```
                        IFNDEF EDDT,<
                        EDDT=1                  ;MAKE EXEC DDT
                >
                ;ALL OTHER REFERENCES TO VERSION # ARE TAKEN CARE OF BY THIS DEFINITION

000022          DDTVER==22

                SUBTTL  2 JUN 69
                REPEAT 0,<


                DDT ASSEMBLY INSTRUCTIONS

                THE SOURCE FILE OF DDT,13 WILL ASSEMBLE INTO SEVERAL DIFFERENT
                VERSIONS; THE ASSEMBLY IS CONTROLLED BY THE VALUE ASSIGNED
                TO THE SYMBOL "EDDT". THE SYMBOL "EDDT" IS DECODED AS FOLLOWS:

                BIT 35  =0;     ASSEMBLE A USER MODE DDT
                        =1;     ASSEMBLE AN EXECUTIVE MODE DDT

                BIT 34  =0;     DO NOT ASSEMBLE THE PAPER TAPE FEATURES INTO DDT
                        =1;     ASSEMBLE THE PAPER TAPE FEATURES BUT ONLY IF
                                        ASSEMBLING AN EXECUTIVE MODE DDT

                BIT 33  =0;     FOR USER MODE DDT ONLY- ASSEMBLE USING THE
                                        "TTCALL" UUO FOR TELETYPE IO
                        =1;     FOR USER MODE DDT ONLY- ASSEMBLE USING THE
                                        "DDTIN" AND "DDTOUT" UUO'S FOR TELETYPE IO

                BIT 32  =1;     ASSEMBLE A FILE DDT

                BITS (0-17)
                        =0;     ASSEMBLE A RELOCATABLE VERSION OF DDT (RELOC 0)
                     NOT=0;     ASSEMBLE AN ABSOLUTE (NON-RELOCATABLE) VERSION
                                        OF DDT WITH A STARTING ADDRESS BEING THE
                                        NUMBER IN BITS 0-17

                (IF THE SYMBOL "EDDT" IS NOT DEFINED AT ALL, DDT WILL BE ASSEMBLED
                        WITH EDDT=0.)

                EXAMPLES OF "EDDT" DEFINITIONS:
                        EDDT=0  ASSEMBLE A RELOCATABLE USER MODE DDT WITH TELETYPE
                                I/O DONE BY THE "TTCALL" UUO

                        EDDT=1  ASSEMBLE A RELOCATABLE EXECUTIVE MODE DDT

                        EDDT=<XWD 4000,3>
                                ASSEMBLE AN ABSOLUTE EXECUTIVE MODE DDT
                                WITH PAPER TAPE FEATURES, AND WHOSE STARTING
                                ADDRESS IS LOCATION 4000.
```

```
54
55
56
57
58
59
60
61
62                          EXAMPLE OF A MACRO ASSEMBLY COMMAND:
63
64                          .^C
65
66                          .R MACRO
67
68                          *DSK:UDDT,/C=TTY:,DTA2:DDT.12
69                          EDDT=0
70                          ^Z
71
72                          END OF PASS 1                       ;THIS LINE TYPED BY MACRO
73                          ^Z
74
75                          NO ERRORS DETECTED                  ;TYPED BY MACRO
76
77                          PROGRAM BREAK IS 003340             ;TYPED BY MACRO
78
79                          *                                   ;TYPED BY MACRO
80
81
82                          OTHER VERSIONS OF DDT ARE ASSEMBLED IN A SIMILAR MANNER
83                          BY DEFINING "EDDT" A DIFFERENT WAY WHERE THE EXAMPLE
84                          DEFINED "EDDT=0".
85
86                          >       ;END OF REPEAT 0
```

```
                        IFNDEF EDDT,<EDDT==0>

                        DEFINE XP (X,,Y.),<
                                IF2,<X,=Y.
                                INTERN X,>>


                        IFN EDDT&1,<DEFINE HEADER (VERSION),<
                                TITLE EDDT      V0'VERSION      -EXEC MODE DDT
                                JOBREL==37
                                JOBSYM==36
                                ZLOW==40
                        >>

                        IFE EDDT&1,<DEFINE HEADER (VERSION),<
                        IFE EDDT&10,<
                                TITLE UDDT      V0'VERSION      -USER MODE DDT>
                        IFN EDDT&10,<
                                TITLE FILDDT    V0'VERSION      -FILE DDT>
                                EXTERN JOBREL,JOBSYM,JOBSA,JOBHRL
                                ZLOW==140>>

                        ;DO NOT SET LOWER CORE IF EXEC DDT(OK USER OR FILDDT)
                        IFE EDDT&1,<
                        JOBVER==137
                                LOC JOBVER
                                DDTVER                  ;PUT VERSION # IN JORVER
                        JOBDDT==74
                                LOC JOBDDT
                                XWD DDTEND,DDT
                        RELOC 0
                        >

                        IFN EDDT&<XWD -1,0>,<LOC <EDDT>B53>

                        HEADER \DDTVER*
                                        ;THE HEADER MACRO CONSTRUCTS THE TITLE AND VERSION #
```

```
                        IFN EDDT&10,<   CM=2             ;DEFINE SOFTWARE CHANS.
                                        DP=3
                        >
                        ;DEFINE ACCUMULATORS

                000000  F=0             ;FLAGS
                000001  P=1             ;PUSH DOWN
                000002  R=<A=2>         ;POINTERS TO TABLES, CORE, ETC.
                000003  S=<B=3>
                000004  W=<C=4>         ;CONTAINS DISPATCH ADDRESS IN WORD ASSEMBLER
                000005  T=5             ;TRANSFER DATA
                000006  W1=6
                000007  W2=7
                000010  SCH=10          ;MODE CONTROL SWITCH FOR OUTPUT
                000011  AR=11           ;MODE CONTROL SWITCH FOR OUTPUT
                000012  ODF=12          ;MODE CONTROL SWITCH FOR OUTPUT - CURRENT RADIX
                000013  TT=13           ;TEMPORARY
                000014  TT1=14  ;TEMPORARY

                        ;DEFINE I/O DEVICE MNEMONICS FOR DDT USE
                000004  PRS==4
                000120  TTYY==120
                000104  PTRR==104
                000100  PTPP==100

                        ;DEFINE PUSH DOWN LENGTH
                000050  LPDL=50          ;MAX LENGTH PUSH DOWN LIST

                        ;DEFINE BITS FOR USE IN LEFT HALF OF ACCUMULATOR F
                200000  COMF==200000            ;COMMA TYPED FLAG
                100000  TIF==100000            ;TRUNCATE TO 18 BITS -  SET BY SPACE OR COMMA
                000100  PTF==100               ; +, -, OR * HAS BEEN TYPED
                000400  CTF==400
                000004  SF==4           ;SYLLABLE FLAG
                000001  QF==1           ;QUANTITY TYPED IN TO WORD ASSEMBLER

                000040  CF==40          ; $  TYPED
                010000  CCF==10000      ; $$  TYPED
                000002  MF==2           ;MINUS SIGN TYPED IN
                000020  LTF==20         ;LETTER TYPED IN TO CURRENT SYLLABLE
                000010  ROF==10         ;REGISTER OPEN FLAG
                004000  STF==4000
                001000  FAF==1000              ; < TYPED
                002000  SAF==2000              ; > TYPED

                020000  FPF==20000             ; . TYPED IN
                400000  FEF==400000            ; E FLAG

                000200  MLF==200               ;*FLAG
                040000  DVF==40000             ;DIVIDE FLAG

                        ;PID IS 20 IF SYM TAB POINTER IS INDIRECT JOBSYM
                000000  PID==0          ;=0 IF SYMBOL TABLE POINTER IS IN JOBSYM
```

```
178
179                             ;DEFINE BITS FOR USE IN RIGHT HALF OF ACCUMULATOR F
180
181                     000002  ITF==2  ;INSTRUCTION TYPED IF ITF=1
182                     000004  OUTF==4 ;OUTPUT IF OUTF=1
183                     000400  CF1==400                ;OUTPUT 1 REGISTER AS CONSTANT
184                     002000  LF1==2000               ;OUTPUT 1 REGISTER AS FORCED SYMBOLIC OR CONSTANT
185                     000001  Q2F==1          ;NUMBER TYPED AFTER ALT MODE
186                     000010  R20F==10        ;TEMP FLAG USED IN SETUP
187                     000020  SBF==20
188                     000200  NAF==200                ;NEGATIVE ADDRESSES PERMISSABLE
189                     004000  POWF==4000              ;ARGUMENT FOR EXPONENT COMING
190
191                             ;DEFINE SYMBOL TABLE SYMBOL TYPES
192                     040000  GLOBAL==040000          ;GLOBAL SYMBOL
193                     100000  LOCAL==100000
194                     740000  PNAME==740000           ;PROGRAM NAME
195                     200000  DELI==200000            ;DELETE INPUT
196                     400000  DELO==400000            ;DELETE OUTPUT
197
198  000000                     DDT:    INTERN DDTEND   ;DECLARE END OF DDT AS INTERNAL, FOR
199                                                     ; USER TO SEE (USER MODE) AND ONCE ONLY CODE
200                                                     ; (MONITOR)
201                                     IFE EDDT&1,<    ENTRY DDT
202                                             >
203                                     IFN EDDT&1,<    INTERNAL DDT
204                     000000'                         DDTX=DDT
205                                                     ENTRY DDTX              ;NEEDED BY MONITOR>
206
207
208                             ;DEFINE $ SYMBOLS INTERNAL TO DDT
209                             OPDEF DDTINT [Z 0,]                     ;ADDRESS FLAG FOR INTERNAL REGISTERS
210
211                             RADIX 10
212                     000010  NBP==8  ;NUMBER OF BREAKPOINTS
213                             DEFINE DBPNT (Z.)<XP $'Z,'B,<DDTINT B1ADR+3*Z,-3>>
214                     000000  ZZ==0
215                             REPEAT NBP,<DBPNT \<ZZ==ZZ+1>>
216                             DBPNT \<ZZ==ZZ+1>↑XP $1B,<DDTINT B1ADR+3*1-3>↑
217                             DBPNT \<ZZ==ZZ+1>↑XP $2B,<DDTINT B1ADR+3*2-3>↑
218                             DBPNT \<ZZ==ZZ+1>↑XP $3B,<DDTINT B1ADR+3*3-3>↑
219                             DBPNT \<ZZ==ZZ+1>↑XP $4B,<DDTINT B1ADR+3*4-3>↑
220                             DBPNT \<ZZ==ZZ+1>↑XP $5B,<DDTINT B1ADR+3*5-3>↑
221                             DBPNT \<ZZ==ZZ+1>↑XP $6B,<DDTINT B1ADR+3*6-3>↑
222                             DBPNT \<ZZ==ZZ+1>↑XP $7B,<DDTINT B1ADR+3*7-3>↑
223                             DBPNT \<ZZ==ZZ+1>↑XP $8B,<DDTINT B1ADR+3*8-3>↑
224                             RADIX 8
225
226
227                             XP $M,<DDTINT MSK>↑
228                             XP $I,<DDTINT SAVPI>↑
229
230                     000036  SETUWP=36
```

```
231                             ;DEFINE I/O DEVICE MNEMONICS
232
233                             IFN EDDT&1,<
234
235                             XP PI,4B11↑
236                             XP PTP,100B11↑
237                             XP PTR,104B11↑
238                             XP CDR,114B11↑
239                             XP TTY,120B11↑
240                             XP LPT,124B11↑
241                             XP DC,200B11↑
242                             XP DIS,130B11↑
243                             XP PLI,140B11↑
244                             XP CR,150B11↑
245                             XP DSK,170B11↑
246                             XP UTC,210B11↑
247                             XP UTS,214B11↑
248                             XP DCSA,300B11↑
249                             XP DCSB,304B11↑

250                             XP DF,270B11↑
251                             XP MTC,220B11↑
252                             XP MTS,224B11↑
253                             XP MTM,230B11↑
254                             XP DLS,240B11↑
255                             XP MDF,260B11↑
256                             XP DTC,320B11↑
257                             XP DTS,324B11↑
258                             XP TMC,340B11↑
259                             XP TMS,344B11   >↑
260
261                             ;DEFINE EXTENDED OPERATIONS
262
263                             XP JOV,2554B11↑
264                             XP JEN,2545B11↑
265                             XP HALT,2542B11↑
266
267
```

```
                        IFE EDDT&12,<
000000  264000  002360' DDT:    JSR SAVE
000001  260040  001353'        PUSHJ P,REMOVB
000002  554240  002261'        HLRZ T,ESTU             ;THIS SEQUENCE INITS SYM TABLE LOGIC
000003  274240  002261'        SUB T,ESTU
000004  200220  002274'        MOVE W,@SYMP
000005  270240  000004         ADD T,W                 ;IF THE TOP OF THE UNDEFINED SYM TAB DOES
000006  602240  777777         TRNE T,-1               ; NOT POINT TO BOTTOM OF REGULAR SYM TAB,THEN
000007  552220  002261'        HRRZM W,ESTU            ; RE-INIT UNDEFINED SYM TABLE POINTER, ESTU.
000010  200240  002260'        MOVE T,PRGM
000011  274240  000004         SUB T,W                 ;IF THE SYM TABLE PNTR AND THE PROGRAM
000012  651240  000005         TSC T,T                 ; NAME (PRGM) PNTR DO NOT END UP IN THE
000013  200320  002260'        MOVE W1,PRGM            ; SAME PLACE, OR THEY DO NOT BOTH START ON
000014  430320  000004         XOR W1,W                ; AN EVEN (OR BOTH ON ODD) LOCATION, OR
000015  602320  000001         TRNE W1,1               ; PRGM .GE. 0, THEN RE-INIT PRGM.
000016  254000  000021'        JRST DDT0               ;EVEN-ODD PROBLEM
000017  336000  000005         SKIPN T                 ;POINTERS DON'T END TOGETHER
000020  331000  002260'        SKIPL PRGM              ;IF PRGM .GE. 0, THEN RE-INIT
000021  202220  002260' DDT0:   MOVEM W,PRGM           ;PRGM←C(JOBSYM)
                        >
```

```
                        IFN EDDT&10,<
                        DDT:    CALLI
                                MOVEI T,SYMSET
                                MOVEM T,JOBREN          ;SET REENTER ADDRESS
                                MOVEI P,PS
                                INIT 1,17
                                SIXBIT .DSK.
                                %
                                HALT .-3
                                SETZM CRASH+3           ;CLEAR PPN
                                SETZM COMNDS+3          ;CLEARN PPN IN CASE FILDDT SAVED AFTER STARTING
                                SETZM SNAP+3            ;ALSO GOOD PRACTICE TO BE SELF INITIALIZING
                                SETZM CRASHS            ;ASSUME NO CRASH.SAV(IE, PEEK AT CURRENT MONITOR)
                                LOOKUP 1,CRASH
                                JRST NOCSH              ;LET USER PASS, HE MAY WANT TO PEEK AT MON,
                                                        ; OR JUST WANT TO TYPE OUT VALUE OF SOME SYMBOLS
                                SETOM CRASHS            ;FLAG CRASH,SAV EXISTS ON DSK
                                USETI 1,1
                                INPUT 1,RSILST

                                STATZ 1,740000
                                HALT .-3
                                SETOM RSAVE             ;FORCE READING OF "CURRENT" BLOCK
                                                        ; ON FIRST EXAMINE
                        NOCSH:  INIT CM,0
                                SIXBIT .SYS.
                                XWD 0,CBUF
                                HALT .-3
                                SETOM COMAND            ;ASSUME A COMMAND FILE
                                LOOKUP CM,COMNDS
                        NOLPT:  SETZM COMAND            ;NO FILE IF NOT FOUND
                                SKIPN COMAND
                                JRST DD1                ;USE TTY I/O
                                INIT DP,1
                                SIXBIT .LPT.            ;PREPARE FOR LPT OUTPUT IN AL MODE
                                XWD LBUF,0
                                JRST NOLPT              ;TREAT AS THO NO FILE
                                ENTER DP,SNAP
                                JRST NOLPT              ;TREAT AS NO FILE IF CAN'T ENTER
                                JRST DD1

                        REPEAT 0,<
                        HOW TO COPY MONITOR SYMBOL TABLE INTO FILDDT
                        1) USE FDDD10 TO LOAD AN UNRUN VERSION OF YOUR MONITOR
                           AS SAVED AFTER BUILDING OR LOADING,
                        2) RENAME IT AS CRASH,SAV
                        3) LOAD A PRISTINE VERSION OF FILDDT AND START IT TO SET
                           REENTER ADDRESS,
                        4) TYPE ^C   REENTER
                        5) FILDDT WILL EXPAND AS NECESSARY AND COPY THE SYMBOL
                           TABLE FROM CRASH,SAV INTO ITSELF
                        6) AFTER CARRIAGE RETURN IS TYPED FILDDT IS DONE.
                        7) SAVE FILDDT WITH A NEW NAME SO AS NOT TO CONFUSE IT WITH THE
                           ORIGINAL FILDDT.SAV
```

```
342
343                     THE MONITOR CAN BE LOADED IN ANY OF THREE WAYS(IN ORDER OF PREFERENCE):
344                        1. UNDER TIME SHARING WITH REGULAR LOADER AND COMMON
345                        2. UNDER REGULAR 10/30 MONITOR WITH REGULAR 10/30 LOADER & COMMON
346                        3. UNDER SPECIAL 10/30 MONITOR(SPMON) WITH BUILD
347
348                     THE 3 WAYS LEAVE DDTSYM(36),JOBSYM(116) & T30SYM(131) IN DIFFERENT STATES:
349
350                           DDTSYM          JOBSYM          T30SYM
351
352                        1. JUNK            S,T,PTR         JUNK
353                        2. JUNK            JUNK(NON-NEG)   S,T,PTR
354                        3. S,T,PTR         S,T,PTR         JUNK
355                     >
356
357                     DDTSYM=36
358
359                     T30SYM=131
360
361                     EXTERNAL JOBFF,JOBREN
362
363                     SYMSET: MOVEI R,JOBSYM
364                             PUSHJ P,FETCH
365                             JRST ERR
366                             MOVEM T,JOBSYM          ;POSSIBLE SYMBOL POINTER
367                             MOVEI R,DDTSYM
368                             PUSHJ P,FETCH
369                             JRST ERR
370                             MOVEM T,DDTSTP          ;YET ANOTHER CANDIDATE
371                             MOVEI R,T30SYM          ;ASSUME LOADED BY 10/30
372                             PUSHJ P,FETCH
373                             JRST ERR
374                             SKIPL JOBSYM            ;OR IS JOBSYM A POINTER (-VE)?
375                             JRST REGT30             ;LOADED BY 10/30
376                             SKIPL DDTSTP            ;DDTSYM A VALID POINTER (-VE)?
377                             MOVE T,JOBSYM           ;NO TAKE JOBSYM AS THE POINTER
378                     REGT30: MOVEM T,JOBSYM          ;SAVE IT OVER OLD JOBSYM
379                             HLRES T,T
380                             MOVMS T,T               ;LENGTH OF SYMBOL TABLE
381                             MOVE W,JOBFF
382                             ADDI W,1300             ;LEAVE SPACE FOR COMMAND & LPT BUFFERS
383                                                     ; OR DSK BUFFERS AND EXTRA SYMBOL DEFNS.
384                             HRRZ W1,W               ;SAVE LOC FOR COPY
385                             ADD W,T                 ;ADD TABLE LENGTH
386                             IORI W,1777             ;REQUEST INTEGRAL # OF K.
387                             CALLI W,11              ;GET CORE
388                             HALT                    ;UGH!
389                             MOVE R,JOBSYM           ;WHEREABOUTS OF MONITOR SYMBOLS
390                             HRRM W1,JOBSYM          ;NOW POINT TO FILDDT SYMBOLS
391                     TCOPY:  PUSHJ P,FETCH           ;GET A WORD
392                             JRST ERR
393                             MOVEM T,0(W1)           ;STASH IT
394                             AOS W1
```

```
                        AORJN R,TCOPY
                        JRST NOLPT              ;GO TRY IT
                DDTSTP: 7                       ;SAVE POSSIBLE SYMBOL POINTER
                >
```

```
399  000022  260040  002421' DD1:    PUSHJ P,CRF
400  000023  621000  000010 DD1.5:  TLZ F,ROF                ;CLOSE ANY OPEN REGISTER
401  000024  200240  003147'        MOVE T,[XWD SCHM,SCH]
402  000025  251240  000012         RLT T,ODF                ;LOAD ACS
403  000026  402000  002251' DD2:    CLEARM PRNC              ;PARENTHESES COUNT
404  000027  201040  003220'        MOVEI P,PS
405  000030  200240  002261' LIS:    MOVE T,ESTU
406  000031  202240  002262'        MOVEM T,ESTUT            ;INIT UNDEFINED SYM ASSEM
407  000032  630000  003150'        TDZ F,[XWD 777777-ROF-STF,LF1+CF1+SBF+2+Q2F]
408  000033  630000  003151' LIS0:   TDZ F,[XWD 777777-ROF-STF-FAF-SAF,NAF]
409  000034  402000  002247'        CLEARM,WRD
410  000035  402000  002252' LIS1:   CLEARM,FRASE
411  000036  201240  000001 LIS2:   MOVEI T,1
412  000037  202240  002256'        MOVEM T,FRASE1
413  000040  621000  040200         TLZ F,MLF+DVF
414  000041  621000  030044 L1:     TLZ F,CF+CCF+SF+FPF              ;TURN OFF CONTROL, SYL, PERIOD FLAG
415  000042  402000  002253'        CLEARM,SYL
416  000043  402000  002265' L1RPR:  CLEARM,SYM
417  000044  201240  000006         MOVEI T,6
418  000045  202240  002356'        MOVEM T,TEM              ;INIT SYMBOL COUNTER
419  000046  200240  003152'        MOVE T,[POINT 7,TXT]
420  000047  202240  003016'        MOVEM T,CHP              ;SETUP FOR OPEVAL SYMBOL
421  000050  402000  002257'        CLEARM,DEN
422  000051  402000  002250'        CLEARM,WRD2
423
424  000052  260040  002430' L2:     PUSHJ P,TIN              ;PICK UP CHARACTER
425  000053  301240  000141         CAIL T,"A"+40            ;LOWER CASE A
426  000054  303240  000172         CAILE T,"Z"+40           ;LOWER CASE Z
427  000055  254000  000057'        JRST .+2
428  000056  640240  000040         TRC T,40                 ;CHANGE LOWER CASE TO UPPER CASE
429  000057  603000  000040         TLNE F,CF                ;CONTROL FLAG
430  000060  254000  000065'        JRST L21
431  000061  307240  000132         CAIG T,"Z"               ;Z
432  000062  305240  000101         CAIGE T,"A"              ;A
433  000063  254000  000065'        JRST .+2
434  000064  254000  000252'        JRST LET
435  000065  200100  000005 L21:    MOVE R,T
436  000066  303240  000137         CAILE T,137      ;DISPATCH TABLE HAS ENTRIES ONLY .LE. 137
437  000067  254000  000151'        JRST ERR
438  000070  231100  000003         IDIVI R,3                ;REMAINDER GIVES COLUMN, QUOTIENT GIVES ROW
439  000071  135203  002506'        LDB W,BDISP(R+1)         ;GET 12 BIT ADDRESS FROM DISPATCH TABLE
440  000072  305200  000617         CAIGE W,MULT-DDT         ;FIRST EVAL ROUTINE
441  000073  254004  000000'        JRST DDT(W)
442
```

```
443  000074  200240  002253'          MOVE T,SYL
444  000075  627000  000020          TLZN F,LTF
445  000076  254000  000326'          JRST POWER
446  000077  200240  003153'          MOVE T,[XWD OPEVAL,EVAL]        ;GET ADDRESSES OF LOOKUP ROUTINES
447  000100  336000  002247'          SKIPN WRD                       ;IF C(WRD)=0, CALL OPEVAL FIRST, OTHERWISE EVAL FIRST
448  000101  207000  000005          MOVSS T
449  000102  202240  002360'          MOVEM T,SAVE
450  000103  254000  000106'          JRST L213
451
452  000104  557240  002360' L212:    HLRZS T,SAVE                    ;GET ADDRESS OF THE OTHER LOOKUP ROUTINE
453  000105  322240  000167'          JUMPE T,UND1                    ;IF ADR=0, THEN SYMBOL UNDEFINED
454  000106  260045  000000  L213:    PUSHJ P,(T)     ;CALL OPEVAL OR EVAL
455  000107  254000  000104'          JRST L212                       ;SYMBOL NOT FOUND
456  000110  623000  000002  L4:      TLZE F,MF
457  000111  210240  000005          MOVN T,T
458  000112  607000  000004          TLNN F,SF
459  000113  302000  000626          CAIE W,LPRN-DDT
460  000114  254000  000116'          JRST .+2
461  000115  254000  000626'          JRST LPRN

462
463  000116  250240  002256'          EXCH T,FRASE1
464  000117  607000  040000          TLNN F,DVF
465  000120  223240  002256'          IMULB T,FRASE1
466  000121  623000  040000          TLZE F,DVF
467  000122  233240  002256'          IDIVB T,FRASE1
468  000123  305200  000622          CAIGE W,ASSEM-DDT
469  000124  254004  000000'          JRST DDT(W)                     ;MULTIPLY OR DIVIDE
470  000125  273240  002252'          ADDB T,FRASE
471  000126  305200  000660          CAIGE W,SPACE-DDT
472  000127  254004  000000'          JRST DDT(W)                     ; + - @ ,
473
474  000130  270240  002247'          ADD T,WRD
475  000131  603000  100000          TLNE F,TIF                      ;TRUNCATE INDICATOR FLAG
476  000132  500240  002247'          HLL T,WRD                       ;TRUNCATE
477  000133  202240  002247'          MOVEM T,WRD
478  000134  607000  000001          TLNN F,QF
479  000135  200240  002254'          MOVE T,LWT
480  000136  402000  000002          CLEARM,R
481  000137  200370  002262'          MOVE W1,ESTUT
482  000140  316370  002261'          CAMN W1,ESTU
483  000141  254000  000144'          JRST L5
484  000142  303200  000735          CAILE W,CARR-DDT
485  000143  254000  000151'          JRST ERR
486  000144  307200  003664  L5:      CAIG W,RPRN-DDT
487  000145  254004  000000'          JRST DDT(W)
488  000146  261040  000512'          PUSH P,KILRET
489  000147  336000  002251'          SKIPN PRNC
490  000150  254004  000000'          JRST DDT(W)
```

```
  491
  492  000151  201300  000077  ERR:    MOVEI W1,"?"
