LEVEL II
DOUBLE-PRECISION SUBROUTINE PROGRAM

Catalog Number 26-1704

Radio Shack
TRS-80
MICRO
COMPUTER
SYSTEM

SOFTWARE

The DOUBLE-PRECISION program lets you obtain 15-digit accuracy with sine, cosine, arctangent, natural logarithm, exponential, and square root functions. These functions, together with LEVEL II BASIC'S "DEFDBL", and "DEFINT", provide a high degree of precision, for any applications that need a high level of accuracy.

The compact subroutines are very efficient. If you own a LEVEL II - 4K machine, you can use these routines without concern. Enough memory will normally be left for your main program.

CUSTOM MANUFACTURED IN U.S.A. BY RADIO SHACK TC A DIVISION OF TANDY CORPORATION

INTRODUCTION

The DOUBLE-PRECISION program lets you obtain 15-digit accuracy with sine, cosine, arctangent, natural logarithm, exponential, and square root functions. These functions, together with LEVEL II BASIC'S "DEFDBL", and "DEFINT", provide a high degree of precision, for any applications that need a high level of accuracy.

The compact subroutines are very efficient. If you own a LEVEL II - 4K machine, you can use these routines without concern. Enough memory will normally be left for your main program.

The three programs included are;
SUBROUTINES

The actual Double-Precision Subroutines are designed to be included in your program. To use them, simply execute a GOSUB to the particular routine required. Double-Precision Derived Functions may be included with slight modifications. You can DELETE any routines you won't need to conserve memory.

ACCURACY TEST 1.1

The Accuracy Test is a program designed to demonstrate the degree of precision that can be obtained with the Subroutine Functions. Each Accuracy test asks for a value to test, and an increment value. Each of the tests is then run 13 times, using your values. The results are graphically displayed, to aid you in determining the ACTUAL degree of precision obtained for any given number. In this manner, you can "Pre-test" some of the values that you plan to use in a long complex chain calculation to predict the final degree of accuracy, and the probable amount of error likely to occur.

DOUBLE-PRECISION CALCULATOR 1.1

Converts the TRS-80 to a calculator mode to give you the opportunity to easily experiment with the Double-Precision functions in degrees/angles. A unique "**ILLEGAL ARGUMENT**" sign will appear if the argument you enter will cause an error condition.

All of the subroutines use the "Taylor Expansion Method" to support their precision. Extensive testing indicates that these routines will be accurate to 15 and 16 places in the majority. In a few tests we have determined that a 14th place digit could be in error by one or two units under certain conditions.

THEORY AND OPERATION

SUBROUTINES

All angles are presented in radians.

The square root function is calculated by using the single-precision SQR function. The result is refined into double-precision, by dividing it into the argument, and then averaging with the quotient. This procedure is then repeated. With all other functions, the argument is reduced repeatedly, until it can be evaluated by a power series. The result is obtained by appropriate expansion of the value (of the power series) the same number of times the argument was reduced.

DESCRIPTION

CALL	FUNCTION	ARGUMENT	RESULT
GOSUB 40100	Natural Logarithm	Z	Z2
GOSUB 40200	Exponential	Z	Z2
GOSUB 40300	Cosine	Z	Z2
GOSUB 40400	Sine	Z	Z2
GOSUB 40500	Arctangent	Z	Z2
GOSUB 40600	Square Root	Z	Z3
GOSUB 40620	PI/2	none	Z9
GOSUB 40640	Power Expansion	(subroutine for above functions)	

IMPORTANT NOTE:

In order to use these subroutines, you must define "Z" as a double-precision "variable" early in the program. (See "Level II" Manual, Page 4/3) At the start of your program, insert "DEFDBL Z". You must also keep in mind the fact that these subroutines have some Variable Names reserved "For Subroutine Use Only". The Reserved Variable Names are; Z2, Z3, Z4, Z9, I, IØ, I1, I2, and I3. YOU CANNOT USE THESE NAMES IN YOUR PROGRAM, AND STILL MAINTAIN 16 DIGIT ACCURACY. Once entered, the argument in Z is destroyed. Most functions output by the variable Z2. SQR outputs by the variable Z3. PI/2 outputs as Z9. You must not redefine the value held in Z9. Once calculated, the subroutines for the trigonometric functions expect to find the value for PI/2 stored in Z9.

