

IDENTIFICATION

PRODUCT CODE: DEC-12-AJAA-LA
PRODUCT NAME: FOCAL-12 LISTING
DATE CREATED: JANUARY 11, 1971
MAINTAINER: SOFTWARE SERVICES

COPYRIGHT © 1971
DIGITAL EQUIPMENT
CORPORATION

```

1 /FOCL12.37
2 /COPYRIGHT 1970; DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
3 PMODE /*****
4 FIXMRI FPOW=5000/PSEUDOC-FLOATING POINT INSTRUCTIONS.
5 FIXMRI FADD=1000
6 FIXMRI FSUB=2000
7 FIXMRI FMUL=4000
8 FIXMRI FDIV=3000
9 FIXMRI FGET=0000
10 FIXMRI FPUT=6000
11 7000 FNOR=7000
12 0000 FEXT=0
13 0000 FXIT=0
14 4407 FINT=JMS I 7
15 6101 SMP=6101
16 /MISCELLANEOUS ITEMS
17 0001 *1
18 0021 5403 JMP I ,*2 /INTERRUPT PROCESSOR ENTRY .
19 0000 LWETMP; 0 /*****
20 0003 2603 INTRPT
21 0004 0004 DDTJR, DDTJR /USED FOR DEBUGGING
22 0013 0013 P13, 13 /CONSTANT
23 0000 0100 C100, 100 /CONSTANT
24 0000 T#00 /TEXT FIELD NO.
25 0000 P#00 /PROGRAM FIELD NO.
26 7000 CDF=7000 /((X=MEM) - OPR
27 0007 6400 FPNT /ADDRESS OF FLOATING POINT INTERPRETER, (LOC *7)
28 /AUTO-INDEX REGISTERS = (START OF SAVE BY QUAD)
29 0010 0000 AXIN, 0 /STORAGE INDEX (LOC *10)
30 0011 0000 XRT, 0 /EXTRA XR
31 0012 0000 XRT2, 0 /EXTRA XR
32 0013 3600 PDLXR, BEGIN=1 /PUSHDOWN LIST INDEX REGISTER.
33 0014 3117 FLTXR, IOBUF=1 /XR FOR FLOATING POINT
34 0015 0000 FLTXR2, 0 /EXTRA FOR F.P.
35 0016 7402 TELSW, HLT /TELETYPE IN PROGRESS SWITCH
36 0017 0017 TEXTP#, /TEXT POINTERS (LOC *17)
37 0017 3214 AXOUT, FRSTX /OUTPUT INDEX
38 0020 0000 XCT, 0 /UNPACK SWITCH
39 0021 0000 GTEM, 0 /UNPACK STORAGE
40 0022 2407 PC, FLTZER /PROGRAM COUNTER
41 0023 0000 THISLN, 0 /LINE POINTER FROM 'FINDLN'
42 0024 0000 THISOP, 0 /CURRENT 'EVAL' OPERATION
43 0025 0000 LASTLN, 0 /BACK POINTER FROM 'FINDLN'
44 0026 0001 DEBGSW, 1 /DEBUG SWITCH ; NON-ZERO FOR LITERAL.
45 0027 0000 PACKST, 0 /RUBOUT PROTECTION
46 0030 0000 PT1, 0 /VARIABLE POINTER
47 0031 3216 LASTV, RUFBEQ /ADDRESS OF LAST VARIABLE
48 0032 0000 T1, 0 /TEMPORARY REGISTER - MAIN
49 0033 0000 T3, 0 /TEMP REGISTER FOR OUTPUT
50 0034 0000 INBUF, 0 /KEYBOARD INPUT BUFFER
51 0035 4617 BOTTOM, FEXP=1 /*****/LAST LOCATION CURRENTLY AVAILABLE IN FIELD ZERO **
52 0036 0000 INSUB, 0 /0= GETCI #0 = READC
53 0037 0000 HINBUF, 0 /HIGH SPEED INPUT BUFFER
54 /PAGE ZERO OF THE
55 /FLOATING POINT ARITHMETIC INTERPRETER FOR FOCAL

```

```

56      0242      *40
57      0147      0000      EX1,      1      /OPERAND STORAGE
58      0041      0000      AC1H,      3
59      0042      0000      AC1L,      3
60      0143      0000      OVER1,      2
61      0144      0000      FLAC=,      /FLOATING ACCUMULATOR
62      0144      0000      EXP,      0      /F.A.
63      0045      0000      WORD,      0
64      0046      0000      LORD,      0
65      0047      0000      OVER2,      3
66      0050      0000      SIGNF,      0      /FLOATIN SIGN
67      0051      6605      MINSKI, ACINS      /NEGATE FLAC SUBROUTINE
68      0052      2004      FISW,      2004      /OUTPUT FORMAT
69      0053      6724      INTEGER, FIX      /FIX FLAC
70      1345      GETSGN=TAD FLAC*1
71      5536      RETURN=JMP I EFUN3I
72      0054      *54
73      /VARIABLES = INITIALIZED FOR THE DIALOGUE
74      SORTCN, 0      /NUMBER IN TABLE FROM SORTC
75      LASTOP, 0      /LAST OPERATION FOR EVAL
76      EFOP=,      /FUNCTION CODE,
77      ATSW, 0      /ASK-TYPE SWITCH
78      CNTR, =20      /DELETE AND ERROR COUNTER(USED BY F.P. ALSO)
79      STARTV=,      /END FOR BK
80      BUFR, 3216      /NEXT LOCATION IN BUFFER = LAST LOCATION OF TEXT:
81      GADD, 0      /*****
82      XCTIN, 133      /PACK SWITCH
83      OUTDEV, XOUTL      /POINTER TO OUT, SUB. (OUTL)=FOR DEBUGGING
84      INDEV, XI33      /POINTER TO IN, SUB. (I33)=FOR DEBUGGING
85      NAGSW, 0001      /NOT ALL AND/OR GROUP SWITCH(4000=ONE;1=ALL;0=GROUP);(0000)=FOR TSS=8
86      CHAR, 215      /THE MOST IMPORTANT REGISTER
87      LINENO, 0000      /LINE NUMBER READ BY GETLN;(0400)=FOR TSS=8
88      GINC, WORDS*2      /#6 FOR 4=WORD = CONSTANT
89      T2, 0      /TEMP REGISTER = FOR NEW INST, ROUTINES.
90      /FOR DEBUGGING, SET OUTL AND I33 INTO OUTDEV AND INDEV;
91      /ALSO PATCH THE ERROR ROUTINE = FOUR
92      /PATCHES PLUS TWO FOR THE HIGH SPEED READER.
93      LIST6=, /INPUT LIST FOR "SFOUND".
94      0072      0214      214      /F,F.
95      0073      0207      207      /BELL
96      0074
97      0074      0203      203      /CONTROL=C FOR DEBUGGING AND TSS8
98      0075      0337      337      /LEFT ARR
99      0076      0212      212      /L.F.
100     0077
101     0077      0215      215      /EXCRETION LIST
102     0100      7482      7482      /LIST BRANCHER,
103     /MPSW, HLT      /(SEARCH CHARACTER)=VARIABLE
104     /#0000 FOR TRACE ON,
105     /THE REST OF PAGE ZERO IS PURE TO THE MULTI-USER SYSTEM
106     0101      7700      7700      /LEFT MASK
107     0102      0256      256      /PERIOD
108     0103      7701      7701      /EXTEND CODE TEST
109     0104      7600      7600      /GROUP MASK
110     0105      7760      7760      /CONSTANT

```

111	0106	177	P177,	177	/STEP MASK
112	0107	317	P17,	17	/BCD MASK
113	0110	277	P277,	277	/"?"
114	0111	7776	M2,	-2	/CONSTANT
115	0112	7477	MINUSA,	-301	/CONSTANT
116	0113	260	C260,	260	/ASCII FOR ZERO
117	0114	7540	M240,	-240	/SPACE TEST
118	0115	7522	MPER,	-256	/PERIOD TEST
119	0116	7563	MCR,	-215	/C.R. TEST
120	0117	7775	MFLT,	=WORDS	/# -4 FOR 4-WORD
121	0120	7773	M5,	=5	/PAREN TEST
122	0121	7767	M11,	=11	/PAREN TEST
123	0122	0077	P77,	77	/RIGHT MASK
124	0123	0200	C200,	200	/CONSTANTS
125	0124	4000	P4000,	4000	/NAGSW TEST CONSTANT (FOR PDP-5)
126	0125	2032	FLARGP,	FLARG	/DATA ADDRESS
127	0126	2157	PTCH,	CHIN	/GENERAL CHARACTER INPUT ROUTINE.
128	0127	5715	DOUBLE,	MULT2	/MULTIPLY FLAG BY 2
129	0130	6000	FOUTPUT,	FLOUTP	/FLOATING OUTPUT
130	0131	6200	FINPUT,	FLINTP	/FLOATING INPUT
131	0132	3140	COMBUF,	COMEIN	/COMMAND BUFFER START
132	0133	3206	CFRS,	FRST	/ADDRESS OF DUMMY LINE.
133	0134	3140	END,	COMEIN	/FIRST LOCATION USED IN 8K.
134	0135	3216	ENDT,	BUFBEG	/START OF STORAGE AREA **
135	0136	2021	EFUN3I,	EFUN3	/FUNCTION RETURN
136	0137	2407	CFRSX,	FLTZER	/POINTER TO ZERO DATA
137					
138					
139					/FINPUT! USES CHAR AND GETC OR READC TO DEVELOP
140					/A NUMBER WHICH IS THEN STORED VIA PT1.
141		0003			WORDS=3 /OR 4
142					/NEW INSTRUCTIONS!
143		4540	PUSHJ=JMS I,		/RECURSIVE SUBROUTINE CALL
144	0140	0521	XPIUSHJ		
145		1413	POPA=TAD I PDLXR,		/RESTORE AC
146		5541	POPJ=JMP I,		/SUBROUTINE RETURN
147	0141	1565	XPOPJ		
148		4542	PUSHA=JMS I,		/SAVE AC
149	0142	0477	XPUSHA		
150		4543	PUSHF=JMS I,		/SAVE GROUP OF DATA
151	0143	0534	PD2		
152		4544	POPF=JMS I,		/RESTORE GROUP
153	0144	0554	PD3		
154		4545	GETC=JMS I,		/UNPACK A CHARACTER
155	0145	2274	UTRA		
156		4546	PACKC=JMS I,		/PACK A CHARACTER
157	0146	2502	PACBUF		
158		4547	SORTJ=JMS I,		/SORT AND BRANCH ON AC OR CHAR
159	0147	1312	SORTB		
160		4550	SORTC=JMS I,		/SORT CHAR
161	0150	0721	XSORTC		
162		4551	PRINTC=JMS I,		/PRINT AC OR CHAR
163	0151	2465	OUT		
164		4552	READC=JMS I,		/READ DATA INTO CHAR AND PRINT IT
165	0152	2157	RDIV,	CHIN	

```

166      4553      PRNTLN=JMS I , /PRINT c(LINENO)
167      0153      2425      XPRNT
168      4554      GETLN=JMS I , /UNPACK AND FORM A LINENUMBER
169      0154      302      XGETLN
170      4555      FINDLN=JMS I , /SEARCH FOR A GIVEN LINE
171      0155      2244      XFIND
172      4556      ENDLN=JMS I , /INSERT LINE POINTERS
173      0156      2360      XENDLN
174      4557      RTL6=JMS I , /ROTATE LEFT SIX
175      0157      413      XRTL6
176      4560      SPNOR=JMS I , /IGNORE SPACES AND LEADING ZEROS
177      0160      1535      XSPNOR
178      4561      TESTN=JMS I , /PERIOD; OTHER; NUMBER
179      0161      1546      XTESTN
180      4562      TSTLPR=JMS I , /SKIP IF 5<SORTCN<= 11 (I.E. AN L=PAR)
181      0162      2037      LPRTST
182      4563      TSTGRP=JMS I , /SKIP IF G(AC) = G(LINENO)
183      0163      744      GRPTST
184      4564      TESTC=JMS I , /TERM; NUMBER; FUNCTION; LETTER- AND IGNORE SPACES;
185      0164      700      XTESTC
186      4565      DELETE=JMS I , /REMOVE OLD TEXT LINE
187      0165      2064      PSIN, XDELETE
188      4566      ERROR2=JMS I , /EXCESS SOMETHING ERROR
189      4566      ERROR3=JMS I , /MISCELLANEOUS ERROR
190      4566      ERROR4=JMS I , /FORMAT ERROR
191      0166      2726      ERR2
192      /USED BY 8K
193      /FOCAL'S COMMAND/INPUT DRIVER
194      0167      *167      /*****
195      0167      0000      SUBS2; 0      /*****
196      0170      0000      LESUB2; 0      /*****
197      0171      0000      SUBS; 0      /*****
198      0172      6163      LEFPUT; LEPUT      /*****
199      0173      0000      LESUBS; 0      /*****
200      0174      7657      PWAIT; WAIT      /*****
201      0175      7672      PCLEAR; CLEAR      /*****
202      0176      3601      BEGIN /BECOMES (RECOVR+1) **
203      0177      7610      START; SKP CLA /PROGRAM START FROM SELF
204      0200      5576      JMP I ,=2 /CONSOLE START; SW=200,
205      0201      1137      TAD CFRSX / (PC) => 0
206      0202      3022      DCA PC /FOR COMMAND MODE
207      0203      7001      IAC /USE ONE IN THE AC TO
208      0204      3120      DCA DMPSW /INIT UNPACK AND TRACE SWITCH,
209      0205      3026      DCA DEBGSW /ENABLE TRACE FOR INPUT OF (?),
210      0206      1226      TAD COMBOT /PROTECT COMMAND BUFFER,
211      0207      3013      DCA POLXR /NO PATCH TEST,
212      0210      1225      TAD CSTAR /ANNOUNCE PRESENCE
213      0211      4551      PRINTC /BY TYPING THE LEAD-IN CHARACTER
214      0212      1132      IBAR, TAD COMBUF /INITIALIZE COMMAND BUFFER
215      0213      3010      DCA AXIN /FOR UNPACKING,
216      0214      3062      DCA XCTIN
217      0215      1132      TAD COMBUF /RUBOUT PROTECTION
218      0216      3027      DCA PACKST
219      0217      4552      IGNOR, READC /READ COMMAND STRING
220      0220      4547      SORTJ

```

```

221 0221 0073          LIST7=1
222 0222 0474          INLIST=LIST7
223 0223 4546          PACKC          /SAVE STRING CHARACTER,
224 0224 5217          JMP IGNOR
225 0225 0252          CSTAR, 252          /ACKNOWLEDGE CHARACTER
226 0226 0220          COMBUT, COMEOUT+12      /END OF COMMAND BUFFER, LESS PROTECTION COUNT.
227                    /COMMAND/INPUT PROCESSOR
228 0227 4546          IRETN, PACKC          /START TO PACK C.R.
229 0228 4546          PACKC          /FINISH C.R.
230 0231 1132          TAD COMBUF          /INITIALIZE "TEXTP"
231 0232 3017          GONE, DCA AXOUT          /SETUP CURRENT LINE
232 0233 3020          DCA XCT
233 0234 4545          GETC          /READ FIRST CHARACTER,
234 0235 1035          TAD BOTTOM          /INIT PUSH=DOWN=LIST
235 0236 3013          DCA PDLXR
236 0237 4560          SPNOR          /IGNORE LEADING BLANKS
237 0240 4561          TESTN          /DOES THE LINE BEGIN WITH 1-9?
238 0241 5362          JMP GZERR          /PERIOD =ILLEGAL GROUP ZERO USAGE
239 0242 5271          JMP INPUTX          /NO
240 0243 0226          ISZ DEBGSW          /YES, DISABLE TRACE FOR REPACKING
241 0244 4554          GETLN          /READ THIS LINE NUMBER
242 0245 1124          TAD P4000          /TEST FOR SINGLE LINE.
243 0246 1065          TAD NAGSW
244 0247 7640          SZA CLA
245 0250 4566          ERROR3          /ILLEGAL LINE NUMBER ON INPUT
246 0251 1060          TAD BUFR          /SET POINTERS
247 0252 3010          DCA AXIN
248 0253 3062          DCA XCTIN
249 0254 1067          TAD LINENO          /SAVE LINE #
250 0255 3410          DCA I AXIN          / (X=MEM)
251 0256 4560          SPNOR          /IGNORE SPACES AFTER LINE NUMBER
252 0257 7410          SKP
253 0260 4545          GETC          /READ 1ST AFTER LINENO TERMINATOR,
254 0261 4546          SRETN, PACKC          /SAVE TEXT AND RESTORE DATA FIELD
255 0262 1066          TAD CHAR          /TEST FOR END OF INPUT STRING
256 0263 1116          TAD MCR
257 0264 7640          SZA CLA
258 0265 5260          JMP ,=5
259 0266 4565          DELETE          /REMOVE OLD LINE, IF ANY,
260 0267 4556          ENDLN          /INSERT NEW LINE
261 0270 5177          JMP START          /POINTERS MUST BE REINITIALIZED
262 0271 4540          INPUTX, PUSHJ          /PROCESS IMMEDIATE COMMAND,
263 0272 0611          PROC
264 0273 1422          TAD I PC          /CHECK NEXT LINE (X=MEM)
265 0274 7450          SNA          /END OF PROGRAM?
266 0275 5177          JMP START          /YES
267 0276 3022          DCA PC          /SAVE NEW LINE NO.
268 0277 1022          TAD PC          /START NEW LINE
269 0300 7401          IAC
270 0301 5232          JMP GONE          /PROCESS OTHER COMMANDS
271                    /TEXT LINE BUFFER FORMAT*
272                    /#1 : POINTER OR ZERO IN LAST
273                    /#2 : LINENO
274                    /#3 - #N+1 : TEXT
275                    /#N : C.R.

```

276			/LINE NUMBER FORMATION	
277	0302	4567	XGETLN, 0	/DEVELOP I.D. = "GETLN"
278	0303	4567	SPNOR	/IGNORE LEADING SPACES,
279	0304	1266	TAD CHAR	/"ALL" IS A SPECIAL ARGUMENT.
280	0305	1112	TAD MINUSA	
281	0306	7650	SNA CLA	
282	0307	5322	JMP TESTA	
283	0310	3036	DCA INSUB	/CALL 'GETC' FROM 'INPUT' VIA 'DECON'
284	0311	4771	JMS I LCON	/(DECONV - IN FLOAT.)
285	0312	1247	TAD FLAC*3	/GROUP TOO LARGE?
286	0313	1372	AND P7740	
287	0314	1046	TAD FLAC*2	
288	0315	7640	SZA CLA	
289	0316	4566	ERROR2	/GROUP NUMBER TOO LARGE
290	0317	1047	TAD FLAC*3	
291	0320	4557	RTL6	
292	0321	7004	RAL	
293	0322	3067	TESTA, DCA LINENO	
294	0323	4561	TESTN	/TEST3
295	0324	4545	GETC	/READ STEP NUMBER,
296	0325	4561	TESTN	/TEST4, OTHER
297	0326	5340	JMP GERR	/DOUBLE PERIODS
298	0327	5352	JMP GEXIT	/OTHER
299	0330	1054	TAD SORTCN	/NUMBER
300	0331	7106	RTL CLL	
301	0332	1054	TAD SORTCN	
302	0333	7004	RAL	
303	0334	1067	TAD LINENO	
304	0335	3067	DCA LINENO	
305	0336	4545	GETC	/READ SECOND STEP NUMBER.
306	0337	4561	TESTN	/TEST4, OTHER
307	0340	4566	GERR, ERROR4	/DOUBLE PERIODS
308	0341	5352	JMP GEXIT	/OTHER
309	0342	1054	TAD SORTCN	/NUMBER
310	0343	1067	TAD LINENO	
311	0344	3067	DCA LINENO	
312	0345	4545	GETC	/TEST FOR CORRECT TERMINATOR
313	0346	4561	TESTN	/CHECK SIZE
314	0347	5340	JMP GERR	/.
315	0350	7410	SKP	
316	0351	4566	ERROR2	/TOO LARGE A LINE NUMBER.
317	0352	7100	GEXIT, CLL	/CLEAR LINK BIT
318	0353	1067	TAD LINENO	/TEST FOR GROUP NUMBER,
319	0354	104	AND P7600	
320	0355	7640	SZA CLA	
321	0356	7020	CML	
322	0357	1067	TAD LINENO	
323	0360	106	AND P177	/REPAIR "NAGSW"
324	0361	7460	SNL SZA	
325	0362	4566	GZERR, ERROR2	/0.X = ERROR:ILLEGAL LINE NUMBER.
326	0363	7640	SZA CLA	
327	0364	1373	TAD P2020	
328	0365	7020	CML	
329	0366	7004	RAL	
330	0367	3065	DCA NAGSW	

```

331 0370 5702          JMP I XGETLN
332 0371 5600          LCON, DECONV
333 0372 7740          P7740, 7740
334 0373 2000          P2000, 2000
335                    /RANGE OF ACCEPTIBLE LINE NUMBERS = 1,21 TO 31,99
336                    /NAGSW:
337                    /GROUP=2000
338                    /LINE=4000
339                    /ALL=0001
340                    /LIST OF FUNCTION ADDRESSES. (NAMES ARE IN "FNTABL")
341                    FNTABF=.
342 0374 2016          XABS /ABS =ABSOLUTE VALUE
343 0375 2012          XSGN /SGN =SIGN PART
344 0376 1156          XINT /ITR =INTEGER PART
345 0377 7602          XDISP /DIS /*****
346 0400 1145          XRN /RAN =RANDOM NUMBER
347 0401 1341          XADC /ADC =READ ANALOG TO DIGITAL CONVERTER
348 0402 5000          ARTN /ATN =
349 0403 4620          FEYP /EXP =EXPONENTIAL FUNCTIONS
350 0404 5040          FLOG /LOG =
351 0405 5204          FSIN /SIN =TRIG FUNCTIONS
352 0406 5177          FCOS /COS =
353 0407 7400          XSORT /SQT =SQUARE ROOT
354 0410 2725          PFNEW, ERRORS /NEW =USER DEFINED FUNCTIONS
355 0411 2725          PFX, ERRORS /FX /*****
356 0412 2725          PFZ, ERRORS /FZ /*****
357 0413 0000          XRTL6, 0 /ROTATE AC LEFT SIX = "RTL6"
358 0414 7106          CLL RTL
359 0415 7006          RTL
360 0416 7006          RTL
361 0417 5613          JMP I XRTL6
362                    /RECURSIVE OPERATE, EXECUTE, OR CALL
363 0420 4554          DO, GETLN /EXECUTE ONE LINE, A GROUP, OR ALL
364 0421 1022          TAD PC /SAVE ADDRESS
365 0422 4542          PUSHA /OF CURRENT LINE
366 0423 4543          PUSHF /SAVE REST OF THIS LINE
367 0424 0017          TEXTP /ADDRESS OF TEXT POINTERS
368 0425 4543          DGRP, PUSHF /SAVE NAGSW; CHAR; AND LINENO.
369 0426 0065          NAGSW
370 0427 1065          TAD NAGSW /CHECK DATA FROM GETLN.
371 0430 7710          SPA CLA /SKIP IF GROUP OR ALL
372 0431 5263          JMP DOONE /DO ONE LINE
373 0432 4555          FINDLN /INIT FOR GROUP AND SET THISLN
374 0433 7000          NOP
375 0434 1023          TAD THISLN /TEST FOR GOOD GROUP NUMBER.
376 0435 3011          DCA XRT
377 0436 1411          TAD I XRT /(X=MEM)
378 0437 4563          TSTGRP
379 0440 4566          ERROR2 /NO SUCH GROUP NUMBER
380 0441 4543          DGRP1, PUSHJ /EXECUTE OBJECT LINE AND SET PC.
381 0442 1026          PROCESS-2
382 0443 4544          POPF /RESTORE THE DATA
383 0444 0065          NAGSW
384 0445 1422          TAD I PC /CHECK FOR END OF TEXT (X=MEM)
385 0446 7450          SNA

```


386	0447	5271	JMP DCONT	/ALL DONE
387	0450	7001	IAC	
388	0451	3030	DCA PT1	/SAVE POINTER TO LINENO
389	0452	1365	TAD NAGSW	/CHECK FOR GROUP
390	0453	7740	SMA SZA CLA	
391	0454	5260	JMP ,+4	/DO ALL
392	0455	1430	TAD I PT1	/TEST GROUP (X=MEM)
393	0456	4563	TSTGRP	
394	0457	5271	JMP DCONT	/NOT IN GROUP
395	0460	1430	TAD I PT1	/READ NEXT LINE NO. (X=MEM)
396	0461	3067	DCA LINENO	
397	0462	5225	JMP DGRP	/CONTINUE THE SUBROUTINE
398	0463	4555	DOONE, FINDLN	/FIND THE LINE
399	0464	4566	ERROR2	/NO SUCH LINE NUMBER
400	0465	4540	PUSHJ	/EXECUTE IT
401	0466	0610		
402	0467	4544	POPF	PROCESS /RESTORE CHAR
403	0470	0065		
404	0471	4544	DCONT, POPF	NAGSW /RESTORE TEXT POINTERS
405	0472	0017		
406	0473	1413	POPA	TEXTP /RESTORE ADDRESS OF CURRENT LINE.
407	0474	3022	DCA PC	
408	0475	5676	JMP I ,+1	/CONTINUE PROCESSING THIS LINE.
409	0476	0611		
410			PROC	
411	0477	0000	/PUSHDOWN LIST CONTROLS	
412	0500	3071	XPUSHA, 0	/PUSHDOWN THE AC = "PUSHA"
413	0501	7040	DCA T2	/BACKUP POINTER
414	0502	4310	CMA	/AND THEN
415	0503	1071	JMS PCHK	/CHECK CORE USAGE
416	0504	3413	TAD T2	/OK
417	0505	7040	DCA I PDLXR	/PUSH DOWN LIST POINTER
418	0506	4310	CMA	/BACKUP AGAIN
419	0507	5677	JMS PCHK	
420	0510	0000	JMP I XPUSHA	
421	0511	1013	PCHK, 0	
422	0512	3013	TAD PDLXR	/INC IN AC
423	0513	1013	DCA PDLXR	
424	0514	7141	TAD PDLXR	
425	0515	1031	CIA CLL	
426	0516	7630	TAD LASTV	
427	0517	4566	SZL CLA	
428	0520	5710	ERROR3	/STORAGE FILLED BY PUSH-DOWN LIST
429	0521	0000	JMP I PCHK	
430	0522	1721	XPUSHJ, 0	/RECURSIVE SUBROUTINE CALL = "PUSHJ"
431	0523	3071	TAD I XPUSHJ	
432	0524	7040	DCA T2	/SAVE SUBR. ADDR.
433	0525	4310	CMA	
434	0526	1321	JMS PCHK	
435	0527	7001	TAD XPUSHJ	
436	0530	1413	IAC	
437	0531	7340	DCA I PDLXR	/SAVE RETURN
438	0532	4310	CMA	
439	0533	5471	JMS PCHK	
440	0534	0000	JMP I T2	/TRANSFER CONTROL
			PD2, 0	/SAVE A FLOATING POINT NUMBER = "PUSHF"

441	0535	7240	CLÄ CMA	/COMPUTE VARIABLE ADDR
442	0536	1734	TAD I ,=2	
443	0537	3011	DCA XRT	
444	0540	2334	ISZ PD2	/FIX RETURN
445	0541	1117	TAD MFLT	/COMPUTE PUSH. POINTER
446	0542	4310	JMS PCHK	
447	0543	1117	TAD MFLT	
448	0544	3071	DCA T2	
449	0545	1411	TAD I XRT	
450	0546	3413	DCA I PDLXR	
451	0547	2071	ISZ T2	
452	0550	5345	JMP ,=3	
453	0551	1117	TAD MFLT	/RESET POINTER
454	0552	4310	JMS PCHK	
455	0553	5734	JMP I PD2	
456				
457				
458	0554	0000	PD3, 0	/RESTORE A FLOATING POINT NUMBER = "POPF"
459	0555	7240	CLÄ CMA	/GET VAR. ADDR,
460	0556	1754	TAD I PD3	
461	0557	2354	ISZ PD3	
462	0560	3011	DCA XRT	
463	0561	1117	TAD MFLT	
464	0562	3071	DCA T2	
465	0563	1413	TAD I PDLXR	/MOVE
466	0564	3411	DCA I XRT	
467	0565	2071	ISZ T2	
468	0566	5363	JMP ,=3	
469	0567	5754	JMP I PD3	/EXIT
470		0570	INLIST=.	/INPUT CONTROL CHARACTERS
471	0570	2740	RECOVR	/C.C. = BREAK
472	0571	0212	IBAR	/B.A. = RESTART
473	0572	0217	IGNOR	/L.F. = IGNORE
474	0573	0227	IRETN	/C.R. = TERMINATE STRING
475	0574	1075	FLIST2, FLIMIT	/,=STANDARD
476	0575	1137	FINFIN	/I=SHORT
477	0576	2725	ERROR5	/CR=DUMB
478	0577	1065	FLIST1, FINCR	/,=STANDARD FORMAT
479	0600	0610	PROCESS	/I=SETIPLUS ,,,
480	0601	0614	PC1	/C.R.=SET COMMAND.
481	0602	7472	MF, =306	/USED BY TESTC
482			/PRIMARY CONTROL AND TRANSFER	
483	0603	4554	GOTO, GETLN	/READ THE LINE NUMBER REQUESTED
484	0604	4555	FINDLN	/LOCATE IT AND RESET TEXTP
485	0605	4566	ERROR2	/NOT THERE
486	0606	1023	TAD THISLN	/SET PC
487	0607	3022	DCA PC	
488	0610	4545	PROCESS, GETC	/TEST FOR END OF LINE
489	0611	1066	PROC, TAD CHAR	/FIRST CHARACTER READY = USE PROC
490	0612	1116	TAD MCR	
491	0613	7650	SNA CLA	
492	0614	5541	PC1, POPJ	/EXIT "PROCESS"
493	0615	4550	SORTC	/IGNORE "SPACE", ",", AND "I".
494	0616	1374		
495	0617	5210	JMP PROCESS	

