```
*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***


             *     *    *     ****** ******   *****             *
             **   **   * *   *       *     * *     *           **
             * * * *  *****  *       ******  *     *         ** *
             * *** * *     * *       *  **   *     *    *       *
             *  *  * *     *  ****** *    **  *****     *    *******
                                                       *
        *    ******              ***                          *****   *****
       **          *              *                          *     * *   ***
     ** *       ***  *******      *   *   *  * ****  *******  *****  *  *  *
        *          *         *    *   *   *  **    *         *     * ***   *
     ******* ******           ****    *****  *     *          *****   *****

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***

*** START MACRO,1(517) - INPUT LINE IS IN L$ START ***
```

```
10 LINEINPUT ">";L$
20 GOSUB 1000
30 PRINT "ADD: "AD$
40 PRINT "OPC: "OD$
50 PRINT "FL1: "F1$
60 PRINT "FL2: "F2$
65 IF EP THEN PRINT "A parsing error was detected"
70 GOTO 10
1000 'INPUT LINE IS IN L$
1005 'OBJECT IS TO DIVIDE UP INTO:
1010 ' AD$=ADDRESS STRING
1015 ' OC$=OPERATOR CODE
1020 ' F1$=FIRST FIELD
1025 ' F2$=SECOND FIELD
1030 'COMMENTS WILL BE IGNORED
1035 'FIELDS WILL BE SET UP AS FOLLOWS:
1040 'ADDRESS: OPCODE FIRST,LAST ;COMMENT
1045 'ANY BLANK FIELD WILL RETURN NULL
1050 'FIRST WILL CLEAN UP BLANKS
1060 GOSUB 2000
1065 'PARSE THROUGH THE WHOLE THING
1066 AD$="":OD$="":F1$="":F2$="":EP=0
1070 P1=1
1080 GOSUB 3000
1090 IF PT$="" AND PC$="" THEN RETURN
1100 IF PT$<>":" THEN 1200
1110 IF PC$="" THEN EP=-1:RETURN
1120 AD$=PC$
1130 IF MID$(L$,P1,1)=" " THEN P1=P1+1
1140 GOSUB 3000
1200 IF PT$="" AND PC$="" THEN RETURN
1210 IF PT$="" THEN OD$=PC$:RETURN
1220 IF PT$<>" " AND PT$<>";" THEN 1300
1230 OD$=PC$
1235 IF PT$=";" THEN RETURN
1240 GOSUB 3000
1300 IF PT$="" AND PC$="" THEN RETURN
1310 IF PT$="" THEN F1$=PC$:RETURN
1320 IF PT$<>"," AND PT$<>";"  THEN 1400
1325 IF MID$(L$,P1,1)=" " THEN P1=P1+1
1330 F1$=PC$
1335 IF PT$=";" THEN RETURN
1340 GOSUB 3000
1400 IF PT$="" AND PC$="" THEN RETURN
1410 IF PT$="" THEN F2$=PC$:RETURN
1430 F2$=PC$
1500 GOSUB 3000:IF PT$=" " OR PT$="" OR PT$=";" THEN RETU +
 RN ELSE EP=-1:RETURN
2000 'CLEAN UP BLANKS
2005 CI$=CHR$(9):DS$="  ":SS$=" ":NL$=""
2010 IF L$=NL$ THEN 2045
2015 CF=INSTR(L$,CI$):IF CF<>0 THEN MID$(L$,CF,1)=SS$:GOT +
 O 2015
```

```
2020 CF=INSTR(L$,DS$):IF CF<>0 THEN IF CF=1 THEN L$=MID$( +
 L$,2):GOTO 2020
       ELSE L$=LEFT$(L$,CF-1)+MID$(L$,CF+1):GOTO 2020
2025 IF L$=NL$ THEN 2045
2030 IF LEFT$(L$,1)=SS$ THEN L$=MID$(L$,2)
2035 IF L$=NL$ THEN 2045
2040 IF RIGHT$(L$,1)=SS$ THEN L$=LEFT$(L$,LEN(L$)-1)
2041 PRINT L$
2045 RETURN
3000 'GO THROUGH FILE UNTIL NON ALPHA-NUMERIC
3010 'GO FROM POSITION P1, INCREMENTING AS YOU GO ALONG
3020 'PLACE PARSED TEXT IN PC$
3030 'PLACE TERMINATING CHARACTER IN PT$
3035 PT$="":PC$=""
3040 PR$=MID$(L$,P1,1):P1=P1+1
3060 IF PR$>="A" AND PR$<="Z" THEN 3090
3070 IF PR$>="a" AND PR$<="z" THEN 3090
3080 IF PR$="$" OR PR$="." OR PR$="%" THEN 3090
3085 GOTO 3500
3090 PC$=PC$+PR$
3100 GOTO 3040
3500 PT$=PR$:RETURN
50000 'DATA MNEMONIC,BYTES,REGTYPE,MULTYPE,MASK OCTAL
50005 'MNEMONIC IS AN ASCII STRING FROM 1-6 CHARS LONG
50010 'BYTES IS THE NUMBER OF BYTES
50015 'REGTYPE SPECIFIES THEN REGISTER POSITIONING
50020 ' AND TYPE OF POSITIONING
50025 'MULTYPE SPECIFIES THE SECOND OR THIRD BYTE COMPOSI +
 TION
50030 'MASK OCTAL IS THE ACTUAL INSTRUCTION
50035 '
50040 'REGTYPE IS DEFINED AS FOLLOWS
50045 ' 1 = EXP TYPE, OCTAL DIGIT,  BINARY=NN EXP NNN
50050 ' 2 = DESINATION REGISTER TYPE BINARY=NN DDD NNN