SUBROUTINE SPECIFICATIONS

NATURAL LOGARITHM:

Purpose of Function	Computes Z2=LOG(Z)
Input	Z
Output	Z2
Reserved Variables Used	I, IØ, I2, I3
Error Conditions	Z< =Ø
Other Subroutines Used	40600, 40640
Calling Procedure	GOSUB 40400

EXPONENTIAL:

Purpose of Function	Computes Z2=EXP(Z)
Input	Z
Output	Z2
Reserved Variables Used	I, IØ
Error Conditions	overflow for Z>88 or Z<-88
Other Subroutines Used	none
Calling Procedure	GOSUB 40200

COSINE:

Purpose of Function	Computes Z2=COS(Z)
Input	Z (in Radians)
Output	Z2 (in Radians)
Reserved Variables Used	Z9
Error Conditions	none
Other Subroutines Used	40620, 40400
Calling Procedure	GOSUB 40300

SINE:

Purpose of Function	Computes Z2=SIN(Z)
Input	Z (in Radians)
Output	Z2 (in Radians)
Reserved Variables Used	I, IØ, I2, Z9
Error Conditions	none
Other Subroutines Used	40620
Calling Procedure	GOSUB 40400

ARCTANGENT:

Purpose of Function	Computes Z2=ATN(Z)
Input	Z (in Radians)
Output	Z2 (in Radians)
Reserved Variables Used	I, IØ, I1, I2, I3, Z3, Z9
Error Conditions	none
Other Subroutines Used	40620, 40640
Calling Procedure	GOSUB 40500

SQUARE ROOT:

Purpose of Function	Computes Z3=SQR(Z)
Input	Z
Output	Z3
Reserved Variables Used	none
Error Conditions	Z<=Ø
Other Subroutines Used	none
Calling Procedure	GOSUB 40600

PI/2:

Purpose of Function	Computes π/2
Input	none
Output	Z9
Reserved Variables Used	Z2, Z4
Error Conditions	none
Other Subroutines Used	40500
Calling Procedure	GOSUB 40620

ACCURACY TEST

This program performs four tests to demonstrate the accuracy of the double-precision functions. The results show the maximum degree of error allowed for any given number:

TEST	**BEST POSSIBLE RESULT**
SIN^2+COS^2-1	Ø
ATN(SIN/COS)	Results = Argument
LOG(EXP)	Results = Arguments
$(SQR)^2$	Results = Arguments

CALCULATOR

The Calculator program lets you enter any function of the form;

XXX(Argument)

wherein XXX is SIN, COS, ATN, LOG, EXP, or SQR. The argument cannot be an expression. All angles are in degrees. If the argument causes an error condition, an "**ILLEGAL ARGUMENT**" message appears. If XXX is not one of the six defined functions, a "WHAT?_" will appear. Type END to stop the program.

EXAMPLES;

Computer;	You type;	Computer displays;
?_	SIN(30)	.4999999999999999
?_	COS(30)	.8660254037844387
?_	LOG(1)	0
?_	EXP(1)	2.718281828459043
?_	ATN(1)	45
?_	SQR(-1)	**ILLEGAL ARGUMENT**
?_	END	READY >_

DERIVED FUNCTIONS

You can also achieve Double-Precision results for Derived Functions — by simply calling out "DEFDBL" for all Constants, and the appropriate GOSUB for each part of the Derived Function that requires the Double Precision Subroutine Series. We have listed the Derived Functions, and the function as expressed in terms of LEVEL II Basic;