  493  000152  254000  000156'         JRST WRONG1
  494  000153  201300  000125  UNDEF:  MOVEI W1,"U"
  495  000154  254000  000156'         JRST WRONG1
  496  000155  200300  003154' WRONG:  MOVE W1,[ASCII /XXX/]
  497  000156  201040  003220' WRONG1: MOVEI P,PS
  498  000157  260040  002240'         PUSHJ P,TEXT
  499  000160  260040  002424'         PUSHJ P,LCT             ;TYPE TAB
  500  000161  260040  002456'         PUSHJ P,LISTEN          ;GOBBLE ANY INPUT CHARACTER
  501  000162  255000  000000          JFCL
  502  000163  254000  000026'         JRST DD2
  503  000164  201040  003220' RET:    MOVEI P,PS
  504  000165  260040  002424'         PUSHJ P,LCT             ;COMMON RETURN FOR TAB:,JRST LIS
  505  000166  254000  000026'         JRST DD2
```

```
506
507  000167  200100  002262' UND1:   MOVE R,ESTUT            ;UNDEFINED SYM ASSEMBLER
508  000170  574140  002262'         HLRE S,ESTUT
509  000171  240140  777777         ASH S,-1                ;SETUP EVAL END TEST
510  000172  260040  000415'         PUSHJ P,EVAL2
511  000173  306200  000622         CAIN W,ASSEM-DDT
512  000174  607000  000010         TLNN F,ROF
513  000175  254000  000153'         JRST UNDEF
514  000176  332000  002251'         SKIPE PRNC
515  000177  254000  000153'         JRST UNDEF
516  000200  201240  000043         MOVEI T,"#"
517  000201  302200  000622         CAIE W,ASSEM-DDT
518  000202  260040  002440'         PUSHJ P,TOUT
519
520  000203  210100  003155'         MOVN R,[XWD 2,2]
521  000204  273100  002262'         ADDB R,ESTUT
522  000205  200240  002265'         MOVE T,SYM
523  000206  661240  040000         TLO T,GLOBAL
524  000207  202242  000000         MOVEM T,(R)

525  000210  550240  002272'         HRRZ T,LLOCO
526  000211  603000  000002         TLNE F,MF
527  000212  661240  400000         TLO T,400000
528  000213  202242  000001         MOVEM T,1(R)
529  000214  201240  000000         MOVEI T,0
530  000215  254000  000110'         JRST L4
531
532  000216  260040  002421' QUESTN: PUSHJ P,CRF             ;LIST UNDEFINED SYMBOLS
533  000217  200100  002261'         MOVE R,ESTU
534  000220  325100  000022' QUEST1: JUMPGE R,DD1
535  000221  200242  000000         MOVE T, (R)
536  000222  334300  002261'         SKIPA W1,ESTU
537
538  000223  270300  003155' QUEST2: ADD W1,[XWD 2,2]
539  000224  312246  000000         CAME T,(W1)
540  000225  254000  000223'         JRST QUEST2
541  000226  312100  000006         CAME R,W1
542  000227  254000  000232'         JRST QUEST4
543  000230  260040  002103'         PUSHJ P,SPT
544  000231  260040  002421'         PUSHJ P,CRF
545  000232  270100  003155' QUEST4: ADD R,[XWD 2,2]
546  000233  254200  000220'         JRST QUEST1
```

```
547  000234  405240  000017  NUM:    ANDI T,17                ;T HOLDS CHARACTER
548  000235  603000  020040          TLNE F,CF+FPF
549  000236  254000  000306'        JRST NM1
550  000237  200200  002253'        MOVE W,SYL
551  000240  242200  000003         LSH W,3
552  000241  270200  000005         ADD W,T
553  000242  202200  002253'        MOVEM W,SYL
554  000243  200200  002257'        MOVE W,DEN
555  000244  221200  000012         IMULI W,12               ;CONVERT TO DECIMAL
556  000245  270200  000005         ADD W,T
557  000246  202200  002257'        MOVEM W,DEN
558  000247  344240  000266'        AOJA T,LE1A
559
560  000250  334240  003156' DOLLAR: SKIPA T,[46+101-13]     ;RADIX 50 $ TO BE
561  000251  201240  000135  PERC:   MOVEI T,47+101-13       ;PERCENT SIGN
562  000252  641000  020004  LET:    TLC F,SF+FPF             ;EXPONENT IFF LTF'*FEF'*(T=105)*SF*FPF=1
563  000253  627000  420024          TLZN F,LTF+FEF+SF+FPF
564  000254  302240  000105         CAIE T,105               ; E
565  000255  665000  000020         TLOA F,LTF
566  000256  665000  400000         TLOA F,FEF
567  000257  254000  000265'        JRST LET1
568  000260  627000  000002         TLZN F,MF
569  000261  334300  002253'        SKIPA W1,SYL
570  000262  210300  002253'        MOVN W1,SYL
571  000263  202300  002263'        MOVEM W1,FSV
572  000264  402000  002257'        CLEARM DEN
573  000265  275240  000066  LET1:   SUBI T,101-13            ;FORM RADIX 50 SYMBOL
574  000266  661000  000005  LE1A:   TLO F,SF+QF
575  000267  200200  002265' LE2:    MOVE W,SYM
576  000270  201105  000066          MOVEI R,101-13(T)
577  000271  221200  000050         IMULI W,50               ;CONVERT TO RADIX 50
578  000272  270200  000005         ADD W,T
579  000273  375000  002356'        SOSGE TEM                ;IGNORE CHARACS AFTER 6
580  000274  254000  000052'        JRST L2
581  000275  202200  002265'        MOVEM W,SYM
582  000276  136100  003016'        IDPB R,CHP
583  000277  202200  002265'        MOVEM W,SYM
584  000300  254000  000052'        JRST L2
```

```
585  000301  250240  002250' NUM1:   EXCH T,WRD2              ;FORM NUMBER AFTER $
586  000302  221240  000012         IMULI T,12
587  000303  272240  002250'        ADDM T,WRD2
588  000304  660000  000001         TRO F,Q2F
589  000305  254000  000052'        JRST L2
590
591  000306  603000  000040  NM1:    TLNE F,CF
592  000307  254000  000301'        JRST NUM1
593  000310  201300  000006         MOVEI W1,6                ;FORM FLOATING POINT NUMBER
594  000311  350000  000312'        AOS NM1A
595  000312  201340  000000  NM1A:   MOVEI W2,0
596  000313  205100  201400         MOVSI R,201400
597  000314  622340  000001  NM1A1:  TRZE W2,1
598  000315  164106  002223'        FMPR R,FT(W1)
599  000316  322340  000321'        JUMPF W2,NM1B
600  000317  242340  777777         LSH W2,-1
601  000320  367300  000314'        SOJG W1,NM1A1
602  000321  205305  211000  NM1B:   MOVSI W1,211000(T)
603  000322  164100  000006         FMPR    R,W1              ;COMPUTE VALUE OF NEW DIGIT

604  000323  147100  002264'        FADRB   R,FH              ;ADD VALUE INTO FLOATING NO.
605  000324  202100  002253'        MOVEM R,SYL
606  000325  344240  000266'        AOJA T,LE1A
607
608  000326  607000  400000  POWER:  TLNN F,FEF
609  000327  254000  001110'        JRST L4          ;NO EXPONENT
610  000330  302200  000624'        CAIE W,PLUS
611  000331  306200  000623'        CAIN W,MINUS
612  000332  662000  004000         TROE F,POWF
613  000333  624000  004000         TRZA F,POWF
614  000334  254004  000000         JRST (W)                  ; E+-
615
616  000335  200340  002257'        MOVE W2,DEN
617  000336  402000  002252'        CLEARM FRASE
618  000337  201300  002222'        MOVEI W1,FT-1
619  000340  623000  000002         TLZE F,MF
620  000341  201300  002231'        MOVEI W1,FT01
621  000342  334240  002263'        SKIPA T,FSV
622  000343  242340  777777  POW2:   LSH W2,-1
623  000344  622340  000001         TRZE W2,1
624  000345  164246  000000         FMPR T,(W1)
625  000346  322340  001110'        JUMPF W2,L4
626  000347  364300  000343'        SOJA W1,POW2
```

```
  627  000350  200240  002271' PERIOD: MOVE T,LLOC
  628  000351  603000  000004         TLNE F,SF                ;SYLLABLE STARTED
  629  000352  200240  002257'        MOVE T,DEN
  630  000353  202240  002253'        MOVEM T,SYL
  631  000354  603000  020000         TLNE    F,FPF            ;HAS A PERIOD BEEN SEEN BEFORE?
  632  000355  661000  000020         TLO     F,LTF            ;YES, TWO PERIODS MAKES A SYMBOL
  633  000356  667000  020005         TLON F,FPF+SF+QF
  634  000357  201240  000000         MOVEI T,0
  635  000360  231240  000400         IDIVI T,400
  636  000361  332000  000005         SKIPE T
  637  000362  641240  243000         TLC T,243000
  638  000363  641300  233000         TLC W1,233000
  639  000364  140240  003157'        FAD     T,[0]            ;NORMALIZE T AND W1
  640  000365  140300  003157'        FAD     W1,[0]
  641  000366  144240  000006         FADR    T,W1
  642  000367  202240  002264'        MOVEM T,FH
  643  000370  513000  000312'        HLLZS NM1A
  644  000371  201240  000045         MOVEI T,45               ;RADIX 50 PERIOD
  645  000372  254000  000267'        JRST LE2
  646
  647  000373  201240  002275' PILOC:  MOVEI T,SAVPI           ;ADR SETUP FOR $I
  648  000374  665240  000000  QUANIN: TLOA T,(DDTINT)         ;MARK FOR ADR INTERNAL TO DDT
  649  000375  200240  002254' QUAN:   MOVE T,LWT              ;PICK UP LAST QUANTITY TYPED
  650  000376  202240  002253' QUAN1:  MOVEM T,SYL
  651  000377  661000  000005  QUAN2:  TLO F,SF+QF             ;WRD,SYL STARTED
  652  000400  621000  010040         TLZ F,CF+CCF
  653  000401  254000  000052'        JRST L2
  654
  655  000402                  CONTRO:                          ;SOME KIND OF ALTMODE
  656  000402  201240  000044  IFN EDDT&1,<    MOVEI T,"$"     ;$
  657  000403  260040  002440'         PUSHJ P,TOUT            ;TYPE OUT $
  658                           >
  659  000404  663000  000040         TLOE F,CF
  660  000405  661000  010000         TLO F,CCF
  661  000406  254000  000052'        JRST L2
```

```
662  000407  200100  002260' EVAL:   MOVE R,PRGM              ;LOOK UP SYMBOL
663  000410  525300  237777  EVAL0:  HRLOI W1,37777+DELI
664  000411  574160  002274'         HLRE S,@SYMP
665  000412  240140  777777         ASH S,-1          ;SETUP END TEST
666  000413  254000  000420'         JRST EVAL3
667
668  000414  270100  003155' EVAL1:  ADD R,[XWD 2,2]
669  000415  331000  000002  EVAL2:  SKIPL R
670  000416  200120  002274'         MOVE R,@SYMP
671  000417  347140  000432'         AOJG S,CPOPJ              ;TRNASFER IF NO SYMBOL FOUND
672  000420  200242  000000  EVAL3:  MOVE T,(R)
673  000421  430240  002265'         XOR T,SYM
674  000422  607240  740000         TLNN T,PNAME
675  000423  665300  100000         TLOA W1,LOCAL
676  000424  612240  000006         TDNE T,W1
677  000425  254000  000414'         JRST EVAL1
678  000426  607240  340000         TLNN T,340000
679  000427  254000  000414'         JRST EVAL1
680  000430  200242  000001         MOVE T,1(R)

681  000431  350001  000000  CPOPJ1: AOS (P)           ;FOUND SYMBOL, SKIP
682  000432  263040  000000  CPOPJ:  POPJ P,
683
684                          ;BIT 40 - DELETE OUTPUT
685                          ; 20 - DELETE INPUT
686                          ; 10 - LOCAL
687                          ; 04 -GLOBAL
688                          ; NO BITS - PROGRAM NAME
689
690  000433  260040  002502' TEXI:   PUSHJ P,TEXIN             ;INPUT TEXT
691  000434  202240  002253'         MOVEM T,SYL
692  000435  201300  000005         MOVEI W1,5
693  000436  201200  000000         MOVEI T-1,0
694  000437  260040  002502'         PUSHJ P,TEXIN
695  000440  306240  000033         CAIN T,33                 ;NEW ALT MODE, ESCAPE
696  000441  254000  000377'         JRST QUAN2
697  000442  603000  000040         TLNE F,CF
698  000443  254000  000460'         JRST SIXBIN
699  000444  334000  000000         SKIPA
700  000445  260040  002502' TEXI2:  PUSHJ P,TEXIN
701  000446  316240  002253'         CAMN T,SYL
702  000447  364300  000453'         SOJA W1,TEXI3
703  000450  241240  777771         ROT T,-7
704  000451  246270  000007         LSHC T-1,7
705  000452  364300  000445'         SOJA W1,TEXI2
706
707  000453  246200  777735  TEXI3:  LSHC T-1,-43
708  000454  321300  000376'         JUMPL W1,QUAN1
709  000455  242240  000007         LSH T,7
710  000456  364300  000454'         SOJA W1,.-2
```

```
000457  260040  002502' SIXBI1: PUSHJ P,TEXIN     ; INPUT TEXT (SIXBIT)
000460  316240  002253' SIXBIN: CAMN T,SYL
000461  254000  000476'         JRST SIXBI2
000462  301240  000141          CAIL T,"A"+40              ;IS CHAR BETWEEN LOWER CASE "A" AND
000463  303240  000172          CAILE T,"Z"+40             ; LOWER CASE "Z"?
000464  334000  000000          SKIPA                      ;NO
000465  640240  000040          TRC T,40                   ;YES, CONVERT TO UPPER CASE
000466  301240  000040          CAIL T," "                 ;IS CHAR IN SIXBIT SET?
000467  303240  000137          CAILE T,"←"
000470  254000  000151'         JRST ERR                   ;NO
000471  405240  000077          ANDI T,77                  ;YES, MASK TO 6 BITS
000472  640240  000040          TRC T,40                   ;CONVERT TO SIXBIT FORM
000473  241240  777772          ROT T,-6
000474  246200  000006          LSHC T-1,6
000475  364300  000457'         SOJA W1,SIXBI1
000476  200240  000004  SIXBI2: MOVE T,T-1
000477  321300  000376'         JUMPL W1,QUAN1
000500  242240  000006          LSH T,6
000501  364300  000477'         SOJA W1,.-2

000502  607000  000020  KILL:   TLNN F,LTF                 ;DELETE SYMBOLS
000503  254070  000151'         JRST ERR
000504  260040  000407'         PUSHJ P,EVAL
000505  254020  000513'         JRST KILL1
000506  201240  000002          MOVEI T,DELO/200000                ;DELETE OUTPUT
000507  603000  010000          TLNE F,CCF
000510  201240  000001          MOVEI T,DELI/200000                ;NO INPUT OR OUTPUT
000511  137240  003160'         DPB T,[POINT 2,(R),1]      ;LEFT 2 BITS IN SYMBOL
000512  254000  000164' KILRET: JRST RET                   ;USED AS A CONSTANT


000513  200100  002261' KILL1:  MOVE R,ESTU               ;REMOVE UNDEFINED SYMS
000514  325100  000153'         JUMPGE R,UNDEF
000515  260040  000410' KILL2:  PUSHJ P,EVAL0
000516  254000  000164'         JRST RET
000517  260040  000521'         PUSHJ P,REMUN
000520  254000  000515'         JRST KILL2

000521  200140  003155' REMUN:  MOVE S,[XWD 2,2]         ;REMOVE ONE UNDEFINED SYMBOL
000522  273140  002261'         ADDB S,ESTU
000523  200203  777776          MOVE W,-2(S)
000524  202202  000000          MOVEM W,(R)
000525  200203  777777          MOVE W,-1(S)
000526  202202  000001          MOVEM W,1(R)
000527  263040  000000          POPJ P,
```

```
  756  000530  607000  000020  TAG:    TLNN F,LTF    ; NO LETTERS IS ERROR
  757  000531  254000  000151'        JRST ERR    ; GO SAY ERROR
  758  000532  603000  001000        TLNE F,FAF    ; DEFINE SYMBOLS
  759  000533  254000  000540'        JRST DEFIN                ;A<B!
  760  000534  603000  000040        TLNE F,CF                 ;DEFINE SYMBOL AS OPEN REGISTER
  761  000535  254000  000604'        JRST SETNAM
  762  000536  200200  002272'        MOVE W,LLOCO
  763  000537  552200  002267'        HRRZM W,DEFV
  764
  765  000540  260040  000407' DEFIN:  PUSHJ P,EVAL              ;DEFINED SYMBOL?
  766  000541  254000  000543'        JRST DEF1                 ;NO - DEFINE
  767  000542  254000  000553'        JRST DEF2                 ;YES, REDEFINE
  768  000543  210100  003155' DEF1:   MOVN R,[XWD 2,2]
  769  000544  273120  002274'        ADDB R,@SYMP     ;MOVE UNDEFINED TABLE 2 REGISTERS
  770  000545  550240  002261'        HRRZ T,ESTU
  771  000546  275240  000002        SUBI T,2
  772  000547  504240  002261'        HRL T,ESTU
  773  000550  542240  002261'        HRRM T,ESTU
  774  000551  335000  002261'        SKIPGE ESTU

  775  000552  251242  777777        BLT T,-1(R)
  776  000553  200240  002267' DEF2:  MOVE T,DEFV
  777  000554  202242  000001        MOVEM T,1(R)              ;PUT IN NEW VALUE
  778  000555  205240  040000        MOVSI T,GLOBAL
  779  000556  437240  002265'        IORB T,SYM
  780  000557  202242  000000        MOVEM T,(R)               ;PUT IN NEW SYM AS GLOBAL
  781  000560  200100  002261'        MOVE R,ESTU
  782
  783  000561  325100  000164' DEF3:  JUMPGE R,RET             ;PATCH IN VALUE FOR UNDEF SYM ENTRY
  784  000562  200240  002265'        MOVE T,SYM
  785  000563  312242  000000        CAME T,(R)
  786  000564  254000  000602'        JRST DEF4
  787  000565  200140  002267'        MOVE S,DEFV
  788  000566  335002  000001        SKIPGE, 1(R)
  789  000567  210140  000003        MOVN S,S
  790  000570  261040  000002        PUSH P,R
  791  000571  200102  000001        MOVE R,1(R)
  792  000572  260040  001454'        PUSHJ P,FETCH
  793  000573  254000  000151'        JRST ERR
  794  000574  270140  000005        ADD S,T
  795  000575  542140  000005        HRRM S,T
  796  000576  260040  001446'        PUSHJ P,DEP
  797  000577  255000  000000         JFCL
  798  000600  262040  000002        POP P,R
  799  000601  260040  000521'        PUSHJ P,REMUN
  800  000602  270100  003155' DEF4:  ADD R,[XWD 2,2]          ;REMOVE THE NOW DEFINED SYMBOL
  801  000603  254000  000561'        JRST DEF3
```

```
802  000604  200120  002274' SETNAM: MOVE R,@SYMP             ;SET PROGRAM NAME - DOLLAR COLON
803  000605  200200  000002  SET1:   MOVE W,R
804  000606  325100  000153' SET2:   JUMPGE R,UNDEF
805  000607  200242  000000          MOVE T,(R)
806  000610  270100  003155'         ADD R,[XWD 2,2]
807  000611  603240  740000          TLNE T,PNAME
808  000612  254000  000606'         JRST SET2
809  000613  312240  002265'         CAME T,SYM
810  000614  254000  000605'         JRST SET1
811  000615  202200  002260'         MOVEM W,PRGM
812  000616  254000  000164'         JRST RET
813
814                                  ;***ROUTINES BEYOND HERE EVALUATE THEIR ARGUMENT***
815  000617  665000  000300  MULT:   TLOA F,PTF+MLF           ;*
816  000620  661000  040100  DIVD:   TLO F,OVF+PTF            ;SINGLE QUOTE
817  000621  254000  000041'         JRST L1
818
819  000622  254000  000624' ASSEM:  JRST PLUS                ;#
820  000623  661000  000002  MINUS:  TLO F,MF
821  000624  661000  000100  PLUS:   TLO F,PTF
822  000625  254000  000036'         JRST LIS2
823
824  000626  311040  003161' LPRN:   CAML P,[XWD LPDL-4,0]    ;LEFT PARENTHESIS
825  000627  254000  000151'         JRST ERR
826  000630  261040  000000          PUSH P,F                 ;RECURSE FOR OPEN PAREN
827  000631  261040  002247'         PUSH P,WRD
828  000632  261040  002252'         PUSH P,FRASE
829  000633  261040  002256'         PUSH P,FRASE1
830  000634  350000  002251'         AOS,PRNC
831  000635  254000  000030'         JRST LIS
832
833  000636  515200  000020  INDIRECT:       HRLZI W,20              ;@
834  000637  437200  002247'         IORB W,WRD
835  000640  661000  000001          TLO F,QF
836  000641  254000  000036'         JRST LIS2
837
838  000642  200100  000005  ACCF:   MOVE R,T                 ;COMMA PROCESSOR
839  000643  205240  000000  ACCCF:  MOVSI T,-.               ;LEFT HALF OF A,,B
840  000644  663000  200000          TLOE F,COMF              ;COMMA TYPED BEFORE?
841  000645  254000  000656'         JRST ACCF1               ;YES
842  000646  542100  000643'         HRRM R,ACCCF             ;NO, SAVE LEFT HALF OF A,,B
843  000647  510240  000002          HLLZ T,R
844  000650  135300  003162'         LDB W1,[POINT 3,WRD,2]   ;CHECK FOR IO INSTRUCTION
845  000651  231300  000007          IDIVI W1,7
846  000652  242176  000027          LSH R,27(W1)
847  000653  270240  000002          ADD T,R
848  000654  273240  002247'         ADDB T,WRD
849  000655  254000  000661'         JRST SPACE+1
```

```
850  000656  202240  002247' ACCF1:  MOVEM T,WRD              ;SET LEFT HALF OF A,,B
851  000657  254000  000661'         JRST SPACE+1
852
853  000660  603000  000001  SPACE:  TLNE F,QF
854  000661  661000  100000          TLO F,TIF
855  000662  621000  000102          TLZ F,MF+PTF
856  000663  254000  000035'         JRST LIS1
857
858  000664  607000  000001  RPRN:   TLNN F,QF                 ))
859  000665  201240  000000          MOVEI T,0
860  000666  204240  000005          MOVS T,T
861  000667  375000  002251'         SOSGE,PRNC
862  000670  254000  000151'         JRST ERR
863  000671  262040  002256'         POP P,FRASE1
864  000672  262040  002252'         POP P,FRASE
865  000673  262040  002247'         POP P,WRD
866  000674  262040  000000          POP P,F
867  000675  603000  000100          TLNE F,PTF
868  000676  603000  000004          TLNE F,SF

869  000677  254000  000703'         JRST RPRN1
870  000700  202240  002253'         MOVEM T,SYL
871  000701  661000  000005          TLO F,QF+SF
872  000702  254000  000043'         JRST L1RPR
873  000703  273240  002247' RPRN1:  ADDB T,WRD
874  000704  661000  000001          TLO F,QF
875  000705  254000  000042'         JRST L1RPR-1
```

```
                        ;REGISTER EXAMINATION LOGIC

000706  260040  000766' LINEF:  PUSHJ P,DEPRA    ;NEXT REGISTER
                        IFE EDDT&1,<PUSHJ P,CRNRB
                                JRST .+2>
000707  260040  002421' LI0:    PUSHJ P,CRF
000710  350240  002271'         AOS T,LLOC
000711                  LI1:            ;PUSHJ P,LINCHK         ;TRUNCATE ADRS (UNLESS INSIDE DDT)
000711  552240  002271'         HRRZM T,LLOC
000712  552240  002272'         HRRZM T,LLOCO
000713  260040  001620'         PUSHJ P,PAD
000714  201240  000057          MOVEI T,"/"
000715  603000  004000          TLNE F,STF
000716  201240  000041          MOVEI T,"!"
