```
                                        63900   SUBTTL  POLYNOMIAL EVALUATOR AND THE RANDOM NUMBER GENERATOR
                                        63920   IFN     EXTFNC,<
                                        63940           ;EVALUATE P(X**2)*X
                                        63960           ;POINTER TO DEGREE+1 IS IN (HL)
                                        63980           ;THE CONSTANTS FOLLOW THE DEGREE
                                        64000           ;CONSTANTS SHOULD BE STORED IN REVERSE ORDER, FAC HAS X
                                        64020           ;WE COMPUTE:
                                        64040           ; C0*X+C1*X**3+C2*X**5+C3*X**7+...+C(N)*X**(2*N+1)
002615' 001000  000315                  64060   POLYX:  CALL    PUSHF           ;SAVE X
002614' 000000  001205'
002615' 000000  002550'
002616' 001000  000021                  64080           LXI     D,FMULTT        ;PUT ADDRESS OF FMULTT ON STACK SO WHEN WE
002617' 000000  000515'
002620' 000000  002614'
002621' 001000  000325                  64100           PUSH    D               ; RETURN WE WILL MULTIPLY BY X
002622' 001000  000345                  64120           PUSH    H               ;SAVE CONSTANT POINTER
002623' 001000  000315                  64140           CALL    MOVRF           ;SQUARE X
002624' 000000  001240'
002625' 000000  002617'
002626' 001000  000315                  64160           CALL    FMULT
002627' 000000  000517'
002630' 000000  002624'
002631' 001000  000341                  64180           POP     H               ;GET CONSTANT POINTER
                                        64200                                   ;FALL INTO POLY


                                        64260           ;POLYNOMIAL EVALUATOR
                                        64280           ;POINTER TO DEGREE+1 IS IN (HL), IT IS UPDATED
                                        64300           ;THE CONSTANTS FOLLOW THE DEGREE
                                        64320           ;CONSTANTS SHOULD BE STORED IN REVERSE ORDER, FAC HAS X
                                        64340           ;WE COMPUTE:
                                        64360           ; C0+C1*X+C2*X**2+C3*X**3+...+C(N-1)*X**(N-1)+C(N)*X**N
002632' 001000  000315                  64380   POLY:   CALL    PUSHF           ;SAVE X
002633' 000000  001205'
002634' 000000  002627'
002635' 001000  000176                  64400           MOV     A,M             ;GET DEGREE
002636' 001000  000043                  64420           INX     H               ;INCREMENT POINTER TO FIRST CONSTANT
002637' 001000  000315                  64440           CALL    MOVFM           ;MOVE FIRST CONSTANT TO FAC
002640' 000000  001222'
002641' 000000  002633'
002642' 001000  000006                  64460           XWD     1000,006        ;"MVI   B" OVER NEXT BYTE
002643' 001000  000361                  64480   POLY1:  POP     PSW             ;GET DEGREE
002644' 001000  000301                  64500           POPR                    ;GET X
002645' 001000  000321
002646' 001000  000075                  64520           DCR     A               ;ARE WE DONE?