496	2622	1266	TAD CHAR	/SAVE COMMAND CHARACTER
497	2621	1375	AND P337	/EXECUTE LOWER CASE ALSO
498	2622	4542	PUSHA	
499	2623	4545	GETC	/GO TO TERMINATOR
500	2624	4552	SORTC	
501	2625	1374		GLIST=1
502	2626	7410	SKP	
503	2627	5223	JMP ,=4	
504	2630	1413	POPA	
505	2631	4547	SORTJ	/GO TO COMMAND
506	2632	773		COMLST=1
507	2633	1165		COMGO=COMLST
508	2634	4566	ERROR2	/ILLEGAL COMMAND
509		614	COMMENTS=PC1	/ALSO IS CONTINUE
510				
511				
512			/OUTPUT COMMAND TEXT	
513	2635	4554	WRITE: GETLN	/SET LINENO
514	2636	2826	ISE DEBGSW	/DISABLE TRACE
515	2637	4555	FINDLN	/SEARCH FOR LINE NUMBER
516	2640	5267	JMP WTESTG	/NOT THERE OR GROUP
517	2641	1067	TAD LINENO	
518	2642	7640	SZA CLA	
519	2643	4553	PRNTLN	/PRINT LINE NUMBER AND A SPACE.
520	2644	4545	GETC	
521	2645	4551	PRINTC	/PRINT TEXT OF A LINE.
522	2646	1066	TAD CHAR	
523	2647	1116	TAD MCR	
524	2650	7640	SZA CLA	/SKIP IF END OF LINE
525	2651	5244	JMP ,=5	
526	2652	1423	TAD I THISLN	/TEST FOR END OF TEXT (X=MEM)
527	2653	7450	WTEST2: SNA	
528	2654	5271	JMP WX=2	/EXIT/DO NEXT INDIRECT LINE.
529	2655	7001	IAC	
530	2656	3030	DCA PT1	/SAVE POINTER TO LINENO OF NEXT
531	2657	1265	TAD NAGSW	
532	2660	7700	SMA CLA	
533	2661	1430	TAD I PT1	/(X=MEM)
534	2662	4563	TSTGRP	/TRY NEXT LINENO FOR GROUP.
535	2663	5273	JMP WX	
536	2664	1430	WALL: TAD I PT1	/SET LINENO (X=MEM)
537	2665	3067	DCA LINENO	
538	2666	5237	JMP WRITE+2	
539	2667	1023	WTESTG: TAD THISLN	/INIT GROUP PRINTOUT
540	2670	5253	JMP WTEST2	
541	2671	3026	DCA DEBGSW	
542	2672	5541	POPJ	
543	2673	1065	WX: TAD NAGSW	
544	2674	7750	SPA SNA CLA	/SKIP IF ALL
545	2675	5271	JMP WX=2	
546	2676	4551	PRINTC	/PRINT C.R. AGAIN
547	2677	5264	JMP WALL	
548	2700	0000	XTESTC: 0	/TEST THE NATURE OF THE NEXT ALPHANUMERIC = "TESTC"
549	2701	4560	SPNOR	/IGNORE SPACES
550	2702	4552	SORTC	/TEST THE VARIABLE TERMINATORS

```

551 0703 1771          TERMS=1
552 0704 5700          JMP I XTESTC /YES = SORTCN IS SET
553 0705 1066          TAD CHAR /NO
554 0706 2300          ISZ XTESTC
555 0707 1202          TAD MF
556 0710 7650          SNA CLA /TEST FOR "F"
557 0711 5317          JMP XT3
558 0712 4561          TESTN
559 0713 5700          JMP I XTESTC /
560 0714 7410          SKP /OTHER
561 0715 5700          JMP I XTESTC /NUMBER
562 0716 2300          ISZ XTESTC
563 0717 2300          XT3, ISZ XTESTC /RETURNS:ITIN;FJA
564 0720 5700          JMP I XTESTC
565 0721 0000          XSORTC, 0 /SORT CHAR AGAINST TABLE = "SORTCN"
566 0722 1721          TAD I XSORTC
567 0723 3012          DCA XRT2 /1ST ARG IS LIST=1
568 0724 1412          TAD I XRT2
569 0725 7510          SPA /LIST IS ENDED BY A NEGATIVE NUMBER
570 0726 5340          JMP SEXC /2ND EXIT = NOT IN LIST
571 0727 7041          CIA
572 0730 1066          TAD CHAR
573 0731 7640          SZA CLA /COMPARE
574 0732 5324          JMP ,=6
575 0733 1721          TAD I XSORTC /COMPUTE INCREMENT I 0 = N
576 0734 7040          CMA
577 0735 1012          TAD XRT2
578 0736 3054          DCA SORTCN
579 0737 7410          SKP /1ST EXIT = YES
580 0740 2321          SEXC, ISZ XSORTC
581 0741 2321          ISZ XSORTC
582 0742 7300          CLA CLL
583 0743 5721          JMP I XSORTC
584 0744 0000          GRPTST, 0 /AC VS LINENO = "TSTGRP"
585 0745 0104          AND P7600
586 0746 7041          CIA
587 0747 3071          DCA T2
588 0750 1067          TAD LINENO
589 0751 0104          AND P7600
590 0752 1071          TAD T2
591 0753 7650          SNA CLA
592 0754 2344          ISZ GRPTST
593 0755 5744          JMP I GRPTST
594
595 /INPUT FROM TEXT OR KEYBOARD;
596 /IF BACK-ARROW, RESTART INPUT
596 0756 0000          INPUT, 0 /INPUT A CHARACTER
597 0757 1036          TAD INSUB /NON-ZERO FOR KEYBOARD
598 0760 7640          SZA CLA
599 0761 5364          JMP ,+3
600 0762 4545          GETC
601 0763 5756          JMP I INPUT
602 0764 4552          READC
603 0765 4547          SORTJ
604 0766 6776          SPECIAL=1
605 0767 3402          INFIX=SPECIAL

```

606 0770 5756
607 0771 1435
608 0772 610
609 0773 614
610
611 774
612 0774 323
613 0775 306
614 0776 311
615 0777 304
616 1222 307
617 1221 303
618 1222 301
619 1223 324
620 1224 317
621 1225 305
622 1226 327
623 1227 315
624 1210 321
625 1211 322
626 1212 314
627
628
629
630
631 1213 4564
632 1214 4637
633 1215 2013
634 1216 4640
635 1217 1111
636 1220 3032
637 1221 1045
638 1222 7510
639 1223 2032
640 1224 7750
641 1225 2032
642 1226 7410
643 1227 5765
644 1230 4547
645 1231 1375
646 1232 7373
647 1233 4545
648 1234 5230
649 1235 4545
650 1236 5225
651 1237 1601
652 1240 2051
653
654 1241
655 1241 4540
656 1242 1401
657 1243 4560
658 1244 1066
659 1245 1335
660 1246 7442

JMP I INPUT
ILIST, IF1 /,
PROCESS /I
PC1 /CR
COMLST=, /ENGLISH=FRENCH
/COMMAND DECODING LIST
323 /SET = ORGANIZE
306 /FOR = QUAND
311 /IF = SI
304 /DO = FAIZ
307 /GOTO = VA
303 /COMMENT= COMMENTE
301 /ASK = DEMANDE
324 /TYPE = TAPE
317 /OUTPUT /*****
305 /ERASE = BIFFE
327 /WRITE = INSCRIS
315 /MODIFY = MODIFIE
321 /QUIT = ARRETE
322 /RETURN = RETOURNE
314 /LIBR****
/THIS COMMAND LIST IS SPEED OPTIMIZED.
/CONDITIONAL TRANSFER PROCESS,
IF, TESTC /IGNORE SPACES AND TEST
JMS I IECALL /T
ISZ PDLXR /N=DUMP THE (EPOP)
JMS I IPART /F=CHECK FOR PAREN MATCH
TAD M2 /A
DCA T1
TAD FLAC*1 /TEST =,0,*
SPA
ISZ T1 /N=TO -1,-2,-3
SPA SNA CLA
IF3, ISZ T1 /COUNT COMMAS
SKP
JMP I COMGO+4 /TRANSFER
SORTJ /SEARCH TEXT UNTILL ,I.C.R.
TLIST=1
ILIST=TLIST
GETC
JMP ,=4
IF1, GETC /MOVE PAST COMMA
JMP IF3
IECALL, ECALL
IPART, PARTEST
/LOOP CONTROL STATEMENT
SETT=, /SUBSET OF "FOR".
FOR, PUSHJ /LOOPS, ETC.
GETARG /LOOK FOR "=" NEXT
SPNOR /IGNORE SPACES
TAD CHAR
TAD MEQ
SZA

661	1047	4566	ERROR4	/LEFT OF "=" IN ERROR: 'FOR' OR 'SET'
662	1050	1730	TAD PT1	
663	1051	4542	PUSHA	/SAVE POINTER TO VARIABLE
664	1052	4540	PUSHJ	
665	1053	1612	EVAL=1	/GET INITIAL VALUE EXPRESSION
666	1054	1413	POPA	
667	1055	3030	DCA PT1	
668	1056	4407	FINT	/INITIALIZE NOV.
669	1057	6430	FPUT I PT1	
670	1060	0000	FXIT	
671	1061	4547	SORTJ	/TEST LAST CHAR FROM "EVAL"
672	1062	1375	TLIST=1	
673	1063	7201	FLIST1=TLIST	
674	1064	4566	ERROR4	/EXCESS R=PAR
675	1065	1030	FINCR, TAD PT1	/SAVE VARIABLE ADDRESS *
676	1066	4542	PUSHA	
677	1067	4540	PUSHJ	/EVALUATE THE INCREMENT, IF ANY.
678	1070	1612	EVAL=1	
679	1071	4547	SORTJ	/TEST TERMINATORS
680	1072	1375	TLIST=1	
681	1073	7176	FLIST2=TLIST	
682	1074	4566	ERROR4	/ILLEGAL TERMINATOR IN 'FOR'
683	1075	4543	FLIMIT, PUSHF	/SAVE THE INCREMENT. *
684	1076	2032	FLARG	
685	1077	4540	PUSHJ	/GET THE LIMIT(NO ERROR DETECTION AFTER LIMIT)
686	1100	1612	EVAL=1	
687	1101	4543	FCONT, PUSHF	/SAVE THE LIMIT *
688	1102	2032	FLARG	
689	1103	4543	PUSHF	/SAVE TEXT OF OBJECT STATEMENTS
690	1104	0017	TEXTP	
691	1105	4540	PUSHJ	/DO THE OBJECT STATEMENTS
692	1106	0610	PROCESS	
693	1107	4544	POPF	/RESTORE REMAINING TEXT.
694	1110	0017	TEXTP	
695	1111	4544	POPF	/GET LIMIT
696	1112	2032	FLARG	
697	1113	4544	POPF	/GET INCREMENT
698	1114	7470	ITER1	
699	1115	1413	POPA	/GET VARIABLE ADDRESS
700	1116	3030	DCA PT1	
701	1117	4407	FINT	/INCREMENT AND TEST
702	1120	1430	FGET I PT1	/LOAD THE VARIABLE
703	1121	1733	FADD I FINKP	/INCREMENT IT
704	1122	6430	FPUT I PT1	/CHANGE IT
705	1123	2525	FSUB I FLARGP	/TEST IT
706	1124	0000	FXIT	
707	1125	1045	TAD FLAC*1	
708	1126	7740	SMA SEA CLA	
709	1127	5541	POPJ	/END OF LOOP
710	1130	1030	TAD PT1	
711	1131	4542	PUSHA	/SAVE ADDRESS *
712	1132	4543	PUSHF	/SAVE INCREMENT AGAIN *
713	1133	7470	FINKP, ITER1	
714	1134	5301	JMP FCONT	
715	1135	7503	MEQ, -275	

716 1135 7524
717 1137 4543
718 1143 2405
719 1141 5301
720
721
722
723 1142 0000
724 1143 2000
725 1144 0000
726 1145 4407
727 1146 1342
728 1147 4755
729 1150 6342
730 1151 0000
731 1152 3342
732 1153 3044
733 1154 5536
734 1155 6160
735
736 1156 4453
737 1157 7200
738 1160 5536
739
740 1161 1041
741 1162 1041
742 1163 1013
743 1164 0420
744 1165 0603
745 1166 0614
746 1167 1200
747 1170 1201
748 1171 7706
749 1172 2206
750 1173 0635
751 1174 1254
752 1175 0177
753 1176 1563
754 1177 6346
755
756 1200 7240
757 1201 3056
758 1202 4547
759 1203 1367
760 1204 0200
761 1205 2056
762 1206 5223
763 1207 4540
764 1210 1401
765 1211 1066
766 1212 4542
767 1213 1253
768 1214 4551
769 1215 2036
770 1216 7001

RCOM, -254
FINFIN; PUSHF /SET INCREMENT TO ONE.
FLTONE
JMP FCONT
/
/SAME PRAN = JUST MOVED
/
RANO, 0000 /*****
2000 /*****
0000 /*****
XRAN, FINT /*****
FADD RANO /*****
FMUL I CRUDDY /*****
FPUT RANO /*****
EXIT /*****
DCA RANO /*****
DCA FLAG /*****
JMP I EFUN3I /*****
CRUDDY; RANMUL /*****
/TAKE THE INTEGER PART
XINT, JMS I INTEGER /(FIX)
CLA
JMP I EFUN3I
COMGO: /COMMAND ROUTINE ADDRESSES
SETT
FOR
IF
DO
GOTO /(REFERENCED)
COMMENT
ASK
TYPE
OUTPUT /*****
ERASE
WRITE
MODIFY
START /RETURN TO COMMAND MODE VIA 'QUIT'
RETRN
LTAPE /*****
/INPUT-OUTPUT STATEMENTS
ASK, CLA CMA /REMEMBER WHICH CALL.
TYPE, DCA ATSW
TASK, SORTJ /SPECIAL CHAR? *****
ALIST=1
ATLIST=ALIST
ISZ ATSW /TEST QUOTE SWITCH
JMP TYPE2
PUSHJ /DO ASK; SETUP PT1
GETARG
TAD CHAR /SAVE IN-LINE CHARACTER.
PUSHA
TAD COL /TYPE COLON
PRINTC /((CLA)= TO SUPRESS "I"
ISZ INSUB /INDICATE 'READC'
IAC /POINT PAST CHAR

```

771 1217 4531 JMS I FINPUT /READ DATA AND SAVE
772 1220 1413 POPA /RE-TEST LAST TERMINATOR
773 1221 3066 DCA CHAR
774 1222 5230 JMP ASK /CONTINUE PROCESSING
775 1223 4542 TYPE2, PUSHJ /DO TYPE
776 1224 1613 EVAL
777 1225 4530 JMS I FOUTPUT /PRINT
778 1226 5201 JMP TYPE
779 1227 2026 TQUOT, ISZ DEBSW /DISABLE TRACE
780 1232 4545 GETC /TYPE LITERALS
781 1231 4547 SORTJ
782 1232 1531
783 1233 1645 TLIST2=1
784 1234 4551 TLIST3=TLIST2
785 1235 5230 PRINTC
786 1236 4545 JMP TQUOT+1
787 1237 4554 TINTR, GETC /PASS PERCENT SIGN
788 1240 1067 GETLN /READ FORMAT CONTROL "X7,03"
789 1241 3052 TAD LINENO
790 1242 5202 DCA FISW /SAVE FORMAT CODE
791 1243 1077 JMP TASK
792 1244 4463 TCRLF2, TAD CCR /SPLAT=CR ALONE
793 1245 7001 JMS I OUTDEV
794 1246 1077 IAC /NON-PRINTING DELAY FOR C.R, *****
795 1247 4551 TCRLF, TAD CCR /EXCLAMATION POINT=CR,LF,
796 1250 3026 PRINTC
797 1251 4545 TASK4, DCA DEBSW /*
798 1252 5202 GETC /*
799 1253 0272 JMP TASK
800 COL, 272 /"I"
801 /IF DEBSW=0 I ENABLE FLIP=FLOP "DMPSW"
802 / #0I DISABLE AND RETURN ALL?" I S.
803 /IF DMPSW = 0I TRACE ON, IF ENABLED
804 / #0I TRACE OFF
805 /IF BOTH = 0 I PRINT TRACE.
806 /SEARCH ROUTINES
807 MODIFY, GETLN /READ LINE NO,
808 1254 4554 FINDLN /LOOK IT UP NOW,
809 1255 4555 ERROR2 /NOT THERE = BAD COMMAND UNLESS ZERO,
810 1256 4566 TAD BUFR /SET POINTERS
811 1257 1060 DCA AXIN /FOR INPUT
812 1260 3010 DCA XCTIN
813 1261 3062 TAD LINENO /COPY THE SAME LINE NUMBER,
814 1262 1067 DCA I AXIN /(X=MEM)
815 1263 3410 TAD AXIN /SAVE START OF NEW LINE
816 1264 1010 DCA PACKST
817 1265 3027 JMS I INDEV /READ THE TELETYPE INPUT SILENTLY,
818 1266 4464 DCA LIST3+1 /SAVE SEARCH CHARACTER
819 1267 3100 ISZ DEBSW /NO BREAKS,
820 1270 2026 SCHAR, GETC /TYPE+TEST=F,F,
821 1271 4545 PRINTC /PLAYBACK THE TEXT
822 1272 4551 SORTJ /LOOK FOR MATCH
823 1273 4547
824 1274 0076 LIST3=1
825 1275 1267 LISTGO=LIST3
PACKC /SAVE NEW LINE,
JMP SCHAR

```


826 1300 1060
827 1301 7001
828 1302 3012
829 1303 3062
830 1304 4552
831 1305 4547
832 1306 0271
833 1307 1267
834 1310 4546
835 1311 5304
836 1312 0000
837 1313 7450
838 1314 1066
839 1315 7041
840 1316 3071
841 1317 1712
842 1320 2312
843 1321 3012
844 1322 1412
845 1323 7510
846 1324 5336
847 1325 1071
848 1326 7640
849 1327 5322
850 1330 1012
851 1331 1712
852 1332 3071
853 1333 1471
854 1334 3071
855 1335 5471
856 1336 2312
857 1337 7300
858 1340 5712
859
860 1341 4453
861 1342 0360
862 1343 1357
863 1344 3347
864 1345 6002
865 1346 6141
866 1347 0100
867 1350 0002
868 1351 6001
869 1352 3045
870 1353 3046
871 1354 7326
872 1355 3044
873 1356 5536
874 1357 0100
875 1360 0037
876 1361
877 1361 1271
878 1362 1266
879 1363 2740
880 1364 1300

SBAR, TAD BUFR /RESTART=B,A.
IAC
DCA AXIN /SET POINTERS
DCA XCTIN
SFOUND, READC /READ FROM KEYBOARD
SORTJ /TEST
LIST6=1
SRNLST=LIST6
SGOT, PACKC /PACK CHAR,
JMP SFOUND /MORE
SORTB, 0 /SORT AND BRANCH ROUTINE, = "SORTJ"
SNA
TAD CHAR /ASSUME CHAR IF AC=0
CIA
DCA T2 /SAVE SORT ITEM
TAD I SORTB /FIRST ARG IS LIST LESS ONE
ISZ SORTB /2AND IS INTRA-LIST LENGTH
DCA XRT2
TAD I XRT2
SPA /**LISTS ENDED BY NEGATIVE NUMBERS**
JMP SEX /READ EXIT
TAD T2 /FIND ADDRESS
SEA CLA
JMP ,=5
TAD XRT2 /MATCH FOUND.
TAD I SORTB
DCA T2
TAD I T2
DCA T2 /DEBUG I AC = ADDRESS
JMP I T2
SEX, ISZ SORTB /MATCH NOT FOUND.
CLA CLL
JMP I SORTB /RETURN TO CALLING SEQUENCE.
/ANALOGUE TO DIGITAL CONVERSION FOR PDP-12
XADC, JMS I INTEGER
AND 037 /*****
TAD OSAMP /*****
DCA ,*3 /*****
IOF /*****
6141 /LINC /*****
0100 /SAM ? /*****
0002 /PDP /*****
ION /*****
DCA FLAC+1 /*****
DCA FLAC+2 /*****
CLA CLL CML RTL /*****
DCA FLAC /*****
JMP I EFUN3I /*****
OSAMP, 0100 /SAM 0 /*****
037, 37 /*****
SRNLST=, /MODIFY CONTROL CHARACTER TABLE
SCHAR /F,F. = CONTINUE
SCONT /BELL = CHANGE SEARCH CHARACTER
RECOVR /C.C. = BREAK
SBAR /B.A. = RESTART

```

881 1365 1267 SCONT*1 /L.F. = FINISH THE LINE AS BEFORE,
882 1366 LISTGO=,
883 1366 261 SGETN /C.R. = END THE LINE HERE AS IS,
884 1367 1312 SGOI /CHAR = SEARCH CHARACTER
885 1370 ALIST=, /ASK/TYPE LIST OF CONTROLS,
886 1370 245 245 /%
887 1371 242 242 /"
888 1372 241 241 /|
889 1373 243 243 /#
890 1374 244 244 /S///
891 1375 GLIST=,
892 1375 240 240 /SPACE
893 1376 TLIST=,
894 1376 254 254 /;
895 1377 273 273 /|
896 1400 215 215 /C.R.
897
898 /THIS LIST IS ENDED BY 'TESTC',
899 1401 4564 GETARG, TESTC /FIRST LETTER OF ARG
900 1402 7200 P7200, 7200 /***** LETS F THRU
901 1403 4566 ERROR4 /*****
902 1404 7000 NOP /*****
903 1405 3062 GETVAR, DCA XCTIN /PACK INTO ADD,
904 1406 4546 PACKC
905 1407 4545 GETC /SECOND LETTER
906 1410 4550 SORTC /TERMINATOR?
907 1411 1771 TERMS=1
908 1412 5224 JMP GSERCH /YES
909 1413 1066 TAD CHAR /NO
910 1414 0122 AND P77 /SAVE 2ND LETTER OF NAME
911 1415 1061 TAD QADD
912 1416 3061 DCA QADD
913 1417 4545 GETC /IGNORE THE REST
914 1420 4550 SORTC
915 1421 1771 TERMS=1
916 1422 5224 JMP GSERCH
917 1423 5217 JMP ,=4
918 1424 4562 GSERCH, TSTLPR /LOOK FOR SUBSCRIPT VIA SORTCN
919 1425 5235 JMP GS1 /NOT SUBSCRIBED BY L=PAR,
920 1426 1061 TAD QADD /SAVE NAME
921 1427 3056 DCA EFOP /FOR RECURSIVE AND ERROR CHECK
922 1430 4663 JMS I GECALL /TO EVAL
923 1431 1413 POPA
924 1432 3061 DCA QADD /RESTORE NAME
925 1433 4662 JMS I PTEST /TEST PAREN MATCH, ETC,
926 1434 1453 JMS I INTEGER /CONVERT TO 12-BIT NUMBER,
927 1435 3171 GS1, DCA SUBS /SAVE SUBSCRIPT
928 1436 1061 TAD QADD /***** LETS F THRU
929 1437 1061 AND P7700 /*****
930 1440 1202 TAD P7200 /*****
931 1441 7050 SNA CLA /*****
932 1442 5322 JMP FFF /*****
933 1443 1060 TAD STARTV /SEARCH FOR VARIABLE(CHANGE FOR X=MEM)
934 1444 3030 GS3, DCA PT1
935 1445 1030 TAD PT1

```

936	1446	7041	CIA		
937	1447	1031	TAD LASTV	/TEST FOR END OF LIST	
938	1450	7750	SPA SNA CLA		
939	1451	5264	JMP GS2	/END SEARCH	
940	1452	1430	TAD I PT1	/GET TABLE ENTRY	
941	1453	7041	CIA		
942	1454	1061	TAD GADD		
943	1455	7650	SNA CLA		
944	1456	5310	JMP GFND1	/FOUND XX	
945	1457	1030	TAD PT1	/TRY NEXT ONE	
946	1460	1072	TAD GINC		
947	1461	5244	JMP GS3		
948	1462	2051	PARTEST		
949	1463	1601	ECALL		
950	1464	1031	TAC LASTV	/ADD THE VARIABLE	
951	1465	1005	TAD P13	/TEST STORAGE LIMITS	
952	1466	7141	CIA CLL		
953	1467	1013	TAD POLXR		
954	1470	7620	SNL CLA		
955	1471	4566	ERROR3		
956	1472	1031	TAD LASTV	/UPDATE THE LIST.	
957	1473	1070	TAD GINC		
958	1474	3031	DCA LASTV		
959	1475	1061	TAD GADD	/SAVE NAME	
960	1476	3430	DCA I PT1		
961	1477	2030	ISZ PT1	/SAVE SUBSCRIPT	
962	1500	1171	TAD SUBS		
963	1501	3430	DCA I PT1		
964	1502	2030	ISZ PT1	/SET PT1	
965	1503	4407	FINY		
966	1504	1537	FGET I CFRSX		
967	1505	6430	FPUT I PT1		
968	1506	6000	EXIT		
969	1507	5541	POPJ	/EXIT	
970	1510	1030	TAD PT1	/FOUND SAME	
971	1511	3011	DCA XRT	/TEST SUBSCRIPTS	
972	1512	1411	TAD I XRT		
973	1513	7041	CIA		
974	1514	1171	TAD SUBS		
975	1515	7640	SZA CLA		
976	1516	5257	JMP GS4	/WRONG SUBSCRIPT	
977	1517	2030	ISZ PT1	/SET POINTER TO DATA	
978	1520	2030	ISZ PT1		
979	1521	5541	POPJ		
980	1522	3030	DCA PT1	/***** SAVES SUBSCRIPT ON F	
981	1523	1061	TAD GADD	/*****	
982	1524	3002	DCA LWETMP	/*****	
983	1525	1045	TAD WORD	/*****	
984	1526	3170	DCA LESUB2	/*****	
985	1527	1171	TAD SUBS	/*****	
986	1530	3167	DCA SUBS2	/*****	
987	1531	5541	POPJ	/*****	
988	1532	242	TLIST2.	/*****	
989	1533	215		/*****	
990	1534	7520	M260.	/*****	

```

991
992 1535 1000 XSPNOR, 0 /*****
993 1536 1066 TAD CHAR /IGNORE LEADING SPACES = "SPNOR"
994 1537 1114 TAD M242
995 1540 7640 SZÄ CLA
996 1541 5735 JMP I XSPNOR
997 1542 4545 GETC
998 1543 5336 JMP XSPNOR+1
999
1000 1544 7506 M272, -272 /***** RECODING FOR SPACE
1001 1545 0012 012, 12 /*****
1002
1003
1004 1546 0000 XTESTN, 0 /*****
1005 1547 1066 TAD CHAR /RETURNS: ; OTHER: NUMBER = "TESTN"
1006 1550 1115 TAD MPR
1007 1551 7640 SZÄ CLA
1008 1552 2346 ISZ XTESTN
1009 1553 1066 TAD CHAR /***** RECODING FOR SPACE
1010 1554 1344 TAD M272 /*****
1011 1555 7100 CLL /*****
1012 1556 1345 TAD 012 /*****
1013 1557 3054 DCÄ SORTCN /*****
1014 1560 7430 SZL /*****
1015 1561 2346 ISZ XTESTN /*****
1016 1562 5746 JMP I XTESTN /*****
1017
1018 /EXIT FROM A "DO" SUBROUTINE
1019 1563 1137 RETRN, TAD CFRSX /(PC) => 0
1020 1564 3022 DCÄ PC
1021 1565 1413 XPOPJ, TAD I PDLXR /RECURSIVE EXIT = "POPJ"
1022 1566 3071 DCÄ T2
1023 1567 5471 JMP I T2
1024
1025 1570 1570 ATLIST=, /ASK-TYPE CONTROL CHARACTER TABLE
1026 1571 1236 TINTR /X = FORMAT DELIMITER
1027 1572 1227 TQUOT /" = LITERAL DELIMITER
1028 1573 1246 TCRLF /I = CARRIAGE RETURN AND LINE FEED
1029 1574 1243 TCRLF2 /# = CARRIAGE RETURN ONLY
1030 1574 3052 TDUMP /S/ = DUMP THE SYMBOL TABLE CONTENTS
1031 1575 1250 TASK4 /SP = TERMINATOR FOR NAMES
1032 1576 1250 TASK4 /, = TERMINATOR FOR EXPRESSIONS
1033 1577 0610 PROCESS /| = TERMINATOR FOR COMMANDS
1034 1600 0614 PC1 /C.R. = TERMINATOR FOR STRINGS
1035
1036 /S = FOR TDUMP: TERMINATES THE COMMAND.
1037 /EVALUATE AN EXPRESSION WHICH
1038 /TERMINATES WITH AN R-PAR, | OR C.R. AND
1039 /LEAVE THE RESULT IN FLAG AND IN FLAG.
1040 ECALL, 0 /RECURSIVE CALL TO "EVAL"
1041 TAD SORTCN /SAVE 'SORTCN', 'LASTOP', AND 'EFOP'
1042 PUSHA
1043 TAD LASTOP
1044 PUSHA
1045 TAD EFOP /SAVE FUNCTION CODE.
1046 PUSHA
1047 TAD ECALL /RETURN TO CALLING

```

1046	1611	4542	PUSHA	/ADDRESS AFTER NEXT POPJ
1047	1612	4545	GETC	/MOVE PAST EXTRA CHARACTER
1048	1613	3755	EVAL. DCA LASTOP	/EVALUATION CONTROLLER (CHECKPOINT ?)
1049	1614	4564	TESTC	/TEST CHARACTER AND IGNORE SPACES
1050	1615	5227	JMP ETERM1	/TERMINATOR
1051	1616	5332	JMP ENUM	/NUMBER
1052	1617	5343	JMP EFUN	/FUNCTION
1053	1620	4540	PUSHJ	/LETTER OF VARIABLE
1054	1621	1405	GETVAR	/FIND OR CREATE VARIABLE; ALSO SET PT1.
1055	1622	4564	OPNEXT; TESTC	/PT1=>ARG
1056	1623	5244	JMP ETERMN	/T
1057	1624	212	ECHOLST; 0212	/N=ERROR IN FORMAT
1058	1625	377	0377	/F
1059	1626	4566	ERROR4	/L = MISSING OPERATOR
1060	1627	1137	ETERM1; TAD CFRSX	/SET PT1.
1061	1630	3030	DCA PT1	/TO POINT TO ZERO
1062	1631	1111	TAD M2	/TEST FOR UNARY OPERATIONS
1063	1632	1054	TAD SORTCN	
1064	1633	7450	SNA	
1065	1634	5247	JMP ETERM	/CREATE DUMMY FOR UNARY MINUS
1066	1635	7001	IAC	
1067	1636	7650	SNA CLA	
1068	1637	5323	JMP ARGNXT	/IGNORE UNARY PLUS
1069	1640	1054	TAD SORTCN	/TEST FOR NULL PARENS.
1070	1641	1121	TAD M11	
1071	1642	7710	SPA CLA	
1072	1643	5364	JMP ELPAR	/MIGHT BE AN L-PAR.
1073	1644	4562	ETERMN; TSTLPR	
1074	1645	7410	SKP	
1075	1646	4566	ERROR4	/OPERATOR MISSING BEFORE PAREN
1076	1647	1054	ETERM; TAD SORTCN	/SET FROM "TESTC"="SORTC"
1077	1650	3024	DCA THISOP	
1078	1651	1024	TAD THISOP	
1079	1652	1121	TAD M11	
1080	1653	7700	SMA CLA	/END?
1081	1654	3024	DCA THISOP	/ "THISOP" EQUIV. TO END OF EXP.
1082	1655	1024	ETERM2; TAD THISOP	/COMPARE PRIORITIES
1083	1656	7041	CIA	
1084	1657	1055	TAD LASTOP	
1085	1660	7710	SPA CLA	
1086	1661	5310	JMP EPAR	/CONTINUE
1087	1662	1055	TAD LASTOP	/FIND OPERATION
1088	1663	7112	CLL RTR	
1089	1664	7012	RTR	
1090	1665	1331	TAD OPTABL	
1091	1666	3274	DCA FLOP	
1092	1667	1055	TAD LASTOP	
1093	1670	7640	SZA CLA	/TEST FOR END OF DATA INTO FLOATING AC.
1094	1671	4544	POPF	/GET LAST DATA
1095	1672	0044	FLAC	
1096	1673	4407	FINT	
1097	1674	0000	FLOP; 00	/ (FLOPR I PT1) == /
1098	1675	6525	FPUT I FLARGP	/SAVE RESULT
1099	1676	0000	FXIT	
1100	1677	1125	TAD FLARGP	