000717  260040  002440'         PUSHJ P,TOUT
000720  621000  000010  LI2:    TLZ F,ROF
000721  260040  002424'         PUSHJ P,LCT
000722  200100  002272'         MOVE R,LLOCO
000723  260040  001454'         PUSHJ P,FETCH
000724  254000  000151'         JRST ERR
000725  661000  000010          TLO F,ROF
000726  603000  004000          TLNE F,STF
000727  254000  000026'         JRST DD2
000730  254000  001532'         JRST CONSYM             ;RETURN IS A POPJ

                        REPEAT 0,<
                        LINCHK: CAML T,[DDTINT SAVPIJ   ;TRUNCATE ADDRESSES
                                CAMLE T,[DDTINT BNADR+2]
                                HRRZS T
                                MOVEM T,LLOC
                                MOVEM T,LLOCO
                                POPJ P,
                        >

000731  260040  000766' VARRW:  PUSHJ P,DEPRA            ;↑
000732  260040  002421'         PUSHJ P,CRF
000733  370240  002271'         SOS T,LLOC
000734  254000  000711'         JRST LI1

000735  260040  000766' CARR:   PUSHJ P,DEPRA            ;CLOSE REGISTER
000736  254000  000022'         IFN EDDT&1,<JRST DD1>
                                IFE EDDT&1,<    PUSHJ P,TIN
                                CAIN T,15
                                JRST .-2
                                JRST DD1,5>
```

```
   922
   923  000737  664000  002400  OCON:   TROA F,LF1+CF1          ;OPEN AS CONSTANT
   924  000740  620000  000400  OSYM:   TRZ F,CF1                ;OPEN SYMBOLICALLY
   925  000741  664000  002000          TROA F,LF1
   926  000742  665000  004000  SUPTYO: TLOA F,STF               ;SUPPRESS TYPEOUT
   927  000743  621000  004000  SLASH:  TLZ F,STF                ;TYPE OUT REGISTER
   928  000744  607000  000001          TLNN F,QF                ;WAS ANY QUANTITY TYPED?
   929  000745  254000  000751'         JRST SLAS1               ;NO. DO NOT CHANGE MAIN SEQUENCE
   930  000746  200100  002271'         MOVE R,LLOC              ;YES. SAVE OLD SEQUENCE AND
   931  000747  202100  002273'         MOVEM R,SAVLOC
   932  000750  552240  002271'         HRREM T,LLOC     ;PUSHJ P,LINCHK         ;TRUNCATE ADRS- SET UP NEW SEQUENCE
   933  000751  552240  002272' SLAS1:  HRREM T,LLOCO
   934  000752  254000  000720'         JRST LI2
   935
   936  000753  607000  000010  ICON:   TLNN F,ROF       ;REGISTER OPENED OR ERR
   937  000754  254000  000151'         JRST ERR
   938  000755  260040  001436'         PUSHJ P,DEPRS
   939  000756  254000  000751'         JRST SLAS1
```

```
940  000757  260040  001436' TAB:    PUSHJ P,DEPRS   ;OPEN REGISTER OF Q
941  000760  201245  777777          MOVEI T,-1(T)
942  000761  250240  002271'         EXCH T,LLOC             ;SET UP NEW SEQUENCE AND
943  000762  202240  002273'         MOVEM T,SAVLOC          ;SAVE OLD SEQUENCE
944  000763  561240  700000          HRROI T,700000  ;3 RUBOUTS
945  000764  260040  002237'         PUSHJ P,TEXTT
946  000765  254000  000707'         JRST LI0
947
948  000766  200100  002273' DEPRA:  MOVE R,SAVLOC
949  000767  603000  000040          TLNE F,CF               ;RESTORE OLD SEQUENCE IF $CR,$CF, OR
950  000770  250100  002271'         EXCH R,LLOC             ;IF $↑ OR $BS WAS TYPED
951  000771  202100  002273'         MOVEM R,SAVLOC          ;SETUP "NEW" OLD SEQUENCE
952  000772  603000  000010          TLNE F,ROF              ;IF REGISTER IS BEING CHANGED
953  000773  607000  000001          TLNN F,QF               ;REMOVE ALL PREVIOUS UNDEFINED
954  000774  254000  001436'         JRST DEPRS              ;SYMBOL REFERENCES TO IT
955  000775  200100  002261'         MOVE R,ESTU
956  000776  202300  002261'         MOVEM W1,ESTU
957  000777  325100  001436' DEPRA2: JUMPGE R,DEPRS
958  001000  550222  000001          HRRZ W,1(R)
959  001001  316200  002272'         CAMN W,LLOCO
960  001002  260040  000521'         PUSHJ P,REMUN
961  001003  270100  003155'         ADD R,[XWD 2,2]
962  001004  254000  000777'         JRST DEPRA2
963
964  001005  664000  002400  EQUAL:  TROA F,LF1+CF1          ;=
965  001006  620000  000400  ESYM:   TRZ F,CF1                ;@
966  001007  660000  002000          TRO F,LF1
967  001010  260040  001532'         PUSHJ P,CONSYM
968  001011  254000  000164'         JRST RET
969
970  001012  242240  777742 R50PNT: LSH T,-36          ;RADIX 50 SYMBOL PRINTER
971  001013  620240  000003          TRZ T,3
972  001014  260040  002045'         PUSHJ P,TOC
973  001015  260040  002426'         PUSHJ P,TSPC
974  001016  201300  002254'         MOVEI W1,LWT    ;SETUP FOR SPT
975  001017  254000  002103'         JRST SPT
976
977  001020  211340  000006 SIXBP:   MOVNI W2,6                ;SIXBIT PRINTER
978  001021  200300  002254'         MOVE W1,LWT
979  001022  201240  000000 SIXBP1:  MOVEI T,0
980  001023  245240  000006          ROTC T,6
981  001024  271240  000040          ADDI T,40
982  001025  260040  002440'         PUSHJ P,TOUT
983  001026  341340  001022'         AOJL W2,SIXBP1
984  001027  263040  000000          POPJ P,
```

```
  985                                   ;MODE CONTROL SWITCHES
  986
  987  001030  201170  002426  TEXO:    MOVEI R,TEXTT-HLFW      ;T ASSUME 7 BIT ASCII
  988  001031  202240  002250'          MOVE T,WRD2
  989  001032  306240  000006           CAIN T,6                ;CHECK FOR $6T
  990  001033  201170  777207           MOVEI R,SIXBP-HLFW      ;SET MODE SWITCH FOR SIXBIT
  991  001034  306240  000005           CAIN T,5                ;CHECK FOR $5T
  992  001035  201170  777201           MOVEI R,R50PNT-HLFW     ;SET MODE SWITCH FOR RADIX 50
  993  001036  271170  777461  HWRDS:   ADDI R,HLFW-TFLOT              ;H
  994  001037  271170  000371  SFLOT:   ADDI R,TFLOT-PIN               ;F
  995  001040  271170  777472  SYMBOL:  ADDI R,PIN-FTOC         ;S
  996  001041  271170  002045' CON:     ADDI R,FTOC             ;C
  997  001042  552170  000010           HRRZM R,SCH
  998  001043  254000  001056'          JRST BASE1
  999
 1000  001044  622000  000001  RELA:    TRZE F,Q2F              ;CHANGE ADDRESS MODE TO RELATIE
 1001  001045  254000  001052'          JRST BASECH
 1002  001046  201170  777555           MOVEI R,PADSO-TOC
 1003  001047  271170  002045' ABSA:    ADDI R,TOC              ;A
 1004  001050  552170  000011           HRRZM R,AR
 1005  001051  254000  001056'          JRST BASE1
 1006
 1007  001052  202240  002250' BASECH:  MOVE T,WRD2             ;$NR  CHANGE OUTPUT RADIX TO N, N>1
 1008  001053  306240  000002           CAIGE T,2
 1009  001054  254000  000151'          JRST ERR
 1010  001055  552240  000012           HRRZM T,ODF
 1011  001056  204140  003147' BASE1:   MOVS S,[XWD SCHM,SCH]
 1012  001057  607000  010000           TLNN F,CCF
 1013  001060  254000  000035'          JRST LIS1
 1014  001061  251140  002354'          BLT S,ODFM      ;WITH $$, MAKE MODES PERMANENT
 1015  001062  254000  000164'          JRST RET
 1016
 1017  001063  202240  002254' SEMIC:   MOVEM T,LWT             ;SEMICOLON TYPES IN CURRENT MODE
 1018  001064  254020  000010           JRST @SCH
```

```
1019                            ;GO AND EXECUTE LOGIC
1020
1021  001065  505240  254000  GO:     HRLI T,(JRST)              ;G
1022                            IFE EDDT&1,<
1023                                    TLON F,QF
1024                                    HRR T,JOBSA>    ;GET STARTING ADDRESS
1025
1026  001066  607000  000001  XEC:    TLNN F,QF                  ;X
1027  001067  254000  000151'         JRST ,ERR
1028  001070  202240  002356' XEC0:   MOVEM T,TEM
1029  001071  260040  002421'         PUSHJ P,CRF
1030  001072  260040  002474'         PUSHJ P,TTYLEV
1031  001073  260040  001337'         PUSHJ P,INSRTB
1032  001074  265240  002401'         JSP T,RESTORE
1033  001075  256000  002356'         XCT,TEM
1034  001076  254000  000000' XEC1:   JRST DDT                   ;USED  AT PROC0
1035  001077  264000  002360'         JSR,SAVE
1036  001100  260040  001353'         PUSHJ P,REMOVB
1037  001101  260040  002421'         PUSHJ P,CRF
1038  001102  254000  000022'         JRST DD1
```

```
1039                                    ;BREAK POINT LOGIC
1040                                    BP1:    REPEAT NBP,<    0               ;JSR TO HERE FOR BREAKPOINT
1041                                            JSA T, BCOM
1042                                            0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1043                                    >
1044  001103  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1045  001104  266240  001133'                 JSA T, BCOM
1046  001105  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1047
1048  001106  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1049  001107  266240  001133'                 JSA T, BCOM
1050  001110  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1051
1052  001111  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1053  001112  266240  001133'                 JSA T, BCOM
1054  001113  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1055
1056  001114  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1057  001115  266240  001133'                 JSA T, BCOM

1058  001116  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1059
1060  001117  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1061  001120  266240  001133'                 JSA T, BCOM
1062  001121  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1063
1064  001122  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1065  001123  266240  001133'                 JSA T, BCOM
1066  001124  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1067
1068  001125  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1069  001126  266240  001133'                 JSA T, BCOM
1070  001127  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1071
1072  001130  000000  000000                   0               ;JSR TO HERE FOR BREAKPOINT
1073  001131  266240  001133'                 JSA T, BCOM
1074  001132  000000  000000                   0               ;HOLDS INSTRUCTION WHILE BREAKPOINT IS IN PLACE
1075
1076
1077                    001105' B1INS=BP1+2
1078                    001130' BPN=.-3
```

```
1079  001133  000000  000000  BCOM:   0
1080  001134  262240  001303'         POP T,LEAV                ;MOVE INSTRUCTION TO LEAV
1081  001135  201245  001176         MOVEI T,B1SKP-B1INS+1(T)
1082  001136  542240  001145'         HRRM T,BCOM3              ;CONDITIONAL BREAK SETUP
1083  001137  201245  000001         MOVEI T,B1CNT-B1SKP(T)
1084  001140  542240  001147'         HRRM T,BCOM2              ;PROCEDE COUNTER SETUP
1085  001141  200245  776600         MOVE T,BP1-B1CNT(T)
1086  001142  621240  010000  IFN EDDTR1,<     TLZ T,010000              ;TURN OFF USER MODE BIT>
1087  001143  502240  001164'         HLLM T,LEAV1              ;SAVE FLAGS FOR RESTORING
1088  001144  250240  001133'         EXCH T,BCOM
1089
1090  001145  332000  002302' BCOM3:  SKIPE B1SKP              ;ADDR MOD TO LOOK AT COND. INST.
1091  001146  256020  001145'         XCT @.-1
1092  001147  377000  002303' BCOM2:  SOSG B1CNT              ;ADDR MOD TO LOOK AT PROCEED COUNTER
1093  001150  254000  001165'         JRST BREAK
1094
1095  001151  202240  002337'         MOVEM T,AC0+T
1096  001152  135240  003163'         LDB T,[POINT 9,LEAV,8]    ;GET INSTRUCTION
1097  001153  301240  000264           CAIL T,264        ;JSR
1098  001154  303240  000266           CAILE T,266       ;JSA,JSP
1099  001155  606240  000700           TRNN T,700        ;UUO
1100  001156  254000  001237'         JRST PROC1                ;MUST BE INTERPRETED
1101  001157  302240  000260           CAIE T,260        ;PUSHJ
1102  001160  306240  000256           CAIN T,256        ;XCT
1103  001161  254000  001237'         JRST PROC1                ;MUST BE INTERPRETED
1104  001162  200240  002337'         MOVE T,AC0+T
1105  001163  254120  001164'         JRST 2,@LEAV1             ;RESTORE FLAGS, GO TO LEAV
1106
1107  001164  000000  001303' LEAV1:  XWD 0,LEAV
```

```
1108  001165  264000  002360' BREAK:  JSR SAVE                ;SAVE THE WORLD
1109  001166  260040  001353'         PUSHJ P,REMOVB          ;REMOVE BREAKPOINTS
1110  001167  370240  001145'         SOS T,BCOM3
1111  001170  553000  000005         HRRZS T                 ;GET ADR OF CONDITIONAL BREAK INST
1112  001171  275240  002276'         SUBI T,B1ADR-3          ;CHANGE TO ADDRESS OF $0B
1113  001172  231240  000003         IDIVI T,3               ;QUOTIENT IS BREAK POINT NUMBER
1114  001173  542240  001215'         HRRM T,BREAK2           ;SAVE BREAK POINT #
1115  001174  200300  003164'         MOVE W1,[ASCII /$0B>/]  ;PRELIMINARY TYPEOUT MESSAGE
1116  001175  337020  001147'         SKIPG @BCOM2            ;TEST PROCEED COUNTER
1117  001176  660300  000174         TRO W1,">"+1            ;CHANGE T TO /$0B>>/
1118  001177  137240  003165'         DPB T,[POINT 4,W1,13]   ;INSERT BREAK POINT # IN MESSAGE
1119  001200  260040  002242'         PUSHJ P,TEXT2
1120  001201  200240  001133'         MOVE T,BCOM
1121  001202  502240  002275'         HLLM T, SAVPI           ;SAVE PROCESSOR FLAGS
1122  001203  201245  777777         MOVEI T,-1(T)
1123  001204  260040  001620'         PUSHJ P,PAD             ;TYPE PC AT BREAK
1124  001205  550260  001145'         HRRZ T,@BCOM3
1125  001206  542240  001231'         HRRM T,PROC0            ;SETUP ADDRESS OF BREAK
1126  001207  554260  001145'         HLRZ T,@BCOM3

1127  001210  322240  001214'         JUMPE T,BREAK1          ;TEST FOR REGISTER TO EXAMINE
1128  001211  260040  002424'         PUSHJ P,LCT             ;PRINT TAB
1129  001212  554260  001145'         HLRZ T,@BCOM3
1130  001213  260040  000711'         PUSHJ P,LI1             ;EXAMINE REGISTER C($NB)LEFT
1131  001214  205140  400000  BREAK1: MOVSI S,400000
1132  001215  241140  000000  BREAK2: ROT S,.-.               ;ROT BY # OF BREAK POINT
1133  001216  260040  002456'         PUSHJ P,LISTEN          ;DONT PROCEED IF TTY KEY HIT
1134  001217  616140  002331'         TDNN S,AUTOPI           ;DONT PROCEED IF NOT AUTOMATIC
1135  001220  254000  000164'         JRST RET                ;DONT PROCEED
1136  001221  254000  001227'         JRST PROCD1
1137
1138  001222  607000  000001  PROCEDE: TLNN F,OF              ;N$P    ;PROCEED AT A BREAKPOINT
1139  001223  201240  000001         MOVEI T,1
1140  001224  202260  001147'         MOVEM T,@BCOM2
1141  001225  550100  001145'         HRRZ R,BCOM3
1142  001226  260040  001426'         PUSHJ P,AUTOP
1143  001227  260040  002421' PROCD1: PUSHJ P,CRF
1144  001230  260040  002474'         PUSHJ P,TTYLEV
1145  001231  551100  001076' PROC0:  HRRZI R,XEC1             ;MODIFIED TO ADDR OF BREAKPOINT
1146  001232  260040  001454'         PUSHJ P,FETCH
1147  001233  254000  001310'         JRST BPLUP1             ;ONLY GET HERE IF MEMORY SHRANK
1148  001234  202240  001303'         MOVEM T,LEAV
1149  001235  260040  001337'         PUSHJ P,INSRTB
1150  001236  254000  001242'         JRST PROC2
1151
1152  001237  200240  002337' PROC1:  MOVE T,AC0+T
1153  001240  264000  002360'         JSR SAVE
1154  001241  255000  000000         JFCL
1155  001242  201200  000100  PROC2:  MOVEI W,100
1156  001243  202200  002357'         MOVEM W,TEM1            ;SETUP MAX LOOP COUNT
1157  001244  254000  001253'         JRST IXCT5
```

```
1158                                IXCT4:  IFE EDDT&1,<    SUBI T,041
1159                                        JUMPE T,RPLUP
1160                                        AOJGE T,IXCT6>                          ;DONT PROCEDE FOR INIT
1161                                                                                ;DONT INTERPRET FOR SYSTEM UUOS
1162  001245  202100  000040                MOVEM R,40                            ;INTERPRET FOR NON-SYSTEM UUOS
1163  001246  201100  000041                MOVEI R,41
1164  001247  371000  002357' IXCT:   SOSL TEM1
1165  001250  260040  001454'              PUSHJ P,FETCH
1166  001251  254000  001307'              JRST BPLUP                            ;BREAKPOINT LOOPING OR FETCH FAILED
1167  001252  202240  001303'              MOVEM T,LEAV
1168  001253                        IXCT5:  IFN EDDT&1,<
1169  001253  135240  003163'              LDB T,[POINT 9,LEAV,8]                ;GET INSTRUCTION
1170  001254  306240  000254                CAIN T,254                            ;DON'T DO ANYTHING TO JRST IN EXEC MODE
1171  001255  254000  001302'              JRST IXCT6>
1172  001256  515740  002332'              HRLZI 17,AC0
1173  001257  251740  000017                BLT 17,17
1174  001260  201260  001303'              MOVEI T,@LEAV
1175  001261  137240  003166'              DPB T,[POINT 23,LEAV,35]              ;STORE EFFECTIVE ADDRESS
1176  001262  135300  003167'              LDB W1,[POINT 4,LEAV,12]              ;PICK UP AC FIELD
1177  001263  135240  003163'              LDB T,[POINT 9,LEAV,8]                ;PICK UP INSTRUCTION FIELD
1178  001264  201040  003220'              MOVEI P,PS
1179  001265  306240  000260                CAIN T,260
1180  001266  254000  001313'              JRST  IPUSHJ                          ;INTERPRET PUSHJ
1181
1182  001267  306240  000264                CAIN T,264
1183  001270  254000  001323'              JRST IJSR                             ;INTERPRET JSR
1184  001271  306240  000265                CAIN T,265
1185  001272  254000  001333'              JRST IJSP                             ;INTERPRET JSP
1186  001273  306240  000266                CAIN T,266
1187  001274  254000  001317'              JRST IJSA                             ;INTERPRET JSA
1188  001275  200100  001303'              MOVE R,LEAV
1189  001276  606240  000700                TRNN T,700
1190  001277  254000  001245'              JRST IXCT4                            ;INTERPRET UUO
1191  001300  306240  000256                CAIN T,256
1192  001301  254000  001247'              JRST IXCT                             ;INTERPRET XCT
1193
1194  001302  265240  002401' IXCT6:  JSP T,RESTORE
1195  001303  000000  000000  LEAV:   0                                           ;INSTRUCTION MODIFIED
1196  001304  254020  001133'              JRST @RCOM
1197  001305  350000  001133'              AOS RCOM
1198  001306  254020  001133'              JRST @RCOM
1199
1200  001307  260040  001353' BPLUP:  PUSHJ P,REMOVB                            ;BREAKPOINT PROCEED ERROR
1201  001310  264000  002360' BPLUP1: JSR SAVE
1202  001311  255000  000000                JFCL
1203  001312  254000  000151'              JRST ERR
```

```
1204  001313  137300  003170' IPUSHJ: DPB W1,[POINT 4,CPUSHP,12]      ;STORE AC FIELD INTO A PUSH
1205  001314  402000  002256'         CLEARM,TEM3
1206  001315  200240  001303'         MOVE T,LEAV
1207  001316  254000  002402'         JRST RESTR1
1208
1209  001317  200240  001133' IJSA:   MOVE T,BCOM              ;INTERPRET JSA
1210  001320  504240  001303'         HRL T,LEAV
1211  001321  250246  002332'         EXCH T,AC0(W1)
1212  001322  254000  001325'         JRST IJSR2
1213
1214  001323  200240  001133' IJSR:   MOVE T,BCOM              ;INTERPRET JSR
1215  001324  500240  002275'         HLL T,SAVPI
1216  001325  200100  001303' IJSR2:  MOVE R,LEAV
1217  001326  260040  001446'         PUSHJ P,DEP
1218  001327  254000  001307'          JRST BPLUP              ;ERROR, CAN'T STORE
1219  001330  354240  001303'         AOSA T,LEAV
1220  001331  200240  001303' IJSR3:  MOVE T,LEAV
1221  001332  254000  002401'         JRST RESTORE
1222

1223  001333  200200  001133' IJSP:   MOVE W,BCOM              ;INTERPRET JSP
1224  001334  500240  002275'         HLL T,SAVPI
1225  001335  202206  002332'         MOVEM W,AC0(W1)
1226  001336  254000  001331'         JRST IJSR3
1227
1228                          ;INSERT BREAKPOINTS
1229
1230  001337  200140  003171' INSRTB: MOVE S,[JSR BP1]
1231  001340  332103  001176  INSRT1: SKIPE R,B1ADR-BP1(S)
1232  001341  260040  001454'         PUSHJ P,FETCH
1233  001342  254000  001347'         JRST INSRT3
1234  001343  202243  000002         MOVEM T,B1INS-BP1(S)
1235  001344  200240  000003         MOVE T,S
1236  001345  260040  001446'         PUSHJ P,DEP
1237  001346  255000  000000          JFCL                    ;HERE ONLY IF CAN'T WRITE IN HIGH SEG
1238  001347  271140  000003  INSRT3: ADDI S,3
1239  001350  317140  003172'         CAMG S,[JSR BPN]
1240  001351  254000  001340'         JRST INSRT1
1241  001352  263040  000000         POPJ P,
1242
1243                          ;REMOVE BREAKPOINTS
1244
1245  001353  201140  002326' REMOVB: MOVEI S,BNADR
1246  001354  200243  776604  REMOV1: MOVE T,B1INS-B1ADR(S)
1247  001355  332103  000000         SKIPE R,(S)
1248  001356  260040  001446'         PUSHJ P,DEP
1249  001357  255000  000000          JFCL                    ;HERE ONLY IF NO WRITE IN HIGH SEG
1250  001360  275140  000003         SUBI S,3
1251  001361  301140  002301'         CAIL S,B1ADR
1252  001362  254000  001354'         JRST REMOV1
1253  001363  254000  002463'         IFN EDDT&1,<JRST TTYRET>
1254                                  IFE EDDT&1,<POPJ P,>
1255                          ;IN EXEC MODE, SAVE UP TTY STATUS       ;IN USER MODE, DONE BY SAVE
```

```
1256                            ;ALL $B COMMANDS GET HERE IN FORM: <A>$<N>B
1257
1258
1259  001364  623000  000001  BPS:    TLZE F,QF               ;HAS <A> BEEN TYPED?
1260  001365  254000  001374'        JRST BPS1               ;YES
1261  001366  622000  000001        TRZE F,Q2F              ;NO, HAS <N> BEEN TYPED?
1262  001367  254000  001400'        JRST BPS2               ;YES
1263  001370  200240  003173'        MOVE T,[XWD B1ADR,B1ADR+1]      ;NO, COMMAND IS $B - CLEAR ALL BREAKPOINTS
1264  001371  402000  002301'        CLEARM B1ADR
1265  001372  251240  002331'        BLT T,AUTOPI            ;CLEAR OUT ALL BREAKPOINTS AND AUTO PROCEDE REGESTER
1266  001373  254000  000164'        JRST RET
1267
1268  001374  626000  000001  BPS1:   TRZN F,Q2F               ;HAS <N> BEEN TYPED?
