012            RTR                     /MOVE SR09 TO THE LINK
1041    1524  7010            RAR
1042    1525  7630            SZL CLA                 /IS OUTPUT TO A PRINTER ?
1043    1526  5331            JMP     .+3             /YES
1044    1527  4305            JMS     TTY             /NO - OUTPUT TO THE TTY
1045    1530  5720            JMP I   TOUT            /RETURN
1046    1531  6662            SFLG                    /SET OR CLEAR FLAG IN PRINTER
1047    1532  6661            CFLG                    /WHICH PRINTER IS AVAILABLE ?
1048    1533  4363            JMS     LP08            /IT IS THE LP08
1049    1534  4777'           JMS     LP12            /IT IS THE LP12
1050    1535  5720            JMP I   TOUT            /RETURN
1051
1052    1536  0000    TCHR,   0                       /SAVE OUTPUT CHARACTER HERE
1053
1054                    /IF A PRINTER WAS USED, PRINT THE LINE AND EXIT
1055
1056    1537  7604    PRT,    LAS                     /GET THE SWITCH REGISTER
1057    1540  7012            RTR                     /MOVE SR09 TO THE LINK
1058    1541  7010            RAR
1059    1542  7620            SNL CLA                 /WAS THE TTY USED ?
1060    1543  5644            JMP I   TYPE            /YES - EXIT
1061    1544  6662            SFLG                    /SET OR CLEAR PRINTER FLAG
1062    1545  6661            CFLG                    /WHICH LINE PRINTER ?
1063    1546  5355            JMP     PRT8            /LP08
1064    1547  1166            TAD     C10             /LP12 END LINE CONTROL
1065    1550  6652            LCF                     /CLEAR THE FLAGS
1066    1551  6664            LPR                     /PRINT THE LINE
1067    1552  6661            LSD                     /WAIT FOR PRINTER TO FINISH
1068    1553  5352            JMP     .-1
1069    1554  5644            JMP I   TYPE            /EXIT
1070    1555  1107    PRT8,   TAD     T215            /CARRIAGE RETURN CHARACTER
1071    1556  6666            LPC                     /PRINT THE LINE
1072    1557  6661            LSF                     /WAIT FOR PRINTER TO FINISH
1073    1560  5357            JMP     .-1
1074    1561  7300            CLA CLL
1075    1562  5644            JMP I   TYPE            /EXIT
1076
```

```
1077
1078                    /OUTPUT TO THE LP08 LINE PRINTER
1079
1080    1563  0000      LP08,   0
1081    1564  6663              LSR                     /CHECK FOR PRINTER ERROR
1082    1565  7610              SKP CLA                 /PRINTER IS OK
1083    1566  5763              JMP I   LP08            /PRINTER IS NOT READY - EXIT
1084    1567  1336              TAD     TCHR            /GET OUTPUT CHARACTER
1085    1570  6666              LPC                     /OUTPUT IT TO THE PRINTER
1086    1571  6661              LSF                     /WAIT FOR PRINTER TO FINISH
1087    1572  5371              JMP     .-1
1088    1573  7300              CLA CLL
1089    1574  2363              ISZ     LP08            /INC RETURN OVER LP12 CALL
1090    1575  5763              JMP I   LP08            /EXIT
1091
1092
1093    1577  1600
              1600              PAGE
1094
1095                    /OUTPUT TO THE LP12 PRINTER
1096
1097    1600  0000      LP12,   0
1098    1601  6651              LSE                     /CHECK FOR PRINTER ERROR
1099    1602  7610              SKP CLA                 /PRINTER IS OK
1100    1603  5600              JMP I   LP12            /PRINTER IS NOT READY - EXIT
1101    1604  1534              TAD I   CTCHR           /GET OUTPUT CHARACTER
1102    1605  1133              TAD     C-212           /SUBTRACT LINE FEED
1103    1606  7450              SNA                     /LINE FEED ?
1104    1607  5221              JMP     LP12E           /YES - IGNORE IT
1105    1610  1160              TAD     C-3             /SUBTRACT CARRIAGE RETURN
1106    1611  7650              SNA CLA                 /CARRIAGE RETURN ?
1107    1612  5221              JMP     LP12E           /YES - IGNORE IT
1108    1613  1534              TAD I   CTCHR           /GET OUTPUT CHARACTER
1109    1614  0132              AND     C77             /STRIP IT TO 6 BITS
1110    1615  6652              LCF                     /CLEAR FLAGS
1111    1616  6654              LLB                     /LOAD CHAR INTO PRINT BUFFER
1112    1617  6661              LSD                     /WAIT FOR PRINTER TO FINISH
1113    1620  5217              JMP     .-1
1114    1621  7300      LP12E,  CLA CLL
1115    1622  5600              JMP I   LP12            /EXIT
1116
1117
1118
1119
```

