```
                .TITLE FPU11    EDIT 002 FOR XVM                770214
;
;
;                       FIRST PRINTING, AUGUST 1974
;
; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO
; CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED
; AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.
; DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPON-
; SIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS
; DOCUMENT.
;
; THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FUR-
; NISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON
; A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH
; INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR
; USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PRO-
; VIDED IN WRITING BY DIGITAL.
;
; DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY
; FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIP-
; MENT THAT IS NOT SUPPLIED BY DIGITAL.
;
; COPYRIGHT (C) 1974, BY DIGITAL EQUIPMENT CORPORATION
;
;
        .EJECT
```

```
                        /COPYRIGHT 1974,  DIGITAL EQUIPMENT CORP., OSLO, NORWAY.
                        /DAVE STOKES DEC SWS/OSLO/NORWAY
                        /FPU.--- EQUIVALENT OF PPA. PUNCH HANDLER
                        /CALLING SEQUENCE:
                        /       CAL + .DAT SLOT (9-17)
                        /       FUNCTION
                        /       N ARGS, WHERE N IS A FUNCTION OF "FUNCTION"
                        /       NORMAL RETURN
                        /ASSEMBLY PARAMETERS:
                        /NOSPL=0 FOR NON SPOOLED TASK VERSION
                        /
                        /       EDIT #2 FOR XVM-DOS MADE BY H BERGKVIST PEAB 770214
                        /
                        /
000003 A                APILVL=3                /LOWEST UNICHANNEL LEVEL
000070 A                APISLT=70               /ARBITARY UNUSED SLOT #
                        /
706161 A                FPSF=APILVL*20+706101       /SKIP FLAG FPSF=706161
706001 A                SIOA=706001                    /SKIP ON DATA ACCEPTED BY THE PDP11
706006 A                LIOR=706006                    /CLEAR "DONE" FLAG AND LOAD REG FOR
                                                       /     THE PDP11.
706164 A                CAPI=APILVL*20+706104          /CLEAR FLAG
                        /
000100 A                .SCOM=100
000003 A                .MED=3
000137 A                EXERRS=.SCOM+37
                                .IFUND NOSPL
000006 A                DEVCOD=6                /CODE FOR FP DRIVER IN PIREX
                                .ENDC
                                .IFDEF NOSPL
                        DEVCOD=206              /SAME DRIVER, DISABLE SPOOLING!
                                .ENDC
                                .GLOBL PPF.
```