1269  001375  254000  001412'        JRST BPS3               ;NO
1270  001376  200100  000005        MOVE R,T                ;YES, PROCESS THE COMMAND A$NB
1271  001377  660000  000002        TRO F,2
1272  001400  200240  002250' BPS2:   MOVE T,WRD2
1273  001401  301240  000001        CAIL T,1
1274  001402  303240  000010        CAILE T,NBP
1275  001403  254000  000151'        JRST ERR
1276  001404  221240  000003        IMULI T,3
1277  001405  271240  002276'        ADDI T,B1ADR-3
1278  001406  626000  000002        TRZN F,2
1279  001407  254000  001675'        JRST MASK2
1280  001410  250100  000005        EXCH R,T
1281  001411  254000  001423'        JRST BPS5
1282
1283  001412  201100  002301' BPS3:   MOVEI R,B1ADR           ;PROCESS THE COMMAND A$B
1284  001413  550272  000000  BPS4:   HRRZ W,(R)
1285  001414  302205  000000        CAIE W,(T)
1286  001415  336002  000000        SKIPN (R)
1287  001416  254000  001423'        JRST BPS5
1288  001417  271100  000003        ADDI R,3
1289  001420  307100  002326'        CAIG R,BNADR
1290  001421  254000  001413'        JRST BPS4
1291  001422  254000  000151'        JRST ERR
1292  001423  202242  000000  BPS5:   MOVEM T,(R)
1293  001424  402002  000001        CLEARM,1(R)
1294  001425  402002  000002        CLEARM,2(R)
1295
1296  001426  275100  002301' AUTOP:  SUBI R,B1ADR           ;AUTO PROCEDE SETUP SUBROUTINE
1297  001427  231100  000003        IDIVI R,3
1298  001430  201140  000001        MOVEI S,1
1299  001431  242142  000000        LSH S,(R)
1300  001432  412140  002331'        ANDCAM S,AUTOPI
1301  001433  603000  010000        TLNE F,CCF
1302  001434  436140  002331'        IORM S,AUTOPI
1303  001435  263040  000000        POPJ P,
```

```
1324                            ;FETCH AND DEPOSIT INTO MEMORY
1325
1326
1327  001436  202240  002254' DEPRS:  MOVEM T,LWT              ;DEPOSIT REGISTER AND SAVE AS LWT
1328  001437  200100  002272'         MOVE R,LLOCO             ;QUAN TYPED IN REGIS EXAM
1329  001440  623000  000010         TLZE F,ROF
1330  001441  607000  000001         TLNN F,QF
1331  001442  263040  000000         POPJ P,0
1332                                          ;CAIL R,DDT
1333                                          ;CAILE R,DDTEND-1
1334  001443  260040  001446'         PUSHJ P,DEP              ;STORE AWAY
1335  001444  254000  000151'         JRST ERR                 ;CAN'T STORE (IN DDT OR OUT OF BOUNDS)
1336  001445  263040  000000         POPJ P,                  ;RETURN
1337
1338                            ;DEPOSIT INTO MEMORY SUBROUTINE
1339
1340                            IFE EDDT&10,<
1341                            DEP:    IFE EDDT&1,<
1342                                    JSP TT1,CHKADR           ;LEGAL ADDRESS?

1343                                    JRST DEP4                ;YES BUT IN HI SEGMENT>
1344  001446  606100  777760         TRNN R,777760
1345  001447  254000  001452'         JRST DEPAC               ;DEPOSIT IN AC
1346
1347  001450  202242  000000  IFE EDDT&10,<   MOVEM T,(R) >
1348  001451  254000  000431'         JRST CPOPJ1              ;SKIP RETURN
1349
1350  001452  202242  002332' DEPAC:  MOVEM T,AC0(R)           ;DEPOSIT IN AC
1351  001453  254000  000431'         JRST CPOPJ1              ;SKIP RETURN
1352
1353                            IFE EDDT&1,<
1354                            DEP4:   MOVEI W,0
1355                                    CALLI W,SETUWP           ;IS HI SEGMENT PROTECTED? TURN OFF
1356                                    POPJ P,                  ;PROTECTED, NO SKIP RETURN
1357                                    MOVEM T,(R)              ;STORE WORD IN HI SEGMENT
1358                                    TRNE W,1                 ;WAS WRITE PROTECT ON?
1359                                    CALLI W,SETUWP           ;YES, TURN IT BACK ON
1360                                    JFCL
1361                                    JRST CPOPJ1              ;SKIP RETURN
1362                            >
1363                            >       ;END NOT-FILDDT COND.
1364
1365                            IFN EDDT&10,<DEP=CPOPJ1>         ;ALWAYS OK NO-OP IF FILE DDT
```

```
1346                            ;FETCH FROM MEMORY SUBROUTINE
1347
1348                            FETCH:  IFE EDDT&10,<IFE EDDT&1,
1349                                    <JSP TT1,CHKADR         ;LEGAL ADDRESS?
1350                                    JFCL>                   ;HIGH OR LOW OK FOR FETCH
1351  001454  606100  777760          TRNN R,777760           ;ACCUMULATOR?
1352  001455  334242  002332'         SKIPA T,AC0(R)          ;YES
1353  001456  200242  000000          MOVE T,(R)              ;NO
1354  001457  254000  000431'         JRST CPOPJ1             ;SKIP RETURN ONLY FOR LEGAL ADDRESS>
1355
1356                                    IFN EDDT&10,< ; SKIPN CRASHS            ;CRASH.SAV EXIST?
1357                                    JRST MONPEK             ;NO - GO PEEK AT RUNNING MONITOR
1358                                    MOVEM R,TEM4            ;SAVE THE AOBJN POINTER
1359                                    HRRZ R,R                ;STRIP OFF POSSIBLE COUNT
1360                                    ADD R,OFFSET
1361                                    IDIVI R,4000            ;R HAD LOCATION
1362                                    CAIL R,30               ;R=NO. OF INPTS  R+1=LOCATION
1363                                    POPJ P,                 ;LARGER THAN 48K
1364                                    TRNN S,777000           ;S=R+1
1365                                    JUMPE R,RSDNT           ;LOC IS IN RESIDENT BLOCK
1366                                    CAMN R,RSAVE            ;IS LOC INCORE ?
1367                                    JRST INCORE             ;S=LOC
1368                                    MOVEM R,RSAVE           ;INPT NO.
1369                                    IMULI R,20              ;16/INPT
1370                                    USETI 1,1(R)            ;BLK 0 DOES NOT EXIST
1371                                    INPUT 1,CURLST          ;GET 16 BLKS
1372                                    STATZ 1,740000
1373                                    HALT .-2                ;YEP
1374                            INCORE: SKIPA T,CURENT(S)
1375                            RSDNT:  MOVE T,RSIDNT(S)
1376                                    MOVE R,TEM4             ;RESTORE AOBJN POINTER
1377                                    JRST CPOPJ1
1378                            TEM4:   0                       ;HOLD AOBJN POINTER
1379                            MONPEK: HRRZ T,R                ;REMOVE COUNT
1380                                    CALLI T,33              ;DO PEEK UUO
1381                                    JRST CPOPJ1             ;RETURN VALUE IN AC T
1382                                    >
1383
1384                                    IFE EDDT&1,<    ;DO ADDRESS CHECKS ONLY IN USER MODE
1385                            CHKADR: HRRZ TT,JOBREL          ;GET HIGHEST ADDRESS IN LOW SEGMENT
1386                                    CAIL TT,(R)             ;CHECK FOR WITHIN LOW SEGMENT
1387                                    JRST 1(TT1)             ;ADDRESS IS OK IN LOW SEGMENT, SKIP RETURN
1388                                    HRRZ TT,JOBHRL          ;GET HIGHEST ADDRESS IN HIGH SEGMENT
1389                                    TRNE R,400000           ;IS THE ADDRESS IN HIGH SEGMENT?
1390                                    CAIGE TT,(R)            ;IS THE ADR TOO BIG FOR HIGH SEGMENT?
1391                                    POPJ P,                 ;NO,YES- ILL. ADR.
1392                                    JRST @(TT1)
1393
1394                                    >       ;END OF IFE EDDT&1
1395
1396                    000432'         IFN EDDT&1,<CHKADR==CPOPJ>      ;NO ADDRESS CHECKS IN EXEC MODE
```

```
1397  001460  202240  002267' FIRARG: MOVEM T,DEFV
1398  001461  661000  001000          TLO F,FAF
1399  001462  254000  001465'         JRST ULIM1
1400  001463  661000  002000  ULIM:   TLO F,SAF
1401  001464  552240  002270'         HRREM T,ULIMIT
1402  001465  607000  000001  ULIM1:  TLNN F,QF
1403  001466  254000  000151'         JRST ERR
1404  001467  254000  000033'         JRST LISØ
1405
1406
1407  001470  331100  002260' LOOK:   SKIPL R,PRGM     ;LOOK UP SYMBOL
1408  001471  200120  002274'         MOVE R,@SYMP
1409  001472  574160  002274'         HLRE S,@SYMP
1410  001473  240140  777777          ASH S,-1          ;SETUP COUNT FOR LENGTH OF SYM TABLE
1411  001474  621000  400000          TLZ F,400000
1412  001475  515340  600000          HRLZI W2,DELO+DELI
1413  001476  202240  002356'         MOVEM T,TEM
1414
1415  001477  612342  000000  LOOK1:  TDNE W2,(R)

1416  001500  254000  001522'         JRST LOOK3
1417  001501  200242  000000          MOVE T,(R)
1418  001502  607240  740000          TLNN T,PNAME      ;NAME
1419  001503  665340  100000          TLOA W2,LOCAL
1420  001504  334240  002356'         SKIPA T,TEM
1421  001505  254000  001522'         JRST LOOK3
1422  001506  200202  000001          MOVE W,1(R)
1423  001507  430200  000005          XOR W,T
1424  001510  321200  001522'         JUMPL W,LOOK3
1425  001511  274242  000001          SUB T,1(R)
1426  001512  321240  001522'         JUMPL T,LOOK3
1427  001513  325000  001517'         JUMPGE F,LOOK2
1428  001514  200202  000001          MOVE W,1(R)
1429  001515  274206  000001          SUB W,1(W1)
1430  001516  323200  001522'         JUMPLE W,LOOK3
1431  001517  540330  000002  LOOK2:  HRR W1,R                ;POINTER BEST VALUE SO FAR
1432  001520  661030  400000          TLO F,400000
1433  001521  322240  002102'         JUMPE T,SPTØ
1434  001522  270130  003155' LOOK3:  ADD R,[XWD 2,2]
1435  001523  331030  000002          SKIPL R
1436  001524  200120  002274'         MOVE R, @SYMP
1437  001525  343140  001477'         AOJLE S,LOOK1     ;TERMINATING CONDITION
1438  001526  200240  002356'         MOVE T,TEM
1439  001527  603030  400000          TLNE F,400000
1440  001530  274246  000001          SUB T,1(W1)
1441  001531  254070  000431'         JRST CPOPJ1
```

```
1442  001532  202240  002254' CONSYM: MOVEM T,LWT
1443  001533  606000  002000          TRNN F,LF1
1444  001534  254020  000010          JRST @SCH                ;PIN OR FTOC
1445  001535  602000  000400          TRNE F,CF1
1446  001536  254000  002045'         JRST  FTOC
1447
1448  001537                    PIN:                           ;PRINT INSTRUCTION
1449  001537  641240  700000          TLC T,700000
1450  001540  647240  700000          TLCN T,700000
1451  001541  254000  001647'         JRST INOUT               ;IN-OUT INSTRUCTION OR NEG NUM
1452  001542  404240  003174'         AND T,[XWD 777000,0]     ;EXTRACT OPCODE BITS
1453  001543  322240  001611'         JUMPE T,HLFW             ;TYPE AS HALF WORDS
1454  001544  260040  003032'         PUSHJ P,OPTYPE
1455  001545  205240  777000          MOVSI   T,777000
1456  001546  404240  002254'         AND     T,LWT
1457  001547  606000  000002          TRNN F,ITF               ;HAS INSTRUCTION BEEN TYPED?
1458  001550  260040  001470'         PUSHJ P,LOOK             ;NO, LOOK IN SYMBOL TABLE
1459  001551  664000  000200          TROA F,NAF               ;INSTRUCTION TYPED, ALLOW NEG ADDRESSES
1460  001552  254000  001611'         JRST HLFW                ;NOT FOUND, OUTPUT AS HALFWORDS
1461  001553  260040  002426'         PUSHJ P,TSPC
1462  001554  135240  003175'         LDB T,[XWD 270400,LWT]   ;GET AC FIELD
1463  001555  322240  001561'         JUMPE T,PI4
1464  001556  260040  001620'         PUSHJ P,PAD
1465  001557  201300  000054  PI3A:   MOVEI W1,","
1466  001560  260040  002240'         PUSHJ P,TEXT
1467  001561  200300  002254' PI4:    MOVE W1,LWT
1468  001562  201240  000100          MOVEI T,"@"
1469  001563  603300  000020          TLNE W1,20               ;CHECK FOR INDIRECT BIT
1470  001564  260040  002440'         PUSHJ P,TOUT
1471  001565  550240  002254'         HRRZ T,LWT
1472  001566  135200  003176'         LDB W,[XWD 331100,LWT]   ;INSTRUCTION BITS
1473  001567  301200  000240          CAIL W,240
1474  001570  303200  000247          CAILE W,247
1475  001571  254000  001607'         JRST PI8                 ;ALL (EXCEPT ASH,ROT,LSH) HAVE SYMBOLIC ADRS
1476  001572  607300  000020          TLNN W1,20
1477  001573  306200  000243          CAIN W,<JFFO>+-33
1478  001574  254000  001607'         JRST PI8                 ;JFFO AND @ GET SYMBOLIC ADDRESSES
1479  001575  260040  001641'         PUSHJ P,PADS3A  ;ONLY ABSOLUTE ADDRESSING FOR LSH, ASH, AND ROT
1480  001576  620000  000200  PI7:    TRZ F,NAF
1481  001577  135100  003177'         LDB R,[XWD 220400,LWT]   ;INDEX REGISTER CHECK
1482  001600  322100  001624'         JUMPE R,PADS1            ;EXIT
1483  001601  201240  000050          MOVEI T,"("
1484  001602  260040  002440'         PUSHJ P,TOUT
1485  001603  200240  000002          MOVE T,R
1486  001604  260040  001620'         PUSHJ P,PAD
1487  001605  201240  000051          MOVEI T,")"
1488  001606  254000  002440'         JRST TOUT                ;EXIT
1489
1490  001607  260040  001620' PI8:    PUSHJ P,PAD
1491  001610  254000  001576'         JRST PI7
```

```
1492                            HLFW:   REPEAT 0,<      MOVE T,LWT
1493                                    CAML T,[DDTINT SAVPI]
1494                                    CAMLE T,[DDTINT BNADR+2]
1495                                    SKIPA
1496                                    JRST PAD>
1497  001611  554240  002254'           HLRZ T,LWT              ;PRINT AS HALF WORDS
1498  001612  322240  001617'           JUMPE T,HLFW1           ;TYPE ONLY RIGHT ADR IF LEFT ADR=0
1499  001613  660000  000200            TRO F,NAF               ;ALLOW NEGATIVE ADDRESSES
1500  001614  260040  001620'           PUSHJ P,PAD
1501  001615  205300  261300            MOVSI W1,(ASCII /,,/)
1502  001616  260040  002242'           PUSHJ P,TEXT2           ;TYPE ,,
1503  001617  550240  002254' HLFW1:    HRRZ T,LWT
1504
1505                                    ;PRINT ADDRESSES (ARG USUALLY 18 BITS BUT CAN BE 36 BITS)
1506
1507  001620  405240  777777  PAD:     ANDI T,-1
1508  001621  254020  000011            JRST @AR                ;PADS0 OR PAD1
1509  001622  322240  002213' PADS0:   JUMPE T,FP7B            ;PRINT A ZERO
1510  001623  260040  001470'           PUSHJ P,LOOK

1511  001624  263040  000000  PADS1:   POPJ P,0
1512  001625  200346  000001            MOVE W2,1(W1)
1513  001626  305240  000100            CAIGE T,100
1514  001627  305340  000060            CAIGE W2,60
1515  001630  254070  001640'           JRST PADS3
1516  001631  202240  002356'           MOVEM T,TEM
1517  001632  325000  001637'           JUMPGE F,PAD1
1518  001633  260040  002102'           PUSHJ P,SPT0
1519  001634  201240  000053            MOVEI T,"+"
1520  001635  260040  002440' PADS1A:   PUSHJ P,TOUT
1521  001636  550240  002356'           HRRZ T,TEM
1522  001637  254000  002045' PAD1:    JRST TOC                ;EXIT
1523
1524  001640  200240  002356' PADS3:   MOVE T,TEM
1525  001641  602000  000200  PADS3A:  TRNE F,NAF
1526  001642  305240  776000            CAIGE T,776000
1527  001643  254000  002045'           JRST TOC
1528  001644  212240  002356' PADS3B:  MOVNM T,TEM
1529  001645  201240  000055            MOVEI T,"-"
1530  001646  254000  001635'           JRST PADS1A
```

```
1531  001647  650240  003200' INOUT:  TDC T,[XWD -1,400000]   ;IO INSTRUCTION OR NEG NUM
1532  001650  656240  003200'         TDCN T,[XWD -1,400000]
1533  001651  254000  001644'         JRST PADS3B             ;TYPE AS NEG NUM
1534  001652  135100  003201'         LDB R,[POINT 7,T,9]     ;PICK OUT IO DEVICE BITS
1535  001653  301100  000160          CAIL R,700--2           ;IF DEVICE .L. 700, THEN TYPE
1536  001654  254000  001611'         JRST HLFW               ;TYPE AS HALF WORDS
1537  001655  135100  003202'         LDB R,[POINT 3,T,12]
1538  001656  137100  003203'         DPB R,[POINT 6,T,8]     ;MOVE IO BITS OVER FOR OP DECODER
1539  001657  260040  003032'         PUSHJ P,OPTYPE
1540  001660  260040  002426'         PUSHJ P,TSPC
1541  001661  205240  077400          MOVSI T,077400
1542  001662  404240  002254'         AND T,LWT
1543  001663  322240  001561'         JUMPE T,PI4
1544  001664  260040  001470'         PUSHJ P,LOOK            ;LOOK FOR DEVICE NUMBER
1545  001665  254000  001557'         JRST PI3A
1546  001666  200240  002356'         MOVE T,TEM
1547  001667  242240  777750          LSH T,-30
1548  001670  260040  002045'         PUSHJ P,TOC
1549  001671  254000  001557'         JRST PI3A
1550  001672  603000  000001  MASK:   TLNE F,QF
1551  001673  254000  001700'         JRST MASK1
1552  001674  201240  002300'         MOVEI T,MSK
1553  001675  201200  000001  MASK2:  MOVEI W,1
1554  001676  202200  002256'         MOVEM W,FRASE1
1555  001677  254000  000374'         JRST QUANIN
1556  001700  202240  002300' MASK1:  MOVEM T,MSK
1557  001701  254000  000164'         JRST RET
```

```
1558  001702  661000  000020  EFFEC:  TLO F,LTF
1559  001703  550240  000005          HRRZ T,T
1560  001704  201100  774000  WORD:   MOVEI R,322000-326000   ;JUMPE-JUMPN
1561  001705  271100  326240  NWORD:  ADDI R,326000+40*T      ;JUMPN T,
1562  001706  506100  001730'        HRLM R,SEAR2
1563  001707  627000  000001        TLZN F,QF
1564  001710  254000  000151'        JRST ERR
1565  001711  452240  002247'        SETCAM T,WRD
1566  001712  205240  777772        MOVSI T,FRASE-DEN-1            ;PREVENT TYPE OUT OF DD" PARTS
1567  001713  462005  002252'        SETCMM FRASE(T)
1568  001714  253240  001713'        AOBJN T,.-1
1569  001715  200240  002270'        MOVE T,ULIMIT
1570  001716  603000  002000        TLNE F,SAF
1571  001717  661000  000001        TLO F,QF                       ;SIMULATE A $Q TYPED
1572  001720  260040  002000'        PUSHJ P,SETUP
1573  001721  260040  002421'        PUSHJ P,CRF
1574  001722  260040  001454' SEAR1:  PUSHJ P,FETCH
1575  001723  254000  001736'        JRST SEAR2B
1576  001724  603000  000020        TLNE F,LTF      ;CHECK FOR EFFECTIVE ADDRESS SEARCH

1577  001725  254000  001756'        JRST EFFEC0
1578  001726  444240  002247'        EQV T,WRD
1579  001727  404240  002300'        AND T,MSK
1580  001730  322240  001744' SEAR2:  JUMPE T,SEAR3           ;OR JUMPN T
1581  001731  350100  002267' SEAR2A: AOS R,DEFV       ;GET NEXT LOCATION
1582  001732  260040  002456'        PUSHJ P,LISTEN   ;ANYTHING TYPED?
1583  001733  313100  002270'        CAMLE R,ULIMIT   ;OR END OF SEARCH?
1584  001734  254000  001742'        JRST SEARFN      ;YES
1585  001735  254000  001722'        JRST SEAR1       ;NO, LOOK SOME MORE
1586
1587  001736  201100  377777 SEAR2B: MOVEI R,400000-1         ;MOVE UP TO HI SEGMENT
1588  001737  437100  002267'        IORB R,DEFV              ;PUT IN MEMORY TOO
1589  001740  606100  400000        TRNN R,400000            ;ALREADY IN HI SEGMENT?
1590  001741  254000  001731'        JRST SEAR2A              ;NO
1591  001742  462000  002254' SEARFN: SETCMM LWT              ;COMPLEMENT BITS BACK AND STOP SEARCH
1592  001743  254000  000022'        JRST DD1
```

```
1593  001744  200100  002267' SEAR3:  MOVE R,DEFV
1594  001745  260040  001454'         PUSHJ P,FETCH
1595  001746  254000  000151'         JRST ERR
1596  001747  621000  004000          TLZ F,STF          ;GET RID OF SUPPRESS TYPEOUT MODE
1597  001750  200240  002267'         MOVE T,DEFV
1598  001751  260040  000711'         PUSHJ P,LI1        ;CALL REGISTER EXAMINATION LOGIC TO TYPE OUT
1599  001752  260040  002421'         PUSHJ P,CRF
1600  001753  462000  002254'         SETCMM LWT
1601  001754  462000  002356'         SETCMM TEM
1602  001755  254000  001731' SEAR4:  JRST  SEAR2A
1603
1604  001756  201200  000100  EFFEC0: MOVEI W,100
1605  001757  202200  002356'         MOVEM W,TEM
1606  001760  200200  000005  EFFEC1: MOVE W,T
1607  001761  135100  003204'         LDB R,[POINT 4,T,17]    ;GET IR FIELD
1608  001762  322100  001766'         JUMPE R,EFFEC2
1609  001763  260040  001454'         PUSHJ P,FETCH
1610  001764  254000  000151'         JRST ERR
1611  001765  270240  000004          ADD T,W
1612  001766  540100  000005  EFFEC2: HRR R,T
1613  001767  607200  000020          TLNN W,20               ;INDIRECT BIT CHECK
1614  001770  254000  001775'         JRST EFFEC3
1615  001771  372000  002356'         SOSE,TEM
1616  001772  260040  001454'         PUSHJ P,FETCH
1617  001773  254000  001755'         JRST SEAR4
1618  001774  254000  001760'         JRST EFFEC1
1619  001775  444240  002247' EFFEC3: EQV T,WRD
1620  001776  405240  777777          ANDI T,777777
1621  001777  254000  001730'         JRST SEAR2
```

```
1622  002000  607000  000001  SETUP:  TLNN F,QF               ;QUANTITY TYPED?
1623
1624                          IFN EDDT&10,<MOVEI T,137777      ;DON'T SEARCH BEYOND 48K>
1625                          IFE EDDT&10,<
1626                          IFE EDDT&1,<MOVEI T,777777       ;NO, GET LAST ADR>
1627  002001  550260  002274' IFN EDDT&1,<HRRZ T,@SYMP          ;NO, GET 1ST ADR SYM TABLE>
1628                          >
1629
1630  002002  552240  002270'         HRRZM T,ULIMIT           ;SAVE LAST ADDRESS OF SEARCH
1631  002003  553100  002267'         HRRZS R,DEFV             ;GET 1ST ADDRESS
1632  002004  607000  001000          TLNN F,FAF               ;WAS A 1ST ADR SPECIFIED?
1633  002005  403100  002267'         SETZB R,DEFV             ;NO, MAKE IT ZERO
1634  002006  313100  002270'         CAMLE R,ULIMIT           ;LIMITS IN A REASONABLE ORDER?
1635  002007  254000  000151'         JRST ERR                 ;NO
1636  002010  263040  000000          POPJ P,          ;YES, RETURN
1637
1638  002011  607000  010000  ZERO:   TLNN F,CCF
1639  002012  254000  000151'         JRST ERR
1640  002013  260040  002000'         PUSHJ P,SETUP

1641  002014  550160  002274'         HRRZ S,@SYMP     ;GET 1ST ADR OF SYMBOL TABLE
1642  002015  574320  002274'         HLRE W1,@SYMP    ;GET LENGTH OF SYM TABLE
1643  002016  274300  000003          SUB W1,S         ;GET NEG OF LAST ADR
1644  002017  213000  000006          MOVNS W1         ;GET POS LAST ADR
1645  002020  201240  000000          MOVEI T,0        ;0 TO STORE IN MEMORY
1646  002021  606100  777760  ZERO1:  TRNN R,777760
1647  002022  254000  002035'         JRST ZEROR       ;OK TO ZERO AC'S
1648  002023  305100  000040          CAIGE R,ZLOW
1649  002024  201100  000040          MOVEI R,ZLOW     ;DON'T ZERO 20 THRU ZLOW
1650  002025  301100  000000'         CAIL R,DDT
1651  002026  303100  003270'         CAILE R,DDTEND
1652  002027  254000  002031'         JRST .+2
1653  002030  201100  003270'         MOVEI R,DDTEND   ;DON'T ZERO DDT
1654  002031  311100  000003          CAML R,S
1655  002032  313100  000006          CAMLE R,W1
1656  002033  254000  002035'         JRST .+2
1657  002034  550100  000006          HRRZ R,W1        ;DON'T ZERO SYMBOL TABLE
1658  002035  313100  002270'  ZEROR:  CAMLE R,ULIMIT   ;ABOVE LIMITS?
1659  002036  254000  000022'         JRST DD1         ;YES, STOP
1660  002037  260040  001446'         PUSHJ P,DEP      ;DEPOSIT T
1661  002040  664100  377777          TROA R,377777    ;
1662  002041  344100  002021'         AOJA R,ZERO1
1663  002042  606100  400000          TRNN R,400000    ;HI SEGMENT?
1664  002043  344100  002021'         AOJA R,ZERO1     ;NO, KEEP GOING
1665  002044  254000  000022'         JRST DD1         ;FINISH
```

```
1666  002045                        FTOC:           ;NUMERIC OUTPUT SUBROUTINE
1667  002045  550300  000012  TOC:   HRRZ W1,ODF
1668  002046  306300  000010        CAIN W1,10              ;IS OUPUT RADIX NOT OCTAL, OR
1669  002047  607240  777777        TLNN T,-1               ;ARE THERE  NO LEFT HALF BITS?
