```
                                        ;AT EXIT (HL)+=(HL)+4
001243' 001000  000136          MOVRM:  MOV     E,M             ;GET LO
001244' 001000  000043                  INX     H               ;POINT TO MO
001245' 001000  000126                  MOV     D,M             ;GET MO
001246' 001000  000043                  INX     H               ;POINT TO MO
001247' 001000  000116                  MOV     C,M             ;GET MO
001250' 001000  000043                  INX     H               ;POINT TO EXPONENT
001251' 001000  000106                  MOV     B,M             ;GET EXPONENT
001252' 001000  000043          INXHRT: INX     H               ;INC POINTER TO BEGINNING OF NEXT NUMBER
001253' 001000  000311                  RET                     ;ALL DONE


                                        ;MOVE NUMBER FROM FAC TO MEMORY [(HL)]
                                        ;ALTERS A,B,D,E,H,L
001254' 001000  000021          MOVMF:  LXI     D,FACLO         ;GET POINTER TO FAC
001255' 000000  001241*
001256' 000000  001241'
                                        ;FALL INTO MOVE


                                        ;MOVE NUMBER FROM (DE) TO (HL)
                                        ;ALTERS A,B,D,E,H,L
                                        ;EXITS WITH (DE)+=(DE)+4, (HL)+=(HL)+4
001257' 001000  000006          MOVE:   MVI     B,4             ;SET COUNTER
001260' 000000  000004
                                IFE     LENGTH-2,<
                                        XWD     1000,076        ;"MVI   A" OVER NEXT BYTE
                                MOVVFM: XCHG>                   ;MOVE NUMBERS INTO THE FAC
001261' 001000  000032          MOVE1:  LDAX    D               ;GET WORD, ENTRY FROM VMOVMF
001262' 001000  000167                  MOV     M,A             ;PUT IT WHERE IT BELONGS
001263' 001000  000023                  INX     D               ;INCREMENT POINTERS TO NEXT WORD
001264' 001000  000043                  INX     H
001265' 001000  000005                  DCR     B               ;SEE IF DONE
001266' 001000  000302                  JNZ     MOVE1
001267' 000000  001261'
001270' 000000  001255'
001271' 001000  000311                  RET


                                        ;UNPACK THE FAC AND THE REGISTERS
                                        ;ALTERS A,C,H,L
                                        ;WHEN THE NUMBER IN THE FAC IS UNACKED, THE ASSUMED ONE IN THE
                                        ;MANTISSA IS RESTORED, AND THE COMPLEMENT OF THE SIGN IS PLACED
                                        ;IN FAC+1
001272' 001000  000041          UNPACK: LXI     H,FAC-1         ;POINT TO MO AND SIGN
001273' 777777  777777*
001274' 000000  001267'
001275' 001000  000176                  MOV     A,M             ;GET MO AND SIGN
001276' 001000  000365                  PUSH    PSW             ;SAVE SIGN
001277' 001000  000366                  ORI     200             ;RESTORE THE HIDDEN ONE
001300' 000000  000200
001301' 001000  000167                  MOV     M,A             ;SAVE MO
001302' 001000  000361                  POP     PSW             ;GET SIGN
```



```
001303' 001000  000256          XRA     M               ;GET COMPLEMENT OF SIGN IN MSB
001304' 001000  000043          INX     H               ;POINT TO TEMPORARY SIGN BYTE
001305' 001000  000043          INX     H
001306' 001000  000167          MOV     M,A             ;SAVE COMPLEMENT OF SIGN
001307' 001000  000171          MOV     A,C             ;GET MO AND SIGN OF THE REGISTERS
001310' 001000  000365          PUSH    PSW             ;SAVE SIGN
001311' 001000  000366          ORI     200             ;RESTORE THE HIDDEN ONE
001312' 000000  000200
001313' 001000  000117          MOV     C,A             ;SAVE THE MO
001314' 001000  000361          POP     PSW             ;GET THE SIGN BACK
001315' 001000  000256          XRA     M               ;COMPARE SIGN OF FAC AND SIGN OF REGISTERS
001316' 001000  000311          RET                     ;ALL DONE


                                IFE     LENGTH-2,<
                                REPEAT  0,<             ;VPUSHF WILL BE IN-LINE IN F3
                                        ;PUT ANY TYPE VALUE ON THE STACK FROM THE FAC
                                        ;STRINGS ARE TREATED AS INTEGERS
                                        ;ALTERS A,B,C,H,L
                                VPUSHF: LDA     VALTYP          ;GET THE VALUE TYPE
                                        CPI     4               ;SET FLAGS ACCORDING TO VALTYP
                                        LXI     H,FACLO         ;GET POINTER TO LO IN FAC
                                        PUSHM                   ;PUSH FACLO+0,1 ON THE STACK