```
                                        .TITLE [illegible]L ENTRANCE
00000 R 040417 R    PPF.[illegible]     DAC     FPCALP          /SAVE CAL POINTER.
00001 R 040420 R                        DAC     FPARGP          /AND ARGUMENT POINTER.
00002 R 440420 R                        ISZ     FPARGP          /POINTS TO WORD 2 - FUNCTION CODE.
00003 R 200451 R                      LAC DSTEST
00004 R 740200 A                    SZA
00005 R 600011 R                    JMP NEW
00006 R 200421 R                    LAC FPUND
00007 R 740200 A                    SZA
00010 R 600006 R                    JMP .-2
                                    /
                                    /FIRST TIME THROUGH SET UP THE TCB ETC
                                    /
00011 R 600423 R    NEW     JMP     INIT    /FIRST TIME THRU DO SETUP CAL
                                    /       /AND SET-UP TCB AND BUFFER. OVERWRITE
                                    /       /JUMP WITH NO-OP
                                    /
00012 R 220420 R                    LAC*    FPARGP
00013 R 440420 R                    ISZ     FPARGP          /POINTS TO WORD 3 - BUFFER ADDRESS.
00014 R 500536 R                    AND     (17777          /STRIP OFF UNIT NUMBER.
00015 R 340537 R                    TAD     (JMP LTABL-1    /DISPATCH TO PROCESS FUNCTION.
00016 R 040017 R                    DAC     .+1
00017 R 740040 A                    XX
00020 R 600110 R    LTABL   JMP     FPIN            /1 - .INIT
00021 R 741000 A                    SKP             /2 - .FSTAT,.RENAM,.DLETE - IGNORE
00022 R 600032 R                    JMP     FPER06  /3 - .SEEK - ERROR
00023 R 440420 R                    ISZ     FPARGP  /4 - .ENTER - IGNORE
00024 R 600116 R                    JMP     FPNEXT  /5 - .CLEAR - IGNORE
00025 R 600347 R                    JMP     FPCLOS  /6 - .CLOSE
00026 R 600116 R                    JMP     FPNEXT  /7 - .MTAPE - IGNORE
00027 R 600032 R                    JMP     FPER06  /10 - .READ - ERROR.
00030 R 600120 R                    JMP     FPWRIT  /11 - .WRITE
00031 R 600366 R                    JMP     FPWAIT  /12 - .WAIT OR .WAITR
00032 R 760006 A    FPER06  LAW     6               /ILLEGAL HANDLER FUNCTION.
00033 R 600100 R                    JMP SETERR
00034 R 760012 A    IOPS12  LAW 12          /UC15 WENT AWAY         EDIT #2
00035 R 600100 R                    JMP SETERR      /               EDIT #2
00036 R 760055 A    IOPS55  LAW 55          /NO TCB AREA FOR US     EDIT #2
00037 R 600100 R                    JMP SETERR      /               EDIT #2
00040 R 760067 A    IOPS67  LAW 67          /ATTEMPT REF. >32K      EDIT #2
00041 R 600100 R                    JMP SETERR      /               EDIT #2
```

```
                                        .TITLE INTERRUPT SERVICE
                                /
                                /FPU. INTERRUPT SERVICE
00042 R 600052 R    FPINT   JMP     FPPIC   /PIC ENTRY, JUMP TO CODE
00043 R 040425 R            DAC     FPAC    /SAVE INTERRUPTED AC
00044 R 200042 R            LAC     FPINT   /GET INTERRUPTED PC
00045 R 040430 R            DAC     FPOUT   /SAVE FOR COMMON EXIT
00046 R 200540 R            LAC     (JMP FPPIC /RESTORE PIC ENTRY
00047 R 040042 R            DAC     FPINT
00050 R 200541 R            LAC     (NOP    /WE DON'T NEED ION IN COMMON EXIT
00051 R 600056 R            JMP     FPICM   /JOIN COMMON CODE
                                /
00052 R 040425 R    FPPIC   DAC     FPAC    /PIC CODE, SAV AC
00053 R 220542 R            LAC*    (0      /GET INTERRUPTED PC
00054 R 040430 R            DAC     FPOUT   /SAVE
00055 R 200543 R            LAC     (ION    /NEED INTERRUPT ON INST. IN COMMON CODE
00056 R 040066 R    FPICM   DAC     FPISW
00057 R 706164 A            CAPI            /CLEAR FLAG, NOW IN COMMON CODE
00060 R 220435 R            LAC*    FPEV    /EVENT VARIABLE FROM PIREX
00061 R 742010 A            RTL             /PDP-11 (MINUS) BIT TO OUR AC0
00062 R 743120 A            SPA!RTR         /+ IS OK
00063 R 600071 R            JMP     FPIERR  /ERROR, GO LOOK
00064 R 140421 R    FPIRT   DZM     FPUND   /CLEAR UNDERWAY FLAG
00065 R 200425 R    FPIRT1  LAC     FPAC    /RESTORE AC
00066 R 740040 A    FPISW   HLT             /ION OR NOP
00067 R 703344 A            DBR
00070 R 620430 R            JMP*    FPOUT
                                /
                                /
00071 R 500544 R    FPIERR  AND     (177777 /KEEP REAL 16 BITS FROM PDP-11
00072 R 540545 R            SAD     (177001 /CODE FROM OUT OF NODES IN PIREX
00073 R 600076 R            JMP     RETRY   /JUST TRY AGAIN, LEAVING FPUND SET
00074 R 340546 R            TAD     (600000 /MAKE - NUMBER FOR IOPS
00075 R 600100 R            JMP     SETERR  /TREAT AS REGULAR IOPS ERROR
                                /                       /NOTE THAT THIS SHOULDN'T HAPPER.