1670  002050  254000  002057'       JRST TOCA               ;YES, DO NOTHING SPECIAL
1671  002051  542240  002056'       HRRM T,TOCS             ;NO, TYPE AS HALF WORD CONSTANT
1672  002052  557000  000005        HLRZS T                 ;GET LEFT HALF
1673  002053  260040  002062'       PUSHJ P,TOC0            ;TYPE LEFT HALF
1674  002054  205300  261300        MOVSI W1,(ASCII /,,/)
1675  002055  260040  002242'       PUSHJ P,TEXT2           ;TYPE ,,
1676  002056  201240  000000  TOCS:  MOVEI T,.-.            ;GET RIGHT HALF BACK
1677  002057  550300  000012  TOCA:  HRRZ W1,ODF             ;IS OUTPUT RADIX DECIMAL?
1678  002060  306300  000012        CAIN W1,12
1679  002061  254000  002073'       JRST TOC4               ;YES,TYPE SIGNED WITH PERIOD
1680  002062  246240  777735  TOC0:  LSHC T,-43
1681  002063  242300  777777        LSH W1,-1               ;W1=T+1
1682  002064  235260  000012        DIVI T,@ODF
1683  002065  506301  000000        HRLM W1,0(P)
1684  002066  332000  000005        SKIPE T
1685  002067  260040  002062'       PUSHJ P,TOC0
1686  002070  554241  000000        HLRZ T,0(P)
1687  002071  271240  000060        ADDI T,"0"
1688  002072  254000  002440'       JRST TOUT
1689
1690  002073  214100  000005  TOC4:  MOVM A,T                ;TYPE AS SIGNED DECIMAL INTEGER
1691  002074  325240  002077'       JUMPGE T,TOC5
1692  002075  201240  000055        MOVEI T,"-"
1693  002076  260040  002440'       PUSHJ P,TOUT
1694  002077  260040  002205'  TOC5: PUSHJ P,FP7             ;DECIMAL PRINT ROUTINE
1695  002100  201240  000056  TOC6: MOVEI T,"."
1696  002101  254000  002440'       JRST TOUT
1697
1698                                ;SYMBOL OUTPUT SUBROUTINE
1699
1700  002102  552300  002266'  SPT0:  HRRZM W1,SPSAV          ;SAVE POINTER TO TYPED SYM
1701  002103                     SPT:          ;RADIX 50 SYMBOL PRINT
1702  002103  135240  003205'       LDB T,[POINT 32,0(W1),35]       ;GET SYMBOL
1703  002104  231240  000050  SPT1:  IDIVI T,50
1704  002105  506301  000000        HRLM W1,0(P)
1705  002106  322240  002110'       JUMPE T,SPT2
1706  002107  260040  002104'       PUSHJ P,SPT1
1707  002110  554241  000000  SPT2:  HLRZ T,0(P)
1708  002111  322240  000432'       JUMPE T,CPOPJ           ;FLUSH NULL CHARACTERS
1709  002112  271240  000257        ADDI T,260-1
1710  002113  303240  000271        CAILE T,271
1711  002114  271240  000007        ADDI T,301-272
1712  002115  303240  000332        CAILE T,332
1713  002116  275240  000070        SUBI T,334-244
1714  002117  306240  000243        CAIN T,243
1715  002120  201240  000256        MOVEI T,256
1716  002121  254000  002440'       JRST TOUT
```

```
1717  002122  201240  000002  SYMD:   MOVEI T,DELO/200000      ;$D ;DELETE LAST SYM & PRINT NEW
1718  002123  550100  002266'         HRRZ R,SPSAV             ;PICK UP POINTER TO LAST SYM
1719  002124  322100  000151'         JUMPE R,ERR
1720  002125  137240  003160'         DPB T,[POINT 2,(R),11    ;STORE SEMI-DELETE BITS IN SYMBOL
1721  002126  200240  002254'         MOVE T,LWT
1722  002127  254000  001532'         JRST CONSYM              ;PRINT OUT NEXT BEST SYMBOL
```

```
1723                            ;FLOATING POINT OUTPUT
1724
1725  002130  200100  000005  TFLOT:  MOVE A,T
1726  002131  327100  002140'         JUMPG A, TFLOT1
1727  002132  322100  002147'         JUMPE A,FP1A
1728  002133  213000  000002         MOVNS A
1729  002134  201240  000055         MOVEI T,"-"
1730  002135  260040  002440'         PUSHJ P,TOUT
1731  002136  623100  400000         TLZE A,400000
1732  002137  254000  002147'         JRST FP1A
1733  002140  607100  000400  TFLOT1: TLNN A, 400
1734  002141  254000  002077'         JRST TOC5               ;IF UNNORMALIZED, TYPE AS DECIMAL INTEGER
1735
1736  002142  201140  000000  FP1:    MOVEI B,0
1737  002143  315100  002231'         CAMGE A,FT01
1738  002144  254000  002167'         JRST FP4
1739  002145  311100  002217'         CAML A,FT8
1740  002146  344140  002167'         AOJA B,FP4
1741  002147  201200  000000  FP1A:   MOVEI C,0
1742
1743  002150  225100  000400  FP3:    MULI A,400
1744  002151  244142  777535         ASHC B,-243(A)
1745  002152  402000  002357'         SETZM TEM1          ;INIT 8 DIGIT COUNTER
1746  002153  332100  000003         SKIPE A,B           ;DON'T TYPE A LEADING 0
1747  002154  260040  002205'         PUSHJ P,FP7         ;PRINT INTEGER PART OF 8 DIGITS
1748  002155  260040  002100'         PUSHJ P,TOC6                ;PRINT DECIMAL POINT
1749  002156  211100  000010         MOVNI A,10
1750  002157  270100  002357'         ADD A,TEM1
1751  002160  200340  000004         MOVE W1,C
1752  002161  200240  000006  FP3A:   MOVE T,W1
1753  002162  225240  000012         MULI T,12
1754  002163  260040  002213'         PUSHJ P,FP7B
1755  002164  332000  000006         SKIPE,W1
1756  002165  341170  002161'         AOJL A,FP3A
1757  002166  263040  000000         POPJ P,
1758
1759  002167  211200  000006  FP4:    MOVNI C,6
1760  002170  201340  000000         MOVEI W2,0
1761  002171  240340  000001  FP4A:   ASH W2,1
1762  002172  256023  002232'         XCT,FCP(B)
1763  002173  254000  002176'         JRST FP4B
1764  002174  164123  002233'         FMPR A,@FCP+1(B)
1765  002175  435340  000001         IORI W2,1
1766  002176  346200  002171' FP4B:   AOJN C,FP4A
1767  002177  261040  000007         PUSH P,W2           ;SAVE EXPONENT
1768  002200  261043  002235'         PUSH P,FSGN(B)      ;SAVE "E+" OR "E-"
1769  002201  260040  002150'         PUSHJ P,FP3         ;PRINT OUT FFF.FFF PART OF NUMBER
1770  002202  262040  000006         POP P,W1            ;GET "E+" OR "E-" BACK
1771  002203  260040  002240'         PUSHJ P,TEXT
1772  002204  262040  000002         POP P,A             ;GET EXPONENT BACK
```

```
1773  002205  231100  000012  FP7:    IDIVI A,12              ;DECIMAL OUTPUT SUBROUTINE
1774  002206  350000  002357'        AOS,TEM1
1775  002207  506141  000000        HRLM B,(P)
1776  002210  322100  002212'        JUMPE A,FP7A1
1777  002211  260040  002205'        PUSHJ P,FP7
1778
1779  002212  554241  000000  FP7A1:  HLRZ T,(P)
1780  002213  271240  000260  FP7B:   ADDI T,260
1781  002214  254000  002440'        JRST TOUT
1782
1783  002215  353473  426555          353473426555    ;1.0E32
1784  002216  266434  157116          266434157116    ;1.0E16
1785  002217  233575  360400  FT8:    233575360400    ;1.0E8
1786  002220  216470  400000          216470400000    ;1.0E4
1787  002221  207620  000000          207620000000    ;1.0E2
1788  002222  204500  000000          204500000000    ;1.0E1
1789  002223  201400  000000  FT:     201400000000    ;1.0E0
1790  002224  026637  304365          026637304365    ;1.0E-32
1791  002225  113715  126246          113715126246    ;1.0E-16

1792  002226  146527  461671          146527461671    ;1.0E-8
1793  002227  163643  334273          163643334273    ;1.0E-4
1794  002230  172507  534122          172507534122    ;1.0E-2
1795  002231  175631  463146  FT01:   175631463146    ;1.0E-1
1796                  002232' FT0=FT01+1
1797
1798  002232  313104  002232' FCP:    CAMLE A, FT0(C)
1799  002233  315104  002223'         CAMGE A, FT(C)
1800  002234  000004  002232'         Z FT0(C)
1801
1802  002235  425320  000000  FSGN:   ASCII .E-.
1803  002236  425260  000000          ASCII .E+.
1804
1805  002237  200300  000005  TEXTT:  MOVE W1,T
1806  002240  607300  774000  TEXT:   TLNN W1,774000          ;LEFT JUSTIFIED UNLESS LEFT CHAR IS NULL
1807  002241  242300  000035          LSH W1,35
1808  002242  201240  000000  TEXT2:  MOVEI T,0               ;7 BIT ASCII TEXT OUTPUT SUBROUTINE
1809  002243  246240  000007          LSHC T,7
1810  002244  260040  002440'        PUSHJ P,TOUT
1811  002245  326300  002242'        JUMPN W1,TEXT2
1812  002246  263040  000000          POPJ P,
```

```
1813  002247  000000  000000  WRD:    0
1814  002250  000000  000000  WRD2:   0
1815  002251  000000  000000  PRNC:   0
1816
1817  002252  000000  000000  FRASE:  0       ;DONT CHANGE ORDER, SEE  SEARC+3
1818  002253  000000  000000  SYL:    0
1819  002254  000000  000000  LWT:    0
1820  002255  000000  000000  TEM2:   0
1821  002256                  FRASE1:
1822  002256  000000  000000  TEM3:   0
1823  002257  000000  000000  DEN:    0
1824
1825  002260  000000  000000  PRGM:   0
1826  002261  000000  000000  ESTU:   0
1827  002262  000000  000000  ESTUT:  0
1828  002263  000000  000000  FSV:    0
1829  002264  000000  000000  FH:     0
1830  002265  000000  000000  SYM:    0
1831  002266  000000  000000  SPSAV:  0       ;POINTER TO LAST SYMBOL TYPED
1832  002267  000000  000000  DEFV:   0
1833  002270  000000  000000  ULIMIT: 0
1834  002271  000000  000000  LLOC:   0
1835  002272  000000  000000  LLOCO:  0
1836  002273  000000  000000  SAVLOC: 0               ;THE ADR OF OLD REGISTER EXAMINATION SEQUENCE
1837  002274  000020  000036  SYMP:   XWD PID,JOBSYM
```

```
1838  002275  000000  000000  SAVPI:  0
1839  002276  000000  001177          1177
1840  002277  000000  000000  SAVTTY: 0
1841  002300  777777  777777  MSK:    XWD -1,-1
1842  002301  000000  000000  B1ADR:  0
1843  002302  000000  000000  B1SKP:  0
1844  002303  000000  000000  B1CNT:  0
1845
1846                          REPEAT NRP*3-3, <       0>
1847  002304  000000  000000          0
1848  002305  000000  000000          0
1849  002306  000000  000000          0
1850  002307  000000  000000          0
1851  002310  000000  000000          0
1852  002311  000000  000000          0
1853  002312  000000  000000          0
1854  002313  000000  000000          0
1855  002314  000000  000000          0
1856  002315  000000  000000          0

1857  002316  000000  000000          0
1858  002317  000000  000000          0
1859  002320  000000  000000          0
1860  002321  000000  000000          0
1861  002322  000000  000000          0
1862  002323  000000  000000          0
1863  002324  000000  000000          0
1864  002325  000000  000000          0
1865  002326  000000  000000          0
1866  002327  000000  000000          0
1867  002330  000000  000000          0
1868
1869                  002326' BNADR=.-3
1870  002331  000000  000000  AUTOPI: 0
1871
1872                          IFE EDDT&10,<
1873  002332                  AC0:    BLOCK 17
1874
1875  002351  000000  000000  AC17:   0
1876                          >
1877
1878  002352  000000  001537' SCHM:   EXP PIN          ;DO NOT CHANGE ORDER
1879  002353  000000  001622' ARM:    EXP PADSO
1880  002354  000200  000010  ODFM:   EXP 10
1881
1882  002355  000000  000000  SARS:   0
1883  002356  000000  000000  TEM:    0
1884  002357  000000  000000  TEM1:   0
1885                          IFN EDDT&10,< OFFSET:   0 >
```

```
1886                            IFN EDDT&1&<EDDT>B36,<
1887
1888
1889                            PUNCH:  TLC F,FAF+QF
1890                                    TLCE F,FAF+QF
1891                                    JRST ERR                ;ONE ARGUMENT MISSING
1892                            PUN2:   ADDI T,1
1893                                    HRRZM T,TEM1
1894                                    SUB T,DEFV
1895                                    JUMPLE T,ERR
1896
1897                            PUN1:   MOVEI T,4               ;PUNCH 4 FEED HOLES
1898                                    PUSHJ P,FEED
1899                                    TLNE F,CF               ;PUNCH NON-ZERO BLOCKS?
1900                                    JRST PUNZ               ;YES
1901                                    HRRZ R,DEFV
1902                                    IORI R,37
1903                                    ADDI R,1
1904                                    CAMLE R,TEM1
1905                                    MOVE R,TEM1
1906                                    EXCH R,DEFV
1907                                    MOVE T,R
1908                                    SUB T,DEFV
1909                                    HRL R,T
1910                                    JUMPGE R,RET            ;EXIT OR PUNCH
1911
1912                            PBLK:   MOVE T,R
1913                                    SOS W,T         ;INIT CHECKSUM
1914                                    PUSHJ P,PWRD
1915                            PBLK1:  PUSHJ P,FETCH
1916                                    JRST ERR
1917                                    ADD W,T
1918                                    PUSHJ P,PWRD
1919                                    AOBJN R,PBLK1
1920                                    MOVE T,W
1921                                    PUSHJ P,PWRD
1922                                    JRST PUN1
1923
1924
1925                            ;PUNCH NON-ZERO BLOCKS
1926
1927                            PUNZ0:  AOS DEFV                ;LOOK AT NEXT WORD
1928                            PUNZ:   HRRZ W,DEFV             ;ENTER HERE - GET STARTING ADDRESS
1929                                    MOVE R,W
1930                                    SUB W,TEM1              ;CALCULATE NEGATIVE LENGTH
1931                                    HRL R,W                 ;SET UP AOBJN POINTER
1932                                    JUMPGE R,RET            ;FINISHED?
1933                                    CAMG R,[XWD -40,0]      ;BLOCK LONGER THAN 40?
1934                                    HRLI R,-40              ;YES, FIX IT UP
1935                                    MOVSI W1,400000         ;W1 NEGATIVE MEANS FLUSH 0 WORDS
1936                            PUNZ2:  PUSHJ P,FETCH           ;GET WORD FROM MEMORY
1937                                    JRST ERR
1938                                    JUMPE T,[AOJA W1,PUNZ4] ;IF WORD IS 0, INDEX 0 WORD COUNTER
```

```
1939                            MOVEI W1,0              ;CLEAR 0 WORD COUNTER
1940                    PUNZ4:  JUMPL W1,PUNZ0          ;FLUSH 0 WORD, GET ANOTHER
1941                            CAIL W1,3               ; NOSKIP FOR 3RD 0 WORD AFTER NON 0 WORD
1942                            AOSA R                  ;ADVANCE R TO LAST ADR+1
1943                            AOBJN R,PUNZ2
1944                            ADD W1,DEFV             ;CALCULATE DEFV-R+W1=-WORD COUNT
1945                            SUB W1,R
1946                            HRLM W1,DEFV            ;PUT -WC IN LEFT HALF OF FA
1947                            EXCH R,DEFV             ;SAVE ADR FOR NEXT BLOCK, GET POINTER
1948                            JRST PRLK
1949
1950
1951
1952
1953                    LOADER: TLNE F,QF
1954                            JRST ERR
1955                            MOVEI T,400
1956                            PUSHJ P,FEED
1957                            MOVE R,LOADE

1958                    LOAD1:  MOVE T,0(R)
1959                            PUSHJ P,PWRD
1960                            AOBJN R,LOAD1
1961                            MOVEI T,20
1962                    LOAD2:  PUSHJ P,FEED
1963                            JRST RET
1964
1965                    BLKEND: TLNN F,QF               ;BLOCK END
1966                            MOVE T,[JRST 4,DDT]
1967                            TLNN T,777000           ;INSERT JRST IF NO OPCODE
1968                            TLO T,(JRST)
1969                            PUSH P,T
1970                            MOVEI T,20
1971                            PUSHJ P,FEED
1972                            POP P,T
1973                            PUSHJ P,PWRD
1974                            PUSHJ P,PWRD    ;EXTRA WORD FOR READER TO STOP ON
1975                            MOVEI T,400
1976                            JRST LOAD2
1977
1978                    PWRD:   MOVEI W1,6
1979                    PWRD2:  ROT T,6
1980                            CONSZ PTPP,20
1981                            JRST .-1
1982                            CONO PTPP,52
1983                            DATAO PTPP,T
1984                            SOJG W1,PWRD2
1985                            POPJ P,0
1986
1987                    FEED:   CONSZ PTPP,20
1988                            JRST .-1
1989                            CONO PTPP,10
1990                            DATAO PTPP,FEED1
1991                            SOJN T,FEED
```

```
1992                            FEED1:  POPJ P,0        ;ADDRESS USED AS A CONSTANT
1993
1994
1995
1996
1997                            LOADR:
1998
1999                            PHASE 0                 ;RIM10B CHECKSUM LOADER
2000                                    XWD -16,0
2001                            BEG:    CONO PTRR,60
2002                                    HRRI AA,RD+1
2003                            RD:     CONSO PTRR,10
2004                                    JRST .-1
2005                                    DATAI PTRR,@TBL1-RD+1(AA)
2006                                    XCT     TBL1-RD+1(AA)
2007                                    XCT     TBL2-RD+1(AA)
2008                            AA:     SOJA AA,
2009
2010                            TBL1:   CAME CKSM,ADR
2011                                    ADD CKSM,1(ADR)
2012                                    SKIPL CKSM,ADR
2013
2014                            TBL2:   JRST 4,BEG
2015                                    AOBJN ADR,RD
2016                            ADR:    JRST BEG+1
2017                            CKSM=ADR+1
2018
2019                            DEPHASE
2020
2021                            LOADE:  XWD LOADB-.,LOADR
2022                            >       ;END OF IFN EDDT$1&<EDDT>B36
2023                                    ;FOR PAPER TAPE IO
2024
```

```
2025  002360  000000  000000  SAVE:   ?                ;SAVE THE ACS AND PI SYSTEM
2026  002361  336000  002355'         SKIPN SARS
2027  002362  254000  002365'         JRST SAV1
2028  002363  350000  002360'         AOS SAVE
2029  002364  254000  002374'         JRST SAV5
2030  002365                  SAV1:   IFN EDDT&1,<
2031  002365  700640  002275'         CONI PRS,SAVPI
2032  002366  700620  002276'         CONO PRS, @SAVPI+1>
2033  002367  202740  002351'         MOVEM 17,AC17
2034  002370  551740  002332'         HRRZI 17,AC0
2035  002371  251740  002350'         RLT 17,AC0+16
2036  002372  200240  002360'         MOVE T, SAVE
2037  002373  502240  002275'         HLLM T, SAVPI
2038  002374                  SAV5:
2039                          ;IFE EDDT&1,<HRRZS SAVUWP               ;ASSUME UWP WILL BE ZEROED
2040                          ;        SETZB T,F              ;T=F=0
2041                          ;        CALLI T,SETUWP                 ;REQUEST TO CLEAR UWP
2042                          ;        HRROS SAVUWP                   ;FAILED, UWP (BIT 0) STILL 1
2043                          ;        SKIPN SARS                     ;USER'S STATE SAVED ALREADY?

2044                          ;        HRRM T,SAVUWP>                 ;NO, SAVE STATE OF UWP
2045
2046  002374  476000  002355'         SETOM SARS                     ;FLAG PROTECTING SAVED REGISTERS
2047  002375  201040  003220'         MOVEI P,PS
2048                                   IFE EDDT&1,<PUSHJ P,TTYRET>    ;IN USER MODE, GET INTO DDT MODE
2049  002376  200240  003147'         MOVE T,[XWD SCHM,SCH]
2050  002377  251240  000012          RLT T,ODF              ;LOAD THE ACS WITH MODE SWITCHES
2051  002400  254020  002360'         JRST @SAVE
2052
2053  002401  476000  002256' RESTORE: SETOM TEM3     ;RESTORE ACS AND PI SYSTEM
2054  002402  542240  002360' RESTR1: HRRM T,SAVE
2055  002403  200240  002275'         MOVE T,SAVPI
2056  002404  502240  002360'         HLLM T, SAVE
2057                          IFN EDDT&1,<
2058  002405  404240  002276'         AND T, SAVPI+1
2059  002406  435240  002000          IORI T, 2000    ;TURN ON CHANNELS
2060  002407  202240  002275'         MOVEM T, SAVPI>
2061                          ;IFE EDDT&1,<HRRZ T,SAVUWP              ;GET OLD UWP
2062                          ;        CALLI T,SETUWP         ;CHANGE IT TO OLD
2063                          ;        JFCL>                  ;ES MACHT NICHTS
2064  002410  515740  002332'         HRLZI 17,AC0
2065  002411  251740  000017          RLT 17,17
2066  002412  402000  002355'         SETZM SARS
2067  002413  331000  002256'         SKIPL,TEM3
2068  002414  261000  001133' CPUSHP: PUSH BCOM       ;PROGRAM MODIFIED AT IPUSHJ
2069  002415  700620  002275' IFN EDDT&1,<    CONO PRS,@SAVPI>
2070  002416  254120  002360'         JRST 2,@SAVE
```

```
2071  002417  201240  000015  CRN:    MOVEI T,15              ;CARRIAGE RETURN
2072  002420  254000  002440'         JRST TOUT
2073
2074                                  IFE EDDT&1,<
2075                                  CRNRB:  PUSHJ P,CRN
2076                                          MOVEI T,177
2077                                          JRST TOUT>
2078
2079  002421  260040  002417' CRF:    PUSHJ P,CRN
2080  002422  201240  000012         MOVEI T,12              ;LINE FEED
2081  002423  254000  002440'         JRST TOUT
2082
2083  002424  260040  002426' LCT:    IFN EDDT&1,<PUSHJ P,TSPC
2084  002425  260040  002426'         PUSHJ P,TSPC>
2085                                  IFE EDDT&1,<MOVEI T,11
2086                                  JRST TOUT>              ;TYPE A TAB
2087
2088  002426  201240  000040  TSPC:   MOVEI T,40              ;SPACE
2089  002427  254000  002440'         JRST TOUT
```

```
2090                                IFN EUDTA1,<    ;EXECUTIVE MODE TELETYPE I/O
2091
2092  002430  260040  002456' TIN:    PUSHJ P,LISTEN          ;TELETYPE CHARACTER INPUT
2093  002431  254000  002430'          JRST .-1
2094  002432  302240  000175          CAIE T,175
2095  002433  306240  000176          CAIN T,176
2096  002434  201240  000033          MOVEI T,33        ;CHANGE ALL ALT MODES TO NEW
2097  002435  306240  000177          CAIN T,177        ;RUBOUT?
