
-- GDA3.TXT

```
10 ON ERROR GOTO 3050
20 '
30 '  GENERAL DATA ACQUISITION
40 '  WRITTEN BY JAMES R. ASHBURN
50 '  JUNE  1986
60 '
70 '  SET COLORS
80 '
90 CLS:SCREEN 2:CLOSE
100 LOCATE 1,1
110 '
120 '    INPUT NUMBER OF CHANNELS, ETC.
130 '
140 PRINT "Number of channels ( 1 - 8 ): ":
PRINT "Starting channel   ( 1 -   ): ":
PRINT "Number of points per minute : ":PRINT
150 PRINT "Do you want manual adjustment of the lock-in (Y/N) ? "
160 LOCATE 1,31:INPUT "",CHN%:IF (CHN%<1 OR CHN%>8) THEN 160
170   PTSLIM1%=10000/CHN%:PTSLIM2%=2000/CHN%:
IF PTSLIM1%>2000 THEN PTSLIM1%=2000
180   CHN1%=CHN%-1:
CHE%=8-CHN1%
190 LOCATE 2,25:PRINT CHE%:LOCATE 2,31:INPUT "",CHS%:
IF (CHS%<1 OR CHS%>CHE%) THEN 190
200   CHS1%=CHS%-1
210 LOCATE 3,31:INPUT "",PPM%
220   SPP=60/PPM%
230 LOCATE 5,54:INPUT "",ANS$:IF (ANS$<>"Y" AND ANS$<>"y") THEN CHL%=0:GOTO 270
240 PRINT "For which channel ..... (";CHS%;"-";CHS%+CHN1%;") ? "
250 LOCATE 6,52:INPUT "",CHL%:IF (CHL%<CHS% OR CHL%>CHS%+CHN1%) THEN 250
270   CHL1%=CHL%-1
280 '
290 '    SET UP ARRAYS AND INITIALIZE VARIABLES
300 '
310 DIM DT1(CHN1%,PTSLIM1%),DT2(CHN1%,PTSLIM2%)
320 DIM OFFSET%(100),TYPES%(6),GAIN%(16),RETURNAGAIN%(16),ADCDATA%(8)
330 DAOI%=0:DAOI$(0)="-":DAOI$(1)="*"
340 PTS2%=1:PTS1%=0:CHA1%=CHS1%
350 XON$=CHR$(17):XOF$=CHR$(19)
360 FOR I%=0 TO 7:OTTO%(I%)=0:NEXT I%
370 O(1)=0:O(2)=0:S(1)=0:S(2)=0:XYI%=0
380 CC%(1)=0:CC%(2)=1
390 '
400 '    SET UP SCREEN
410 '

```

```