                                        JM      VPUSHD          ;RETURN IF THE DATA WAS AN INTEGER OR A STRING
                                        PUSHM                   ;PUSH FAC+1,0 ON THE STACK
                                        JZ      VPUSHD          ;RETURN IF WE HAD A SINGLE PRECISION NUMBER
                                        LXI     D,DFACLO        ;WE HAVE A DOUBLE PRECISON NUMBER
                                        PUSHM                   ;PUSH ITS 4 LO BYTES ON THE STACK
                                        PUSHM
                                VPUSHD:>                        ;ALL DONE


                                        ;MOVE ANY TYPE VALUE FROM MEMORY [(HL)] TO FAC
                                        ;ALTERS A,B,D,E,H,L
                                VMOVFA: LXI     H,ARGLO         ;ENTRY FROM DADD, MOVE ARG TO FAC
                                VMOVFM: LXI     D,MOVVFM        ;GET ADDRESS OF LOCATION THAT DOES
                                        JMP     VMVVFM          ; AN "XCHG" AND FALLS INTO MOVE1


                                        ;MOVE ANY TYPE VALUE FROM FAC TO MEMORY [(HL)]
                                        ;ALTERS A,B,D,E,H,L
                                VMOVAF: LXI     H,ARGLO         ;ENTRY FROM FIN, DMUL10, DDIV10
                                        ;MOVE FAC TO ARG
                                VMOVMF: LXI     D,MOVE1         ;GET ADDRESS OF MOVE SUBROUTINE
                                VMVVFM: PUSH    D               ;SHOVE IT ON THE STACK
                                        LXI     D,FACLO         ;GET FIRST ADDRESS FOR INT, SNG
                                        LDA     VALTYP          ;GET THE VALUE TYPE
                                        ANI     177             ;STRINGS LOOK LIKE REALS
                                        MOV     B,A             ;SET UP THE COUNT
                                        CPI     10              ;DO WE HAVE DBL?
                                        RNZ                     ;WE DO NOT, GO DO THE MOVE
                                        LXI     D,DFACLO        ;WE DO, GET LO ADDR OF THE DBL NUMBER
                                        RET>                    ;GO DO THE MOVE
```

```
 1324                                   19980   PAGE
```



```
 1325                                   20000   SUBTTL  COMPARE TWO NUMBERS
 1326                                   20020           ;COMPARE TWO SINGLE PRECISION NUMBERS
 1327                                   20040           ;A=1 IF ARG ,LT, FAC
 1328                                   20060           ;A=0 IF ARG=FAC
 1329                                   20080           ;A=-1 IF ARG ,GT, FAC
 1330                                   20100           ;DOREL DEPENDS UPON THE FACT THAT FCOMP RETURNS WITH CARRY ON
 1331                                   20120           ; IFF A HAS 377
 1332                                   20140           ;ALTERS A,H,L
 1333  001317' 001000  000170           20160   FCOMP:  MOV     A,B             ;CHECK IF ARG IS ZERO
 1334  001320' 001000  000267           20180           ORA     A
 1335  001321' 001000  000512           20200           JZ      SIGN
 1336  001322' 000000  000000*
 1337  001323' 000000  001273'
 1338  001324' 001000  000041           20220           LXI     H,FCOMPS        ;WE JUMP TO FCOMPS WHEN WE ARE DONE
 1339  001325' 000000  001141'
 1340  001326' 000000  001322'
 1341  001327' 001000  000345           20240           PUSH    H               ;PUT THE ADDRESS ON THE STACK
 1342  001330' 001000  000357           20260           FSIGN                   ;CHECK IF FAC IS ZERO
 1343  001331' 001000  000171           20280           MOV     A,C             ;IF IT IS, RESULT IS MINUS THE SIGN OF ARG
 1344  001332' 001000  000510           20300           RZ                      ;IT IS
 1345  001333' 001000  000041           20320           LXI     H,FAC+1         ;POINT TO SIGN OF FAC
 1346  001334' 777777  777777*
 1347  001335' 000000  001325'
 1348  001336' 001000  000256           20340           XRA     M               ;SEE IF THE SIGNS ARE THE SAME
 1349  001337' 001000  000171           20360           MOV     A,C             ;IF THEY ARE DIFFERENT, RESULT IS SIGN OF ARG
 1350  001340' 001000  000370           20380           RM                      ;THEY ARE DIFFERENT
 1351  001341' 001000  003315           20400           CALL    FCOMP2          ;CHECK THE REST OF THE NUMBER
 1352  001342' 000000  001347'
 1353  001343' 000000  001334'
 1354  001344' 001000  000037           20420   FCOMPD: RAR                     ;NUMBERS ARE DIFFERENT, CHANGE SIGN IF
 1355  001345' 001000  000251           20440           XRA     C               ; BOTH NUMBERS ARE NEGATIVE
