

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 0311

MACHO 47(113) 03:12 10-SEP-75 PAGE 21-4
FORMULA EVALUATION CODE

4682 005500# 000000 00547#
4683 005501# 001000 00537#
4684 005502# 000000 000003
4685 005503# 001000 00312
4686 005504# 000000 000000*
4687 005505# 000000 00547#
4688 005506# 001000 000041
4689 005507# 000000 00163#
4690 005510# 001000 000015
4691 005512# 000000 000043
4693 005513# 001000 000106
4694 005514# 001000 000043
4695 005515# 001000 000305
4696 005516# 001000 000372
4697 005517# 000000 005536#
4698 005520# 000000 005507#
4699 005523# 001000 000041
4700 005524# 000000 005523#
4701 005523# 001000 000106
4702 005524# 001000 000043
4703 005525# 001000 000305
4704 005526# 001000 000342
4705 005527# 000000 005536#
4706 005530# 000000 005517#
4707 005531# 001000 000041
4708 005532# 000000 005513#
4709 005533# 000000 005521#
4710 005534# 001000 000367
4711 005535# 001000 000367
4712 005536# 001000 000115
4713 005537# 001000 000107
4714 005540# 001000 000305
4715 005541# 001000 000001
4716 005542# 000000 005642#
4717 005543# 000000 005532#
4718 005544# 001000 000303
4719 005545# 001000 000315
4720 005546# 000000 001575#
4722 005547# 000000 005542#
4723 005550# 001000 000303
4724 005551# 000000 005541#
4725 005552# 000000 005546#
4726 005553# 001000 000315
4727 005554# 000000 003256#
4728 005555# 000000 005551#

MACRO 47(113) 03:12 10-SEP-75 PAGE 21-4
FORMULA EVALUATION CODE

03302 CPI 3 ;AND SET THE CONDITION CODES BASED ON IT
03320 JZ THERR ;JBLW UP ON STRINGS
03340 LXI H,FACLO ;GET POINTER TO LO IN FAC
03342 MOV C,M ;C,M
03344 INX H ;H,M
03346 MOV B,M ;B,M
03348 INX H ;H,M
03360 PUSH B ;B
03380 JM VPUSHD ;IPUSH FACLO+0,1 ON THE STACK
03382 MOV C,M ;C,M
03384 INX H ;H,M
03386 MOV B,M ;B,M
03388 INX H ;H,M
03400 PUSH B ;B
03420 JPO VPUSHD ;JALL DONE IF WE HAD A SNG
03440 LXI H,FACLO ;WE HAVE A DOUBLE PRECISION NUMBER
03460 PUSHM ;IPUSH ITS 4 LO BYTES ON THE STACK
03480 PUSHM ;
03500 VPUSHD: MOV C,E ;JC)=OPERATOR NUMBER
03500 MOV B,A ;IB)=TYPE OF VALUE ON THE STACK
03580 PUSH B ;ISAVE THESE THINGS FOR APPLOP
03600 LXI B,APPLOP ;GENERAL OPERATOR APPLICATION
03620 FINITMP: PUSH B ;ROUTINE -- DOES TYPE CONVERSIONS
03640 FINITMP: PUSH B ;ISAVE PLACE TO GOTOTYPE
03660 LHLD TEMP3 ;RESET THE TEXT POINTER
03680 JMP LPOPER ;IPUSH ON THE PRECEDENCE AND READ MORE
03700 ;JFORMULA
03720 ;
03740 ; FOR EXPONENTIATION WE WANT TO FORCE THE CURRENT VALUE IN THE FAC
03760 ; TO BE SINGLE PRECISION, WHEN APPLICATION TIME COMES WE FORCE
03780 ; THE RIGHT HAND OPERAND TO SINGLE PRECISION AS WELL
03800 ;
03820 EXPSTK: CALL FRCNSG ;COERCE LEFT HAND OPERAND
03840 ANDORD: PUSH D ;
03860 CALL FRINT ;ISAVE THE PRECEDENCE (70 OR 80)
03880 LXI B,FPWRQS# ;PLACE TO COERCE RIGHT HAND
03900 MVI D,127 ;OPERAND AND DO EXPONENTIATION
03920 JMP FINITMP ;RESTORE THE PRECEDENCE
03940 ;
03960 ;
03980 ; FOR "AND" AND "OR" WE WANT TO FORCE THE CURRENT VALUE IN THE FAC
04000 ; FAC TO BE AN INTEGER, AND AT APPLICATION TIME FORCE THE RIGHT
04020 ; HAND OPERAND TO BE AN INTEGER
04040 ANDORD: PUSH D ;
04060 CALL FRINT ;ISAVE THE PRECEDENCE (70 OR 80)
04080 POP D ;
04100 PUSH H ;
04120 LXI B,DANDOR ;
04140 FINITMP ;
04160 ;
04180 ;
04200 ; HERE TO BUILD AN ENTRY FOR A RELATIONAL OPERATOR
04220 ; STRINGS ARE TREATED SPECIALLY, NUMERIC COMPARES ARE DIFFERENT
04240 ; FROM MOST OPERATOR ENTRIES ONLY IN THE FACT THAT AT THE
04260 ; BOTTOM INSTEAD OF HAVING RETAUP, DOCHP AND THE RELATIONAL
04280 ; BITS ARE USED, STRINGS HAVE STRCMP, THE POINTER AT THE STRING DESCRIPTOR,
04300 ; DOCHP AND THE RELATIONAL BITS,
04320 ;
04340 FINREL: MOV A,B ;IA)=OLD PRECEDENCE
04360 CPI 100 ;RELATIONALS HAVE PRECEDENCE 100
04380 RNC ;
04400 ;APPLY EARLIER OPERATOR IF IT HAS
04420 PUSH B ;HIGHER PRECEDENCE
04440 PUSH D ;ISAVE THE OLD PRECEDENCE
04460 LXI D,SCODE+25604 ;IB)=PRECEDENCE=100
04480 ;IE)=DISPATCH OFFSET FOR
04500 ;ICOMPARES IN APPLUP#4
04520 ;IN CASE THIS IS A NUMERIC COMPARE
04540 LXI H,DOUCMP ;ROUTINE TO TAKE COMPARE ROUTINE RESULT

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 0311

4735 005550# 001000 000315
4736 005551# 000000 000004#
4737 005552# 000000 005554#
4738 005561# 001000 000001
4739 005562# 000000 000000*
4740 005563# 000000 005557#
4741 005564# 001000 000026
4742 005565# 000000 000177
4743 005566# 001000 000303
4744 005567# 000000 005544#
4745 005568# 000000 005562#
4746 005569# 000000 005563#
4747 005570# 000000 005564#
4748 005560# 001000 000303
4749 005561# 000000 005544#
4750 005562# 000000 005600#
4751 005571# 001000 000225
4752 005572# 000000 005554#
4753 005573# 000000 005563#
4754 005574# 000000 005564#
4755 005575# 000000 005565#
4756 005576# 001000 000321
4757 005576# 001000 000345
4758 005577# 001000 000001
4759 005600# 000000 005437#
4760 005601# 000000 005573#
4761 005602# 001000 000303
4762 005603# 000000 005544#
4763 005604# 000000 005600#
4764 005605# 000000 005600#
4765 005606# 000000 005600#
4766 005607# 001000 000170
4767 005608# 001000 000376
4768 005609# 000000 000144
4769 005610# 001000 000320
4770 005611# 001000 000303
4771 005611# 001000 000303
4772 005612# 001000 000325
4773 005609# 001000 000325
4774 005613# 001000 000321
4775 005614# 000000 0262084#
4776 005615# 000000 005603#
4777 005616# 001000 000320
4778 005617# 001000 000321
4779 005618# 001000 000325
4780 005619# 000000 005604#
4781 005614# 000000 0262084#
4782 005615# 000000 005603#
4783 005616# 001000 000320
4784 005617# 001000 000321
4785 005618# 001000 000325
4786 005619# 001000 000304#
4787 005617# 000000 005675#

MACRO 47(113) 03:12 10-SEP-75 PAGE 21-5
FORMULA EVALUATION CODE

03840 CALL PUSHF ;INPUT IT ON THE STACK
03860 LXI B,FPWRQS# ;PLACE TO COERCE RIGHT HAND
03880 LXI B,FPWRQS# ;
03900 MVI D,127 ;OPERAND AND DO EXPONENTIATION
03920 JMP FINITMP ;RESTORE THE PRECEDENCE
03940 ;
03960 ;
03980 ; FOR "AND" AND "OR" WE WANT TO FORCE THE CURRENT VALUE IN THE FAC
04000 ; FAC TO BE AN INTEGER, AND AT APPLICATION TIME FORCE THE RIGHT
04020 ; HAND OPERAND TO BE AN INTEGER
04040 ANDORD: PUSH D ;
04060 CALL FRINT ;ISAVE THE PRECEDENCE (70 OR 80)
04080 POP D ;
04100 PUSH H ;
04120 LXI B,DANDOR ;
04140 FINITMP ;
04160 ;
04180 ;
04200 ; HERE TO BUILD AN ENTRY FOR A RELATIONAL OPERATOR
04220 ; STRINGS ARE TREATED SPECIALLY, NUMERIC COMPARES ARE DIFFERENT
04240 ; FROM MOST OPERATOR ENTRIES ONLY IN THE FACT THAT AT THE
04260 ; BOTTOM INSTEAD OF HAVING RETAUP, DOCHP AND THE RELATIONAL
04280 ; BITS ARE USED, STRINGS HAVE STRCMP, THE POINTER AT THE STRING DESCRIPTOR,
04300 ; DOCHP AND THE RELATIONAL BITS,
04320 ;
04340 FINREL: MOV A,B ;IA)=OLD PRECEDENCE
04360 CPI 100 ;RELATIONALS HAVE PRECEDENCE 100
04380 RNC ;
04400 ;APPLY EARLIER OPERATOR IF IT HAS
04420 PUSH B ;HIGHER PRECEDENCE
04440 PUSH D ;ISAVE THE OLD PRECEDENCE
04460 LXI D,SCODE+25604 ;IB)=PRECEDENCE=100
04480 ;IE)=DISPATCH OFFSET FOR
04500 ;ICOMPARES IN APPLUP#4
04520 ;IN CASE THIS IS A NUMERIC COMPARE
04540 LXI H,DOUCMP ;ROUTINE TO TAKE COMPARE ROUTINE RESULT

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF F3 MAC 6-SEP-64 03111 MACRO 4/[113] 03:12 10-SEP-75 PAGE 21-6
FORMULA EVALUATION CODE

```

4788 005620* 001000 005614*          04560           JAND RELATIONAL BITS AND RETURN THE ANSWER
4789 005621* 001000 005615*          04560           PUSH H
4791 005622* 001000 005615*          04560           CALL GETYPE
4792 005623* 001000 005617*          04560           JONES A JMP TO RETURN WHEN DONE
4793 005624* 001000 005617*          04620           JSEE IF WE HAVE A NUMERIC COMPARE
4794 005625* 001000 005617*          04620           JYES, BUILD AN APPLOP ENTRY
4795 005626* 001000 005617*          04620           JNZE NUMREL
4796 005627* 001000 005623*          04640           LHLD FACLO
4797 005628* 001000 005623*          04640           JGET THE POINTER AT THE STRING DESCRIPTOR
4798 005629* 001000 005626*          04660           PUSH H
4799 005630* 001000 005626*          04660           LXI B,STRCMP
4800 005631* 001000 005626*          04660           JSAVE IT FOR STRCMP
4801 005634* 001000 005626*          04680           LXI B,STRCMP
4802 005635* 001000 005626*          04680           JSTRING COMPARE ROUTINE
4803 005636* 001000 005631*          04700           JMP FINITMP
4804 005637* 001000 005630*          04700           JPUT THE ADDRESS, REGET THE TEXT POINTER
4805 005640* 001000 005544*          04720           JSAVE THE PRECEDENCE AND SCAN
4806 005641* 001000 005635*          04722           INMORE OF THE FORMULA
4807 04740           ; APPLOP IS RETURNED TO WHEN IT IS TIME TO APPLY AN ARITHMETIC
4808 04760           ; OR NUMERIC COMPARISON OPERATION.
4809 04762           ; THE STACK HAS A DOUBLE BYTE ENTRY WITH THE OPERATOR
4810 04764           ; NUMBER AND THE VALTY OF THE VALUE ON THE STACK.
4811 04766           ; APPLOP TAKES WHOLE VALUE OFF THE STACK.
4812 04768           ; APPLOP TAKES WHOLE VALUE OFF THE STACK.
4813 04770           ; WILL OCCUR AT END AND CONVERTS THE ARGUMENTS. APPLOP
4814 04772           ; USES DIFFERENT CALLING CONVENTIONS FOR EACH VALUE TYPE.
4815 04776           ; INTEGERS: LEFT IN [D,E], RIGHT IN [M,L]
4816 04778           ; SINGLES: LEFT IN [B,C,D,E] RIGHT IN THE FAC
4817 04780           ; DOUBLES: LEFT IN FAC RIGHT IN ARG
4818 04798           ;
4821 005642* 001000 005051          04800           APPLOP: POP B           ;[B]=STACK OPERAND VALUE TYPE
4822 04820           ;[C]=OPERATOR OFFSET
4823 005643* 001000 005171          04840           MOV A,C           ;SAVE IN MEMORY SINCE THE STACK WILL BE BUSY
4824 005644* 001000 005062          04860           STA DPTRTYP
4825 005645* 001000 005144*          04860           ;A RAM LOCATION
4826 005646* 001000 005640*          04880           ;APLOP#STACK OPERAND VALUE TYPE
4827 005647* 001000 005170*          04880           ;[C]=OPERATOR OFFSET
4828 005650* 001000 005376          04900           MOV A,B           ;CHECK FOR DOUBLE
4829 005651* 001000 0050610          04900           CPI 8            ;PRECISION ENTRY ON THE STACK
4830 005652* 001000 005142*          04920           JZ STKDBL
4831 005653* 001000 005172*          04920           ;FORCE FAC TO DOUBLE
4832 005654* 001000 005645*          04940           LDA VALTYP
4833 005655* 001000 005072*          04940           JSEE IF THE FAC IS DOUBLE PRECISION
4834 005656* 001000 005154*          04940           ;[B]=STACK OPERAND VALUE TYPE
4835 005657* 001000 005653*          04940           ;[C]=OPERATOR OFFSET
4836 005660* 001000 0050316          04960           CPI 8            ;JAND IF SO, CONVERT THE STACK OPERAND
4837 005661* 001000 0050810          04960           ;[B]=STACK OPERAND
4838 005662* 001000 0050312          04980           JZ FACDBL
4839 005663* 001000 0050775*          04980           ;TO DOUBLE PRECISION
4840 005664* 001000 005656*          04980           ;[B]=STACK OPERAND

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF F3 MAC 6-SEP-64 03111 MACRO 47/[113] 03:12 10-SEP-75 PAGE 21-7
FORMULA EVALUATION CODE

```

4841 005665* 001000 005127          05000           MOV D,A           ;SAVE THE VALUE TYPE OF THE FAC
4842 005666* 001000 0050176          05020           MOV D,B           ;SAVE IF THE STACK ENTRY IS SINGLE
4843 005667* 001000 005075*          05020           CPI 4            ;PRECISION AND IF SO, CONVERT
4844 005668* 001000 005094*          05040           ;THE FAC TO SINGLE PRECISION
4845 005671* 001000 0050312          05060           JZ STKSHG
4846 005672* 001000 0050621*          05060           ;THE FAC TO SINGLE PRECISION
4847 005673* 001000 005665*          05080           ;JSEE IF THE FAC IS SINGLE PRECISION
4848 005674* 001000 0050172*          05080           ;JAND IF SO CONVERT THE STACK TO SINGLE
4849 005675* 001000 0050376          05100           CPI 3            ;PRECISION
4850 005676* 001000 0050322          05120           JNC FACSNG
4851 005677* 001000 0050700*          05120           ;PRECISION
4852 005700* 001000 0050634*          05140           ;NOTE: THE STACK MUST BE INTEGER AT THIS POINT
4853 005701* 001000 0050571*          05140           ;BLLOW UP ON RIGHT HAND STRING OPERAND
4854 005702* 001000 0050312          05160           JZ TMERR
4855 005703* 001000 005504*          05160           ;THE STACK MUST BE INTEGER AT THIS POINT
4856 005704* 001000 0050700*          05180           LXI H,INTOSP
4857 005705* 001000 0050041          05180           ;INTEGER INTEGER CASE
4858 005706* 001000 0050700*          05200           MVI B,B
4859 005710* 001000 0050908          05200           ;ISPECIAL DISPATCH FOR SPEED
4860 005711* 001000 0050920          05220           DAD B            ;[H,L] POINTS TO THE ADDRESS TO GO TO
4861 005712* 001000 0050611          05240           DAD B            ;[H,L]=ROUTINE ADDRESS
4862 005713* 001000 0050611          05260           MOV C,M
4863 005714* 001000 0050616          05260           INX H
4864 005715* 001000 0050643          05280           MOV B,M
4865 005716* 001000 0050196          05300           MOV D,M
4866 005720* 001000 0050542          05320           POP D            ;[D,E]=LEFT HAND OPERAND
4867 005721* 001000 0050542          05340           LHLD FACLO
4868 005721* 001000 0051637*          05340           ;[H,L]=RIGHT HAND OPERAND
4871 005722* 001000 0050700*          05360           PUSH B
4872 005723* 001000 0050305          05360           ;IDISPATCH
4873 005724* 001000 0050311          05380           RET
4874 05400           ; THE STACK OPERAND IS DOUBLE PRECISION, SO
4875 05420           ; THE FAC MUST BE FORCED TO DOUBLE PRECISION, MOVED INTO ARG
4876 05440           ; AND THE STACK VALUE POPED INTO THE FAC
4877 05460           ;
4878 05480           ;
4879 005722* 001000 0050315          05520           STKDBL: CALL FRCUBL
4880 005725* 001000 0050654*          05520           ;MAKE THE FAC DOUBLE PRECISION
4881 005727* 001000 0050721*          05540           CALL VMOVAF
4882 005730* 001000 0050315          05540           ;MOVE THE FAC INTO ARG
4883 005731* 001000 0050000*          05560           POP H
4884 005732* 001000 0050726*          05560           SHLD DFACLO+2
4885 005733* 001000 0050341          05580           ;POP OFF THE STACK OPERAND INTO THE FAC
4886 005734* 001000 0051632*          05580           DFACLO+2
4887 005735* 001000 0051635*          05580           SHLD DFACLO+2
4888 005737* 001000 0050341          05580           POP H
4889 005740* 001000 0050042          05600           SHLD DFACLO
4890 005741* 001000 0051633*          05600           ;STORE LOW BYTES AWAY
4892 005742* 001000 0050735*          05620           DFACLO
4893 005743* 001000 0050301          05620           SNGDBL: POPR
4894 05620           ;POP OFF A FOUR BYTE VALUE

```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF F3 MAC 6=SEP=64 03111 MACRO 47(113) 03112 10=SEP=75 PAGE 21=8 FORMULA EVALUATION CODE

```

4994 005744* 001000 000321          05640 CALL MUVR   JINTO THE FAC
4995 005745* 001000 000315          05640 SETDBL: CALL FRCDBL  JMAKE SURE THE LEFT OPERAND IS
4996 005746* 001000 000315          05660 IFN LENGTH=2,<
4997 005747* 001000 005746*          05660 IFN STRING,<
4998 005750* 001000 000315          05660 XRA A
4999 005751* 002000 005746*          05660 STA VALTYP>>  JASSUME THE VALUE WILL BE NUMERIC
4999 005752* 000000 005746*          05690 CHRGRT
4999 005753* 001000 0000041         05680 JC FIN      JIF NUMERIC, INTERPRET CONSTANT
4999 005754* 000000 000672*         05700 LXI H,DBLUSP  JDDOUBLE PRECISION
4999 005755* 000000 000672*         05700 DDUOSP: LUA DPTRTYP  JDISPATCH TO A DOUBLE PRECISION ROUTINE
4999 005756* 001000 00000087        05720 DDUOSP: LUA DPTRTYP  JRECALL WHICH OPERAND IT WAS
4999 005757* 001000 00000087        05740 RLC
4999 005758* 001000 000305         05760 PUSH B
4999 005759* 001000 0008117        05800 MOV C,A
4999 005760* 001000 00000085        05820 MVI B,0
4999 005761* 001000 00000086        05840 DAD B
4999 005762* 001000 00000011        05860 POP B
4999 005763* 001000 0003031        05880 MOV A,M
4999 005764* 001000 00000016        05900 INX H
4999 005765* 001000 00000005        05920 MOV H,M
4999 005766* 001000 00000006        05940 MOV L,A
4999 005767* 001000 00000007        05960 PCML
4999 005768* 001000 00000008        05980 FAND PERFORM THE OPERATION, RETURNING
4999 005769* 001000 00000009        06000 JTO RETADR, EXCEPT FOR COMPARES WHICH
4999 005770* 001000 00000010        06020 JRETURN TO DOCPM
4999 005771* 001000 00000011        06040 ; THE FAC IS DOUBLE PRECISION AND THE STACK IS EITHER
4999 005772* 001000 00000012        06060 ; INTEGER OR SINGLE PRECISION AND MUST BE CONVERTED
4999 005773* 001000 00000013        06080 ;
4999 005774* 001000 00000014        06100 FACDBL: PUSH B
4999 005775* 001000 00000015        06120 VMOVAF  JSAVE THE STACK VALUE TYPE
4999 005776* 001000 00000016        06140 POP PSW
4999 005777* 001000 00000017        06160 STA VALTYP  JPUT IT IN VALTYP FOR THE FORCE
4999 005778* 001000 00000018        06180 ;ROUTINE
4999 005779* 001000 00000019        06200 CPI 4
4999 005780* 001000 00000020        06220 ;SEE IF ITS SINGLE, SO WE KNOW
4999 005781* 001000 00000021        06240 JZ SNGOBL  JHOW TO POP THE VALUE OFF
4999 005782* 001000 00000022        06260 ;IT'S SINGLE PRECISION
4999 005783* 001000 00000023        06280 POP H
4999 005784* 001000 00000024        06300 SHLD FACLO  JPOD DO A POPR / CALL MUVR
4999 005785* 001000 00000025        06320 ;POP OFF THE INTEGER VALUE
4999 005786* 001000 00000026        06340 ;ISAVE IT FOR CONVERSION
4999 005787* 001000 00000027        06360

```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF F3 MAC 6=SEP=64 03111 MACRO 47(113) 03112 10=SEP=75 PAGE 21=9 FORMULA EVALUATION CODE