2098  002436  254000  000155'         JRST WRONG        ;YES, TYPE XXX
2099  002437  602240  000140          TRNE T,140        ;DON'T ECHO CR,LF,ALT,TAB,BACK SPACE,ETC
2100  002440  307240  000004  TOUT:   CAIG T,04         ;DON'T TYPE EOT OR LOWER CHARS
2101  002441  263040  000000          POPJ P,
2102  002442  506241  000000          HRLM T,(P)
2103  002443  221240  200401          IMULI T,200401          ;GENERATE PARITY
2104  002444  404240  003206'         AND T,[111111111]
2105  002445  220240  003206'         IMUL T,[111111111]
2106  002446  544241  000000          HLR T,(P)
2107  002447  603240  000010          TLNE T,10
2108  002450  640240  000200          TRO T,200               ;MAKE PARITY EVEN
2109  002451  712320  000020          CONSZ TTYY,20
2110  002452  254000  002451'         JRST .-1
2111  002453  712140  000005          DATAO TTYY,T
2112  002454  405240  000177          ANDI T,177              ;FLUSH PARITY
2113  002455  263040  000000          POPJ P,0
2114
2115  002456  712340  000040  LISTEN: CONSO TTYY,40   ;LISTEN FOR TTY
2116  002457  263040  000000          POPJ P,
2117  002460  712040  000005          DATAI TTYY,T
2118  002461  405240  000177          ANDI T,177
2119  002462  254000  000431'         JRST CPOPJ1
2120
2121  002463  201240  003410  TTYRET: MOVEI  T,3410
2122  002464  201340  040000  TTY1:   MOVEI W2,40000
2123  002465  712320  000120          CONSZ TTYY,120
2124  002466  367340  002465'         SOJG W2,.-1
2125  002467  712240  002277'         CONI TTYY,SAVTTY
2126  002470  712040  000007          DATAI TTYY,W2
2127  002471  506340  002277'         HRLM W2,SAVTTY
2128  002472  712205  000000          CONO TTYY,(T)
2129  002473  263040  000000          POPJ P,
2130  002474  220240  002277' TTYLEV: MOVE T,SAVTTY
2131  002475  620240  000160          TRZ T,160
2132  002476  600240  003600          TRO T,3600
2133  002477  602240  000210          TRNE T,10
2134  002500  620240  000200          TRZ T,200
2135  002501  254000  002464'         JRST TTY1
2136
2137
2138  002502  260040  002430' TEXIN:  PUSHJ P,TIN       ;INPUT SUBROUTINE FOR TEXT MODES
2139  002503  606240  000140          TRNN T,140
2140  002504  254000  002440'         JRST TOUT         ;ECHO CHARACTERS (0-37) NOT ECHOED
2141  002505  263040  000000          POPJ P,
2142                              >
```

```
2143                            IFE EDDT&1,<     ;USER MODE TELETYPE I/O
2144                              IFN EDDT&4,<   ;ASSEMBLE WITH OLD DDT MODE IO
2145
2146                            TIN:    MOVE T,POUTBF           ;GET NEXT CHARACTER ROUTINE
2147                                    CAME T,[POINT 7,INBFF]
2148                                    PUSHJ P,FINOUT
2149
2150                            IFE EDDT&10,<   ILDB T,PINBFF >
2151                            IFN EDDT&10,<   PUSHJ P,INCHR >
2152
2153                                    CAIE T,176
2154                                    CAIN T,175
2155                                    MOVEI T,33      ;CHANGE TO NEW ALT MODE
2156                                    CAIN    T,177   ;RUBOUT?
2157                                    JRST    WRONG   ;YES PRINT XXX
2158                                    JUMPN T,CPOPJ
2159                                    MOVE T,[POINT 7,INBFF]
2160                                    MOVEM T,PINBFF
2161                                    CALL T,[SIXBIT /DDTIN/]
2162                                    JRST TIN
2163
2164                            TOUT:   JUMPE   T,CPOPJ ;OUT PUT A CHARACTER FLUSH NULLS
2165
2166                            IFN EDDT&10,<   SKIPE COMAND
2167                                    JRST PUTCHR >
2168
2169                                    IDPB T,POUTBF
2170                                    CAIE T,12
2171                                    POPJ P,
2172                            TTYLEV:
2173                            FINOUT: MOVEI T,0
2174                                    IDPB T,POUTBF
2175                                    MOVE T,[POINT 7,INBFF]
2176                                    MOVEM T,PINBFF
2177                                    MOVEM T,POUTBF
2178                                    CALL T,[SIXBIT /DDTOUT/]
2179                                    CLEARM INBFF
2180                                    POPJ P,
2181
2182                            PINBFF: POINT 7,INBFF
2183                            POUTBF: POINT 7,INBFF
2184
2185                            LISTEN=CPOPJ
2186                            INBFF:  BLOCK 21
2187
2188                            TTYRET: MOVE T,[POINT 7,INBFF]
2189                                    MOVEM T,POUTBF
2190                                    MOVEM T,PINBFF
2191                                    CLEARM INBFF
2192                                    POPJ P,
2193
2194                            TEXIN=TIN       ;USE NORMAL INPUT FOR TEXT WHEN IN USER MODE
2195                            >
```

```
2196
2197
2198                            IFE EDDT&4,< ;ASSEMBLE WITH TTCALL TELETYPE IO
2199
2200                            OPDEF TTCALL [51B8]
2201
2202                            TEXIN:
2203                            TIN:
2204
2205                            IFE EDDT&10,<   TTCALL 0,T              ;GET NEXT CHARACTER INTO T >
2206                            IFN EDDT&10,<   PUSHJ P,INCHR >
2207
2208                                    CAIE T,175
2209                                    CAIN T,176
2210                                    MOVEI T,33              ;CHANGE OLD ALT MODES TO NEW
2211                                    CAIN T,177
2212                                    JRST WRONG              ;TYPE XXX FOR A RUBOUT
2213                                    POPJ P,
2214

2215                            TOUT:
2216
2217                            IFN EDDT&10,<   SKIPE COMAND            ;IS THERE A COMMAND FILE?
2218                                    JRST PUTCHR                     ;YES >
2219
2220                                    TTCALL 1,T              ;OUTPUT A CHARACTER
2221                                    POPJ P,
2222
2223                            LISTEN: TTCALL 2,T              ;GET NEXT CHAR, NO IO WAIT
2224                                    POPJ P,                 ;NO CHARACTER EXISTED, RETURN
2225                                    JRST CPOPJ1             ;CHARACTER WAS THERE, SKIP RETURN
2226
2227                            TTYRET: TTCALL 6,               ;WHEN RETURNING TO DDT, FLUSH ALL
2228                                    POPJ P,                 ;WAITING INPUT CHARACTERS
2229
2230                            TTYLEV==CPOPJ                   ;NOTHING SPECIAL TO DO WHEN LEAVING DDT
2231
2232                            >       ;END OF IFN EDDT&4
2233
2234                            IFN EDDT&10,<
2235                            INCHR:  SKIPE COMAND
2236                                    JRST GETCHR
2237
2238                            IFN EDDT&4,<    ILDB T,PINBFF           ;NO COMMAND FILE >
2239                            IFE EDDT&4,<    TTCALL 0,T              ;O/P CHAR, >
2240
2241                                    POPJ P,
2242
2243                            GETCHR: SOSLE CBUF+2            ;ANY REMAINING?
2244                                    JRST GETOK              ;YES.
2245                                    INPUT CM,
2246                                    STATZ CM,740000
2247                                    HALT .+1
2248                                    STATZ CM,20000          ;END-OF-FILE?
```

```
2249                            JRST GETEND
2250
2251                    GETOK:  ILDB T,CBUF+1
2252                            JUMPE T,GETCHR          ;BYPASS ZERO CHARACTERS
2253                            PUSHJ P,PUTCHR          ;COPY INPUT TO OUTPUT FILE
2254                            POPJ P,
2255
2256                    GETEND: CLOSE DP,               ;CLOSE OUTPUT WHEN INPUT EXHAUSTED
2257                            RELEASE DP,
2258                            RELEASE CM,
2259                            JRST NOLPT              ;REVERT TO TTY WHEN COMMANDS EXHAUSTED
2260
2261                    PUTCHR: SOSLE LBUF+2            ;ANY ROOM?
2262                            JRST PUTOK              ;YES
2263                            OUTPUT DP,
2264                            STATZ DP,740000         ;ERRORS?
2265                            HALT .+1                ;YES
2266
2267                    PUTOK:
2268                            IDPB T,LBUF+1   ;DEPOSIT CHAR,
2269                            POPJ P,
2270
2271                    >       ;END OF IFN EDDT&10
2272                    >       ;END OF IFE EDDT&1
```

```
2273  002506  301402  002511' BDISP:  POINT 12,DISP(R),11
2274  002507  141402  002511'         POINT 12,DISP(R),23
2275  002510  001402  002511'         POINT 12,DISP(R),35
2276  002511                  DISP:
2277                            DEFINE D (Ž1,Ž2,Ž3)<            <Ž1-DDT>←30+<Ž2-DDT>←14+Ž3-DDT>
2278                                    ;THIS MACRO PACKS 3 ADDRESSES INTO ONE WORD; EACH ADR IS 12 BITS
2279
2280                    000151' IFE EDDT&1&<EDDT>B36,<  PUNCH==ERR
2281                    000151'         BLKEND==ERR
2282                    000151'         LOADER==ERR>
2283
2284  002511  015101  510151  D ERR,ERR,ERR↑           <ERR-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2285  002512  015101  510151  D ERR,ERR,ERR↑           <ERR-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2286  002513  015101  510731  D ERR,ERR,VARRW↑                 <ERR-DDT>←30+<ERR-DDT>←14+VARRW-DDT↑
2287  002514  075707  060151  D TAB,LINEF,ERR↑                 <TAB-DDT>←30+<LINEF-DDT>←14+ERR-DDT↑
2288  002515  015107  350151  D ERR,CARR,ERR↑          <ERR-DDT>←30+<CARR-DDT>←14+ERR-DDT↑
2289  002516  015101  510151  D ERR,ERR,ERR↑           <ERR-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2290  002517  015101  510151  D PUNCH,ERR,ERR↑                 <PUNCH-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2291  002520  015101  510151  D ERR,ERR,ERR↑           <ERR-DDT>←30+<ERR-DDT>←14+ERR-DDT↑

2292  002521  015101  510151  D ERR,ERR,ERR↑           <ERR-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2293  002522  040201  510151  D CONTROL,ERR,ERR↑               <CONTROL-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2294  002523  015101  510660  D ERR,ERR,SPACE↑                 <ERR-DDT>←30+<ERR-DDT>←14+SPACE-DDT↑
2295  002524  074204  330622  D SUPTYO,TEXI,ASSEM↑             <SUPTYO-DDT>←30+<TEXI-DDT>←14+ASSEM-DDT↑
2296  002525  025002  510151  D DOLLAR,PERC,ERR↑               <DOLLAR-DDT>←30+<PERC-DDT>←14+ERR-DDT↑
2297  002526  062006  260664  D DIVD,LPRN,RPRN↑                <DIVD-DDT>←30+<LPRN-DDT>←14+RPRN-DDT↑
2298  002527  061706  240642  D MULT,PLUS,ACCF↑                <MULT-DDT>←30+<PLUS-DDT>←14+ACCF-DDT↑
2299  002530  062303  500743  D MINUS,PERIOD,SLASH↑            <MINUS-DDT>←30+<PERIOD-DDT>←14+SLASH-DDT↑
2300  002531  023402  340234  D NUM,NUM,NUM↑           <NUM-DDT>←30+<NUM-DDT>←14+NUM-DDT↑
2301  002532  023402  340234  D NUM,NUM,NUM↑           <NUM-DDT>←30+<NUM-DDT>←14+NUM-DDT↑
2302  002533  023402  340234  D NUM,NUM,NUM↑           <NUM-DDT>←30+<NUM-DDT>←14+NUM-DDT↑
2303  002534  023405  301063  D NUM,TAG,SEMIC↑                 <NUM-DDT>←30+<TAG-DDT>←14+SEMIC-DDT↑
2304  002535  146010  051463  D FIRARG,EQUAL,ULIM↑             <FIRARG-DDT>←30+<EQUAL-DDT>←14+ULIM-DDT↑
2305  002536  021606  361047  D QUESTN,INDIRECT,ABSA↑          <QUESTN-DDT>←30+<INDIRECT-DDT>←14+ABSA-DDT↑
2306  002537  136410  412122  D BPS,CON,SYMD↑          <BPS-DDT>←30+<CON-DDT>←14+SYMD-DDT↑
2307  002540  170210  371065  D EFFEC,SFLOT,GO↑                <EFFEC-DDT>←30+<SFLOT-DDT>←14+GO-DDT↑
2308  002541  103603  730151  D HWRDS,PILOC,BLKEND↑            <HWRDS-DDT>←30+<PILOC-DDT>←14+BLKEND-DDT↑
2309  002542  050201  511672  D KILL,LOADER,MASK↑              <KILL-DDT>←30+<LOADER-DDT>←14+MASK-DDT↑
2310  002543  170525  511222  D NWORD,BITO,PROCEDE↑            <NWORD-DDT>←30+<BITO-DDT>←14+PROCEDE-DDT↑
2311  002544  037510  441040  D QUAN,RELA,SYMBOL↑              <QUAN-DDT>←30+<RELA-DDT>←14+SYMBOL-DDT↑
2312  002545  103001  510151  D TEXO,ERR,ERR↑          <TEXO-DDT>←30+<ERR-DDT>←14+ERR-DDT↑
2313  002546  170410  660151  D WORD,XEC,ERR↑          <WORD-DDT>←30+<XEC-DDT>←14+ERR-DDT↑
2314  002547  201107  370753  D ŽERO,OCON,ICON↑                <ŽERO-DDT>←30+<OCON-DDT>←14+ICON-DDT↑
2315  002550  074007  311006  D OSYM,VARRW,PSYM↑               <OSYM-DDT>←30+<VARRW-DDT>←14+PSYM-DDT↑
2316
2317                            ;THIS TABLE DOES NOT HAVE ENTRIES FOR CHARS .GE. 140; THESE
2318                            ; ARE DETECTED AS ERRORS NEAR L21:
```

```
2319
2320  002551  201170  002566' BITO:   MOVEI   R,BITT  ;BYTE OUTPUT SUBROUTINE
2321  002552  551440  002245'         HRRZI   AR,TOC
2322  002553  626000  000001          TRZN    F,02F
2323  002554  254000  000151'         JRST    ERR
2324  002555  200240  002250'         MOVE    T,WRD2
2325  002556  202240  002614'         MOVEM   T,SVBTS
2326  002557  201240  000044          MOVEI   T,'036
2327  002560  230240  002250'         IDIV    T,WRD2
2328  002561  332000  003006          SKIPE   T+1
2329  002562  271240  000001          ADDI    T,1
2330  002563  202240  002615'         MOVEM   T,SVBTS2
2331  002564  550400  000002          HRRZ    SCH,R
2332  002565  254000  001056'         JRST    BASE1
2333
2334  002566  200240  002615' BITT:   MOVE    T,SVBTS2
2335  002567  202240  002617'         MOVEM   T,SVBT2
2336  002570  200340  002254'         MOVE    T+1,LWT
2337  002571  202340  002616'         MOVEM   T+1,SVBT3
2338  002572  261040  002254'         PUSH P,LWT
2339  002573  201240  000000  BITT2:  MOVEI   T,0
2340  002574  200340  002614'         MOVE    T+2,SVBTS
2341  002575  246247  000000          LSHC    T,(T+2)
2342  002576  202240  002254'         MOVEM   T,LWT
2343  002577  202340  002616'         MOVEM   T+1,SVBT3
2344  002600  302440  001622'         CAIE    AR,PADSO
2345  002601  260040  002057'         PUSHJ   P,TOCA
2346  002602  302440  002045'         CAIE    AR,TOC
2347  002603  260040  001537'         PUSHJ   P,PIN
2348  002604  377000  002617'         SOSG    SVBT2
2349  002605  254000  002612'         JRST BITT4
2350  002606  201240  000054          MOVEI   T,","
2351  002607  260040  002440'         PUSHJ   P,TOUT
2352  002610  202340  002616'         MOVE    T+1,SVBT3
2353  002611  254000  002573'         JRST    BITT2
2354
2355  002612  262040  002254' BITT4:  POP P,LWT
2356  002613  263040  000000          POPJ P,
2357
2358  002614  000000  000000  SVBTS:  0
2359  002615  000000  000000  SVBTS2: 0
2360  002616  000000  000000  SVBT3:  0
2361  002617  000000  000000  SVBT2:  0
```

```
2362
2363                        ;DESCRIPTION OF OP DECODER FOR DDT:
2364                        ;
2365                        ;         THE ENTIRE INSTRUCTION SET FOR THE PDP-6 CAN BE COMPACTED INTO
2366                        ;A SPACE MUCH SMALLER THAN ONE REGISTER FOR EVERY SYMBOL.  THIS OCCURS
2367                        ;BECAUSE OF THE MACHINE ORGANIZATION AND INSTRUCTION MNEMONICS CHOSEN
2368                        ;FOR THE PDP-6.  FOR EXAMPLE, IF BITS (0-2) OF AN INSTRUCTION EQUAL
2369                        ;101(2) THE INSTRUCTION IS A HALF WORD INSTRUCTION AND AN "H" MAY
2370                        ;BE ASSUMED. "T" MAY BE ASSUMED FOR ALL TEST INSTRUCTIONS (WHICH
2371                        ;BEGIN WITH 110(2).
2372                        ;
2373                        ;       THE TABLE TBL IN DDT CONSISTS OF 9 BIT BYTES, 4 TO A WORD.
2374                        ;THE NUMBERS IN THE BYTES HAVE THE FOLLOWING SIGNIFICANCE:
2375                        ;0-37(8):       THIS IS A DISPATCH COMMAND FOR THE OP-DECODER INTERPRETER.
2376                        ;       LET THE RIGHT MOST TWO BITS EQUAL N; LET THE NEXT 3 BITS
2377                        ;       EQUAL P.
2378                        ;
2379                        ;       THE CONTENTS OF INST (INSTRUCTION) CONTAIN IN THE RIGHT
2380                        ;       MOST NINE BITS THE BINARY FOR THE MACHINE INSTRUCTION.

2381                        ;       P AND N REFER TO THE CONTENTS OF INST, AND THE OP DECODER
2382                        ;       WILL PRODUCE AN ANSWER D GIVEN P, N, AND THE CONTENTS
2383                        ;       OF INSTX N+1 GIVES THE NUMBER OF BITS IN INST; P GIVES THE
2384                        ;       POSITION (FROM THE RIGHT EDGE) OF THE N+1 BITS.
2385                        ;
2386                        ;       EXAMPLE: P = 6
2387                        ;                N = 2
2388                        ;
2389                        ;;      C(INST) = ,010 101 100(2)
2390                        ;
2391                        ;       THE RESULT = D = 010(2) = 2(8)
2392                        ;
2393                        ;       D IS USED AS A DISPATCH ON THE NEXT BYTES IN THE TABLE.
2394                        ;       IF D = 5, 5 BYTES IN THE TABLE (DON'T COUNT THE BYTES WHICH
2395                        ;       PRINT TEXT, 41-72(8)) ARE SKIPPED OVER AND THE 6TH BYTE RESUMES
2396                        ;       THE INTERPRETATION.
2397                        ;
2398                        ;40(8)  THIS IS A STOP CODE; WHEN THIS IS REACHED INTERPRETATION
2399                        ;       IS FINISHED.
```

```
2400                    ;41(8)-72(8)     THE ALPHABET IS ENCODED INTO THIS RANGE.
2401                    ;                41- A
2402                    ;                42- B
2403                    ;                72- Z
2404                    ;                WHEN A BYTE IN THIS RANGE IS REACHED, ITS CORRESPONDING
2405                    ;                LETTER IS TYPED.
2406                    ;
2407                    ;73(8)-777(8)    THIS IS A TRANSFER BYTE.  IF THE BYTE IN THIS RANGE IS
2408                    ;                CONSIDERED TO BE A, TRANSFER INTERPRETATION TO THE A-73(8)RD
2409                    ;                BYTE IN THE TABLE.
2410                    ;
2411                    ;MACROS ASSEMBLE THE TABLE TBL:
2412                    ; 1.    A NUMBER FOLLOWED BY * ASSEMBLES A DISPATCH BYTE.  THE FIRST
2413                    ;       DIGIT IS THE POSITION; THE SECOND DIGIT IS THE SIZE.
2414                    ; 2.    A POINT (.) ASSEMBLES A STOP CODE.
2415                    ; 3.    A NAME FOLLOWED BY A SLASH ASSEMBLES A TRANSFER TO THE
2416                    ;       SYMBOLICALLY NAMED BYTE.
2417                    ; 4.    A STRING OF LETTERS TERMINATED BY A SPACE, COMMA, OR POINT,
2418                    ;       ASSEMBLE INTO A STRING OF BYTES, EACH BYTE BEING ONE LETTER.
2419                    ;
2420                    ;EXAMPLE OF BINARY TO SYMBOLIC DECODING:
2421                    ;       THE MACHINE CODE FOR JRST IS 254
2422                    ;           INST    0  1  0  1  0  1  1  0  0
2423                    ;       THE INTERPRETER STARTS WITH THE FIRST BYTE IN THE TABLE (63*),
2424                    ;       THE RESULT OF APPLYING THIS TO C(INST) GIVES 2.  SKIPPING OVER
2425                    ;       2 BYTES IN THE TABLE AND INTERPRETING THE THIRD RESULTS IN
2426                    ;       HAK/ BEING INTERPRETED.  AT HAK:, THERE IS A 33*.  APPLYING
2427                    ;       THIS TO C(INST) RESULTS IN 5 NON PRINTING BYTES BEING SKIPPED
2428                    ;       OVER:
2429                    ;           1.  MV/
2430                    ;                MOV     PRINTING TEXT
2431                    ;           2.  MO/
2432                    ;           3.  ML/
2433                    ;           4.  UV/
2434                    ;           5.  SH/
2435                    ;
2436                    ;H1/ IS THE NEXT BYTE INTERPRETER.  AT H1: 03* IS FOUND SO
2437                    ;4 BYTES ARE SKIPPED OVER:
2438                    ;               EXC     PRINTING TEXT
2439                    ;           1.  S3/
2440                    ;               BL      PRINTING TEXT
2441                    ;               T       PRINTING TEXT
2442                    ;           2.  .
2443                    ;           3.  A0/
2444                    ;           4.  A08/
2445                    ;           THE NEXT LETTERS JRS ARE TYPED OUT,  THEN T/ IS FOUND.  AT
2446                    ;T; A T IS TYPED OUT; THEN A "." IS FOUND AND EVERYTHING STOPS.
2447                    ;
2448                    ;           THE TABLE IS ALSO USED GOING FROM SYMBOLIC TO BINARY BUT A
2449                    ;       TREE SEARCH METHOD IS USED.
```

```
2450                            REPEAT 0,<
2451
2452                            DEFINE REDEF (XX)<
2453                            DEFINE INFO (AA,BB)<
2454                            AA XX'BB>>
2455
2456
2457                            DEFINE BYTE (L)<
2458                            XLIST
2459                            REDEF %
2460                             ZZ==0
2461                             ZZZ==0
2462                             ZZM==1
2463
2464                             IRPC L,<
2465                                     Z=="L"
2466                                     IFE Z-":",<INFO <>,<==CLOC>
2467                                             IFNDEF FIR.,<FIR.==CLOC>
2468                                             IFGE CLOC+73-1000-FIR.,<PRINTX OPTABLE TOO LONG>

2469                                             Z==0>
2470                                     IFE Z-"/",<IF1 <OUTP 1>
2471                                             IF2,<INFO OUTP,+73-FIR.>
2472                                             Z==0>
2473                             IFE Z-"*",<OUTP <ZZ&74/2+ZZ&7-1>
2474                                     Z==0>
2475                             IFE <Z-",">*<Z-",">*<Z-40>,<IFN ZZZ,<
2476                                                             REPEAT 5,<ZZ==ZZZ&77
2477                                                                     IFN ZZ,<OUTP ZZ>
2478                                                                     ZZZ==ZZZ/100>>
2479                                                             IFE Z-",",<OUTP 40>
2480                                                             Z==0>
2481                             IFN Z,<INFO REDEF,L
2482                                     ZZ==ZZ*13+Z&7
2483                                     ZZZ==ZZZ+<Z-40>*ZZM
2484                                     ZZM==ZZM*100>
2485                             IFE Z,<REDEF %
2486                                     ZZ==0
2487                                     ZZZ==0
2488                                     ZZM==1>>
2489                            LIST>
2490
2491                            DEFINE OUTP (A)<
2492                                     BINRY==BINRY*400+BINRY*400+A
2493                                     BINC==BINC-1
2494                                     IFE BINC,<EXP BINRY
2495                                             BINRY==0
2496                                             BINC==4>
2497                                     CLOC==CLOC+1>
2498
2499                            TBL:             ;OPDECODER BYTE TABLE
2500
2501                            BINRY==0
2502                            CLOC==0          ;SET BYTE LOCATION COUNTER TO 0
```

```
2503                            BINC==4          ;INIT BYTES/WORD COUNTER
2504
2505                            BYT9 <63↑UUO/FLO/HAK/ACCP/BOOLE/H HWT/T ACBM/>
2506
2507                            ;IO INSTRUCTIONS
2508                            BYT9 <21↑BO/CON,11↑OI/S,01↑Z/O/>
2509                            BYT9 <BD:01↑BLK,IO/DATA,IO:11↑I/O/OI:01↑O/I/>
2510
2511                            ;UUOS
2512                            BYT9 <UUO:51↑,,32↑U40/U50/U60/21↑U703/11↑USET/01↑>
2513                            BYT9 <LOOKU,P/ENTE,R/USET:USET,01↑I/O/>
2514                            BYT9 <U40:03↑CAL/INI T/.....,CALL I/>
2515                            BYT9 <U60:21↑U603/01↑IN,BPUT/OUT,BPUT:11↑BU,F:F,,PU,T/>
2516                            BYT9 <U603:01↑U6062/STAT,11↑O:O,,Z:Z,,U6062:11↑S,U62/G,U62:ETST,S/>
2517                            ;BYTE AND FLOATING INSTRUCTIONS
2518
2519                            BYT9 <FLO:51↑BYTE/F 32↑ AD A/SB A/MP A/DV A:>
2520                            BYT9 <21↑LMB/R,IMB/LMB:02↑,,L:L,,M:M,,B:B,,BYTE:32↑,,,,03↑UF,PA/DF,N/>
2521                            BYT9 <FS C/IB P:P,,I LD/LD:LD B/I DP/DP:DP B/>
2522
2523                            ;FWT,FIXED POINT ARITH,MISC.