                                /
                                /
00076 R 100407 R    RETRY JMS FPSET
00077 R 600065 R            JMP     FPIRT1  /EXIT FROM INTERRUPT
                                /
                                /
```

```
                                    .TITLE ERROR ROUTINE
                                /
00100 R 040107 R    SETERR  DAC ERRNUM
00101 R 740000 A    ERLOOP  NOP                 /'JMP FPTRY' IF IOPS 4 ERROR.
00102 R 200107 R            LAC ERRNUM
00103 R 120547 R    EROUT   JMS* (EXERRS
00104 R 600101 R            JMP ERLOOP
00105 R 777777 A            LAW -1
00106 R 062025 A            .SIXBT 'FPU'
00107 R 000000 A    ERRNUM  0                   /HOLDS ERROR NUMBER FOR REPEAT.
```

```
                                .TITLE .INIT FUNCTION
                        /
                        /.INIT
                        /
00110 R 440420 R    FPIN    ISZ     FPARGP
00111 R 200422 R          LAC BUFSIZ              /52 WORDS FOR STANDARD
00112 R 060420 R            DAC* FPARGP             /RETURN TO USER.
00113 R 440420 R            ISZ FPARGP              /NOW POINTS TO RETURN.
00114 R 100467 R       JMS FPHEAD
00115 R 600217 R       JMP MAIN4+2
                        /
                        /NORMAL CAL EXIT
                        /
00116 R 703344 A    FPNEXT  DBR
00117 R 620420 R            JMP*    FPARGP
```

```
                                .TITLE WRITE FUNCTION
                               /
                               /.WRITE
                               /
00120 R 220417 R    FPWRIT LAC* FPCALP      /GET DATA MODE FROM USERS CAL.
00121 R 500550 R             AND     (1000  /MAKE SKP-NOP IN MIX
00122 R 240551 R            XOR (SKP     /SKP=ASCII  NOP=IMAGE
00123 R 040440 R             DAC     MIX
00124 R 220420 R             LAC*    FPARGP          /USER BUFFER ADDRESS.
00125 R 440420 R             ISZ FPARGP      /NOW POINTS TO WORD COUNT
00126 R 040441 R             DAC     TCHAR   /SAVE POINTER TO BUFFER HEADER
00127 R 723002 A             AAC     2       /MAKE X12 POINT TO DATA NOT HEADER
00130 R 040431 R             DAC     X12     /GETTER POINTER
00131 R 500552 R             AND (700000     /BUFFER >32K ?  /EDIT #2
00132 R 740200 A             SZA             /               /EDIT #2
00133 R 600040 R             JMP IOPS67      /YES. FATAL ERROR       /EDIT #2
00134 R 140443 R            DZM FPLNST   /START OF NEW BUFFER INDICATOR AND CHAR COUNT
00135 R 140447 R       DZM FPBYTE   /CHAR-BYTE INDICATOR
00136 R 140450 R       DZM FPWORD   /CHAR-WORD INDICATOR
00137 R 140455 R       DZM EOLINE   /END OF LINE INDICATOR
                               /
                               /  SET UP LIMIT OF INPUT BUFFER SIZE TO PREVENT DATA OVERRUN
                               /  FOR BOTH IOPS ASCII AND IMAGE ASCII
                               /
00140 R 777000 A             LAW     17000   /GET PAIR COUNT FROM LEFT HALF
00141 R 520441 R             AND*    TCHAR
00142 R 742030 A             SWHA            /BRING TO RIGHT. PAIR COUNT INCLUDES HEADER
                               /             /PAIR COUNT. WE ISZ BEFORE LOOP SO THAT'S
                               /             /OK. IOPS NOW SET XCPT CMA!IAC
00143 R 400440 R             XCT     MIX     /SKP IF ASCII, NOP IF IMAGE
00144 R 741000 A           SKP     /FOR IMAGE MODE
00145 R 741031 A             SKP!CMA!IAC     /IOPS COMPLEMENTED TO CORRECT VALUE
00146 R 100530 R           JMS MAKEIM    /MAKE COUNT FOR IMAGE
                               /             /TWO WORDS FOR HEADER. WE ISZ BEFORE LOOP.