```
1120
1121    1623  0000    GFLD,   0                       /GET INST FIELD BITS IN AC9-11
1122    1624  7300            CLA CLL
1123    1625  6224            RIF                     /GET FIELD BITS
1124    1626  7010            RAR                     /MOVE TO AC9-11
1125    1627  7012            RTR
1126    1630  5623            JMP I   GFLD
1127
1128    1631  0000    CLRCT,  0                       /CLEAR ERROR COUNTERS
1129    1632  7300            CLA CLL
1130    1633  3100            DCA     ECNT1
1131    1634  3101            DCA     ECNT2
1132    1635  3102            DCA     ECNT3
1133    1636  5631            JMP I   CLRCT
1134
1135
1136
1137    1637  0000    EFLAG,  0                       /GET TOTAL NO. OF ERRORS
1138    1640  7300            CLA CLL
1139    1641  1100            TAD     ECNT1
1140    1642  1101            TAD     ECNT2
1141    1643  1102            TAD     ECNT3
1142    1644  7430            SZL                     /OVERFLOW?
1143    1645  7240            STA                     /YES-AC=7777
1144    1646  5637            JMP I   EFLAG
1145
1146    1647  0000    PFLD,   0                       /SET DATA FIELD=INST FIELD
1147    1650  3257            DCA     PF?             /SAVE AC
1148    1651  6224            RIF                     /GET INSTRUCTION FIELD BITS
1149    1652  1171            TAD     [CDF            /ADD CDF INSTRUCTION
1150    1653  3254            DCA     .+1             /STORE IN NEXT INSTRUCTION
1151    1654  6201            CDF                     /CDF=INST FIELD
1152    1655  1257            TAD     PF?             /RESTORE AC
1153    1656  5647            JMP I   PFLD            /RETURN
1154    1657  0000    PF?,    0
1155
1156    1660  0000    CLEAR,  0
1157    1661  4531            JMS I   [SAVE           /SAVE THE BINARY LOADER
1158    1662  4231            JMS     CLRCT           /ZERO ERROR COUNTERS
1159    1663  3103            DCA     PC?             /ZERO PASS COUNTER
1160    1664  1130            TAD     [7000           /GET A "NOP" INSTRUCTION
1161    1665  3261            DCA     CLEAR+1         /DELETE SAVE BIN CALL
1162    1666  7402            HLT                     /SET SR09-11 = UPPER LIMIT
1163    1667  7604            LAS                     /GET UPPER LIMIT
1164    1670  0144            AND     [7              /DELETE BITS 0-8
1165    1671  3104            DCA     LIMIT           /SET UPPER LIMIT
1166    1672  7402            HLT                     /SET SWITCHES FOR RUNNING
1167    1673  5660            JMP I   CLEAR           /RETURN
1168
```

```
1169    1674  7472    ETXT1,  TEXT    "<!> ERROR IN FPP-12 DATA WORD"
        1675  4005
        1676  2222
        1677  1722
        1700  4011
        1701  1640
        1702  0620
        1703  2055
        1704  6162
        1705  4004
        1706  0124
        1707  0140
        1710  2717
        1711  2204

1170    1712  0000            0
1171    1713  7472    ETXT2,  TEXT    "<!> ERROR IN BASE TABLE"
        1714  4005
        1715  2222
        1716  1722
        1717  4011
        1720  1640
        1721  0201
        1722  2305
        1723  4024
        1724  0102
        1725  1405

1172    1726  0000            0
1173    1727  7472    ETXT3,  TEXT    "<!> ERROR IN UNUSED MEMORY"
        1730  4005
        1731  2222
        1732  1722
        1733  4011
        1734  1640
        1735  2516
        1736  2523
        1737  0504
        1740  4015
        1741  0515
        1742  1722
        1743  3100