```

4997 006016* 001000 000303          06320 JMP SETDBL  JSET IT UP
4998 006017* 000000 000575*          06340 STKSGN: CALL FRCSNG  JCONVERT THE FAC IF NECESSARY
4999 006018* 000000 0006014*          06360 ; THIS IS THE CASE WHERE THE STACK IS SINGLE PRECISION
4999 006019* 000000 0006015*          06380 ; AND THE FAC IS EITHER SINGLE PRECISION OR INTEGER
4999 006020* 000000 0006016*          06400 ;ROUTINE
4999 006021* 001000 000315          06420 STKSNG: CALL FRCSNG  JCONVERT THE FAC IF NECESSARY
4999 006022* 000000 0005554*          06440 POPR
4999 006023* 000000 0006017*          06460 SNGDQ: LXI H,SNGUSP  JPUT THE LEFT HAND OPERAND IN THE REGISTERS
4999 006024* 001000 000301          06480 ;PUT THE DISPATCH ADDRESS
4999 006025* 000000 0005555*          06500 ;FOR THE SINGLE PRECISION OPERATOR ROUTINES
4999 006026* 001000 000314          06520 DUDSP
4999 006027* 000000 0006022*          06540 ;DISPATCH
4999 006028* 001000 000303          06560 ;THIS IS THE CASE WHERE THE FAC IS SINGLE PRECISION AND THE STACK
4999 006029* 000000 0006027*          06580 ; IS AN INTEGER.
4999 006030* 001000 000315          06600 FACNSG: POP H
4999 006031* 000000 0005557*          06620 CALL PUSHF  JPOD OFF THE INTEGER ON THE STACK
4999 006032* 000000 0006032*          06640 CALL CONS1H  ;SAVE THE FAC ON THE STACK
4999 006033* 001000 000315          06660 ;CONVERT [H,L] TO A SINGLE PRECISION
4999 006034* 000000 0006034*          06680 CALL MUVRF  JNUMBER IN THE FAC
4999 006035* 001000 000315          06700 CALL MUVRF  JPUT THE LEFT HAND OPERAND IN THE REGISTERS
4999 006036* 000000 0005557*          06720 POP H
4999 006037* 000000 0006032*          06740 SHLD FAC-1  JRESTORE THE FAC
4999 006038* 001000 00000004          06760 ;FROM THE STACK
4999 006039* 000000 0006034*          06780 ;FOR THE SINGLE PRECISION OPERATOR ROUTINES
4999 006040* 001000 000341          06800 ;DISPATCH
4999 006041* 000000 0006041*          06820 EVAL: ;THIS IS THE CASE WHERE THE FAC IS SINGLE PRECISION AND THE STACK
4999 006042* 000000 0006041*          06840 IFN LENGTH=2,<
4999 006043* 001000 000341          06860 IFN STRING,<
4999 006044* 000000 0006044*          06880 XRA A
4999 006045* 001000 000341          06900 STA VALTYP>>  JASSUME THE VALUE WILL BE NUMERIC
4999 006046* 000000 0006044*          06920 CHRGRT
4999 006047* 001000 000341          06940 JC FIN      JIF NUMERIC, INTERPRET CONSTANT
4999 006048* 000000 0006044*          06960
4999 006049* 001000 000341          06980
4999 006050* 000000 0006044*          07000
4999 006051* 001000 000341          07020
4999 006052* 001000 000341          07040
4999 006053* 001000 000341          07060
4999 006054* 000000 0006137*          07080
4999 006055* 001000 000341          07100
4999 006056* 000000 0006058*          07120
4999 006057* 001000 000303          07140
4999 006058* 000000 0006026*          07160
4999 006059* 001000 000303          07180
4999 006060* 000000 0006054*          07200
4999 006061* 001000 000327          07220
4999 006062* 001000 000332          07240

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF F3 MAC 6-SEP-64 03111 MACHO 47(113) 03:12 10-SEP-75 PAGE 21-10 FORMULA EVALUATION CODE

```

5000 006063* 000000 005072*
5001 006064* 000000 006057*
5002 006065* 000000 006056*
5003 006066* 000000 0060512*
5004 006067* 000000 006053*
5005 006068* 000000 0060522*
5006 006071* 000000 006164*
5007 006072* 000000 006066*
5008 006073* 001000 000376
5009 006074* 000000 000250
5010 006075* 001000 000312
5011 006076* 001000 000315
5012 006077* 000000 006071*
5013 006100* 001000 000376
5014 006101* 000000 000056
5015
5016 006102* 001000 000312
5017 006103* 000000 006063*
5018 006104* 000000 006076
5019 006105* 000000 006076
5020 006106* 000000 006051
5021 006107* 001000 000312
5022 006110* 000000 006146*
5023 006111* 000000 006103*
5024
5025 006112* 001000 000376
5026 006113* 000000 000042
5027 006114* 001000 000315
5028 006115* 000000 006040*
5029 006116* 000000 006110*
5030
5031
5032
5033 006117* 001000 000376
5034 006120* 000000 000240
5035 006121* 001000 000312
5036 006122* 000000 006412*
5037 006123* 000000 006115*
5038
5039 006124* 001000 000376
5040 006125* 000000 000243
5041 006126* 001000 000312
5042 006127* 000000 007443*
5043 006130* 001000 000326
5044 006131* 001000 000326
5045 006132* 000000 000252
5046 006133* 001000 000322
5047 006134* 000000 006204*
5048 006135* 000000 006127*
5049
5050
5051
5052 006136* 001000 000317 PARCHK: SYNCMD "("

```

Formula Evaluation

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF F3 MAC 6-SEP-64 03111 MACHO 47(113) 03:12 10-SEP-75 PAGE 21-11 FORMULA EVALUATION CODE

```

5053 006137* 000000 000050
5054
5055 006140* 001000 000315
5056 006141* 000000 005334*
5057 006142* 000000 006134*
5058 006143* 001000 000317
5059 006144* 000000 000051
5060 006145* 001000 000311
5061 006146* 000000 000311
5062
5063
5064
5065 006146* 001000 000226
5066 006147* 000000 000175
5067
5068 006150* 001000 000315
5069 006151* 000000 005341*
5070 006152* 000000 006141*
5071 006153* 001000 000052*
5072 006154* 000000 001605*
5073 006155* 000000 006151*
5074 006156* 001000 000345
5075
5076 006157* 001000 000315
5077 006160* 000000 000000*
5078 006161* 000000 006154*
5079
5080
5081 006162* 001000 000345
5082
5083
5084
5085 006162* 001000 000341
5087 006163* 001000 000311
5088 006164* 001000 000315
5089 006165* 000000 006525*
5090 006166* 000000 006160*
5091
5092 006167* 001000 000345
5093
5094
5095
5096
5097 006168* 001000 000353
5098
5099
5100
5101
5102 006171* 001000 000042
5103 006172* 000000 001637*
5104 006173* 000000 006165*
5105

```

Formula Evaluation

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF MACHO 47(113) 03:12 10-SEP-75 PAGE 21-12
 F3 MAC 6-SEP-64 0311 FORMULA EVALUATION CODE

```

 5106          006174* 001000 000315    08240  IFE LENGTH=2,<
 5107          006175* 000000 000315    08250  CALL GETYPE      JFOR STHINGS WE JUST LEAVE
 5108          006176* 000000 000315*
 5109          006177* 001000 000317*
 5110          006177* 001000 000304    08280  CNZ VMOVFM>  JA POINTER IN THE FAC
 5111          006200* 000000 000000*
 5112          006201* 000000 000317*
 5113          08300  IFN LENGTH=2,<
 5114          08320  LDA VALTYP
 5115          08340  ORA A
 5116          08360  CZ MOVFM>> JIF NOT,ACTUALLY TRANSFER THE VALUE INTO
 5117          08380  POP H             THE FAC USING (HL) AS THE POINTER,
 5118          006202* 001000 000341    08400  RET              RESTORE THE TEXT POINTER
 5119          006203* 001000 000311    08420
 5120          006204* 001000 000006    08440  ISFUN: MVI B,0
 5121          006205* 000000 000000
 5122          006206* 001000 000007    08460  RLC
 5123          006207* 001000 000117    08480  MOV C,A      MULTIPLY BY 2
 5124          006208* 001000 000305    08500  PUSH B
 5125          006210* 001000 000327    08520  CHGET
 5126          006211* 001000 000327    08540  IFN STRING,<
 5127          006212* 001000 000171    08560  MOV A,C      JSAVE THE FUNCTION # ON THE STACK
 5128          006213* 001000 000051    08580  NUMGFn=##2*LASNUM+2*ONEFUN+1
 5129          006214* 000000 000051    08600  CPI NUMGFn  JIS IT PAST LASNUM?
 5130          006215* 001000 000332    08620  JC OKNRM     JNO,MUST BE NORMAL FUNCTION
 5131          006216* 000000 000251*
 5132          006217* 000000 000200*
 5133
 5134          08640  J MOST FUNCTIONS TAKE A SINGLE ARGUMENT,
 5135          08660  J THE RETURN ADDRESS OF THESE FUNCTIONS IS A SMALL ROUTINE
 5136          08680  J THAT CHECKS TO MAKE SURE VALTYP IS 0 (NUMERIC) AND POPS OFF
 5137          08700  J THE TEXT POINTER, SO NORMAL FUNCTIONS THAT RETURN STRING RESULTS (I,E, CHR$)
 5138          08720  J MUST PUT OFF THE RETURN ADDRESS OF LABCK, AND POP OFF THE
 5139          08740  J TEXT POINTER AND THEN RETURN TO FRMELV,
 5140          08760  J
 5141          08780  J
 5142          08800  J THE SO CALLED "FUNNY" FUNCTIONS CAN TAKE MORE THAN ONE ARGUMENT.
 5143          08820  J THE FIRST OF WHICH MUST BE STRING AND THE SECOND OF WHICH
 5144          08840  J MUST BE A NUMBER BETWEEN 0 AND 255, THE TEXT POINTER IS
 5145          08860  J PASSED TO THESE FUNCTIONS SO ADDITIONAL ARGUMENTS
 5146          08880  J CAN BE READ, THE TEXT POINTER IS PASSED IN (D,E),
 5147          08900  J THE CLOSE PARENTHESIS MUST BE CHECKED AND RETURN IS DIRECTLY
 5148          08920  J TO FRMELV WITH (M,L) SETUP AS THE TEXT POINTER POINTING BEYOND THE ")"
 5149          08940  J THE POINTER TO THE DESCRIPTOR OF THE STRING ARGUMENT
 5150          08960  J IS STORED ON THE STACK UNDERNEATH THE VALUE OF THE INTEGER
 5151          08980  J ARGUMENT (2 BYTES)
 5152          09000  J
 5153          006220* 001000 000317    09020  SYNCHK "("
 5154          006221* 000000 000050    09040  CALL FRMELV   JFIRST ARGUMENT ALWAYS
 5155          006222* 001000 000315    09060  CALL FRMELV   JSTRING -- SECOND INTEGER
 5156          006223* 000000 000536*
 5157          006224* 000000 000216*

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF MACHO 47(113) 03:12 10-SEP-75 PAGE 21-13
 F3 MAC 6-SEP-64 0311 FORMULA EVALUATION CODE

```

 5159          006225* 001000 000317    09080  SYNCHK 44      JTWO ARGS SO COMMA MUST DELIMIT
 5160          006225* 000000 000054
 5161          006226* 001000 000315    09100  CALL CHKSTR   JMAKE SURE THE FIRST ONE WAS STRING
 5162          006227* 000000 000416*
 5163          006231* 000000 000223*
 5164          006232* 001000 000353    09120  XCHG
 5165          006233* 001000 000052    09140  LHLD FACLO   J(D,E)=TXTPTR
 5166          006234* 000000 001637*
 5167          006235* 000000 000230*
 5168          006236* 001000 000343    09160  XTHL      JGET PTR AT STRING DESCRIPTOR
 5169
 5170          006237* 001000 000345    09180
 5171          006240* 001000 000353    09200  PUSH H      JSAVE THE STRING PTR
 5172          006241* 001000 000315    09220  XCHG      J(D,E)=TXTPTR
 5173          006242* 000000 011020*    09240  CALL GETBYT  J(E)*VALUE OF FORMULA
 5174          006243* 000000 0006234*   09260  XCHG      JTEXT POINTER INTO (D,E)
 5175          006244* 001000 000353    09280
 5176          006245* 001000 000343    09300  XTHL      J(M,L)=INT VALUE OF SECOND ARGUMENT
 5177          006246* 001000 000393    09320
 5178          006247* 000000 0006276*   09340  JMP FINGO>  JSAVE INT VALUE OF SECOND ARG
 5179          006248* 001000 000393    09360  OKNRM: CALL PARCHK J(M,L)=FUNCTION NUMBER
 5180          006249* 000000 0006276*   09380
 5181          006250* 000000 0006242*
 5182          006251* 001000 000315    09400
 5183          006252* 000000 0006136*   09420  IFE LENGTH=2,<
 5184          006253* 000000 0006247*   09440  XTHL      JAND MAKE SURE ITS FOLLOWED BY ")"
 5185
 5186          006254* 001000 000343    09460
 5187          006254* 001000 000343    09480  ; CHECK IF SPECIAL COERCION MUST BE DONE FOR ONE OF THE TRANSCENDENTAL
 5188          006254* 000000 000016    09500  ; FUNCTIONS (RND, SQRT, COS, SIN, TAN, ATN, LOG, AND EXP)
 5189          006255* 001000 000175    09520  ;
 5190          006255* 000000 000016    09540  MOV A,L      J(A)=FUNCTION NUMBER
 5191          006255* 001000 000175    09550  BOTCON=SQRTK-UNEFUN*#2
 5192          006255* 000000 000016    09560  CPI BOTCON   JLESS THAN SQUARE ROOT?
 5193          006255* 001000 000175    09580  JC NOTFRF   JDON'T FORCE THE ARGUMENT
 5194          006255* 000000 000016
 5195          006255* 001000 000175
 5196          006255* 000000 000016
 5197          006260* 001000 000352    09620  PUSH H      JSAVE THE FUNCTION NUMBER
 5198          006261* 000000 0006272*   09640  CC FRCSNG   JIF NOT, FORCE FAC TO SINGLE-PRECISION
 5199          006262* 000000 0006252*   09660
 5200          006263* 001000 000035    09680  TOPCON==<ATNTK-UNEFUN>*#2
 5201          006263* 000000 000376    09700  CPI TOPCON   JBIGGER THAN ARC-TANGENT?
 5202          006264* 000000 000035
 5203          006265* 001000 000035    09720
 5204          006265* 000000 000324    09740
 5205          006266* 000000 0006282*   09760
 5206          006267* 000000 0006261*   09780
 5207          006271* 001000 000341    09800  NOTFRF: POP H      JRESTORE THE FUNCTION NUMBER
 5208          006272* 001000 000021    09820
 5209          006272* 000000 0006162*   09840  LXI D,LABCK  JRETURN ADDRESS
 5210          006273* 000000 0006162*   09860
 5211          006274* 000000 0006267*
```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 0311

MACHO 47(113) 03:12 10-SEP-75 PAGE 21-14
FORMULA EVALUATION CODE

5212 006275# 001000 000105 00720 PUSH D ;MAKE THEM REALLY COME BACK
5213 006276# 001000 000001 00740 FINGO: LXI B,FUNSP ;FUNCTION DISPATCH TABLE
5214 006277# 000000 000103#
5215 006300# 000000 000273#
5216 006301# 001000 000011 00760 IFE DAD B ;ADD ON THE OFFSET
5217 00780 LENGTH,<
5218 00800 PUSHM
5219 00820 RET> ;GO TO THE ADDRESS POINTED TO BY [H,L]
5220 00840 IFN LENGTH,<
5221 00860 MOV C,M ;FASTER THAN PUSHM
5222 00863# 001000 000116 INX H
5223 00863# 001000 000043 INX M
5224 00863# 001000 000146 MOV M,M
5225 00863# 001000 000151 00920 MOV L,C
5226 00863# 001000 000351 00940 PCHL> ;GO PERFORM THE FUNCTION
5227 00960 IFE LENGTH=2,<
5228 00980 ;
5229 10000 ; GET THE VALTYP AND SET CONDITION CODES AS FOLLOWS:
5230 10020 ; CONDITION CODE TRUE SET FALSE SET
5231 10030 ;
5232 10040 ; SIGN INT#2 STR,SNG,DBL
5233 10060 ; ZERO STR=3 INT,SNG,DBL
5234 10080 ; ODD PARITY SNG#4 INT,STR,DBL
5235 10100 ; NO CARRY DBL=10 INT,STR,SNG
10120 ;
5236 006307# 001000 000072 10140 GETYPE: LOA VALTYP ;GET THE VALTYP
5237 006311# 000000 001543#
5238 006311# 001000 000077#
5239 006312# 001000 000057#
5240 006313# 000000 000010#
5241 006314# 001000 000075#
5242 006315# 001000 000075#
5243 006316# 001000 000075#
5244 006317# 001000 000311 10160 CPI 8 ;SET CARRY CORRECTLY
5245 10200 IFN LENGTH=2,<
5246 10300 IFN LENGTH,<
5247 10320 ORI XHD "01000,"0366 ;DORIN, FLAG AS "OK"
5248 10340 ;AND USE COMMON "AND" CODE
5249 10360 ANDE XRA A ;FLAG AS "AND"
5250 10380 ANDON: ;
5251 10400 PUSH PSW
5252 10420 IFN STRING,<CALL CHKNUM> ;GET [D,E]=INT VALUE AND CHECK SIZE
5253 10440 CALL DEINT
5254 10460 POP PSW
5255 10480 XCHG ;[M,L]=INT VALUE
5256 10500 POP B ;GET HIGH ORDER OFF
5257 10520 XTHL ;PUT INT VALUE ON
5258 10540 ;GET LOW OF SECOND ARG OFF
5259 10560 XCHG ;[D,E]=LOW OF SECOND ARG
5260 10580 CALL MUVFR
5261 10600 PUSH PSW
5262 10620 CALL DEINT ;GET [D,E]=INT VALUE
5263 10640 POP PSW

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 0311

MACHO 47(113) 03:12 10-SEP-75 PAGE 21-15
FORMULA EVALUATION CODE

5264 10660 ;
5265 10680 POP B ;GET FIRST ARG AND CHECK SIZE
5266 10700 MOV A,C ;[D,E]=INT VALUE OF SECOND ARG
5267 10720 LXI H,014ACF ;SETUP JUMP ADDRESS
5268 10740 JNZ DRFIN ;IT WAS "0" SO FINISH UP
5269 10760 ANA E ;JAND TWO LOW ORDER
5270 10780 MOV C,A ;ISAVE ANSWER IN [C]
5271 10800 MOV A,B ;[A]>HIGH ORDER SECOND ARG
5272 10820 ANA D ;[A]>HIGH ORDER OF ANSWER
5273 10840 PCHL ;FLDQT [A,C] AS ANSWER
5274 10860 ORFIN: ORI E ;TOP TWO LOW ORDERS
5275 10880 MOV D,A ;ISAVE ANSWER LOW ORDER IN [C]
5276 10900 MOV A,B ;[A]>HIGH ORDER SECOND ARG
5277 10920 ORA D ;JOR TWO HIGH ORDERS
5278 10940 ;[A]>HIGH ORDER OF ANSWER
5279 10960 ;FLDQT [A,C] AS ANSWER
5280 10980 FINREL: LXI H,PTDURL ;MAKE [H,L] POINT AT OPERATOR ADDRESS
5281 11000 IFN STRING,<
5282 11020 LDVAL VALTYP ;STORE VALUE TYPE AS LOW
5283 11040 RAL ;OPERATOR BIT OF [E]
5284 11060 MOV A,D ;GET RELATIONAL MEMORIES IN [A]
5285 11080 RAL ;MOVE CARRY BIT IN
5286 11100 MOV E,> ;KEEP THIS BYTE IN [E]
5287 11120 IFE STRING,<MOV E,D> ;
5288 11140 MVI D,100 ;PRECEDENCE OF ALL RELATIONAL
5289 11160 ;OPERATORS IS 100
5290 11180 MOV A,B ;[C]>PRECEDENCE OF OLD OPERATOR
5291 11200 CMP D ;SEE IF TIME TO APPLY OLD OPERATOR
5292 11220 RNC ;IF OLD OPERATOR HAS EQUAL OR GREATER
5293 11240 JMP DUPREC ;PRECEDENCE THAN IT MUST BE APPLIED
5294 11260 ;SEE IF TIME TO APPLY
5295 11280 ;JAND IF NOT SAVE INFO ON THE STACK
5296 11300 ;
5297 11320 PTDURL: ADR(DOREL) ;ADDRESS OF RELATIONAL
5298 11340 ;OPERATOR APPLICATION
5299 11360 ;ROUTINE
5300 11380 ;
5301 11400 ; TIME TO PERFORM A RELATIONAL OPERATOR
5302 11420 ;[C] CONTAINS THE BITS AS TO WHICH RELATIONAL
5303 11440 ;OPERATOR IT WAS (IF STRINGS ON
5304 11460 ;LOW ORDER BIT SAYS WHETHER IT WAS STRING OR NOT)
5305 11480 ;
5306 11500 DOREL: MOV A,C ;GET MEMORIES INTO [A]
5307 11520 IFN STRING,<
5308 11540 ORA A ;
5309 11560 RAR> ;CARRY=WHETHER WAS STRING OR NOT
5310 11580 POPR ;POP OFF LEFT RESULT
5311 11600 PUSH PSW ;ISAVE WHICH OPERATOR IT WAS
5312 11620 IFE STRING,<
5313 11640 CALL FCMP> ;DO A NUMERIC COMPARE
5314 11660 IFN STRING,<
5315 11680 CALL CKVAL ;SEE IF VALTYP MATCHES
5316 11700 ICARRY AND SET ZERO IN THE