00147 R 040423 R             DAC     INIT    /INTO CONTROLLER, BOTH MODES
00150 R 440420 R             ISZ     FPARGP  /MOVE ARG POINTER TO EXIT
00151 R 200317 R             LAC     GETIN   /INIT. CHAR GETTER
00152 R 040312 R             DAC     GETSW
                               /
                               /  MAIN LOOP TO TRANSFER CHAR'S TO HANDLER BUFFER
                               /
00153 R 100300 R    MAIN    JMS     GETCH   /CHARACTER GETTER, LEAVES IT IN AC
00154 R 040441 R        DAC TCHAR
00155 R 400440 R        XCT MIX   /ASCII OR IMAGE
00156 R 600211 R        JMP MAIN1+1   /IMAGE MODE
00157 R 200443 R       LAC FPLNST   /ASCII MODE
00160 R 740200 A         SZA
00161 R 600173 R       JMP MAIN5
00162 R 200441 R       LAC TCHAR
00163 R 540553 R       SAD (12
00164 R 600221 R       JMP LFCODE
00165 R 540554 R       SAD (13
```

```
220  00166 R 600224 R        JMP VTCODE
221  00167 R 540555 R        SAD (14
222  00170 R 600235 R        JMP FFCODE
223  00171 R 200553 R        LAC (12
224  00172 R 100253 R        JMS PUTCH
225  00173 R 200441 R    MAIN5 LAC TCHAR
226  00174 R 540556 R        SAD (15   /IS IT CR
227  00175 R 440455 R        ISZ EOLINE  /YES SO SHOW IT FOR LATER
228  00176 R 540557 R        SAD (175   /IS IT ALT MODE
229  00177 R 440455 R        ISZ EOLINE  /YES
230  00200 R 540553 R        SAD (12
231  00201 R 600153 R        JMP MAIN
232  00202 R 540554 R        SAD (13
233  00203 R 600153 R        JMP MAIN
234  00204 R 540555 R        SAD (14
235  00205 R 600153 R        JMP MAIN
236  00206 R 540560 R        SAD (11
237  00207 R 600246 R        JMP TABCOD
238  00210 R 100504 R    MAIN1 JMS PARITY
239  00211 R 100253 R        JMS PUTCH
240  00212 R 200455 R        LAC EOLINE    /IS IT END OF LINE
241  00213 R 741200 A        SNA            /YES
242  00214 R 600153 R        JMP MAIN
243  00215 R 200443 R    MAIN4   LAC FPLNST   /COUNT IN FPLNST
244  00216 R 060442 R        DAC* FPBUF     /TO TCBBUFFER
245  00217 R 100407 R        JMS FPSET
246  00220 R 600116 R         JMP FPNEXT    /AND NORMAL CAL EXIT
247                          /
248                          /
249  00221 R 200553 R    LFCODE LAC (12
250  00222 R 100253 R        JMS PUTCH
251  00223 R 600153 R        JMP MAIN
252  00224 R 200561 R    VTCODE LAC (213
253  00225 R 100253 R        JMS PUTCH
254  00226 R 777774 A        LAW -4
255  00227 R 040452 R        DAC PARSAV
256  00230 R 200562 R        LAC (377
257  00231 R 100253 R        JMS PUTCH
258  00232 R 440452 R        ISZ PARSAV
259  00233 R 600230 R        JMP .-3
260  00234 R 600153 R        JMP MAIN