1174    1744  0000            0
1175    1745  7472    EDT,    7472                    /CRLF
1176    1746  0000    EFLDT,  0000                    /BAD DATA FIELD
1177    1747  0000    EADT,   0000                    /BAD DATA ADDRESS
1178    1750  0000            0000
1179    1751  4040            4040
1180    1752  4040            4040                    /SPACES
1181    1753  0000    EGDT,   0000                    /GOOD DATA
1182    1754  0000            0000
1183    1755  4040            4040                    /SPACES
1184    1756  0000    EBDT,   0000                    /BAD DATA
1185    1757  0000            0000
1186    1760  0000            0000                    /TERMINATOR
```

```
1187
1188    1761  7472    ETXT4,  TEXT    "<!ADDRESS   GOOD  BAD"
        1762  2104
        1763  0422
        1764  0523
        1765  2340
        1766  4040
        1767  0717
        1770  1704
        1771  4040
        1772  0201
        1773  0400
1189    1774  0000            0
1190    1775  7472    TXT1,   TEXT    "<!0000  ERRORS IN FIELD "
        1776  6060
        1777  6060
        2000  4040
        2001  0522
        2002  2217
        2003  2223
        2004  4011
        2005  1640
        2006  0611
        2007  0514
        2010  0440

1191    2011  0000    TXT1F,  0
1192    2012  0000            0
1193
1194    2013  7472    TXT2,   TEXT    "<!GOING TO FIELD  "
        2014  0717
        2015  1116
        2016  0740
        2017  2417
        2020  4006
        2021  1105
        2022  1404
        2023  4040

1195    2024  0000    TXT2F,  0
1196    2025  0000            0
1197
1198    2026  7472    LIN2,   7472            /RETURN + LINE FEED
1199    2027  7472            7472            /RETURN + LINE FEED
1200    2030  0000            0000
1201
1202    2031  7472    PET1,   TEXT    "<!0000 FPP DATA ERRORS"
        2032  6060
        2033  6060
        2034  4006
        2035  2020
        2036  4004
        2037  0124
        2040  0140
        2041  0522
```

```
                2042  2217
                2043  2223

1203            2044  0000          0
1204            2045  7472    PET2, TEXT    "<10000 BASE TABLE ERRORS"
                2046  6060
                2047  6060
                2050  4002
                2051  0123
                2052  0540
                2053  2401
                2054  0214
                2055  0540
                2056  0522
                2057  2217
                2060  2223

1205            2061  0000          0
1206            2062  7472    PET3, TEXT    "<10000 UNUSED MEMORY ERRORS"
                2063  6060
                2064  6060
                2065  4025
                2066  1625
                2067  2305
                2070  0440
                2071  1505
                2072  1517
                2073  2231
                2074  4005
                2075  2222
                2076  1722
                2077  2300
1207
1208            2100  7472    EPT,  TEXT    "<<END PASS  "
                2101  0516
                2102  0440
                2103  2001
                2104  2323
                2105  4040

1209            2106  0000    EPTV, 0
1210            2107  0000          0
1211            2110  0000          0
```

```
1212
1213                    /THIS ROUTINE SAVES THE LAST MEMORY PAGE
1214                    /IN FIELD 0 THE FIRST TIME THIS PROGRAM
1215                    /IS RAN AFTER LOADING,  THE LAST PAGE IS MOVE
1216                    /TO AN AREA WITHIN THE PROGRAM AND CARRIED WITH
1217                    /THE PROGRAM ANY TIME IT IS RELOCATED
1218
1219    2111  0000      SAVE,   0
1220    2112  7200              CLA
1221    2113  1147              TAD     [SBUF-1         /GET ADDRESS OF SAVE BUFFER
1222    2114  3010              DCA     10              /SAVE IN AUTO INDEX REG
1223    2115  1470              TAD I   CPNTR           /GET WORD FROM LAST PAGE
1224    2116  3410              DCA I   10              /STORE IN BUFFER
1225    2117  2070              ISZ     CPNTR           /INCREMENT PAGE ADDRESS
1226    2120  5315              JMP     .-3             /GET NEXT WORD
1227    2121  1150              TAD     [7600           /CLA INSTRUCTION
1228    2122  3020              DCA     20              /CLEAR CALL TO THIS ROUTINE
1229    2123  7604              LAS                     /GET SWITCH REGISTER
1230    2124  0144              AND     [7              /GET FIELD LIMIT
1231    2125  3104              DCA     LIMIT           /SET LIMIT
1232    2126  5711              JMP I   SAVE            /EXIT
1233
```