```

5316          11720      INUMERIC CASE
5319          11720      LXI H,DOCMP
5320          11720      PUSH H
5321          11720      JZ FCOMP
5322          11720      J00 NUMERIC COMPARE
5323          11720      XRA A
5324          11720      STA VALTYP>>
5325          11840      IFN STRING,<
5326          11860      ;
5327          11880      ; THE FOLLOWING ROUTINE COMPARES TWO STRINGS
5328          11900      ; ONE WITH DESC IN (D,E) OTHER WITH DESC, IN (FACTL, FACTL+1)
5329          11920      ; A&B IF STRINGS EQUAL
5330          11920      ; A&B IF B,C,D,E POINTER FALSE
5331          11940      ; A&1 IF B,C,D,E LT. FACTL
5332          11980      ;
5333          12000      IFN LENGTH=2,<
5334          12020      STRCMP: PUSH D      JSAVE DESC, POINTER TO FIRST STR,
5335          12040      CALL FREACF    JFREE THE FACTL STR
5336          12060      POP D       JRESTORE 1ST DESC., POINTER
5337          12080      PUSHH H      JSAVE LENGTH
5338          12100      PUSHM A      JSAVE COUNTER
5339          12120      CALL FRETHP    JFREE 1ST DESC., POINTER
5340          12140      CALL MOVRM    J(B,C) POINT AT FIRST CHAR
5341          12160      POP H       JGET 2ND CHARACTER POINTER IN H
5342          12180      XTHL        JGET 2ND CHARACTER COUNTER IN L
5343          12200      MOV D,L>
5344          12240      IFE LENGTH=2,<
5345          12260      STRCMP: CALL FRESTR   JFREE UP THE FAC STRING, AND GET THE
5346          006320# 001000 000315
5347          006321# 000000 010431#
5348          006322# 000000 006310#
5349          12280      ;
5350          006323# 001000 000176
5351          006324# 001000 000043
5352          006325# 001000 000116
5353          12340      MOV A,M      JPOINTER TO THE FAC DESCRIPTOR IN (M,L)
5354          12360      INX H       JSAVE THE LENGTH OF THE FAC STRING IN [A]
5355          12380      MOV B,M      JSAVE THE POINTER AT THE FAC STRING
5356          12400      INX H       JDATA IN (B,C)
5357          12420      PUSH B      JGET THE STACK STRING POINTER
5358          12440      PUSH PSW    JSAVE THE POINTER AT THE FAC STRING DATA
5359          12460      CALL FRETMP   JSAVE THE FAC STRING LENGTH
5360          006334# 000000 010440#
5361          006335# 000000 006321#
5362          12480      ;
5363          006335# 001000 000321
5364          006337# 001000 000136
5365          006340# 001000 000043
5366          006341# 001000 000116
5367          006342# 001000 000043
5368          006343# 001000 000106
5369          006344# 001000 000341
5370          006345# 001000 000173
5371          12500      POP D       JIN [M,L]
5372          12520      MOV E,M      J(D,E)=LENGTH OF FAC STRING
5373          12540      INX H       J(E)=LENGTH OF STACK STRING
5374          12560      MOV C,M      J(B,C)=POINTER AT STACK STRING
5375          12580      INX H       JTEST BACK 2ND CHARACTER POINTER
5376          12600      MOV B,M>
5377          12620      POP H       JBOTH STRINGS ENDED
5378          12640      CSLOOP: MOV A,E
5379          12660      CSLOOP: MOV A,E

```

```

5371          006340# 001000 000262
5372          006341# 001000 000310
5373          006342# 001000 000212
5374          006351# 001000 000267
5375          006352# 001000 000457
5376          006353# 001000 000310
5377          006354# 001000 000257
5378          006355# 001000 000273
5379          006356# 001000 000074
5380          006357# 001000 000320
5381          12680      ORA D      JTEST BY ADDING THE LENGTHS TOGETHER
5382          12700      RZ
5383          12720      MOV A,D      JIF SO, RETURN WITH A ZERO
5384          12740      ORA A      JFACTL STRING LENGTH
5385          12760      ORA A      JIF END, OTHER MUST NOT HAVE
5386          12780      CHA
5387          12800      RZ
5388          12820      XRA A      JMAKE #1
5389          12840      CMP E      JTEST
5390          12860      INR A      JMUST NOT HAVE BEEN ZERO, TEST CASE
5391          12880      INR A      JOF B,C,D,E STRING HAVING ENDED FIRST
5392          12900      INR A      JRETURN WITH A#1
5393          12920      RNC
5394          12940      DEC D      JTEST THE CONDITION
5395          12960      DCR E      JDECREMENT BOTH CHARACTER COUNTS
5396          12980      LOAX B      JGET CHARACTER FROM B,C,D,E STRING
5397          13000      CMP M      JCOMPARE WITH FACTL STRING
5398          13020      INX H      JUMP POINTERS (INX DOESNT Clobber CC$)
5399          13040      INX B
5400          13060      JZ CSLOOP   JIF BOTH THE SAME, MUST BE MORE TO STRINGS
5401          006361# 000000 006345#
5402          006370# 000000 006344#
5403          006371# 001000 000635
5404          006362# 001000 000012
5405          006363# 001000 000276
5406          006364# 001000 000043
5407          006365# 001000 000005
5408          006366# 001000 000312
5409          006367# 000000 006345#
5410          13080      IFN LENGTH=2,<
5411          13100      DOCMP: INR A      JSETUP BITS
5412          13120      ADC A      J4<LESS 2>EQUAL 1>GREATER
5413          13140      POP B      JWHAT DID HE WANT?
5414          13160      ANA B      JANY BITS MATCH?
5415          13180      ADI 255    JMAP 0 TO 8
5416          13200      SBB A      JAND ALL OTHERS TO 377
5403          13220      IFE LENGTH=2,<
5404          13240      CALL CONIA##    JCONVERT [A] TO AN INTEGER SIGNED
5405          006400# 001000 000315
5406          006405# 000000 000000#
5407          006406# 001000 000305
5408          006410# 000000 005554#
5409          006411# 000000 006405#
5410          13280      IFN LENGTH=2,<
5411          13300      JMP FLDAT   JMAKE FAC=[A] SIGNED
5412          13320      FLDAT
5413          13340      ;
5414          13360      ;
5415          13380      NOTER: MVI D,90    JNOT# HAS PRECEDENCE 90
5416          13400      CALL LUPPER  JGET PERFORM
5417          13420      IFN STRING,CALL CHKNUM> JMAKE SURE ITS INT
5418          13440      CALL DEINT   JGET VALUE IN (D,E)
5419          13460      MOV A,E
5420          13480      CHA
5421          13500      MOV C,A      JCOMPLEMENT
5422          13520      MOV A,D
5423          13540      CHA      JCOMPLEMENT HIGH ORDER TOO

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05:11

MACHO 47(113) 03:12 10-SEP-75 PAGE 21-18
FORMULA EVALUATION CODE

5424 15560 CALL GIVACF ;FLUAT [A,C] AS RESULT
5425 15580 POP ;TAKE RETURN ADDRESS OF FRMELV
5426 15590 JMP RETADP>> ;LOAD AND COERCE THE RIGHT
5427 15620 ;PLACE SO THE TEXT POINTER
5428 15640 ;WILL GET SET UP TO WHAT IT WAS
5429 15660 ;WHEN LOPER RETURNED.
5430 15680 IFE LENGTH=2,<
5431 006412* 001000 000026 15700 NOTER: MVI D,90 ;NOT HAS PRECEDENCE 90, SO
5432 006413* 000000 000132 15720 CALL LOPER ;FORMULA EVALUATION IS ENTERED WITH A DUMMY
5433 006414* 001000 000315
5434 006415* 000000 005573*
5435 006416* 000000 006410*
5436 15730 CALL FRCINT ;ENTRY OF 98 ON THE STACK
5437 006417* 001000 000315 15740 CALL FRCINT ;COERCE THE ARGUMENT TO INTEGER
5438 006420* 000000 005573*
5439 006421* 000000 006415*
5440 006422* 001000 000857 15760 MOV A,L ;COMPLEMENT [M,L]
5441 006423* 001000 000857 15780 CHA
5442 006424* 001000 000857 15800 MOV L,A
5443 006425* 001000 0008174 15820 MOV A,M
5444 006426* 001000 000857 15840 CHA
5445 006427* 001000 000147 15860 MOV M,A
5446 006430* 001000 000042 15880 SHLD FACLO ;UPDATE THE FAC
5447 006431* 000000 001637*
5448 006432* 000000 006420*
5449 006433* 001000 000301 15900 POP B ;JFRMEVL, AFTER SEEING THE PRECEDENCE
5450 15910 ;TUF 98 THINKS IT IS APPLYING AN OPERATOR
5451 15915 ;SO IT HAS THE TEXT POINTER IN TEMP2 SO
5452 15920 JMP RETADP ;RETURN TO REFETCH IT
5453 006434* 001000 000303
5454 006435* 000000 005354*
5455 006436* 000000 006431*
5456 14042 ;
5457 14044 ;DANDOR APPLIES THE "AND" AND "OR" OPERATORS
5458 14046 ;AND SHOULD BE USED TO IMPLEMENT ALL LOGICAL OPERATORS.
5459 14048 ;WHENEVER AN OPERATOR IS APPLIED, ITS PRECEDENCE IS IN [B].
5460 14050 ;THIS FACT IS USED TO DISTINGUISH BETWEEN "AND" AND "OR".
5461 14052 ;THE RIGHT HAND ARGUMENT IS COERCED TO INTEGER, JUST AS
5462 ;THE LEFT HAND ONE WAS WHEN IT WAS PUSHED ON THE STACK.
5463 14056 ;
5464 006437* 001000 000305 14060 DANDOR: PUSH B ;SAVE THE PRECEDENCE "OR"=70
5465 006440* 001000 000315 14060 CALL FRCINT ;COERCE RIGHT HAND ARGUMENT TO INTEGER
5466 006441* 000000 006420*
5467 006442* 000000 006435*
5468 006443* 001000 000361 14100 POP PSW ;GET BACK THE PRECEDENCE TO DISTINGUISH
5469 14120 ;"AND" AND "OR".
5470 006444* 001000 000321 14140 POP D ;POP OFF THE LEFT HAND ARGUMENT
5471 006445* 001000 000376 14160 CPI 70 ;SET ZERO FOR "OR".
5472 006446* 000000 000106
5473 006447* 001000 000173 14180 MOV A,E ;SETUP LOH IN [A]
5474 006450* 001000 000312 14200 JZ ORFIN ;DO "OR" IF PRECEDENCE WAS 70
5475 006451* 000000 006465*
5476 006452* 000000 006441*
5477 006453* 001000 000245 14220 ANA L
5478 006454* 001000 000157 14240 MOV L,A
5479 006455* 001000 000174 14260 HUV A,M
5480 006456* 001000 000242 14280 ANA D
5481 006457* 001000 000147 14300 MOV M,A
5482 006460* 001000 000383 14320 JMP MAKINT ;RETURN THE INTEGER [M,L]
5483 006461* 000000 000008*
5484 006462* 000000 006451*
5485 14322 ; AS THE "AND"ED RESULT
5486 006463* 001000 000265 14324 ORFIN: ORA L
5487 006464* 001000 000157 14326 MOV L,A
5488 006465* 001000 000174 14328 MOV A,M
5489 006466* 001000 000242 14330 ORA D
5490 006467* 001000 000147 14332 MOV M,A
5491 006470* 001000 000383 14334 JMP MAKINT> ;RETURN THE INTEGER [M,L]
5492 006471* 000000 006461*
5493 006472* 000000 006461*
5494 14336 ; AS THE "OR"ED RESULT
5495 14340 PAGE

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05:11

MACHO 47(113) 03:12 10-SEP-75 PAGE 21-19
FORMULA EVALUATION CODE

5477 006453* 001000 000245 14220 ANA L
5478 006454* 001000 000157 14240 MOV L,A
5479 006455* 001000 000174 14260 HUV A,M
5480 006456* 001000 000242 14280 ANA D
5481 006457* 001000 000147 14300 MOV M,A
5482 006460* 001000 000383 14320 JMP MAKINT ;RETURN THE INTEGER [M,L]
5483 006461* 000000 000008*
5484 006462* 000000 006451*
5485 14322 ; AS THE "AND"ED RESULT
5486 006463* 001000 000265 14324 ORFIN: ORA L
5487 006464* 001000 000157 14326 MOV L,A
5488 006465* 001000 000174 14328 MOV A,M
5489 006466* 001000 000242 14330 ORA D
5490 006467* 001000 000147 14332 MOV M,A
5491 006470* 001000 000383 14334 JMP MAKINT> ;RETURN THE INTEGER [M,L]
5492 006471* 000000 006461*
5493 006472* 000000 006461*
5494 14336 ; AS THE "OR"ED RESULT
5495 14340 PAGE

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05111

MACHO 47(113) 05112 10-SEP-75 PAGE 22
DIMENSION & VARIABLE SEARCHING

5496 006473# 001000 000053 14360 SUBTTL DIMENSION & VARIABLE SEARCHING
5497 006474# 001000 000054 14380 DIMCON: DCX H ISEE IF COMMA ENDED THIS VARIABLE
5498 006475# 001000 000510 14400 CHRGET
5499 006476# 001000 000511 14420 RZ ;IF TERMINATOR, GOOD BYE
5500 006477# 002000 000054 14440 SYNCMD 44 MUST BE COMMA

5502 14442 ;
5503 14444 ; THE "DIMM" CODE SETS DIMFLG AND THEN FALLS INTO THE VARIABLE
5504 14446 ; SEARCH ROUTINE, THE VARIABLE SEARCH ROUTINE LOOKS AT
5505 14448 ; DIMFLG AT THREE DIFFERENT POINTS:

5506 14450 ;
5507 14452 ; 1) IF AN ENTRY IS FOUND, DIMFLG BEING ON INDICATES
5508 14454 ; A DOUBLE QUOTED DIMENSIONED VARIABLE
5509 14456 ; 2) WHEN A NEW ENTRY IS BEING BUILT DIMFLG'S BEING ON
5510 14458 ; INDICATES THE INDIICES SHOULD BE USED FOR
5511 14460 ; THE SIZE OF EACH INDICE, OTHERWISE THE DEFAULT
5512 14462 ; OF TEN IS USED,
5513 14464 ; 3) WHEN THE BUILD ENTRY CODE FINISHES, ONLY IF DIMFLG IS
5514 14466 ; OFF WILL INDEXING BE DONE
5515 14468 ;
5516 006500# 001000 000001 14478 DIM: LXI B,0DIMCON IPLACE TO COME BACK TO
5517 006501# 000000 006473#
5518 006502# 000000 006471#
5519 006503# 001000 000305 14480 PUSH B *01000,"0366 J"OKIN' NON ZERO THING
5520 006504# 001000 000366 14500 XHD *01000,"0366 J"OKIN' NON ZERO THING
5521 14520 ;
5522 14540 ; ROUTINE TO READ THE VARIABLE NAME AT THE CURRENT TEXT POSITION
5523 14550 ; AND PUT IT PORTED TO THE VALUE IN [D]-[E] (MLL IS UPDATED
5524 14560 ; TO POINT TO THE CHARACTER AFTER THE VARIABLE NAME)
5525 14580 ; VALTYP IS SETUP, NOTE THAT EVALUATING SUBSCRIPTS IN
5526 14620 ; A VARIABLE NAME CAN CAUSE RECURSIVE CALLS TO PTGET SO AT
5527 14640 ; THAT POINT ALL VALUES MUST BE STORED ON THE STACK,
5528 14660 ;
5529 006505# 001000 000257 14680 PTRET1: XRA A JMAKE [A]#
5530 006506# 001000 000062 14700 STA DIMFLG JFLAG IT AS SUCH
5531 006507# 000000 00114#
5532 006510# 000000 006501#
5533 006511# 001000 000106 14720 MOV B,M JGET FIRST CHARACTER IN [B]
5534 006512# 001000 000315 14740 PTR6T2: CALL ISLET JCHECK FOR LETTER
5535 006513# 000000 003612#
5536 006514# 000000 005057#
5537 006515# 001000 000332 14760 JC SNERR JMUST HAVE A LETTER
5538 006516# 000000 002072#
5539 006517# 000000 006513#
5540 006520# 001000 000257 14780 XRA A
5541 006521# 001000 000117 14800 MOV C,A JASSUME NO SECOND CHARACTER
5542 14820 IFN LENGTH=2,<
5543 14840 IFN STRING,<
5544 14860 STA VALTYP>> JDEFAULT IS ZERO (NUMERIC)
5545 006522# 001000 000527 14880 CHRGET JGET THE FOLLOWING CHARACTER
5546 14900 IFE LENGTH,<
5547 14920 JNC NUSEC> JONLY NUMBERS ALLOWED
5548 14940 IFN LENGTH,<

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05111

MACHO 47(113) 05112 10-SEP-75 PAGE 22-1
DIMENSION & VARIABLE SEARCHING

5549 006523# 001000 000332 14960 JC ISSEC JCARRY SET BY CHRGET IF CHARACTER IS
5550 006524# 000000 006534#
5551 006525# 000000 006516#
5552 14980 ;
5553 006526# 001000 000315 15000 CALL ISLET JNUMERIC
5554 006527# 000000 003612#
5555 006530# 000000 006524#
5556 006531# 001000 000332 15020 JC NUSEC> JALLOW ALPHABETICS
5557 006532# 000000 006547#
5558 006533# 000000 006527#
5559 006534# 001000 000117 15040 ISSEC: MOV C,A JIT IS A NUMBER--SAVE IN C
5560 006535# 001000 000327 15060 EATEM: CHRGET JLLOOK AT NEXT CHARACTER
5561 15080 IFN LENGTH,<
5562 15100 JC EATEM JSKIP NUMERICS
5563 006537# 000000 006535#
5564 006540# 000000 006532#
5565 006541# 001000 000315 15120 CALL ISLET
5566 006542# 000000 003612#
5567 006543# 000000 006537#
5568 006544# 001000 000322 15140 JNC EATEM> JSKIP ALPHABETICS
5569 006545# 000000 006535#
5570 006546# 000000 006542#
5571 006547# 000000 006542# 15160 NUSEC:
5572 15180 IFE LENGTH=2,< LXI D,HAVTYP JSAVE JUMPS BY USING RETURN ADDRESS
5573 006547# 001000 000021 15200 ;
5574 006550# 000000 006574#
5575 006551# 000000 006545#
5576 006552# 000000 006526#
5577 006553# 000000 006610# 15220 PUSH D
5578 006554# 000000 006610# 15240 MVI D,D,8 JASSUME ITS DOUBLE PRECISION
5579 006555# 000000 006376# 15260 CPI "#" JCHECK THE CHARACTER
5580 006556# 000000 006045#
5581 006557# 001000 000310 15280 RZ JWHEN WE MATCH, SETUP VALTYP
5582 006560# 001000 000026 15300 MVI D,D,2 JCHECK FOR INTEGER
5583 006561# 000000 006002#
5584 006562# 001000 000376 15320 CPI "#"
5585 006563# 000000 006015#
5586 006564# 001000 000510 15340 RZ
5587 006565# 001000 000024 15360 INR D JCHECK FOR STRING
5588 006566# 001000 000376# 15380 CPI "#"
5589 006567# 000000 006044#
5590 006568# 001000 000310 15400 RZ
5591 006571# 001000 000024 15420 INR D JSINGLE PRECISION IS THE DEFAULT
5592 006572# 001000 000053 15440 DCX H JNO MARKING CHARACTER
5593 006573# 001000 000311 15460 RET JGET RID OF RETURN ADDRESS
5594 006574# 001000 000026# 15480 HAVTYP: MOV A,D JSETUP VALTYP
5595 006575# 001000 000062 15500 STA VALTYP
5596 006576# 000000 001543#
5597 006577# 000000 006550#
5598 006600# 001000 000327 15520 CHRGET> JREAD PAST TYPE MARKER
5599 15540 IFN LENGTH=2,<
5600 15560 IFN STRING,< SUI "#"
5601 15580 CPI "#" JIS IT A STRING?