261  00235 R 200555 R    FFCODE LAC (14
262  00236 R 100253 R        JMS PUTCH
263  00237 R 777740 A        LAW -40
264  00240 R 040452 R        DAC PARSAV
265  00241 R 750000 A        CLA
266  00242 R 100253 R        JMS PUTCH
267  00243 R 440452 R        ISZ PARSAV
268  00244 R 600241 R        JMP .-3
269  00245 R 600153 R        JMP MAIN
270  00246 R 200560 R    TABCOD LAC (11
271  00247 R 100253 R        JMS PUTCH
```

```
00250 R 200562 R        LAC (377
00251 R 100253 R        JMS PUTCH
00252 R 600153 R        JMP MAIN
                        /
                        /
                        /
00253 R 000000 A    PUTCH 0
00254 R 440443 R        ISZ FPLNST
00255 R 040446 R        DAC PUTSAV
00256 R 200447 R        LAC FPBYTE
00257 R 240563 R        XOR (1
00260 R 040447 R        DAC FPBYTE
00261 R 740200 A        SZA
00262 R 600267 R        JMP PUTBOT
00263 R 200446 R        LAC PUTSAV
00264 R 742030 A        SWHA
00265 R 744020 A        RCR
00266 R 600275 R        JMP PUTTOP
00267 R 200436 R    PUTBOT LAC FPBUFD
00270 R 340450 R        TAD FPWORD
00271 R 040445 R        DAC PUTADD
00272 R 440450 R         ISZ FPWORD
00273 R 160445 R        DZM* PUTADD
00274 R 200446 R        LAC PUTSAV
00275 R 260445 R    PUTTOP XOR* PUTADD
00276 R 060445 R        DAC* PUTADD
00277 R 620253 R        JMP* PUTCH
                        /       CHARACTER UNPACKING ROUTINE
                        /
                        /
                        /  THIS ROUTINE 'OWNS' THE MQ
                        /
                        /
                        /  CHARACTERS ARE OBTAINED FROM X12 POINTER. EACH CHAR
                        /  IS RETURNED RIGHT JUSTIFIED IN THE AC
                        /  INIT HAS A MINUS COUNT OF THE WORDS TO BE OBTAINED
                        /  FROM THE INPUT POINTER X12
                        /
00300 R 000000 A    GETCH   0
00301 R 400440 R            XCT     MIX     /SKIP IF IT IS ASCII,NOP IF IMAGE
00302 R 741000 A            SKP
00303 R 620312 R            JMP*    GETSW   /GETSW IS POINTER TO CORRECT ACTION ON ONTHE
                        /                   /CORRECT ONE OF THE FIVE POSSIBLE CHAR'S
                        /
                        /  NOW DO IMAGE MODE
                        /
00304 R 440423 R            ISZ     INIT
00305 R 741000 A            SKP             /SKP ON NOT THRU YET
00306 R 600215 R            JMP     MAIN4   /DONE
00307 R 220431 R            LAC*    X12
00310 R 440431 R            ISZ     X12
00311 R 600313 R            JMP     GETCM   /FINISH UP IN COMMON
```