2524
2525                            BYT9 <HAK:33↑MV/MV:MOV MO/ML/DV/SH/H1/JP/>
2526                            BYT9 <21↑ADD IMB/SU BIMB:B IMB:02↑,,I:I,,M/B/MO:22↑>
2527                            BYT9 <EIMS:E IMS/S IMS/N IMS/M IMS:02↑,,I/M/S:S,,>
2528                            BYT9 <ML:21↑I ML1/ML1:MUL IMB/DV:21↑I DV1/DV1:>
2529                            BYT9 <DI DV2:V IMB/H1:03↑EXC S3/BL T:T,,AO/AO:AOBJ,>
2530                            BYT9 <AOB/JRS T/JFC L/XC T/,AOB:01↑P/N/>
2531                            BYT9 <JP:03↑PU/PU:PUSH PUS/PO/PO:POP POP/JS,R:R,,>
2532                            BYT9 <JS P/JS PA:A,,JR PA/PUS:01↑J:J,,,POP:>
2533                            BYT9 <01↑,,J/SH:02↑A S2/ROT S1/L S2:S S3:H S1/JFF O/S1:21↑,,C:C,,>
2534
2535                            ;ARITH COMP,SKIP,JUMP
2536
2537                            BYT9 <ACCP:42↑CA CA1/SJ/A JS/S JS:O 31↑>
2538                            BYT9 <J COMP/S COMP/CA1:31↑I COMP/M COMP/>
2539                            BYT9 <SJ:31↑JUM PSJ/SKI PSJ:P COMP:>
2540                            BYT9 <03↑,,L/E:E,,L E/PA/G E/N:N,,G,,>
2541
2542                            ;HALF WORDS
2543
2544                            BYT9 <HWT:51↑HW1/21↑R HW2/L HW2:R HW3/HW1:>
2545                            BYT9 <21↑L HW4/R HW4:L HW3:32↑IMS/Z IMS/O IMS/EIMS/>
2546
2547                            ;TEST INSTRUCTIONS
2548
2549                            BYT9 <ACBM:31↑AC1/01↑D AC2/S AC2/AC1:01↑R AC2/L,>
2550                            BYT9 <AC2:42↑N EAN/Z EAN/C EAN/O EAN:12↑,,E/PA/N/>
2551
2552
2553
2554                            ;BOOLEAN
2555
```

```
                        BYT9 <BOOLE:24+ST/AN:AND B2/AN/ST/AN/ST/>
                        BYT9 <X OR:OR B2/I OR/AN/EQ OV2/ST/OR/ST/OR/OR/>
                        BYT9 <ST:SET B2:24+Z IMB/IMB/CA:C TA/TM:M IMB/>
                        BYT9 <CM:C TM/TA:A IMB/IMB/IMB/CR:C BIMB/IMB/CA/>
                        BYT9 <CA/CM/CM/CB/D IMB/>

                        ;MORE UUO'S
                        BYT9 <U50:03+OPE,N/TT,CAL:CAL L/,,,,RENAM,E/I,N/OU,T/>
                        BYT9 <U703:02+CLOS,E/RELEA,S/MTAP,E/UGET,F/>

                        REPEAT BINC,<BINRY==BINRY*400+BINRY*400>
                        IFN BINRY,<EXP BINRY>
                        >       ;END OF REPEAT 0
```

```
2569                    IFN FDDT&10,<                   ;FILDDT STUFF
2570    CRASH:  SIXBIT .CRASH.          ;CANONICAL NAME FOR CRASH
2571            SIXBIT .SAV.
2572            Z
2573            Z
2574    COMNDS: SIXBIT .FILDDT.         ;CANONICAL NAME FOR COMMAND LIST
2575            SIXBIT .TXT.
2576            Z
2577            Z
2578    SNAP:   SIXBIT .SNAP.           ;NAME FOR OUTPUT IF TO RETRIEVABLE DEVICE
2579            SIXBIT .LST.
2580            Z
2581            Z
2582    CBUF:   BLOCK 3                 ;RING HEADERS
2583    LBUF:   BLOCK 3
2584    COMAND: Z                       ;-1 IF COMMAND FILE ,0 IF NOT
2585    CRASHS: Z                       ;-1 IF CRASH.SAV ON DISK ,0 IF PEEK AT MONITOR
2586            AC0=.
2587            AC17=.+17
2588    RSIDNT: BLOCK 1000              ;LOCS 0-777 ALWAYS IN CORE
2589    CURENT: BLOCK 4000              ;WINDOW TO THE FILE ON DISK
2590    RSAVE:  BLOCK 1                 ;INDEX OF THE CURRENT BLOCK, 0,1,...
2591    RSILST: IOWD 1000,RSIDNT
2592            Z
2593    CURLST: IOWD 4000,CURENT
2594            Z
2595            >
```

```
2596                            SUBTTL  OP DECODER
2597
2598                            DEFINE BYTS (A) <IRP A,<
2599                            A>>
2600
2601                            IF1,<
2602
2603                            DEFINE  .ADR    (A) <
2604                            %'A==   CLOC
2605                            FIR.==  CLOC
2606                            DEFINE  .ADR    (A) <
2607                            %'A==   CLOC
2608                            IFGE    <LASTB==CLOC+73-FIR,>-1000, <PRINTX OPTABLE TOO LONG>>>
2609
2610                            DEFINE  .TRA    <
2611                            CLOC==  CLOC+1 ;>
2612
2613                            SYN     .TRA,   .DIS
2614

2615                            DEFINE  .TXT    (A) <
2616                            IFNB    <A>,    <IRPC A,<CLOC==CLOC+1>>>
2617
2618                            DEFINE  .END    (A) <
2619                            IFNB    <A>,    <IRPC A,<CLOC==CLOC+1>>
2620                            CLOC==  CLOC+1>
2621
2622                            >       ;END OF IF1
2623                            IF2,<
2624
2625                            DEFINE .ADR (A)<IFN %'A-CLOC,<PRINTX PHASE ERR AT: %'A>>
2626
2627                            DEFINE  .TRA    (A) <
2628                            OUTP    %'A+73-FIR,>
2629
2630                            DEFINE  .DIS    (A) <
2631                            OUTP    A&70/2+A&7-1>
2632
2633                            DEFINE  .TXT    (A) <
2634                            IFNB    <A>,    <IRPC A,<OUTP "A"-40>>>
2635
2636                            DEFINE  .END    (A) <
2637                            IFNB    <A>,    <IRPC A,<OUTP "A"-40>>
2638                            OUTP    40>
2639
2640                            DEFINE OUTP (A)<
2641                            BINRY== BINRY+<A>*<BINC==BINC-9>
2642                            IFE BINC, <
2643                                            +BINRY
2644                            BINRY==0
2645                            BINC==^D36 >
2646                            CLOC==CLOC+1 >
2647
2648                            >
```

```
2649  002620                    TBL:  ;OPDECODER BYTE TABLE
2650
2651                    000000  BINRY== 0
2652                    000000  CLOC== 0 ;SET BYTE LOCATION COUNTER TO 0
2653                    000044  BINC== ^D36 ;INIT BYTES/WORD COUNTER
2654
2655                            ;**********THE ARGUMENT FOR THE FOLLOWING "BYT9" MACRO
2656                            ;**************TERMINATES AT THE NEXT COMMENT WITH: **************
2657
2658                            BYT9 <
2659
2660                            .DIS 63,.TRA UUO,.TRA FLO,.TRA HAK,.TRA ACCP,.TRA BOOLE
2661                                    .TXT H,.TRA HWT,.TXT T,.TRA ACBM
2662
2663
2664                            ;IO INSTRUCTIONS
2665
2666                            .DIS 21,.TRA BO,.TXT CON,.DIS 11,.TRA OI,.TXT S,.DIS Ø1,.TRA Z,.TRA O
2667                            .ADR BO,.DIS Ø1,.TXT BLK,.TRA IO,.TXT DATA,.ADR IO,.DIS 11,.TRA I,.TRA O
2668                                    .ADR OI,.DIS Ø1,.TRA O,.TRA I
2669                            ;UUOS
2670
2671                            .ADR UUO,.DIS 51,.END,.TXT,.DIS 32,.TRA U4Ø,.TRA U5Ø,.TRA U6Ø
2672                                    .DIS 21,.TRA U7Ø3,.DIS 11,.TRA USET,.DIS Ø1
2673                            .TXT LOOKU,.TRA P,.TXT ENTE,.TRA R,.ADR USET,.TXT USET,.DIS Ø1,.TRA I,.TRA O
2674                            .ADR U4Ø,.DIS Ø3,.TRA CAL,.TXT INI,.TRA T,.END,.END,.END,.END,.END,.TXT CALL,.TRA I
2675                            .ADR U6Ø,.DIS 21,.TRA U6Ø3,.DIS Ø1,.TXT IN,.TRA BPUT,.TXT OUT
2676                                    .ADR BPUT,.DIS 11,.TXT BU,.ADR F,.END F,.TXT,.TXT PU,.TRA T
2677                            .ADR U6Ø3,.DIS Ø1,.TRA U6Ø62,.TXT STAT,.DIS 11,.ADR O,.END O,.TXT,.ADR Z,.END Z,.TXT
2678                                    .ADR U6Ø62,.DIS 11,.TXT S,.TRA U62,.TXT G,.ADR U62,.TXT ETST,.TRA S
2679
2680                            ;BYTE AND FLOATING INSTRUCTIONS
2681
2682                            .ADR FLO,.DIS 51,.TRA BYTE,.TXT F,.DIS 32,.TXT,.TXT AD,.TRA A,.TXT SB
2683                                    .TRA A,.TXT MP,.TRA A,.TXT DV,.ADR A
2684                            .DIS 21,.TRA LMB,.TXT R,.TRA IMB,.ADR LMB,.DIS Ø2,.END,.TXT
2685                                    .ADR L,.END L,.TXT,.ADR M,.END M,.TXT
2686                            .ADR B,.END B,.TXT,.ADR BYTE,.DIS 32,.END,.END,.END,.TXT
2687                                    .DIS Ø3,.TXT UF,.TRA PA,.TXT DF,.TRA N
2688                            .TXT FS,.TRA C,.TXT IB,.ADR P,.END P,.TXT,.TXT I,.TRA LD
2689                                    .ADR LD,.TXT LD,.TRA B,.TXT I,.TRA DP,.ADR DP,.TXT DP,.TRA B
2690
2691                            ;FWT-FIXED POINT ARITH-MISC
2692
2693                            .ADR HAK,.DIS 33,.TRA MV,.ADR MV,.TXT MOV,.TRA MO,.TRA ML,.TRA DV
2694                                    .TRA SH,.TRA H1,.TRA JP
2695                            .DIS 21,.TXT ADD,.TRA IMB,.TXT SU,.ADR BIMB,.TXT B,.ADR IMB,.DIS Ø2,.END,.TXT
2696                                    .ADR I,.END I,.TXT,.TRA M,.TRA B,.ADR MO,.DIS 22
2697                            .ADR EIMS,.TXT E,.TRA IMS,.TXT S,.TRA IMS,.TXT N,.TRA IMS,.TXT M
2698                                    .ADR IMS,.DIS Ø2,.END,.TXT,.TRA I,.TRA M,.ADR S,.END S,.TXT
2699                            .ADR ML,.DIS 21,.TXT I,.TRA ML1,.ADR ML1,.TXT MUL,.TRA IMB
2700                                    .ADR DV,.DIS 21,.TXT I,.TRA DV1
2701                            .ADR DV1,.TXT DI,.ADR DV2,.TXT V,.TRA IMB,.ADR H1,.DIS Ø3,.TXT EXC,.TRA S3,.TXT BL
```

```
2702                            .ADR T,.END T,.TXT,.TRA AO,.ADR AO,.TXT AOBJ
2703                    .TRA AOB,.TXT JRS,.TRA T,.TXT JFC,.TRA L,.TXT XC,.TRA T,.END
2704                            .ADR AOB,.DIS Ø1,.TRA P,.TRA N
2705                    .ADR JP,.DIS Ø3,.TRA PU,.ADR PU,.TXT PUSH,.TRA PUS,.TRA PO
2706                            .ADR PO,.TXT POP,.TRA POP,.TXT JS,.ADR R,.END R,.TXT
2707                    .TXT JS,.TRA P,.TXT JS,.ADR PA,.END A,.TXT,.TXT JR,.TRA PA
2708                            .ADR PUS,.DIS Ø1,.ADR J,.END J,.END,.TXT,.ADR POP
2709                    .DIS Ø1,.END,.TXT,.TRA J,.ADR SH,.DIS Ø2,.TXT A,.TRA S2,.TXT ROT,.TRA S1,.TXT L
2710                            .ADR S2,.TXT S,.ADR S3,.TXT H,.TRA S1,.TXT JFF,.TRA O
2711                            .ADR S1,.DIS 21,.END,.TXT,.ADR C,.END C,.TXT
2712
2713                    ;ARITH COMP-SKIP-JUMP
2714
2715                    .ADR ACCP,.DIS 42,.TXT CA,.TRA CA1,.TRA SJ,.TXT A,.TRA JS,.TXT S
2716                            .ADR JS,.TXT O,.DIS 31
2717                    .TXT J,.TRA COMP,.TXT S,.TRA COMP,.ADR CA1,.DIS 31,.TXT I,.TRA COMP,.TXT M,.TRA COMP
2718                    .ADR SJ,.DIS 31,.TXT JUM,.TRA PSJ,.TXT SKI,.ADR PSJ,.TXT P,.ADR COMP
2719                    .DIS Ø3,.END,.TXT,.TRA L,.ADR E,.END E,.TXT,.TXT L,.TRA E,.TRA PA,.TXT G,.TRA E
2720                            .ADR N,.END N,.TXT,.END G,.TXT

2721
2722                    ;HALF WORDS
2723
2724                    .ADR HWT,.DIS 51,.TRA HW1,.DIS 21,.TXT R,.TRA HW2,.TXT L,.ADR HW2,.TXT R,.TRA HW3
2725                    .ADR HW1,.DIS 21,.TXT L,.TRA HW4,.TXT R,.ADR HW4,.TXT L
2726                            .ADR HW3,.DIS 32,.TRA IMS,.TXT Z,.TRA IMS,.TXT O,.TRA IMS,.TRA EIMS
2727
2728                    ;TEST INSTRUCTIONS
2729
2730                    .ADR ACBM,.DIS 31,.TRA AC1,.DIS Ø1,.TXT O,.TRA AC2,.TXT S,.TRA AC2
2731                            .ADR AC1,.DIS Ø1,.TXT R,.TRA AC2,.TXT L
2732                    .ADR AC2,.DIS 42,.TXT N,.TRA EAN,.TXT Z,.TRA EAN,.TXT C,.TRA EAN,.TXT O
2733                            .ADR EAN,.DIS 12,.END,.TXT,.TRA E,.TRA PA,.TRA N
2734
2735                    ;BOOLEAN
2736
2737                    .ADR BOOLE,.DIS 24,.TRA ST,.ADR AN,.TXT AND,.TRA B2,.TRA AN,.TRA ST,.TRA AN,.TRA ST
2738                    .TXT X,.ADR OR,.TXT OR,.TRA B2,.TXT I,.TRA OR,.TRA AN,.TXT EQ
2739                            .TRA DV2,.TRA ST,.TRA OR,.TRA ST,.TRA OR,.TRA OR
2740                    .ADR ST,.TXT SET,.ADR B2,.DIS 24,.TXT Z,.TRA IMB,.TRA IMB
2741                            .ADR CA,.TXT C,.TRA TA,.ADR TM,.TXT M,.TRA IMB
2742                    .ADR CM,.TXT C,.TRA TM,.ADR TA,.TXT A,.TRA IMB,.TRA IMB,.TRA IMB
2743                            .ADR CP,.TXT C,.TRA BIMB,.TRA IMB,.TRA CA
2744                    .TRA CA,.TRA CM,.TRA CM,.TRA CB,.TXT O,.TRA IMB
2745
2746                    ;MORE UUO'S
2747
2748                    .ADR U50,.DIS Ø3,.TXT OPE,.TRA N,.TXT TT,.ADR CAL,.TXT CAL,.TRA L,.END,.END,.END
2749                            .TXT,.TXT RENAM,.TRA E,.TXT I,.TRA N,.TXT OU,.TRA T
2750                    .ADR U703,.DIS Ø2,.TXT CLOS,.TRA E,.TXT RELEA,.TRA S
2751                            .TXT MTAP,.TRA E,.TXT UGET,.TRA F
2752
2753                    ;**********THIS TERMINATES THE "RYT9" MACRO ARGUMENT******
2754                    >;IRP
```

```
 2755  002620  032112  234323                +BINRY
 2756  002621  532660  050604                +BINRY
 2757  002622  064630  010073                +BINRY
 2758  002623  043057  056004                +BINRY
 2759  002624  107063  000221                +BINRY
 2760  002625  217000  042054                +BINRY
 2761  002626  053104  044041                +BINRY
 2762  002627  064041  004350                +BINRY
 2763  002630  217000  217350                +BINRY
 2764  002631  024040  015147                +BINRY
 2765  002632  744167  010775                +BINRY
 2766  002633  004140  000054                +BINRY
 2767  002634  057057  053065                +BINRY
 2768  002635  307045  056064                +BINRY
 2769  002636  045464  065063                +BINRY
 2770  002637  045064  000350                +BINRY
 2771  002640  217002  753051                +BINRY
 2772  002641  056051  417040                +BINRY
 2773  002642  040040  040040                +BINRY
 2774  002643  043041  054054                +BINRY
 2775  002644  350010  210000                +BINRY
 2776  002645  051056  200057                +BINRY
 2777  002646  065064  004042                +BINRY
 2778  002647  065046  040060                +BINRY
 2779  002650  065417  000223                +BINRY
 2780  002651  063064  041064                +BINRY
 2781  002652  004057  040072                +BINRY
 2782  002653  040004  063227                +BINRY
 2783  002654  047045  064063                +BINRY
 2784  002655  064370  024267                +BINRY
 2785  002656  046015  041044                +BINRY
 2786  002657  253063  042253                +BINRY
 2787  002660  055060  253044                +BINRY
 2788  002661  066010  257062                +BINRY
 2789  002662  346001  040054                +BINRY
 2790  002663  040055  040042                +BINRY
 2791  002664  040015  040040                +BINRY
 2792  002665  040002  065046                +BINRY
 2793  002666  473044  046600                +BINRY
 2794  002667  346063  530051                +BINRY
 2795  002670  042060  040051                +BINRY
 2796  002671  313054  044265                +BINRY
 2797  002672  051320  044060                +BINRY
 2798  002673  265016  325055                +BINRY
 2799  002674  057066  354372                +BINRY
 2800  002675  401507  410446                +BINRY
 2801  002676  010041  044044                +BINRY
 2802  002677  346063  065042                +BINRY
 2803  002700  001040  051040                +BINRY
 2804  002701  263265  011045                +BINRY
 2805  002702  364063  364056                +BINRY
 2806  002703  364055  001040                +BINRY
 2807  002704  350263  063040                +BINRY
```

```
2858  002705  210051  375055                  +BINRY
2859  002706  065054  346010                  +BINRY
2860  002707  051474  044051                  +BINRY
2861  002710  266346  002045                  +BINRY
2862  002711  070043  520042                  +BINRY
2863  002712  054064  047422                  +BINRY
2864  002713  241057  042052                  +BINRY
2865  002714  443052  062063                  +BINRY
2866  002715  417052  046043                  +BINRY
2867  002716  261070  043417                  +BINRY
2868  002717  040000  307600                  +BINRY
2869  002720  002450  060065                  +BINRY
2870  002721  063050  500456                  +BINRY
2871  002722  060057  060504                  +BINRY
2872  002723  052063  062040                  +BINRY
2873  002724  052063  307052                  +BINRY
2874  002725  063041  040052                  +BINRY
2875  002726  062473  000052                  +BINRY
2876  002727  040040  000040                  +BINRY

2877  002730  501001  041517                  +BINRY
2878  002731  062057  064526                  +BINRY
2879  002732  054063  050526                  +BINRY
2880  002733  052046  046217                  +BINRY
2881  002734  010040  043040                  +BINRY
2882  002735  021043  041550                  +BINRY
2883  002736  555041  542063                  +BINRY
2884  002737  057014  052566                  +BINRY
2885  002740  063566  014051                  +BINRY
2886  002741  566055  566014                  +BINRY
2887  002742  052065  055565                  +BINRY
2888  002743  063053  051060                  +BINRY
2889  002744  002040  261045                  +BINRY
2890  002745  040054  571473                  +BINRY
2891  002746  047571  056040                  +BINRY
2892  002747  047040  024614                  +BINRY
2893  002750  010062  612054                  +BINRY
2894  002751  062621  010054                  +BINRY
2895  002752  620062  054015                  +BINRY
2896  002753  364072  364057                  +BINRY
2897  002754  364355  014637                  +BINRY
2898  002755  000044  643063                  +BINRY
2899  002756  643000  062643                  +BINRY
2900  002757  054021  056653                  +BINRY
2901  002760  072653  043653                  +BINRY
2902  002761  057005  040571                  +BINRY
2903  002762  473600  013711                  +BINRY
2904  002763  041056  044714                  +BINRY
2905  002764  662711  662711                  +BINRY
2906  002765  070057  062714                  +BINRY
2907  002766  051673  662045                  +BINRY
2908  002767  061406  711673                  +BINRY
2909  002770  711673  673063                  +BINRY
2910  002771  045064  013072                  +BINRY
```

```
 2861  002772  346346  043726                    +BINRY
 2862  002773  055346  043722                    +BINRY
 2863  002774  041346  346346                    +BINRY
 2864  002775  043345  346720                    +BINRY
 2865  002776  720724  724732                    +BINRY
 2866  002777  057346  002057                    +BINRY
 2867  003000  060045  600064                    +BINRY
 2868  003001  064043  041054                    +BINRY
 2869  003002  261040  040040                    +BINRY
 2870  003003  062045  056041                    +BINRY
 2871  003004  055571  051600                    +BINRY
 2872  003005  057065  417001                    +BINRY
 2873  003006  043054  057063                    +BINRY
 2874  003007  571062  045054                    +BINRY
 2875  003010  045041  370055                    +BINRY
 2876  003011  064041  060571                    +BINRY
 2877  003012  065047  045064                    +BINRY
 2878
 2879
 2880                          IF1,<   RLOCK   <CLOC+3>/4>
 2881
 2882  003013  203000  000000  IF2,<   IFN BINC-.D36,< +BINRY>>
 2883
 2884
 2885
 2886                          IFNDEF CLOC.,<CLOC.==CLOC>
 2887                          IFN CLOC.-CLOC,<PRINTX PHASE ERROR IN OPTABLE>
```

```
2888
2889  003014  000000  003015' PNTR:   EXP INST          ;POINTER TO BITS IN INST
2890  003015  000000  000000  INST:   0                 ;BINARY FOR INSTRUCTION
2891  003016  000000  000000  CHP:    0                 ;CHAR POINTER INTO TXT, TXT+1
2892  003017                  TXT:    BLOCK 2           ;STORE INPUT TEXT FOR OPEVAL
2893  003021  000000  000000  SAVPDL: 0                 ;SAVE PUSH DOWN LIST POINTER
2894
2895  003022  441100  002620' BTAB:   POINT 9,TBL       ;TABLE USED TO GET NEXT BYTE POINTER
2896  003023  331100  002620'         POINT 9,TBL,8     ;FOR TRANSFER BYTE
2897  003024  221100  002620'         POINT 9,TBL,17
2898  003025  111100  002620'         POINT 9,TBL,26
2899
2900  003026  201240  000000  OPEVAL: MOVEI T,0                ;EVALUATE FOR AN OP CODE
2901  003027  136240  003016'         IDPB T,CHP              ;INSERT NULL IN TEXT FOR SYMBOL
2902  003030  202040  003021'         MOVEM P,SAVPDL
2903  003031  624000  000004         TRZA F,OUTF
2904  003032  660000  000004  OPTYPE: TRO F,OUTF               ;TYPE AN OPCODE SYMBOLICALLY
2905  003033  242240  777745         LSH T,-33
2906  003034  202240  003015'         MOVEM T,INST            ;GET OPCODE INTO RIGHT 9 BITS

2907  003035  200240  003152'         MOVE T,[XWD 440700,TXT]
2908  003036  202240  003016'         MOVEM T,CHP             ;FOR OPEVAL,SETUP POINTER TO INPUT TEXT
2909  003037  620000  000002         TRZ F,ITF               ;CLEAR INSTRUCTION TYPED FLAG
2910  003040  403100  000006         CLEARB R,W1
2911  003041  200340  003022'         MOVE W2,BTAB
2912  003042  134240  000007  DC1:    ILDB T,W2               ;GET NEXT BYTE IN TBL
2913  003043  303240  000040         CAILE T,40
2914  003044  301240  000073         CAIL T,73
2915  003045  365100  003042'         SOJGE R,DC1             ;SKIP OVER # BYTES = C(R)
2916  003046  327100  003042'         JUMPG R,DC1             ;SKIP OVER ALPHA TEXT WITHOUT COUNTING
2917  003047  275240  000040         SUBI T,40
2918  003050  322240  003077'         JUMPF T,DECX            ;TRANSFER ON ASTOP CODE
2919  003051  327240  003062'         JUMPG T,DC2
2920  003052  137240  003207'         DPB T,[XWD 340500,PNTR] ;SETUP R ON A DISPATCH BYTE
2921  003053  620240  777774         TRZ T,-4
2922  003054  350000  000005         AOS T
2923  003055  137240  003210'         DPB T,[XWD 300600,PNTR]
2924  003056  606000  000004         TRNN F,OUTF
2925  003057  254000  003116'         JRST DC6                ;FOR OPEVAL ONLY
2926  003060  135100  003014'         LDB R,PNTR              ;GET # BYTES TO SKIP OVER
2927  003061  254000  003042'         JRST DC1
2928
2929  003062  571245  777745  DC2:    HRREI T,-33(T)
2930  003063  321240  003071'         JUMPL T,DECT            ;TYPE OUT A LETTER
2931  003064  201305  000025         MOVEI W1,FIR,(T)                ;BYTE IS A TRANSFER
2932  003065  231300  000004         IDIVI W1,4
2933  003066  200347  003022'         MOVE W2,BTAB(W2)        ;CALCULATE POINTER TO NEXT BYTE
2934  003067  271346  000000         ADDI W2,(W1)
2935  003070  254000  003042'         JRST DC1
2936
```

```
2937
2938  003071  602000  000004  DECT:   TRNE F,OUTF
2939  003072  254000  003142'         JRST DC8          ;TYPE OUT A LETTER
2940  003073  134300  003016'         ILDB W1,CHP       ;GET NEXT INPUT LETTER
2941  003074  302305  000133         CAIE W1,133(T)    ;COMPARE WITH ASSUMED NEXT LETTER
2942  003075  254000  003103'         JRST NOMAT        ;DOESNT MATCH
2943  003076  254000  003042'         JRST DC1          ;MATCHES, TRY NEXT
2944
2945  003077  602000  000004  DECX:   TRNE F,OUTF        ;STOP (CODE 40) HAS BEEN SEEN
2946  003100  263040  000000         POPJ P,           ;IF FOR OUTPUT, RETURN
2947  003101  134300  003016'         ILDB W1,CHP       ;GET NEXT INPUT CHAR IF ANY
2948  003102  322300  003132'         JUMPE W1,DC7      ;DOES # OF CHARS MATCH
2949  003103  262040  000002  NOMAT:  POP P,R           ;NO, BACK UP AND TRY SOME MORE
2950  003104  262040  000007         POP P,W2
2951  003105  262040  003014'         POP P,PNTR
2952  003106  262040  003016'         POP P,CHP
2953  003107  350000  000002  NOMAT1: AOS R             ;ASSUME NEXT NUMBER FOR BIN VALUE
2954  003110  137100  003014'         DPB R,PNTR        ;STUFF INTO ANSWER
2955  003111  135100  003014'         LDB R,PNTR
2956  003112  326100  003120'         JUMPN R,DC6AA     ;IF =0, BYTE WAS TOO BIG
2957  003113  312040  003021'         CAME P,SAVPDL
2958  003114  254000  003103'         JRST NOMAT        ;NOT AT TOP LEVEL
2959  003115  263040  000000         POPJ P,           ;UNDEFINED, FINALLY
2960
2961  003116  201100  000000  DC6:    MOVEI R,0         ;ASSUME 0 FOR INITIAL BINARY VALUE
2962  003117  137100  003014'         DPB R,PNTR
2963  003120  316040  003021'  DC6AA: CAMN P,SAVPDL
2964  003121  254000  003125'         JRST DC6BB
2965  003122  135241  777776         LDB T,-2(P)       ;OLD VALUE OF PNTR
2966  003123  312241  000000         CAME T,(P)
2967  003124  254000  003107'         JRST NOMAT1
2968  003125  261040  003016'  DC6BB: PUSH P,CHP
2969  003126  261040  003014'         PUSH P,PNTR
2970  003127  261040  000007         PUSH P,W2
2971  003130  261040  000002         PUSH P,R
2972  003131  254000  003042'         JRST DC1
2973
2974  003132  200040  003021'  DC7:    MOVE P,SAVPDL           ;RESTORE PUSH DOWN POINTER
2975  003133  200240  003015'         MOVE T,INST
2976  003134  242240  000033         LSH T,33                ;PUSH BINARY INTO POSITION FOR OPEVAL
2977  003135  135100  003211'         LDB R,[POINT 3,T,8]
2978  003136  641240  700000         TLC T,700000
2979  003137  647240  700000         TLCN T,700000
2980  003140  137100  003212'         DPB R,[POINT 10,T,12]   ;ONLY DONE FOR IO INSTRUCTIONS
2981  003141  254000  000431'         JRST CPOPJ1             ;SYMBOL FOUND, SKIP RETURN
2982
2983  003142  660000  000002  DC8:    TRO F,ITF               ;SET INSTRUCTION TYPED FLAG
2984  003143  201245  000133         MOVEI T,133(T)
2985  003144  260040  002440'         PUSHJ P,TOUT            ;OUTPUT A LETTER
2986  003145  402000  002266'         CLEARM SPSAV            ;SO ED WONT TRY TO DELETE OP CODES
2987  003146  254000  003042'         JRST DC1
```

```
2988  003147                    LIT
2989  003147  302352' 000010
2990  003150  773767  002423
2991  003151  770767  000200
2992  003152  440770  003017'
2993  003153  003026' 000407'
2994  003154  542613  000000
2995  003155  000002  000002
2996  003156  000000  000134
2997  003157  000000  000000
2998  003160  420202  000000
2999  003161  000044  000000
3000  003162  410300  002247'
3001  003163  331170  001303'
3002  003164  221410  237000
3003  003165  260400  000006
3004  003166  002770  001303'
3005  003167  270400  001303'
3006  003170  270400  002414'

3007  003171  264000  001103'
3008  003172  264070  001130'
3009  003173  002371' 002302'
3010  003174  777000  000000
3011  003175  270400  002254'
3012  003176  331170  002254'
3013  003177  220470  002254'
3014  003200  777777  400000
3015  003201  320700  000005
3016  003202  270300  000005
3017  003203  330600  000005
3018  003204  220400  000005
3019  003205  004006  000000
3020  003206  000011  111111
3021  003207  340500  003014'
3022  003210  300600  003014'
3023  003211  330300  000005
3024  003212  271200  000005
3025
3026
3027  003220            PS:     BLOCK LPDL
3028
3029
3030  003270            DDTEND:           ;ONLY STARTING ADDRESS FOR FILDDT
3031                                      ;NO START ADDRESS FOR EXEC OR USER DDT
3032                                      ;BECAUSE MONITOR IS LOADED WITH BOTH EXEC AND USER DDTS
3033                                      ;BUT STILL WANTS TO BE STARTED AT ITS OWN START ADDRESS
3034                                      ;USER DDT IS LOADED LAST. - T.H.