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF MACRO 47([113]) 03112 10-SEP-75 PAGE 22-2
 F3 MAC 6-SEP-64 03111 DIMENSION & VARIABLE SEARCHING

```

  5602          15600    JNZ     NOTSTR   IF NOT VALTYP ALREADY#0
  5603          15620    INR     A         IF[A]=1
  5604          15640    STA     VALTYP   FLAG THIS AS A STRING
  5605          15660    REC
  5606          15680    ADD     C         MAKE [A]+$128
  5607          15700    MOV     C,A      BACK INTO [C] WITH STRING BIT ON
  5608          15720    CHRGET
  5609          15740    NOTSTR: >>
  5610          15760    IFN     LENGTH,< GET FLAG WHETHER TO ALLOW ARRAYS
  5611 006601* 001000 000072 15780    LDA     SUBFLG
  5612 006602* 000000 001601*          IFN     LENGTH,<
  5613 006603* 000000 006575* 15800    ADD     M>
  5614 006604* 001000 000006 15820    CPI     "("
  5615          15840    JZ      ISARY    IF ADD ONTO CHARACTER
  5616 006605* 001000 000376          IFN     LENGTH,<
  5617 006606* 000000 000050 15840    JZ      ISARY    IFKARY PERHAPS (IF SUBFLG SET NEVER WILL MATCH)
  5618 006607* 001000 000312
  5619 006608* 000000 006745*
  5620 006611* 000000 006602*
  5621          15860    IFN     LENGTH,<
  5622 006612* 001000 000257 15880    XHA     A         ISALLOW PARENTS AGAIN
  5623 006613* 001000 000662 15900    STA     SUBFLG>
  5624 006614* 000000 001601*
  5625 006615* 000000 006610*
  5626 006616* 001000 000345
  5627          15920    PUSH    H         ISAVE THE TEXT POINTER
  5628 006617* 001000 000172 15940    LENGTH=2,<
  5629 006620* 001000 000352 15960    MOV     A,D      ISVALUE TYPE INTO [A]
  5630 006621* 001000 001621* 15990    LHLD    VARTAB   I(M,L)=PLACE TO START THE SEARCH
  5631 006622* 000000 006614*
  5632 006623* 001000 000365
  5633 006624* 001000 000353 16000    LUPFND: PUSH  PSW      ISAVE THE VALUE TYPE
  5634 006625* 001000 000056 16020    XCMD
  5635 006626* 000000 001623*
  5636 006627* 001000 0006621*
  5637 006628* 001000 000667
  5638 006631* 001000 000341 16060    COMPAR
  5639 006632* 001000 000312 16080    POP     H         FSEE IF THE END HAS BEEN REACHED
  5640 006633* 000000 006671* 16100    JZ      NUTFNS   F(M)=VALTYP
  5641 006634* 000000 006626*
  5642 006635* 001000 000032 16120    LDAX    D         ISGET THE VALTYP OF THIS SIMPLE VARIABLE
  5643 006636* 001000 000157 16140    MOV     L,A      ISSAVE SO WE KNOW HOW MUCH TO SKIP
  5644 006637* 001000 000274 16160    CMP     M         ISCOMPARE WITH OUR VALTYP
  5645 006638* 001000 000283 16180    INX     D
  5646 006641* 001000 000382 16200    JNZ     NOTIT1   NOT RIGHT KIND -- SKIP IT
  5647 006642* 000000 006668*
  5648 006643* 000000 006653*
  5649 006644* 001000 000032 16220    LDAX    D         I(A)=FIRST CHARACTER OF THIS VARIABLE
  5650 006645* 001000 000271 16240    CMP     C         ISEE IF OUR VARIABLE MATCHES
  5651 006646* 001000 000302 16260    JNZ     NOTIT1
  5652 006647* 000000 006600*
  5653 006650* 000000 006642*
  5654 006651* 001000 000023 16280    INX     D
  
```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF MACRO 47([113]) 03112 10-SEP-75 PAGE 22-3
 F3 MAC 6-SEP-64 03111 DIMENSION & VARIABLE SEARCHING

```

  5655 006652* 001000 000032 16300    LDAX    D         ISEE IF SECOND CHARACTER MATCHES
  5656 006653* 001000 000270 16320    CMP     B
  5657 006654* 001000 000312 16340    JZ      FINPTR   ITHAT WAS IT, ALL DONE
  5658 006655* 001000 000742*
  5659 006656* 000000 006647*
  5660 006657* 001000 000070
  5661          16360    XHD    "#U1000,-076  I'MVI A," AROUND THIS INX SINCE THE POINTER
  5662 006660* 001000 000023 16380    INX     D         IS ALREADY INCREMENTED
  5663 006661* 001000 000023 16400    NOTIT1: INX  D
  5664 006662* 001000 00174 16420    INX     D
  5665 006663* 001000 00174 16440    MOV     A,M      ISKIP OVER THE
  5666          16460    LMDL    ARYTAB   JOURNAL VARIABLE SINCE WE DIDN'T MATCH
  5667 006667* 000000 006623*
  5668 006668* 000000 006655*
  5669 006669* 001000 000031 16520    DAD     D         ISAVE THE VALTYP IN [A]
  5670 006669* 001000 000503 16540    JMP     LUPFND   IADD ON THE POINTER
  5671 006667* 000000 006623*
  5672 006670* 000000 006655*
  5673 006671* 001000 000505 16560    NOTFNS: PUSH  B      ISAVE THE LOOKS
  5674 006672* 001000 000114 16580    MOV     C,H      I(B,C)=LENGTH OF THIS VARIABLE
  5675 006673* 001000 000187 16600    MOV     B,A      I(D,E)=
  5676 006674* 001000 000305 16620    PUSH    B         ISAVE THE VALTYP ON THE STACK
  5677 006675* 001000 000003 16640    INX     B         IMAKE THE LENGTH INCLUDE
  5678          16660    INX     B         ITHE LOOKS TOO
  5679 006676* 001000 000003 16680    INX     B
  5680 006677* 001000 000003 16700    INX     B>
  5681          16720    IFN     LENGTH=2,< PLACE TO STOP SEARCHING
  5682          16730    LMDL    ARYTAB
  5683          16750    XCMD
  5684          16780    LMDL    VARTAB   IGET THE PLACE TO START
  5685          16800    LUPFND: COMPAR  ISEE IF WE ARE THERE
  5686          16820    JZ      NOTFNS   ICOULDN'T FIND THIS VARIABLE
  5687          16840    INX     B         ISDO MAKE ROOM FOR IT
  5688          16860    MOV     A,C
  5689          16880    SUB     M         IIS THIS VARIABLE THE ONE?
  5690          16900    INX     M
  5691          16920    INX     B
  5692          16940    SUB     M         ITRY SECOND CHARACTER MATCHING
  5693          16960    NOTIT1: INX  M
  5694          16980    JZ      FINPTR   ITHAT WAS IT!
  5695          17000    INX     H         ISKIP OVER THAT ONE--NOT IT
  5696          17020    INX     M
  5697          17040    INX     M
  5698          17060    INX     M
  5699          17080    JMP     LUPFNU   ITRY AGAIN
  5700          17100    NOTFNS: PUSH  B         REMEMBER WHAT THIS
  5701          17120    LMDL    STREND   IVARIABLE LOOKS LIKE
  5702          17140    LXI    B,+SCODE> ITHE AMOUNT TO SHOVE
  5703          17160    LMDL    STREND   IEVERYTHING UP BY
  5704          17180    LMDL    STREND   ITHE CURRENT END OF STORAGE
  5705 0066700* 001000 000052
  5706 0066701* 000000 001625*
  5707 0066702* 000000 006667*
  
```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF
F3 MAC 6-SEP-68 03:11

MACRO 47([113]) 03:12 10-SEP-75 PAGE 22-4
DIMENSION & VARIABLE SEARCHING

5706 006705* 001000 000345 17200 PUSH H ISAVE THIS #
5709 006704* 001000 000011 17220 DAD B ADD ON THE AMOUNT OF SPACE
5711 006705* 001000 000301 17240 POP B EXTRA NOT BEING USED
5712 006706* 001000 000345 17260 PUSH B POF OFF HIGH ADDRESS TO MOVE
5713 006707* 001000 000315 17280 CALL BLTU ISAVE NEW CANDIDATE FOR STREND
5714 006710* 000000 002005* 17300 IBLK TRANSFER AND MAKE SURE
5715 006711* 000000 006701* 17320 WE ARE NOT OVERFLOWING THE
5716 006712* 001000 000341 17340 ISTACK SPACE
5718 006713* 001000 000042 17360 POP H I(M,L)=NEW STREND
5720 006711* 000000 001625* SHLD STREND ISTORE SINCE WAS OK
5721 006713* 000000 006710* 17480 ITHERE WAS ROOM, AND BLOCK TRANSFER
5723 006716* 001000 000140 17420 IWAS DONE, SO UPDATE POINTERS
5725 006717* 001000 000151 17440 MOV H,B IGET BACK IN,L POINTING AT THE END
5726 006720* 001000 000042 17460 MOV L,C IOF THE NEW VARIABLE
5727 006721* 000000 000153* 17480 SHLD ARYTAB IUPDATE THE ARRAY TABLE POINTER
5728 006722* 001000 000144 17500 I(M,L) IS RETURNED POINTING TO THE
5729 006723* 001000 000053 17500 ZEROER: DCX H END OF THE VARIABLE SO WE
5730 006724* 001000 000066 17520 MVI M,B
5731 006725* 000000 000000 17540 COMPARE IZERO BACKWARDS TO [D,E] WHICH
5732 006726* 001000 000347 17560 JNZ ZEROER IPOINTS TO THE START OF THE VARIABLE
5733 006727* 001000 000302 17580 I
5734 006730* 000000 006723* 17600 I
5735 006731* 000000 006721* 17620 I
5736 006732* 001000 000321 17640 IFE LENGTH=2,< I
5738 006733* 001000 000163 17660 POP D IPUT VALTYP
5739 006734* 001000 000045 17680 MOV M,E ISTORE AS PART OF THE LOOKS
5740 006735* 001000 000321 17700 INX H
5741 006736* 001000 000163 17720 PUSH D
5742 006737* 001000 000043 17740 MOV M,E IPUT DESCRIPTION
5743 006740* 001000 000162 17760 INX H
5745 006741* 001000 000353 17780 MOV M,D IOF THIS VARIABLE
5746 006741* 001000 000353 17800 XCHG FINPTR INTO [D,E] INTO MEMORY
5747 006742* 001000 000023 17820 IFN LENGTH=2,< I
5748 006743* 001000 000311 17840 INX H IPOINT AT THE VALUE
5749 006744* 001000 000311 17860 FINPTR: XCHG VARIABLE POINTER INTO [D,E]
5750 006745* 001000 000341 17880 POP H IRESTORE THE TEXT POINTER
5751 006744* 001000 000311 17900 RET
5752 006746* 001000 000321 17940 IFE MULDIM,< I
5755 006748* 001000 000321 17960 ISARY: PUSH B IREMEMBER WHAT VARIABLE LOOKS
5756 006749* 001000 000321 17980 I
5757 006750* 000000 000000 18000 IFN STRING,< ILINE
5758 006751* 001000 000321 18020 PUSH H ISAVE THE TXTPTK
5759 006752* 001000 000321 18040 LHLD DIMFLG I(L)=DIMFLG [H]=VALTYP
5760 006753* 001000 000321 18060 XTHL> IPUT VALTYP AND DIMFLG ON THE STACK

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF
F3 MAC 6-SEP-68 03:11

MACRO 47([113]) 03:12 10-SEP-75 PAGE 22-5
DIMENSION & VARIABLE SEARCHING

5761 18000 IAND RESTORE THE TEXT POINTER
5762 18100 IFE STRING,<
5763 18120 LDA DIMFLG ISINCE THIS CALL IS RECURSIVE
5764 18140 PUSH PSW> JDIMFLG MUST BE SAVED ON THE STACK
5765 18140 CALL INTIOX IEVALUATE THE INDEX INTO [D,E]
5766 18180 SYNCHK "0" IMAKE SURE HE CLOSED IT
5767 18200 IFN STRING,< I(L)=DIMFLG [H]=VALTYP
5768 18220 XTHL> ITEXT POINTER ONTO THE STACK
5769 18240 LHLD DIMFLG ISAVE BOTH VALUES BACK
5770 18260 POP H IRESTORE THE TEXT POINTER
5772 18300 IFE STRING,<
5773 18320 POP PSW GET DIMFLG OFF THE STACK
5774 18340 STA DIMFLG RESTORE IT
5775 18360 XTHL> I(M,L) GET VARIABLE DESCRIPTOR
5776 18380 LHLD ARYTAB ITEXT POINTER IS PUT ONTO
5777 18400 XWD "U1000,1 ITHE STACK
5778 18420 XCHG ID(E)=DESCRIPTOR
5779 18440 DAD H I(M,L)=INDEX
5780 18460 INX H IMULTIPLY BY 4 TO GET
5781 18480 PUSH H ISAVE INDEX
5782 18500 LHD STREND ISAVE INDEX
5783 18520 LHLD ARYTAB IPLACE TO START SEARCH
5784 18540 XWD "U1000,1 I"LLXI B," OVER THE NEXT 2
5785 18560 LOPFD02: POP B I(B,C)=LENGTH OF LAST VARIABLE
5786 18580 DAD B ISKIP OVER LAST VARIABLE BY ADDING ITS
5787 18600 XCHG ILENGTH ONTO [H,L]
5788 18620 PUSH H ID(E) GET CURRENT SEARCH POINT
5789 18640 LHD STREND ISAVE THE VARIABLE LOOK
5790 18660 COMPARE GET PLACE TO STOP
5791 18680 XCHG USE IF WE ARE THERE
5792 18700 POP D I(M,L) GETS SEARCH POINT
5793 18720 JZ NOTFDU IPOP OFF VARIABLE LOOKS
5794 18740 PUSHH H ICOULDNT FIND IT
5795 18760 XTHL> IPUT ON LOOKS OF VARIABLE
5796 18780 IWE ARE EXAMINING
5797 18800 XWD "M,L" ON THE STACK AND
5798 18820 LHD STREND ILOOKS FOR VARIABLE WE ARE
5799 18840 COMPARE JEAMINING INTO [H,L]
5800 18860 POP H IIS THIS THE VARIABLE
5801 18880 PUSHH H IPUSH LENGTH OF VARIABLE
5802 18900 XTHL> IBEING EXAMINED ONTO THE STACK
5804 18940 JNZ LOPFD2 IF NO MATCH, GO LOOK SOMEWHERE
5805 18960 LDA DIMFLG IIS THIS VARIABLE TRYING TO BE
5806 18980 ORA A DIMENSIONED AND ALREADY
5807 19000 MVI E,ERR00 IEXISTS?
5808 19020 JNZ ERROR ITTHATS ERROR ERR00
5809 19040 MAKDFN: PUP D IPOP OFF LENGTH OF THIS VARIABLE
5811 19060 DCX D IDECREMENT LENGTH SO WE CAN
5812 19100 IJUST LOOK AT "CARRY" AFTER
5813 19120 ICALLING COMPAR

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05111

MACHO 47(113) 05112 10-SEP-75 PAGE 22-6
DIMENSION & VARIABLE SEARCHING

5014 19140 XTHL ; TRADE POINTER AT VARIABLE WITH
5015 19160 INDEX INTO THE VARIABLE
5016 19180 COMPAN ; SEE IF INDEX IS TOO BIG
5017 19200 MVI E,ERRBS ; IF INDEX ERROR
5018 19220 JNC ERROR ; SINCE LENGTH REALLY HAS AN
5019 19240 ; EXTRA ONE ADDED TO IT
5020 19260 ; IF INDEX-LENGTH DOESN'T CARRY
5021 19280 ; THE IS IN TROUBLE
5022 19300 PUP D ; POP OFF POINTER AT VARIABLE
5023 19320 DAD D ; ADD IT TO THE INDEX
5024 19340 POP D ; POP OFF TEXT POINTER
5025 19360 XCMB ; TEXT POINTER INTO [H,L]
5026 19380 RET ; VARIABLE POINTER INTO (D,E)
5027 19400
5028 19420 NOTFD01 MOV M,E ; INPUT LOOKS DOWN
5029 19440 INX M
5030 19460 MOV M,D
5031 19480 INX M
5032 19500 LXI D,SCODE+44 ; DEFAULT SIZE IS 10
5033 19520 LDA DIMFLG ; FAR WE DIMENSIONING
5034 19540 ORA A
5035 19560 JZ NOTDIM
5036 19580 PUP D ; INPUT INDEX BACK ON
5037 19600 PUSH D
5038 19620 INX D
5039 19640 INX D
5040 19660 INX D
5041 19680 IND D
5042 19700 NOTDIM: PUSH D
5043 19720 MOV M,E ; INPUT LENGTH DOWN
5044 19740 INX M
5045 19760 MOV M,D
5046 19780 INX M
5047 19800 PUSH M
5048 19820 DAD D
5049 19840 CALL REASON ; MAKE SURE WE'RE NOT RUNNING
5050 19860 ; INTO THE STACK
5051 19880 SHLD STRENU ; SETUP NEW STORAGE END
5052 19900 POP D
5053 19920 ZERIT2: DCX H
5054 19940 MVI M,B
5055 19960 COMPAN
5056 19980 JNZ ZERIT2
5057 20000 JMP MAKOFN> ; FINISH UP
5058
5059 20040 PAGE

Format Evaluation

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05111

MACHO 47(113) 05112 10-SEP-75 PAGE 23
MULTIPLE DIMENSION CODE

5060 20000 SUBTTL MULTIPLE DIMENSION CODE
5061 20100 IFN MULDIM<
5062 20120 ;
5063 20140 ; FORMAT OF ARRAYS IN CORE
5064 20160 ;
5065 20180 ; DESCRIPTOR
5066 20200 ; LOW BYTE = SECOND CHARACTER (200 BIT IS STRING FLAG)
5067 20220 ; HIGH BYTE = FIRST CHARACTER
5068 20240 ; LENGTH OF ARRAY IN CORE IN BYTES (DOES NOT INCLUDE DESCRIPTOR)
5069 20260 ; NUMBER OF DIMENSIONS 1 BYTE
5070 20280 ; FOR EACH DIMENSION STARTING WITH THE FIRST A LIST
5071 20300 ; (2 BYTES EACH) OF THE MAX INDICE+1
5072 20320 ; THE VALUES
5073 20340 ;
5074 20360 ISARY: PUSH M ;SAVE DIMFLG AND VALTYP FOR RECURSION
5075 20380 LMD DIMFLG
5076 20400 XTHL ;TEXT POINTER BACK INTO [H,L]
5077 20420 MVI D,0 ;SET # DIMENSIONS = 0
5078 20440 INDLOP: PUSH D ;SAVE NUMBER OF DIMENSIONS
5079 20460 PUSH B ;SAVE LOOKS
5080 20480 CALL INTDX ;EVALUATE INDIC INTO (D,E)
5081 20500 ;
5082 20520 POP B ;POP OFF THE LOOKS
5083 20530 POP PSW ;FA NUMBER OF DIMENSIONS SO FAR
5084 20550 XCHG ;[D,E]=TEXT POINTER
5085 20560 XTHL ;[H,L]=INDICE
5086 20580 ;
5087 20600 PUSH H ;RESAVE VALTYP AND DIMFLG
5088 20620 XCHG ;[H,L]=TEXT POINTER
5089 20640 INR A ;INCREMENT OF DIMENSIONS
5090 20660 MOV D,A ;#NUMBER OF DIMENSIONS
5091 206700 ; 001000 000317 20680 MOV A,M ;GET TERMINATING CHARACTER
5092 206710 ; 001000 000317 20690 CPI 44 ;A COMMA SO MORE INDICES FOLLOW?
5093 206720 ; 001000 000317 20700 JZ INDLOP ;IF SO, READ MORE
5094 206730 ; 001000 000317 20710 ;
5095 206740 ; 001000 000317 20720 SYNCHK "}" ;MAKE SURE IT ENDED PROPERLY
5096 206750 ; 001000 000317 20730 ;
5097 206760 ; 001000 000317 20740 SHLD TEMP2 ;SAVE THE TEXT POINTER
5098 206770 ; 001000 000317 20750 ;
5099 206780 ; 001000 000317 20760 ;
5100 206790 ; 001000 000317 20770 ;
5101 206800 ; 001000 000317 20780 ;
5102 206810 ; 001000 000317 20790 ;
5103 206820 ; 001000 000317 20800 PUP H ;[H,L]=VALTYP & DIMFLG
5104 206830 ; 001000 000317 20810 SHLD DIMFLG ;SAVE VALTYP AND DIMFLG
5105 206840 ; 001000 000317 20820 ;
5106 206850 ; 001000 000317 20830 ;
5107 206860 ; 001000 000317 20840 PUSH D ;SAVE NUMBER OF DIMENSIONS