420 CLS
430 LOCATE 1,63:PRINT "Maximum:"
440 LOCATE 2,65:PRINT PTSLIM1%;" d.p."
450 LOCATE 3,65:PRINT INT(PTSLIM1%/PPM%);:PRINT " min"
460 LOCATE 7,63:PRINT "(D)ata Acq: ";:LOCATE 7,75:PRINT DAOI$(DAOI%)
470 LOCATE 5,63:PRINT "Point #: "
480 LOCATE 6,63:PRINT "#Pts Avgd:"
490 LOCATE 8,63:PRINT "(C)lear graph"
500 LOCATE 9,63:PRINT "CAS Input"
510 FOR K%=CHS1% TO CHS1%+CHN1%
520 LOCATE 10+K%-CHS1%,62:PRINT STR$(K%+1);:PRINT DAOI$(-OTTO%(K%));"L"
530 NEXT K%
540 LOCATE 10+CC%(1),62:PRINT "X"
550 LOCATE 10+CC%(2),62:PRINT "Y"
560 LOCATE 23,63:PRINT "ctrl-end:FINISHED"
570 LOCATE 24,63:PRINT "alt-R :RESTART";
580 VIEW (0,0)-(479,199)
590 WINDOW (-4096,-4096)-(4096,4096)
600 GOSUB 630
610 GOTO 2370
620 '
630 ' SET UP GRAPHICS SCREEN
640 '
650 LINE (-4096,-4096)-(4096,4096),,B
660 FOR K%=-4096 TO 4096 STEP 8.192000E+02
670 LINE (K%+1,-4096)-(K%+1,-3996):LINE (K%+1,4096)-(K%+1,3996)
680 LINE (-4096,K%)-(-3996,K%):LINE (4096,K%)-(3996,K%)
690 NEXT K%
700 LINE (0,3900)-(0,4096):LINE (0,-3900)-(0,-4096):
LINE (3900,0)-(4096,0):LINE (-3900,0)-(-4096,0):
LINE (100,0)-(-100,0):LINE (0,100)-(0,-100)
710 RETURN
720 '
730 ' CHANGE SENSITIVITY AND/OR CHANNEL(S) TO BE PLOTTED
740 '
750 IF KEE$<>CHR$(13) THEN LOCATE 18,65:IF KEE$=CHR$(8) THEN
XYS$=LEFT$(XYS$,LEN(XYS$)+(LEN(XYS$)<>0)):RETURN ELSE YYS$=XYS$+KEE$:RETURN
760 EE%=INT(VAL(RIGHT$(XYS$,LEN(XYS$)+(LEN(XYS$)>1)))):IF NOT((LEFT$(XYS$,1)="C"
OR LEFT$(XYS$,1)="c") AND (EE%>=CHS% AND EE%<=CHS%+CHN1%)) THEN
S(XYI%)=VAL(XYS$)-S(XYI%)*(VAL(XYS$)=0):GOSUB 1110:GOTO 785
770 LOCATE 10+CC%(XYI%),62:PRINT MID$("XY ",3+XYI%*(CC%(3-
XYI%)=CC%(XYI%)),1):CC%(XYI%)=EE%-CHS%:LOCATE 10+CC%(XYI%),62:PRINT
MID$("XY",XYI%,1)
785 YYS$="":SCHI%=-1:XYI%=0:LOCATE 18,63:PRINT SPC(10)
790 RETURN
800 '
810 ' READ ADCDATA ARRAY
820 '

```

```

830 SCI%=0
840 KEE$=INKEY$:IF XYI% THEN GOSUB 730:PRINT XY$;" ":KEE%=0:GOTO 860
850 KEE%=0:IF KEE$<>" " THEN KEE1$=LEFT$(KEE$,1):KEE2$=RIGHT$(KEE$,1):
KEE%=INSTR(" AaSsDdXxYyCcLl",KEE1$)-16*(ASC(KEE2$)=117)-17*(VAL(KEE1$)>0)
-18*(ASC(KEE2$)=19):RETURN
860 CALL INPADC%(POSITION%,RETURN_GAIN%(0),ADC_DATA%(0))
870 IF NOT(CHL1%) THEN IF ABS(ADC_DATA%(CHL1%)/RETURN_GAIN%(CHL1%))>4000
THEN GOSUB 1970:GOSUB 2040'ADJ. SETTINGS IF OVL D.