```
                                    /
00312 R 000000 A    GETSW   0              /POINTER TO CORRECT ACTION. INIT'ED FROM GETIN
                                    /                  /FILLED BY JMS GETSW AFTER EACH CHAR
00313 R 400440 R    GETCM   XCT     MIX    /COMMON FINISH UP, STRIP XTRA BITS
00314 R 741000 A          SKP     /IMAGE MODE DOES THIS
00315 R 500564 R            AND (177     /ASCII DOES THIS
00316 R 620300 R             JMP*    GETCH  /OUT
                                    /
00317 R 000321 R    GETIN   GET1           /INIT GETSW TO POINT TO FIRST CHAR ACTION
                                    /
                                    /  INDIVIDULA CHARACTER ACTION
                                    /
00320 R 100312 R    GETQ    JMS     GETSW  /AFTER 5TH CHAR, POINT BACK TO FIRST
                                    /
00321 R 440423 R    GET1    ISZ     INIT   /OUT OF PAIRS?
00322 R 600324 R            JMP     .+2    /CONTINUE IF OK
00323 R 600215 R            JMP     MAIN4
00324 R 220431 R            LAC*    X12    /FIRST WORD OF PAIR
00325 R 440431 R            ISZ     X12
00326 R 652000 A            LMQ            /INTO MQ FOR SHIFTING
00327 R 640607 A            LLS     7
00330 R 100312 R            JMS     GETSW  /DONE, LEAVE POINTER FOR SECOND CHAR
00331 R 640607 A    GET2    LLS     7      /SECOND CHAR
00332 R 100312 R            JMS     GETSW  /LEAVING POINTER FOR THIRD
00333 R 640604 A    GET3    LLS     4      /THE HALF-AND-HALF CHAR
00334 R 040312 R            DAC     GETSW  /VERY TEMPORARY
00335 R 220431 R            LAC*    X12    /CAN'T END IN MIDDLE OF PAIR
00336 R 440431 R            ISZ     X12
00337 R 652000 A            LMQ            /SECOND WORD TO SHIFTER
00340 R 200312 R            LAC     GETSW  /BRING BACK FIRST
00341 R 640603 A            LLS     3      /COMPLETE CHAR
00342 R 100312 R            JMS     GETSW  /LEAVING POINTER TO FOURTH ACTION
00343 R 640607 A    GET4    LLS     7
00344 R 100312 R            JMS     GETSW  /LEAVING FOR 5
00345 R 640607 A    GET5    LLS     7
00346 R 600320 R            JMP     GETQ   /BACK TO TOP FOR POINTER TO 1
                                    /
                                    /
                                    /
                                    /
                                    /
```

```
                                    .TITLE .CLOSE FUNCTION
                                /
                                /
                                /.CLOSE
                                /
00347 R 440353 R    FPCLOS  ISZ FPCLSW      /777777 IN AC IF NOT THRU CLOSE CODE
00350 R 600354 R            JMP     FPCLDN          /DONE.
00351 R 703344 A    FPCALX  DBR
00352 R 620417 R            JMP*    FPCALP          /HANG ON CAL.
00353 R 777777 A    FPCLSW  777777                  /-1 = .CLOSE NOT DONE.
00354 R 777777 A    FPCLDN  LAW -1
00355 R 040353 R            DAC     FPCLSW          /INITIALIZE .CLOSE INDICATOR
00356 R 100467 R        JMS FPHEAD
                        .IFDEF NOSPL
                            JMP ATLAST
                        .ENDC
00357 R 100407 R            JMS FPSET
00360 R 200421 R            LAC FPUND
00361 R 740200 A            SZA
00362 R 600360 R            JMP .-2
00363 R 200565 R            LAC (177404
00364 R 060442 R            DAC* FPBUF
00365 R 600217 R    ATLAST JMP MAIN4+2
```

```
                                .TITLE .WAIT FUNCTION
                        /
                        /.WAIT OR .WAITR
                        /
00366 R 220417 R    FPWAIT  LAC* FPCALP
00367 R 500550 R            AND     (1000
00370 R 741200 A            SNA                     /BIT 8 = 1 FOR .WAITR
00371 R 600406 R            JMP     FPWAT1          /.WAIT - GO HANG ON CAL.
00372 R 220420 R            LAC* FPARGP     /CHECK 32K LIMIT                EDIT #2
00373 R 500552 R            AND (700000     /                               EDIT #2
00374 R 740200 A            SZA             /SKIP IF OK                     EDIT #2
00375 R 600040 R            JMP IOPS67      /                               EDIT #2
00376 R 200552 R            LAC     (700000         /LINK, ETC.
00377 R 500417 R            AND     FPCALP
00400 R 040417 R            DAC     FPCALP
00401 R 220420 R            LAC*    FPARGP          /15-BIT BUSY ADDRESS.
00402 R 500566 R            AND     (77777
00403 R 240417 R            XOR     FPCALP
00404 R 040417 R            DAC     FPCALP
00405 R 440420 R            ISZ     FPARGP
00406 R 600116 R    FPWAT1 JMP      FPNEXT          /OK - RETURN.
                        /
                        /
                        / SETUP AND OUTPUT TO PRINTER.