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF
F3 MAC 6-SEP-64 05111

MACHO 47(113) 03112 10-SEP-75 PAGE 23=1
MULTIPLE DIMENSION CODE

5913 ;
5914 ;
5915 ;
5916 ;
5917 007011* 001000 000052
5918 007012* 000000 001623*
5919 007013* 000000 007006*
5920 007014* 001000 000076
5921 007015* 001000 000031
5922 ;
5923 007016* 001000 000555
5924 007017* 001000 000506
5925 007020* 000000 001625*
5926 007021* 000000 007612*
5927 007022* 001000 000353
5928 007023* 001000 000347
5929 007024* 001000 000072
5930 007025* 001000 001543*
5931 007026* 001000 000000*
5932 007027* 001000 000000*
5933 007028* 001000 000312
5934 007029* 001000 007104*
5935 007031* 000000 007025*
5936 ;
5937 007032* 001000 000276
5938 007033* 001000 000043
5939 007034* 001000 000302
5940 007035* 000000 000050*
5941 007036* 000000 007039*
5942 007037* 001000 000176
5943 007040* 001000 000271
5944 007041* 001000 000043
5945 007042* 001000 000302
5946 007043* 000000 007051*
5947 007044* 000000 007035*
5948 007045* 001000 000176
5949 007046* 001000 000270
5950 ;
5951 007047* 001000 000076
5952 007050* 001000 000043
5953 007051* 001000 000043
5954 007052* 001000 000136
5955 007053* 001000 000043
5956 007054* 001000 000126
5957 007055* 001000 000133
5958 007056* 001000 000302
5959 007057* 000000 007015*
5960 007060* 000000 007043*
5961 ;
5962 007061* 001000 000072
5963 007062* 000000 001542*
5964 007063* 000000 007057*
5965 007064* 001000 000267
5966 ;
5967 007065* 001000 000036
5968 007066* 000000 000012
5969 007067* 001000 000302
5970 007068* 000000 002102*
5971 ;
5972 ;
5973 ;
5974 ;
5975 ;
5976 ;
5977 007072* 001000 000361
5978 007073* 001000 000276
5979 ;
5980 ;
5981 007074* 001000 000312
5982 007075* 000000 007245*
5983 007076* 000000 007070*
5984 ;
5985 007077* 001000 000039
5986 007100* 000000 000011
5987 007101* 001000 000303
5988 007102* 000000 002102*
5989 007103* 000000 007075*
5990 ;
5991 ;
5992 ;
5993 ;
5994 ;
5995 ;
5996 ;
5997 ;
5998 ;
5999 ;
6000 ;
6001 ;
6002 ;
6003 ;
6004 ;
6005 ;
6006 ;
6007 ;
6008 ;
6009 ;
6010 ;
6011 ;
6012 ;
6013 ;
6014 ;
6015 007104*
6016 ;
6017 007104* 001000 000167
6018 007105* 001000 000043

20860 ;
20880 ; AT THIS POINT [B,C]=LOOKS, THE TEXT POINTER IS IN TEMP2,
20900 ; THE INDICES ARE ALL ON THE STACK, FOLLOWED BY THE NUMBER OF DIMENSIONS,
20920 ;
20940 LHLD ARYTAB ; I[M,L]=PLACE TO START THE SEARCH
20960 XWD "01000,-076 ;"MV A," AROUND THE NEXT BYTE,
20980 LDPPDA: DAD 0 ISKIP OVER THIS ARRAY SINCE IT'S
21000 ; NOT THE ONE
21020 XCHG ; ID[E]=CURRENT SEARCH POINT
21040 LHLD STRENQ ; GET THE PLACE TO STOP INTO [M,L]
21050 XCHG ; I[M,L]=SEARCH POINT
21060 COMPAR ; ISSTOPPING TIME?
21080 LENGTH=2,<
21100 IFE LDA VALTYP> ;
21120 JZ NOTFOU ; YES, COULDN'T FIND THIS ARRAY
21140 IFE LENGTH=2,<
21160 CMP M ; SEE IF THE VALTYP ARE THE SAME
21180 INX M
21200 JNZ NMARY2> ;
21220 MOV A,M ; GET FIRST CHARACTER
21240 CMP C ; SEE IF IT MATCHES
21260 INX M
21280 JNZ NMARY1 ; NOT THIS ONE
21300 ;
21320 MOV A,M ; GET SECOND CHARACTER
21340 CMP C ; AND OTHER MATCH?
21360 IFE LENGTH=2,<
21380 XWD "01000,-076 ; ISKIP THIS INCREMENT WITH "MV A,"
21400 NMARY2: INX H ;
21420 NMARY1: INX H ; POINT TO SIZE ENTRY
21440 MOV E,M ; ID[E]=LENGTH
21460 INX H ; FOR THE ARRAY BEING LOOKED AT
21480 MOV D,M
21500 INX H
21520 JNZ LUFPDA ; IF NO MATCH, SKIP THIS ONE
21540 ;
21560 LUA DIMFLG ; AND TRY AGAIN
21580 ; SEE IF CALLED BY "DIM"
21600 MVI E,ERRD ; "DOUBLY DIMENSIONED" ERROR
21620 JNZ ERROR ;
21640 ;
21660 ; TEMP2=THE TEXT POINTER
21680 ; WE HAVE LOCATED THE VARIABLE WE WERE LOOKING FOR
21700 ; AT THIS POINT [M,L] POINTS BEYOND THE SIZE TO THE NUMBER OF DIMENSIONS
21720 ; THE INDICES ARE ON THE STACK FOLLOWED BY THE NUMBER OF DIMENSIONS
21740 ;
21760 POP PSW ; I[A]=NUMBER OF DIMENSIONS
21780 CMP H ; MAKE SURE THE NUMBER GIVEN NOW AND
21800 ; AND WHEN THE ARRAY WAS SET UP ARE THE
21820 ; ISAME
21840 JZ GETDEF ; JUMP OFF AND READ
21860 ;
21880 BSERR: MVI E,ERRBS ; "THE INDICES...,"
21900 ; "SUBSCRIPT OUT OF RANGE"
21920 ;
21940 ; HERE WHEN VARIABLE IS NOT FOUND IN THE ARRAY TABLE
21960 ;
21980 ; BUILDING AN ENTRY:
22000 ;
22020 ; PUT DOWN THE DESCRIPTOR
22040 ; SETUP NUMBER OF DIMENSIONS
22060 ; MAKE SURE THERE IS ROOM FOR THE NEW ENTRY
22080 ; REMEMBER VARPTR
22100 ; TALLY#4 (VALTYP FOR THE EXTENDED)
22120 ; SKIP 2 LOC'S FOR LATER FILL IN -- THE SIZE
22140 ; LOOP: GET AN INDICE
22160 ;
22180 ; TALLY#4 +1 DOWN AT VARPTR AND INCREMENT VARPTR
22200 ; TALLY#4 +1 NUMBER+1
22220 ; DECREMENT NUMBER+DIMS
22240 ; JNZ LOOP
22260 ; CALL REASON WITH [M,L] REFLECTING LAST LOC OF VARIABLE
22280 ; UPDATE STREND
22300 ; ZERO BACKWARDS
22320 ; MAKE TALLY INCLUDE MAXDIMS
22340 ; PUT DOWN TALLY
22360 ; IF LAYERED BY DIMENSION, RETURN
22380 ; OTHERWISE INDEX INTO THE VARIABLE AS IF IT
22400 ; WERE FOUND ON THE INITIAL SEARCH
22420 NUTFOU: IFE LENGTH=2,<
22440 MOV M,A ; INPUT DOWN THE VARIABLE TYPE
22460 INX H

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6=SEP=64 05111

MACRO 47(113) 05112 10-SEP-75 PAGE 23-3
MULTIPLE DIMENSION CODE

6019 007105* 001000 000137 22500 MOV E,A
6020 007107* 001000 000206 22520 MVI D,D> ;(D,E)=SIZE OF ONE VALUE (VALTYP)
6021 007110* 001000 000000 22540 IFN LENGTH=2,<
6022 007111* 001000 000161 22560 LXI D,SCODE+4>
6023 007112* 001000 000043 22580 MOV M,C ;INITIALIZE TALLY TO FOUR
6024 007113* 001000 000160 22600 INX H
6025 007114* 001000 000043 22620 MOV M,B
6026 007115* 001000 000361 22640 INX H
6027 007116* 001000 000052 22660 PUP PSW ;(A)=NUMBER OF DIMENSIONS
6028 007117* 001000 000052 22680 STA TEMP6 ;SETUP GETSTK CALL
6029 007118* 001000 000052 22700 CALL GETSTK ;GET SPACE FOR DIMENSION ENTRIES
6030 007119* 001000 000315 22720 TEMP6: PCHL ;PLACE TO STORE NUMBER OF DIMENSIONS
6031 007120* 001000 000000 22740 ;FOR GETSTK AND LATER RECALL
6032 007121* 001000 000000 22760 SHLD TEMP3 ;IIIMPURE!! PCHL TO CONFUSE DISASSEMBLY
6033 007122* 001000 0002024 22780 ;SAVE THE LOCATION TO PUT THE SIZE
6034 007123* 001000 0007117*
6035 007124* 001000 000351 22720 TEMP6: PCHL ;PLACE TO STORE NUMBER OF DIMENSIONS
6036 007125* 001000 000000 22740 ;FOR GETSTK AND LATER RECALL
6037 007126* 001000 000160 22760 SHLD TEMP3 ;IIIMPURE!! PCHL TO CONFUSE DISASSEMBLY
6038 007127* 001000 0007122*
6039 007128* 001000 0001575*
6040 007129* 001000 0007122*
6041 007130* 001000 000000 22800 FIN
6042 007131* 001000 0000043 22820 INX H ;SKIP OVER THE SIZE LOCATIONS
6043 007131* 001000 0000043 22840 INX H
6044 007132* 001000 000101 22860 MOV B,C ;(B)=NUMBER OF DIMENSIONS
6045 007133* 001000 000000 22880 ;THIS DEPENDS ON THE FACT THAT GETSTK
6046 007134* 001000 000160 22900 ;RETURNS ITS ARGUMENT IN [C]
6047 007135* 001000 000000 22920 MOV M,B ;STORE THE NUMBER OF DIMENSIONS
6048 007134* 001000 0000043 22940 INX H
6049 007135* 001000 0000072 22960 LOPPTA: LUA DIMFLG ;CALLED BY DIMENSION?
6050 007135* 001000 0001542*
6051 007137* 001000 0007126*
6052 007140* 001000 0000267 22980 DRA A
6053 007141* 001000 0000170 23000 MOV A,B ;(A)=NUMBER OF DIMENSIONS
6054 007142* 001000 000000 23020 LXI B,SCODE+11 ;ASSUME ITS NOT "DIM"
6055 007143* 001000 000013*
6056 007144* 001000 0007136*
6057 007145* 001000 0000312 23040 JZ NOTDIM ;DEFAULT DIMENSIONS TO TEN
6058 007145* 001000 0007152*
6059 007147* 001000 000000 23060 POP B ;POP OFF AN INDICE INTO [B,C]
6060 007151* 001000 0000003 23080 INX B ;ADD ONE TO IT FOR THE ZERO ENTRY
6061 007151* 001000 0000161 23100 MOV M,C ;PUT THE MAXIMUM DOWN
6062 007151* 001000 0000003 23120 INX H
6063 007151* 001000 0000160 23140 MOV M,B
6064 007151* 001000 0000169 23160 INX H
6065 007151* 001000 0000003 23180 PUSH PSW ;SAVE THE NUMBER OF DIMENSIONS
6066 007150* 001000 0000365 23200 PUSH H ;SAVE THE POINTER INTO THE NEW ENTRY
6067 007151* 001000 0000345 23220 CALL UMULT ;MULTIPLY [B,C]=NEWMAX BY CURTOL=[D,E]
6068 007160* 001000 0000315 23220
6069 007161* 001000 0000000*
6070 007162* 001000 0007146*
6071 007163* 001000 0000353 23240 XCMB ;(D,E)=NEW CURTOL

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6=SEP=64 05111

MACRO 47(113) 05112 10-SEP-75 PAGE 23-4
MULTIPLE DIMENSION CODE

6072 007164* 001000 0000341 23260 POP H ;GET THE POINTER INTO THE ENTRY BACK
6073 007165* 001000 0000301 23280 POP B ;GET THE NUMBER OF DIMENSIONS BACK
6074 007165* 001000 0000005 23300 DCR B ;DECREMENT THE NUMBER OF DIMENSIONS LEFT
6075 007165* 001000 0000302 23320 JNZ LUPPTA ;HANDLE THE OTHER INDICES
6076 007166* 001000 0007155*
6077 007171* 001000 0007161*
6078 007172* 001000 0000102 23340 MOV B,D ;(B,C)=SIZE
6079 007173* 001000 0000113 23360 MOV C,E
6080 007174* 001000 0000353 23380 XCHG ;(D,E)=START OF VALUES
6081 007175* 001000 0000031 23400 DAD D ;(H,L)=END OF VALUES
6082 007176* 001000 0000332, 23420 JC BSERR ;OUT OF MEMORY POINTER BEING GENERATED?
6083 007177* 001000 000000 007077*
6084 007200* 001000 000000 23440 CALL REASON ;SEE IF THERE IS ROOM FOR THE VALUES
6085 007201* 001000 0000315 23460 ;
6086 007202* 001000 0002045*
6087 007203* 001000 0007177*
6088 007204* 001000 0000042 23480 SHLD STREND ;UPDATE THE END OF STORAGE
6089 007205* 001000 0001625*
6090 007206* 001000 000000 007202*
6091 007207* 001000 0000053 23480 ZERITA: DCX H ;ZERO THE NEW ARRAY
6092 007211* 001000 0000006 23500 MVI M,B
6093 007211* 001000 0000000 23520 ;
6094 007212* 001000 0000347 23520 CUMPAR ;BACK AT THE BEGINNING?
6095 007213* 001000 0000302 23540 JNZ ZERITA ;NO,ZERO MORE
6096 007214* 001000 0007207*
6097 007215* 001000 00007205*
6098 007216* 001000 0000003 23560 INX B ;ADD ONE TO THE SIZE TO INCLUDE
6099 007217* 001000 0000053 23580 ;THE BYTE FOR THE NUMBER OF DIMENSIONS
6100 007217* 001000 0000147 23600 MOV H,A ;(H)=ZERO
6101 007220* 001000 0000072 23620 LDA DIMFLG
6102 007221* 001000 0001542*
6103 007222* 001000 0007214*
6104 007223* 001000 0000067 23640 DRA A ;ARE WE DIMENSIONING?
6105 007224* 001000 0000072 23660 LDA TEMP6 ;GET THE NUMBER OF DIMENSIONS
6106 007225* 001000 0007124*
6107 007226* 001000 0000000 007221*
6108 007227* 001000 0000157 23680 MOV L,A ;(L)=NUMBER OF DIMENSIONS
6109 007230* 001000 0000051 23700 DAD H ;(M,L)=NUMBER OF DIMENSIONS TIMES TWO
6110 007231* 001000 0000011 23720 DAU B ;ADD UP THE SIZE
6111 007232* 001000 0000000 23740 ;TO GET THE TOTAL NUMBER OF BYTES USED
6112 007232* 001000 0000353 23760 XCHG ;(D,E)=TOTAL SIZE
6113 007233* 001000 0000056 23780 LHLD TEMP3 ;PLACE TO STORE SIZE
6114 007234* 001000 0001575*
6115 007235* 001000 0000000 007225*
6116 007236* 001000 0000163 23800 MOV M,E ;PUT DOWN THE SIZE
6117 007237* 001000 0000043 23820 INX H
6118 007238* 001000 0000162 23840 MOV M,D
6119 007239* 001000 0000043 23860 INX H
6120 007240* 001000 0000302 23880 JNZ FINNOW
6121 007243* 001000 0007551*
6122 007244* 001000 0007234*
6123 ;
6124 ; AT THIS POINT [H,L] POINTS BEYOND THE SIZE TO THE NUMBER OF DIMENSIONS

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF
F3 MAC 6-SEP-64 0511

MACHO 47(113) 05112 10-SEP-75 PAGE 23-5
MULTIPLE DIMENSION CODE

6125 23940 ; STRATEGY:
6126 23960 ; NUMDIM=NUMBER OF DIMENSIONS
6127 24000 ;
6128 24000 ; INLPNMIGET A NEW INDICE
6129 24020 ; POP NEW MAX INTO CURMAX
6130 24040 ; MAKE SURE INDICE IS NOT TOO BIG
6131 24060 ; MULTIPLY CURTOL BY CURMAX
6132 24080 ; ADD INDICE TO CURTOL
6133 24100 ; NUMDIM*NUMDIM=1
6134 24120 ; JNZ INLPNM
6135 24140 ; USE CURTOL#4 (VALTYP FOR EXTENDED) AS OFFSET
6136 24160 ;
6137 007251* 001000 000043 24180 GETDEF: INX H ;POINT PAST THE NUMBER OF DIMENSIONS
6138 007246* 001000 000001 24200 LXI B,SCODE ;CURTOL=ZERO
6139 007277* 001000 000000*
6140 007250* 000000 007243*
6141 007251* 001000 000026 24220 XWD "U1000,-026 ;"MVI D," AROUND THE NEXT BYTE
6142 007252* 001000 000341 24240 INLPNM: POP H ;I(H,L)= POINTER INTO VARIABLE ENTRY
6143 007253* 001000 000136 24260 MOV E,M ;{D,E}=MAXIMUM FOR THE CURRENT INDICE
6144 007254* 001000 000132 24280 INX H
6145 007255* 001000 000126 24300 MOV W,D,M
6146 007256* 001000 000043 24320 INX H
6147 007257* 001000 000343 24340 XTHL ;I(H,L)=CURRENT INDICE
6148 007260* 001000 000365 24360 PUSH PSW ;POINTER INTO THE VARIABLE GOES ON THE STACK
6149 007261* 001000 000347 24380 COMPARE ;ISAVE THE NUMBER OF DIMENSIONS
6150 007262* 001000 000522 24400 JNC BSERR ;SEE IF THE CURRENT INDICE IS TOO BIG
6151 007263* 001000 000777*
6152 007264* 001000 0007247*
6153 007265* 001000 000345 24440 PUSH H ;ISAVE THE CURRENT INDICE
6154 007266* 001000 000315 24460 CALL UMULT ;CURTOL=CURTOL*CURRENT MAXIMUM
6155 007267* 001000 007161*
6156 007270* 001000 007263*
6157 007271* 001000 000321 24480 POP D ;INDICE INTO {D,E}
6158 007272* 001000 000031 24500 ADD D,D ;ADD THE INDICETO CURTOL
6159 007273* 001000 001543*
6160 007274* 001000 00075 24520 POP PSW ;GET THE NUMBER OF DIMENSIONS IN {A}
6161 007275* 001000 000175 24540 DCR A ;SEE IF ALL THE INDICES HAVE BEEN PROCESSED
6162 007275* 001000 000104 24560 MOV B,M ;{B,C}={CURTOL IN CASE WE LOOP BACK
6163 007276* 001000 000115 24580 MOV C,L
6164 007277* 001000 000302 24600 JNZ INLPNM ;PROCESS THE REST OF THE INDICES
6165 007300* 000000 007252*
6166 007301* 000000 007267*
6167 007302* 001000 000072 24620 IFE LENGTH=2,<
6168 007303* 001000 001543* 24640 LDA VALTYP ;ISAVE HOW BIG THE VALUES ARE
6169 007304* 000000 007300*
6170 007305* 001000 000000*
6171 007306* 001000 000104 24660 FAND MULTIPLY BY THAT SIZE
6172 007307* 001000 0000104 24680 MOV B,M ;ISAVE THE ORIGINAL VALUE FOR MULTIPLYING
6173 007308* 001000 000115 24700 MOV C,L ;BY THREE
6174 007309* 001000 000051 24720 DAD M ;MULTIPLY BY TWO AT LEAST
6175 007310* 001000 000326 24740 SUI 4 ;FOR INTEGERS AND STRINGS
6176 007311* 000000 000004 24760
6177 007304* 000000 007267*
6178 007305* 001000 000000*
6179 007306* 001000 000000*
6180 007307* 001000 000000*
6181 007308* 001000 000000*
6182 007309* 001000 0000512 24780 DAD H ;NOW MULTIPLIED BY FOUR
6183 007310* 001000 0007326* 24800 JZ D0NHUL ;IF SINGLE ALL DONE
6184 007320* 000000 007315*
6185 007321* 001000 000051 24820 JC SMLVAL
6186 007322* 001000 0000542 24840 SMLVAL: DAD H ;BY EIGHT FOR DOUBLES
6187 007323* 000000 007326*
6188 007324* 000000 007317*
6189 007325* 001000 0000011 24880 DAD B ;FOR STRINGS
6190 007326* 000000 000000*
6191 007327* 001000 000000*
6192 007328* 001000 000000*
6193 007329* 001000 000000*
6194 007326* 001000 000501 24900 D0NHUL:> ;ADD IN THE ORIGINAL
6195 25000
6196 007327* 001000 000011 25020 IFN LENGTH=2,< ;MULTIPLY CURTOL BY FOUR
6197 007328* 001000 000000*
6198 007330* 001000 000353 25040 DAD H ;POP OFF THE ADDRESS OF WHERE THE VALUES
6199 007331* 001000 0000052 25060 DAD H ;BEGIN
6200 007332* 000000 001685*
6201 007333* 000000 007323*
6202 007334* 001000 000055 25100 DCX H ;IREAD THE TERMINATING CHARACTER
6203 007335* 001000 000527 25120 CHRGET ;CHARGE
6204 007336* 001000 000311 25140 RET>
6205 25160 PAGE