1101	1720	3030		DCA PT1	
1102	1721	1224		TAD THISOP	
1103	1722	1055		TAD LASTOP	/=0?
1104	1723	7657		SNA CLA	
1105	1724	5541		POPJ	/EXIT "EVAL"
1106	1725	1413		POPA	/GET PRIOR OP
1107	1726	3055		DCA LASTOP	
1108	1727	5255		JMP ETERM2	/COMPARE THIS OP
1109	1710	4562	EPAR,	TSTLPR	/TEST FOR SUB-EXPRESSION
1110	1711	7410		SKP	
1111	1712	5366		JMP EPAR2	/GO EVALUATE EXPRESSION
1112	1713	1055		TAD LASTOP	/CONTINUE READING THE EXPRESSION
1113	1714	4542		PUSHA	/SAVE "LASTOP",
1114	1715	1030		TAD PT1	
1115	1716	3320		DCA ,*2	
1116	1717	4543		PUSHF	/SAVE LAST ARGUMENT
1117	1727	1000			
1118	1721	1024		TAD THISOP	/MORE TO COME
1119	1722	3055		DCA LASTOP	
1120	1723	4545	ARGNXT,	GETC	/READ 1ST CHAR OF AN ARG.
1121	1724	4564		TESTC	/DO SPECIAL CHECK
1122	1725	5364		JMP ELPAR	/COULD BE LEFT PAREN
1123	1726	5332		JMP ENUM	/N
1124	1727	5343		JMP EFUN	/F
1125	1730	5220		JMP OPNEXT*2	/L
1126	1731	1430	OPTABL,	FGET I PT1	/BASE FOR OPERATION COMPUTATION
1127	1732	4543	ENUM,	PUSHF	/TO PROCESS A NUMBER,SAVE AC
1128	1733	2044		FLAC	
1129	1734	1125		TAD FLARGP	/SET POINTER AS FOR A VARIABLE,
1130	1735	3030		DCA PT1	
1131	1736	3036		DCA INSUB	/POINT TO 'GETC' AND USE CHAR
1132	1737	4531		JMS I FINPUT	/READ TEXT NUMBER => (PT1)
1133	1740	4544		POPF	/RESTORE THE AC
1134	1741	2044		FLAC	
1135	1742	5222		JMP OPNEXT	/CONTINUE
1136	1743	3056	EFUN,	DCA EFOP	/SET CODE
1137	1744	4545		GETC	/READ FUNCTION NAME.(1,2,OR 3 LETTERS)
1138	1745	4564		TESTC	/***** SEPARATES FILE BECAUSE F DIGIT
1139	1746	5355		JMP EFUN2	/*****
1140	1747	5771		JMP I PFNUM	/*****
1141	1750	7000		NOP	/*****
1142	1751	1056		TAD EFOP	/*****
1143	1752	7104		CLL RAL	/MISH=MASH HASH CODE
1144	1753	1066		TAD CHAR	
1145	1754	5343		JMP EFUN	
1146	1755	4562	EFUN2,	TSTLPR	
1147	1756	4566		ERROR4	/MUST BE FOLLOWED BY PARENS TO SET ARGUMENT
1148	1757	4201		JMS ECALL	/CALL "EVAL" TO COMPUTE ARGUMENT
1149	1760	1413		POPA	/BRANCH ON FUNCTION CODE;RETURN VIA EFUN3;
1150	1761	4547		SORTJ	
1151	1762	2166			
1152	1763	6205			
1153	1764	4562	ELPAR,	TSTLPR	/LEFT PAREN OR FELL THROUGH FUNCTION TABLE
1154	1765	4566		ERROR4	/DOUBLE OPERATORS OR ILLEGAL FUNCTION NAME;
1155	1766	4201	EPAR2,	JMS ECALL	/EVALUATE NESTED EXPRESSION

```

1156 1757 2213   ISE PDLXR           /DUMP EXTRA ARG.
1157 1772 5536   JMP I EFUN3
1158 1771 6311   PFNUM, FNUM          /*****
1159 1772 1772   TERMS: /TERMINATOR TABLE FOR 'EVAL' AND 'GETVAR'
1160 1772 2240   240 /SPACE 0
1161 1773 2253   253 /* 1
1162 1774 2255   255 /- 2
1163 1775 2257   257 // 3
1164 1776 2252   252 /* 4
1165 1777 2336   336 /UP ARR 5
1166 2200 2250   250 /* 6 L=PARS
1167 2201 2333   333 /* 7
1168 2222 2274   274 /* 10
1169 2203 2251   251 /* 11 R=PARS
1170 2224 2335   335 /* 12
1171 2225 2276   276 /* 13
1172 2206 2254   254 /* 14
1173 2207 2273   273 /* 15
1174 2210 2215   215 /C.R. 16
1175 2211 2275   275 /* TO END GETARG FROM 'SET'
1176
1177 2212 4543   /TWO MINOR FUNCTIONS
1178 2213 2405   XSGN, PUSHF          /TAKE SIGN#1 OF FLARG
1179 2214 4544   POPF
1180 2215 2244   FLAC
1181 2216 1233   XABS, TAD FLARG*1    /TAKE ABSOLUTE VALUE OF FLAG
1182 2217 7710   SPA CLA             /SKIP TO CONTINUE
1183 2220 4451   JMS I MINSKI        /NEGATE THE FLOATING AC
1184
1185 2221 4407   /CONTINUATION OF FUNCTION CALLS.
1186 2222 7000   EFUN3, FINT
1187 2223 6232   FNOR                /NORMALIZE FUNCTION RETURN
1188 2224 2000   FPUT FLAG           /SAVE FUNCTION VALUE
1189 2225 1125   EXIT
1190 2226 3230   TAD FLARGP          /SET POINTER
1191 2227 4251   DCA PT1
1192 2230 5631   JMS PARTEST
1193 2231 1622   JMP I ,+1           /FUNCTION RETURN IS OK
1194
1195 2232 2000   FLAG, 0             /DATA TEMPORARY STORAGE
1196 2233 2000   0
1197 2234 2000   0
1198 2235 2000   0
1199 2236 2003   PS, 3
1200 2237 2202   LPRST, 0           /SKIP IF LEFT PAREN. = 'YSTLPR'
1201 2240 1054   TAD SORTCN
1202 2241 1121   TAD M11
1203 2242 7722   SMA CLA
1204 2243 5637   JMP I LPRST
1205 2244 1054   TAD SORTCN
1206 2245 1120   TAD M5
1207 2246 7740   SMA SZA CLA
1208 2247 2237   ISE LPRST
1209 2250 5637   JMP I LPRST
1210 2251 2000   PARTEST,C        /TEST THE PAREN MATCHINGS

```

1211	2052	1413	POPA	/RESTORE LAST OPERATION
1212	2053	3255	CALL LASTOP	
1213	2054	1236	TAD P3	/+3 TO COMPARE CODES
1214	2055	1413	POPA	/GET LAST PAREN CODE.
1215	2056	7041	CIA	/CHECK FOR PAREN MATCH.
1216	2057	1254	TAD SORTCN	/(STILL GET FROM THE LAST "EVAL")
1217	2060	7640	SZA CLA	/SKIP IF MATCH
1218	2061	4566	ERROR4	/PAREN ERROR
1219	2062	4545	GETC	/MOVE PAST R=PAR
1220	2063	5651	JMP I PARTEST	
1221			/THE DELETE A LINE ROUTINE	
1222	2064	1020	XDELETE,0	/UNCHAIN A LINE AND RECOVER THE SPACE.
1223	2065	6002	IOF	/PROTECT POINTER CHANGES FROM INTERRUPTIONS
1224	2066	4555	FINDLN	/SETS "THISLN" AND "LASTLN".
1225	2067	5664	JMP I XDELETE	/ALREADY GONE
1226	2070	2026	ISE DEBGSW	/DISABLE TRACE
1227	2071	4545	GETC	/MEASURE LENGTH
1228	2072	1066	TAD CHAR	
1229	2073	1116	TAD MCR	
1230	2074	7640	SZA CLA	
1231	2075	5271	JMP ,=4	
1232	2076	1017	TAD AXOUT	/SAVE LAST ADDRESS
1233	2077	7040	CMA	
1234	2100	1023	TAD THISLN	
1235	2101	3057	DCA CNTR	/LENGTH < 0
1236	2102	1133	TAD CFRS	/IT IS ILLEGAL TO DELETE THE FIRST LINE
1237	2103	7041	CIA	
1238	2104	1023	TAD THISLN	
1239	2105	7650	SNA CLA	
1240	2106	5177	JMP START	/JUST IGNORE SUCH COMMANDS
1241	2107	7000	CDP T	/CHANGE DATA FIELD TO TEXT,(X=MEM)
1242	2110	1423	TAD I THISLN	/DISCONNECT
1243	2111	3425	DCA I LASTLN	
1244	2112	1133	TAD CFRS	/START LIST AT TOP
1245	2113	3071	DCA T2	/EXAMINATION ADDRESS
1246	2114	1471	TAD I T2	/GET THE NEXT ADDR.
1247	2115	7450	SNA	/TEST FOR END
1248	2116	5331	JMP DONE	/YES=WRAP UP ALL.
1249	2117	3032	DCA T1	/SAVE NEXT ADDRESS.
1250	2120	1023	TAD THISLN	/COMPARE LINE POSITIONS
1251	2121	7141	CIA CLL	
1252	2122	1032	TAD T1	
1253	2123	7630	SZL CLA	/SKIP IF THISLN > X
1254	2124	1057	TAD CNTR	/CHANGE (X) TO ACCOUNT FOR
1255	2125	1032	TAD T1	/GARBAGE COLLECTION.
1256	2126	3471	DCA I T2	
1257	2127	1032	TAD T1	/GET NEXT
1258	2130	5313	JMP DOK	
1259			/GARBAGE COLLECTION	
1260	2131	7040	DONE, CMA	/BACKUP L FOR XR
1261	2132	1023	TAD THISLN	
1262	2133	3011	DCA XRT	
1263	2134	1057	TAD CNTR	/SETUP END OF HOSE
1264	2135	7040	CMA	
1265	2136	1023	TAD THISLN	

1266	37	3012	DCA XRT2	
1267	2147	1057	TAD CNTR	/CORRECT END OF BUFFER POINTER,
1268	2141	1060	TAD BUFR	
1269	2142	1060	DCA BUFR	
1270	2143	1010	TAD AXIN	/COMPUTE COUNT
1271	2144	7040	CMÄ	
1272	2145	1012	TAD XRT2	
1273	2146	3032	DCA T1	
1274	2147	1010	TAD AXIN	
1275	2150	1057	TAD CNTR	
1276	2151	3010	DCA AXIN	
1277	2152	1412	TAD I XRT2	/SIPHON LOWER PART,
1278	2153	3411	DCA I XRT	
1279	2154	2032	ISZ T1	
1280	2155	5352	JMP ,=3	
1281	2156	5265	JMP XDELETE+1	/RESET 'LASTLN', 'THISLN', AND DATA FIELD'
1282	2157	0000	0	/READ IN A CHARACTER SUBR. = "READC"
1283	2160	4464	JMS I INDEV	
1284	2161	3066	DCA CHAR	
1285	2162	4550	SORTC	/LINEFEED OR RUBOUT?
1286	2163	1623	ECHOLST=1	
1287	2164	5757	JMP I CHIN	/YES
1288	2165	4551	PRINTC	/ECHO THE INPUT
1289	2166	5757	JMP I CHIN	
1290		2167	FNTABL=.	
1291	2167	2533	2533	/ABS
1292	2170	2650	2650	/SGN
1293	2171	2636	2636	/ITR
1294	2172	2565	2565	/DIS
1295	2173	2630	2630	/RAN
1296	2174	2517	2517	/ADC
1297	2175	2572	2572	/ATN
1298	2176	2624	2624	/EXP
1299	2177	2625	2625	/LOG
1300	2200	2654	2654	/SIN
1301	2201	2575	2575	/COS
1302	2202	2702	2702	/SQT
1303	2203	2631	2631	/NEW
1304	2204	0330	0330	/FX
1305	2205	0332	0332	/FZ
1306				/*****
1307	2206	4564	/ERASE SINGLE LINES, GROUPS, OR VARIABLES	
1308	2207	5241	ERASE: TESTC	/TEST THE SECOND WORD, IF ANY.
1309	2210	5224	JMP ERVX	/ERASE VARIABLES
1310	2211	5215	JMP ERL	/LINES OR GROUPS
1311	2212	1066	JMP ,+4	/ERROR
1312	2213	1112	TAD CHAR	/ALL TEXT
1313	2214	7440	TAD MINUSA	
1314	2215	4566	SZÄ	
1315	2216	1135	ERROR3	/BAD ARG FOR ERASE;
1316	2217	3060	TAD ENDT	/ERASE ALL TEXT **
1317	2220	3533	DCA BUFR	
1318	2221	1060	DCA I CFRS	/(X=MEM)
1319	2222	3031	TAD STARTV	/ERASE VARIABLES **
1320	2223	5177	DCA LASTV	
			JMP START	/POINTERS MAY BE DIFFERENT NOW.

1321	2224	4554	ERL,	GETLN	/ERASE LINES.
1322	2225	1060		TAD BUFR	/PROTECT REST OF TEXT.
1323	2226	3317		DCA AXIN	
1324	2227	4565	ERG,	DELETE	/EXTRACT ONE LINE
1325	2232	2023		ISE THISLN	
1326	2231	1065		TAD NAGSW	
1327	2232	7700		SMA CLA	
1328	2233	1423		TAD I THISLN	/(X=MEM)
1329	2234	4563		TSTGRP	/SKIP IF G(AC) = G(LINENO)
1330	2235	5221		JMP ERV	
1331	2236	1423		TAD I THISLN	/(X=MEM)
1332	2237	3067		DCA LINENO	
1333	2240	5227		JMP ERG	
1334	2241	1060	ERVX,	TAD STARTV	/INIT VARIABLES MAY BE INDIRECT COMMAND
1335	2242	3031		DCA LASTV	
1336	2243	5541		POPJ	
1337				/ROUTINE CALLED VIA "FINDLN":	
1338				/SEARCH FOR A GIVEN LINE I.D. = ["LINENO"]	
1339				/1ST RETURN IF NOT FOUND,	
1340				/2ND IF FOUND,	
1341				/"THISLN" = FOUND LINE OR NEXT LARGER,	
1342				/"LASTLN" = LESSER AND/OR LAST,	
1343				/"TEXTP" IS SET	
1344	2244	0000	XFIND,	0	
1345	2245	1133		TAD CFRS	/INITIALIZE POINTERS TO FIRST LINE
1346	2246	3025		DCA LASTLN	
1347	2247	1133		TAD CFRS	
1348	2250	3023	FINDN,	DCA THISLN	/SAVE THIS ONE
1349	2251	1023		TAD THISLN	
1350	2252	3011		DCA XRT	
1351	2253	1067		TAD LINENO	
1352	2254	7141		CLL CMA IAC	/CLEAR LINK AND NEGATE LINENO,
1353	2255	1411		TAD I XRT	/LINENO=0 WILL ALSO BE FOUND(X=MEM)
1354	2256	7450		SNA	
1355	2257	2244		ISE XFIND	/*****
1356	2260	7630		SZL CLA	
1357	2261	5267		JMP FEND3	/PAST IT.
1358	2262	1023		TAD THISLN	/MOVE POINTERS
1359	2263	3025		DCA LASTLN	
1360	2264	1423		TAD I THISLN	/END OF TEXT? (X=MEM)
1361	2265	7440		SZA	
1362	2266	5250		JMP FINDN	/NOT YET
1363					/*****
1364					/*****
1365	2267	1023	FEND3,	TAD THISLN	/1ST RETURN = NOT FOUND
1366	2270	7001		IAC	
1367	2271	3017		DCA AXOUT	/SET "TEXTP".
1368	2272	3020		DCA XCT	
1369	2273	5644		JMP I XFIND	
1370	2274	1000	UTRA,	0	/UNPACK CHARACTER, = "GETC"
1371	2275	4330		JMS GET1	
1372	2276	7710	UTE,	SPA CLA	/NORM & EXTEND
1373	2277	1006		TAD C100	/300=337 & 340=376
1374	2300	1357		TAD M137	/240=276 & 200=236
1375	2301	1066		TAD CHAR	

1376	2322	745J	SNÄ	
1377	2323	5316	JMP UTX	/?" FOUND
1378	2324	1275	TAD P337	
1379	2325	3366	UTQ, DCA CHAR	
1380	2326	1226	TAD DEBGSW	
1381	2327	1120	TAD DMPSW	
1382	2310	7650	SNÄ CLA	/PRINT ONLY IF BOTH ARE ZERO.
1383	2311	4551	PRINTC	
1384	2312	5674	JMP I UTRA	
1385	2313	4330	EXTR, JMS GET1	
1386	2314	7040	CMA	
1387	2315	5276	JMP UTE	
1388	2316	1226	UTX, TAD DEBGSW	/TEST FOR TRACE=ENABLED
1389	2317	7640	SZÄ CLA	
1390	2320	5326	JMP ,+6	
1391	2321	1100	TAD DMPSW	/FLIP THE TRACE FLOP
1392	2322	7650	SNÄ CLA	
1393	2323	7001	IAC	
1394	2324	3100	DCA DMPSW	
1395	2325	5275	JMP UTRA+1	/GET NEXT CHARACTER INSTEAD.
1396	2326	1110	TAD P277	/TRACE DISABLED = RETURN "?"
1397	2327	5305	JMP UTO	
1398	2330	0000	GET1, 0	/UNPACK 6=BITS
1399	2331	2020	ISZ XCT	/STARTS=0
1400	2332	5345	JMP GET3	
1401	2333	1021	TAD GTEM	
1402	2334	1122	GEND, AND P77	
1403	2335	3066	DCA CHAR	/SAVE
1404	2336	1066	TAD CHAR	
1405	2337	1103	TAD M77	
1406	2340	7650	SNÄ CLA	
1407	2341	5313	JMP EXTR	/EXTENDED
1408	2342	1066	TAD CHAR	
1409	2343	1356	TAD M40	
1410	2344	5730	JMP I GET1	
1411	2345	1417	GET3, TAD I AXOUT	/(X=MEM)
1412	2346	3021	DCA GTEM	
1413	2347	7040	CMA	
1414	2350	3020	DCA XCT	
1415	2351	1021	TAD GTEM	
1416	2352	7112	RTR CLL	
1417	2353	7012	RTR	
1418	2354	7012	RTR	
1419	2355	5334	JMP GEND	
1420	2356	7740	M40, =40	
1421	2357	7641	M137, =137	
1422	2360	0000	XENDLN, 0	/TERMINATE THE BUFFERED LINE = "ENDLN"
1423	2361	7000	ODF T	/(X=MEM)
1424	2362	1425	TAD I LASTLN	/SAVE OLD POINTER
1425	2363	3460	DCA I BUFR	
1426	2364	1060	TAD BUFR	/POINT TO NEW LAST LINE
1427	2365	3425	DCA I LASTLN	
1428	2366	1061	TAD QADD	/CHECK FOR EXTRA INFO
1429	2367	7440	SZÄ	
1430	2370	3410	DCA I AXIN	

1431	2371	1210	TAD AXIN	/COMPUTE NEW END OF BUFFER
1432	2372	7331	I/O	
1433	2373	3360	DCA BUFR	
1434	2374	1060	TAD STARTV	/RESET VARIABLE LIST (X-MEM)
1435	2375	3031	DCA LASTV	
1436	2376	5760	JMP I XENDLN	
1437		2377	TLIST3=	/LITERAL TERMINATORS
1438	2377	1251	TASK4	/"
1439	2400	614	PCI	/C.R. = AUTOMATIC QUOTE MATCH
1440		2401	INFIX="	/DATA CONTROL CHARACTERS
1441	2401	6202	FLINTP*2	/LEFT ARROW = KILL
1442	2402	757	INPUT+1	/RUBOUT = IGNORE
1443	2403	757	INPUT+1	/L.F. = IGNORE
1444	2404	6250	ENDFI*5	/ALT MODE = EXIT
1445	2405	0001	FLTONE;	/(NO RELATIVE REFERENCES)
1446	2406	0000	FLTZER;	
1447	2407	0000		
1448	2410	0000		
1449	2411	0000		
1450	2412	0000		
1451	2413	7766	M12,	=12 /DECIMAL CONVERSION FACTOR FOR "PRNT"
1452	2414	0000	I33,	0 /NO=INTERRUPT INPUT ROUTINE
1453	2415	6031	KSP	
1454	2416	5215	JMP ,=1	
1455	2417	6036	KR9	
1456	2420	0106	AND P177	/IGNORE PARITY BIT
1457	2421	7450	SNA	
1458	2422	5215	JMP ,=5	
1459	2423	1123	TAD C200	
1460	2424	5614	JMP I I33	
1461	2425	0000	XPRNT,	0 /PRINT A LINE NUMBER = "PRNTLN"
1462	2426	1067	TAD LINENO	
1463	2427	4557	RTL6	
1464	2430	0122	AND P77	
1465	2431	4242	JMS PRNT	/TWO DIGIT "PART" NUMBER
1466	2432	1102	TAD PER	
1467	2433	4551	PRINTC	/PERIOD FOR SEPARATION
1468	2434	1067	TAD LINENO	
1469	2435	4242	JMS PRNT	/TWO DIGIT "STEP" NUMBER.
1470	2436	1356	TAD M140	
1471	2437	3066	DCA CHAR	/SAVE SPACE IN CHAR,
1472	2440	4551	PRINTC	/PRINT TRAILING SPACE
1473	2441	5625	JMP I XPRNT	
1474		0032	VAL=T1	
1475	2442	0000	PRNT,	0 /PRINT TWO DECIMAL DIGITS
1476	2443	0106	AND P177	
1477	2444	3032	DCA VAL	
1478	2445	1113	TAD C260	
1479	2446	3033	DCA T3	
1480	2447	5252	JMP ,+3	
1481	2450	2333	ISZ T3	
1482	2451	3032	XYZ,	DCA VAL
1483	2452	1032	TAD VAL	
1484	2453	1213	TAD M12	
1485	2454	7500	SMA	

1486 55 5253
 1487 2456 7240
 1488 2457 1233
 1489 2460 4551
 1490 2461 1232
 1491 2462 1113
 1492 2463 4551
 1493 2464 5642
 1494 2465 7222
 1495 2466 7450
 1496 2467 1266
 1497 2470 1116
 1498 2471 7450
 1499 2472 5276
 1500 2473 1277
 1501 2474 4463
 1502 2475 5665
 1503 2476 1277
 1504 2477 4463
 1505 2520 1276
 1506 2521 5274
 1507 2522 2000
 1508 2523 1110
 1509 2524 7241
 1510 2525 1266
 1511 2526 7450
 1512 2527 1352
 1513 2510 1101
 1514 2511 7450
 1515 2512 5755
 1516 2513 1353
 1517 2514 3271
 1518 2515 1271
 1519 2516 2354
 1520 2517 1356
 1521 2520 7440
 1522 2521 1354
 1523 2522 7650
 1524 2523 5332
 1525 2524 1271
 1526 2525 1122
 1527 2526 7440
 1528 2527 4335
 1529 2530 7200
 1530 2531 5702
 1531 2532 1122
 1532 2533 4335
 1533 2534 5324
 1534 2535 2000
 1535 2536 2262
 1536 2537 5357
 1537 2540 1261
 1538 2541 3410
 1539 2542 3261
 1540 2543 1213

OUT.

OUTX.
OUTCR.

PACBUF.

PA1.

PACX.

ESCA.

PCK1.

JMP XYZ-1
 CLÄ
 TAD T3
 PRINTC
 TAD VAL
 TAD C260
 PRINTC
 JMP I PRNT
 0
 SNÄ
 TAD CHAR
 TAD MCR
 SNÄ
 JMP OUTCR
 TAD CCR
 JMS I OUTDEV
 JMP I OUT
 TAD CCR
 JMS I OUTDEV
 TAD CLF
 JMP OUTX-1
 0
 TAD P277
 CIÄ
 TAD CHAR
 SNÄ
 TAD P40
 TAD M100
 SNÄ
 JMP I RUBIT
 TAD P377
 DCA T2
 TAD T2
 AND C140
 TAD M140
 SZÄ
 TAD C140
 SNÄ CLA
 JMP ESCA
 TAD T2
 AND P77
 SZÄ
 JMS PCK1
 CDF P
 JMP I PACBUF
 TAD P77
 JMS PCK1
 JMP PA1
 0
 ISÄ XCTIN
 JMP ROT
 TAD QADD
 DCA I AXIN
 DCA QADD
 TAD PDLXR

/OUTPUT A CHARACTER = "PRINTC"
 /USE (AC) OR (CHAR)

/PACK A CHARACTER = "PACKC"

/CHANGE 277 TO 337

/TEST FOR RUBOUT.

/SAVE INPUT ITEM
 /SO THAT QUESTION DOESN'T MAKE
 /CHAR LOOK LIKE A LEFT-ARROW

/DATA WORD.

/340-377 AND 200-237
 /240-337

/IGNORE 300

/(X-MEM)

/=0 TO START

/(X-MEM)

/CLEAR PACKING WORD
 /CHECK FOR OVERFLOW

```

1541 2544 7141 CMA IAC CLL
1542 2545 1005 TAD P13 /RESERVATIONS FOR PUSH-DOWN LIST
1543 2546 1010 TAD AXIN
1544 2547 7620 SNL CLA
1545 2550 5735 JMP I PCK1
1546 2551 4566 ERROR2 /FULL BUFFER
1547 2552 1040 P40, 40
1548 2553 1377 P377, 377
1549 2554 140 C140, 140
1550 2555 3004 RUBIT, RUB1
1551 2556 7640 M140, =140
1552 2557 4557 ROT, RTL6 /(EAE)
1553 2560 3161 DCA QADD
1554 2561 7040 CMA
1555 2562 3062 DCA XCTIN
1556 2563 5735 JMP I PCK1
1557 /
1558 /PART OF INTERFACE TO FLD1 TO ALLOW
1559 /GETTING OF CHARS FROM TEXT
1560 /
1561 2564 4545 CGETX, GETC /*****
1562 2565 1066 TAD CHAR /*****
1563 2566 6213 6213 /CIF CDF 10/*****
1564 2567 5770 JMP I ,+1 /*****
1565 2570 1137 CGETRET /*****
1566 2571 4566 ERRFIL, ERROR4 /*****
1567 2572 4540 LM, PUSHJ /*
1568 2573 1612 EVAL=1 /*
1569 2574 4453 JMS I INTEGER /*
1570 2575 6212 6212 /*
1571 2576 5777 JMP I ,+1 /*
1572 2577 1402 LMAKE /*
1573 /USED BY BK
1574 2600 *2600
1575 /INTERRUPT PROCESSOR,
1576 2600 0 SAVAC, 0 /CONTENTS OF AC
1577 2601 0000 SAVLK, 0 /CONTENTS OF LINK
1578 2602 7575 MBREAK, =203 /CONTROL=C
1579 2603 3200 INTRPT, DCA SAVAC /SAVE WORKING DATA
1580 2604 7010 RAR
1581 2605 3201 DCA SAVLK
1582 2606 6041 TSP /GIVE OUTPUT PRIORITY
1583 2607 5225 JMP KINT
1584 2610 6042 TCF
1585 2611 3016 DCA TELS# /TURN OFF THE IN-PROGRESS FLAG,
1586 2612 1665 TAD I OPTRI
1587 2613 7450 SNA
1588 2614 5225 JMP KINT /DONE
1589 2615 6044 TPC /TYPE NEXT,
1590 2616 3016 DCA TELS# /CLEAR AC AND TURN ON THE FLAG,
1591 2617 3665 DCA I OPTRI /ZERO OUT THE DATA AREA
1592 2620 1265 TAD OPTRI
1593 2621 7001 IAC
1594 2622 107 AND P17
1595 2623 1263 TAD OPTR0

```

1596 4 3265
 1597 2625 6031
 1598 2626 5246
 1599 2627 6036
 1600 2630 126
 1601 2631 7450
 1602 2632 5246
 1603 2633 1123
 1604 2634 3262
 1605 2635 1262
 1606 2636 1202
 1607 2637 7650
 1608 2640 5340
 1609 2641 1034
 1610 2642 7640
 1611 2643 4566
 1612 2644 1262
 1613 2645 3034
 1614 2646 6131
 1615 2647 5253
 1616 2650 6135
 1617 2651 7200
 1618 2652 3261
 1619
 1620
 1621
 1622 2653 6244
 1623 2654 1201
 1624 2655 7104
 1625 2656 1200
 1626 2657 6001
 1627 2660 5400
 1628 2661 0000
 1629 2662 0000
 1630 2663 3120
 1631 2664 3120
 1632 2665 3120
 1633 2666 0000
 1634 2667 1034
 1635 2670 7450
 1636 2671 4574
 1637 2672 3276
 1638 2673 3034
 1639 2674 1276
 1640 2675 5666
 1641 2676 0000
 1642 2677 3266
 1643 2700 6001
 1644 2701 1664
 1645 2702 7640
 1646 2703 4574
 1647 2704 6002
 1648 2705 1016
 1649 2706 7640
 1650 2707 5314

DCA OPTRI
 KINT. KSF /CHECK FOR KEYBOARD FIRST
 JMP EXIT
 KR9 /READ BUFFER AND CLEAR FLAG TO FETCH NEXT
 AND P177 /IGNORE BLANK AND L-T AND PARITY BIT.
 SNA
 JMP EXIT
 TAD C200
 DCA SIN
 TAD SIN
 TAD MBREAK /MANUAL STOP?
 SNA CLA
 JMP RECOVR
 TAD INBUF /ANY SPACE?
 SZA CLA
 ERROR2 /WILL WAIT FOR OUTPUT BUFFER
 TAD SIN
 DCA INBUF /SAVE INPUT
 CLSK /*****
 JMP NOCLK /*****
 CLSA /*****
 CLA /*****
 DCA CLKFLG /*****
 /
 /KW12 CLOCK INTERRUPT ROUTINE
 /
 NOCLK: RMP
 TAD SAVLK
 RAL CLL
 TAD SAVAC
 ION
 EXITJ: JMP I 0 /MODIFIED FOR PDP-5
 CLKFLG: 0 /***** SET TO 0 EVERY INTERRUPT
 SIN, 0
 OPTRO: IOBUF /OUTPUT POINTERS
 OPTRO: IOBUF /VARS
 OPTRI: IOBUF
 X133: 0 /VIA (INDEV)
 TAD INBUF /ANY INPUT?
 SNA /***** REFRESH SCOPE WHILE WAITING
 JMS I PWAIT /***** FOR INPUT
 DCA XOUTL
 DCA INBUF /CLEAR INPUT BUFFER
 TAD XOUTL
 JMP I X133
 XOUTL: 0 /VIA (OUTDEV)
 DCA X133 /SAVE CURRENT CHARACTER,
 ION /BE SURE INTERRUPT IS ON,
 TAD I OPTRO /ANY ROOM?
 SZA CLA /A CHARACTER IS NON-ZERO
 JMS I PWAIT /***** REFRESH SCOPE
 IOF
 TAD TELS W /IN PROGRESS?
 SZA CLA
 JMP ,*5

1651 2710 1266
 1652 2711 6146
 1653 2712 3316
 1654 2713 5323
 1655 2714 1266
 1656 2715 3664
 1657 2716 1264
 1658 2717 7001
 1659 2720 1107
 1660 2721 1263
 1661 2722 3264
 1662 2723 6001
 1663 2724 5676
 1664
 1665 2725 3326
 1666 2726 1000
 1667 2727 7240
 1668 2730 1326
 1669 2731 3067
 1670 2732 6001
 1671 2733 1016
 1672 2734 7640
 1673 2735 5333
 1674 2736 6002
 1675 2737 5342
 1676 2740 1123
 1677 2741 3067
 1678
 1679 2742 1105
 1680 2743 3057
 1681 2744 7040
 1682 2745 1263
 1683 2746 3010
 1684 2747 2016
 1685 2750 7000
 1686 2751 3410
 1687 2752 2057
 1688 2753 5351
 1689 2754 3034
 1690 2755 1263
 1691 2756 3265
 1692 2757 1263
 1693 2760 3264
 1694 2761 7040
 1695 2762 6046
 1696 2763 1101
 1697 2764 4551
 1698 2765 4553
 1699 2766 2022
 1700 2767 1422
 1701 2770 7450
 1702 2771 5377
 1703 2772 3067
 1704 2773 1101
 1705 2774 4551

```

TA X133 /NO
TLS /TYPE CHARACTER,
DCA TELS. /SET IN-PROGRESS FLAG,
JMP ,+12 /RETURN
TAD X133 /SEND DATA
DCA I OPTRO
TAD OPTRO /SET POINTERS
IAC
AND P17
TAD OPTRO
DCA OPTRO
ION
JMP I XOUTL

/ERROR RECOVERY PROCEEDURE
ERROR5; DCA ,+1 /ERROR CALLED FROM A TABLE
ERR2, 0 /LIMIT EXCEEDED
CLA CMA /COMPUTE CALLING ADDRESS (ALSO "SPACE")
TAD ERR2 /AND USE IT AS ERROR NUMBER.
DCA LINENO /SAVE ERROR CODE.
ION / (JMP,+4) = FOR DEBUGGING
TAD TELS. /WAIT FOR OUTPUT TO FINISH
SZC CLA
JMP ,+2
IOF /DISABLE INTERRUPT FOR INITIALIZATIONS
JMP ,+3
RECOVR; TAD C200 /SAVE ERROR NUMBER
DCA LINENO

/****
TAD M20 /SETUP INIT COUNT
DCA CNTR
CMA
TAD OPTRO
DCA AXIN /INIT I/O BUFFERS.
ISZ TELS. /*
CDF / (X=MEM RESET)
DCA I AXIN
ISZ CNTR
JMP ,+2
DCA INBUF /INIT KEY=BUFR.
TAD OPTRO /INIT TTY POINTERS.
DCA OPTRI
TAD OPTRO
DCA OPTRO
RECOVX; CMA /PREPARE A STOP BIT FOR TTY
TLS /AND RAISE FLAG. (NOP) = FOR DEBUGGING
TAD P7700 /MAKE A "?".
PRINTC /AND TURN ON THE INTERRUPT
PRNTLN /PRINT ERROR NUMBER AND,
ISZ PC
TAD I PC /UNLESS IT IS ZERO. (X=MEM)
SNA
JMP ,+6
DCA LINENO
TAD P7700
PRINTC /PRINT ATSIGN