                        /
00407 R 000000 A    FPSET   0
00410 R 160435 R       DZM* FPEV
00411 R 200424 R            LAC     FPTCB   /SEND TCB POINTER TO PDP-11
00412 R 706001 A            SIOA                    /MAKE SURE ITS ABLE TO GET IT
00413 R 600412 R            JMP     .-1             /NOTE THAT THIS IS PROTECTED SINCE
                                                    /    THE LIOR WILL BE ISSUED DIRECTLY
                                                    /    AFTER THE SIOA (FREE INSTRUCTION).
00414 R 706006 A            LIOR
00415 R 040421 R            DAC FPUND               /SET I/O BUSY FLAG.
00416 R 620407 R            JMP* FPSET
```

```
                                .TITLE INITIALIZATION CODE AND TEMPORARIES
                        /
00417 R 000000 A    FPCALP  0               /POINTER TO CAL ADDR
00420 R 000000 A    FPARGP  0               /POINTER ARGUMENTS OF CAL
00421 R 777772 A    FPUND   -6              /0=FREE,+=BUSY,-=ERROR
00422 R 000064 A    BUFSIZ   64           /STANDARD BUFFER SIZE IS 52 DECIMAL
                        /                     /COUNTS UP TO INITAL 0 BELOW
                        /
00423 R 200541 R    INIT    LAC     (NOP    /WRITE OVER JUMP TO HERE
00424 R 040011 R    FPTCB   DAC     NEW     /PREVENT RE-ENTRY
                        /
                        /  NOW SET UP POINTERS TO BUFFER AND TCB LOC'S
                        /
00425 R 220567 R    FPAC    LAC*    (.SCOM+100  /POINTER TO TABLE OF POINTERS
00426 R 741200 A            SNA             /IF ISN'T ONE, UC15 GONE        EDIT #2
00427 R 600034 R            JMP IOPS12      /                               EDIT #2
00430 R 723006 A    FPOUT   AAC 6           /OUR POINTER IN TABLE +6        EDIT #2
00431 R 040423 R    X12     DAC     INIT
00432 R 220423 R    PUTP    LAC*    INIT    /POINTER TO TCB
00433 R 741300 A            SNA!SPA         /IS THERE A TCB FOR US          EDIT #2
00434 R 600036 R            JMP IOPS55      /                               EDIT #2
00435 R 040424 R    FPEV  DAC FPTCB   /POINTER TO TCB
00436 R 040423 R    FPBUFD  DAC INIT     /POINTER TO INTERNAL BUFFER
00437 R 723002 A    TABC   AAC 2
00440 R 040435 R    MIX   DAC FPEV
00441 R 723002 A    TCHAR   AAC 2
00442 R 040437 R    FPBUF   DAC TABC
00443 R 723005 A    FPLNST   AAC 5
00444 R 040436 R    FPLNDN   DAC FPBUFD
                        /
                        /  MAKE TCB
                        /
00445 R 200570 R    PUTADD   LAC (APISLT*400+APILVL
00446 R 060423 R    PUTSAV   DAC* INIT
00447 R 440423 R    FPBYTE ISZ INIT
00450 R 200571 R    FPWORD LAC (DEVCOD  /PIREX CODE FOR FP DRIVER
00451 R 060423 R    DSTEST DAC*     INIT
00452 R 440423 R    PARSAV ISZ      INIT    /ZERO THRU FIRST BUFFER LOC
00453 R 160423 R    PARCNT DZM*     INIT
00454 R 440421 R    PARIND ISZ      FPUND
00455 R 600452 R    EOLINE   JMP .-3
00456 R 200423 R       LAC INIT       /ADDRESS OF PIREX BUFFER START