```
1234
1235    2127  0000    SBUF,   0
1236
1237          2200            PAGE
1238
1239    2200  0000            0
1240
1241          2400            PAGE
1242
1243          2375            *,+3
1244
1245    2375  0000            0
1246    2376  0000            0
1247    2377  0000            0
1248    2400  0000    END,    0
1249
1250
1251                          .
1252    0130  7000
1253    0131  2111
1254    0132  0077
1255    0133  7566
1256    0134  1536
1257    0135  7776
1258    0136  0040
1259    0137  4040
1260    0140  0116
1261    0141  7710
1262    0142  0115
1263    0143  6040
1264    0144  0007
1265    0145  0045
1266    0146  5400
1267    0147  2126
1268    0150  7600
1269    0151  0665
1270    0152  6202
1271    0153  0400
1272    0154  0020
1273    0155  7770
1274    0156  0100
1275    0157  7771
1276    0160  7775
1277    0161  0046
1278    0162  0044
1279    0163  0052
1280    0164  1030
1281    0165  0030
1282    0166  0010
1283    0167  2400
1284    0170  0070
1285    0171  6201
1286    0172  2375
1287    0173  0200
1288    0174  0745
```

```
1289    0175  1077
1290    0176  1116
1291    0177  1660
```

```
0000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0100  11111111  11111111  11000000  11111111  11111111  11111111  11111111  11111111

0200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0300  11111111  11111111  11111111  11111111  11111111  11111110  00000000  00001111

0400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0500  11111111  11111111  11111111  11111111  11111111  11111100  00000000  01111111

0600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0700  11111111  11111111  11111111  11111111  11111111  11111111  11110011  11111111


1000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1100  11111111  11111111  11111111  11111111  11111111  11111111  11000111  11111111

1200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1300  11111111  11111111  11111111  11111111  10000000  00000000  00000000  00111111

1400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1500  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111101

1600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1700  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111


2000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2100  11111111  11111111  11111111  00000000  00000000  00000000  00000000  00000000

2200  10000000  00000000  00000000  00000000  00000000  00000000  00000000  00000000
2300  00000000  00000000  00000000  00000000  00000000  00000000  00000000  00000111

2400  10000000  00000000  00000000  00000000  00000000  00000000  00000000  00000000
2500  00000000  00000000  00000000  00000000  00000000  00000000  00000000  00000000