```


1706	3075	4551	PRINTC	/PRINT SPACE	IN AND
1707	3076	4553	PRNTLN	/PRINT LINE OF ERROR.	
1708	3077	4577	TAD CCR		
1709	3078	4551	PRINTC		
1710	3079	1126	TAD PTCH	/RESET "READC"	
1711	3080	3152	DCA RDIV	/IF AN ERROR OCCURS.	
1712	3083	5177	JMP START	/INTERRUPT WILL BE RE-ENABLED SOON.	
1713			/CHARACTER REMOVAL ROUTINE		
1714	3084	1062	RUB1, TAD XCTIN	/RUBOUT ONE LETTER	
1715	3085	7640	SZA CLA		
1716	3086	5214	JMP ,+6		
1717	3087	1010	TAD AXIN		
1718	3088	7041	CIA		
1719	3089	1027	TAD PACKST		
1720	3090	9700	SMA CLA	/TEST NULL LINE	
1721	3091	5641	JMP I RUB5		
1722	3092	1251	TAD SPLAT	/FOR A RUBOUT ACKNOWLEDGEMENT	
1723	3093	4551	PRINTC		
1724	3094	1010	TAD AXIN		
1725	3095	3071	DCA T2		
1726	3096	7000	ODF T	/(X=MEM)	
1727	3097	2062	ISZ XCTIN	/TEST HALF	
1728	3098	5242	JMP RUB2		
1729	3099	1471	TAD I T2	/"ADD" IS FULL.	
1730	3100	2122	AND P77		
1731	3101	1103	TAD M77		
1732	3102	7640	SZA CLA	/TEST FOR EXTEND	
1733	3103	5237	JMP RUB4		
1734	3104	7040	RUB3, CMA	/SET SWITCH	
1735	3105	3062	DCA XCTIN		
1736	3106	7040	CMA	/BACKUP POINTER	
1737	3107	1010	TAD AXIN		
1738	3108	3010	DCA AXIN		
1739	3109	1471	TAD I T2	/RESET ADD	
1740	3110	3101	AND P7700		
1741	3111	3061	RUB4, DCA QADD		
1742	3112	5641	JMP I RUB5		
1743	3113	2530	RUB5, PACX		
1744	3114	1471	RUB2, TAD I T2	/CHECK FOR EXTENDED	
1745	3115	1101	AND P7700		
1746	3116	1006	TAD C100		
1747	3117	7640	SZA CLA		
1748	3118	5230	JMP RUB3		
1749	3119	3471	DCA I T2	/SAVE CORRECTION	
1750	3120	5231	JMP RUB3+1		
1751	3121	334	SPLAT, 334		
1752			/SYMBOL TABLE TYPEOUT ROUTINE		
1753	3122	1060	TDUMP, TAD STARTV	/INIT POINTER FOR SYMBOL DUMP,(X=MEM)	
1754	3123	3030	DCA PT1		
1755	3124	1031	TAD LASTV	/TEST FOR END OF LIST	
1756	3125	7041	CIA		
1757	3126	1030	TAD PT1		
1758	3127	7650	SNA CLA		
1759	3128	5541	POPJ		
1760	3129	1430	TAD I PT1	/GET THE VARIABLE	

1761	3262	3316	DCA OP*1	/(DCA I (4)=FOR(X=MEM))SAVE NAME
1762	3263	3315	TAD OP	/SETUP UNPACK POINTERS
1763	3264	3317	DCA AXOUT	
1764	3265	3323	DCA XCT	
1765	3266	4545	GETC	/READ AND PRINT "XX("
1766	3267	4551	PRINTC	
1767	3270	4545	GETC	
1768	3271	4551	PRINTC	
1769	3272	4545	GETC	
1770	3273	4551	PRINTC	
1771	3274	2030	ISZ PT1	
1772	3275	1430	TAD I PT1	/PRINT SUBSCRIPT TO 99
1773	3276	4714	JMS I PRNT2	
1774	3277	4545	GETC	/PRINT ")"
1775	3100	4551	PRINTC	
1776	3101	2030	ISZ PT1	
1777	3102	4407	FINT	/PICK UP VALUE
1778	3103	4430	FGET I PT1	
1779	3104	4020	FXIT	
1780	3105	4530	JMS I FOUTPUT	/PRINT VALUE
1781	3106	1077	TAD CCR	
1782	3107	4551	PRINTC	
1783	3110	1070	TAD GINC	
1784	3111	1111	TAD M2	
1785	3112	1030	TAD PT1	
1786	3113	5253	JMP TDUMP*1	
1787	3114	2442	PRNT2, PRNT	
1788	3115	3115	OP,	/ (X=MEM)
1789	3116	0000		/ (X=MEM)
1790	3117	5051		/(THESE GO IN 10005 FOR X=MEM)
1791			/OUTPUT CHARACTER BUFFER (ADDRESS IS A MULTIPLE OF 20)	
1792		3120	IOBUF=3120	
1793		3140	COMEIN=IOBUF*20 /COMMAND = INPUT BUFFER	
1794		3206	COMEOUT=COMEIN*46	
1795		3206	*COMEOUT	
1796	3206	0000	FRST, 0	/TEXT POINTER
1797	3207	0000	0000	/DUMMY LINE NO.
1798	3210	0340	0340	/*****
1799	3211	0617	0617	/FO
1800	3212	0301	0301	/CA
1801	3213	1455	1455	/*****
1802	3214	6162	FRSTX, 6162	/*****
1803	3215	7715	7715	/DUMMY C.R.
1804			/TO SAVE TEXT ,SAVE C(BUFR), C(LASTV), AND C(FRST TO C(BUFR))	
1805			/WITH ODT=JR46, THE TAPES MAY BE TOGETHER WITH	
1806			/THE SYMBOLIC DUMP LAST I FOCAL + FLOAT + DIALOG .	
1807			/LOADING THE LAST SECTION MAY BE CONSIDERED OPTIONAL.	
1808		3216	BUFREG=,	/TEXT BUFFER STARTS HERE.
1809		3600	*3600	
1810	3600	2741	01,	RECOVR*1/STARTING ADDRESS
1811	3601	1200	BEGIN, TAD 01	/INITIALIZE ANY B=FAMILY COMPUTER.
1812	3602	3176	DCA START=1	
1813	3603	7000	NOP/(IOPRESET)	/*****
1814	3604	4575	JMS I PCLEAR	/***** INITIALIZE POINT DISPLAY
1815	3605	7300	CLA CLL	

1816 361 3414
 1817 3617 2857
 1818 3617 5206
 1819 3611 7200
 1820 3612 6213
 1821 3613 3667
 1822 3614 1262
 1823 3615 3670
 1824 3616 1263
 1825 3617 3671
 1826 3620 6201
 1827 3621 4666
 1828 3622 3655
 1829 3623 6212
 1830 3624 4664
 1831 3625 6211
 1832 3626 2400
 1833 3627 6211
 1834 3632 7400
 1835 3631 400
 1836 3632 6212
 1837 3633 4667
 1838 3634 3651
 1839 3635 6132
 1840 3636 6134
 1841 3637 7240
 1842 3640 6133
 1843 3641 1261
 1844 3642 6132
 1845 3643 6135
 1846 3644 7200
 1847 3645 6046
 1848 3646 6001
 1849 3647 5650
 1850 3650 2216
 1851 3651 0110
 1852 3652 0030
 1853 3653 0076
 1854 3654 0002
 1855 3655 0100
 1856 3656 0025
 1857 3657 0023
 1858 3660 0001
 1859 3661 0101
 1860 3662 5772
 1861 3663 5773
 1862 3664 7200
 1863 3665 7773
 1864 3666 7774
 1865 3667 7775
 1866 3670 7776
 1867 3671 7777
 1868 4620
 1869 4620 1045
 1870 4621 7710

T12,

DCA I FLTXR
 ISZ CNTR/INITIALIZED BY LOAD.
 JMP ,=2 /CLEAR INPUT BUFFER
 CLA /***** FIX UP DIAL I/O ROUTINES
 6213 /CIF CDF 10/*****
 DCA I G7775 /*****
 TAD G5772 /*****
 DCA I G7776 /*****
 TAD G5773 /*****
 DCA I G7777 /*****
 6201 /CDF 0 /*****
 JMS I G7774 /*****
 GBLOK /*****
 6212 /CIF 10 /*****
 JMS I G7200 /*****
 6211 /CDF 10 /*****
 2400 /*****
 6211 /CDF 10 /*****
 7400 /*****
 400 /*****
 6212 /CIF 10 /*****
 JMS I G7775 /***** WRITE MILDRED INTO UPPER
 RITEOU /***** SOURCE WORKING AREA
 CLLR /***** INITIALIZE CLOCK
 CLEN /*****
 CLA CMA /*****
 CLAB /*****
 TAD G101 /*****
 CLLR /*****
 CLSA /*****
 CLA /*****
 TIS /*****
 ION /*****
 JMP I ,+1 /*****
 ERT /***** ERASE ALL
 RITEOU: 110 /*****
 30 /*****
 76 /*****
 2 /*****
 GBLOK: 100 /*****
 25 /*****
 23 /*****
 1 /*****
 G101: 101 /*****
 G5772: 5772 /*****
 G5773: 5773 /*****
 G7200: 7200 /*****
 G7773: 7773 /*****
 G7774: 7774 /*****
 G7775: 7775 /*****
 G7776: 7776 /*****
 G7777: 7777 /*****
 *4600*20
 FEXP, GETSGN /TAKE ABSOLUTE VALUE
 SPA CLA

1871	4622	4724	JMS I NEGP
1872	4623	3033	DCA T3 /C(SIGN)=-1 IF I X2<<
1873	4624	4407	FINT
1874	4625	4313	FMUL LG2E
1875	4626	4675	FPUT I X2
1876	4627	300	FEXT
1877	4632	4453	JMS I INTEGER /TAKE INTEGER PART
1878	4631	3325	DCA FLAG2 /SAVE LOW ORDER DATA
1879	4632	4407	FINT
1880	4633	7300	FNOR
1881	4634	6676	FPUT I XSQ2
1882	4635	2675	FGET I X2
1883	4636	2676	FSUB I XSQ2
1884	4637	6675	FPUT I X2
1885	4640	4675	FMUL I X2
1886	4641	6676	FPUT I XSQ2
1887	4642	1310	FADD DF
1888	4643	6326	FPUT TEMP
1889	4644	305	FGET CF
1890	4645	3326	FDIV TEMP
1891	4646	2675	FSUB I X2
1892	4647	1277	FADD AF
1893	4650	6326	FPUT TEMP
1894	4651	302	FGET BF
1895	4652	4676	FMUL I XSQ2
1896	4653	1326	FADD TEMP
1897	4654	6326	FPUT TEMP
1898	4655	2675	FGET I X2
1899	4656	3326	FDIV TEMP
1900	4657	4321	FMUL TWO
1901	4660	1316	FADD ONE
1902	4661	0000	FEXT
1903	4662	1325	TAD FLAG2
1904	4663	1044	TAD FLAC
1905	4664	3044	DCA FLAC
1906	4665	2033	ISE T3
1907	4666	5536	RETURN
1908	4667	4407	FINT
1909	4670	6675	FPUT I X2
1910	4671	1316	FGET ONE
1911	4672	3675	FDIV I X2
1912	4673	0000	FEXT
1913	4674	5536	RETURN
1914			/CONSTANTS FOR FEXP
1915	4675	5321	X2, X
1916	4676	5325	XSQ2, XSQR
1917	4677	0004	AF, 0004
1918	4700	2372	2372
1919	4701	1402	1402
1920	4702	7774	BF, 7774
1921	4703	2157	2157
1922	4704	5157	5157
1923	4705	0012	CF, 0012
1924	4706	5454	5454
1925	4707	343	0343

1926	4710	007	DF,	0007
1927	4711	2566		2566
1928	4712	5341		5341
1929	4713	001	LG2E,	0001
1930	4714	2705		2705
1931	4715	2435		2435
1932	4716	001	ONE,	0001
1933	4717	2000		2000
1934	4720	000		0000
1935	4721	002	TWO,	0002
1936	4722	2000		2000
1937	4723	000		0000
1938	4724	5163	NEGP,	FNEG
1939	4725	000	FLAG2,	0
1940	4726	000	TEMP,	0
1941	4727	000		0
1942	4730	000		0
1943	4731	000		0

/MAIN ALGORITHM FOR ARCTANGENT

1945	4732	4407	ARCALG,	FINT
1946	4733	3675		FGET I X2
1947	4734	4675		FMUL I X2
1948	4735	6676		FPUT I XSQ2
1949	4736	4374		FMUL BET2
1950	4737	1371		FADD BET1
1951	4740	4676		FMUL I XSQ2
1952	4741	1366		FADD BET2
1953	4742	6326		FPUT TEMP
1954	4743	4363		FGET ALF2
1955	4744	4676		FMUL I XSQ2
1956	4745	1366		FADD ALF1
1957	4746	4676		FMUL I XSQ2
1958	4747	1355		FADD ALF2
1959	4750	4675		FMUL I X2
1960	4751	3326		FDIV TEMP
1961	4752	0000		FEXT
1962	4753	5754		JMP I ,+1
1963	4754	5024		ARCRTN

/CONSTANTS = FLOATING ARC TANGENT

1964	4755	0000	ALF2,	0000
1965	4756	2437		2437
1966	4757	1643		1643
1967	4760	7777	ALF1,	7777
1968	4761	3304		3304
1969	4762	4434		4434
1970	4763	7773	ALF2,	7773
1971	4764	3306		3306
1972	4765	5454		5454
1973	4766	0000	BET2,	0000
1974	4767	2437		2437
1975	4770	1646		1646
1976	4771	0000	BET1,	0000
1977	4772	2427		2427
1978	4773	2323		2323
1979	4774	7775	BET2,	7775

1981 4775 3427
 1982 4776 7452
 1983
 1984
 1985 5222 1245
 1986 5221 7710
 1987 5222 4363
 1988 5223 3333
 1989 5224 4407
 1990 5225 6635
 1991 5226 2637
 1992 5227 0000
 1993 5210 1045
 1994 5211 7710
 1995 5212 5221
 1996 5213 4407
 1997 5214 2637
 1998 5215 3635
 1999 5216 6635
 2000 5217 0000
 2001 5220 7240
 2002 5221 3362
 2003 5222 5623
 2004 5223 4732
 2005 5224 2362
 2006 5225 5634
 2007 5226 4407
 2008 5227 6635
 2009 5230 4636
 2010 5231 2635
 2011 5232 0000
 2012 5233 5634
 2013 5234 5301
 2014
 2015 5235 5321
 2016 5236 5315
 2017 5237 4716
 2018 5240 1245
 2019 5241 7450
 2020 5242 4566
 2021 5243 7710
 2022 5244 4566
 2023 5245 4407
 2024 5246 6756
 2025 5247 2637
 2026 5250 0000
 2027 5251 1045
 2028 5252 7450
 2029 5253 5536
 2030 5254 7710
 2031 5255 5264
 2032 5256 4407
 2033 5257 637
 2034 5260 3756
 2035 5261 6756

/FLOATING POINT ARC TANGENT
 *5222
 ARCTN, GETSGN /TAKE ABSOLUTE VALUE
 SPA CLA
 JMS FNEG
 DCA T3
 FINT
 FPUT I X1
 FSUB I CON1
 FEXT
 GETSGN
 SPA CLA
 JMP GO /LESS THAN ONE
 FINT
 FGET I CON1
 FDIV I X1
 FPUT I X1
 FEXT
 CLA CMA
 DCA FLAG1 /SIGN FLAG OF RESULT
 JMP I ,+1 /CALL ALGORITHM
 ARCALG
 ARCRN, IS2 FLAG1 /RETURN HERE
 JMP I EXIT1
 FINT
 FPUT I X1
 FGET I PI2
 FSUB I X1
 FEXT
 JMP I ,+1
 EXIT1, EXIT2
 /CONSTANTS FOR ARCTANGENT
 X1, X
 PI2, PI0T
 CON1, ONE
 FLOG, GETSGN /FLOATING LOGARITHM
 SNA
 ERROR3 /ZERO ARGUMENT FOR LOG
 SPA CLA
 ERROR4 /*
 FINT
 FPUT I TEM
 FSUB I CON1
 FEXT
 GETSGN
 SNA
 RETURN
 SNA CLA
 JMP STARTL
 FINT
 FGET I CON1
 FDIV I TEM
 FPUT I TEM

2036	5062	7240	FEXT
2037	5063	7240	CLÄ CMA
2038	5064	3233	STARTL; DCÄ T3
2039	5065	1205	TAD P13
2040	5066	3044	DCA FLAC
2041	5067	7040	CMA
2042	5070	1756	TAD I TEM
2043	5071	3045	DCA FLAC+1
2044	5072	3046	DCA FLAC+2
2045	5073	3047	DCA FLAC+3
2046	5074	7001	IAC
2047	5075	3756	DCA I TEM
2048	5076	4407	FINT
2049	5077	4357	FMUL LOG2
2050	5100	6635	FPUT I X1
2051	5101	756	FGET I TEM
2052	5102	2637	FSUB I CON1
2053	5103	6756	FPUT I TEM
2054	5124	4353	FMUL LOG8
2055	5105	1350	FADD LOG7
2056	5106	4756	FMUL I TEM
2057	5107	1345	FADD LOG6
2058	5110	4756	FMUL I TEM
2059	5111	1342	FADD LOG5
2060	5112	4756	FMUL I TEM
2061	5113	1337	FADD L4
2062	5114	4756	FMUL I TEM
2063	5115	3334	FADD L3
2064	5116	4756	FMUL I TEM
2065	5117	1331	FADD L2
2066	5120	4756	FMUL I TEM
2067	5121	1326	FADD L1
2068	5122	4756	FMUL I TEM
2069	5123	1635	FADD I X1
2070	5124	0000	FEXT
2071	5125	5634	JMP I EXIT1
2072	5126	0000	L1, 0000
2073	5127	3777	3777
2074	5130	7742	7742
2075	5131	7777	L2, 7777
2076	5132	4000	4000
2077	5133	4100	4100
2078	5134	7777	L3, 7777
2079	5135	2517	2517
2080	5136	0310	0310
2081	5137	7776	L4, 7776
2082	5140	4113	4113
2083	5141	7211	7211
2084			/LOGARITHM CONSTANTS
2085	5142	7776	LOG5, 7776
2086	5143	2535	2535
2087	5144	3301	3301
2088	5145	7775	LOG6, 7775
2089	5146	0746	0746
2090	5147	0771	0771

```

2091 5150 7774 LOG7, 7774
2092 5151 2236 2236
2093 5152 4334 4334
2094 5153 7771 LOG8, 7771
2095 5154 4544 4544
2096 5155 1735 1735
2097 5156 4726 TEM, TEMP
2098 5157 0000 LOG2, 0
2099 5160 2613 2613
2100 5161 4414 4414
2101 5162 0000 FLAG1, 0
2102 5163 0000 FNEG, 0
2103 5164 4451 JMS I MINSKI
2104 5165 7240 CLA CMA
2105 5166 5763 JMP I FNEG
2106 5167 6213 LO, 6213 /CIF CDF 10/*****
2107 5170 5126 JMP XLO /*****
2108 5171 6213 LC, 6213 /CIF CDF 10/*****
2109 5172 5130 JMP XLC /*****
2110 5173 6213 LL, 6213 /CIF CDF 10/*****
2111 5174 5132 JMP XLL /*****
2112 /FLOATING POINT SINE AND COSINE
2113
2114
2115
2116 5177 5177 *5177
2117 5177 4407 FCOS, FINT /COS(X)=SIN(PI/2-X)
2118 5200 6321 FPUT X
2119 5201 0315 FGET PIOT
2120 5202 2321 FSUB X
2121 5203 0000 FEXT
2122 5204 1045 FSIN, GETSGN
2123 5205 7740 SMA SZA CLA
2124 5206 5214 JMP MOD
2125 5207 1045 GETSGN
2126 5210 7700 SMA CLA
2127 5211 5536 RETURN /YES SIN(0)=0
2128 5212 4451 JMS I MINSKI
2129 5213 7040 CMA /NO: SIN(-X)=-SIN(X)
2130 5214 3033 MOD, DCA T3
2131 /REDUCE X MODULO 2 PI
2132 5215 4407 FINT
2133 5216 3305 FDIV TWOPI
2134 5217 6325 FPUT XOR
2135 5220 0000 FEXT
2136 5221 4453 JMS I INTEGER
2137 5222 4407 FINT
2138 5223 7000 FNOR
2139 5224 6321 FPUT X
2140 5225 0325 FGET XOR
2141 5226 2321 FSUB X
2142 5227 4305 FMUL TWOPI
2143 5230 6321 FPUT X
2144 5231 2311 FSUB PI /X<PI?
2145 5232 0000 FEXT

```


2146	5233	1045	GETSGN	
2147	5234	7710	SPA CLA	
2148	5235	5244	JMP PCHECK	/YES
2149	5236	4427	FINT	/NO, SIN(X-PI)=-SIN(X)
2150	5237	6321	FPUT X	
2151	5240	0000	FEXT	
2152	5241	1033	TAD T3	/INVERT THE SIGN
2153	5242	7040	CMA	
2154	5243	3033	DCA T3	
2155	5244	4407	PCHECK, FINT	/X<PI/?
2156	5245	1321	FGET X	
2157	5246	2315	FSUB PIOT	
2158	5247	0002	FEXT	
2159	5250	1045	GETSGN	
2160	5251	7710	SPA CLA	
2161	5252	5260	JMP PALG	/YES
2162	5253	4407	FINT	/NO
2163	5254	0311	FGET PI	/SIN(X)=SIN(PI-X)
2164	5255	2321	FSUB X	
2165	5256	6321	FPUT X	
2166	5257	0000	FEXT	
2167	5260	4407	PALG, FINT	
2168	5261	0321	FGET X	
2169	5262	3315	FDIV PIOT	
2170	5263	6321	FPUT X	
2171	5264	4321	FMUL X	
2172	5265	6325	FPUT XSQR	
2173	5266	0331	FGET C9	
2174	5267	4325	FMUL XSQR	
2175	5270	1335	FADD C7	
2176	5271	4325	FMUL XSQR	
2177	5272	1341	FADD C5	
2178	5273	4325	FMUL XSQR	
2179	5274	1345	FADD C3	
2180	5275	4325	FMUL XSQR	
2181	5276	1315	FADD PIOT	
2182	5277	4321	FMUL X	
2183	5300	0000	FEXT	
2184	5301	2033	EXIT2, ISZ T3	
2185	5302	5536	RETURN	
2186	5303	4451	JMS I MINSKI	
2187	5304	5536	RETURN	
2188			/CONSTANTS AND POINTERS	
2189	5305	0003	TWOPI, 0003	
2190	5306	3110	3110	
2191	5307	3756	3756	/(3755) = FOR 4-WORD
2192	5310	3235	3235	
2193	5311	0002	PI, 0002	
2194	5312	3110	3110	
2195	5313	3756	3756	
2196	5314	3235	3235	
2197	5315	0001	PIOT, 0001	/USED BY SINE AND COSINE
2198	5316	3110	3110	
2199	5317	3756	3756	
2200	5320	3235	3235	

2201	5321	000	X,	0000
2202	5322	000		0000
2203	5323	000		0000
2204	5324	000		0000
2205	5325	000	XSGR,	0000
2206	5326	000		0000
2207	5327	000		0000
2208	5332	000		0000

/SINE CONSTANTS

2210	5331	7764	C9,	7764
2211	5332	2501		2501
2212	5333	7015		7015
2213	5334	1042		1042
2214	5335	7771	C7,	7771
2215	5336	5464		5464
2216	5337	5514		5514
2217	5340	6150		6150
2218	5341	7775	C5,	7775
2219	5342	2431		2431
2220	5343	5361		5361
2221	5344	4736		4736
2222	5345	0000	C3,	0000
2223	5346	5325		5325
2224	5347	0414		0414
2225	5350	3167		3167

/END OF EXTENDED FUNCTIONS.

/

/HANDLES 0 I, EXPRESSION

/SETS CLOCK ACCORDING TO EXPRESSION

/

SETCLK: PUSHJ /*****

EVAL=1 /*****

FINT /*****

FMUL MHUNDRO /*****

FEXT /*****

CLLR /*****

CLEN /*****

JMS I INTEGER /*****

CLAB /*****

CLA /*****

TAD C100 /*****

CLLR /*****

TAD C200 /*****

CLEN /*****

TAD 04600 /*****

CLLR /*****

CLA /*****

JMP I ,+1 /*****

PROC /*****

04600, 4600 /*****

MHUNDRO,71470010 /*****

2252
2253

2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308

5400
5401
5402
5403
5404
5405
5406
5407
5410
5411
5412
5413
5414
5415
5416
5417
5420
5421
5422
5423
5424
5425
5426
5427
5430
5431
5432
5433
5434
5435
5436
5437
5440
5441
5442
5443
5444
5445
5446
5447
5450
5451
5452
5453
5454
5455
5456

```

/IN THE COMMENTS BELOW:-
/ F = NUMBER OF DIGITS TO BE OUTPUT      =FISW
/ D = NUMBER OF DECIMAL PLACES           =DECP
/ E = DECIMAL EXPONENT                    =BEXP
/ P = NUMBER OF PLACES REMAINING TO BE
/      PRINTED BEFORE DECIMAL POINT

*5400
DIGITS=6      /NUMBER OF DECIMAL DIGITS OUT
TGO, 0
DCA SCOUNT    /SAVE MAX. NUMBER OF DIGITS AVAILABLE = *SET COUNTS*
TAD FISW
RTL6
AND P77
DCA T1
TAD T1
CIA          /NO, COMPUTE FIELD SIZES
SNA
TAD MD
DCA FCOUNT
TAD FISW    /(JMP FPRNT) = FOR NO ROUNDING,
SNA        /FLOATING OUTPUT?
JMP R6     /YES, ROUND OFF TO MAX. NO. PLACES
AND P77
DCA DECP
TAD FCOUNT
TAD DECP
SPA        / F=D > 0 ?
JMP ,+5    /YES
CLA CMA    /NO,
TAD T1
DCA DECP   /MAKE 0 = F=1
CMA
TAD T3     /COMPARE DECIMAL EXPONENT
SMA        / F=D > E?
CLA        /NO, ROUND OFF TO F PLACES
TAD T1     /YES
SPA        / D+E < 0 ?
JMP FPRNT=2 /YES, NO ROUNDING NEEDED, GO TO PRINT
TAD MD     /NO, ROUND TO D+E PLACES,
SMA        /TO A MAXIMUM OF D PLACES
CLA
R6, TAD RND2 / *ROUND UP *
DCA T2     /SAVE NUMBER+1 OF PLACES TO ROUND TO.
TAD I BUFST
TAD T2     /SET UP BUFFER ADDRESS AT WHICH
DCA PLCE   /ROUNDING OFF SHOULD START
TAD T2
CIA        /SET UP COUNT OF MAXIMUM NUMBER
DCA T2     /OF CARRIES ALLOWABLE
TAD K5     /LITTLE EXTRA ON FIRST DIGIT,
RET, ISZ I PLCE /ADD 1 TO DIGIT AT CURRENT POSITION
TAD I PLCE
TAD UM12
SPA CLA    /CARRY REQUIRED?
JMP FPRNT  /NO, GO TO OUTPUT
    
```

2309	5457	3736	DCA I PLCE	/YES, MAKE CURRENT DIGIT ZERO
2310	5460	2771	ISZ T2	/BEGINNING OF BUFFER REACHED?
2311	5461	5321	JMP DECR	/NO, DECREMENT BUFFER ADDRESS AND REPEAT
2312	5462	2736	ISZ I PLCE	/YES, SET MANTISSA TO 0.1
2313	5463	2833	ISZ T3	/COMPENSATE BY INCREMENTING EXPONENT
2314	5464	7200	CLA	
2315	5465	1952	FPRNT, TAD FISW	/AUTO-INDEX REGISTER ALREADY SET, = *PRINT*
2316	5466	7650	SNA CLA	/ F = 0 ?
2317	5467	5356	JMP FLOUT	/YES, OUTPUT AS FLOATING NUMBER
2318	5470	1335	TAD FCOUNT	
2319	5471	1033	TAD T3	
2320	5472	7540	SMA SZA	/ E > F ?
2321	5473	5355	JMP FLOUT=1	/YES, CONVERT TO E FORMAT
2322	5474	1333	TAD DECP	
2323	5475	7500	SMA	/ E < F=D ?
2324	5476	7200	CLA	/NO, TAKE P = E
2325	5477	7041	CIA	/YES, TAKE P = F=D
2326	5520	1033	TAD T3	
2327	5501	7041	CIA	
2328	5522	3232	DCA T1	/SET UP MINUS P
2329	5523	1033	TAD T3	/PRINT DD,DDD
2330	5524	1032	TAD T1	
2331	5525	7650	SNA CLA	/ P = E ?
2332	5526	5343	JMP DIG	/YES, PRINT DIGIT
2333	5527	1032	TAD T1	/NO,
2334	5510	7001	IAC	
2335	5511	7710	SPA CLA	/ P > 1 ?
2336	5512	1105	TAD M20	/YES, TAKE SPACE (240=260); OTHERWISE ZERO
2337	5513	4336	IN, JMS OUTA	/PRINT CHARACTER
2338	5514	2032	ISZ T1	/P CHARACTERS PRINTED?
2339	5515	5303	JMP BACK	/NO
2340	5516	1102	TAD PER	/YES,
2341	5517	4551	PRINTC	/PRINT DECIMAL POINT
2342	5520	5303	JMP BACK	
2343	5521	7040	DECR, CMA	/BACKUP TO TOP OF BUFFER.
2344	5522	1336	TAD PLCE	
2345	5523	3336	DCA PLCE	
2346	5524	5252	JMP RET	
2347	5525	0004	K5, 4	
2348	5526	7772	MD, =DIGITS	
2349	5527	0007	RND2, DIGITS+1	
2350	5530	7766	OM12, =12	
2351	5531	6150	BUFST, SADR	
2352	5532	6154	OPUT, OUTDG	
2353	5533	0000	DECP, 0	/MODIFIABLE LOCATIONS
2354	5534	0000	SCOUNT, 0	
2355	5535	0000	FCOUNT, 0	
2356	5536	5536	PLCE=,	
2357	5536	0000	OUTA, 0	/MODIFIED REGISTERS.
2358	5537	4732	JMS I OPUT	/PRINT CHARACTER
2359	5540	2335	ISZ FCOUNT	/F CHARACTERS PRINTED?
2360	5541	5736	JMP I OUTA	/NO, RETURN
2361	5542	5620	JMP I TGO	/YES, NUMBER FINSHED
2362	5543	7040	DIG, CMA	
2363	5544	1033	TAD T3	/REDUCE E, BY 1