FUNCTION	**FUNCTION EXPRESSED IN TERMS OF LEVEL II BASIC FUNCTIONS**
SECANT	SEC(X) = 1/COS(X)
COSECANT	CSC(X) = 1/SIN(X)
COTANGENT	COT(X) = 1/TAN(X)
INVERSE SINE	ARCSIN(X) = ATN(X/SQR (—X*X+1))
INVERSE COSINE	ARCCOS(X) = —ATN(X/SQR (—X*X+1))+1.5708
INVERSE SECANT	ARCSEC(X) = ATN(SQR(X*X—1)) +(SGN(X)—1)*1.5708
INVERSE COSECANT	ARCCSC(X) = ATN(1/SQR (X*X—1))+(SGN(X)—1)*1.5708
INVERSE COTANGENT	ARCCOT(X) = —ATN(X)+1.5708
HYPERBOLIC SINE	SINH(X) = (EXP(X)—EXP(—X))/2
HYPOBOLIC COSINE	COSH(X) = (EXP(X)/EXP(—X))/2
HYPERBOLIC TANGENT	TANH(X) = —EXP(—X)/(EXP(X) +EXP(—X)*2+1
HYPERBOLIC SECANT	SECH(X) = 2/(EXP(X)+EXP(—X))
HYPERBOLIC COSECANT	CSCH(X) = 2/(EXP(X)—EXP(—X))
HYPERBOLIC COTANGENT	COTH(X) = EXP(—X)/(EXP(X) —EXP(—X))*2/1
INVERSE HYPERBOLIC SINE	ARGSINH(X) = LOG(X+SQR (X*X+1))
INVERSE HYPERBOLIC COSINE	ARGCOSH(X) = LOG(X+SQR (X*X—1))
INVERSE HYPERBOLIC TANGENT	ARGTANH(X) = LOG ((1+X)/ (1—X))/2
INVERSE HYPERBOLIC SECANT	ARGSECH(X) = LOG((SQR(—X*X+ 1)+1)/X)
INVERSE HYPERBOLIC COSECANT	ARGSCH(X) = LOG((SGN(X)*SQR (X*X+1)+1)/X)
INVERSE HYPERBOLIC COTANGENT	ARGCOTH(X) = LOG((X+1)/ (X—1))/2

```
40000 END
40020 ' DOUBLE PRECISION ROUTINES
40040 'COPYRIGHT TANDY CORP. 1978
40050 '
40052 ' *** NATURAL LOG ***
40054 '
40100 Z2=LOG(Z):I0=0:I2=SGN(Z2):IF I2<0 Z=1/Z
40110 IF Z<1.065 THEN 40130
40120 GOSUB 40600:Z=Z3:I0=I0+1:GOTO 40110
40130 Z=(Z-1)/(Z+1):Z2=Z*Z:Z3=Z
40140 I3=9:GOSUB 40640:Z=Z+Z
40150 IF I0=0 THEN 40170
40160 FOR I=1 TO I0:Z=Z+Z:NEXT
40170 Z2=Z*I2:RETURN
40180 '
40182 ' *** EXPONENTIAL ***
40184 '
40200 Z2=EXP(ABS(Z)):I0=0
40210 IF Z*Z<.004 THEN 40230
40220 Z=Z/2:I0=I0+1:GOTO 40210
40230 Z2=1:FOR I2=8 TO 1 STEP -1:Z2=Z2*Z/I2+1:NEXT
40240 IF I0=0 THEN 40260
40250 FOR I=1 TO I0:Z2=Z2*Z2:NEXT
40260 RETURN
40270 '
40272 ' *** COSINE ***
40274 '
40300 GOSUB 40620 'COS
40310 Z=Z9-Z
40320 '
40322 ' *** SINE ***
40324 '
40400 GOSUB 40620 'SIN
40410 I0=0:I2=SGN(Z):Z2=Z9+Z9:Z=ABS(Z):Z=Z-INT(Z/(Z2+Z2))*(Z2+Z2)
40420 IF Z>Z2 THEN Z=Z-Z2:I2=-I2
40430 IF Z>Z9 THEN Z=Z2-Z
40440 IF ABS(Z)<.063 THEN 40460
40450 Z=Z/3:I0=I0+1:GOTO 40440
40460 Z2=-Z*Z:Z2=(((Z2/42+1)*Z2/20+1)*Z2/6+1)*Z
40470 IF I0=0 THEN 40490
40480 FOR I=1 TO I0:Z2=(3-4*Z2*Z2)*Z2:NEXT
40490 Z2=Z2*I2:RETURN
40492 '
40494 ' *** ARCTANGENT ***
40496 '
40500 GOSUB 40620
40510 I0=0:I1=0:I2=SGN(Z)
40520 Z=ABS(Z):IF Z>1 Z=1/Z:I1=1
40530 IF Z<.077 THEN 40560
40540 Z2=Z:Z=Z*Z+1:GOSUB 40600:Z=Z2/(Z3+1)
40550 I0=I0+1:GOTO 40530
40560 Z3=Z:I3=-11:GOSUB 40640
```

```