002647' 001000  000310                  64540           RZ                      ;YES, RETURN
002650' 001000  000325                  64560           PUSHR                   ;NO, SAVE X
002651' 001000  000305
002652' 001000  000365                  64580           PUSH    PSW             ;SAVE DEGREE
002653' 001000  000345                  64600           PUSH    H               ;SAVE CONSTANT POINTER
002654' 001000  000315                  64620           CALL    FMULT           ;EVALUATE THE POLY, MULTIPLY BY X
002655' 000000  000517'
002656' 000000  002640'
```



```
002657' 001000  000341                  64640           POP     H               ;GET LOCATION OF CONSTANTS
002660' 001000  000315                  64660           CALL    MOVRM           ;GET CONSTANT
002661' 000000  001243'
002662' 000000  002655'
                                        64680           PUSH    H               ;STORE LOCATION OF CONSTANTS SO FADD AND FMULT
002663' 001000  000345
002664' 001000  000315                  64700           CALL    FADD            ; WILL NOT SCREW THEM UP, ADD IN CONSTANT
002665' 000000  000025'
002666' 000000  002661'
002667' 001000  000341                  64720           POP     H               ;MOVE CONSTANT POINTER TO NEXT CONSTANT
002670' 001000  000303                  64740           JMP     POLY1>          ;SEE IF DONE
002671' 000000  002643'
002672' 000000  002665'


                                        64800           ;PSUEDO-RANDOM NUMBER GENERATOR
                                        64820           ;IF ARG=0, THE LAST RANDOM NUMBER GENERATED IS RETURNED
                                        64840           ;IF ARG .LT. 0, A NEW SEQUENCE OF RANDOM NUMBERS IS STARTED
                                        64860           ; USING THE ARGUMENT
                                        64880           ;TO FORM THE NEXT RANDOM NUMBER IN THE SEQUENCE, WE MULTIPLY THE
                                        64900           ;PREVIOUS RANDOM NUMBER BY A RANDOM CONSTANT, AND ADD IN ANOTHER
                                        64920           ;RANDOM CONSTANT.  THEN THE MO AND LO BYTES ARE SWITCHED, THE
                                        64940           ;EXPONENT IS PUT WHERE IT WILL BE SHIFTED IN BY NORMAL, AND THE
                                        64960           ;EXPONENT IN THE FAC SET TO 200 SO THE RESULT WILL BE LESS THAN 1.
                                        64980           ;THIS IS THEN NORMALIZED AND SAVED FOR THE NEXT TIME.
                                        65000           ;THE MO AND LO BYTES WERE SWITCHED SO WE HAVE A RANDOM CHANCE OF
                                        65020           ;GETTING A NUMBER LESS THAN OR GREATER THAN .5
002673'                                 65040   RND:
                                        65060   IFN     LENGTH-2,<
002673' 001000  000357                  65080           FSIGN>          ;GET SIGN OF ARG
                                        65100   IFE     LENGTH-2,<
                                        65120           CALL    VSIGN           ;GET THE SIGN OF THE ARG
                                        65140           PUSH    PSW             ;SAVE THE SIGN
                                        65160           CM      FNCSNG          ;IF IT IS NEGATIVE, FORCE IT TO BE A SNG
                                        65180           MVI     A,4             ; SINCE WE WILL USE IT
                                        65200           STA     VALTYP          ;MAKE SURE THE RESULT IS "SINGLE PRECISION"
                                        65220           POP     PSW>            ;GET THE SIGN BACK
002674' 001000  000372                  65240           JM      RND1            ;START NEW SEQUENCE IF NEGATIVE
002675' 000000  002725'
002676' 000000  002671'
002677' 001000  000041                  65260           LXI     H,RNDX          ;GET LAST NUMBER GENERATED
002700' 000000  002752'
002701' 000000  002675'
002702' 001000  000315                  65280           CALL    MOVFM
002703' 000000  001222'
002704' 000000  002700'
002705' 001000  000310                  65300           RZ                      ;RETURN LAST NUMBER GENERATED IF ZERO
                                        65320   IFE     EXTFNC,<
                                        65340           CALL    FMULTS>         ;MULTIPLY BY CONSTANT A
                                        65360   IFN     EXTFNC,<
002706' 001000  000315                  65380           CALL    MOVRM
002707' 000000  001243'
002710' 000000  002703'
```

```
                                                65400           CALL    FMULT>
                                                65420           MOVRI   150,050,261,106 ;ADD IN CONSTANT OF ORDER 2^(-24)
                                                65440           CALL    FADD
                                                65460   RND1:   CALL    MOVRF           ;SWITCH MO AND LO BYTES,
                                                65480           MOV     A,E             ;GET LO
                                                65500           MOV     E,C             ;PUT MO IN LO BYTE
                                                65520           MOV     C,A             ;PUT LO IN MO BYTE
                                                65540           MVI     M,200           ;MAKE RESULT POSITIVE