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF MACRO 47(113) 03:12 10-SEP-75 PAGE 24
 F3 MAC 6-SEP-64 03:11 FRE FUNCTION AND INTEGER TO FLOATING ROUTINES

```

 6207          25200 SUBTLL FRE FUNCTION AND INTEGER TO FLOATING ROUTINES
 6208          25200 IFN LENGTH,<
 6209 007337* 001000 000052 25200 IFE LHD STREND      ;GET END OF VARIABLE AND TEXT SPACE
 6210 007340* 000000 001629* 25200 XCHG
 6211 007341* 000000 00733c* 25200 LXI H,SCODE      ;PUT IT IN [D,E] FOR SUBTRACTION
 6212 007342* 001000 000353 25200
 6213 007343* 001000 000041 25200
 6214 007344* 000000 00000f 25200
 6215 007345* 000000 007348* 25200
 6216 007346* 001000 000071 25200 DAD SP           ;PUT THE STACK POINTER IN [H,L]
 6217          25200 IFN STRING,<
 6218          25200 IFE LENGTH=2,<                ;GET TYPE
 6219 007347* 001000 000315 25200 CALL GETTYPE
 6220 007350* 000000 000050? 25200
 6221 007351* 000000 007344* 25200 JZ GIVDBL>
 6222 007352* 001000 000302 25370 JNZ GIVDBL>
 6223 007353* 000000 007372* 25370
 6224 007354* 000000 007356* 25370
 6225          25370 IFN LENGTH=2,<
 6226          25370 IFE LD A,VALTYP      ;HAS THE ARGUMENT A STRING?
 6227          25370 OPA A
 6228          25440 JZ GIVDBL>
 6229 007355* 001000 000315 25440 CALL FREFAC      ;NO, GIVE FREE VARIABLE SPACE
 6230 007356* 000000 010434* 25440
 6231 007357* 000000 007353* 25440
 6232          25440 CALL GARBAZ      ;TO GIVE FREE STRING SPACE
 6233 007360* 001000 000315 25500
 6234 007361* 000000 010004? 25500
 6235 007362* 000000 007356* 25500
 6236 007363* 001000 000052 25500 LHD STKTOP      ;BOTTOM OF FREE AREA
 6237 007364* 000000 001615* 25500
 6238 007365* 000000 007361* 25500
 6239 007366* 001000 000353 25540 XCHG
 6240 007367* 001000 000052 25540 LHD FRETUP>>    ;TOP OF FREE AREA
 6241 007370* 000000 001573* 25540
 6242 007371* 000000 007364* 25540
 6243          25540
 6244          25540 ; THIS ROUTINE SUBTRACTS [D,E] FROM [M,L]
 6245          25540 ; AND FLOATS THE RESULT LEAVING IT IN FAC,
 6246          25540 ;
 6247          25660 IFE LENGTH=1,<
 6248          25660 GIVUBL: MOV A,L      ;DO THE SUBTRACTION
 6249          25760 SUB E
 6250          25760 MOV C,A
 6251          25760 MOV A,H
 6252          25760 SBB D
 6253          25780 GIVACF1 MOV B,C>
 6254          25790 IFN LENGTH=2,<
 6255          25790 GIVABF: MOV D,B
 6256          25820 MVI E,0      ;GET ZERO IN LOW
 6257          25840 IFN STRING,<
 6258          25840 LXI H,VALTYP      ;FLAG VALUE TYPE AS NUMERIC
 6259          25860 MOV H,E>

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF MACRO 47(113) 03:12 10-SEP-75 PAGE 24-1
 F3 MAC 6-SEP-64 03:11 FRE FUNCTION AND INTEGER TO FLOATING ROUTINES

```

 6260          25900 MVI B,144      ;SETUP TO FLOAT [B,C]
 6261          25920 JMP FLDRTR>
 6262          26120 IFE LENGTH=2,<
 6263 007372* 001000 000175 26120 GIVUBL: MOV A,L      ;[M,L]=[M,L]-[D,E]
 6264 007373* 001000 000223 26160 SUB E
 6265 007374* 001000 000157 26180 MOV L,A
 6266 007375* 001000 000174 26200 MOV A,M
 6267 007376* 001000 000232 26220 SBB D
 6268 007377* 001000 000021 26240 XWD "01000,-021   ;SKIP THE NEXT TWO BYTES WITH "LXI D,""
 6269 007400* 001000 000157 26260 SNGFLT: MOV L,A      ;MAKE [A] AN UNSIGNED INTEGER
 6270 007401* 001000 000257 26280 XRA A
 6271 007402* 001000 000047 26300 GIVINT: MOV H,A
 6272 007403* 001000 000503 26320 JMP HAKINT>
 6273 007424* 000000 000471* 26322 IFN LENGTH,<
 6274 007425* 000000 007370* 26322 IFE LPTSW,<
 6275          26324 LPDS: LDA LPTPOS
 6276          26326 JMP SNGFLT>
 6277          26328 POS: LDA TTYPOS      ;GET TELETYPE POSITION
 6278 007406* 001000 000072 26330
 6279 007407* 000000 000047* 26332 IFN LENGTH=2,<
 6280 007408* 000000 000047* 26332 SNGFLT: MOV B,A      ;RETURN FLOATING 1 BYTE
 6281 007410* 000000 007404* 26333 XRA A      ;UNSIGNED FROM A
 6282          26333 JMP GIVABF>>    ;GIVING 0255
 6283          26334
 6284          26336
 6285          26338
 6286          26360 PAGE

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF F3 MAC 6-SEP-64 05:11 MACRO 47(113) 03:12 10-SEP-75 PAGE 25 SIMPLE=USER=DEFINED=FUNCTION CODE

```

26268          26380      SUBTTL SIMPLE=USER=DEFINED=FUNCTION CODE
26269          26390      IFN FUNCTS,<
26270          26420      ; NOTE ONLY SINGLE ARGUMENTS ARE ALLOWED TO FUNCTIONS
26271          26440      ; AND FUNCTIONS MUST BE OF THE SINGLE LINE FORM:
26272          26460      ; DEF FN(A)XX?XX?
26273          26480      ; NO STRINGS CAN BE INVOLVED WITH THESE FUNCTIONS
26274
26275          26520      ; IDEA! CREATE A FUNNY SIMPLE VARIABLE ENTRY
26276          26540      ; WHOSE FIRST CHARACTER (SECOND WORD IN MEMORY)
26277          26560      ; HAS THE 208 BIT SET,
26278          26580      ; THE VALUE WILL BE:
26279          26600      ;
26280          26620      ; A TXTPTR TO THE FORMULA
26281          26640      ; A PTR TO THE ARGUMENT VARIABLE
26282          26660      ;
26283          26680      ; FUNCTION NAMES CAN BE LIKE "FNA4"
26284          26700      ;
26285
26286 007411* 001000 000315          26780      DEF: CALL GETFNH    ;GET A POINTER TO THE
26287 007412* 000000 007550*          26780      IFN LENGTH=2,<             ;FUNCTION VARIABLE
26288 007413* 000000 007407*          26780      LXI B,DATA              ;EVENTUALLY RETURN TO "DATA"
26289          26800      ;
26290          26820      LXI B,DATA              ;FUNCTION VARIABLE
26291          26840      PUSH B                ;AND SKIP THE FORMULA
26292          26860      PUSH D                ;SAVE A POINTER TO IT
26293          26880      CALL ERDRIK            ;DEF IS "ILLEGAL DIRECT"
26294
26295 007414* 001000 000001          26880      SYNCHK "("             ;MUST HAVE "("
26296 007415* 000000 004072*          26880      SYNCHK ")"             ;MUST CLOSE IT WITH ")"
26297 007416* 001000 007412*          26880      ;
26298 007417* 001000 000305          26880      SYNCHK EQUALTK        ;MUST HAVE EQUAL
26299 007418* 001000 000325          26880      MOV B,H                ;PUT THE TXTPTR ON THE STACK
26300 007419* 001000 000051          26880      MOV C,L                ;;(B,L)=PTR TO FUNCTION VARIABLE
26301 007420* 001000 000317          26880      XTHL                 ;;(B,C)=TXTPTR
26302 007421* 001000 000050          26880      ;
26303 007422* 001000 000315          26900      SYNCHK "("             ;SINCE WE STORE A TEXT POINTER
26304 007423* 001000 007415*          26900      SYNCHK ")"             ;GET POINTER TO ARGUMENT
26305 007424* 001000 000317          26900      CALL PTRGET            ;GET POINTER TO ARGUMENT
26306 007425* 000000 000050          26900      ;
26307 007426* 001000 000315          26920      ;
26308 007427* 001000 000505          26920      ;
26309 007428* 000000 007422*          26940      ;
26310
26311          26960      IFN LENGTH=2,<             ;AND GO TO "DATA" SKIPPING THE
26312          26980      IFN STRING,<CALL CHKNUM>>ISTRINGS ILLEGAL
26313          27000      SYNCHK ")"             ;MUST CLOSE IT WITH ")"
26314
26315 007431* 001000 000317          27020      SYNCHK EQUALTK        ;MUST HAVE EQUAL
26316 007432* 000000 000051          27020      MOV B,H                ;PUT THE TXTPTR ON THE STACK
26317 007433* 001000 000317          27020      MOV C,L                ;;(B,L)=PTR TO FUNCTION VARIABLE
26318 007434* 000000 000260          27020      XTHL                 ;;(B,C)=TXTPTR
26319 007435* 001000 000104          27040      ;
26320 007436* 001000 000115          27060      ;
26321 007437* 001000 000343          27080      ;
26322 007438* 001000 000305          27100      ;
26323 007439* 001000 007550*          27120      ;
26324 007440* 001000 000317          27140      JMP DEFFIN            ;PUT DOWN THE TEXT=POINTER
26325 007441* 001000 007521*          27160      ;
26326 007442* 000000 007427*          27180      ;
26327
26328          27160      ;
26329          27180      ;
26330
26331
26332
26333
26334
26335
26336
26337
26338
26339
26340

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF F3 MAC 6-SEP-64 05:11 MACRO 47(113) 03:12 10-SEP-75 PAGE 25+1 SIMPLE=USER=DEFINED=FUNCTION CODE

```

6341          27200      ;
6342          27220      ;
6343          27240      ;
6344 007443* 001000 000315          27260      FNDDER: CALL GETFNH    ;GET A POINTER TO
6345 007444* 000000 007550*          27260      FNDDER: CALL GETFNH    ;THE FUNCTION DEFINITION IN [D,E]
6346 007445* 000000 007441*          27260      FNDDER: CALL GETFNH    ;GET A POINTER TO
6347          27280      ;
6348 007446* 001000 000325          27300      PUSH D                ;SAVE THE POINTER
6349 007447* 001000 000315          27320      CALL PARMK            ;EVALUATE THE VALUE TO BE PASSED
6350 007448* 001000 006136*          27340      IFN LENGTH=2,<             ;THE FUNCION DEFINITION IN [D,E]
6351 007451* 000000 007444*          27340      IFN STRING,<CALL CHKNUM>>JARG CANNOT BE STRING
6352          27360      XTHL                 ;;(M,L)=POINTER TO FUNCTION DEF
6353          27380      PUSHM                ;TEXT POINTER GOES ON THE STACK
6354 007452* 001000 000343          27400      PUSHM                ;PUSH THE POINTER AT THE FORMULA
6355 007453* 001000 000367          27420      XTHL                 ;INTO THE STACK
6356 007454* 001000 000321          27440      POP D                ;;(D,E)=PTR TO FORMULA
6357 007455* 001000 006367          27460      PUSHM                ;;(M,L)=POINTER TO THE
6358 007456* 001000 000341          27480      POP M                ;ARGUMENT ON THE STACK
6359 007457* 001000 000367          27500      PUSHFM               ;SAVE ARGS OLD VALUE ON THE STACK
6360 007458* 001000 000367          27520      ;
6361 007460* 001000 000053          27540      DCX H                ;DCX H
6362 007461* 001000 000053          27560      DCX H                ;DCX H
6363 007462* 001000 000053          27580      DCX H                ;POINT TO FRONT OF ARG AGAIN
6364 007463* 001000 000053          27600      DCX H                ;DCX H
6365 007464* 001000 000053          27620      DCX H                ;DCX H
6366 007465* 001000 006345          27640      PUSH H                ;SAVE IT
6367 007466* 001000 000347          27660      COMPAR              ;ISHOULDN'T BE EQUAL UNLESS
6368 007467* 001000 000347          27680      ;FUNCTION WAS NEVER DEFINED
6369 007468* 001000 000036          27700      PUSH D                ;SAVE FORMULA TEXT POINTER
6370 007469* 001000 000036          27720      MVI E,ERRRUF         ;NOW [D,E] FREE SO CHECK IF (ZERO) SET
6371 007470* 001000 000036          27740      JZ ERROR             ;JZ ERROR
6372 007471* 001000 000022          27760      ;
6373 007472* 001000 000312          27780      CALL NUVHF            ;PUT CURRENT FAC INTO OUR ARG VARIABLE
6374 007473* 000000 002102*          27780      ;
6375 007474* 000000 007458*          27780      ;
6376 007475* 000000 000000          27780      ;
6377 007476* 001000 000315          27780      ;
6378 007477* 000000 007473*          27780      ;
6379 007478* 001000 000000          27780      ;
6380
6381 007500* 001000 000541          27800      POP H                ;POP OFF FORMULA TXTPTR
6382          27810      IFN LENGTH=2,<             ;POP OFF FORMULA TXTPTR
6383          27820      CALL FRMNUM            ;EVALUATE IT AND MUST SURE ITS NUMERIC
6384          27822      IFE LENGTH=2,<             ;EVALUATE IT AND MUST SURE ITS NUMERIC
6385 007501* 001000 000315          27824      CALL FRMVEL            ;POP OFF FORMULA TXTPTR
6386 007502* 000000 007536*          27826      PUSH H                ;POP OFF FORMULA TXTPTR
6387 007503* 001000 000345          27828      CALL FRCSNG            ;POP OFF FORMULA TXTPTR
6388 007504* 001000 000315          27828      ;
6389 007505* 001000 000315          27828      ;
6390 007506* 000000 006667*          27828      ;
6391 007507* 000000 007502*          27830      POP M                ;POP OFF FORMULA TXTPTR
6392 007510* 001000 000341          27840      DCX H                ;POP OFF FORMULA TXTPTR
6393 007511* 001000 000053          27840      ;

```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF MACRO 47(113) 03:12 10-SEP-75 PAGE 25-2
 F3 MAC 6-SEP-64 03:11 SIMPLE=USER=DEFINED=FUNCTION CODE

```

6394 007512* 001000 000327 27860 CHNGET ISEE IF TERMINATED
6395 007513* 001000 000302 27860 JNZ SNERR IF NOT SYNTAX ERROR
6396 007514* 000000 002072* 27860
6397 007515* 000000 007506* 27860
6398 27900 ;TO BE NICE SHOULD HAVE NEW CURLIN
6399 27920 ;BUT VERY MESSY
6400 007516* 001000 000341 27940 POP H IPOP OFF PTR AT ARG VARIABLE
6401 007517* 001000 000321 27960 POP D
6402 007520* 001000 000301 27980 POP B> IPOP OFF OLD VALUE
6403 28000 IFN MULDIMISTRINGIFUNCTS,<
6404 28020 DEFFINI MOV M,C
6405 28040 INX H
6406 007521* 001000 000161 28060 MOV M,B
6407 007522* 001000 000163 28080 PUTUEI: INX H
6408 007523* 001000 000163 28100 MOV M,E
6409 007526* 001000 000845 28120 INX H
6410 007527* 001000 000162 28140 MOV M,D
6411 007530* 001000 000341 28160 POP H
6412 007531* 001000 000511 28180 RET>
6413 28200 IFN FUNCTS,<
6414 28220
6415 28240 ; SUBROUTINE TO SEE IF WE ARE IN DIRECT MODE AND
6416 28260 ; COMPLAIN IF SO
6417 28280 ;
6418 007532* 001000 000345 28300 ERDIR: PUSH H
6419 007533* 001000 000052 28300 LMD CURLIN ;SAVE THEIR [H,L]
6420 007534* 000000 000160* 28320 ;SEE WHAT THE CURRENT LINE IS
6421 007535* 001000 0007514* 28340 INX H
6422 007536* 001000 000043 28360 MOV A,H
6423 007537* 001000 000174 28380 ORA L
6424 007540* 001000 000265 28400 POP H
6425 007541* 001000 000341 28420 RNZ
6426 007542* 001000 000300 28440 MVI E,ERRID
6427 007543* 001000 000036 28460 ;RETURN IF NOT
6428 007544* 000000 000014 28480 ;"ILLEGAL DIRECT" ERROR
6429 007545* 001000 000303 28500
6430 007546* 000000 002102* 28520 ; SUBROUTINE TO GET A POINTER TO A FUNCTION NAME
6431 007547* 000000 007534* 28540 GETFNM: SYNCCHK FNTR ;MUST START WITH "FN"
6432 28560 MVI A,128 ;DONT ALLOW AN ARRAY
6433 28580 STA SUBFLG ;DONT RECOGNIZE THE "(" AS
6434 007550* 001000 000517 28600 ;THE START OF AN ARRAY REFERENCE
6435 007551* 000000 000043 28620 ORA M ;PUT FUNCTION BIT ON
6436 007552* 001000 000076 28640 MOV B,A ;GET FIRST CHARACTER INTO [B]
6437 007553* 001000 000200 28660 IFN LENGTH=2,<
6438 007554* 001000 000062 28680 STRING,<CALL PTRGT2> ;REALLY GET THE POINTER
6439 007555* 001000 001601*
6440 007556* 000000 007546*
6441 007557* 000000 007546*
6442 28700 JMP CHKNUH>> ;MAKE SURE ITS NOT A STRING NAME
6443 007558* 001000 000266 28720 IFE STRING,<LENGTH=2,<JMP PTRGT2>>
6444 007560* 001000 000107 28740 PAGE
6445 28760
6446 28780
6447 28800
6448 007561* 001000 000363 28820
6449 007562* 000000 006512* 28840
6450 007563* 000000 007555* 28860
6451 28880
  
```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF MACRO 47(113) 03:12 10-SEP-75 PAGE 25-3
 F3 MAC 6-SEP-64 03:11 SIMPLE=USER=DEFINED=FUNCTION CODE

```

6447 007561* 001000 000363 28700 JMP CHKNUH>> ;MAKE SURE ITS NOT A STRING NAME
6448 007562* 000000 006512* 28720 IFE STRING,<LENGTH=2,<JMP PTRGT2>>
6449 007563* 000000 007555* 28740 PAGE
6450 28760
6451 28780
  
```

```

6452          28760  SUBTTL STRING FUNCTIONS
6453          28780  IFN   STRING,<           ;STRING HANDLING SUBROUTINES
6454          28782  ;
6455          28820  ; THE STRS FUNCTION TAKES A NUMBER AND GIVES
6456          28840  ; A STRING WITH THE CHARACTERS THE OUTPUT OF THE NUMBER
6457          28860  ; WOULD HAVE GIVEN
6458          28880  ;
6459  007564*  28900  STRS:             ;THE STRS FUNCTION TAKES A NUMBER AND GIVES
6460          28920  IFN   LENGTH=2,<       ;MAKE SURE THE ARGUMENT
6461          28940  CALL  CHKNUM>        ;IS A NUMERIC
6462          28960  CALL  FOUT            ;DO ITS OUTPUT
6463  007564*  001000  000315
6464  007565*  000000  004457*
6465  007566*  000000  007562*
6466  007567*  001000  000315
6467  007570*  000000  007637*
6468  007571*  000000  007565*
6469  007572*  001000  000315
6470  007573*  000000  010546*
6471  007574*  000000  007570*
6472  007575*  001000  000001
6473  007576*  000000  010546*
6474  007577*  000000  007575*
6475  007600*  001000  000305
29000  CALL  STRLIT           ;SCAN IT AND TURN IT INTO A STRING
29020  CALL  FREFAC           ;FREE UP THE TEMP
29040  LXI   B,FINBCK
29060  PUSH  B               ;SET UP ANSWER IN NEW TEMP
29080  ;
29100  ; STRCPY CREATES A COPY OF THE STRING
29120  ; WHOSE DESCRIPTOR IS POINTED TO [H,L].
29140  ; ON RETURN [D,E] POINTS TO DSCTMP
29160  ; WHICH HAS THE STRING INFO (LENGTH,WHERE COPIED TO)
29180  ;
29200  STRCPY: MOV  A,M           ;GET LENGTH
29220  INX   H               ;MOVE UP TO THE POINTER
29240  IFN   LENGTH=2,<
29260  INX   H>
29280  PUSH  H               ;GET POINTER TO POINTER OF ARG
29300  CALL  GETSPA           ;GET THE SPACE
29320  POP   H               ;FIND OUT WHERE STRING TO COPY
29340  PUSHH  M              ;REALLY IS IN [B,C]
29360  POP   B               ;SETUP DSCTMP
29380  CALL  STRAD2           ;SETUP DSCTMP
29400  PUSH  H               ;SAVE POINTER TO DSCTMP
29420  MOV   L,A              ;GET CHARACTER COUNT INTO [L]
29440  CALL  MOVSTH           ;MOVE THE CHARS IN
29460  POP   D               ;RESTORE POINTER TO DSCTMP
29480  RET                ;RETURN
29500  007624*  001000  000315
29520  STRINI: CALL  GETSPA           ;GET SOME STRING SPACE ([A] CHARs)

```