40570 IF I0=0 GOTO 40578
40574 FOR I=1 TO I0:Z=Z+Z:NEXT
40578 Z2=Z
40582 IF I1=1 Z2=Z9-Z2
40586 Z2=Z2*I2
40590 RETURN
40592 '
40594 ' *** SQUARE ROOT ***
40596 '
40600 Z3=SQR(Z):Z3=(Z3+Z/Z3)/2:Z3=(Z3+Z/Z3)/2:RETURN
40610 '
40612 ' *** Z9 = PI/2 ***
40614 '
40620 IF Z9<>0 RETURN
40630 Z4=Z:Z=1:GOSUB 40510:Z9=Z2+Z2:Z=Z4:RETURN
40632 '
40634 ' *** POWER EXPANSION SUBROUTINE ***
40636 '
40640 Z2=Z*Z:FOR I=3 TO ABS(I3) STEP 2:Z3=SGN(I3)*Z3*Z2
40650 Z=Z+Z3/I:NEXT:RETURN
```

```
100 ' DOUBLE PRECISION LEVEL II BASIC FUNCTIONS
110 ' SIN,COS,ATN,LOG,EXP,SQR
120 '
130 ' ACCURACY TEST
140 'COPYRIGHT TANDY CORP. 1978
145 DEFDBL Z:DEFINT I-N:ONERRORGOTO9000
150 CLS:PRINTTAB(20)"ACCURACY TEST 1.1":PRINT
160 FORJ=1TO4:PRINTTAB(15);J;") ";:GOSUB300:NEXT
165 PRINTTAB(15);" 0 ) STOP":PRINT
170 PRINTTAB(15);:INPUT"ENTER SELECTION";J
180 IFJ=0THEN ONERRORGOTO0:END ELSE GOSUB 200:GOTO150
200 M=0:GOSUB400:CLS:PRINTTAB(15);:GOSUB300:PRINT"ARGUMENT"TAB(32)"RESULT
210 ER=0:ZA=ZS+CDBL(M)*ZI:Z=ZA
220 ONJGOTO1000,2000,3000,4000
230 IFER=0THENPRINTZA;TAB(25);Z2
240 M=M+1:IFM<13THEN 210
250 PRINT@970,"PRESS ENTER TO CONTINUE";:INPUTA$:RETURN
300 ONJGOTO310,320,330,340
310 PRINT"(SIN ↑ 2) + (COS ↑ 2) -1":RETURN
320 PRINT"ATN (SIN / COS)":RETURN
330 PRINT"LOG (EXP)":RETURN
340 PRINT"(SQR) ↑ 2":RETURN
400 PRINT:PRINT"ENTER INITIAL ARGUMENT VALUE"TAB(32);:INPUT ZS
410 PRINT"ENTER INCREMENTING VALUE"TAB(32);:INPUTZI:RETURN
1000 GOSUB40400:IFER=1THEN RESUME 230
1010 ZB=Z2:Z=ZA:GOSUB40300:IFER=1THEN RESUME 230
1020 Z2=ZB*ZB+Z2*Z2-1D0:GOTO230
2000 GOSUB40620:ZC=INT(ZA/Z9):ZD=ZA-Z9*ZC:Z=ZD:GOSUB40400:IFER=1THEN RESUME 230
2010 ZB=Z2:Z=ZD:GOSUB40300:IFER=1THEN RESUME 230
2020 Z=ZB/Z2:GOSUB40500:IFER=1THEN RESUME 230
2030 Z2=Z2+Z9*ZC:GOTO230
3000 GOSUB40200:IFER=1THEN RESUME 230 ELSE Z=Z2
3010 GOSUB40100:IFER=1THEN RESUME 230 ELSE GOTO 230
4000 GOSUB40600:IFER=1THEN RESUME230 ELSE Z2=Z3*Z3:GOTO230
9000 PRINTZA;TAB(25)"** ILLEGAL ARGUMENT **":ER=1:RETURN
40000 END
40020 ' DOUBLE PRECISION ROUTINES
40050 '
40052 ' *** NATURAL LOG ***
40054 '
40100 Z2=LOG(Z):I0=0:I2=SGN(Z2):IF I2<0 Z=1/Z
40110 IF Z<1.065 THEN 40130
40120 GOSUB 40600:Z=Z3:I0=I0+1:GOTO 40110
40130 Z=(Z-1)/(Z+1):Z2=Z*Z:Z3=Z
40140 I3=9:GOSUB 40640:Z=Z+Z
40150 IF I0=0 THEN 40170
40160 FOR I=1 TO I0:Z=Z+Z:NEXT
40170 Z2=Z*I2:RETURN
40180 '
40182 ' *** EXPONENTIAL ***
40184 '
40200 Z2=EXP(ABS(Z)):I0=0
40210 IF Z*Z<.004 THEN 40230
40220 Z=Z/2:I0=I0+1:GOTO 40210
```

```
40230 Z2=1:FOR I2=8 TO 1 STEP -1:Z2=Z2*Z/I2+1:NEXT