                                                65560           DCX     H               ;GET POINTER TO EXPONENT
                                                65580           MOV     B,M             ;PUT EXPONENT IN OVERFLOW POSITION
                                                65600           MVI     M,200           ;SET EXP SO RESULT WILL BE BETWEEN 0 AND 1
                                                65620           CALL    NORMAL          ;NORMALIZE THE RESULT
                                                65640           LXI     H,RNDX          ;SAVE RANDOM NUMBER GENERATED FOR NEXT
                                                65660           JMP     MOVMF           ; TIME
                                                65700           ;CONSTANTS AND STORAGE FOR RND
                                                65720   RNDX:   122     ;LAST RANDOM NUMBER GENERATED, BETWEEN 0 AND 1
                                                65740           307
                                                65760           117
                                                65780           200
                                                65800           172     ;RANDOM NUMBER OF ORDER 2^24
                                                65820           104
                                                65840           065
                                                65860           230
                                                65880   PAGE
```



```
                                                65900   SUBTTL  SINE, COSINE AND TANGENT FUNCTIONS
                                                65920   IFN     EXTFNC,<
                                                65940           ;COSINE FUNCTION
                                                65960           ;IDEA:  USE COS(X)=SIN(X+PI/2)
                                                65980   COS:    LXI     H,PI2           ;ADD PI/2 TO FAC
                                                66000           CALL    FADDS>
                                                66020                                   ;FALL INTO SIN
                                                66080           ;SINE FUNCTION
                                                66100           ;IDEA:  USE IDENTITIES TO GET FAC IN QUADRANTS I OR IV
                                                66120           ;THE FAC IS DIVIDED BY 2*PI AND THE INTEGER PART IS IGNORED BECAUSE
                                                66140           ;SIN(X+2*PI)=SIN(X),  THEN THE ARGUMENT CAN BE COMPARED WITH PI/2 BY
                                                66160           ;COMPARING THE RESULT OF THE DIVISION WITH PI/2/(2*PI)=1/4.
                                                66180           ;IDENTITIES ARE THEN USED TO GET THE RESULT IN QUADRANTS I OR IV.
                                                66200           ;AN APPROXIMATION POLYNOMIAL IS THEN USED TO COMPUTE SIN(X).
                                                66220   SIN:    CALL    PUSHF           ;DIVIDE FAC BY 2*PI
                                                66240           MOVRI   203,111,017,333 ;AFTER DIVIDING BY 2*PI, RESULT IS
                                                66260           CALL    MOVFR           ; BETWEEN 0 AND 1
                                                66280           POPR
                                                66300           CALL    FDIV
                                                66320           CALL    PUSHF           ;DISREGARD INTEGER PART SINCE SIN
                                                66340           CALL    INT             ; IS PERIODIC WITH PERIOD 2*PI
                                                66360           POPR
                                                66380           CALL    FSUB
                                                66400   IFE     EXTFNC,<
                                                66420           LXI     B,177*400+$CODE ;GET 1/4
                                                66440           MOV     D,C
                                                66460           MOV     E,C
                                                66480           CALL    FSUB>
```

```
                                                66500   IFN     EXTFNC,<
                                                66520           LXI     H,FR4           ;SEE WHAT QUADRANT WE ARE IN
                                                66540           CALL    FSUBS>
                                                66560   FSIGN
                                                66580           STC                     ;SET QUADRANT I FLAG
                                                66600           JP      SIN1            ;FIRST QUADRANT, GET BACK ORIGINAL X
                                                66620           CALL    FADDM           ;ADD 1/2
                                                66640   FSIGN
                                                66660           ORA     A               ;CLEAR CARRY
                                                66680   SIN1:   PUSH    PSW             ;SAVE QUADRANT FLAG
                                                66700           CP      NEG             ;NEGATE IF IN QUADRANTS I, II OR III
                                                66720   IFE     EXTFNC,<
                                                66740           LXI     B,177*400+$CODE ;GET 1/4