880 I%=CHS1%-1
890 WHILE (I%-(CHN1%+CHS1%)) AND NOT(SCI%)
900 I%=I%+1
910 IF GAIN%(I%)=RETURN_GAIN%(I%) THEN
TEMP=ADC_DATA%(I%)*2.500000/RETURN_GAIN%(I%): DT2(I%-CHS1%,PTS2%)=-
(I%<>CHL1%)*TEMP-(I%=CHL1%)*(TEMP/10000)*SEN: NTP=DT2(I%-
CHS1%,PTS2%):LOCATE 10+I%-CHS1%,67:GOSUB 1030 ELSE SCI%=-1
920 WEND
930 I%=CHS1%-1
940 WHILE (I%-(CHN1%+CHS1%)) AND NOT(SCI%)
950 I%=I%+1
960 IF ABS(ADC_DATA%(I%))>4000 THEN GAIN%(I%)=GAIN%(I%)/(1-(GAIN%(I%)<>1)):
GOSUB 1670:LOCATE 10+I%-CHS1%,65:PRINT CHR$(76-INT(LOG(GAIN%(I%))/LOG(2)
+5.000000E-01)):SCI%=-1:BEEP
970 IF NOT(OTTO%(I%)) THEN 990
980 IF ABS(ADC_DATA%(I%))<1500 AND GAIN%(I%)<>2048 THEN GAIN%(I%)=GAIN%(I%)*2:
GOSUB 1670:LOCATE 10+I%-CHS1%,65:PRINT CHR$(76-INT(LOG(GAIN%(I%))/
LOG(2)+5.000000E-01)):SCI%=-1
990 WEND
1000 IF SCI% THEN 830
1010 RETURN
1020 '
1030 ' PRINT IN APPROPRIATE UNITS
1040 '
1050 UNITS$="m"
1060 IF ABS(NTP)<1 THEN NTP=NTP*1000:UNITS$="√ä":IF ABS(NTP)<1 THEN
NTP=NTP*1000: UNITS$="n":GOTO 1080 ELSE GOTO 1080
1070 IF ABS(NTP)>=1000 THEN NTP=NTP/1000:UNITS$=""
1080 PRINT USING "+###.###";NTP;:PRINT " ";UNITS$;"V ";
1090 RETURN
1100 '
1110 ' PRINT SENSITIVITY
1120 '
1130 LOCATE 20,63:PRINT "XS: ";:NTP=S(1):GOSUB 1030
1140 LOCATE 22,63:PRINT "YS: ";:NTP=S(2):GOSUB 1030
1150 RETURN
1160 '
1170 ' AVERAGE DATA
1180 '
1190 IF PTS2%=0 THEN RETURN

```

```

1200 PTS1%=PTS1%+1
1210 FOR K%=0 TO CHN1%
1220   SUM(K%)=0
1230   FOR J%=1 TO PTS2%
1240     SUM(K%)=SUM(K%)+DT2(K%,J%)
1250   NEXT J%
1260   DT1(K%,PTS1%)=SUM(K%)/PTS2%
1270 NEXT K%
1280 IF (S(1)*S(2)=0) THEN 1340
1290 XTP=DT1(CC%(1),PTS1%)-O(1):YTP=DT1(CC%(2),PTS1%)-O(2)
1300 IF SCHI% THEN XTP=0:GOSUB 1390:YTP=0:GOSUB 1440:SCHI%=0
1310 IF ABS(XTP)>(S(1)/2) THEN GOSUB 1390
1320 IF ABS(YTP)>(S(2)/2) THEN GOSUB 1440
1330 PSET ((DT1(CC%(1),PTS1%)-O(1))*8192/S(1),(DT1(CC%(2),PTS1%)-
O(2))*8192/S(2))
1340 LOCATE 5,73:PRINT USING "#####";PTS1%
1350 LOCATE 6,73:PRINT USING "#####";PTS2%
1360 PTS2%=0:TIME1=TIME2
1370 RETURN
1380 '
1390 '   CHANGE X-OFFSET
1400 '
1410 O(1)=INT((DT1(CC%(1),PTS1%)+4.000000E-01*S(1)*SGN(XTP))*10/S(1)+5.000000E-
01)*S(1)/10:LOCATE 19,63:      PRINT "XO: ";:NTP=O(1):GOSUB 1030
1420 RETURN
1430 '
1440 '   CHANGE Y-OFFSET
1450 '
1460 O(2)=INT((DT1(CC%(2),PTS1%)+4.000000E-01*S(2)*SGN(YTP))*10/S(2)+5.000000E-