00457 R 060437 R        DAC* TABC      /TO THE TCB WORD
00460 R 040442 R        DAC FPBUF      /AND A POINTER FOR US LADS
00461 R 000070 A            CAL     APISLT  /ISSUE SETUP CAL TO ESTABLISH INTERRUPTS
00462 R 000016 A            16
00463 R 706161 A            FPSF
00464 R 000042 R            FPINT
00465 R 140451 R            DZM DSTEST
00466 R 600011 R            JMP     NEW     /WHEW, DONE
00467 R 000000 A    FPHEAD 0
00470 R 200572 R       LAC (212
```

```
00471 R 060442 R        DAC* FPBUF
00472 R 744020 A        RCR
00473 R 740031 A        TCA
00474 R 040446 R        DAC PUTSAV
00475 R 200436 R        LAC FPBUFD
00476 R 040445 R        DAC PUTADD
00477 R 160445 R        DZM* PUTADD
00500 R 440445 R        ISZ PUTADD
00501 R 440446 R        ISZ PUTSAV
00502 R 600477 R        JMP .-3
00503 R 620467 R        JMP* FPHEAD
                      /
00504 R 740040 A    PARITY XX
00505 R 040452 R        DAC PARSAV
00506 R 777771 A        LAW -7
00507 R 040453 R        DAC PARCNT
00510 R 140454 R        DZM PARIND
00511 R 200452 R        LAC PARSAV
00512 R 744020 A    PARLOP RCR
00513 R 741400 A        SZL
00514 R 440454 R        ISZ PARIND
00515 R 440453 R        ISZ PARCNT
00516 R 600512 R        JMP PARLOP
00517 R 200454 R        LAC PARIND
00520 R 500563 R        AND (1
00521 R 741200 A        SNA
00522 R 600526 R        JMP PAREND
00523 R 200452 R        LAC PARSAV
00524 R 240567 R        XOR (200
00525 R 741000 A        SKP
00526 R 200452 R    PAREND LAC PARSAV
00527 R 620504 R        JMP* PARITY
00530 R 000000 A    MAKEIM  0
00531 R 723777 A            AAC -1      /SUBTRACT HEADER WORD PAIR COUNT
00532 R 744010 A            RCL         /MAKE TRUE COUNT
00533 R 740030 A            IAC         /ADD ON 1 TO COMPENSATE FOR ISZ
00534 R 740031 A            TCA         /AND NEGATE IT
00535 R 620530 R            JMP* MAKEIM
        000000 A              .END
00536 R 017777 A *L
00537 R 600017 R *L
00540 R 600052 R *L
00541 R 740000 A *L
00542 R 000000 A *L
00543 R 700042 A *L
00544 R 177777 A *L
00545 R 177001 A *L
00546 R 600000 A *L
00547 R 000137 A *L
00550 R 001000 A *L
00551 R 741000 A *L
00552 R 700000 A *L
```

```
00553 R 000012 A *L
00554 R 000013 A *L
00555 R 000014 A *L
00556 R 000015 A *L
00557 R 000175 A *L
00560 R 000011 A *L
00561 R 000213 A *L
00562 R 000377 A *L
00563 R 000001 A *L
00564 R 000177 A *L
00565 R 177404 A *L
00566 R 077777 A *L
00567 R 000200 A *L
00570 R 034003 A *L
00571 R 000006 A *L
00572 R 000212 A *L
    SIZE=00573     NO ERROR LINES
```

```
.GV/BUF  MACRO
.INIT    MACRO   153
.MED     000003   52*
.MTAPE   MACRO
.MTRAN   MACRO
.OV/RLA  MACRO
.PUT     MACRO
.RAND    MACRO
.READ    MACRO
.RENAM   MACRO
.RTRAN   MACRO
.SCOM    000100   51*    53    436
.SEEK    MACRO
.SETUP   MACRO
.SYSID   MACRO
.TIMER   MACRO
.TRAN    MACRO
.USER    MACRO
.WAIT    MACRO   388
.WAITR   MACRO
.WRITE   MACRO   168
.XV/MOF  MACRO
.XV/MON  MACRO
```