2600
2700


3000
3100

3200
3300

3400
3500

3600
3700
```

4000
4100

4200
4300

4400
4500

4600
4700

5000
5100

5200
5300

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

AFLD	0042	EP1	0677	LSD	6661
APNTR	0043	EP2	0710	LSE	6651
APT	0052	EP3	0720	LSF	6661
ASC	1400	EP4	0730	LSR	6663
ASC1	1436	EP5	0736	MODE	0020
ASC2	1437	EPASS	0663	MOVEP	1016
ASC3	1440	EPNTR	0072	NFLD	1017
ASC4	1441	EPT	2100	P0	0055
ASC5	1442	EPTV	2106	PCT	0103
ASC77	1443	ERET	1313	PDPT	0062
ASCB	1423	ERLOOP	1326	PET1	2031
BASE	0041	ERROR	1200	PET2	2045
BTCK	1043	ERROR1	1026	PET3	2062
CEND	0453	ERROR2	1054	PFLD	1647
CFLD	0216	ERROR3	1131	PFT	1657
CFLG	6661	ETCNT	0077	PPNTR	0071
CKFLD	0627	ETXT	1265	PRT	1537
CKMEM	0503	ETXT1	1674	PRT8	1555
CKREL	0645	ETXT2	1713	RE22	1011
CLCOR	0206	ETXT3	1727	RELO	1000
CLEAR	1660	ETXT4	1741	RESCDF	0466
CLRCT	1631	FCLA	0002	RESTOR	0745
CMFLD	0520	FLDA	0000	RFLD	0415
CMPEND	0600	FPAUSE	0001	RUN	0317
CMPR	0400	FPC	0053	SAVE	2111
CPNTR	0070	FPCOM	6553	SBUF	2127
CRLF	0115	FPICL	6532	SE22	1116
CATA	0040	FPP	0030	SE24	1077
CATA1	0044	FPPZ	0614	SFLG	6642
CATA2	0045	FPRST	6556	START	0200
CATA3	0046	FPST	6555	STFPP	0251
CATA4	0047	FRJA	0036	T1	0106
DFLD	0413	FSTA	6000	T100	0113
EADT	1747	FSTEP	0332	T212	0110
EBD	1274	FSW	0105	T215	0107
EBDT	1756	GFLD	1623	T240	0114
EBUF	0116	INCLA	0763	T77	0111
ECNT	0076	INCLAP	0341	TADDR	0074
ECNT1	0100	JA	1030	TCHR	1536
ECNT2	0101	K2	0027	TCRLF	1451
ECNT3	0102	K3	0356	TM40	0112
ECONV	1252	LAP	0121	TOUT	1520
EDT	1745	LCF	6652	TPNTR	0073
EFLAG	1637	LIM7	0455	TRET	1471
EFLD	1231	LIMIT	0104	TTY	1505
EFLDT	1746	LIN2	2026	TXT1	1775
EGD	1271	LLB	6654	TXT1F	2011
EGDT	1753	LP08	1563	TXT2	2013
END	2400	LP12	1600	TXT2F	2024
END12	1072	LP12E	1621	TYPA	1473
ENDET	1320	LPC	6666	TYPE	1444
ENDREL	0656	LPR	6664	WCNT	0075

ERRORS DETECTED: 0

LINKS GENERATED: 73

RUN-TIME: 11 SECONDS

2K CORE USED

```
L0131   1157   1253#
L0132   1109   1254#
L0133   1102   1255#
L0134   1101   1108   1256#
L0135    917   1257#
L0136    907   1258#
L0137    895    897    899   1259#
L0140    873   1260#
L0141    862   1261#
L0142    849   1262#
L0143    842   1263#
L0144    762   1164   1230   1264#
L0145    713   1265#
L0146    666   1266#
L0147    620   1221   1267#
L0150    618   1227   1268#
L0151    592   1269#
L0152    537   1270#
L0153    529   1271#
L0154    430   1272#
L0155    419    433   1273#
L0156    418    603    864   1274#
L0157    397   1275#
L0160    352    508    847    879   1105   1276#
L0161    350   1277#
L0162    348   1278#
L0163    286   1279#
L0164    281   1280#
L0165    267   1281#
L0166    247    386    469    519    662    723   1064   1282#
L0167    244    303    438    466   1283#
L0170    230    376    416    452    520    534    839    861   1284#
L0171    221    308    355    402    420    440    506    521    663    716    865   1149   1285#
L0172    219   1286#
L0173    167    555   1287#
L0174    116   1288#
L0175    113   1289#
L0176    110    759   1290#
L0177    107   1291#
L0374    316    336#
L0375    309    337#
L0376    217    240    280    283    299    338#
L0377    216    228    256    339#
L0571    450    472#
L0572    447    473#
L0573    436    462    474#
L0574    435    475#
L0575    432    460    476#
L0576    368    477#
L0577    363    478#
L0766    606    634#
L0767    599    635#
L0770    566    578    586    594    636#
```

L0771	561	569	581	589	597	637#
L0772	558	638#				
L0773	533	639#				
L0774	532	640#				
L0775	495	641#				
L0776	488	642#				
L0777	486	622	643#			
L1165	799	802#				
L1166	793	803#				
L1167	784	804#				
L1170	726	805#				
L1171	725	806#				
L1172	693	721	786	800	807#	
L1173	690	718	794	808#		
L1174	689	809#				
L1175	688	715	810#			
L1176	656	811#				
L1177	655	671	777	812#		
L1372	900	932#				
L1373	898	933#				
L1374	881	883	893	934#		
L1375	853	935#				
L1376	844	885	890	936#		
L1377	843	896	937#			
L1577	1049	1093#				