40240 IF I0=0 THEN 40260
40250 FOR I=1 TO I0:Z2=Z2*Z2:NEXT
40260 RETURN
40270 '
40272 ' *** COSINE ***
40274 '
40300 GOSUB 40620 'COS
40310 Z=Z9-Z
40320 '
40322 ' *** SINE ***
40324 '
40400 GOSUB 40620 'SIN
40410 I0=0:I2=SGN(Z):Z2=Z9+Z9:Z=ABS(Z):Z=Z-INT(Z/(Z2+Z2))*(Z2+Z2)
40420 IF Z>Z2 THEN Z=Z-Z2:I2=-I2
40430 IF Z>Z9 THEN Z=Z2-Z
40440 IF ABS(Z)<.063 THEN 40460
40450 Z=Z/3:I0=I0+1:GOTO 40440
40460 Z2=-Z*Z:Z2=(((Z2/42+1)*Z2/20+1)*Z2/6+1)*Z
40470 IF I0=0 THEN 40490
40480 FOR I=1 TO I0:Z2=(3-4*Z2*Z2)*Z2:NEXT
40490 Z2=Z2*I2:RETURN
40492 '
40494 ' *** ARCTANGENT ***
40496 '
40500 GOSUB 40620
40510 I0=0:I1=0:I2=SGN(Z)
40520 Z=ABS(Z):IF Z>1 Z=1/Z:I1=1
40530 IF Z<.077 THEN 40560
40540 Z2=Z:Z=Z*Z+1:GOSUB 40600:Z=Z2/(Z3+1)
40550 I0=I0+1:GOTO 40530
40560 Z3=Z:I3=-11:GOSUB 40640
40570 IF I0=0 GOTO 40578
40574 FOR I=1 TO I0:Z=Z+Z:NEXT
40578 Z2=Z
40582 IF I1=1 Z2=Z9-Z2
40586 Z2=Z2*I2
40590 RETURN
40592 '
40594 ' *** SQUARE ROOT ***
40596 '
40600 Z3=SQR(Z):Z3=(Z3+Z/Z3)/2:Z3=(Z3+Z/Z3)/2:RETURN
40610 '
40612 ' *** Z9 = PI/2 ***
40614 '
40620 IF Z9<>0 RETURN
40630 Z4=Z:Z=1:GOSUB 40510:Z9=Z2+Z2:Z=Z4:RETURN
40632 '
40634 ' *** POWER EXPANSION SUBROUTINE ***
40636 '
40640 Z2=Z*Z:FOR I=3 TO ABS(I3) STEP 2:Z3=SGN(I3)*Z3*Z2
40650 Z=Z+Z3/I:NEXT:RETURN
```

```
100 ' DOUBLE PRECISION LEVEL II BASIC FUNCTIONS
110 ' SIN,COS,ATN,LOG,EXP,SQR
120 '
130 ' CALCULATOR
140 'COPYRIGHT TANDY CORP. 1978
150 DEFDBL Z:DEFINT I-N:ONERRORGOTO400
160 FORM=1TO6:READ F$(M):NEXT:GOSUB40620
170 DATA SIN,COS,ATN,LOG,EXP,SQR
180 CLS:PRINTTAB(15)"DOUBLE PRECISION CALCULATOR 1.1":PRINT
185 PRINTTAB(19)"ANGLES ARE IN DEGREES"
190 PRINT"ENTER FUNCTIONS -- IE. 'SIN(30)' -- 'END' TO STOP
200 INPUT A$:F1$=LEFT$(A$,3):IFA$="END"ORA$="STOP"THENONERRORGOTO0:END
210 FORM=1TO6:IFF$(M)=F1$THEN220ELSENEXT:PRINTTAB(15)"WHAT?":GOTO200
220 Z=CDBL(VAL(MID$(A$,5,LEN(A$)-5)))
230 IFM=1ORM=2THEN Z=Z*Z9/9D1
240 ONMGOSUB40400,40300,40500,40100,40200,40600:IFM=6THENZ2=Z3
250 IFM=3THENZ2=Z2*9D1/Z9
260 PRINTTAB(15);Z2:GOTO200
400 PRINTTAB(15)"** ILLEGAL ARGUMENT **":RESUME200
40000 END
40020 ' DOUBLE PRECISION ROUTINES
40050 '
40052 ' *** NATURAL LOG ***
40054 '
40100 Z2=LOG(Z):I0=0:I2=SGN(Z2):IF I2<0 Z=1/Z
40110 IF Z<1.065 THEN 40130
40120 GOSUB 40600:Z=Z3:I0=I0+1:GOTO 40110
40130 Z=(Z-1)/(Z+1):Z2=Z*Z:Z3=Z
40140 I3=9:GOSUB 40640:Z=Z+Z
40150 IF I0=0 THEN 40170
40160 FOR I=1 TO I0:Z=Z+Z:NEXT
40170 Z2=Z*I2:RETURN
40180 '
40182 ' *** EXPONENTIAL ***
40184 '