01)*S(2)/10:LOCATE 21,63:      PRINT "YO: ";:NTP=O(2):GOSUB 1030
1470 RETURN
1480 '
1490 '   ACQUIRE DATA
1500 '
1510 GOSUB 810' READ ADCDATA ARRAY
1520 ON KEE%+1 GOTO
1610,1610,1590,1590,1580,1580,1550,1550,1560,1560,1570,1570,1540,1540,1541,1541,
2640,1600,1530
1530 RUN
1540 CLS:GOSUB 630:GOTO 1510
1541 SENC%=SENC%-1:GOSUB 2060:GOTO 1510
1550 DAOI%=ABS(DAOI%-1):LOCATE 7,75:PRINT DAOI$(DAOI%):GOTO 1510
1560 XYI%=1:LOCATE 18,63:PRINT "X:":GOTO 1510
1570 XYI%=2:LOCATE 18,63:PRINT "Y:":GOTO 1510
1580 GAIN%(CHA1%)=GAIN%(CHA1%)/(1-(GAIN%(CHA1%)<>1)):GOSUB 1670:LOCATE 10+CHA1%-
CHS1%,65:PRINT CHR$(76-INT(LOG(GAIN%(CHA1%))/LOG(2)+5.000000E-01)):GOTO 1510
1590 OTTO%(CHA1%)=NOT(OTTO%(CHA1%)):LOCATE 10+CHA1%-CHS1%,64:
PRINT DAOI$(-OTTO%(CHA1%)):GOTO 1510

```

```

1600 PCHA1%=VAL(KEE$)-1:IF (PCHA1%>=CHS1% AND PCHA1%<=CHS1%+CHN1%) THEN
LOCATE 10+CHA1%-CHS1%,63:PRINT RIGHT$(STR$(CHA1%+1),1):CHA1%=PCHA1%:
LOCATE 10+CHA1%-CHS1%,63:PRINT ">":GOTO 1510
1610 IF DAOI%=0 THEN TIME1=TIMER:PTS2%=1:GOTO 1510
1620 TIME2=TIMER:IF ((TIME2-TIME1)>SPP OR PTS2%=PTSLIM2%) THEN GOSUB 1170
1630 PTS2%=PTS2%+1
1640 IF PTS1%=PTSLIM1% THEN 2640
1650 GOTO 1490
1660 '
1670 '     SET/ADJUST GAIN
1680 '
1690 CALL SETGAIN%(POSITION%,GAIN%(0))
1700 RETURN
1710 '
1720 '     WAIT FOR RESPONSE
1730 '
1740 RES$=""
1750 IF LOC(1)=0 THEN 1750
1760 RES1$=INPUT$(LOC(1),#1)
1770 IF RES1$="?" THEN PRINT "LOCKIN ERROR":END
1780 IF RES1$="*" THEN RETURN
1790 RES$=RES$+RES1$
1800 GOTO 1750
1810 RETURN
1820 '
1830 '     PAUSE
1840 '
1850 T1=TIMER
1860 T2=TIMER:IF T2-T1<PAUSE% THEN 1860
1870 RETURN
1880 '
1890 '     CALCULATE SENSITIVITY AND OFFSET
1900 '
1910 SENM%=SENC% MOD 3 + 1
1920 SENM%=SENM%-2*(SENM%=3)
1930 SEN=SENM%*10^(INT(SENC%/3)-4)' IN MILLIVOLTS
1950 RETURN
1960 '
1970 '     ADJUST SENSITIVITY AND OFFSET
1980 '
2000 SENC%=SENC%+1:GOSUB 2060:RETURN
2030 '
2040 '     SET SENSITIVITY AND OFFSET
2050 '
2060 PRINT #1,"SEN ";SENC%
2070     GOSUB 1720
2080     XOFF$=STR$(-XOFF%)
2110 GOSUB 1890:                                REM  CALCULATE SENSITIVITY AND OFFSET

```

```

2120 PAUSE%=3:GOSUB 1830:          REM PAUSE FOR LOCKIN TO SETTLE
2130 RETURN
2250 '
2260 '     READ SENSITIVITY AND OFFSET
2270 '