2364	545	3033	DCA T3	
2365	5546	2334	ISZ SCOUNT	/ARE ALL SIG. FIGS. USED?
2366	5547	5353	JMP ,*4	/NO
2367	5550	7040	CMA	/YES,
2368	5551	3334	DCA SCOUNT	/RESET COUNT TO -1
2369	5552	5313	JMP IN	/AND LEAVE C(AC) = 0
2370	5553	1414	TAD I FLTXR	/TAKE NEXT DIGIT FROM BUFFER
2371	5554	5313	JMP IN	
2372			/DO FLOATING OUTPUT	
2373	5555	7200	CLA	/IF OUTPUT TOO LARGE,
2374	5556	4732	FLOUT, JMS I OPUT	/PRINT "0"
2375	5557	1102	TAD PER	
2376	5560	4551	PRINTC	/PRINT " , "
2377	5561	2200	ISZ TGO	/SECOND RETURN
2378	5562	1414	TAD I FLTXR	/TAKE NEXT DIGIT FROM BUFFER
2379	5563	4336	JMS OUTA	/PRINT IT
2380	5564	2334	ISZ SCOUNT	/TEST FOR END OF INPUT
2381	5565	5362	JMP ,*3	/AND REPEAT
2382	5566	7040	CMA	
2383	5567	3334	DCA SCOUNT	/OUTPUT EXTRA ZEROS.
2384	5570	5363	JMP ,*5	
2385	5571	0000	ABSOLV, 0	
2386	5572	1045	TAD WORD	
2387	5573	3050	DCA SIGNF	
2388	5574	1045	TAD WORD	
2389	5575	7710	SPA CLA	
2390	5576	4451	JMS I MINSKI	
2391	5577	5771	JMP I ABSOLV	
2392			/DOUBLE PRECISION DECIMAL-BINARY	
2393			/INPUT AND CONVERSION FOR + OR = XXX...	
2394		5600	*5600	
2395	5600	0000	DECONV, 0	
2396	5601	3046	DCA LORD	
2397	5602	3044	DCA EXP	/ZERO THE EXPONENT AND
2398	5603	3045	DCA WORD	/INITIALIZE FLOATING AC.
2399	5604	3047	DCA OVER2	
2400	5605	3314	DCA DNUMBR	
2401	5606	3050	DCA SIGNF	
2402	5607	1066	TAD CHAR	/ALLOW KEYBOARD SIGN CHECKS.
2403	5610	1264	TAD MPLUS	
2404	5611	7450	SNA	
2405	5612	5220	JMP ,*6	/*SIGN) GET NEXT
2406	5613	1111	TAD M2	/CHECK = SIGN
2407	5614	7640	SZA CLA	
2408	5615	5221	JMP ,*4	
2409	5616	7040	CMA	/INIT SIGN CHECK TO POS.
2410	5617	3050	DCA SIGNF	
2411	5620	4666	JMS I XINPUT	/GET NEXT
2412	5621	1066	TAD CHAR	/A SPACE PERHAPS?
2413	5622	1265	TAD MSPACE	
2414	5623	7650	SNA CLA	
2415	5624	5220	JMP ,*4	
2416	5625	4227	JMS DECON	
2417	5626	5600	JMP I DECONV	
2418	5627	0000	DECON, 0	

2419 5630 1066
 2420 5631 1262
 2421 5632 7653
 2422 5633 5627
 2423 5634 4561
 2424 5635 5627
 2425 5636 5247
 2426 5637 1054
 2427 5640 3313
 2428 5641 4267
 2429 5642 2314
 2430 5643 7640
 2431 5644 4566
 2432 5645 4666
 2433 5646 5230
 2434 5647 1066
 2435 5650 1112
 2436 5651 7710
 2437 5652 5627
 2438 5653 1066
 2439 5654 1263
 2440 5655 7740
 2441 5656 5627
 2442 5657 1066
 2443 5660 1122
 2444 5661 5240
 2445 5662 7473
 2446 5663 7446
 2447 5664 7525
 2448 5665 7540
 2449 5666 0756
 2450 5667 0000
 2451 5670 1047
 2452 5671 3043
 2453 5672 1046
 2454 5673 3042
 2455 5674 1045
 2456 5675 3041
 2457 5676 3312
 2458 5677 4315
 2459 5700 4315
 2460 5701 4333
 2461 5702 4315
 2462 5703 1313
 2463 5704 3043
 2464 5705 3042
 2465 5706 3041
 2466 5707 4333
 2467 5710 1312
 2468 5711 5667
 2469 5712 0000
 2470 5713 0000
 2471 5714 0000
 2472 5715 0000
 2473 5716 1047

TAD CHAR /TEST LEAD CHARACTER FOR TERMINATOR
 TAD MINE
 SNA CLA
 JMP I DECON /E
 TESTN
 JMP I DECON /.
 JMP DTST /OTHER
 TAD SORTCN /N
 DCA DIGIT /YES
 JMS MULT10 /REMAIN MUST BE SINCE OVERFLOW IS CHECKED
 ISZ DNUMBR /COUNT DIGITS
 SZA CLA
 ERROR2 /INPUT=OVERFLOW ERROR
 JMS I XINPUT
 JMP DECON*1 /CONTINUE
 DTST. TAD CHAR /ALLOW A-Z
 TAD MINUSA
 SPA CLA
 JMP I DECON
 TAD CHAR
 TAD MINUSZ
 SZA SMA CLA
 JMP I DECON /USE SIX BITS OF ASCII
 TAD CHAR
 AND P77
 JMP DSAVE
 MINE. =305 / (7532) FOR AMPERSAND
 MINUSZ. =332
 MPLUS. =253
 MSPACE. =240
 XINPUT. INPUT
 MULT10. 0 /ROUTINE TO MULTIPLY FLAG BY TEN (10)
 TAD OVER2
 DCA OVER1
 TAD LORD /DOUBLE PRECISION WORD
 DCA AC1L /BY TEN (DECIMAL)
 TAD HORD /REMAIN=REMAINDER
 DCA AC1H
 DCA REMAIN /CLEAR OVERFLOW WORD
 JMS MULT2 /CALL SUBROUTINE TO
 JMS MULT2 /MULTIPLY BY TWO
 JMS DUBLAD /CALL DOUBLE ADD
 JMS MULT2
 TAD DIGIT /ADD LAST DIGIT RECEIVED
 DCA OVER1
 DCA AC1L
 DCA AC1H
 JMS DUBLAD
 TAD REMAIN /EXIT WITH REMAINDER
 JMP I MULT10 /IN AC
 REMAIN. 0
 DIGIT. 0 /STORAGE FOR DIGIT
 DNUMBR. 0 /NUMBER OF DIGITS
 MULT2. 0 /MULTIPLY OVER2, LORD, HORD BY 2
 TAD OVER2

2474	5717	7104	CLL RAL /CARRY INSERT BIT IS IN LINK
2475	5720	3047	DCA OVER2
2476	5721	1046	TAD LORD
2477	5722	7004	RAL
2478	5723	3046	DCA LORD
2479	5724	1045	TAD HORD
2480	5725	7004	RAL
2481	5726	3045	DCA HORD
2482	5727	1312	TAD REMAIN
2483	5730	7004	RAL
2484	5731	3312	DCA REMAIN
2485	5732	5715	JMP I MULT2
2486	5733	7000	DUBLAD, 0 /TRIPLE PRECISION ADDITION
2487	5734	7300	CLA CLL
2488	5735	1047	TAD OVER2
2489	5736	1043	TAD OVER1
2490	5737	3047	DCA OVER2
2491	5740	7004	RAL
2492	5741	1046	TAD LORD
2493	5742	1042	TAD AC1L
2494	5743	3046	DCA LORD
2495	5744	7004	RAL
2496	5745	1045	TAD HORD
2497	5746	1041	TAD AC1H
2498	5747	3045	DCA HORD
2499	5750	7004	RAL
2500	5751	1312	TAD REMAIN /WITH OVERFLOW
2501	5752	3312	DCA REMAIN
2502	5753	5733	JMP I DUBLAD
2503	5754	7000	DIV1, 0 /SHIFT OPERAND RIGHT
2504	5755	7300	CLA CLL /TRIPLE PRECISION
2505	5756	1041	TAD AC1H
2506	5757	7510	SPA
2507	5760	7120	CLL CML
2508	5761	7010	RAR
2509	5762	3041	DCA AC1H
2510	5763	1042	TAD AC1L
2511	5764	7010	RAR
2512	5765	3042	DCA AC1L
2513	5766	1043	TAD OVER1
2514	5767	7010	RAR
2515	5770	3043	DCA OVER1
2516	5771	2040	ISE EX1
2517	5772	5754	JMP I DIV1
2518	5773	5754	JMP I DIV1
2519	5774	4566	FSSERR, ERROR4 /***** (SUBSCRIPT ERROR FOR FILE VARIABLE-OR NOT DEFINED)
2520		6000	*6000
2521			/FLOATING OUTPUT CONVERSION ROUTINE
2522	6000	6000	FLOUTP, 0
2523	6001	7610	SKP CLA /***** /GETS RID OF # IN PRINTOUT
2524			LMODE
2525	6002	6377	OPTR, 6377 /*****
2526			PHODE
2527	6003	1045	TAD HORD /NUMBER>0??
2528	6004	7700	SMA CLA

/PRINT "E" OR A SPACE.

2529 6005 1334
 2530 6006 1336
 2531 6007 4551
 2532 6010 4753
 2533 6011 3033
 2534 6012 1344
 2535 6013 7510
 2536 6014 5227
 2537 6015 7440
 2538 6016 1341
 2539 6017 7750
 2540 6020 5234
 2541 6021 4407
 2542 6022 4744
 2543 6023 0000
 2544 6024 7001
 2545 6025 1033
 2546 6026 5211
 2547 6027 4407
 2548 6030 4752
 2549 6031 0000
 2550 6032 7040
 2551 6033 5225
 2552 6034 3745
 2553 6035 3746
 2554 6036 1350
 2555 6037 3014
 2556 6040 1044
 2557 6041 7140
 2558 6042 3354
 2559 6043 1343
 2560 6044 3044
 2561 6045 4527
 2562 6046 2354
 2563 6047 5245
 2564 6050 1746
 2565 6051 7450
 2566 6052 5270
 2567 6053 1342
 2568 6054 7710
 2569 6055 5264
 2570 6056 7001
 2571 6057 3414
 2572 6060 2044
 2573 6061 1342
 2574 6062 2033
 2575 6063 7000
 2576 6064 1746
 2577 6065 2033
 2578 6066 7000
 2579 6067 7410
 2580 6070 4747
 2581 6071 3414
 2582 6072 2044
 2583 6073 5270

FG02.

FG03.

FG04.

FG05.

TAD SMSP
 TAD SMIN
 PRINTC
 JMS I ABSOL2
 DCA T3
 TAD EXP
 SPA
 JMP FG03
 SZA
 TAD M4
 SPA SNA CLA
 JMP FG04
 FINT
 FMUL I PPTEN
 FEXT
 IAC
 TAD T3
 JMP FG02
 FINT
 FMUL I TENPT
 FEXT
 CMA
 JMP ,=6
 DCA I DPT
 DCA I REPT
 TAD SADR
 DCA FLT XR
 TAD EXP
 CMA CLL
 DCA OUTDG
 TAD DCOUNT
 DCA EXP
 JMS I DOUBLE.
 ISZ OUTDG
 JMP ,=2
 TAD I REPT
 SNA
 JMP FG05
 TAD FM12
 SPA CLA
 JMP ,*7
 IAC
 DCA I FLT XR
 ISZ EXP
 TAD FM12
 ISZ T3
 NOP
 TAD I REPT
 ISZ T3
 NOP
 SKP
 JMS I MI0PT
 DCA I FLT XR
 ISZ EXP
 JMP ,=3

/INITIALIZE DECIMAL EXPONENT
/IS EXP 0 TO 47

/TOO LARGE!MULTIPLY BY 1/10.

/MULTIPLY BY TWO TO POSITION BIT0
/CLEAR OVERFLOW WORD
/INIT BUFFER POINTER

/COMPUTE BITS IN 1ST DIGIT

/TEMP COUNT
/SETUP COUNT OF TOTAL OUTPUT

/ROTATE OUT THE 1ST 4 BITS

/TEST FOR 10-15,0,1-9

/IGNORE 1ST ZERO

/0=9

/OUTPUT A 1
/COUNT THE DIGIT
/CORRECT REMAINDER
/BUMP DECIMAL EXPONENT

/COMPUTE RESULTANT OR SECOND DIGIT

/IE. ,672X10=6+.72.. ETC

/ALL DIGITS OUTPUT??
/NO! CONTINUE

ADDRESS	DATA	OPERATION	COMMENT
2584	6074	1350	TAD SADR /INIT BUFFER POINTER
2585	6075	3014	DCA FLT XR
2586	6076	1343	TAD DCOUNT
2587	6077	4751	JMS I ROUND /OUTPUT MANTISSA
2588	6100	5600	JMP I FLOUTP /FIXED POINT DONE
2589	6101	1333	TAD CHRT /PRINT "E"
2590	6102	4551	PRINTC
2591			/OUTPUT THE EXPONENT
2592	6103	1033	TAD T3 /TAKE ABSOLUTE VALUE OF EXPONENT
2593	6104	7510	SPA
2594	6105	7041	CIA
2595	6106	3045	DCA HORD /SAVE + POWER
2596	6107	1033	TAD T3 /PRINT SIGN
2597	6110	7700	SMA CLA
2598	6111	1111	TAD M2
2599	6112	1336	TAD SMIN
2600	6113	4551	PRINTC
2601	6114	1045	TAD HORD
2602	6115	2044	ISZ EXP
2603	6116	1337	TAD M144
2604	6117	7500	SMA
2605	6120	5315	JMP ,=3
2606	6121	1340	TAD C144
2607	6122	3045	DCA HORD /SAVE TENS AND UNITS
2608	6123	7040	CMR /OUTPUT HUNDREDS
2609	6124	1044	TAD EXP
2610	6125	7440	SZA /UNLESS ZERO
2611	6126	4354	JMS OUTDG
2612	6127	1045	TAD HORD /PRINT TWO DIGITS
2613	6130	4732	JMS I PRNTI
2614	6131	5600	JMP I FLOUTP
2615	6132	2442	PRNTI, PRNT
2616	6133	0305	CHRT, 305 /E (0246) - FOR AMPERSAND
2617	6134	7763	SMSP, 240=255 /
2618	6135	0275	PEQ, 275
2619	6136	0255	SMIN, 255
2620	6137	7634	M144, =144 /=100
2621	6140	0144	C144, 0144 /+100
2622	6141	7774	M4, =4
2623	6142	7766	FM12, =12
2624	6143	7771	DCOUNT, =DIGITS=1 /NUMBER OF DIGITS OUTPUT
2625	6144	6275	PPTEN, PTEN /IEI
2626	6145	5713	OPT, DIGIT
2627	6146	5712	REPT, REMAIN /OVERFLOW FROM INTEGER MULTIPLY
2628	6147	5667	M10PT, MULT10
2629	6150	7467	SADR, BUFFER=1
2630	6151	5400	ROUND, TGO /ACTUAL OUTPUT ROUTINE
2631	6152	6271	TENPT, TEN
2632	6153	5571	ABSOL2, ABSOLV
2633	6154	0000	OUTDG, 0 /OUTPUT ONE DIGIT
2634	6155	1113	TAD C260
2635	6156	4551	PRINTC
2636	6157	5754	JMP I OUTDG
2637	6160	7750	RANMUL, 77501233315733 /*****
	6161	2333	

```

6162 5733
2638 6163 1167 LEPUT, TAD SUBS2 /***** CALLS STORING ROUTINE FOR
2639 6164 3171 DCA SUBS /***** S FN(X)
2640 6165 1170 TAD LESUB2 /*****
2641 6166 3173 DCA LESUBS /*****
2642 6167 1002 TAD LWEIMP /*****
2643 6170 6212 6212 /*****
2644 6171 4775 JMS I STORIT /*****
2645 6172 2407 ISZ I 7 /*****
2646 6173 5774 JMP I ,*1 /*****
2647 6174 6401 FPNT*1 /*****
2648 6175 2000 STORIT, ITSTOR /*****
2649 6176 6213 LS, 6213 /CIF CDF 10/***** LIBRARY SAVE
2650 6177 5134 JMP XLS /*****
2651 /USED BY 8K
2652 /FLOATING POINT INPUT
2653 *6200
2654 6200 0000 FLINTP, 0 /IF C(AC) = 0, USE CHAR
2655 6201 7640 SZR CLA /IF C(AC) NON=ZERO, GET NEXT
2656 6202 4706 JMS I XIN /GET FIRST CHAR
2657 6203 1066 TAD CHAR /IGNORE LEADING SPACES
2658 6204 1114 TAD M240
2659 6205 7650 SNR CLA
2660 6206 5202 JMP ,=4
2661 6207 4702 JMS I DPCVPT /READ FIRST DIGIT GROUP
2662 6210 1066 TAD CHAR /AND SET "SIGNP"
2663 6211 1115 TAD MPER
2664 6212 7640 SZR CLA /ENDED BY PERIOD?
2665 6213 5221 JMP FIG01
2666 6214 4706 JMS I XIN /YES, READ 2AND GROUP
2667 6215 3705 DCA I DPN
2668 6216 4703 JMS I DCONP
2669 6217 1705 TAD I DPN /SAVE NUMBER OF DIGITS IN T3
2670 6220 7041 CMA IAC
2671 6221 3033 FIG01, DCA T3 /NO,
2672 6222 1310 TAD P43
2673 6223 3044 DCA EXP
2674 6224 4704 JMS I RESOLS
2675 6225 4707 JMS I INORM /NORMALIZE FIRST, THEN
2676 6226 4407 FINT
2677 6227 6430 FPUT I PT1 /SAVE NUMBER
2678 6230 0000 FEXT
2679 6231 1066 TAD CHAR
2680 6232 1301 TAD MINUSE
2681 6233 7640 SZR CLA /"E" READ IN?
2682 6234 5246 JMP ENDFI*3 /NO
2683 6235 4706 JMS I XIN /YES, READ 3RD DIGIT GROUP
2684 6236 4702 JMS I DPCVPT /I.E. CONVERT DECIMAL EXPONENT
2685 6237 4704 JMS I RESOLS
2686 6240 1047 TAD OVER2
2687 6241 1033 TAD T3 /C(SEXP)PLACES TO RIGHT
2688 6242 3033 DCA T3 /OF LAST DIGIT
2689 /COMPENSATE FOR DECIMAL EXPONENTS
2690 6243 4407 ENDFI, FINT /RESTORE MANTISSA
2691 6244 0430 FGET I PT1

```

2692 45 000
 2693 6246 1033
 2694 6247 7450
 2695 6250 5600
 2696 6251 7700
 2697 6252 5261
 2698 6253 4407
 2699 6254 4275
 2700 6255 6430
 2701 6256 0000
 2702 6257 7001
 2703 6260 5266
 2704 6261 4407
 2705 6262 4271
 2706 6263 6430
 2707 6264 0000
 2708 6265 7040
 2709 6266 1033
 2710 6267 3033
 2711 6270 5246
 2712 6271 0004
 2713 6272 2400
 2714 6273 0000
 2715 6274 0000
 2716 6275 7775
 2717 6276 3146
 2718 6277 3147
 2719 6300 3150
 2720 6301 7473
 2721 6302 5600
 2722 6303 5627
 2723 6304 7173
 2724 6305 5714
 2725 6306 0756
 2726 6307 7335
 2727 6310 0043
 2728
 2729
 2730
 2731
 2732
 2733
 2734
 2735
 2736
 2737
 2738
 2739 6311 1066
 2740 6312 3756
 2741 6313 4545
 2742 6314 4550
 2743 6315 1771
 2744 6316 7410
 2745 6317 5313
 2746 6320 4562

FEXT
 TAD T3 /TEST DECIMAL EXPONENT
 SNA
 JMP I FLINTP /FINISHED
 SMA CLA
 JMP FIG04
 FINT /, IS TO THE LEFT
 FMUL PTEN /TIMES ,1000
 FPUT I PT1
 FEXT
 IAC
 JMP ,+6
 FIG04, FINT /, IS TO THE RIGHT
 FMUL TEN /MULTIPLY BY 10
 FPUT I PT1
 FEXT
 CMA
 TAD T3
 DCA T3
 JMP ENDFI+3
 TEN, 0004
 2400
 0000
 0000
 PTEN, 7775
 3146
 3147 /((3146) = FOR 4=WORD
 3150
 MINUSE, -305 /((7532) = FOR AMPERSAND
 DPCVPT, DECONV
 DCONP, DECON
 RESOL5, RESOLV
 DPN, DNUMBR
 XIN, INPUT
 INORM, DNORM
 P43, 43
 /END OF FLOATING POINT INPUT
 /7 FREE
 /USED BY H.S. READER
 /
 /CALLS LOADING ROUTINE FOR FILE
 /VARIABLES IN EXPRESSIONS; CALLED BY EFUN3;
 /
 *6311
 FNUM. TAD CHAR /*****
 DCA EFOP /*****
 GETC /*****
 SORTC /*****
 TERMS=1 /*****
 SKP /*****
 JMP ,=4 /*****
 TSTLPR /*****

2747 6321 4566
 2748 6322 4734
 2749 6323 4453
 2750 6324 3171
 2751 6325 1045
 2752 6326 3173
 2753 6327 1413
 2754 6330 6212
 2755 6331 4733
 2756 6332 5536
 2757 6333 1533
 2758 6334 1601
 2759 6335 0000
 2760 6336 4545
 2761 6337 1066
 2762 6340 4542
 2763 6341 4545
 2764 6342 4550
 2765 6343 1374
 2766 6344 5735
 2767 6345 5341
 2768 6346 4335
 2769 6347 1066
 2770 6350 1374
 2771 6351 7640
 2772 6352 5357
 2773 6353 1413
 2774 6354 4547
 2775 6355 6365
 2776 6356 7772
 2777 6357 4566
 2778 6360 5167
 2779 6361 5171
 2780 6362 2572
 2781 6363 5173
 2782 6364 6176
 2783 6365 6375
 2784 6366 0317
 2785 6367 0303
 2786 6370 0315
 2787 6371 0314
 2788 6372 0323
 2789 6373 0307
 2790 6374 7524
 2791 6375 6213
 2792 6376 5136
 2793 6400 6400
 2794
 2795 6400 0000
 2796 6401 7300
 2797 6402 3047
 2798 6403 3043
 2799 6404 1600
 2800 6405 7450
 2801 6406 5600

LOADIT;
 PECALL;
 PASS,

LTAPE,

LERR,
 LGO,

LLIST,

MINCOM;
 LG,

*6400

FPNT,

ERROR4 /*****
 JMS I PECALL /*****
 JMS I INTEGER /*****
 DCA SUBS /*****
 TAD HORD /*****
 DCA LESUBS /*****
 POPA /*****
 6212 /***** FILE NO.
 JMS I LOADIT /*****
 JMP I EFUN3I /*****
 ITLOAD /*****
 ECALL /*****
 GETC /*****
 TAD CHAR /*****
 PUSHA /*****
 GETC /*****
 SORTC /*****
 GLIST=1 /*****
 JMP I PASS /*****
 JMP I =4 /*****
 JMS PASS /*****
 TAD CHAR /*****
 TAD MINCOM /*****
 SZA CLA /*****
 JMP LERR /*****
 POPA /*****
 SORTJ /***** JMPS ON SUBCOMMAND OF LIBR XXXX
 LLIST=1 /*****
 LGO=LLIST /*****
 ERROR4 /*****
 LO /*****
 LC /*****
 LM /*****
 LL /*****
 LS /*****
 LG /*****
 317 /*****
 303 /*****
 315 /*****
 314 /*****
 323 /*****
 307 /*****
 =254 /*****
 6213 /*****
 JMP XLG /*****
 / FLOATING-POINT INTERPRETER FOR FOCAL.
 CLA CLL /*****
 DCA OVER2 /((NOP) = FOR 4-WORD
 DCA OVER1 /((NOP) = FOR 4-WORD.
 TAD I FPNT /GET NEXT INSTRUCTION
 SNA /*****
 JMP I FPNT /FAST EXIT

2802	6407	3264	DCA JUMP	
2803	6410	1264	TAD JUMP	
2804	6411	123	AND C202	/GET PAGE BIT
2805	6412	7650	SNA CLA	/PAGE ZERO?
2806	6413	5216	JMP ,+3	/YES
2807	6414	1134	TAD P7600	/NO
2808	6415	200	AND FPNT	/C(FPNT)0=4 CONTAINS PAGE BITS
2809	6416	3040	DCA ADDR	
2810	6417	1106	TAD P177	/GET 7 BIT ADDRESS
2811	6420	1264	AND JUMP	
2812	6421	1040	TAD ADDR	
2813	6422	3040	DCA ADDR	
2814	6423	1265	TAD INDRCT	/INDIRECT BIT=1?
2815	6424	1264	AND JUMP	
2816	6425	7650	SNA CLA	
2817	6426	5233	JMP LOOP01	/NO-GO ON
2818	6427	1440	TAD I ADDR	/YES ,DEFER ,W/O AUTO=INDEX
2819	6430	7450	SNA	/***** IF PT1 WAS ZERO, IT IS A
2820	6431	5572	JMP I LEFPUT	/***** FILE VARIABLE
2821	6432	3040	DCA ADDR	
2822	6433	2200	ISZ FPNT	
2823	6434	7040	CMA	
2824	6435	1040	TAD ADDR	
2825	6436	3015	DCA FLT XR2	
2826	6437	1264	TAD JUMP	/GET COMMAND
2827	6440	7106	CLL RTL	
2828	6441	7006	RTL	
2829	6442	1107	AND P17	/GET BITS 0=2:IE OPCODE
2830	6443	7450	SNA	
2831	6444	5271	JMP FLGT	
2832	6445	1266	TAD TABLE	/LOOKUP IN TABLE
2833	6446	3264	DCA JUMP	
2834	6447	1664	TAD I JUMP	
2835	6450	7450	SNA	
2836	6451	5267	JMP FLPT	
2837	6452	3264	DCA JUMP	
2838	6453	1306	TAD CEX1	/SAVE FLOATING ARGUMENT, UNLESS 'GET' OR 'PUT'
2839	6454	3014	DCA FLT XR	
2840	6455	1117	TAD MFLT	
2841	6456	3057	DCA CNTR	
2842	6457	1415	TAD I FLT XR2	
2843	6460	3414	DCA I FLT XR	
2844	6461	2057	ISZ CNTR	
2845	6462	5257	JMP ,+3	
2846	6463	5664	JMP I JUMP	/GO THERE
2847	6464	0000	JUMP, 0	
2848		0040	ADDR=EX1	
2849	6465	1400	INDRCT, 0400	
2850	6466	6575	TABLE, ITABLE	
2851	6467	1305	FLPT, TAD CEXP	/EXP TO (ADDR)
2852	6470	5275	JMP ,+5	
2853	6471	1305	FLGT, TAD CEXP	/(ADDR) TO EXP
2854	6472	3015	DCA FLT XR2	
2855	6473	7040	CMA	
2856	6474	1040	TAD ADDR	

LOOP01:

2857	6475	3014	DCI FLT XR	/SAVE 'FROM' ADDRESS
2858	6476	1117	TAD MFLT	/3 OR 4 WORDS
2859	6477	3057	DCA CNTR	
2860	6500	1414	TAD I FLT XR	
2861	6501	3415	DCA I FLT XR2	
2862	6502	2057	ISZ CNTR	
2863	6503	5300	JMP ,=3	
2864	6504	5201	JMP FPNT+1	
2865	6505	1043	CEXP, EXP=1	
2866	6506	1037	CX1, EX1=1	
2867	6507	4767	FLSU, JMS I OPMINS	/FSUP=2 - NEGATE THE OPERAND
2868	6510	4772	FLAD, JMS I ALGN	/FLAD=1 - FIRST ALIGN EXPONENTS
2869	6511	5201	JMP FPNT+1	/RETURN IF NO ALIGNMENT IS POSSIBLE
2870	6512	4774	JMS I RAR2	/TRIPLE PRECISION ADDITION
2871	6513	4773	JMS I RAR1	/SINCE BITS ARE SHIFTED
2872	6514	4775	JMS I TRAD	/RIGHT
2873	6515	4771	NORF, JMS I NORM	/NORMALIZE THE RESULT
2874	6516	5201	JMP FPNT+1	/HINTIUSE 700X FOR FUNCTIONS.
2875			/INTERPRETIVE POWER	
2876	6517	1045	FLEX, TAD HORD	/ZERO?
2877	6520	7200	CLA	/CROCK****
2878	6521	5327	JMP ,+6	
2879	6522	3044	ZERO, DCA EXP	/YES
2880	6523	3045	DCA HORD	
2881	6524	3046	DCA LORD	
2882	6525	3047	DCA OVER2	
2883	6526	5201	JMP FPNT+1	
2884	6527	4543	PUSHF	/AC TO A + POWER
2885	6530	1044	FLAC	
2886	6531	4543	PUSHF	/SETUP ARGUMENT (THE EXPONENT)
2887	6532	1040	EX1	
2888	6533	4544	POPF	
2889	6534	1044	FLAC	
2890	6535	4453	JMS I INTEGER	/ONLY POSITIVE, INTEGER EXPONENTS
2891	6536	7510	SPA	
2892	6537	5344	JMP ,+5	/(COULD DIVIDE)
2893	6540	7040	CMA	
2894	6541	3264	DCA JUMP	/TEMP STORAGE
2895	6542	3043	DCA OVER1	/(NOP) = FOR 4-WORD
2896	6543	1045	TAD HORD	
2897	6544	7640	SZA CLA	
2898	6545	4566	ERROR2	/TOO LARGE OR NEGATIVE EXPONENT
2899	6546	4543	PUSHF	/INITIALIZE TO ONE.
2900	6547	2405		
2901	6550	4544	POPF	FLTONE
2902	6551	1044	FLAC	
2903	6552	4544	POPF	
2904	6553	7470	ITER1	
2905	6554	5362	JMP ,+6	
2906	6555	4543	PUSHF	
2907	6556	7470	ITER1	
2908	6557	4544	POPF	
2909	6560	1040	EX1	
2910	6561	4770	JMS I MULT	/"MULT"
2911	6562	2264	ISZ JUMP	

2912 6563 5355
 2913 6564 5221
 2914 6565 4772
 2915 6566 5201
 2916 6567 7153
 2917 6570 7004
 2918 6571 7335
 2919 6572 6623
 2920 6573 5754
 2921 6574 6757
 2922 6575 5733
 2923 6575
 2924 6576 6510
 2925 6577 6507
 2926 6600 7107
 2927 6601 6565
 2928 6602 6517
 2929 6603 0000
 2930 6604 6515
 2931 6605 0000
 2932 6606 7200
 2933 6607 1047
 2934 6610 7161
 2935 6611 3047
 2936 6612 7004
 2937 6613 1046
 2938 6614 7061
 2939 6615 3046
 2940 6616 7004
 2941 6617 1045
 2942 6620 7061
 2943 6621 3045
 2944 6622 5605
 2945 6623 0000
 2946 6624 1045
 2947 6625 7450
 2948 6626 1046
 2949 6627 7650
 2950 6630 5311
 2951 6631 1041
 2952 6632 7450
 2953 6633 1042
 2954 6634 7450
 2955 6635 1043
 2956 6636 7650
 2957 6637 5623
 2958 6640 1040
 2959 6641 7041
 2960 6642 1044
 2961 6643 7450
 2962 6644 5273
 2963 6645 3205
 2964 6646 1205
 2965 6647 7500
 2966 6650 7041

JMP ,=6
 JMP FPNT+1
 FLMY, JMS I MULT /MULTIPLY
 JMP FPNT+1
 OPMINS; MINUS2
 MULT, DMULT
 NORM, DNORM
 ALGN, ALIGN
 RAR1, DIV1
 RAR2, DIV2
 YRAD, DUPLAD
 ITABLE, =1
 FLAD
 FLSU
 FLOV
 FLMY
 FLEX
 0000
 NORF
 ACMINS; 0 /ROUTINE TO COMPLEMENT FLAG - VIA "MINSKI"
 CLA /***** (IS THIS CLA NECESSARY)
 TAD OVER2 /***** RECODING FOR SPACE
 CLL CML CIA /*****
 DCA OVER2 /*****
 RAL /*****
 TAD LORD /*****
 CML CIA /*****
 DCA LORD /*****
 RAL /*****
 TAD HORD /*****
 CML CIA /*****
 DCA HORD /*****
 JMP I ACMINS
 ALIGN; 0 /SUBROUTINE TO ALIGN
 TAD HORD /BINARY POINTS
 SNA
 TAD LORD /IS MANTISSA ZERO?
 SNA CLA
 JMP NOX1 /YES, RESULT=OPERAND
 TAD AC1H /NO, IS OPERAND ZERO?
 SNA
 TAD AC1L
 SNA
 TAD OVER1
 SNA CLA
 JMP I ALIGN /YES, EXIT;
 TAD EX1
 CMA IAC
 TAD EXP
 SNA /ARE EXPONENTS EQUAL?
 JMP ADONE /YES
 DCA ACMINS
 TAD ACMINS
 SNA /NO
 CIA /NEGATE AND