 1356  001346' 001000  000311           20460           RET                     ;GO SET UP A
 1357
 1358  001347' 001000  000043           20500   FCOMP2: INX     H               ;POINT TO EXPONENT
 1359  001350' 001000  000170           20520           MOV     A,B             ;GET EXPONENT OF ARG
 1360  001351' 001000  000276           20540           CMP     M               ;COMPARE THE TWO
 1361  001352' 001000  000300           20560           RNZ                     ;NUMBERS ARE DIFFERENT
 1362  001353' 001000  000053           20580           DCX     H               ;POINT TO MO
 1363  001354' 001000  000171           20600           MOV     A,C             ;GET MO OF ARG
 1364  001355' 001000  000276           20620           CMP     M               ;COMPARE WITH MO OF FAC
 1365  001356' 001000  000300           20640           RNZ                     ;THEY ARE DIFFERENT
 1366  001357' 001000  000053           20660           DCX     H               ;POINT TO MO OF FAC
 1367  001360' 001000  000172           20680           MOV     A,D             ;GET MO OF ARG
 1368  001361' 001000  000276           20700           CMP     M               ;COMPARE WITH MO OF FAC
 1369  001362' 001000  000300           20720           RNZ                     ;THE NUMBERS ARE DIFFERENT
 1370  001363' 001000  000053           20740           DCX     H               ;POINT TO LO OF FAC
 1371  001364' 001000  000173           20760           MOV     A,E             ;GET LO OF ARG
 1372  001365' 001000  000226           20780           SUB     M               ;SUBTRACT LO OF FAC
 1373  001366' 001000  000300           20800           RNZ                     ;NUMBERS ARE DIFFERENT
 1374  001367' 001000  000341           20820           POP     H               ;NUMBERS ARE THE SAME, DON'T SCREW UP STACK
 1375  001370' 001000  000341           20840           POP     H
 1376  001371' 001000  000311           20860           RET                     ;ALL DONE
 1377
```

```
                        20920   IFE     LENGTH-2,<
                        20940           ;COMPARE TWO INTEGERS
                        20960           ;A=1 IF (DE) ,LT, (HL)
                        20980           ;A=0 IF (DE)=(HL)
                        21000           ;A=-1 IF (DE) ,GT, (HL)
                        21020           ;ALTERS A ONLY
                        21040   ICOMP:  MOV     A,D             ;ARE THE SIGNS THE SAME?
                        21060           XRA     H
                        21080           MOV     A,H             ;IF NOT, ANSWER IS THE SIGN OF (HL)
                        21100           JM      ICOMPS          ;THEY ARE DIFFERENT
                        21120           CMP     D               ;THEY ARE THE SAME, COMPARE THE HO'S
                        21140           JNZ     SIGNS           ;GO SET UP A
                        21160           MOV     A,L             ;COMPARE THE LO'S
                        21180           SUB     E
                        21200           JNZ     SIGNS           ;GO SET UP A
                        21220           RET                     ;ALL DONE, THEY ARE THE SAME
                        21240
                        21260
                        21280           ;COMPARE TWO DOUBLE PRECISION NUMBERS
                        21300           ;A=1 IF ARG ,LT, FAC
                        21320           ;A=0 IF ARG=FAC
                        21340           ;A=-1 IF ARG ,GT, FAC
                        21360           ;ALTERS A,B,C,D,E,H,L
                        21380   DCOMPD: LXI     H,ARGLO         ;ENTRY WITH POINTER TO ARG IN (DE)
                        21400           MVI     B,10            ;SET UP COUNT TO MOVE DBL NUMBERS
                        21420           CALL    MOVE1           ;MOVE THE ARGUMENT INTO ARG
                        21440   DCOMP:  LXI     D,ARG           ;GET POINTER TO ARG
                        21460           LDAX    D               ;SEE IF ARG=0
                        21480           ORA     A
                        21500           JZ      SIGN            ;ARG=0, GO SET UP A
                        21520           LXI     H,FCOMPS        ;PUSH FCOMPS ON STACK SO WE WILL RETURN TO
                        21540           PUSH    H               ; TO IT AND SET UP A
                        21560           FSIGN                   ;SEE IF FAC=0
                        21580           DCX     D               ;POINT TO SIGN OF ARG
                        21600           LDAX    D               ;GET SIGN OF ARG
                        21620           MOV     C,A             ;SAVE IT FOR LATER
                        21640           RZ                      ;FAC=0, SIGN OF RESULT IS SIGN OF ARG
                        21660           LXI     H,FAC-1         ;POINT TO SIGN OF FAC