50055 ' 3 = REGISTER PART TYPE BINARY=NN rpN NNN
50060 ' 4 = SOURCE/DEST TYPE BINARY=NN DDD SSS
50065 ' 5 = X TYPE DESINATION BINARY=NN NXN NNN
50070 ' 6 = SOURCE TYPE BINARY=NN NNN SSS
50075 'MULTYPE IS DEFINED AS FOLLOWS
50080 ' 1 = IMMEDIATE DATA
50085 ' 2 = MEMORY LOCATION
50090 '
50095 'I/O INSTRUCTIONS
50100 DATA IN,2,0,1,333
50105 DATA OUT,2,0,1,323
50110 'INTERUPT INSTRUCTIONS
50115 DATA EI,1,0,0,373
50120 DATA DI,1,0,0,363
50125 DATA HLT,1,0,0,166
50130 DATA RST,1,1,0,307
50135 'CARRY BIT INSTRUCTIONS
50140 DATA CMC,1,0,0,077
50145 DATA STC,1,0,0,067
50150 'NO OPERATION INSTRUCTION
```

```
50155 DATA NOP,1,0,0,000
50160 'SINGLE REGISTER INSTUCTIONS
50165 DATA INR,1,2,0,004
50170 DATA DCR,1,2,0,005
50175 DATA CMA,1,0,0,057
50180 DATA DAA,1,0,0,047
50185 'REGISTER PAIR INSTRUCTIONS
50190 DATA PUSH,1,3,0,305
50195 DATA POP,1,3,0,301
50200 DATA DAD,1,3,0,011
50205 DATA INX,1,3,0,003
50210 DATA DCX,1,3,0,013
50215 DATA XCHG,1,0,0,353
50220 DATA XTHL,1,0,0,343
50225 DATA SPHL,1,0,0,371
50230 'ROTATE ACCUMULATOR INSTRUCTIONS
50235 DATA RLC,1,0,0,007
50240 DATA RRC,1,0,0,017
50245 DATA RAL,1,0,0,027
50250 DATA RAR,1,0,0,037
50255 'DATA TRANSFER INSTUCTIONS
50260 DATA MOV,1,4,0,100
50265 DATA STAX,1,5,0,002
50270 DATA LDAX,1,5,0,012
50275 'REG/MEM TO ACC TRANSFERS
50280 DATA ADD,1,6,0,200
50285 DATA ADC,1,6,0,210
50290 DATA SUB,1,6,0,220
50295 DATA SBB,1,6,0,230
50300 DATA ANA,1,6,0,240
50305 DATA XRA,1,6,0,250
50310 DATA ORA,1,6,0,260
50315 DATA CMP,1,6,0,270
50320 'DIRECT ADDRESS INSTUCTIONS
50325 DATA STA,3,0,2,062
50330 DATA LDA,3,0,2,072
50335 DATA SHLD,3,0,2,042
50340 DATA LHLD,3,0,2,052
50345 'IMMEDIATE INSTRUCTIONS
50350 DATA LXI,3,3,2,001
50355 DATA MVI,2,2,1,006
50360 DATA ADI,2,0,1,306
50365 DATA ACI,2,0,1,316
50370 DATA SUI,2,0,1,326
50375 DATA SBI,2,0,1,336
50380 DATA ANI,2,0,1,346
50385 DATA XRI,2,0,1,356
50390 DATA ORI,2,0,1,366
50395 DATA CPI,2,0,1,376
50400 'JUMP INSTRUCTIONS
50405 DATA PCHL,1,0,0,351
50410 DATA JMP,3,0,2,303
50415 DATA JC,3,0,2,332
50420 DATA JNC,3,0,2,322
```

```
50425 DATA JZ,3,0,2,312
50430 DATA JNZ,3,0,2,302
50435 DATA JM,3,0,2,372
50440 DATA JP,3,0,2,362
50445 DATA JPE,3,0,2,352
50450 DATA JPO,3,0,2,342
50455 'CALL INSTRUCTIONS
50460 DATA CALL,3,0,2,315
50465 DATA CC,3,0,2,334
50470 DATA CNC,3,0,2,324
50475 DATA CZ,3,0,2,314
50480 DATA CNZ,3,0,2,304
50485 DATA CM,3,0,2,374
50490 DATA CP,3,0,2,364
50495 DATA CPE,3,0,2,354
50500 DATA CPO,3,0,2,344
50505 'RETURN INSTRUCTIONS
50510 DATA RET,1,0,0,311
50515 DATA RC,1,0,0,330
50520 DATA RNC,1,0,0,320
50525 DATA RZ,1,0,0,310
50530 DATA RNZ,1,0,0,300
50535 DATA RM,1,0,0,370
50540 DATA RP,1,0,0,360
50545 DATA RPE,1,0,0,350
50550 DATA RPO,1,0,0,340
50555 'DONE
50560 DATA END,0,0,0,000
```

```
Total of 4 pages + 2 banner
*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***


             *     *    *     ****** ******   *****             *
             **   **   * *   *       *     * *     *           **
             * * * *  *****  *       ******  *     *         ** *
             * *** * *     * *       *  **   *     *    *       *
             *  *  * *     * ****** *   **   *****    *    *******
                                                       *
        *    ******              ***                          *****   *****
       **          *              *                          *     * *   ***
     ** *       ***  *******      *   *   *  * ****  *******  *****  *  *  *
        *          *         *    *   *   *  **    *         *     * ***   *
     ******* ******           ****    *****  *     *          *****   *****

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***

*** END MACRO,1(517) - INPUT LINE IS IN L$ END ***
```