2967	6651	3322	DCA	AMOUNT	/SAVE THE DIFFERENCE
2968	6652	1322	TAD	AMOUNT	
2969	6653	1336	TAD	TEST2	
2970	6654	7710	SPA	CLA	/CAN THE EXPONENTS BE ALIGNED?
2971	6655	5275	JMP	NOX	/NO, USE LARGER OF THE TWO.
2972	6656	1235	TAD	ACMINS	/YES, SHIFT THE SMALLER
2973	6657	7700	SMA	CLA	
2974	6660	5265	JMP	ASHFT	
2975	6661	4357	JMS	DIV2	
2976	6662	2322	ISZ	AMOUNT	
2977	6663	5261	JMP	,=2	
2978	6664	5273	JMP	ADONE	
2979	6665	7040	ASHFT,	CMA	
2980	6666	1040	TAD	EX1	
2981	6667	3040	DCA	EX1	
2982	6670	4723	JMS	I TAG1	
2983	6671	2322	ISZ	AMOUNT	
2984	6672	5270	JMP	,=2	
2985	6673	2223	ADONE,	ISZ ALIGN	
2986	6674	5623	JMP	I ALIGN	
2987	6675	1040	NOX,	TAD EX1	/MISSION IMPOSSIBLE!
2988	6676	7700	SMA	CLA	/CHECK FOR SIGN DIFFERENCE
2989	6677	5304	JMP	NOX2	
2990	6700	1044	TAD	EXP	
2991	6701	7700	SMA	CLA	
2992	6702	5623	JMP	I ALIGN	/==
2993	6703	5306	JMP	,+3	/==
2994	6704	1044	NOX2,	TAD EXP	
2995	6705	7700	SMA	CLA	
2996	6706	1205	TAD	ACMINS	/TEMP STORAGE OF DIFFERENCE, BOTH POS EXP OR BOTH NEG;
2997	6707	7740	SMA	SZA CLA	
2998	6710	5623	JMP	I ALIGN	/OK (+=)
2999	6711	1040	NOX1,	TAD EX1	/USE LARGER
3000	6712	3044	DCA	EXP	
3001	6713	1041	TAD	AC1H	
3002	6714	3045	DCA	HORD	
3003	6715	1042	TAD	AC1L	
3004	6716	3046	DCA	LORD	
3005	6717	1043	TAD	OVER1	
3006	6720	3047	DCA	OVER2	
3007	6721	5623	JMP	I ALIGN	
3008	6722	0000	AMOUNT,	0	
3009	6723	5754	TAG1,	DIV1	
3010			/LEAVE 12 BIT ANSWER IN AC UPON RETURN		
3011			/LEAVE FLAC AS AN INTEGER,		
3012	6724	0000	FIX,	0	/VIA (INTEGER)
3013	6725	4751	JMS	I ABSOL	
3014	6726	1044	TAD	EXP	/TEST FOR FRACTION
3015	6727	7750	SPA	SNA CLA	
3016	6730	5353	JMP	FIXM	/DOUBLE CHECK FOR MINUS ONE.
3017	6731	7701	IAC		
3018	6732	3043	DCA	OVER1	
3019	6733	1350	TAD	P27	/INIT ALIGNMENT
3020	6734	3040	DCA	EX1	
3021	6735	4223	JMS	ALIGN	/DO THE ALIGNMENT TO AN INTEGER

3022	6736	0027	TEST2,	0027	/ALREADY DONE! (43)=FOR 4=WORD
3023	6737	2047		ISZ OVER2	
3024	6740	5344		JMP ,+4	
3025	6741	2046		ISZ LORD	
3026	6742	7410		SKP	
3027	6743	2045		ISZ HORD	
3028	6744	3047		DCA OVER2	/CLEAR THE FRACTION
3029	6745	4752		JMS I RESOL	
3030	6746	1046		TAD LORD	/EXIT WITH LOW ORDER RESULT IN AC.
3031	6747	5724		JMP I FIX	
3032	6750	0027	P27,	27	
3033	6751	5571	ABSOL,	ABSOLV	
3034	6752	7173	RESOL,	RESOLV	
3035	6753	3044	FIXM,	DCA EXP	/CLEAR EXPONENT
3036	6754	3045		DCA HORD	
3037	6755	3046		DCA LORD	
3038	6756	5344		JMP TEST2+6	
3039	6757	0000	DIV2,	0	/SHIFT FLAG RIGHT
3040	6760	7300		CLA CLL	
3041	6761	1045		TAD HORD	
3042	6762	7510		SPA	
3043	6763	7020		CML	
3044	6764	7010		RAR	
3045	6765	3045		DCA HORD	
3046	6766	1046		TAD LORD	
3047	6767	7010		RAR	
3048	6770	3046		DCA LORD	
3049	6771	1047		TAD OVER2	
3050	6772	7010		RAR	
3051	6773	3047		DCA OVER2	
3052	6774	2044		ISZ EXP	
3053	6775	5757		JMP I DIV2	
3054	6776	5757		JMP I DIV2	
3055		6777	SPECIAL#		/INPUT CHARACTERS
3056	6777	0337		337	/LEFT ARROW
3057	7000	0377		377	/RUBOUT
3058	7001	0212		212	/L.F.
3059	7002	0375		375	/ALT MODE
3060	7003	7777		=1	
3061					/(A+B*C)*(D+E+F)=A*D,A*E,B*D,B*E
3062	7004	0000	DMULT,	0	/N= PRECISION MULTIPLY WITH
3063	7005	7001		IAC	/PRODUCT IN TRIPLE PRECISION
3064	7006	1040		TAD EX1	/ADD EXPONENTS+1
3065	7007	4324		JMS SIGN	/AND DETERMINE SIGN OF RESULT
3066	7010	7710		SPA CLA	
3067	7011	4353		JMS MINUS2	
3068	7012	3301		DCA DATUM=1	/INITIALIZE RESULT
3069	7013	3300		DCA DATUM=2	
3070	7014	3277		DCA DATUM=3	
3071	7015	3276		DCA DATUM=4	
3072	7016	1045		TAD A	/A*D
3073	7017	3751		SAVE	/STORE IN MP2
3074	7020	1041		TAD D	/SINGLE PRECISION MULTIPLY
3075	7021	4752		MULTY	
3076	7022	0002		2	/ACCUMULATE START IN #2 DATA WORD

3077	7023	1042	TAD E	/A*E
3078	7024	4752	MULTY	
3079	7025	003	3	
3080	7026	1046	TAD B	/B*D
3081	7027	3751	SAVE	
3082	7030	1241	TAD D	
3083	7031	4752	MULTY	
3084	7032	003	3	
3085	7033	1042	TAD E	/B*E
3086	7034	4752	MULTY	
3087	7035	1004	4	
3088	7036	5263	DMULT4, JMP DMDONE	/(DCA DATUM=5)-FOR 4-WORD
3089	7037	3274	DCA DATUM=6	
3090	7040	1043	TAD F	/A*F
3091	7041	3751	SAVE	
3092	7042	1045	TAD A	
3093	7043	4752	MULTY	
3094	7044	004	4	
3095	7045	1046	TAD B	/B*F
3096	7046	4752	MULTY	
3097	7047	005	5	
3098	7050	1047	TAD C	/C*D
3099	7051	3751	SAVE	
3100	7052	1041	TAD D	
3101	7053	4752	MULTY	
3102	7054	004	4	
3103	7055	1042	TAD E	/C*E
3104	7056	4752	MULTY	
3105	7057	005	5	
3106	7060	1043	TAD F	/C*F
3107	7061	4752	MULTY	
3108	7062	1006	6	
3109	7063	1301	DMDONE, TAD DATUM=1	/COPY RESULT
3110	7064	3045	DCA HORO	
3111	7065	1300	TAD DATUM=2	
3112	7066	3046	DCA LORD	
3113	7067	1277	TAD DATUM=3	
3114	7070	3047	DCA OVER2	
3115	7071	4301	JMS MULDIV	
3116	7072	3047	DCA OVER2	/(NOP) - FOR 4-WORD
3117	7073	5604	JMP I DMULT	
3118	7102		DATUM=+6 /INTERMEDIATE STORAGE	
3119	7074	000	0/#6=LOW ORDER RESULT	
3120	7075	000	0/#5	
3121	7076	000	0/#4	
3122	7077	000	0/#3	
3123	7100	000	0/#2	
3124			/#1=HIGH ORDER RESULT	
3125	7101	000	MULDIV, 0 /TERMINATE MULTIPLY AND DIVIDE,	
3126	7102	2050	ISE SIGNF /CORRECT FOR SIGN	
3127	7103	4451	JMS I MINSKI	
3128	7104	4747	JMS I NORMF	/SHIFT LEFT
3129	7105	7000	NOP	/•
3130	7106	5701	JMP I MULDIV	
3131	7107	1041	FLDV, TAD AC1H	/4IDIVIDE

3132	7110	7657	SNA CLA	
3133	7111	4566	ERROR2	/DIVISION BY ZERO
3134	7112	1040	TAD EX1	/SUBTRACT EXPONENTS+1
3135	7113	7041	CMA IAC	
3136	7114	7001	IAC	
3137	7115	4324	JMS SIGN	/SET UP SIGNS
3138	7116	7700	SMA CLA	
3139	7117	4353	JMS MINUS2	/NEGATE DIVISOR
3140	7120	4750	JMS I DIVIDE	/DIVIDE
3141	7121	4301	JMS MULDIV	
3142	7122	5723	JMP I .+1	
3143	7123	6401		
3144			FPNT+1	
3145			/THIS SUBROUTINE PREPARES MULTIPLY AND DIVIDE	
3146			/FOR ANY COMBINATION OF SIGNED ARGUMENTS AND FOR ZERO,	
3147			/THE RESULT OF EITHER IS ZERO IF FLAG = 0,	
3148			/RESULT OF MULTIPLY IS ZERO IF EITHER IS ZERO;	
3149			/DIVISION BY ZERO IS CHECKED BEFORE THIS	
3150			/ROUTINE IS CALLED.	
3151			/THE CALLING AC CONTAINS AN UPDATE VALUE FOR THE	
3152			/EXPONENT, THE RETURNING AC CONTAINS THE SIGN OF	
3153			/THE ARGUMENT FOR FURTHER TESTING BY EACH ROUTINE,	
3153	7124	0000	SIGN, 0	/TEST AND SAVE SIGN OF RESULT
3154	7125	1044	TAD EXP	/COMPUTE NEW EXPONENT FOR MUL=DIV.
3155	7126	3044	DCA EXP	
3156	7127	1124	TAD P4000	/LOAD 4000 TO XOR THE SIGN BITS.
3157	7130	0045	AND HORD	
3158	7131	1041	TAD AC1H	
3159	7132	7700	SMA CLA	/RESULT MAY BE ZERO
3160	7133	7040	CMA	
3161	7134	3050	DCA SIGNF	
3162	7135	1045	TAD HORD	
3163	7136	7450	SNA	
3164	7137	5746	JMP I REVIT	/ANSWER IS ZERO.
3165	7140	7710	SPI CLA	/TAKE ABSOLUTE VALUE OF FLAG
3166	7141	4451	JMS I MINSKI	
3167	7142	1041	TAD AC1H	
3168	7143	7450	SNA	/RESULT OF EITHER MAY BE ZERO
3169	7144	5746	JMP I REVIT	
3170	7145	5724	JMP I SIGN	
3171			/SIGN OF RESULT = SIGNF	
3172			/==1	
3173			/==2	
3174	7146	6522	REVIT, ZERO	
3175	7147	7335	NORMF, DNORM	
3176	7150	7261	DIVIDE, OURDIV	
3177		3751	SAVE=DCA I .	
3178	7151	7256	MP2	
3179		4752	MULTY=JMS I .	
3180	7152	7200	MP4	
3181		0045	A=FLAC+1	
3182		0046	B=FLAC+2	
3183		0047	C=FLAC+3	
3184		0041	D=AC1H	
3185		0042	E=AC1L	
3186		0043	F=OVER1	

3187	7153	000	MINUS2,		/NEGATE OPERAND
3188	7154	7302		CLA CLL	/TRIPLE PRECISION
3189	7155	1043		TAD OVER1	
3190	7156	7041		CMA IAC	
3191	7157	3043		DCA OVER1	
3192	7160	1042		TAD AC1L	
3193	7161	7240		CMA	
3194	7162	7430		SZL	
3195	7163	7101		IAC CLL	
3196	7164	3042		DCA AC1L	
3197	7165	1041		TAD AC1H	
3198	7166	7240		CMA	
3199	7167	7430		SZL	
3200	7170	7101		IAC CLL	
3201	7171	3041		DCA AC1H	
3202	7172	5753		JMP I MINUS2	
3203	7173	0000	RESOLV,	0	
3204	7174	1050		TAD SIGNF	
3205	7175	7710		SPA CLA	
3206	7176	4451		JMS I MINSKI	
3207	7177	5773		JMP I RESOLV	
3208		7200	*7200		
3209	7200	0000	MP4,	0	/SINGLE PRECISION, UNSIGNED MULTIPLY = "MULTY"
3210	7201	7450		SNA	/NO RESULT ADDED IF ZERO
3211	7202	5600		JMP I MP4	
3212					/FOR EAE INSERT THE FOLLOWING:
3213			/7203	3206	DCA ,+3
3214			/7204	1256	TAD MP2
3215			/7205	7425	MQL MUY
3216			/7206	0000	0
3217			/7207	3253	DCA MP5
3218			/7210	7501	MQA
3219			/7211	3255	DCA MP3
3220			/7212	5227	JMP ,+15
3221	7203	3254		DCA MP1	/12 BITS BY 12 BITS
3222	7204	3253		DCA MP5	
3223	7205	1257		TAD THIR	
3224	7206	3255		DCA MP3	
3225	7207	7100		CLL	
3226	7210	1254	MP6,	TAD MP1	
3227	7211	7010		RAR	
3228	7212	3254		DCA MP1	
3229	7213	1253		TAD MP5	
3230	7214	7420		SNL	
3231	7215	5220		JMP ,+3	
3232	7216	7100		CLL	
3233	7217	1256		TAD MP2	
3234	7220	7010		RAR	
3235	7221	3253		DCA MP5	/SAVE HIGH ORDER RESULT
3236	7222	2255		ISZ MP3	
3237	7223	5210		JMP MP6	
3238	7224	1254		TAD MP1	/CORRECT LOW ORDER RESULT
3239	7225	7010		RAR	
3240	7226	3255		DCA MP3	
3241	7227	1600		TAD I MP4	/PICKUP SCALE FACTOR

3242	7237	7141	CIÄ	
3243	7231	1252	TAD DATUMA	/COMPUTE ADDRESS
3244	7232	3254	DCÄ MP1	/TEMP
3245	7233	1255	TAD MP3	/LOW ORDER PART
3246	7234	7100	CLL	
3247	7235	1654	TAD I MP1	/ACCUMULATE
3248	7236	3654	DCÄ I MP1	
3249	7237	2254	ISZ MP1	
3250	7242	7004	RAL	
3251	7241	1253	TAD MP5	
3252	7242	1654	TAD I MP1	
3253	7243	3654	DCÄ I MP1	
3254	7244	7420	SNL	
3255	7245	5600	JMP I MP4	/NO CARRY
3256	7246	2254	ISZ MP1	
3257	7247	2654	ISZ I MP1	
3258	7250	5600	JMP I MP4	/EXIT
3259	7251	5246	JMP ,=3	/CARRY AGAIN
3260			/////	
3261	7252	7102	DATUMA; DATUM	
3262	7253	0000	MP5, 0	/PRODUCT
3263	7254	0000	MP1, 0	/MULTIPLIER
3264	7255	0000	MP3, 0	
3265	7256	0000	MP2, 0	/MULTIPLICAND
3266	7257	7764	THIR, -14	/12 BITS
3267	7260	7751	MIF, -27	/(-43) - FOR 4=WORD(=7735)
3268	7261	0000	DUBDIV, 0	/2 OR 3 PRECISION DIVIDE
3269	7262	3200	DCÄ MP4	
3270	7263	3254	DCÄ MP1	
3271	7264	1260	TAD MIF	/INIT BIT COUNTER
3272	7265	3255	DCÄ MP3	
3273	7266	7410	SKP	
3274	7267	4527	DV3, JMS I DOUBLE	/SHIFT FLAG LEFT
3275	7270	7100	CLL	
3276	7271	1042	TAD AC1L	/COMBINE ONE POSITION AND (4=WORD)
3277	7272	1046	TAD LORD	
3278	7273	3256	DCÄ MP2	/SAVE RESULT
3279	7274	7004	RAL	
3280	7275	1045	TAD HORD	/ADD OVERFLOW
3281	7276	1041	TAD AC1H	
3282	7277	7420	SNL	/SKIP IF OVERFLOW
3283	7300	5304	JMP ,+4	
3284	7301	3045	DCÄ HORD	/UPDATE FLAG
3285	7302	1256	TAD MP2	
3286	7303	3046	DCÄ LORD	
3287	7304	7200	CLÄ	/CLEAR ACCUMULATOR
3288	7305	1254	TAD MP1	/SAVE OVERFLOW BITS CIRCULARLY
3289	7306	7004	RAL	
3290	7307	3254	DCÄ MP1	
3291	7310	1200	TAD MP4	
3292	7311	7004	RAL	
3293	7312	3200	DCÄ MP4	
3294	7313	2255	ISZ MP3	/TEST FOR END OF DIVIDE
3295	7314	5267	JMP DV3	
3296	7315	1254	TAD MP1	/LOAD RESULTS

3297	7316	3046	DCA LORD	
3298	7317	1200	TAD MP4	
3299	7320	3045	DCA HORD	
3300	7321	5661	JMP I DUBDIV	/(NOP)
3301	7322	7004	RAL	/EXTRA FOR 4-WORD
3302	7323	3335	DCA DNORM	
3303	7324	2255	ISE MP3	/TEST FOR END OF DIVIDE
3304	7325	5267	JMP DV3	
3305	7326	1335	TAD DNORM	
3306	7327	3045	DCA HORD	
3307	7330	1200	TAD MP4	
3308	7331	3046	DCA LORD	
3309	7332	1254	TAD MP1	
3310	7333	3047	DCA OVER2	
3311	7334	5661	JMP I DUBDIV	
3312	7335	0000	DNORM, 0	/SUBROUTINE TO NORMALIZE FLAG
3313	7336	4775	JMS I ABSOL3	
3314	7337	4366	JMS TEST4	
3315	7340	1045	TAD HORD	
3316	7341	7450	SNA	/IS MANTISSA=0?
3317	7342	1047	TAD OVER2	
3318	7343	7450	SNA	
3319	7344	1046	TAD LORD	
3320	7345	7650	SNA CLA	
3321	7346	5363	JMP EXIT3	/YES
3322	7347	1045	TAD HORD	
3323	7350	7104	RAL CLL	
3324	7351	7710	SPA CLA	/WILL SHIFT BE TOO FAR?
3325	7352	5360	JMP ,+6	
3326	7353	4527	JMS I DOUBLE	
3327	7354	7140	CMA CLL	
3328	7355	1044	TAD EXP	
3329	7356	3044	DCA EXP	
3330	7357	5347	JMP ,=10	
3331	7360	4776	JMS I RESOL3	
3332	7361	4366	JMS TEST4	/DON'T LEAVE 4000
3333	7362	5735	JMP I DNORM	
3334	7363	3044	EXIT3, DCA EXP	/SET TO ZERO
3335	7364	5735	JMP I DNORM	/RETURN
3336	7365	6757	XRAR2, DIV2	
3337	7366	0000	TEST4, 0	
3338	7367	1045	TAD HORD	/TEST FOR 4000
3339	7370	7510	SPA	
3340	7371	7041	CIA	
3341	7372	7710	SPA CLA	
3342	7373	4765	JMS I XRAR2	/SHIFT BACK
3343	7374	5766	JMP I TEST4	
3344	7375	5571	ABSOL3, ABSOLV	
3345	7376	7173	RESOL3, RESOLV	
3346		7400	*7400	
3347			/PAGE 18	
3348			/FLOATING SQUARE ROOT FUNCTION	
3349	7400	4437	XSORT, FINT	
3350	7401	6274	FPUT FPAC1	/VALUE
3351	7402	0000	FEXT	/NEWTON'S METHOD IS USED

3352	7403	1345	GETSGN	
3353	7404	7710	SPÄ CLA	
3354	7405	4566	ERROR2	/NUMBER IS NEGATIVE=IMAGINARY ROOTS
3355	7406	1044	TAD EXP	/LINK IS =0 FROM FINT
3356	7407	7510	SPA	/MATCH THE SIGN WITH LINK BIT
3357	7410	7020	CML	
3358	7411	7010	RAR	
3359	7412	3270	DCÄ ITER1	/MAKE FIRST APPROXIMATION
3360	7413	7430	SZL	/TEST LSB OF EXP
3361	7414	2270	ISZ ITER1	
3362	7415	7000	07000, NOP	/*****
3363	7416	1267	TAD SQCON1	
3364	7417	3271	DCÄ ITER1+1	
3365	7420	3272	DCÄ ITER1+2	
3366	7421	3273	DCÄ ITER1+3	
3367	7422	1275	TAD FPAC1+1	
3368	7423	7450	SNÄ	
3369	7424	1276	TAD FPAC1+2	
3370	7425	7650	SNÄ CLA	
3371	7426	5265	JMP SQEND	/NUMBER=0
3372	7427	4407	CLCU, FINT	
3373	7430	1274	FGET FPAC1	
3374	7431	3270	FDIV ITER1	
3375	7432	1270	FADD ITER1	
3376	7433	0000	FEXT	
3377	7434	7240	CLÄ CMA	
3378	7435	1044	TAD EXP	
3379	7436	3044	DCÄ EXP	
3380	7437	1044	TAD EXP	
3381	7440	7041	CMA IAC	
3382	7441	1270	TAD ITER1	
3383	7442	7640	SZÄ CLA	/ARE EXPONENTS EQUAL?
3384	7443	5261	JMP ROOTGO	/NO
3385	7444	1045	TAD HORD	/ARE HIGH-ORDER MANTISSAS EQUAL?
3386	7445	7041	CMA IAC	
3387	7446	1271	TAD ITER1+1	
3388	7447	7640	SZÄ CLA	
3389	7450	5261	JMP ROOTGO	/NO
3390	7451	1046	TAD LORD	
3391	7452	7041	CMA IAC	
3392	7453	1272	TAD ITER1+2	/DO LOW-ORDER MANTISSAS AGREE
3393	7454	7500	SMÄ	
3394	7455	7041	CMA IAC	/WITHIN ONE BIT?
3395	7456	7001	IAC	
3396	7457	7700	SMÄ CLA	
3397	7460	5536	RETURN	
3398	7461	4407	ROOTGO, FINT	
3399	7462	6270	FPUT ITER1	
3400	7463	1000	FEXT	
3401	7464	5227	JMP CLCU	
3402	7465	3044	SQEND, DCÄ EXP	
3403	7466	5536	RETURN	
3404	7467	3015	SQCON1, 3015	
3405	7470	7470	BUFFER=,	
3406	7470	0000	ITER1, 0	

3407	7471	8000			
3408	7472	8000			
3409	7473	8000			
3410	7474	8020	FDAL1,	0	
3411	7475	8000		0	
3412	7476	8000		0	
3413	7477	7503			
3414					
3415	7520	8000	SCOPOU;	0	
3416	7501	1106	AND	P177	
3417	7502	1367	TAD	07763	
3418	7503	7440	SZA		
3419	7504	5310	JMP	NOCRLF	
3420	7505	3364	CRLF,	DCA	NCOLS
3421	7506	2365		ISZ	NFEEDS
3422	7507	5321		JMP	ITSOK
3423	7510	1371	NOCRLF;	TAD	07655
3424	7511	7100		CLL	
3425	7512	1006		TAD	C100
3426	7513	7420		SNL	
3427	7514	7610		SKP	CLA
3428	7515	1361		TAD	NLINES
3429	7516	7450		SNA	
3430	7517	5700		JMP	I SCOPOU
3431	7520	2364		ISZ	NCOLS
3432	7521	6002	ITSOK;	IOF	
3433	7522	6141		LINC	
3434					
3435	7523	1644	LMODE		
3436	7524	1362		LDF	4
3437	7525	8011		STH	I OPTR
3438	7526	8002		CLR	
3439				PDP	
3440	7527	6201	PMODE		
3441	7530	2366		6201	
3442	7531	1366		ISZ	NCHARS
3443	7532	1215		TAD	NCHARS
3444	7533	7710		TAD	07000
3445	7534	1361		SPA	CLA
3446	7535	1365		TAD	NLINES
3447	7536	7710		TAD	NFEEDS
3448	7537	5356		SPA	CLA
3449	7540	1366		JMP	NOHANG
3450	7541	6213		TAD	NCHARS
3451	7542	4020		6213	
3452	7543	6031		JMS	WAITER
3453	7544	5340		KSF	
3454	7545	6034		JMP	.04
3455	7546	1372		KRS	
3456	7547	7650		TAD	07566
3457	7550	6032		SNA	CLA
3458	7551	1370		KCC	
3459	7552	3774		TAD	06377
3460	7553	3366		DCA	I PPTR
3461	7554	3365		DCA	NCHARS
				DCA	NFEEDS

/ADDRESS OF NEXT FREE LOCATION IN 10-DIGIT VERSION.

/07503 /*****

/***** OUTPUT ROUTINE FOR SCOPE

/***** STORES CHARS IN FLD1, LOCS 400-777

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

/*****

TOO MANY LINES/CHARS DISPLAYED

HANG ON DISPLAY UNTIL SOMETHING IS TYPED

IGNORE LINE FEED

CLEAR

THE

CHARACTER


```

3462 7555 3364          DCA      NCOLS  /***** DISPLAY
3463 7556 6001  NOHANG; ION      /*****
3464 7557 1364          TAD      NCOLS  /*****
3465 7560 1373          TAD      07716 /*****
3466 7561 7740  NLINES; SMA SZA CLA /*****
3467 7562 5305          JMP      CRLF  /*****
3468 7563 5700          JMP I   SCOPOU /*****
3469 7564 0000  NCOLS; 0      /*****
3470 7565 0000  NFEEDS; 0     /*****
3471 7566 0000  NCHARS; 0     /*****
3472 7567 7763  07763; 7763
3473 7570 6377  06377; 6377
3474 7571 7655  07655; 7655  /*****
3475 7572 7566  07566; 7566  /*****
3476 7573 7716  07716; 7716  /*****
3477 7574 6002  PPTR;  OPTR  /*****
3478          *7576  /*****
3479          /
3480          /FDIS FUNCTION - STORES 2 WORDS
3481          /PER CALL IN 2200 THRU 5777 IN FLD1
3482          /
3483 7576 4453  CALLIN; JMS I  INTEGER /*****
3484 7577 6213          6213  /*****
3485 7600 5601          JMP I   .01  /*****
3486 7601 2071          INCALL /*****
3487 7602 4407  XDISP; FINT  /*****
3488 7603 4251          FMUL   FORHUN /*****
3489 7604 0000  FEXT  /*****
3490 7605 4453  JMS I  INTEGER /*****
3491 7606 7510  SPA  /*****
3492 7607 7041  CIA  /*****
3493 7610 3350  DCA  STAMP /*****
3494 7611 1066  TAD  CHAR  /*****
3495 7612 1256  TAD  MMCOM  /*****
3496 7613 7640  SZA CLA /*****
3497 7614 4566  ERROR3 /*****
3498 7615 4540  PUSHJ  /*****
3499 7616 1612  EVAL=1 /*****
3500 7617 4407  FINT  /*****
3501 7620 4253  FMUL   FIVHUN /*****
3502 7621 0000  FEXT  /*****
3503 7622 4453  JMS I  INTEGER /*****
3504 7623 3351  DCA  STAMP2 /*****
3505 7624 1271  TAD  SPTR  /*****
3506 7625 1247  TAD  MLIMIT
3507 7626 7650  SNA CLA
3508 7627 4566  ERROR3
3509 7630 6002  IOF  /*****
3510 7631 6211  6211 /CDF 10 /*****
3511 7632 7350  CLA CLL CMA RAR
3512 7633 0350  AND  STAMP
3513 7634 3671  DCA I  SPTR  /*****
3514 7635 2271  ISZ  SPTR  /*****
3515 7636 1351  TAD  STAMP2 /*****
3516 7637 1250  TAD  07400  /*****

```

3517	7642	3671	DCA I	SPTR	/*****
3518	7641	2271	ISE	SPTR	/*****
3519	7642	7240	CLA CMA		/*****
3520	7643	3671	DCA I	SPTR	/*****
3521	7644	6201	6201	/CDF 0	/*****
3522	7645	6001	ION		/*****
3523	7646	5536	JMP I	EFUN3I	/*****
3524	7647	2202	MLIMIT;	-5776	/(-LAST LOC OF DISP POINTS=1)
3525	7650	7400	07400;	7400	/*****
3526	7651	011	FORHUN;	1112700	/*****
	7652	2700			
3527	7653	011	FIVHUN;	111377010	/*****
	7654	3770			
	7655	0000			
3528	7656	7524	MMCOM;	-254	/*****
3529			/		
3530			/JMS WAIT IS EQUIVALENT		
3531			/TO JMP .=2 WITH A REFRESH OF		
3532			/THE DISPLAY ON THE WAY		
3533			/		
3534	7657	0000	WAIT;	0	/*****
3535	7660	7346	CLA CLL	CMA RTL	/*****
3536	7661	1257	TAD	WAIT	/*****
3537	7662	3257	DCA	WAIT	/*****
3538	7663	6002	IOF		/*****
3539	7664	1732	TAD I	PNCHARS	/*****
3540	7665	6213	6213	/CIF CDF 10	/*****
3541	7666	4020	JMS	WAITER	/*****
3542	7667	6001	ION		/*****
3543	7670	5657	JMP I	WAIT	/*****
3544	7671	1000	SPTR;	1000	/*****
3545	7672	0000	CLEAR;	0	/***** CLEAR POINTS FROM THE SCOPE
3546	7673	1304	TAD	ODISSP	/*****
3547	7674	3271	DCA	SPTR	/*****
3548	7675	6002	IOF		/*****
3549	7676	6211	6211	/CDF 10	/*****
3550	7677	7240	CLA CMA		/*****
3551	7700	3671	DCA I	SPTR	/*****
3552	7701	6201	6201	/CDF 0	/*****
3553	7702	6001	ION		/*****
3554	7703	5672	JMP I	CLEAR	/*****
3555	7704	2200	ODISSP;	2200	/***** (FORST LOC OF DISP POINTS)
3556	7705	6335	PPASS;	PPASS	
3557	7706	4705	OUTPUT;	JMS I	PPASS
3558	7707	1413	POPA	/*****	JUMPS ON SUBCOMMAND OF OUTPUT XXX
3559	7710	4547	SORTJ	/*****	
3560	7711	7721	OLIST=1	/*****	
3561	7712	7772	OGO=OLIST	/*****	
3562	7713	4566	OERROR;	ERROR3	/*****
3563	7714	7752	OGO,	OC	/*****
3564	7715	7761		OD	/*****
3565	7716	7753		OE	/*****
3566	7717	7763		OS	/*****
3567	7720	7771		OT	/*****
3568	7721	7734		OI	/*****

3569	7722	303	CLIST,	303	/*****			
3570	7723	304		304	/*****			
3571	7724	305		305	/*****			
3572	7725	323		323	/*****			
3573	7726	324		324	/*****			
3574	7727	311		311	/*****			
3575	7730	6377	006377,	6377	/*****			
3576	7731	611	OEXIT,	PROG	/*****			
3577	7732	7566	PNCHARS,	NCHARS	/*****			
3578	7733	6002	POPTR,	OPTR	/*****			
3579	7734	1066	OI,	TAD	CHAR	/*****		
3580	7735	1256		TAD	MMCOM	/*****		
3581	7736	7650		SNA	CLA	/*****		
3582	7737	5746		JMP	I	PSETCLK	/*****	O I, EXPRESSION
3583	7740	2745		ISZ	I	PCLKFLG	/*****	
3584	7741	1745		TAD	I	PCLKFLG	/*****	
3585	7742	7640		SEZ	CLA		/*****	
3586	7743	4257		JMS		WAIT	/*****	
3587	7744	5731		JMP	I	OEXIT		
3588	7745	2661	PCLKFLG,	CLKFLG		/*****		
3589	7746	5351	PSETCLK,	SETCLK		/*****		
3590		7750	*7750			/*****		
3591	7750	0000	STEMP,	0		/*****		
3592	7751	0000	STEMP2,	0		/*****		
3593	7752	4575	OC,	JMS	I	PCLEAR	/*****	
3594	7753	3732	OE,	DCA	I	PNCHARS	/*****	
3595	7754	1330		TAD		006377	/*****	
3596	7755	3733		DCA	I	POPTR	/*****	
3597	7756	3777		DCA	I	PNFEEDS	/*****	
3598	7757	3776		DCA	I	PNCOLS	/*****	
3599	7760	5731		JMP	I	OEXIT	/*****	
3600	7761	7000	OD,	NOP		/*****		
3601	7762	4257		JMS		WAIT	/*****	
3602	7763	6002	OS,	IOF		/*****		
3603	7764	6141				6141	/LINC	/*****
3604	7765	0004				0004	/ESF	/*****
3605	7766	0002				0002	/PDP	/*****
3606	7767	6001		ION			/*****	
3607	7770	1375		TAD		PSCOPOU	/*****	SET OUTDEV TO SCOPOU
3608	7771	1374	OT,	TAD		PXOUTL	/*****	SET OUTDEV TO XOUTL
3609	7772	3063		DCA		OUTDEV	/*****	
3610	7773	5731		JMP	I	OEXIT	/*****	
3611	7774	2676	PXOUTL,	XOUTL		/*****		
3612	7775	4632	PSCOPOU,	SCOPOU=XOUTL		/*****		
3613	7776	7564	PNCOLS,	NCOLS		/*****		
3614	7777	7565	PNFEEDS,	NFEEDS		/*****		
3615		0001	FIELD	1		/*****		