3035                      IFN EDDT&10,<END DDT>
3036                              END
```

NO ERRORS DETECTED

PROGRAM BREAK IS 003270

SYMBOL TABLE

A		000002		ABSA		001047'		AC0		002332'			
AC17		002351'		ACCCF		000643'		ACCF		000642'			
ACCF1		000656'		AR		000011		ARM		002353'			
ASSEM		000622'		AUTOP		001426'		AUTOPI		002331'			
B		000003		B1ADR		002301'		B1CNT		002303'			
B1INS		001105'		B1SKP		002302'		BASE1		001056'			
BASECH		001052'		BCOM		001133'		BCOM2		001147'			
BCOM3		001145'		BDISP		002506'		BINC		000033		DLD	
BINRY	203000	000000	DLD	BITO		002551'		BITT		002566'			
BITT2		002573'		BITT4		002612'		BLKEND		000151'		DLD	
BNADR		002326'		BP1		001103'		BPLUP		001307'			
BPLUP1		001310'		BPN		001130'		BPS		001364'			
BPS1		001374'		BPS2		001400'		BPS3		001412'			
BPS4		001413'		BPS5		001423'		BREAK		001165'			
BREAK1		001214'		BREAK2		001215'		BTAB		003022'			
C		000004		CARR		000735'		CCF		010000		DLD	
CDR	011400	000000	INT	CF		000040	DLD	CF1		000400		DLD	
CHKADR		000432'	DLD	CHP		003016'		CLOC		000755		DLD	
CLOC.		000755	DLD	COMF		200000	DLD	CON		001041'			
CONSYM		001532'		CONTRO		000432'		CPOPJ		000432'			
CPOPJ1		000431'		CPUSHP		002414'		CR	015000	000000		INT	
CRF		002421'		CRN		002417'		CTF		000400		DLD	
DC	020000	000000	INT	DC1		003042'		DC2		003062'			
DC6		003116'		DC6AA		003120'		DC6BB		003125'			
DC7		003132'		DC8		003142'		DCSA	030000	000000		INT	
DCSB	030400	000000	INT	DD1		000022'		DD1.5		000023'			
DD2		000026'		DDT		000000'	INT	DDT0		000021'			
DDTEND		003270'	INT	DDTINT		000000		DDTVER		000022		DLD	
DDTX		000000'	INT	DECT		003071'		DECX		003077'			
DEF1		000543'		DEF2		000553'		DEF3		000561'			
DEF4		000602'		DEFIN		000540'		DEFV		002267'			
DELI		200000	DLD	DELO		400000	DLD	DEN		002257'			
DEP		001446'		DEPAC		001452'		DEPRA		000766'			
DEPRA2		000777'		DEPRS		001436'		DF	027000	000000		INT	
DIS	013000	000000	INT	DISP		002511'		DIVD		000620'			
DLS	024000	000000	INT	DOLLAR		000250'		DSK	017000	000000		INT	
DTC	032000	000000	INT	DTS	032400	000000	INT	DVF		040000		DLD	
EDDT		000001		EFFEC		001702'		EFFEC0		001756'			
EFFEC1		001760'		EFFEC2		001766'		EFFEC3		001775'			
EQUAL		001005'		ERR		000151'		ESTU		002261'			
ESTUT		002262'		EVAL		000407'		EVAL0		000410'			
EVAL1		000414'		EVAL2		000415'		EVAL3		000420'			
F		000000		FAF		001000	DLD	FCP		002232'			
FEF		400000	DLD	FETCH		001454'		FH		002264'			
FIR.		000025	DLD	FIRARG		001460'		FP1		002142'			
FP1A		002147'		FP3		002150'		FP3A		002161'			
FP4		002167'		FP4A		002171'		FP4B		002176'			
FP7		002205'		FP7A1		002212'		FP7B		002213'			
FPF		020000	DLD	FRASE		002252'		FRASE1		002256'			
FSGN		002235'		FSV		002263'		FT		002223'			
FT0		002232'		FT01		002231'		FT8		002217'			
FTOC		002045'		GLOBAL		040000	DLD	GO		001065'			
HALT	254200	000000	INT	HLFW		001611'		HLFW1		001617'			

SYMBOL TABLE

```
HWRDS           001036'                 ICON            000753'                 IJSA            001317'
IJSP            001333'                 IJSR            001323'                 IJSR2           001325'
IJSR3           001331'                 INDIRE          000636'                 INOUT           001647'
INSRT1          001340'                 INSRT3          001347'                 INSRTB          001337'
INST            003015'                 IPUSHJ          001313'                 ITF             000002  DLD
IXCT            001247'                 IXCT4           001245'                 IXCT5           001253'
IXCT6           001302'                 JEN     254500  000000  INT             JOBREL          000037  DLD
JORSYM          000036  DLD             JOV     255400  000000  INT             KILL            000502'
KILL1           000513'                 KILL2           000515'                 KILRET          000512'
L1              000041'                 L1RPR           000043'                 L2              000052'
L21             000065'                 L212            000104'                 L213            000106'
L4              000110'                 L5              000144'                 LASTB           000775  DLD
LCT             002424'                 LE1A            000266'                 LE2             000267'
LEAV            001303'                 LEAV1           001164'                 LET             000252'
LET1            000265'                 LF1             002000  DLD             LI0             000707'
LI1             000711'                 LI2             000720'                 LINEF           000706'
LIS             000030'                 LIS0            000033'                 LIS1            000035'
LIS2            000036'                 LISTEN          002456'                 LLOC            002271'
LLOC0           002272'                 LOADER          000151' DLD             LOCAL           100000  DLD

LOOK            001470'                 LOOK1           001477'                 LOOK2           001517'
LOOK3           001522'                 LPDL            000050                  LPRN            000626'
LPT     012400  000000  INT             LTF             000020  DLD             LWT             002254'
MASK            001672'                 MASK1           001700'                 MASK2           001675'
MDF     026000  000000  INT             MF              000002  DLD             MINUS           000623'
MLF             000200  DLD             MSK             002300'                 MTC     022000  000000  INT
MTM     023000  000000  INT             MTS     022400  000000  INT             MULT            000617'
NAF             000200  DLD             NBP             000010  DLD             NM1             000306'
NM1A            000312'                 NM1A1           000314'                 NM1B            000321'
NOMAT           003103'                 NOMAT1          003107'                 NUM             000234'
NUM1            000301'                 NWORD           001705'                 OCON            000737'
ODF             000012                  ODFM            002354'                 OPEVAL          003026'
OPTYPE          003032'                 OSYM            000740'                 OUTF            000004  DLD
P               000001                  PAD             001620'                 PAD1            001637'
PADS1           001624'                 PADS1A          001635'                 PADS3           001640'
PADS3A          001641'                 PADS3B          001644'                 PADSO           001622'
PERC            000251'                 PERIOD          000350'                 PI      000400  000000  INT
PI3A            001557'                 PI4             001561'                 PI7             001576'
PIB             001607'                 PID             000000  DLD             PILOC           000373'
PIN             001537'                 PLT     014000  000000  INT             PLUS            000624'
PNAME           740000  DLD             PNTR            003014'                 POW2            000343'
POWER           000326'                 POWF            004000  DLD             PRGM            002260'
PRNC            002251'                 PROC0           001231'                 PROC1           001237'
PROC2           001242'                 PROCD1          001227'                 PROCED          001222'
PRS             000004  DLD             PS              003220'                 PSYM            001006'
PTF             000100  DLD             PTP     010000  000000  INT             PTPP            000100  DLD
PTR     010400  000000  INT             PTRR            000104  DLD             PUNCH           000151  DLD
Q2F             000001  DLD             QF              000001  DLD             QUAN            000375'
QUAN1           000376'                 QUAN2           000377'                 QUANIN          000374'
QUFST1          000220'                 QUEST2          000223'                 QUEST4          000232'
QUESTN          000216'                 R               000002                  R20F            000010  DLD
R50PNT          001012'                 RELA            001044'                 REMOV1          001354'
REMOVB          001353'                 REMUN           000521'                 RESTOR          002401'
RESTR1          002402'                 RET             000164'                 ROF             000010  DLD
```

SYMBOL TABLE

RPRN		000664'		RPRN1		000703'		S		000003	
SAF		002000	DLD	SARS		002355'		SAV1		002365'	
SAV5		002374'		SAVE		002360'		SAVLOC		002273'	
SAVPDL		003021'		SAVPI		002275'		SAVTTY		002277'	
SBF		000020	DLD	SCH		000010		SCHM		002352'	
SEAR1		001722'		SEAR2		001730'		SEAR2A		001731'	
SEAR2B		001736'		SEAR3		001744'		SEAR4		001755'	
SEARFN		001742'		SEMIC		001063'		SET1		000605'	
SET2		000606'		SETNAM		000604'		SETUP		002000'	
SETUWP		000036		SF		000004	DLD	SFLOT		001037'	
SIXBI1		000457'		SIXBI2		000476'		SIXBIN		000460'	
SIXBP		001020'		SIXBP1		001022'		SLAS1		000751'	
SLASH		000743'		SPACE		000660'		SPSAV		002266'	
SPT		002103'		SPT0		002102'		SPT1		002104'	
SPT2		002110'		STF		004000	DLD	SUPTY0		000742'	
SVBT2		002617'		SVBT3		002616'		SVBTS		002614'	
SVBTS2		002615'		SYL		002253'		SYM		002265'	
SYMBOL		001040'		SYMD		002122'		SYMP		002274'	
T		000005		TAB		000757'		TAG		000530'	
TBL		002620'		TEM		002356'		TEM1		002357'	
TEM2		002255'		TEM3		002256'		TEXI		000433'	
TEXI2		000445'		TEXI3		000453'		TEXIN		002502'	
TEXO		001030'		TEXT		002240'		TEXT2		002242'	
TEXTT		002237'		TFLOT		002130'		TFLOT1		002140'	
TIF		100000	DLD	TIN		002430'		TMC	034000	000000	INT
TMS	034400	000000	INT	TOC		002045'		TOC0		002062'	
TOC4		002273'		TOC5		002077'		TOC6		002100'	
TOCA		002057'		TOCS		002056'		TOUT		002440'	
TSPC		002426'		TT		000013		TT1		000014	
TTY	012000	000000	INT	TTY1		002464'		TTYLEV		002474'	
TTYRET		002463'		TTYY		000120	DLD	TXT		003017'	
ULIM		001463'		ULIM1		001465'		ULIMIT		002270'	
UND1		000167'		UNDEF		000153'		UTC	021000	000000	INT
UTS	021400	000000	INT	VARRW		000731'		W		000004	
W1		000006		W2		000007		WORD		001734'	
WRD		002247'		WRD2		002250'		WRONG		000155'	
WRONG1		000156'		XEC		001066'		XEC0		001070'	
XEC1		001076'		ZERO		002011'		ZERO1		002021'	
ZEROR		002035'		ZLOW		000040	DLD	ZZ		000010	DLD
$1R		002331'	INT	$2R		002304'	INT	$3B		002307'	INT
$4R		002312'	INT	$5R		002315'	INT	$6B		002320'	INT
$7R		002323'	INT	$8R		002326'	INT	$I		002275'	INT
$M		002300'	INT	%A		000205	DLD	%AC1		000571	DLD
%AC2		000575	DLD	%ACBM		000562	DLD	%ACCP		000464	DLD
%AN		000614	DLD	%AO		000354	DLD	%AOB		000375	DLD
%B		000217	DLD	%B2		000646	DLD	%BD		000025	DLD
%BIMR		000277	DLD	%BOOLE		000612	DLD	%BPUT		000132	DLD
%BYTF		000221	DLD	%C		000462	DLD	%CA		000652	DLD
%CA1		000502	DLD	%CAL		000705	DLD	%CB		000664	DLD
%CM		000656	DLD	%COMP		000520	DLD	%DP		000252	DLD
%DV		000333	DLD	%DV1		000336	DLD	%DV2		000340	DLD
%E		000523	DLD	%EAN		000605	DLD	%EIMS		000307	DLD
%F		000135	DLD	%FLO		000166	DLD	%H1		000342	DLD

SYMBOL TABLE

%HAK	000255	DLD	%HW1	000546	DLD	%HW2	000544	DLD
%HW3	000553	DLD	%HW4	000552	DLD	%HWT	000536	DLD
%I	000302	DLD	%IMB	000300	DLD	%IMS	000316	DLD
%IO	000036	DLD	%J	000433	DLD	%JP	000400	DLD
%JS	000474	DLD	%L	000213	DLD	%LD	000245	DLD
%LMB	000211	DLD	%M	000215	DLD	%ML	000324	DLD
%ML1	000327	DLD	%MO	000306	DLD	%MV	000257	DLD
%N	000532	DLD	%O	000151	DLD	%OI	000041	DLD
%OR	000625	DLD	%P	000241	DLD	%PA	000425	DLD
%PD	000410	DLD	%POP	000436	DLD	%PSJ	000517	DLD
%PU	000402	DLD	%PUS	000432	DLD	%R	000416	DLD
%S	000322	DLD	%S1	000460	DLD	%S2	000451	DLD
%S3	000452	DLD	%SH	000441	DLD	%SJ	000507	DLD
%ST	000643	DLD	%T	000351	DLD	%TA	000660	DLD
%TM	000654	DLD	%U40	000121	DLD	%U50	000676	DLD
%U60	000121	DLD	%U603	000142	DLD	%U6062	000155	DLD
%U62	000161	DLD	%U703	000727	DLD	%USET	000072	DLD
%UUO	000044	DLD	%Z	000153	DLD			

	2817	2818#	2818	2819#	2819	2820#	2820	2821#	2821	2822#	2822	2823#	2823	2824#
	2824	2825#	2825	2826#	2826	2827#	2827	2828#	2828	2829#	2829	2830#	2830	2831#
	2831	2832#	2832	2833#	2833	2834#	2834	2835#	2835	2836#	2836	2837#	2837	2838#
	2838	2839#	2839	2840#	2840	2841#	2841	2842#	2842	2843#	2843	2844#	2844	2845#
	2845	2846#	2846	2847#	2847	2848#	2848	2849#	2849	2850#	2850	2851#	2851	2852#
	2852	2853#	2853	2854#	2854	2855#	2855	2856#	2856	2857#	2857	2858#	2858	2859#
	2859	2860#	2860	2861#	2861	2862#	2862	2863#	2863	2864#	2864	2865#	2865	2866#
	2866	2867#	2867	2868#	2868	2869#	2869	2870#	2870	2871#	2871	2872#	2872	2873#
	2873	2874#	2874	2875#	2875	2876#	2876	2877#	2877	2878#	2878	2882		
BIT0	2310	2320#												
BITT	2320	2334#												
BITT2	2339#	2353												
BITT4	2349	2355#												
BLKEND	2281#	2308												
BNADR	1245	1289	1869#											
BP1	1040#	1077	1085	1230	1231	1234								
BPLUP	1166	1200#	1218											
BPLUP1	1147	1201#												
BPN	1078#	1239												
BPS	1259#	2306												
BPS1	1260	1268#												
BPS2	1262	1272#												
BPS3	1269	1283#												
BPS4	1284#	1290												
BPS5	1281	1287	1292#											
BREAK	1093	1108#												
BREAK1	1127	1131#												
BREAK2	1114	1132#												
BTAB	2895#	2911	2933											
C	134#	1741	1751	1759	1766	1798	1799	1800						
CARR	484	915#	2288											
CCF	162#	414	652	660	736	1012	1301	1638						
CDR	239#	239												
CF	161#	414	429	548	591	652	659	697	760	949				
CF1	183#	407	923	924	964	965	1445							
CHKADR	1396#													
CHP	420	582	2891#	2901	2908	2940	2947	2952	2968					
CLOC	2652#	2755	2755#	2756	2756#	2757	2757#	2758	2758#	2759	2759#	2760	2762#	2761
	2761#	2762	2762#	2763	2763#	2764	2764#	2765	2765#	2766	2766#	2767	2767#	2768
	2768#	2769	2769#	2770	2770#	2771	2771#	2772	2772#	2773	2773#	2774	2774#	2775
	2775#	2776	2776#	2777	2777#	2778	2778#	2779	2779#	2780	2780#	2781	2781#	2782
	2782#	2783	2783#	2784	2784#	2785	2785#	2786	2786#	2787	2787#	2788	2788#	2789
	2789#	2790	2790#	2791	2791#	2792	2792#	2793	2793#	2794	2794#	2795	2795#	2796
	2796#	2797	2797#	2798	2798#	2799	2799#	2800	2800#	2801	2801#	2802	2802#	2803
	2803#	2804	2804#	2805	2805#	2806	2806#	2807	2807#	2808	2808#	2809	2809#	2810
	2810#	2811	2811#	2812	2812#	2813	2813#	2814	2814#	2815	2815#	2816	2816#	2817
	2817#	2818	2818#	2819	2819#	2820	2820#	2821	2821#	2822	2822#	2823	2823#	2824
	2824#	2825	2825#	2826	2826#	2827	2827#	2828	2828#	2829	2829#	2830	2830#	2831
	2831#	2832	2832#	2833	2833#	2834	2834#	2835	2835#	2836	2836#	2837	2837#	2838
	2838#	2839	2839#	2840	2840#	2841	2841#	2842	2842#	2843	2843#	2844	2844#	2845
	2845#	2846	2846#	2847	2847#	2848	2848#	2849	2849#	2850	2850#	2851	2851#	2852
	2852#	2853	2853#	2854	2854#	2855	2855#	2856	2856#	2857	2857#	2858	2858#	2859
	2859#	2860	2860#	2861	2861#	2862	2862#	2863	2863#	2864	2864#	2865	2865#	2866

	1011	1014	1131	1132	1134	1230	1231	1234	1235	1238	1239	1245	1246	1247
	1250	1251	1298	1299	1300	1302	1409	1410	1437	1641	1643	1654		
SAF	168#	408	1400	1570										
SARS	1882#	2026	2046	2066										
SAV1	2027	2030#												
SAV5	2029	2038#												
SAVF	269	449	452	1035	1108	1153	1201	2025#	2028	2036	2051	2054	2056	2070
SAVLOC	931	943	948	951	1836#									
SAVPDL	2893#	2902	2957	2963	2974									
SAVPI	229	647	1121	1215	1224	1838#	2031	2032	2037	2055	2058	2060	2069	
SAVTTY	1840#	2125	2127	2130										
SBF	187#	407												
SCH	138#	401	997	1011	1018	1444	2049	2331						
SCHM	401	1011	1878#	2049										
SEAR1	1574#	1585												
SEAR2	1562	1580#	1621											
SEAR2A	1581#	1590	1602											
SEAR2B	1575	1587#												
SEAR3	1580	1593#												
SEAR4	1602#	1617												
SEARFN	1584	1591#												
SEMIC	1017#	2303												
SET1	803#	810												
SET2	804#	808												
SETNAM	761	802#												
SETUP	1572	1622#	1640											
SETUWP	230#													
SF	158#	414	458	562	563	574	628	633	651	868	871			
SFLOT	994#	2307												
SIXBI1	711#	725												
SIXBI2	713	726#												
SIXBIN	698	712#												
SIXBP	977#	990												
SIXBP1	979#	983												
SLAS1	929	933#	939											
SLASH	927#	2299												
SPACE	471	849	851	853#	2294									
SPSAV	1700	1718	1831#	2986										
SPT	543	975	1701#											
SPT0	1433	1518	1700#											
SPT1	1703#	1706												
SPT2	1705	1707#												
STF	166#	407	408	888	897	926	927	1596						
SUPTY0	926#	2295												
SVBT2	2335	2348	2361#											
SVBT3	2337	2343	2352	2360#										
SVBTS	2325	2340	2358#											
SVBTS2	2330	2334	2359#											
SYL	415	443	550	553	569	570	625	630	650	691	701	712	870	1818#
SYM	416	522	575	581	583	673	779	784	809	1830#				
SYMBOL	995#	2311												
SYMD	1717#	2306												
SYMP	273	664	670	769	802	1408	1409	1436	1627	1641	1642	1837#		

2815#	2816#	2817#	2819	2821	2822	2823	2823#	2824#	2824	2826#	2827	2827#	2828
2829	2830	2831#	2832	2832#	2833#	2834	2834#	2835	2835#	2836#	2837	2838	2839#
2840#	2840	2841	2842	2842#	2843#	2844	2844#	2845#	2845	2846#	2848#	2849#	2852
2850#	2851	2851#	2852	2852#	2854	2856	2857	2857#	2859	2860#	2861	2862	2863
2864	2866	2866#	2867#	2868	2870	2871#	2872	2873	2874#	2875#	2877		