                        21680           XRA     M               ;SEE IF THE SIGNS ARE THE SAME
                        21700           MOV     A,C             ;IF THEY ARE, GET THE SIGN OF THE NUMBERS
                        21720           RM                      ;THE SIGNS ARE DIFFERENT, GO SET A
                        21740           INX     D               ;POINT BACK TO EXPONENT OF ARG
                        21760           INX     H               ;POINT TO EXPONENT OF FAC
                        21780           MVI     B,10            ;SET UP A COUNT
                        21800   DCOMP1: LDAX    D               ;GET A BYTE FROM ARG
                        21820           SUB     M               ;COMPARE IT WITH THE FAC
                        21840           JNZ     FCOMPD          ;THEY ARE DIFFERENT, GO SET UP A
                        21860           DCX     D               ;THEY ARE THE SAME, EXAMINE THE NEXT LOWER
                        21880           DCX     H               ; ORDER BYTES
                        21900           DCR     B               ;ARE WE DONE?
                        21920           JNZ     DCOMP1          ;NO, COMPARE THE NEXT BYTES
                        21940           POP     B               ;THEY ARE THE SAME, GET FCOMPS OFF STACK
```



```
                        21960           RET>                    ;ALL DONE
                        21980   PAGE
```

```
1433                    22000  SUBTTL  CONVERSION ROUTINES BETWEEN INTEGER, SINGLE AND DOUBLE PRECISION
1434                    22020  IFE     LENGTH-2,<
1435                    22040          ;FORCE THE FAC TO BE AN INTEGER
1436                    22060          ;ALTERS A,B,C,D,E,H,L
1437                    22080  FRCINT: LDA     VALTYP          ;SEE WHAT WE HAVE
1438                    22100          CPI     4
1439                    22120          LHLD    FACLO           ;GET FACLO+0,1 IN CASE WE HAVE AN INTEGER
1440                    22140          RC                      ;WE HAVE AN INTEGER, ALL DONE
1441                    22160          JM      TMERR           ;WE HAVE A STRING, THAT IS A "NO-NO"
1442                    22180          CNZ     CONSD           ;IF WE HAVE A DBL, CONVERT IT TO A SNG
1443                    22200          LXI     H,OVERR         ;PUT OVERR ON THE STACK SO WE WILL GET ERROR
1444                    22220          PUSH    H               ; IF NUMBER IS TOO BIG
1445                    22240                                  ;FALL INTO CONIS
1446                    22260
1447                    22280
1448                    22300          ;CONVERT SINGLE PRECISION NUMBER TO INTEGER
1449                    22320          ;ALTERS A,B,C,D,E,H,L
1450                    22340  CONIS:  LDA     FAC             ;GET THE EXPONENT
1451                    22360          CPI     220             ;SEE IF IT IS TOO BIG
1452                    22380          JNC     CONIS2          ;IT IS, BUT IT MIGHT BE -32768
1453                    22400          CALL    QINT            ;IT ISN'T, CONVERT IT TO AN INTEGER
1454                    22420          XCHG                    ;PUT IT IN (HL)
1455                    22440  CONIS1: POP     D               ;GET ERROR ADDRESS OFF STACK
1456                    22460          ;ENTRY FROM SGN, FIN, LINPRT
1457                    22480  CONISS: SHLD    FACLO           ;STORE THE NUMBER IN FACLO
1458                    22500          MVI     A,2             ;SET VALTYP TO "INTEGER"
1459                    22520  CONISD: STA     VALTYP          ;ENTRY FROM CONDS
1460                    22540          RET                     ;ALL DONE
1461                    22560  CONIS2: MOVRI   220,200,000,000 ;CHECK IF NUMBER IS -32768, ENTRY FROM FIN
1462                    22580          CALL    FCOMP
1463                    22600          RNZ                     ;ERROR!  IT CAN'T BE CONVERTED TO AN INTEGER
1464                    22620          MOV     H,C             ;IT IS -32768, PUT IT IN (HL)
1465                    22640          MOV     L,D
1466                    22660          JMP     CONIS1          ;STORE IT IN THE FAC AND SET VALTYP
1467                    22680
1468                    22700
1469                    22720          ;FORCE THE FAC TO BE A SINGLE PRECISION NUMBER
1470                    22740          ;ALTERS A,B,C,D,E,H,L
1471                    22760  FRCSNG: LDA     VALTYP          ;SEE WHAT KIND OF NUMBER WE HAVE
1472                    22780          CPI     4
1473                    22800          RZ                      ;WE ALREADY HAVE AN INTEGER, ALL DONE
1474                    22820          JC      CONSI           ;WE HAVE AN INTEGER, CONVERT IT
1475                    22840          JM      TMERR           ;STRINGS!! -- ERROR!!