```

6505  007625*  000000  007772*
6506  007626*  000000  007628*
6507  007627*  001000  000141
6508  007636*  000000  001570*
6509  007631*  000000  007625*
6510  007632*  001000  000345
6511  007633*  001000  000167
29540  STRAD2: LXI  H,DSCTMP           ;GET DESC. TEMP
29560  STRAD1: PUSH  H               ;SAVE DESC. POINTER
29580  MOV   M,A              ;SAVE CHARACTER COUNT
29600  IFN   LENGTH=2,<
29620  INX   H>               ;MOVE TO ADDRESS FIELD
29640  JMP   PUTDEI           ;FUSE COMMON CODE TO
6512          29660  ;
6513          29680  ;STORE [D,E]=POINTER TO FREE SPACE
6514          29690  ;AND RESTORE ([M,L]) AS THE DESCRIPTOR POINTER
6515          29700  ;
6516          29720  STRLIT: DCX  H           ;ASSUME STR ENDS ON QUOTE
6517          29740  STRLIT: MVI  B,34
6518          29760  ;
6519          29780  ; STRLIT2 TAKES THE STRING LITERAL WHOSE FIRST CHARACTER
6520          29800  ; IS POINTED BY [H,L]+1 AND BUILDS A DESCRIPTOR FOR IT,
6521          29820  ; THE DESCRIPTOR IS INITIALLY BUILT IN DSCTMP, BUT PUTNEW
6522          29840  ; TRANSFERS IT TO THE DESCRIPTOR FIELD OF THE CURRENT PONTER
6523          29860  ; AT THE TEMPORARY IN FACTO, THE CHARACTERS OTHER THAN
6524          29880  ; ZERO THAT TERMINATE THE STRING SHOULD BE SET UP IN (B)
6525          29900  ; AND (U) IT THE TERMINATOR IS A QUOTE, THE QUOTE IS SKIPPED
6526          29920  ; OVER, LEADING QUOTES SHOULD BE SKIPPED BEFORE CALL, ON RETURN
6527          29940  ; THE CHARACTERS AFTER THE STRING LITERAL IS POINTED TO
6528          29960  ; BY [H,L] AND IS IN (A), BUT THE CONDITION CODES ARE
6529          29980  ; NOT SET UP,
6530          29704  ;
6531          29720  STRGET: DCX  H
6532  007637*  001000  000053
6533  007640*  001000  000006
6534  007641*  000000  0000042
6535  007642*  001000  000120
29760  STRLIT3: MOV  D,B
6536  007643*  001000  000345
6537  007644*  001000  000016
29780  STRLIT2: PUSH  H           ;SAVE POINTER TO START OF LITERAL
6538  007645*  000000  000377
29800  MVI  C,255             ;INITIALIZE CHARACTER COUNT
6539  007646*  001000  000043
6540  007647*  001000  000176
6541  007650*  001000  000001
6542  007651*  001000  000267
29820  STRGET: INX  H           ;FETT CHAR
6543  007652*  001000  000312
29840  INR   C               ;BUFP CHARACTER COUNT
6544  007653*  000000  000052*
6545  007654*  000000  007635*
6546  007655*  001000  0000272
29920  CMP   D,U              ;TEST
6547  007656*  001000  000312
29940  JZ   STRFIN           ;TEST
6548  007657*  000000  007665*
6549  007658*  000000  000052*
6550  007661*  001000  0000270
29960  CMP   B,U              ;CLOSING QUOTE
6551  007662*  001000  000302
29980  JNZ   STRGET           ;END, GO BACK FOR MORE
6552  007663*  000000  007646*
6553  007664*  001000  0000757*
6554  007665*  001000  000376
30000  STRFIN: CPI  34           ;IF QUOTE TERMINATES THE STRING
6555  007666*  000000  0000042
30020  CZ   CHRGRTR           ;SKIP OVER THE QUOTE
6556  007667*  001000  000314
6557  007670*  000000  003426*

```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF
F3 MAC 6-SEP-64 03:11

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-2
STRING FUNCTIONS

6558 007671* 000000 007663*
6559 007672* 001000 000543
6560 007673* 001000 000544
6561 007674* 001000 000553
6562 007675* 001000 000171
6563 007676* 001000 000315
6564 007677* 000000 007627*
6565 007678* 000000 007676*

6566 30140 ;
6567 007780* 001000 000547
6568 007780* 001000 000524
6569 007780* 000000 007601*
6570 007784* 000000 007677*

6571 30200 ;
6572 30240 ;
6573 30240 ;
6574 30260 ;
6575 30260 ;
6576 30300 ;
6577 30320 ;
6578 007705* 001000 000021
6579 007706* 000000 001570*
6580 007707* 000000 007705*
6581 007710* 001000 000052
6582 007711* 001000 001547*
6583 007712* 001000 007706*
6584 007713* 001000 000042
6585 007714* 000000 001637*
6586 007715* 000000 007711*

6587 30400 IFE LENGTH=2,<
6588 007716* 001000 000076
6589 007717* 000000 000003
6590 007720* 001000 000062
6591 007721* 000000 001543*
6592 007722* 000000 007714*
6593 007723* 001000 000615
6594 007724* 000000 004233*
6595 007725* 000000 007721*

6596 30400 IFN LENGTH=2,<
6597 30500 MVI A,₁
6598 30520 STA VALTYP
6599 30540 CALL MOVE>
6600 007726* 001000 000547
6601 30560 CMPAR
6602 30580 ;
6603 30600 ;
6604 007730* 000000 000020
6605 007731* 001000 000512
6606 007732* 000000 002102*
6607 007733* 000000 007724*
6608 007734* 001000 000042
6609 007735* 000000 001547*
6610 007736* 000000 007732*

6611 007737* 001000 0000341
6612 007740* 001000 0000175
6613 007741* 001000 000511
6614 30740 ;
6615 30760 ;
6616 30780 ;
6617 30800 ;
6618 007742* 001000 000043
6619 007743* 001000 000515
6620 007744* 000000 007637*
6621 007745* 000000 007735*
6622 30840 ;
6623 30860 ;
6624 30880 ;
6625 007746* 001000 000025
6626 007747* 000000 010434*
6627 007750* 000000 007744*
6628 30940 IFE LENGTH=2,<
6629 30960 CALL GETBCD
6630 30970 INR D
6631 30990 STRPR21 DCR D,<
6632 31000 IFN LENGTH=2,<
6633 31010 CALL MUVRN
6634 31020 INR E
6635 31030 STRPR21 E,>
6636 31040 RZ B
6637 31050 LDAX B
6638 31060 OUTCHR
6639 31070 CPI CR
6640 31200 CZ CRFIN
6641 007751* 001000 000314
6642 007752* 000000 000000*
6643 007753* 000000 007747*
6644 007754* 001000 000024
6645 30970 INR D
6646 007755* 001000 000053
6647 007756* 001000 000052*
6648 007757* 001000 000303
6649 007758* 000000 007755*
6650 007759* 000000 007764*
6651 31280 ;
6652 31300 ;
6653 31320 ;
6654 31340 ;
6655 31360 ;
6656 31380 ;
6657 31400 ;
6658 31420 ;
6659 007772* 001000 000267
6660 007773* 001000 000016
6661 007774* 001000 000361
6662 007775* 001000 000365
6663 007776* 001000 000052

31440 GETSPAI ORA A ; MUST BE NON ZERO, SIGNAL NO GARBAG YET
31460 XWD "010000-0016 ; MMV C" AROUND THE NEXT
31480 TRYG12: POP PSW ; IN CASE COLLECTED WHAT WAS LENGTH?
31500 PUSH PSW ; SAVE IT BACK
31520 LHLD STKTOP ; GET BOTTOM OF STRING SPACE

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 03:11

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-4
STRING FUNCTIONS

6664 007777' 000000 001615'
6665 010000' 000000 007776'
6666 010000' 000000 0016353'
6667 010000' 000000 0016352'
6668 010000' 000000 0015737'
6669 010000' 000000 007777'
6670 010000' 000000 000057
6671 010000' 000000 000117
6672 010000' 000000 000006
6673 010010' 000000 0000377
6674 010011' 000000 000000
6675 010010' 000000 000000
6676 010014' 000000 0000347
6677 010014' 000000 0000352
6678 010015' 000000 010026'
6679 010016' 000000 0100003'
6680 010017' 000000 0000042
6681 010020' 000000 0015737'
6682 010021' 000000 010015'
6683 010022' 000000 0000043
6684 010023' 000000 0000053
6685 010024' 000000 00000341
6686 010025' 000000 00000311
6687 010026' 000000 00000000
6688 010027' 000000 00000000
6689 010030' 000000 0000016
6690 010031' 000000 0000015
6691 010032' 000000 002114'
6692 010033' 000000 0100020'
6693 010034' 000000 00000000
6694 010035' 000000 00000000
6695 010036' 000000 00000001
6696 010037' 000000 00000000
6697 010040' 000000 01000000
6698 010041' 000000 00000000
6699 010042' 000000 00000000
6700 010043' 000000 00000000
6701 010044' 000000 01000000
6702 010044' 000000 0100037'
6703 52000 IFE REALIO,<
6704 52020 MVI A,7
6705 52040 DUTCHR>
6706 010065' 000000 0000042
6707 010065' 000000 0015737'
6708 010067' 000000 0100043'
6709 010050' 000000 00000000
6710 010051' 000000 00000000
6711 010052' 000000 0100046'
6712 010053' 001000 0000345
6713 010054' 001000 0000052
6714 010055' 000000 001615'
6715 010056' 000000 0100051'
6716 32140

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-5
STRING FUNCTIONS

6717 010057' 001000 0000345
6718 010060' 001000 0000041
6719 010061' 001000 0000001
6720 010062' 000000 0100055'
6721 010063' 001000 00000000
6722 010064' 001000 0000052
6723 010065' 000000 001547'
6724 010065' 000000 0100061'
6725 010067' 001000 00000000
6726 010067' 001000 00000000
6727 010067' 001000 00000000
6728 010073' 000000 0100065'
6729 010074' 001000 00000000
6730 010074' 001000 00000000
6731 010075' 000000 010121'
6732 010076' 000000 0100072'
6733 010077' 001000 00000000
6734 010077' 001000 00000000
6735 010100' 000000 001621'
6736 010101' 000000 00000000
6737 010102' 001000 00000000
6738 010103' 001000 00000000
6739 010104' 000000 001623'
6740 010105' 000000 010100'
6741 010106' 001000 00000000
6742 010107' 001000 00000000
6743 010110' 001000 00000000
6744 010111' 001000 00000000
6745 010112' 000000 010104'
6746 010113' 001000 00000000
6747 010114' 001000 00000000
6748 010115' 001000 00000000
6749 010116' 001000 00000000
6750 010117' 001000 00000000
6751 010117' 001000 00000000
6752 010120' 000000 00000000
6753 010121' 001000 00000000
6754 010122' 000000 010100'
6755 010123' 000000 010111'
6756 010124' 001000 00000000
6757 010125' 000000 010121'
6758 010126' 000000 010122'
6759 010127' 001000 00000000
6760 010130' 001000 00000000
6761 010131' 001000 00000000
6762 010132' 000000 00000000
6763 010133' 001000 00000000
6764 010134' 001000 00000000
6765 010135' 001000 00000000
6766 010136' 000000 010125'

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-6
STRING FUNCTIONS

```

6770
6771 010157* 001000 000501 32800 ARYVA2: POP B      IGET RID OF STACK GARBAGE
6772 010158* 001000 000502 32820 ARYVAR1 XCHG LMD    ISAVE ARYVAR IN [D,E]
6773 010159* 001000 000502 32840 LMD STREND      IGET END OF ARRAYS
6774 010160* 000000 001625* 32860 XCHG
6775 010161* 001000 000502 32880 COMPAR      IFLIP BACK
6776 010162* 001000 000512 32900 JZ GRBPAS      ISEE IF DONE WITH ARRAYS
6777 010164* 000000 010254* 32920 CALL MOVRH      IYES, SEE IF DONE COLLECTING
6780 010150* 000000 010142* 32940 IFE LENGTH=2,< 32940 MOV A,M      IGET THE VALUE TYPE INTO [A]
6781 010151* 001000 000516 32960 INX H      ISAVE POINTEN TO DARS
6783 010152* 001000 000543 32980 CALL MOVRH      IADD TO CURRENT POINTER POSITION
6784 010153* 001000 000515 32980 CPI 3      ISEE IF ITS A STRING
6785 010154* 000000 005502* 33000 IFN LENGTH=2,< 33000 DRA M      IGET END CHAR OF VAR NAME IN A
6786 010155* 000000 010147* 33020 PUSH M
6787 33040 INX H      ISAVE POINTEN TO DARS
6788 010157* 001000 000545 33060 IFE LENGTH=2,< 33060 ADD H,B      IADD TO CURRENT POSITION
6789 010158* 001000 000611 33100 CPI 3      ISEE IF ITS A STRING
6790 010159* 001000 000611 33120 JNZ ARYVA2> 33120 JNZ ARYVA2>      IF NOT JUST SKIP IT
6791 010160* 001000 000547 33140 IFN LENGTH=2,< 33140 DRA M      ISEE IF STRING VAR
6792 010161* 001000 000603 33160 JP ARYVA2> 33160 JP ARYVA2>      INO, KEEP ON TRUCKIN
6793 010162* 001000 000502 33180 SHLD TEMP3      ISAVE END OF ARRAY
6794 010163* 001000 000502 33200
6795 010164* 000000 010137* 33220 POP H      IGET BACK CURRENT POSITION
6796 010165* 000000 001575* 33240 MOV C,M      PICK UP NUMBER OF DIMS
6797 010166* 000000 010163* 33260 MVI B,0      MAKE DOUBLE WITH HIGH ZERO
6800 010167* 000000 010163* 33280 DAD B      IGO PAST DIMS
6801 010168* 000000 001575* 33300 DAD B      ISAY ADDING ON TWICE #DIM (2 BYTE GUYS)
6802 010169* 000000 000541 33320 INX H      IONE MORE TO ACCOUNT FOR #DIM'S
6803 010170* 001000 000541 33340 ARYSTH XCHG LMD      ISAVE CURRENT POSIT IN [D,E]
6804 010171* 001000 000516 33360 TEMP3      IGET END OF ARRAY
6805 010172* 001000 000502 33380
6806 010173* 000000 000502 33400
6807 010174* 001000 000611 33420 JZ ARYVAR      IEND OF ARRAY, TRY NEXT ARRAY
6808 010175* 001000 000611 33440 LXI B,ARYSTH      IADDR OF WHERE TO RETURN TO
6809 010176* 001000 000643 33460 DVAR2: PUSH B      IGOES ON STACK
6810 010177* 001000 000553 33480
6811 010178* 001000 000553 33500
6812 010179* 000000 001575* 33520
6813 010180* 000000 001575* 33540
6814 010181* 000000 001575* 33560
6815 010182* 000000 000535 33580 XCHG
6816 010183* 000000 000535 33600 COMPAR      ISEE IF AT END OF ARRAY
6817 010184* 000000 000512 33620 JZ ARYVAR      IEND OF ARRAY, TRY NEXT ARRAY
6818 010207* 000000 010201* 33640 IFN LENGTH=2,< 33640 DRA M      ISEE IF ITS THE NULL STRING
6819 010210* 001000 000001 33660 DVAR1 XRA A      IFORCE DVAR TO CALL GRBVAR
6820 010211* 000000 010177* 33680 DRL 128      ISAVE LENGTH
6821 010212* 000000 010206* 33700 PUSH BH      ISKIP NEXT TWO BYTES
6822 010213* 001000 000305 33720 POP D      IGET POINTER IN [D,E]
6823 010214* 001000 000305 33740 PDP B      IPOP OF STRING LENGTH
6824 010215* 001000 000305 33760 RP      IF WASNT A STR, RETURN
6825 010220* 001000 000443 33780 MVI A,C      ISAVE LENGTH OF STRING
6826 010221* 001000 000126 33800 DRA A>      ISET CONDITION CODES
6827 010222* 001000 000443 33820 RZ      INULL STRING, RETURN
6828 010223* 001000 000443 33840 MOV B,H      IMOVE [H,L] TO [B,C]
6829 010224* 001000 000502 33860 MOV C,L      IMOVE [B,C] TO [H,L]
6830 010225* 001000 000502 33880 LHLD FRETPO      IGET POINTER TO TOP OF STRING FREE SPACE
6831 010227* 000000 001573* 33900
6832 010230* 000000 010211* 33920 COMPAR      IS THIS STRINGS POINTER .LT. FRETPO
6833 010231* 001000 000547 33940 MOV M,B      IMOVE [B,C] BACK TO [H,L]
6834 010232* 001000 000547 33960 MOV L,C      IIF NOT, NO NEED TO HESS WITH IT FURTHER
6835 010233* 001000 000547 33980 POP H      IGET RETURN ADDRESS OFF STACK
6836 010234* 001000 000547 34000 XTHL      ISAVE MAX SEEN SO FAR & SAVE RETURN ADDRESS
6837 010235* 001000 000543 34020 COMPAR      ILETS SEE
6838 010236* 001000 000543 34040 XTHL      ISAVE MAX SEEN & GET RETURN ADDRESS OFF STACK
6839 010237* 001000 000543 34060 PUSH H      ISAVE RETURN ADDRESS BACK
6840 010240* 001000 000543 34080 MVI A,C      IMOVE [B,C] BACK TO [H,L]
6841 010241* 001000 000545 34100 MOV M,B      IPOP OFF MAX SEEN
6842 010242* 001000 000545 34120 MOV L,C      IPOP OFF MAX SEEN OFF STACK
6843 010243* 001000 000545 34140 RNC      IPOP OFF MAX SEEN
6844 010244* 001000 000545 34160 POP B      IPOP OFF MAX SEEN
6845 010245* 001000 000545 34180 POP PSW      IPOP OFF MAX SEEN
6846 010246* 001000 000561 34200 POP PSW      IAND VARIABLE POINTER
6847 010247* 001000 000561 34220 PUSH H      ISAVE NEW VARIABLE POINTER
6848 010248* 001000 000545 34240 PUSH D      IAND NEW MAX POINTER
6849 010249* 001000 000545 34260 PUSH B      ISAVE RETURN ADDRESS BACK
6850 010250* 001000 000511 34280 RET      IAND RETURN
6851 010251* 001000 000511 34300
6852 010252* 001000 000511 34320 F HERE WHEN MADE ONE COMPLETE PASS THRU STRING VARS
6853 010253* 001000 000511 34340
6854 010254* 001000 000531 34360 GRBPAS: POP D      IPOP OFF MAX POINTER
6855 010255* 001000 000541 34380 POP H      IAND GET VARIABLE POINTER
6856 010256* 001000 0008175 34400 MOV A,L      IGET LOW IN
6857 010257* 001000 000664 34420 DRA H      ISEE IF ZERO POINTER
6858 010258* 001000 000310 34440 RZ      IF END OF COLLECTION,
6859 010259* 001000 000311 34460 DGX H      THEN MAYBE RETURN TO GETSPA
6860 010260* 001000 000503 34480
6861 010261* 001000 000503 34480
6862 010262* 001000 000503 34480
6863 010263* 001000 000525 34480
6864 010264* 001000 000505 34480
6865 010265* 001000 000511 34480
6866 010266* 001000 000511 34480
6867 010267* 001000 000511 34480
6868 010268* 001000 000511 34480
6869 010269* 001000 000531 34480
6870 010270* 001000 000541 34480
6871 010271* 001000 0008175 34480
6872 010272* 001000 000664 34480
6873 010273* 001000 000310 34480
6874 010274* 001000 000503 34480
6875 010275* 001000 000503 34480

```

BASIC MCS 8080 GATES/ALLEN/DAVIDUFF
F3 MAC 6-SEP-64 03:11

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-8
STRING FUNCTIONS