                                                66760           MOV     D,C
                                                66780           MOV     E,C
                                                66800           CALL    FADD>
                                                66820   IFN     EXTFNC,<
                                                66840           LXI     H,FR4           ;ADD 1/4,  IN QUADRANTS II, III
                                                66860                                   ; USE THE IDENTITY: SIN(PI-X)=SIN(X)
                                                66880                                   ;IN QUADRANT IV, USE THE IDENTITY:
                                                66900           CALL    FADDS>          ;   SIN(X-2*PI)=SIN(X)
                                                66920           POP     PSW             ;GET QUADRANT FLAG
                                                66940           CNC     NEG             ;NEGATE IF IN QUADRANTS II, III OR IV
                                                66960   IFE     EXTFNC,<                ;EVALUATE APPROXIMATION POLYNOMIAL
                                                66980           CALL    PUSHF           ;SAVE X
                                                67000           CALL    MOVRF           ;SQUARE X
                                                67020           CALL    FMULT
                                                67040           CALL    PUSHF           ;SAVE X**2
                                                67060           LXI     H,SINCON
                                                67080           CALL    MOVFM           ;MOVE FIRST CONSTANT INTO FAC
                                                67100           POPR                    ;GET X**2
                                                67120           MVI     A,4             ;GET DEGREE
                                                67140   POLY1:  PUSH    PSW             ;SAVE DEGREE
                                                67160           PUSHR                   ;SAVE X**2
                                                67180           PUSH    H               ;SAVE CONSTANT POINTER
                                                67200           CALL    FMULT           ;EVALUATE THE POLY, MULTIPLY BY X**2
                                                67220           POP     H               ;GET POINTER TO CONSTANTS
```



```
                                                67240           CALL    MOVRM           ;GET CONSTANT
                                                67260           PUSH    H               ;SAVE POINTER
                                                67280           CALL    FADD            ;ADD IN CONSTANT
                                                67300           POP     H               ;MOVE POINTER TO NEXT CONSTANT
                                                67320           POPR                    ;GET X**2
                                                67340           POP     PSW             ;GET DEGREE
                                                67360           DCR     A               ;SEE IF DONE
                                                67380           JNZ     POLY1           ;NO, DO NEXT TERM
                                                67400           JMP     FMULTI>         ;MULTIPLY BY X AND WE ARE DONE
                                                67420   IFN     EXTFNC,<
                                                67440           LXI     H,SINCON        ;CALCULATE THE SIN BY EVALUATING
                                                67460           JMP     POLYX>          ;  THE APPROXIMATION POLYNOMIAL
                                                67480           ;CONSTANTS FOR SIN, COS
                                                67500   IFN     EXTFNC,<
                                                67520   PI2:    333             ; PI/2
                                                67540           017
                                                67560           111
                                                67580           201
                                                67600   FR4:    000             ; 1/4
                                                67620           000
                                                67640           000
                                                67660           177>
                                                67680   SINCON:
                                                67700   IFN     EXTFNC,<
                                                67720           5>      ;DEGREE
                                                67740           272     ; 39,701067
                                                67760           327
                                                67780           036
                                                67800           206
                                                67820           144     ; -76,57498
                                                67840           046
                                                67860           231
                                                67880           207
                                                67900           130     ; 81,60223
                                                67920           064
                                                67940           043