1476                    22860                                  ;DBL PREC -- FALL INTO CONSD
1477                    22880
1478                    22900
1479                    22920          ;CONVERT DOUBLE PRECISION NUMBER TO A SINGLE PRECISON ONE
1480                    22940          ;ALTERS A,B,C,D,E,H,L
1481                    22960  CONSD:  CALL    MOVRF           ;GET THE HO'S IN THE REGISTERS
1482                    22980          MVI     A,4             ;SET VALTYP TO "SINGLE PRECISON"
1483                    23000          STA     VALTYP
1484                    23020          MOV     A,B             ;CHECK IF THE NUMBER IS ZERO
1485                    23040          ORA     A
```



```
1486                    23060          RZ                      ;IF IT IS, WE ARE DONE
1487                    23080          CALL    UNPACK          ;UNPACK THE NUMBER
1488                    23100          LXI     H,FACLO-1       ;GET FIRST BYTE BELOW A SNG NUMBER
1489                    23120          MOV     B,M             ;PUT IT IN B FOR ROUND
1490                    23140          JMP     ROUND           ;ROUND THE DBL NUMBER UP AND WE ARE DONE
1491                    23160
1492                    23180
1493                    23200          ;CONVERT AN INTEGER TO A SINGLE PRECISION NUMBER
1494                    23220          ;ALTERS A,B,C,D,E,H,L
1495                    23240  CONSI:  LHLD    FACLO           ;GET THE INTEGER
1496                    23260  CONSIH: MVI     A,4             ;SET VALTYP TO "SINGLE PRECISION"
1497                    23280          STA     VALTYP
1498                    23300          MOV     A,H             ;SET UP REGISTERS FOR FLOATR
1499                    23320          MOV     D,L
1500                    23340          MVI     E,0
1501                    23360          MVI     B,220
1502                    23380          JMP     FLOATR          ;GO FLOAT THE NUMBER
1503                    23400
1504                    23420
1505                    23440          ;FORCE THE FAC TO BE A DOUBLE PRECISION NUMBER
1506                    23460          ;ALTERS A,B,C,D,E,H,L
1507                    23480  FRCDBL: LDA     VALTYP          ;SEE WHAT KIND OF NUMBER WE HAVE
1508                    23500          CPI     10
1509                    23520          RZ                      ;WE ALREADY HAVE A DBL, WE ARE DONE
1510                    23540          JNC     TMERR           ;GIVE AN ERROR IF WE HAVE A STRING
1511                    23560          CPI     2               ;SEE IF WE HAVE A SNG OR INT
1512                    23580          CZ      CONSI           ;CONVERT TO SNG IF WE HAVE AN INT
1513                    23600                                  ;FALL INTO CONDS AND CONVERT TO DBL
1514                    23620
1515                    23640
1516                    23660          ;CONVERT A SINGLE PRECISION NUMBER TO A DOUBLE PRECISION ONE
1517                    23680          ;ALTERS A,H,L
1518                    23700  CONDS:  LXI     H,$CODE         ;ZERO H,L
1519                    23720          SHLD    DFACLO          ;CLEAR THE FOUR LOWER BYTES IN THE DOUBLE
1520                    23740          SHLD    DFACLO+2        ; PRECISION NUMBER
1521                    23760          MVI     A,10            ;SET VALTYP TO "DOUBLE PRECISION"
1522                    23780          JMP     CONISD>         ;GO TO IT
1523                    23800  PAGE
```