40200 Z2=EXP(ABS(Z)):I0=0
40210 IF Z*Z<.004 THEN 40230
40220 Z=Z/2:I0=I0+1:GOTO 40210
40230 Z2=1:FOR I2=8 TO 1 STEP -1:Z2=Z2*Z/I2+1:NEXT
40240 IF I0=0 THEN 40260
40250 FOR I=1 TO I0:Z2=Z2*Z2:NEXT
40260 RETURN
40270 '
40272 ' *** COSINE ***
40274 '
40300 GOSUB 40620 'COS
40310 Z=Z9-Z
40320 '
40322 ' *** SINE ***
40324 '
```

```
40400 GOSUB 40620 'SIN
40410 I0=0:I2=SGN(Z):Z2=Z9+Z9:Z=ABS(Z):Z=Z-INT(Z/(Z2+Z2))*(Z2+Z2)
40420 IF Z>Z2 THEN Z=Z-Z2:I2=-I2
40430 IF Z>Z9 THEN Z=Z2-Z
40440 IF ABS(Z)<.063 THEN 40460
40450 Z=Z/3:I0=I0+1:GOTO 40440
40460 Z2=-Z*Z:Z2=(((Z2/42+1)*Z2/20+1)*Z2/6+1)*Z
40470 IF I0=0 THEN 40490
40480 FOR I=1 TO I0:Z2=(3-4*Z2*Z2)*Z2:NEXT
40490 Z2=Z2*I2:RETURN
40492 '
40494 ' *** ARCTANGENT ***
40496 '
40500 GOSUB 40620
40510 I0=0:I1=0:I2=SGN(Z)
40520 Z=ABS(Z):IF Z>1 Z=1/Z:I1=1
40530 IF Z<.077 THEN 40560
40540 Z2=Z:Z=Z*Z+1:GOSUB 40600:Z=Z2/(Z3+1)
40550 I0=I0+1:GOTO 40530
40560 Z3=Z:I3=-11:GOSUB 40640
40570 IF I0=0 GOTO 40578
40574 FOR I=1 TO I0:Z=Z+Z:NEXT
40578 Z2=Z
40582 IF I1=1 Z2=Z9-Z2
40586 Z2=Z2*I2
40590 RETURN
40592 '
40594 ' *** SQUARE ROOT ***
40596 '
40600 Z3=SQR(Z):Z3=(Z3+Z/Z3)/2:Z3=(Z3+Z/Z3)/2:RETURN
40610 '
40612 ' *** Z9 = PI/2 ***
40614 '
40620 IF Z9<>0 RETURN
40630 Z4=Z:Z=1:GOSUB 40510:Z9=Z2+Z2:Z=Z4:RETURN
40632 '
40634 ' *** POWER EXPANSION SUBROUTINE ***
40636 '
40640 Z2=Z*Z:FOR I=3 TO ABS(I3) STEP 2:Z3=SGN(I3)*Z3*Z2
40650 Z=Z+Z3/I:NEXT:RETURN
```

IMPORTANT NOTICE

ALL RADIO SHACK COMPUTER PROGRAMS ARE DISTRIBUTED ON AN "AS IS" BASIS WITHOUT WARRANTY

Radio Shack shall have no liability or responsibility to customer or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by computer equipment or programs sold by Radio Shack, including but not limited to any interruption of service, loss of buisness or anticipatory profits or consequential damages resulting from the use or operation of such computer or computer programs.

NOTE: Good data processing procedure dictates that the user test the program, run and test sample sets of data, and run the system in parallel with the system previously in use for a period of time adequate to insure that results of operation of the computer or program are satisfactory.

RADIO SHACK A DIVISION OF TANDY CORPORATION

U.S.A.: FORT WORTH, TEXAS 76102
CANADA: BARRIE, ONTARIO L4M 4W5

TANDY CORPORATION

AUSTRALIA	BELGIUM	U K
280-316 VICTORIA ROAD	PARC INDUSTRIEL DE NANINNE	BILSTON ROAD
RYDALMERE, N S W 2116	5140 NANINNE	WEDNESBURY STAFFS WS10 7JN

PRINTED IN U.S.A.