                                                67960           207
                                                67980           340     ; -41,34168
                                                68000           135
                                                68020           245
                                                68040           206
                                                68060           332     ; 6,283185
                                                68080           017
                                                68100           111
                                                68120           203

                                                68140   IFN     EXTFNC,<
                                                68200           ;TANGENT FUNCTION
```

```
                                        68220           ;TAN(X)=SIN(X)/COS(X)
4114  003125' 001000  000315           68240   TAN:    CALL    PUSHF           ;SAVE ARG
4115  003126' 000000  001205'
4116  003127' 000000  003066'
4117  003130' 001000  000315           68260           CALL    SIN             ;   TAN(X)=SIN(X)/COS(X)
4118  003131' 000000  002770'
4119  003132' 000000  003126'
4120  003133' 001000  000301           68280           POP     B               ;GET X OFF STACK
4121  003134' 001000  000341           68300           POP     H               ;PUSHF SMASHES (DE)
4122  003135' 001000  000315           68320           CALL    PUSHF
4123  003136' 000000  001205'
4124  003137' 000000  003131'
4125  003140' 001000  000353           68340           XCHG                    ;GET LO'S WHERE THEY BELONG
4126  003141' 001000  000315           68360           CALL    MOVFR
4127  003142' 000000  001225'
4128  003143' 000000  003136'
4129  003144' 001000  000315           68380           CALL    COS
4130  003145' 000000  002762'
4131  003146' 000000  003142'
4132  003147' 001000  000303           68400           JMP     FDIVT>
4133  003150' 000000  000653'
4134  003151' 000000  003145'
4135                                    68420   PAGE
```



```
4136                                    68440   SUBTTL  ARCTANGENT FUNCTION
4137                                    68460   IFN     EXTFNC,<
4138                                    68480           ;IDEA: USE IDENTITIES TO GET ARG BETWEEN 0 AND 1 AND THEN USE AN
4139                                    68500           ;APPROXIMATION POLYNOMIAL TO COMPUTE ARCTAN(X)
4140  003152' 001000  000357           68520   ATN:    FSIGN                   ;SEE IF ARG IS NEGATIVE
4141  003153' 001000  000374           68540           CM      PSHNEG          ;IF ARG IS NEGATIVE, USE:
4142  003154' 000000  002340'
4143  003155' 000000  003150'
4144  003156' 001000  000374           68560           CM      NEG             ;   ARCTAN(X)=-ARCTAN(-X)
4145  003157' 000000  001175'
4146  003160' 000000  003154'
4147  003161' 001000  000072           68580           LDA     FAC             ;SEE IF FAC .GT. 1
4148  003162' 000000  002467*
4149  003163' 000000  003157'
4150  003164' 001000  000376           68600           CPI     201
4151  003165' 000000  000201
4152  003166' 001000  000332           68620           JC      ATN2
4153  003167' 000000  003205'
4154  003170' 000000  003162'
4155  003171' 001000  000001           68640           LXI     B,201*400+$CODE ;GET THE CONSTANT 1
4156  003172' 000000  100400*
4157  003173' 000000  003167'
4158  003174' 001000  000121           68660           MOV     D,C
4159  003175' 001000  000131           68680           MOV     E,C             ;COMPUTE RECIPROCAL TO USE THE IDENTITY:
4160  003176' 001000  000315           68700           CALL    FDIV            ;   ARCTAN(X)=PI/2-ARCTAN(1/X)
4161  003177' 000000  000655'
4162  003200' 000000  003172'
4163  003201' 001000  000041           68720           LXI     H,FSUBS         ;PUT FSUBS ON THE STACK SO WE WILL RETURN
4164  003202' 000000  000011'
4165  003203' 000000  003177'
4166  003204' 001000  000345           68740           PUSH    H               ; TO IT AND SUBTRACT THE REULT FROM PI/2
4167  003205' 001000  000041           68760   ATN2:   LXI     H,ATNCON        ;EVALUATE APPROXIMATION POLYNOMIAL
4168  003206' 000000  003217'
4169  003207' 000000  003202'
4170  003210' 001000  000315           68780           CALL    POLYX
4171  003211' 000000  002613'
4172  003212' 000000  003206'
4173  003213' 001000  000041           68800           LXI     H,PI2           ;GET POINTER TO PI/2 IN CASE WE HAVE TO
4174  003214' 000000  003070'
4175  003215' 000000  003E11'
4176  003216' 001000  000311           68820           RET                     ; SUBTRACT THE RESULT FROM PI/2
4177
4178                                    68860           ;CONSTANTS FOR ATN
4179  003217' 000000  000011           68880   ATNCON: 11      ;DEGREE
4180  003220' 000000  000112           68900           112     ; ,002866226
4181  003221' 000000  000327           68920           327
4182  003222' 000000  000073           68940           073
4183  003223' 000000  000170           68960           170
4184  003224' 000000  000002           68980           002     ; =-,01616574
4185  003225' 000000  000156           69000           156
4186  003226' 000000  000204           69020           204
4187  003227' 000000  000173           69040           173
4188  003230' 000000  000376           69060           376     ; ,04290961
```