3616		001	*1	/*****
3617	0001	000	XQ,	/*****
3618	0002	400	D256, 400	/(REFERENCED AS LOC 2)
3619	0003	200	0200, 200	/(REFERENCED AS LOC 3)
3620	0004	125	085, 125	/(REFERENCED AS LOC 4)
3621	0005	000	GAMMA, 0	/*****
3622	0006	000	CHRCNT, 0	/*****
3623	0007	360	0360, 360	/*****
3624		010	*10	/*****
3625	0010	000	XQ1, 0	/*****
3626	0011	000	BLK2, 0	/UNIT
3627	0012	000	0	/ADDRESS
3628	0013	000	0	/BLOCK NUMBER
3629	0014	001	1	/NUMBER OF BLOCKS
3630	0015	760	0760, 760	/*****
3631	0016	000	ALPHA, 0	/*****
3632	0017	000	BETA, 0	/*****
3633		020	*20	/*****
3634			/	
3635			/ENTERED WITH NO. CHARS IN ACJ REFRESH	
3636			/FOR CHARS AND POINTS	
3637			/	
3638	0020	000	WAITER, 0	/*****
3639	0021	7450	SNÄ	/*****
3640	0022	5061	JMP NOASCII	/*****
3641	0023	7040	CMA	/*****
3642	0024	3006	DCÄ CHRCNT	/*****
3643	0025	1076	TAD 04377	/*****
3644	0026	3005	DCÄ GAMMA	/*****
3645	0027	1007	TAD 0360	/*****
3646	0030	3077	DCÄ Y	/*****
3647	0031	3001	DCÄ XQ	/*****
3648	0032	6141	LINC	/*****
3649			LMODE	
3650	0033	1325	CHRLUP; LDH I GAMMA	/*****
3651	0034	0450	AZE	/*****
3652	0035	6045	JMP GOODY	/*****
3653	0036	2077	ADD Y	/*****
3654	0037	2015	ADD 0760	/*****
3655	0040	1560	BCL I	/*****
3656	0041	7000	7000	/*****
3657	0042	4077	STC Y	/*****
3658	0043	4001	STC XQ	/*****
3659	0044	6056	JMP CHREND	/*****
3660	0045	0241	GOODY; ROL 1	/*****
3661	0046	2003	ADD 0200	/*****
3662	0047	4016	STC ALPHA	/*****
3663	0050	2077	ADD Y	/*****
3664	0051	1756	DSC ALPHA	/*****
3665	0052	1776	DSC I ALPHA	/*****
3666	0053	0221	XSK I XQ	/*****
3667	0054	0221	XSK I XQ	/*****
3668	0055	0011	CLR	/*****
3669	0056	0226	CHREND; XSK I CHRCNT	/*****
3670	0057	6033	JMP CHRLUP	/***** ONE TIME PER CHAR

3671	0060	467	SKP	/*****	
3672	0061	6141	NOASCII, LINC	/*****	
3673	0062	077	SET I BETA	/*****	
3674	0063	2200	2200	/*****	
3675	0064	645	LDF 5	/*****	
3676	0065	6102	JMP SUBR	/*****	
3677	0066	077	SET I BETA	/*****	
3678	0067	2000	2000	/*****	
3679	0070	0646	LDF 6	/*****	
3680	0071	6102	JMP SUBR	/*****	
3681	0072	0002	WEXIT, PDP	/*****	
3682			PNODE		
3683	0073	6203	6203 /CIF CDF 0	/*****	
3684	0074	7200	CLA	/*****	
3685	0075	5420	JMP I WAITER	/*****	
3686	0076	4377	04377, 4377	/*****	
3687	0077	0000	Y, 0	/*****	
3688	0100	171	PSUBS, SUBS	/*****	
3689	0101	0173	PLESUBS, LESUBS	/*****	
3690			LNODE	/*****	
3691	0102	0056	SUBR, SET ALPHA	/*****	DISPLAYS POINTS
3692	0103	0000	0000	/*****	
3693	0104	0415	KST	/*****	
3694	0105	0467	SKP	/*****	
3695	0106	6072	JMP WEXIT	/*****	
3696	0107	0500	IOB	/*****	
3697	0110	6041	TSF	/*****	
3698	0111	0467	SKP	/*****	
3699	0112	6072	JMP WEXIT	/*****	
3700	0113	1017	LDA BETA	/*****	
3701	0114	0467	SKP	/*****	
3702	0115	1037	WAITLP, LDA I BETA	/*****	
3703	0116	0451	AP0	/*****	
3704	0117	6072	JMP WEXIT	/*****	
3705	0120	4005	STC GAMMA	/*****	
3706	0121	1037	LDA I BETA	/*****	
3707	0122	0145	DIS GAMMA	/*****	
3708	0123	0217	XSK BETA	/*****	
3709	0124	6115	JMP WAITLP	/*****	
3710	0125	6016	JMP ALPHA	/*****	
3711			PNODE	/*****	
3712	0126	5527	XLO, JMP I ,+1	/*****	
3713	0127	1431	LOPEN	/*****	
3714	0130	5531	XLG, JMP I ,+1	/*****	
3715	0131	1520	LCLOSE	/*****	
3716	0132	5533	XLL, JMP I ,+1	/*****	
3717	0133	1203	LLOAD	/*****	
3718	0134	5535	XLS, JMP I ,+1	/*****	
3719	0135	1233	LSAVE	/*****	
3720	0136	5537	XLG, JMP I ,+1	/*****	
3721	0137	1202	LCHAIN	/*****	
3722	0140	7774	X7774, 7774		
3723	0141	7775	X7775, 7775		
3724	0142	1171	PLNUM, LNUM		
3725	0143	1000	PGETRHS, GETRHS		

3726	0144	1160	PLOMILD, LOMILD
3727	0145	1177	P5LNAM, LNAME*5
3728	0146	1230	P6LNAM, LNAME*6
3729	0147	0000	CHFLAG, 0
3730	0150	0000	HISS, 0
3731	0151	0000	LOSS, 0
3732	0152	2135	PFILTAB, FILTAB
3733	0153	1342	PLOOKUP, LUKUP
3734	0154	1600	PCOMMON, COMMON
3735	0155	1361	PREPLAC, REPLACE
3736	0156	0000	MYTE*P, 0
3737	0157	0000	MYTMP2, 0
3738	0160	2076	PFINISH, FINISH
3739	0161	0000	SWITCH, 0
3740	0162	0000	SWTMP, 0
3741	0163	2124	PB1FLG, B1FLG=1
3742	0164	0000	MYAC1, 0
3743	0165	0000	MYAC2, 0
3744	0166	0000	MYAC3, 0
3745	0167	0044	P1FLAC, FLAC
3746	0170	0045	P2FLAC, FLAC*1
3747	0171	0046	P3FLAC, FLAC*2
3748	0172	7764	07764, 7764
3749	0173	6000	06000, 6000
3750	0174	7420	07420, 7420
3751	0177	0177	*177
3752	0177	6203	FERROR, 6203
3753	0200	5601	JMP I ,*1
3754	0201	5774	FSSERR
3755	0202	0202	*202
3756	0200	0200	CHARTAB=,=2
3757	0202	4477	4477,7744
	0203	7744	
3758	0204	5177	5177,2651
	0205	2651	
3759	0206	4136	4136,2241
	0207	2241	
3760	0210	4177	4177,3641
	0211	3641	
3761	0212	4577	4577,4145
	0213	4145	
3762	0214	4477	4477,4044
	0215	4044	
3763	0216	4136	4136,2645
	0217	2645	
3764	0220	1077	1077,7710
	0221	7710	
3765	0222	7741	7741,0041
	0223	0041	
3766	0224	4142	4142,4076
	0225	4076	
3767	0226	1077	1077,4324
	0227	4324	
3768	0230	0177	0177,0301
	0231	0301	

3769	0232	3077	307717730
	0233	7732	
3770	0234	3077	307717706
	0235	7706	
3771	0236	4177	417717741
	0237	7741	
3772	0240	4477	447713044
	0241	3044	
3773	0242	4276	427610376
	0243	1376	
3774	0244	4477	447713146
	0245	3146	
3775	0246	5121	512114651
	0247	4651	
3776	0250	4040	404014077
	0251	4077	
3777	0252	0177	017717701
	0253	7701	
3778	0254	0176	017617402
	0255	7402	
3779	0256	0677	067717701
	0257	7701	
3780	0260	1463	146316314
	0261	6314	
3781	0262	0770	077017007
	0263	7007	
3782	0264	4543	454316151
	0265	6151	
3783	0266	4177	417710000
	0267	0000	
3784	0270	1020	102010204
	0271	0204	
3785	0272	0000	000017741
	0273	7741	
3786	0274	2000	200012076
	0275	2076	
3787	0276	1604	160410404
	0277	0404	
3788	0300	0000	000010000
	0301	1000	
3789	0302	7500	750010000
	0303	0000	
3790	0304	7600	700010070
	0305	0070	
3791	0306	7624	762412476
	0307	2476	
3792	0310	5721	572114671
	0311	4671	
3793	0312	6661	666114333
	0313	4333	
3794	0314	5166	516610526
	0315	1526	
3795	0316	7000	700010000
	0317	0000	
3796	0320	3600	360010041

3797	0321	041	
	0322	4100	410010036
	0323	036	
3798	0324	2050	205010050
	0325	050	
3799	0326	404	040410437
	0327	437	
3800	0330	0500	050010006
	0331	006	
3801	0332	404	040410404
	0333	404	
3802	0334	0001	000110000
	0335	000	
3803	0336	0601	060114030
	0337	4030	
3804	0340	4536	453613651
	0341	3651	
3805	0342	2101	210110177
	0343	177	
3806	0344	4523	452312151
	0345	2151	
3807	0346	4122	412212651
	0347	2651	
3808	0350	2414	241410477
	0351	0477	
3809	0352	5172	517210651
	0353	0651	
3810	0354	1506	150614225
	0355	4225	
3811	0356	4443	444316050
	0357	6050	
3812	0360	5126	512612651
	0361	2651	
3813	0362	5122	512213651
	0363	3651	
3814	0364	2200	220010000
	0365	0000	
3815	0366	4601	460110000
	0367	0000	
3816	0370	1000	100014224
	0371	4224	
3817	0372	1212	121211212
	0373	1212	
3818	0374	2442	244210010
	0375	0010	
3819	0376	4020	402012055
	0377	2055	

3820			
3821	1000		
3822			
3823			
3824			
3825			
3826			
3827	1000	0000	

/4000-777 ARE CHARACTER DISPLAY AREA
 *1000
 /
 /GET RIGHT HAND SIDE - USED IN
 /PROCESSING OF COMMANDS (LIBR) WHICH NEED
 /A FILE NAME; EXPECTS THE FORM FILE, UNIT
 /
 GETRMS; 0

3828	1031	3675	DCĀ I	PLEFLAG
3829	1022	1322	TAD	PLNAME
3830	1003	3011	DCĀ	BLK2
3831	1004	1326	TAD	07772
3832	1005	3012	DCĀ	BLK2+1
3833	1006	1324	PLL1, TAD	077
3834	1027	3411	DCĀ I	BLK2
3835	1010	2012	ISE	BLK2+1
3836	1011	5206	JMP	PLL1
3837	1012	1322	TAD	PLNAME
3838	1013	3011	DCĀ	BLK2
3839	1014	1326	TAD	07770
3840	1015	3012	DCĀ	BLK2+1
3841	1016	4333	PLL2, JMS	CGET
3842	1017	5236	JMP	IGOTIT
3843	1020	5330	JMP	RHSERR
3844	1021	1324	AND	077
3845	1022	1277	TAD	M43
3846	1023	7450	SNA	
3847	1024	5261	JMP	NUMSGN
3848	1025	1300	TAD	PP43
3849	1026	3411	DCĀ I	BLK2
3850	1027	2012	ISE	BLK2+1
3851	1030	5216	JMP	PLL2
3852	1031	4333	JMS	CGET
3853	1032	5236	JMP	IGOTIT
3854	1033	5330	JMP	RHSERR
3855	1034	7200	CLĀ	
3856	1035	5231	JMP	=4
3857	1036	1322	IGOTIT, TAD	PLNAME
3858	1037	3011	DCĀ	BLK2
3859	1040	1327	TAD	07774
3860	1041	3012	DCĀ	BLK2+1
3861	1042	1322	TAD	PLNAME
3862	1043	3013	DCĀ	BLK2+2
3863	1044	1411	PLL3, TAD I	BLK2
3864	1045	7106	CLL	RTL
3865	1046	7006	RTL	
3866	1047	7006	RTL	
3867	1050	1411	TAD I	BLK2
3868	1051	3413	DCĀ I	BLK2+2
3869	1052	2012	ISE	BLK2+1
3870	1053	5244	JMP	PLL3
3871	1054	7326	CLĀ CLL	CML RTL
3872	1055	3376	DCĀ	LNAME+4
3873	1056	4301	MORNUM, JMS	OCTNUM
3874	1057	5000	JMP I	GETRHS
3875	1060	5330	JMP	RHSERR
3876			/	
3877			/SCAN OFF THE NUMBER = SET THE FLAG	
3878			/WHICH SAYS IT WAS A NUMBER	
3879			/	
3880	1061	1012	NUMSGN, TAD	BLK2+1
3881	1062	1323	TAD	010
3882	1063	7650	SNA	CLĀ

3883	1064	4301	JMS	OCTNUM
3884	1065	5330	JMP	RHSERR
3885	1066	1371	TAD	LNUM
3886	1067	3545	DCA I	P5LNAM
3887	1070	1276	TAD	FLAGJ
3888	1071	3675	DCA I	PLEFLAG
3889	1072	7240	CLA CMA	
3890	1073	3546	DCA I	P6LNAM
3891	1074	5256	JMP	MORNUM
3892	1075	1462	PLEFLAG, LEFLAG	
3893	1076	5265	FLAGJ, LEFLAG+3&177+5200	
3894	1077	7735	M43, =43	
3895	1100	0043	PP43, 43	
3896	1101	0000	OCTNUM, 0	
3897			/	
3898			/SUBR TO GEN AN OCTAL NUMBER	
3899			/	
3900	1102	3371	PLLP4, DCA	LNUM
3901	1103	4333	JMS	CGET
3902	1104	2301	ISE	OCTNUM
3903	1105	5701	JMP I	OCTNUM
3904	1106	1324	AND	077
3905	1107	1325	TAD	07710
3906	1110	7100	CLL	
3907	1111	1323	TAD	010
3908	1112	3333	DCA	CGET
3909	1113	7420	SNL	
3910	1114	5330	JMP	RHSERR
3911	1115	1371	TAD	LNUM
3912	1116	7106	CLL RTL	
3913	1117	7104	CLL RAL	
3914	1120	1333	TAD	CGET
3915	1121	5302	JMP	PLLP4
3916	1122	1171	PLNAME, LNAME=1	
3917	1123	0010	010, 10	
3918	1124	0077	077, 77	
3919	1125	7710	07710, 7710	
3920	1126	7770	07770, 7770	
3921	1127	7774	07774, 7774	
3922	1130	6203	RHSERR, 6203	/RIGHT HAND SIDE ERROR
3923	1131	5732	JMP I	,+1
3924	1132	6357	LERR	
3925	1133	0000	CGET, 0	/INTERFACE WITH FIELD ZERO
3926	1134	6203	6203	/ JMS CGET
3927	1135	5736	JMP I	,+1 / JMP <COMMA>
3928	1136	2564	CGETX	/ JMP <CARRET OR SEMICOLON>
3929	1137	1354	CGETRET, TAD	07524 / JMP <OTHER(CHAR IS IN AC)>
3930	1140	7450	SNA	
3931	1141	5733	JMP I	CGET
3932	1142	2333	ISE	CGET
3933	1143	1355	TAD	07761
3934	1144	7450	SNA	
3935	1145	5733	JMP I	CGET
3936	1146	1356	TAD	056
3937	1147	7450	SNA	

3938	1150	5733	JMP I	CGET
3939	1151	1357	TAD	0215
3940	1152	2333	ISE	CGET
3941	1153	5733	JMP I	CGET
3942	1154	7524	07524,	7524
3943	1155	7761	07761,	7761
3944	1156	0056	056,	56
3945	1157	0215	0215,	215
3946			/	
3947			/BRINGS MILDRED INTO CORE	
3948			/	
3949	1160	0000	LDMILD;	0
3950	1161	6002	IOF	
3951	1162	4540	JMS I	X7774
3952	1163	1165	MLOBLK	
3953	1164	5760	JMP I	LDMILD
3954	1165	0110	MLOBLK;	110
3955	1166	0030		30
3956	1167	0076		76
3957	1170	0002		2
3958		1171	*1171	
3959	1171	0000	LNUM,	0
3960	1172	0000	LNAME,	01010101010
	1173	0000		
	1174	0000		
	1175	0000		
	1176	0000		
	1177	0000		
3961	1200	0000	MVCTR;	0
3962	1201	0000	MVPTR;	0
3963	1202	7240	LCHAIN;	CLÄ CMA
3964			/	
3965			/LIBRARY LOAD	
3966			/	
3967	1203	3147	LLOAD;	DCÄ CHFLAG
3968	1204	4543	JMS I	PGETRHS
3969	1205	4544	JMS I	PLOMILD
3970	1206	4342	JMS	LUKUP
3971	1207	1546	TAD I	PÖLNAM
3972	1210	7241	CIÄ	
3973	1211	1327	TAD	LLENGTH
3974	1212	7640	SEÄ	CLA
3975	1213	5356	JMP	FILERR*2
3976	1214	1542	TAD I	PLNUM
3977	1215	3324	DCÄ	LSBLK
3978	1216	1545	TAD I	PÖLNAM
3979	1217	3326	DCÄ	FILSTRY
3980	1220	4540	JMS I	X7774
3981	1221	1324	LSBLK	
3982	1222	7350	CLA	CLL CMA RAR
3983	1223	3010	DCÄ	XR1
3984	1224	1410	TAD I	XR1
3985	1225	1174	TAD	07420 /FIRST WD MUST BE 0360
3986	1226	7640	SEÄ	CLA
3987	1227	5356	JMP	FILERR*2

3988 1230 1304
 3989 1231 4262
 3990 1232 5254
 3991
 3992
 3993
 3994 1233 3147
 3995 1234 4543
 3996 1235 4544
 3997 1236 1327
 3998 1237 3546
 3999 1240 4361
 4000 1241 1542
 4001 1242 3324
 4002 1243 1545
 4003 1244 3326
 4004 1245 7350
 4005 1246 3010
 4006 1247 1007
 4007 1250 3410
 4008 1251 4262
 4009 1252 4541
 4010 1253 1324
 4011 1254 6203
 4012 1255 6001
 4013 1256 2147
 4014 1257 5722
 4015 1260 5661
 4016 1261 6603
 4017
 4018
 4019
 4020
 4021 1262 0000
 4022 1263 3306
 4023 1264 1330
 4024 1265 3200
 4025 1266 1600
 4026 1267 2200
 4027 1270 7450
 4028 1271 5275
 4029 1272 3201
 4030 1273 4305
 4031 1274 5266
 4032 1275 1323
 4033 1276 3200
 4034 1277 2201
 4035 1300 4305
 4036 1301 2200
 4037 1302 5277
 4038 1323 5662
 4039 1304 5314
 4040 1305 0000
 4041 1306 7402

TAD LOADJ
 JMS MOO
 JMP XGETOUT
 /
 /LIBRARY SAVE
 /
 LSAVE, DCA CHFLAG
 JMS I PGETRHS
 JMS I PLDMILD
 TAD LLENGTH
 DCA I P6LNAM
 JMS REPLACE
 TAD I PLNUM
 DCA LSBLK
 TAD I P5LNAM
 DCA FILSTRT
 CLA CLL CMA RAR
 DCA XR1
 TAD 0300
 DCA I XR1
 JMS MOO
 JMS I X7775
 LSBLK
 XGETOUT, 6203
 ION
 ISZ CHFLAG
 JMP I PSTART
 JMP I ,+1
 GOTO

/THE WORDS ARE READ/WRITTEN FROM LOC 4000
 /OF FLD1; THIS ROUTINE MOVES THEM THERE
 /

MOO, 0
 DCA DEJUMP
 TAD PTBL
 DCA MVCTR
 MOOLUP, TAD I MVCTR
 ISZ MVCTR
 SNA
 JMP MOOEND
 DCA MVPTR
 JMS MOVMOV
 JMP MOOLUP
 MOOEND, TAD MVCNT
 DCA MVCTR
 ISZ MVPTR
 JMS MOVMOV
 ISZ MVCTR
 JMP ,+3
 JMP I MOO
 LOADJ, JMP NOTSAV
 MOVMOV, 0
 DEJUMP, HIT

4043 1310 1601
 4044 1311 6211
 4045 1312 3410
 4046 1313 5705
 4047 1314 6211
 4048 1315 1410
 4049 1316 6201
 4050 1317 3601
 4051 1320 6211
 4052 1321 5705
 4053 1322 1177
 4054 1323 6366
 4055 1324 0000
 4056 1325 0030
 4057 1326 0000
 4058 1327 0004
 4059 1330 1331
 4060 1331 0035
 4061 1332 0410
 4062 1333 0411
 4063 1334 0412
 4064 1335 0060
 4065 1336 0031
 4066 1337 0013
 4067 1340 3206
 4068 1341 0000
 4069
 4070
 4071
 4072 1342 0000
 4073 1343 6141
 4074 1344 0606
 4075 1345 1020
 4076 1346 1171
 4077 1347 6020
 4078 1350 7354
 4079 1351 0002
 4080 1352 7200
 4081 1353 5742
 4082 1354 0002
 4083 1355 7200
 4084 1356 6203
 4085 1357 5760
 4086 1360 2571
 4087
 4088
 4089
 4090 1361 0000
 4091 1362 6141
 4092
 4093 1363 0606
 4094 1364 1020
 4095 1365 1171
 4096 1366 6022
 4097 1367 7372

TAD I MVPTR
 6211
 DCA I XR1
 JMP I MOVMOV
 6211
 TAD I XR1
 6201
 DCA I MVPTR
 6211
 JMP I MOVMOV
 PSTART, START
 MVCNT, FRST=FEXP
 LSBLK, 0
 30 /*14000
 FILSTR, 0
 LLENGTH, 4
 PTBL, .+1
 BOTTOM
 PFNEW
 PFX
 PF2
 BUFR
 LASTV
 PDLXR
 FRST
 0

/USES MILDREDS LOOKUP

LUKUP, 0
 6141 /LINC
 0606 /LIF 6
 1020 /LDA I
 LNUM
 6020 /JMP 20
 FILERR&1777+6000
 0002 /POP
 CLA
 JMP I LUKUP
 FILERR, 0002 /POP
 CLA
 6203 /CIF CDF 0
 JMP I .+1
 ERRFIL

/USES MILDREDS REPLACE

REPLACE, 0
 LINC
 LMODE
 LIF 6
 LOA I
 LNUM
 JMP 22
 JMP SAMEN /ALREADY THERE

4098	1372	7354	JMP	FILERR	/NOT ENUF ROOM
4099	1371	7375	JMP	ENREPL	
4100	1372	6826	SAMEN, LIF	6	
4101	1373	6824	JMP	24	
4102	1374	7354	JMP	FILERR	/NOT ENUF ROOM; SHOULD NOT HAPPEN
4103	1375	6802	ENREPL, POP		
4104			PMODE		
4105	1376	7200	CLA		
4106	1377	5761	JMP I	REPLACE	
4107		1400	*1400		
4108	1400	7524	MINCMA, =254		
4109	1401	7066	PCHAR, CHAR		
4110	1402	3157	LMAKE, DCA	MYTMP2	/LIBRARY MAKE
4111	1403	6201		6201	
4112	1404	1601	TAD I	PCHAR	
4113	1405	6211		6211	
4114	1406	1200	TAD	MINCMA	
4115	1407	7640	SZA CLA		
4116	1410	5623	JMP I	PRHSERR	
4117	1411	4543	JMS I	PGETRHS	
4118	1412	4544	JMS I	PLDMILD	
4119	1413	1157	TAD	MYTMP2	
4120	1414	3546	DCA I	P6LNAM	
4121	1415	4555	JMS I	PREPLAC	
4122	1416	6203	LXIT, 6203		
4123	1417	6001	ION		
4124	1420	5621	JMP I	PPROC	
4125	1421	0611	PPROC, PROC		
4126	1422	1133	PGETC, CGET		
4127	1423	1130	PRHSERR, RHSERR		
4128	1424	7510	07510, 7510		
4129	1425	0010	0010, 10		
4130	1426	7455	MCS, =323		
4131	1427	0012	CSMCI, 323-311		
4132	1430	0003	CIMCF, 311-306		
4133			/FILTAB ENTRY	TYPE	
4134			/	LENGTH	
4135			/	UNIT	
4136			/	FIRST BLOCK	
4137			/WHERE TYPE	0	UNDEFINED
4138			/	1	UNSIGNED(1 WD)
4139			/	2	SIGNED(2 WD)
4140			/	3	FLOATING POINT(3 WD)
4141	1431	4302	LOPEN, JMS	COMSUB	/LIBRARY OPEN
4142	1432	4022	JMS I	PGETC	
4143	1433	5236	JMP	,+3	
4144	1434	7000	NOP		
4145	1435	5257	JMP	ERXIT	
4146	1436	4306	JMS	GETCX	
4147	1437	1226	TAD	MCS	
4148	1440	7450	SNA		
4149	1441	5251	JMP	ITSSS	
4150	1442	1227	TAD	CSMCI	
4151	1443	7450	SNA		
4152	1444	7450	SNA		

4153	1445	1230	TAD	CIMCF
4154	1446	7640	SZA	CLA
4155	1447	5623	JMP	I PRHSERR
4156	1450	7001	ITSFF,	IAC
4157	1451	7001	ITSSS,	IAC
4158	1452	7001	ITSII,	IAC
4159	1453	3157	DCA	MYTMP2
4160	1454	4622	JMS	I PGETC
4161	1455	5261	JMP	,+4
4162	1456	7000	NOP	
4163	1457	7200	ERXIT,	CLA
4164	1460	5623	JMP	I PRHSERR
4165	1461	4543	JMS	I PGETRHS
4166	1462	0000	LEFLAG;	0 / (OR JMP ,+3 IF GETRHS GOT A #)
4167	1463	4544	JMS	I PLOMID
4168	1464	4553	JMS	I PLOOKUP
4169	1465	1157	TAD	MYTMP2
4170	1466	3556	DCA	I MYTEMP
4171	1467	2156	ISZ	MYTEMP
4172	1470	1546	TAD	I P6LNAM
4173	1471	3556	DCA	I MYTEMP
4174	1472	2156	ISZ	MYTEMP
4175	1473	1542	TAD	I PLNUM
4176	1474	3556	DCA	I MYTEMP
4177	1475	2156	ISZ	MYTEMP
4178	1476	1545	TAD	I P6LNAM
4179	1477	3556	DCA	I MYTEMP
4180	1500	5216	JMP	LXIT
4181	1501	7472	07472,	7472
4182			/	
4183			/SCANS OFF FN AND LEAVES POINTER IN MYTEMP	
4184			/	
4185	1502	0000	COMSUB;	0
4186	1503	4366	JMS	GETCX
4187	1504	1301	TAD	07472
4188	1505	7650	SNA	CLA /F
4189	1506	4366	JMS	GETCX
4190	1507	1224	TAD	07510
4191	1510	7100	CLL	
4192	1511	1225	TAD	0010
4193	1512	7420	SNL	
4194	1513	5257	JMP	ERXIT
4195	1514	7106	CLL	RTL
4196	1515	1152	TAD	PFILTAB
4197	1516	3156	DCA	MYTEMP
4198	1517	5702	JMP	I COMSUB
4199			/	
4200			/LIBRARY CLOSE	
4201			/	
4202	1520	4302	LCLOSE;	JMS COMSUB
4203	1521	4622	JMS	I PGETC
4204	1522	5623	JMP	I PRHSERR
4205	1523	7410	SKP	
4206	1524	5257	JMP	ERXIT
4207	1525	3556	DCA	I MYTEMP

4208 1526 6002
 4209 1527 4560
 4210 1530 7307
 4211 1531 4560
 4212 1532 5216
 4213
 4214
 4215
 4216 1533 0000
 4217 1534 4554
 4218
 4219
 4220
 4221
 4222
 4223 1535 5346
 4224 1536 5341
 4225 1537 1551
 4226 1540 2151
 4227 1541 3164
 4228 1542 1551
 4229 1543 3165
 4230 1544 2151
 4231 1545 5354
 4232 1546 1370
 4233 1547 3164
 4234 1550 1551
 4235 1551 7710
 4236 1552 7040
 4237 1553 3165
 4238 1554 1551
 4239 1555 3166
 4240 1556 6203
 4241 1557 1164
 4242 1560 3567
 4243 1561 1165
 4244 1562 3570
 4245 1563 1166
 4246 1564 3571
 4247 1565 5733
 4248 1566 0000
 4249 1567 4622
 4250 1570 0027
 4251 1571 5623
 4252 1572 5766
 4253 1600
 4254
 4255
 4256
 4257 1600 0000
 4258 1601 376
 4259 1602 7106
 4260 1603 1152
 4261 1604 1156

IOF
 JMS I PFINISH
 CLA CLL IAC RTL
 JMS I PFINISH
 JMP LXIT
 /
 /FILE VARIABLE LOADER
 /
 ITLOAD, 0
 JMS I PCOMMON
 /
 /VARIABLE IS NOW IN MEMORY! LOSS
 /POINT AT IT! ONE OF THE FOLLOWING 3 CHOICES WILL BE TAKEN, ACCORDING
 /TO TYPE
 /
 JMP IRETLD
 JMP SRETLD
 FRETLD, TAD I LOSS
 ISZ LOSS
 SRETLD, DCA MYAC1
 TAD I LOSS
 DCA MYAC2
 ISZ LOSS
 JMP CRETLD
 IRETLD, TAD 027
 DCA MYAC1
 TAD I LOSS
 SPA CLA
 CMA
 DCA MYAC2
 CRETLD, TAD I LOSS
 DCA MYAC3
 6203
 TAD MYAC1
 DCA I P1FLAC
 TAD MYAC2
 DCA I P2FLAC
 TAD MYAC3
 DCA I P3FLAC
 JMP I ITLOAD
 GETCX, 0
 JMS I PGETC
 027, 27
 JMP I PRHSERR
 JMP I GETCX
 *1600
 /
 /SUBSCRIBING FOR FILE VARIABLES
 /ENTER WITH FILE NO. IN AC
 COMMON, 0
 AND 07
 CLL RTL
 TAD PFILTAR
 DCA