2280 PRINT #1,"SEN"
2290 GOSUB 1720:                   REM WAIT FOR RESPONSE
2300 SENC%=VAL(RES$)
2340 GOSUB 1890:                   REM FIND SENS. AND OFFS.
2350 RETURN
2360 '
2370 '     MAIN PROGRAM
2380 '
2390 IF CHL1%>-1 THEN OPEN "COM1:110,N,7,2,DS" AS #1:GOSUB 2260
2400 '
2410 '     INITIALIZE DATA ACQUISITION SYSTEM
2420 '
2430 DMSERR%=0
2440 TYPES%(0)=221
2450 TYPES%(1)=234
2460 TYPES%(6)=231
2470 OPEN "DI-AN" AS #2 LEN=10
2480 FIELD #2,5 AS A$,5 AS B$
2490 GET #2
2500 A%=VAL(A$):B%=VAL(B$)
2510 DEF SEG=A$:CALL B%(TYPES%(0),OFFSET%(0))
2520 '
2530 '     SET GAIN FOR DAS
2540 '
2550 POSITION%=1
2560 FOR K%=0 TO 7:GAIN%(K%)=1:NEXT K%: REM ADJUSTED AUTOMATICALLY
2590 SETGAIN%=OFFSET%(5)
2599 GOSUB 1670:                   REM SET GAIN
2600 INPADC%=OFFSET%(1)
2601 SCI%=0:CALL INPADC%(POSITION%,RETURN_GAIN%(0),ADC_DATA%(0))
2602 FOR I%=0 TO 7:IF GAIN%(I%)<>RETURN_GAIN%(I%) THEN SCI%=-1
2603 NEXT I%:IF SCI% THEN 2601
2620 TIME1=TIMER
2630 GOTO 1480:                     REM ACQUIRE DATA
2640 '
2650 '     SAVE THE DATA FILE
2660 '
2670 VIEW:CLS
2680 INPUT "INSERT DATA DISK IN DRIVE A: AND HIT <RETURN> ",ANS$
2690 PRINT "DIRECTORY OF DRIVE A:"
2700 FILES "A:"
2710 INPUT "ENTER NAME OF DATA FILE (INCLUDING DRIVE, PATH, AND SUFFIX):
",FLNME$:OPEN FLNME$ FOR OUTPUT AS #3

```

```

2720 PRINT #3,""
2730 PRINT #3,"LINE"
2740 PRINT #3,"":PRINT #3,""
2750 NG%=INT((PTS1%-1)/200+1)
2760 PRINT #3,"2,";NG%
2770 PRINT #3,"M"
2780 PRINT #3,"0,1"
2790 PRINT #3,"0,1,1"
2800 FOR I%=1 TO NG%
2810     NP2%=200+(I%=NG%)*(199-((PTS1%-1) MOD 200))
2820     PRINT #3,NP2%";",113,0,20,40"
2830     PRINT #3,DFN$
2840     PRINT #3,"0,1,1"
2850     FOR J%=1 TO NP2%
2860         FOR K%=0 TO CHN1%
2870             PRINT #3,DT1(K%,J%+200*(I%-1));:IF K%<CHN1% THEN PRINT #3,",";
2880             NEXT K%
2890         PRINT #3,""
2900     NEXT J%
2910 NEXT I%
2920 PRINT #3,""
2930 FOR J%=1 TO 10
2940     PRINT #3,"0"
2950 NEXT J%
2960 CLOSE #1,#2,#3
2970 INPUT "DO YOU WANT TO MANIPULATE THE DATA FILE NOW (Y/N) ";ANS$
2980 IF (ANS$="Y" OR ANS$="y") THEN RUN "C:\DAS\1-MANIP.BAS"
2990 INPUT "DO YOU WANT TO RUN ANOTHER TEST (Y/N) ";ANS$
3000 IF (ANS$="Y" OR ANS$="y") THEN RUN
3010 SYSTEM
3020 '
3030 '  ERROR HANDLING ROUTINE
3040 '
3050 IF ERR=53 THEN PRINT "NO FILES":RESUME 2710
3060 IF ERR=61 THEN CLS:PRINT "DISK FULL":RESUME 2680
3070 IF ERR=71 THEN CLS:PRINT "DISK NOT READY":RESUME 2680
3080 IF ERR=70 THEN CLS:PRINT "DISK WRITE PROTECTED":RESUME 2680
3090 IF ERR=64 THEN CLS:PRINT "BAD FILE NAME":RESUME ERL
3100 PRINT "ERROR #";ERR;" IN LINE";ERL
3110 END

```

-- TRUENP.TXT

```

10 DATA -1.203,-1.000,-0.798,-0.597,-0.397,-
0.198,0.000,.197,.392,.585,.777,.968,1.156,1.342,1.527,1.709,1.889,2.067,2.243,2
.416,2.586,2.754,2.920,3.082,3.242,3.399

```

```

20 DATA
3.553,3.704,3.852,3.997,4.138,4.276,4.410,4.541,4.669,4.792,4.912,5.029,5.141,5.