```

6876 01026d* 001000 000106 34500 MOV B,M ;{B}=>HIGH BYTE OF DATA POINTER
6877 010265* 001000 00053 34520 DCX H
6878 010264* 001000 000116 34540 MOV C,M ;{B,C}=>POINTED AT STRING DESC
6879 010265* 001000 000345 34550 PUSH H ;SAVE THIS LOCATION SO THE POINTER
6880 34552 H CAN BE UPDATED AFTER THE STRING IS MOVED
6881 34554
6882 010266* 001000 000253 34560 DCX H
6883 34560 IFN LENGTH=2,<
6884 34600 DCX H>
6885 010267* 001000 000156 34620 MOV L,M ;{L}=>STRING LENGTH
6886 010270* 001000 000046 34640 MVI H,M ;{M,L} GET CHARACTER COUNT
6887 010271* 001000 000000
6888 010272* 001000 000211 34660 DAD B ;{H,L}=>POINTER BEYOND STRING
6889 34680 MOV D,B
6890 010274* 001000 000131 34700 MOV E,C ;{D,E}=>ORIGINAL POINTER
6891 010275* 001000 000253 34720 DCX H ;DON'T HAVE ONE BEYOND STRING
6892 010276* 001000 000104 34740 MOV B,M ;GET TOP OF STRING IN {B,C}
6893 010277* 001000 000115 34760 MOV C,L
6894 010300* 001000 000052 34780 LMLU FRETOP ;GET TOP OF FREE SPACE
6895 010301* 000000 001573* 34790
6896 010302* 000000 001577* 34800 CALL BLTUC ;MOVE STRING
6897 010303* 001000 000515 34800
6898 010304* 000000 002010* 34800
6899 010305* 000000 010301* 34800
6900 010305* 001000 000541 34820 PUP H ;GET BACK POINTER TO DESC.
6901 010307* 001000 000161 34840 MOV M,C ;SAVE FIXED ADDR
6902 010318* 001000 000045 34860 INX H ;MOVE POINTER
6903 010319* 001000 000000 34880 MOV M,B ;HIGH PART
6904 010312* 001000 000151 34900 MOV L,C
6905 010313* 001000 000140 34920 MOV H,B ;{H,L}=>NEW POINTER
6906 010314* 001000 00053 34940 DCX H ;FIX UP FRETOP
6907 010315* 001000 000503 34960 JMP FNDVAR ;AND TRY TO FIND HIGH AGAIN
6908 010316* 000000 010645* 34980
6909 010317* 000000 010504* 34980
6910
6911 35000 ;
6912 35020 ; THE FOLLOWING ROUTINE CONCATENATES TWO STRINGS
6913 35040 ; THE FACLO CONTAINS THE FIRST ONE AT THIS POINT,
6914 35060 ; {H,L} POINTS BEYOND THE + SIGN AFTER IT
6915 35060 ;
6916 010320* 001000 000505 35100 CAT: PUSH B ;PUT OLD PRECEDENCE BACK ON
6917 010321* 001000 000545 35120 PUSH H ;SAVE TEXT POINTER
6918 010322* 001000 000505 35140 LHLD FACLO ;GET POINTER TO STRING DESC.,
6919 010323* 000000 010537* 35160
6920 010324* 000000 010516* 35160 XTHL ;SAVE ON STACK & GET TEXT POINTER
6921 010325* 001000 000543 35180 CALL EVAL ;SAVE TEXT POINTER, GET BACK DESC.
6922 010326* 001000 000515 35180
6923 010327* 000000 000601* 35180
6924 010330* 000000 010523* 35200 XTHL ;SAVE TEXT POINTER, GET BACK DESC.
6925 010331* 000000 000603* 35220 CALL CHKSTR
6926 010332* 000000 000605* 35220
6927 010333* 000000 000630* 35220
6928 010334* 000000 010527* 35220

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 03:11

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-9
STRING FUNCTIONS

6930	018335*	001000	000176	35240	MVH	A,M	
6931	018336*	001000	000345	35260	PUSH	H	SAVE DESC, POINTER,
6932	018337*	001000	000052	35280	LHLD	FACLU	GET POINTER TO 2ND DESC.
6933	018340*	000000	001637*				
6934	018341*	000000	001033*	35300	PUSH	H	SAVE IT
6935	018342*	001000	000345	35320	ADD	H	ADD TWO LENGTHS TOGETHER
6936	018343*	001000	000052	35340	MVI	E,ERRLS	ISEE IF RESULT ,LT, 256
6937	018344*	000000	000011				
6938	018346*	001000	000532	35360	JC	ERROR	JEKOR "LONG STRING"
6939	018347*	000000	002102*				
6940	018356*	000000	010340*				
6941	018351*	001000	000315	35380	CALL	STRINI	GET INITIAL STRING
6942	018352*	000000	007624*				
6943	018353*	000000	010324*				
6944	018354*	001000	000321	35400	POP	D	
6945	018355*	001000	000315	35420	CALL	FRETMP	GET 2ND DESC,
6946	018356*	000000	010400*				
6947	018357*	000000	010352*				
6948	018360*	001000	000343	35440	XTHL		
6949	018361*	001000	000315	35460	CALL	FRETMP2	SAVE POINTER TO IT
6950	018362*	000000	010437*				FREE UP 1ST TEMP
6951	018363*	000000	000356*				
6952	018364*	001000	000315	35480	PUSH	H	SAVE DESC, POINTER (FIRST)
6953	018365*	000000	010352*	35500	LHLD	DSCTMP+2	GET POINTER TO FIRST
6954	018366*	000000	010572*				
6955	018367*	000000	010362*				
6956	018370*	001000	000353	35520	XCHG		
6957	018371*	001000	000315	35540	CALL	MOVINS	IN [D,E]
6958	018372*	000000	010407*				MOVE IN THE FIRST STRING
6959	018373*	000000	010366*				
6960	018374*	001000	000315	35560	CALL	MOVINS	AND THE SECOND
6961	018375*	000000	010371*				
6962	018376*	000000	010376*				
6963	018377*	001000	000641	35580	LXI	H,TSTOP	CAT REENTERS FORMULA EVALUATION AT
6964	018400*	000000	005557*				
6965	018401*	000000	010375*				
6966	018402*	001000	000343	35600	XTHL		
6967	018403*	001000	000345	35620	PUSH	H	TEXT POINTER OFF FIRST
6968	018404*	001000	000303	35640	JMP	PUTNEW	THEN RETURN ADDRESS OF TSTOP
6969	018405*	000000	000770*				
6970	018406*	000000	010406*				
6971							
6972							
6973	018407*	001000	000341	35700	MOVINS:	POP	H
6974	018410*	001000	000345	35720	XTHL		GET RETURN ADDR.
6975				35740	IFE	LENGTH=2,<	PUT BACK, BUT GET DESC.
6976	018411*	001000	000176	35760	MOV	A,M	
6977	018412*	001000	000643	35780	INX	H	(A)=STRING LENGTH
6978	018413*	001000	000116	35800	MOV	C,M	(B,C)=POINTER AT STRING DATA
6979	018414*	001000	000643	35820	INX	H	
6980	018415*	001000	000186	35840	MOV	B,M	
6981	018416*	001000	000157	35860	MOV	L,>	(L)=STRING LENGTH

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05:11

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-10
STRING FUNCTIONS

5982 35980 IFN LENGTH=2,<
5983 35980 PUSH R₁ GET LENGTH ON STACK
5984 35980 PUSHR FAND POINTER
5985 35980 POP B TEXT PTR HERE
5986 35980 POP H> CHARACTER COUNT HERE
5987 010417* 001000 000054 35980 MOVSTR INR L
5988 010420* 001000 000055 36000 MOVLVP DCR L SET CC'S
5989 010421* 001000 000310 36020 RZ 10, NO BYTE TO MOVE
5990 010422* 001000 000012 36040 LDAX B GET CHAR
5991 010423* 001000 000026 36060 STAX D SAVE IT
5992 010424* 001000 000063 36080 INX D MOVE POINTERS
5993 010425* 001000 000023 36100 INX D
5994 010426* 001000 000303 36120 JMP MUVP KEEP DOING IT
5995 010427* 001000 010420*
5996 010430* 000000 010405*
5997 36140 ;
5998 36160 ; FRETMP IS PASSED A POINTER TO A STRING DESCRIPTOR IN [D,E]
5999 36180 ; THIS VALUE IS RETURNED IN [H,L] ALL THE OTHER REGISTERS ARE MODIFIED,
6000 ; AND THERE IS NO CODE TO USE THE STRING DESCRIPTOR WHICH POINTS
7800 ; TO THE LAST TEMPORARY DESCRIPTOR ALLOCATED BY PUTNEW.
7801 36220 ; IF SO, THE TEMPORARY IS FREED UP BY THE UPDATING OF TEMPPT.
7802 36240 ; IF SO, THE TEMPORARY IS FREED UP BY THE UPDATING OF TEMPPT.
7803 36260 ; IF A TEMPORARY IS FREED UP, A FURTHER CHECK IS MADE TO SEE IF THE
7804 36280 ; STRING DATA THAT THAT STRING TEMPORARY POINTED TO IS THE
7805 36300 ; LOWEST PART OF STRING SPACE IN USE.
7806 36320 ; IF SO, FRETME IS UPDATED TO REFLECT THE FACT THAT THAT SPACE IS NO
7807 ; LONGER IN USE. THIS CAUSES DIFFICULTY FOR ASSIGNMENT ("LET") BECAUSE
7808 ; THE TEMPORARY IS USED AS A TEMPORARY UPDATING NEEDS TO SET THE VALUE TO THE RIGHT
7809 ; OF THE EQUAL SIGN IN THE LET. THE ACTUAL DATA
7810 36400 ; IS STILL ACTIVE DATA SINCE A VARIABLE IS BEING SET UP TO POINT
7811 36420 ; AT IT. "LET" FOOLS FRETMP BY SETTING THE LENGTH OF THE
7812 36440 ; TEMPORARY TO ZERO TEMPORARILY.
7813 36460 ;
7814 010431* 001000 000315 36480 FRESTRI CALL CHKSTR MAKE SURE ITS A STRING
7815 010432* 001000 00033*
7816 010433* 001000 010402*
7817 010434* 001000 000052 36500 FREFACI LHLD FACLO
7818 010435* 000000 010437*
7819 010436* 000000 010432*
7820 010437* 001000 000353 36520 FRETME2 XCHG FREE UP THE TEMP IN THE FACLO
7821 010440* 001000 000052 36540 FRETMP: LHLD TEMPPT GET TEMP POINTER
7822 010441* 000000 001547*
7823 010442* 001000 010435*
7824 010443* 001000 000353 36560 DCX H FLOOK AT WHAT IS IN THE LAST TEMP
7825 010444* 001000 000106 36580 MOV B,M ;[B,C]=POINTER AT STRING
7826 010445* 001000 000053 36600 DCX H DECREMENT TEMPPT BY STRSIZ
7827 010446* 001000 000116 36620 MOV C,M
7828 010447* 001000 000053 36640 DCX H
7829 36660 IFN LENGTH=2,<
7830 36680 DCX H>
7831 010450* 001000 000347 36700 COMPAR ;SEE IF [D,E] POINT AT THE LAST
7832 010451* 001000 000353 36720 XCHG RETURN WITH [H,L]
7833 36740 ;POINTING AT CURRENT DESCRIPTOR
7834 010452* 001000 000300 36760 RNZ RETURN NOW IF NOT FREEING DONE

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF
F3 MAC 6-SEP-64 05:11

MACRO 47(113) 03:12 10-SEP-75 PAGE 26-11
STRING FUNCTIONS

7835 010453* 001000 000042 36780 SHLD TEMPPT UPDATE THE TEMP POINTER SINCE
7836 010454* 000000 001547*
7837 010455* 000000 010441*
7838 36800 ;
7839 010456* 001000 000323 36820 PUSH D JSAVE [D,E] TO RETURN IN [H,L]
7840 010457* 001000 000120 36840 MOV D,B ;[D,E]=POINTER AT STRING
7841 010460* 001000 000131 36860 MOV E,C
7842 010461* 001000 000033 36880 DCX D SUBTRACT ONE
7843 010462* 001000 000116 36900 MOV C,M ;C=M LENGTH OF THE STRING FREED UP
7844 010463* 001000 000052 36920 LHLD FRETOP ;SEE IF ITS THE FIRST
7845 010464* 000000 001573*
7846 010465* 000000 010454*
7847 36940 ;
7848 010466* 001000 000347 36960 COMPAR ;ONE IN STRING SPACE
7849 010467* 001000 000302 36980 JNZ NOTLST ;IND SO DON'T ADD
7850 010470* 000000 010477*
7851 010471* 000000 010464*
7852 010472* 001000 000107 37000 MOV B,A ;MAKE [B]=#0
7853 010473* 001000 000111 37020 DAD B ;ADD
7854 010474* 001000 000042 37040 SHLD FRETOP ;AND UPDATE FRETOP
7855 010475* 000000 001573*
7856 010476* 000000 010470*
7857 010477* 001000 000341 37060 NOTLST: POP H ;GET POINTER AT CURRENT DESCRIPTOR
7858 010500* 001000 000311 37080 RET
7859 37100 ;
7860 37120 ; THE FUNCTION LEN(S) RETURNS THE LENGTH OF THE
7861 37140 ; STRING PASSED AS AN ARGUMENT
7862 37160 ;
7863 010501* 001000 000001 37180 LEN: LXI B,SNGFLT ;CALL SNGFLT WHEN DONE
7864 010502* 000000 007400*
7865 010503* 000000 016475*
7866 010504* 001000 000305 37200 PUSH B ;LIKE SO
7867 010505* 001000 000315 37220 CALL FRESTR ;FREE UP TEMP POINTED TO BY FACLO
7868 010506* 000000 010431*
7869 010507* 000000 000006*
7870 010508* 000000 000057 37240 XRA A ;FORCE NUMERIC FLAG
7871 010511* 001000 000127 37260 MOV D,A ;SET HIGH OF [D,E] TO ZERO FOR VAL
7872 37280 IFN LENGTH=2,<
7873 37300 STA VALTYP*
7874 010512* 001000 000176 37320 MOV A,M
7875 010513* 001000 000267 37340 ORA A ;SET CONDITION CODES ON LENGTH
7876 010514* 001000 000311 37360 RET ;RETURN
7877 37380 ;
7878 37400 ; THE FOLLOWING IS THE ASC(S) FUNCTION, IT RETURNS AN INTEGER
7879 ; WHICH IS THE DECIMAL ASCII EQUIVALENT
7880 37420 ;
7881 37440 ;
7882 010515* 001000 000315 37460 ASC: CALL LEN1 ;SET UP ORIGINAL STR
7883 010516* 000000 010505*
7884 010517* 000000 010506*
7885 010517* 001000 000312 37480 JZ FCERR ;NULL STR, BAD ARG.
7886 010521* 001000 010776*
7887 010522* 000000 010516*
7888 010523* 001000 000043 37500 INX H ;BUMP POINTER

7086
 7089
 7090 010524* 001000 000367
 7091 010525* 001000 000341
 7092 010526* 001000 000176
 7093 010527* 001000 000303
 7094 010530* 000000 007400*
 7095 010531* 000000 010521*
 7096
 7097
 7098 010532* 001000 000076
 7099 010533* 000000 000001
 7100 010534* 001000 000315
 7101 010535* 000000 007624*
 7102 010536* 001000 000356
 7103 010537* 001000 000151
 7107 010540* 000000 011023*
 7108 010541* 000000 010555*
 7109 010542* 000000 000157*
 7111 010543* 001000 000165
 7112 010545* 001000 000301
 7113 010546* 001000 000301
 7115 010547* 001000 000303
 7116 010550* 000000 007705*
 7117 010551* 000000 010543*
 7118
 7119
 7120
 7121
 7122
 7123 010552* 001000 000315
 7124 010553* 000000 010764*
 7125 010554* 000000 010550*
 7126 010555* 001000 000257
 7127 010556* 001000 000343
 7128 010557* 001000 000117
 7129 010560* 001000 000345
 7130 010561* 001000 000376
 7131 010562* 001000 000270
 7132 010563* 001000 000332
 7133 010564* 000000 010570*
 7134 010565* 000000 010553*
 7135 010566* 001000 000170
 7136 010567* 001000 000021
 7137 010570* 001000 000016
 7138 010571* 001000 000000
 7139 010572* 001000 000305
 7140 010573* 001000 000315

37520 IFN LENGTH=2,<
 37540 INX H> ;BUMP POINTER
 37560 PUSHM H ;GET ADDRESS
 37580 POP H ;GET ADDR IN [H,L]
 37600 MOV A,M ;GET FIRST CHAR
 37620 JMP SNGFLT IT ;SNGFLT IT
 37640 ; CHR\$(#) CREATS A STRING WHICH CONTAINS AS ITS ONLY
 37650 ; CHARACTER THE ASCII EQUIVALENT OF THE INTEGER ARG (#)
 37660 ; WHICH MUST BE LT, 255.
 37720 ;
 37740 CHRS: MVI A,1 ;ONE CHARACTER STR
 37760 CALL STRINI ;GET STRING IN DSCTMP
 37780 CALL CONINT ;GET INTEGER IN RANGE
 37800 LHLD DSCTMP+2 ;GET ADDR OF STR
 37820 MOV M,E ;SAVE ASCII BYTE
 37840 FINBCKI POP B ;RETURN TO HIGHER LEVEL &
 37860 ;SKIP THE CHNUM CALL,
 37880 JMP PUTNEH ;GO CALL PUTNEW
 37900 ;
 37920 ; THE FOLLOWING IS THE LEFTS(S,#) FUNCTION.
 37940 ; IT TAKES THE LEFTMOST # CHARS OF THE STR.
 37960 ; IF # IS ,GT, THAN THE LEN OF THE STR, IT RETURNS THE WHOLE STR.
 37980 ;
 38000 LEFTS: CALL PREAM ;TEST THE PARAMETERS
 38020 XRA A ;LEFT NEVER CHANGES STRING POINTER
 38040 LEFTS1: XTHL ;SAVE TEXT POINTER
 38060 MOV C,A ;OFFSET NOW IN [C]
 38080 LEFTS2: PUSH H ;SAVE DESC FOR FRETMP
 38100 MOV A,M ;GET STRING LENGTH
 38120 CMP B ;ENTIRE STRING WANTED?
 38140 JC ALLSTK ;IF #CHARS ASKED FOR, GE, LENGTH, YES
 38160 MOV A,B ;GET TRUNCATED LENGTH OF STRING
 38180 XWD A,01000-021 ;SKIP OVER MVI USING "LXI D,"
 38200 ALLSTK: MVI C,0 ;MAKE OFFSET ZERO
 38220 PUSH B ;SAVE OFFSET ON STACK
 38240 CALL GETSPA ;GET SPACE FOR NEW STRING

7141 010574* 001000 007772*
 7142 010575* 000000 010564*
 7143 010576* 001000 000303
 7144 010577* 001000 000341
 7145 010560* 001000 000345
 7146 010601* 001000 000043
 7147
 7148
 7149 010602* 001000 000106
 7150 010603* 001000 000043
 7151 010604* 001000 000046
 7152 010605* 001000 000150
 7153 010606* 001000 000006
 7154 010607* 000000 000000
 7155 010610* 001000 000011
 7156 010611* 001000 000104
 7157 010612* 001000 000015
 7158 010613* 001000 000315
 7159 010614* 001000 000315
 7160 010615* 001000 010574*
 7161 010616* 001000 000157
 7162 010617* 001000 000315
 7163 010620* 000000 010417*
 7164 010621* 000000 010614*
 7165 010622* 001000 000321
 7166 010623* 001000 000315
 7167 010624* 000000 010440*
 7168 010625* 000000 010574*
 7169 010626* 001000 000303
 7170 010627* 000000 007725*
 7171 010630* 000000 010624*
 7172
 7173 010631* 001000 000315
 7174 010632* 000000 010764*
 7175 010633* 000000 010577*
 7176 010634* 001000 000321
 7177 010635* 001000 000329
 7178 010636* 001000 000052
 7179 010637* 001000 000220
 7180 010640* 001000 000303
 7181 010641* 000000 010556*
 7182 010642* 000000 010632*
 7183
 7184
 7185
 7186
 7187
 7188
 7189
 7190 010643* 001000 000353
 7191 010644* 001000 000176
 7192 010645* 001000 000315
 7193 010646* 000000 010767*

38600 ; MID(S, #) RETURNS STR WITH CHARS FROM # POSITION
 38620 ; ONWARD, IF # IS GT LEN(S) THEN RETURN NULL STRING,
 38640 ; MID(S, #, #) RETURNS STR WITH CHARS FROM # POSITION
 38660 ; FOR #2 CHARS, IF #2 GOES PAST END OF STRING, RETURN
 38680 ; AS MUCH AS POSSIBLE,
 38700 ;
 38720 PUSH D ;SAVE BACK FOR LEFT
 38740 LDAX D ;GET PRESENT LEN OF STR
 38760 SUB B ;SUBTRACT 2ND PARM
 38780 JMP LEFTS ;CONTINUE WITH LEFT CODE
 38800 ;
 38940 MID\$: XCHG ;PUT THE TEXT POINTER IN [H,L]
 38960 MOV A,M ;GET THE FIRST CHARACTER
 38980 CALL PREAM2 ;GET OFFSET OFF STACK AND MAKE