4263	1626	3150	DCA	HISS	
4264	1627	1520	TAD I	PSURS	/SUBSCRIPTS
4265	1610	3151	DCA	LOSS	
4266	1611	6211	6211		
4267	1612	1556	TAD I	MYTEMP	
4268	1613	7650	SNÄ CLA		
4269	1614	5177	JMP	FERROR	
4270	1615	1556	TAD I	MYTEMP	
4271	1616	3011	DCA	BLK2	
4272	1617	1411	TAD I	BLK2	/(REFERENCES LOCS 2,3,4)
4273	1620	3011	DCA	BLK2	
4274	1621	3013	DCA	BLK2+2	
4275	1622	1011	PREDIV; TAD	BLK2	/DIVIDES BY NO. ENTRIES/BLOCK
4276	1623	7141	CLL CIA		
4277	1624	1150	TAD	HISS	
4278	1625	7420	SNL		
4279	1626	5232	JMP	DIVDIV	
4280	1627	3150	DCA	HISS	
4281	1630	2013	ISZ	BLK2+2	
4282	1631	5222	JMP	PREDIV	
4283	1632	7200	DIVDIV; CLA		
4284	1633	1172	TAD	07764	
4285	1634	3012	DCA	BLK2+1	/LOW ORDER SUBSCRIPT, THEN POINTER
4286	1635	1151	DIVLUP; TAD	LOSS	
4287	1636	7104	CLL RAL		
4288	1637	3151	DCA	LOSS	
4289	1640	1150	TAD	HISS	
4290	1641	7004	RAL		
4291	1642	3150	DCA	HISS	
4292	1643	1011	TAD	BLK2	
4293	1644	7141	CLL CIA		
4294	1645	1150	TAD	HISS	
4295	1646	7430	SZL		
4296	1647	3150	DCA	HISS	
4297	1650	7200	CLA		
4298	1651	1013	TAD	BLK2+2	
4299	1652	7004	RAL		
4300	1653	3013	DCA	BLK2+2	
4301	1654	7430	SZL		
4302	1655	5177	JMP	FERROR	
4303	1656	2012	ISZ	BLK2+1	
4304	1657	5235	JMP	DIVLUP	
4305	1660	1556	TAD I	MYTEMP	
4306	1661	2156	ISZ	MYTEMP	
4307	1662	7041	CIA		
4308	1663	3012	DCA	BLK2+1	
4309	1664	7410	SKP		
4310	1665	2200	ISZ	COMMON	/SETS UP COMMON XIT ACCORDING TO FILE TYPE
4311	1666	1150	TAD	HISS	
4312	1667	2012	ISZ	BLK2+1	/TBLK (RELATIVE) IS IN BLK2+2
4313	1670	5265	JMP	=3	
4314	1671	3151	DCA	LOSS	
4315	1672	1013	TAD	BLK2+2	
4316	1673	7140	CLL CMA		
4317	1674	1556	TAD I	MYTEMP	/(THE LENGTH)

4318 1675 762A
 4319 1676 5177
 4320 1677 2156
 4321 1700 1556
 4322 1701 3011
 4323 1702 2156
 4324 1703 1556
 4325 1704 1013
 4326 1705 3013
 4327 1706 4351
 4328 1707 7307
 4329 1710 4351
 4330 1711 1161
 4331 1712 7650
 4332 1713 7307
 4333 1714 3161
 4334 1715 6002
 4335 1716 1161
 4336 1717 4560
 4337 1720 1161
 4338 1721 1163
 4339 1722 3010
 4340 1723 7201
 4341 1724 3410
 4342 1725 1011
 4343 1726 3410
 4344 1727 1410
 4345 1730 3012
 4346 1731 1013
 4347 1732 3410
 4348 1733 4540
 4349 1734 0011
 4350 1735 1161
 4351 1736 7106
 4352 1737 7006
 4353 1740 7006
 4354 1741 1173
 4355 1742 1151
 4356 1743 3151
 4357 1744 7346
 4358 1745 1010
 4359 1746 3150
 4360 1747 6001
 4361 1750 5600
 4362 1751 0000
 4363 1752 3162
 4364 1753 1162
 4365 1754 1163
 4366 1755 3010
 4367 1756 1410
 4368 1757 7650
 4369 1760 5751
 4370 1761 1410
 4371 1762 7041
 4372 1763 1011

SN CLA /SUBSCRIPT IS TOO LONG
 JMP FERROR
 ISZ MYTEMP
 TAD I MYTEMP
 DCA BLK2
 ISZ MYTEMP
 TAD I MYTEMP /STARTING TBLK
 TAD BLK2+2
 DCA BLK2+2 /ABSOLUTE TBLK
 JMS CHECK
 CLA CLL IAC RTL
 JMS CHECK
 TAD SWITCH /ALTERNATE THE BUFFERS
 SNA CLA
 CLA CLL IAC RTL
 DCA SWITCH
 IOF
 TAD SWITCH
 JMS I PFINISH
 TAD SWITCH
 TAD PB1FLG
 DCA XR1
 CLA IAC
 DCA I XR1
 TAD BLK2
 DCA I XR1
 TAD I XR1
 DCA BLK2+1
 TAD BLK2+2
 DCA I XR1
 JMS I X7774 /READ IT IN
 BLK2
 TAD SWITCH /THE VARIABLE IS IN MEMORY
 ITSAGO: CLL RTL
 RTL
 RTL
 TAD 06000
 TAD LOSS
 DCA LOSS
 CLA CLL CMA RTL
 TAD XR1
 DCA HISS
 ION
 JMP I COMMON
 CHECK: 0
 DCA SWTMP
 TAD SWTMP
 TAD PB1FLG
 DCA XR1
 TAD I XR1
 SNA CLA
 JMP I CHECK
 TAD I XR1
 CIA
 TAD BLK2

4373	1764	7640	SZĀ	CLA	
4374	1765	5751	JMP	I	CHECK
4375	1766	2010	ISZ		XR1
4376	1767	1410	TAD	I	XR1
4377	1770	7041	CIA		
4378	1771	1013	TAD		BLK2+2
4379	1772	7640	SZĀ	CLA	
4380	1773	5751	JMP	I	CHECK
4381	1774	1162	TAD		SWTMP
4382	1775	5336	JMP		ITSAGO /BLK IS IN MEMORY ALREADY
4383	1776	2007			
4384		2000	07,	7	
4385			*2000		
4386			/		
4387			/FILE VARIABLE STORER		
4388	2000	4000	/		
4389	2001	3010	ITSTOR;	0	
4390	2002	1567	DCĀ		XR1
4391	2003	3164	TAD	I	P1FLAC
4392	2004	1570	DCĀ		MYAC1
4393	2005	3165	TAD	I	P2FLAC
4394	2006	1571	DCĀ		MYAC2
4395	2007	3166	TAD	I	P3FLAC
4396	2010	1010	DCĀ		MYAC3
4397	2011	4554	TAD		XR1
4398	2012	5266	JMS	I	PCOMMON /BLK IS IN MEMORY! LOSS POINTS AT IT
4399	2013	5224	JMP		URETST
4400	2014	1164	JMP		SRETST
4401	2015	3551	FRETST;	TAD	MYAC1
4402	2016	2151	DCĀ	I	LOSS
4403	2017	1165	ISZ		LOSS
4404	2020	3551	TAD		MYAC2
4405	2021	2151	DCĀ	I	LOSS
4406	2022	1166	ISZ		LOSS
4407	2023	5271	TAD		MYAC3
4408	2024	1164	JMP		INCALL
4409	2025	7450	SRETST;	TAD	MYAC1
4410	2026	5244	SNA		
4411	2027	7700	JMP		STOKOK
4412	2030	5251	SMA	CLA	
4413	2031	7100	JMP		STO0BG /MUST BE LESS THAN MAGN. 1
4414	2032	1165	NORMLE;	CLL	
4415	2033	7510	TAD		MYAC2
4416	2034	7020	SPĀ		
4417	2035	7010	CML		
4418	2036	3165	RAR		
4419	2037	1166	DCĀ		MYAC2
4420	2040	7010	TAD		MYAC3
4421	2041	3166	RAR		
4422	2042	2164	DCĀ		MYAC3
4423	2043	5231	ISZ		MYAC1
4424	2044	1165	JMP		NORMLE
4425	2045	3551	STOKOK;	TAD	MYAC2
4426	2046	2151	DCĀ	I	LOSS
4427	2047	1166	ISZ		LOSS
			TAD		MYAC3

4428	2050	5271	STP	INCALL
4429	2051	1165	STOORG, TAD	MYAC2
4430	2052	7120	CLL CML	
4431	2053	7700	SMA CLA	
4432	2054	7360	CMA CML	
4433	2055	7010	RAR	
4434	2056	3551	DCA I	LOSS
4435	2057	2151	ISZ	LOSS
4436	2060	1165	TAD	MYAC2
4437	2061	7700	SMA CLA	
4438	2062	7344	CLA CLL CMA RAL	
4439	2063	7001	IAC	
4440	2064	3551	UZERST, DCA I	LOSS
4441	2065	5272	JMP	CRETST
4442	2066	6203	URETST, 6203	
4443	2067	5670	JMP I	.+1
4444	2070	7576	CALLIN	
4445	2071	3551	INCALL, DCA I	LOSS
4446	2072	7240	CRETST, CLA CMA	
4447	2073	3550	DCA I	HISS
4448	2074	6203	6203	
4449	2075	5600	JMP I	ITSTOR
4450	2076	7000	FINISH, 0	
4451	2077	1163	TAD	PB1FLG
4452	2100	3010	DCA	XR1
4453	2101	1410	TAD I	XR1
4454	2102	7700	SMA CLA	
4455	2103	5676	JMP I	FINISH
4456	2104	1010	TAD	XR1
4457	2105	3321	DCA	BLOCK
4458	2106	7201	CLA IAC	
4459	2107	3721	DCA I	BLOCK
4460	2110	1410	TAD I	XR1
4461	2111	3321	DCA	BLOCK
4462	2112	1410	TAD I	XR1
4463	2113	3322	DCA	BLOCK+1
4464	2114	1410	TAD I	XR1
4465	2115	3323	DCA	BLOCK+2
4466	2116	4541	JMS I	X7775
4467	2117	2121	BLOCK	
4468	2120	5676	JMP I	FINISH
4469	2121	0000	BLOCK, 0	/UNIT
4470	2122	0000	0	/ADDRESS/256
4471	2123	0000	0	/BLOCKNUM
4472	2124	0001	1	/BLOCKCOUNT

```

/
/BXFLG=0 IF THE BUFFER IS FREE
/
  += IF THE BUFFER IS OCCUPIED
/
  -= IF OCCUPIED AND SOMETHING HAS
/
  CHANGED) IE MUST BE WRITTEN OUT
/BXBLK CONTAINS THE TBLK WHICH IS IN THE BUFFER
/PB1FLG POINTS TO B1FLG; ADDIGNS SWITCH MAKES
/IT POINT AT B2FLG

```

4483 2126 0000
 4484 2127 0034
 4485 2130 0000
 4486 2131 0000
 4487 2132 0000
 4488 2133 0035
 4489 2134 0000
 4490
 4491
 4492
 4493
 4494
 4495
 4496
 4497
 4498 2135 0000
 2136 0000
 2137 0000
 2140 0000
 2141 0000
 2142 0000
 2143 0000
 2144 0000
 4499 2145 0000
 2146 0000
 2147 0000
 2150 0000
 2151 0000
 2152 0000
 2153 0000
 2154 0000
 4500 2155 0000
 2156 0000
 2157 0000
 2160 0000
 2161 0000
 2162 0000
 2163 0000
 2164 0000
 4501 2165 0000
 2166 0000
 2167 0000
 2170 0000
 2171 0000
 2172 0000
 2173 0000
 2174 0000
 4502

B1UNIT; 0
 B1BLK; 34
 B2FLG; 0
 B2UNIT; 0
 B2BLK; 35
 /
 /FILE DEFINITIONS = 4 WORDS A PIECE
 /-TYPE (1,2,3=U,S,F; 0 FOR UNDEFINED)
 /-LENGTH (7777 IF #)
 /-UNIT
 /-FIRST BLOCK
 /

FILTAB; 0101010101010

0101010101010

0101010101010

0101010101010

0000	01111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111001
0200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0300	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111

0400
0500
0600
0700

1000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1300	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1500	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11100000
1600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1700	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111110

2000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
2100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111000

2200
2300
2400
2500
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

A	345	C140	2554	D256	0002	EFUN31	2136
ABSOL	6751	C144	6140	D85	0004	ELPAR	1764
ABSOL2	6153	C220	2123	DATUM	7102	ENO	7134
ABSOL3	7375	C260	0113	DATUMA	7252	ENDFI	6243
ABSOLV	5571	C3	5345	DCONP	6303	ENDLN	4556
AC1H	0041	C5	5341	DCONT	2471	ENDT	2135
AC1L	0042	C7	5335	DCOUNT	6143	ENREPL	1375
ACMINS	6605	C9	5331	DDTJR	0004	ENUM	1732
ADDR	0040	CALLIN	7576	DEBSW	0026	EPAR	1710
ADONE	6673	CCR	0077	DECON	5627	EPAR2	1766
AF	4677	CDF	7000	DECONV	5600	ERASE	2206
ALF1	4760	CEX1	6506	DECP	5533	ERG	2227
ALF2	4763	CEXP	6505	DECR	5521	ERL	2224
ALF3	4755	CF	4705	DEJUMP	1306	ERR2	2726
ALGN	6572	CFRS	0133	DELETE	4565	ERRFIL	2571
ALIGN	6623	CFRSX	0137	DF	4710	ERROR2	4566
ALIST	1370	CGET	1133	DGRP	0425	ERROR3	4566
ALPHA	0316	CGETRE	1137	DGRP1	0441	ERROR4	4566
AMOUNT	6722	CGETX	2564	OIG	5543	ERROR5	2725
ARCALG	4732	CHAR	0066	DIGIT	5713	ERT	2216
ARCRN	5024	CHARTA	0200	DIGITS	0006	ERV	2221
ARGNXT	1723	CHECK	1751	DIV1	5754	ERVX	2241
ARTN	5000	CHFLAG	0147	DIV2	6757	ERXIT	1457
ASHFT	6665	CHIN	2157	DIVDIV	1632	ESCA	2532
ASK	1200	CHRCNT	0006	DIVIDE	7150	ETERM	1647
ATLIST	1570	CHREND	0056	DIVLUP	1635	ETERM1	1627
ATSW	0056	CHRLUP	0033	DMDONE	7063	ETERM2	1655
AXIN	0010	CHRT	6133	DMPSW	0100	ETERMN	1644
AXOUT	0017	CIMCF	1430	DMULT	7004	EVAL	1613
B	0046	CLCU	7427	DMULT4	7036	EX1	0040
B1BLK	2130	CLEAR	7672	DNORM	7335	EXIT	2646
B1FLG	2125	CLF	0076	DNUMBR	5714	EXIT1	5034
B1UNIT	2126	CLKFLG	2661	DO	0420	EXIT2	5301
B2BLK	2134	CNTR	0057	DOO	2113	EXIT3	7363
B2FLG	2131	COL	1253	DOONE	2131	EXITJ	2660
B2UNIT	2132	COMBOT	0226	DOONE	0463	EXP	0044
BACK	5503	COMBUF	0132	DOUBLE	0127	EXTR	2313
REGIN	3601	COMEIN	3140	OPCVPT	6302	F	0043
BET1	4771	COMEOU	3206	OPN	6305	FCONT	1101
BET2	4774	COMGO	1161	OPT	6145	FCOS	5177
BETA	0017	COMLST	0774	OSAVE	5640	FCOUNT	5535
BET3	4766	COMMEN	0614	OTST	5647	FEND3	2267
RF	4702	COMMON	1600	DUBDIV	7261	FERROR	0177
RLK2	0011	COMSUB	1502	DUBLAD	5733	FEXP	4620
RLOCK	2121	CON1	5037	OV3	7267	FEXT	0000
ROTTOM	0335	CRETLD	1554	E	0042	FFF	1522
RUFBEQ	3216	CRETST	2072	ECALL	1601	FG02	6011
RUFFER	7470	CRLF	7505	ECHOLS	1624	FG03	6027
RUPR	0060	CRUDDY	1155	EFOP	0056	FG04	6034
RUFST	5531	CSMCI	1427	EFUN	1743	FG05	6070
C	0047	CSTAR	0225	EFUN2	1755	FIG01	6221
C100	0006	D	0041	EFUN3	2021	FIG04	6261

FI 1354
FI1STR 1326
FI1TAB 2135
FINCR 1765
FINDLN 4555
FINDN 2250
FINFIN 1137
FINISH 2076
FINKP 1133
FINPUT 1131
FINT 4407
FISW 052
FIVHUN 7653
FIX 6724
FIXM 6753
FLAC 2044
FLAD 6510
FLAG1 5162
FLAG2 4725
FLAGJ 1076
FLARG 2032
FLARGP 125
FLOV 7107
FLEX 6517
FLGT 6471
FLIMIT 1075
FLINTP 6200
FLIST1 0577
FLIST2 0574
FLMY 6365
FLOG 5040
FLOP 1674
FLOUT 5556
FLOUTP 6000
FLPT 6467
FLSU 6507
FLTONE 2405
FLTXR 014
FLTXR2 0015
FLTZER 2407
FM12 6142
FNFG 5163
FNOR 7000
FNTABF 0374
FNTABL 2167
FNHM 6311
FOR 1041
FORHUN 7651
FOUTPU 1133
FPAC1 7474
FPNT 6400
FPRNT 5465

FRETLD 1537
FRETST 2014
FRST 3206
FRSTX 3214
FSIN 5204
FSSERR 5774
FXIT 0000
G101 3661
G5772 3662
G5773 3663
G7200 3664
G7773 3665
G7774 3666
G7775 3667
G7776 3670
G7777 3671
GAMMA 0005
GBLOK 3655
GECALL 1463
GEND 2334
GERR 0340
GET1 2330
GET3 2345
GETARG 1401
GETC 4545
GETCX 1566
GETLN 4554
GETRHS 1000
GETSGN 1045
GETVAR 1405
GEXIT 0352
GFND1 1510
GINC 0070
GLIST 1375
GO 5021
GONE 0232
GOODY 0045
GOTO 0603
GRPTST 0744
GS1 1435
GS2 1464
GS3 1444
GS4 1457
GSPERCH 1424
GTEM 0021
GZERR 0362
HINBUF 0037
HISS 0150
HORD 0045
I33 2414
IBAR 0212
IECALL 1037

IF 013
IF1 1035
IF3 1025
IGNOR 0217
IGOTIT 1036
ILIST 0771
IN 5513
INBUF 0034
INCALL 2071
INDEV 0064
INDRCT 6465
INFIX 2401
INLIST 0570
INORM 6307
INPUT 0756
INPUTX 0271
INSUR 0036
INTEGE 0053
INTRPT 2603
IOBUF 3120
IPART 1040
IRETLD 1546
IRETN 0227
ITABLE 6575
ITER1 7470
ITLOAD 1533
ITSAGO 1736
ITSFF 1450
ITSII 1452
ITSOK 7521
ITSSS 1451
ITSTOR 2000
JUMP 6464
K5 5525
KINT 2625
L1 5126
L2 5131
L3 5134
L4 5137
LASTLN 0025
LASTOP 0055
LASTV 0031
LC 5171
LCHAIN 1202
LCLOSE 1520
LCON 0371
LDMILD 1160
LEFLAG 1462
LEPUT 0172
LEPUT 6163
LERR 6357
LESUR2 0170

LESURS 2173
LG 6375
LG2E 4713
LGO 6368
LINENO 0067
LIST3 0077
LIST6 0072
LIST7 0074
LISTGO 1366
LL 5173
LLENGT 1327
LLIST 6366
LLOAD 1203
LM 2572
LMAKE 1422
LNAME 1172
LNUM 1171
LO 5167
LOADIT 6333
LOADJ 1304
LOG2 5157
LOG5 5142
LOG6 5145
LOG7 5150
LOG8 5153
LOOP01 6433
LOPEN 1431
LORD 0046
LOSS 0151
LPRTST 2037
LS 6176
LSAVE 1233
LSBLK 1324
LTAPE 6346
LUKUP 1342
LWETMP 0002
LXIT 1416
M100 0101
M10PT 6147
M11 0121
M12 2413
M137 2357
M140 2556
M144 6137
M2 0111
M20 0175
M240 0114
M260 1534
M272 1544
M4 6141
M40 2356
M43 1077

MS	120
M77	103
MBREAK	2642
MCOM	1136
MCR	116
MCS	1426
MD	5526
MEQ	1135
MF	7632
MFIT	117
MHONDR	5375
MIF	7260
MINCHA	1420
MINCOM	6374
MINE	5662
MINSKI	0051
MINUS2	7153
MINUSA	1112
MINUSE	6301
MINUSE2	5663
MLDRK	1165
MLIMIT	7647
MMCOM	7656
MOO	5214
MODIFY	1254
MOO	1262
MOOEND	1275
MOOLUP	1266
MORNUM	1056
MOVMOV	1305
MP1	7254
MP2	7256
MP3	7255
MP4	7200
MP5	7253
MP6	7210
MPPR	1115
MPIUS	5664
MSPACE	5665
MUIDIV	7101
MUIT	6570
MULT10	5667
MUIT2	5715
MUITV	4752
MVCNT	1323
MVCTR	1200
MVPTR	1201
MYAC1	1164
MYAC2	1165
MYAC3	1166
MYTFMP	1156
MYTFMP2	1157

MAGSW	2065
MCHARS	7566
MCOLS	7564
MEGP	4724
MFEEDS	7565
MNLINES	7561
NOASCI	0061
NOCLK	2653
NOCRLF	7510
NOHANG	7556
NORF	6515
NORM	6571
NORMF	7147
NORMLE	2031
NOTSAV	1314
NOX	6675
NOX1	6711
NOX2	6704
NUMSGN	1061
O1	3600
O10	1123
O12	1545
O200	0003
O215	1157
O27	1570
O360	0007
O37	1360
O4377	0076
O4600	5374
O56	1156
O6000	0173
O6377	7570
O7	1776
O7000	7415
O7400	7650
O7420	0174
O7472	1501
O7510	1424
O7524	1154
O7566	7572
O760	0015
O7655	7571
O77	1124
O7710	1125
O7716	7573
O7761	1155
O7763	7567
O7764	0172
O7770	1126
O7774	1127
OC	7752
OCTNUM	1101

OD	7761
ODISSP	7704
OE	7753
OERROR	7713
OEXIT	7731
OGO	7714
OI	7734
OLIST	7722
OM12	5330
ONE	4716
OO10	1425
OO6377	7730
OP	3115
OPMINS	6567
OPNEXT	1622
OPTABL	1731
OPTR	6002
OPTRY	2663
OPTRI	2665
OPTRO	2664
OPUT	5532
OS	7763
OSAMP	1357
OT	7771
OUT	2465
OUTA	5536
OUTCR	2476
OUTDEV	0063
OUTDG	6154
OUTPUT	7706
OUTX	2475
OVER1	0043
OVER2	0047
P	0000
P13	0005
P17	0107
P177	0106
P1FLAC	0167
P2000	0373
P27	6750
P277	0110
P2FLAC	0170
P3	2036
P337	0075
P377	2553
P3FLAC	0171
P40	2552
P4000	0124
P43	6310
P5LNAM	0145
P6LNAM	0146
P7200	1402

P7600	0104
P77	0122
P7700	0121
P7740	0372
PA1	2524
PACBUF	2502
PACKC	4546
PACKST	2027
PACX	2530
PALG	5260
PARTES	2051
PASS	6335
PB1FLG	0163
PC	0022
PC1	0614
PCHAR	1401
PCHECK	5244
PCHK	0510
PCK1	2535
PCLEAR	0175
PCLKFL	7745
PCOMMO	0154
PD2	0534
PD3	0554
PULXR	0013
PECALL	6334
PEO	6135
PER	0102
PFILTA	0152
PFINIS	0160
PFNEW	0410
PFNUM	1771
PFX	0411
PFZ	0412
PGETC	1422
PGETRH	0143
PI	5311
PI2	5036
PIOT	5315
PLCE	5536
PLOMIL	0144
PLEFLA	1075
PLESUB	0101
PLLP1	1006
PLLP2	1016
PLLP3	1044
PLLP4	1102
PLNAME	1122
PLNUM	0142
PLOOKU	0153
PNCHAR	7732
PNCOLS	7776

7777
1413
4544
5541
7733
1100
7705
1421
6144
7574
1622
1155
1423
4551
2442
3114
6132
4553
611
610
7775
7746
1165
1322
1100
1030
1330
1126
6275
1462
4542
4543
4540
1174
7774
1061
5441
6160
1142
6573
6574
1152
4552
2740
2761
5712
1361
6146
6752
7376
6304
7173

RET 5452
RETRN 1563
RETURN 5536
REVIT 7146
RHSERR 1132
RITEOU 3651
RND2 5527
ROOTGO 7461
ROT 2557
ROUND 6151
RTL6 4557
RUB1 3004
RUB2 3042
RUB3 3030
RUB4 3037
RUB5 3041
RUBIT 2555
SADR 6150
SAMEN 1372
SAVAC 2600
SAVE 3751
SAVLK 2601
SBAR 1300
SCHAR 1271
SCONT 1266
SCOPOU 7500
SCOUNT 5534
SETCLK 5351
SETT 1041
SEX 1336
SEXC 0740
SFOUND 1304
SGOT 1310
SIGN 7124
SIGNF 0050
SIN 2662
SMIN 6136
SMP 6101
SMSP 6134
SORTB 1312
SORTC 4550
SORTCN 0054
SORTJ 4547
SPECIA 6777
SPLAT 3051
SPNOR 4560
SPTR 7671
SQCON1 7467
SQEND 7465
SRETLO 1541
SRETN 0261
SRETST 2024

SRNLST 61
START 0177
STARTL 5064
STARTV 0060
STEMP 7750
STEMP2 7751
STOKOK 2044
STOORG 2051
STORIT 6175
SUBR 0102
SUBS 0171
SUBS2 0167
SWITCH 0161
SWTMP 0162
T 0000
T1 0032
T12 3611
T2 0071
T3 0033
TABLE 6466
TAG1 6723
TASK 1202
TASK4 1250
TCRLF 1246
TCRLF2 1243
TOUMP 3052
TELSW 0016
TEM 5156
TEMP 4726
TEN 6271
TENPT 6152
TERMS 1772
TEST2 6736
TEST4 7366
TESTA 0322
TESTC 4564
TESTN 4561
TEXTP 0017
TGO 5400
THIR 7257
THISLN 0023
THISOP 0024
TINTR 1236
TLIST 1376
TLIST2 1532
TLIST3 2377
TQUOT 1227
TRAD 6575
TSTGRP 4563
TSTLPR 4562
TWO 4721
TWOPI 5305

TYPE 1271
TYPE2 1223
URETST 2066
UTE 2276
UTC 2305
UTRA 2274
UTX 2316
UZERST 2064
VAL 0032
WAIT 7657
WAITER 0020
WAITLP 0115
WALL 0664
WEXIT 0072
WORDS 0003
WRITE 0635
WTEST2 0653
WTESTG 0667
WX 0673
X 5321
X1 5035
X2 4675
X7774 0140
X7775 0141
XABS 2016
XADC 1341
XCT 0020
XCTIN 0062
XDELET 2064
XDISP 7602
XENDLN 2360
XFINO 2244
XGETLN 0302
XGETOU 1254
XIS3 2666
XIN 6306
XINPUT 5666
XINT 1156
XLC 0130
XLG 0136
XLL 0132
XLO 0126
XLS 0134
XOUTL 2676
XPOPJ 1565
XPRNT 2425
XPUSHA 0477
XPUSHJ 0521
XQ 0001
XR1 0010
XRAN 1145
XRAR2 7365

YR4	311
YR2	312
YR5	413
YSG1	312
XSORTC	721
XSPADR	1535
XSR2	4676
XSR4	5325
XSORT	7432
XT3	717
XTFSTC	700
YFSTN	1546
YF	2451
Y	377
ZERO	6522

ERRORS DETECTED: 3

LINKS GENERATED: 3

RUNTIME: 40 SECONDS

4K CORE USED

A	12	3292	3191#														
ABSOL	13	3233#															
ABSOL2	2532	2632#															
ABSOL3	7313	3344#															
ABSOLV	2385#	2391	2632	3033	3344												
AC1H	58#	2456	2465	2497	2525	2579	2951	3001	3131	3158	3167	3184	3197	3201			
	3281																
AC1L	59#	2454	2464	2493	2510	2512	2953	3003	3185	3192	3196	3276					
ACMINS	67	2931#	2944	2963	2964	2972	2996										
ADDR	2879	2812	2813	2818	2821	2824	2848#	2856									
ADONE	2962	2978	2985#														
AF	1892	1917#															
ALF1	1956	1968#															
ALF2	1954	1971#															
ALFZ	1958	1965#															
ALGN	2868	2919#															
ALIGN	2919	2945#	2957	2985	2986	2992	2998	3007	3021								
ALIST	759	760	885#														
ALPHA	3631#	3662	3664	3665	3691	3710											
AMOUNT	2967	2968	2976	2983	3008#												
ARCALG	1945#	2004															
ARCRTN	1963	2005#															
ARGNXT	1268	1120#															
ARTN	348	1985#															
ASHFT	2974	2979#															
ASK	746	756#	774														
ATLIST	760	1024#															
ATSW	77#	757	761														
AXIN	29#	215	247	250	810	813	814	828	1270	1274	1276	1323	1430	1431			
	1538	1543	1683	1686	1717	1724	1737	1738									
AXOUT	37#	231	1232	1367	1411	1763											
B	3080	3095	3182#														
R1BLK	4485#																
R1FLG	3741	4482#															
R1UNIT	4483#																
R2BLK	4489#																
R2FLG	4486#																
R2UNIT	4487#																
BACK	2329#	2339	2342														
REGIN	32	202	1811#														
RET1	1950	1977#															
RET2	1949	1980#															
BETA	3632#	3673	3677	3700	3702	3706	3708										
RETZ	1952	1974#															
RF	1894	1920#															
RLK2	3626#	3830	3832	3834	3835	3838	3840	3849	3850	3858	3860	3862	3863	3867			
	3868	3869	3880	4271	4272	4273	4274	4275	4281	4285	4292	4298	4300	4303			
	4308	4312	4315	4322	4325	4326	4342	4345	4346	4349	4372	4378					
	4457	4459	4461	4463	4465	4467	4469#										
PLOCK																	
POTTOM	51#	234	4060														
PUFBEG	47	80	134	1808#													
HUFFER	2629	3475#	3413														
PUPFR	80#	246	809	826	1268	1269	1316	1322	1425	1426	1433	4064					

FFOP	76#	921	1043	1136	1142	2740												
FFUN	1252	1124	1136#	1145														
FFUN2	1139	1146#																
FFUN3	135	1185#																
FFUN3I	71	135#	733	738	873	1157	2756	3523										
FLPAR	1072	1122	1153#															
END	133#																	
FNDFI	1444	2682	2690#	2711														
FNULN	172#	260																
FNOT	134#	1315																
FNREPL	4299	4103#																
FNUM	1051	1123	1127#															
FPAR	1086	1109#																
FPAR2	1111	1155#																
ERASE	749	1307#																
ERG	1324#	1333																
ERL	1309	1321#																
ERR2	191	1666#	1668															
ERRFIL	1566#	4086																
ERROR2	188#	289	316	325	379	399	485	508	808	1546	1611	2431	2898	3133				
	3354																	
ERROR3	189#	245	427	955	1314	2020	3497	3508	3562									
ERROR4	190#	307	661	674	682	901	1059	1075	1147	1154	1218	1566	2022	2519				
	2747	2777																
ERROR5	354	355	356	477	1665#													
FRT	1315#	1850																
ERV	1318#	1330																
ERVX	1308	1334#																
ERXIT	4145	4163#	4194	4206														
ESCA	1524	1531#																
ETERM	1065	1076#																
ETERM1	1050	1060#																
ETERM2	1082#	1108																
ETERMN	1056	1073#																
EVAL	665	678	686	776	1048#	1560	2232	3499										
EX1	57#	2516	2848	2866	2887	2909	2958	2980	2981	2987	2999	3020	3064	3134				
EXIT	1598	1632	1614#															
EXIT1	2006	2013#	2071															
EXIT2	2013	2184#																
EXIT3	3321	3334#																
EXITJ	1627#																	
EXP	62#	2397	2534	2556	2560	2572	2582	2602	2609	2673	2865	2879	2960	2990				
	2994	3000	3014	3035	3052	3154	3155	3328	3329	3334	3355	3378	3379	3380				
	3402																	
FXTR	1385#	1407																
F	3090	3106	3186#															
FCONT	687#	714	719															
FCOS	352	2117#																
FCOUNT	2272	2278	2318	2355#	2359													
FEND3	1357	1365#																
FERROR	3752#	4269	4302	4319														
FEXP	51	349	1869#	4054														
FEXT	12#	1876	1902	1912	1961	1992	2000	2011	2026	2036	2070	2121	2135	2145				