249,5.354,5.454,5.550,5.642,5.730,5.813
30 DATA
5.891,5.965,6.035,6.009,6.158,6.213,6.262,6.306,6.344,6.377,6.404,6.425,6.441,6.
452,6.458
50 DIM WX(61),EC(61)
60 FOR YG=1 TO 61:READ A:WX(YG)=A*1.049999:EC(YG)=3.031600E+02-(YG-1)*5:NEXT YG
70 CLS:COLOR 14,0,0:ON ERROR GOTO 1020
80 ON ERROR GOTO 1020
90 DIM DT(6,5,200),X(5,200),Y(5,200)' MAXIMUM OF FOUR CHANNELS.
100 CLS
110 INPUT "CURRENT: ",CUR
120 INPUT "INSERT DATA DISK IN DRIVE A: AND HIT <RETURN>",&ANS$:PRINT "DIRECTORY
OF DISK A:":FILES "A:":INPUT "Enter name of data file (INCLUDING DRIVE, PATH,
AND SUFFIX): ",FFS$:OPEN FFS$ FOR INPUT AS #3
130 INPUT "Enter new data file name (INCLUDING DRIVE, PATH, AND SUFFIX):
",HYT$:IF HYT$<>FFS$ THEN OPEN HYT$ FOR OUTPUT AS #2:GOTO 150
140 PRINT "NEW FILE MUST HAVE A DIFFERENT NAME":GOTO 130
150 INPUT #3,TITLE$
160 INPUT #3,PROG$
170 INPUT #3,VAXL$:INPUT #3,HAXL$
180 INPUT #3,CF%,NG%
190 INPUT #3,TYP$
200 INPUT #3,MS%,LO%
210 INPUT #3,XMN,XXM,XST
220 FOR J%=1 TO NG%
230   INPUT #3,NIT%(J%),DOT%(J%),O%(J%),SSR%(J%),SSC%(J%)
240   INPUT #3,SUB$(J%)
250   INPUT #3,MN(J%),MX(J%),ST(J%)
260 IF J%<>1 THEN 320
270   LINE INPUT #3,TST$
280   NC%=0:PNT%=1:SL%=0
290   FOR P%=1 TO LEN(TST$):SL%=SL%+1:IF MID$(TST$,P%,1)=", " THEN
DT(NC%,1,1)=VAL(MID$(TST$,PNT%,SL%)):NC%=NC%+1:PNT%=P%+1:SL%=0
300   NEXT P%
310   DT(NC%,1,1)=VAL(MID$(TST$,PNT%,SL%))
320   FOR I%=1-(J%=1) TO NIT%(J%)
330     FOR K%=0 TO NC%
340       INPUT #3,DT(K%,J%,I%)
350     NEXT K%
360   NEXT I%
370 NEXT J%
380 REM MANIPULATE DATA
390 FIN%=-1
400 FOR J%=1 TO NG%
410   FOR I%=1 TO NIT%(J%)
420     FOR K%=0 TO NC%:I(K%+1)=DT(K%,J%,I%):NEXT K%

```



```

430     NUM%=(J%-1)*200+I%
440     Y(J%,I%)=I(3)/CUR' Z IS A DUMMY VARIABLE
450 SE%=INT((I(1)+1.200000)*4.900000+1)-(SE%=0)
460 IF I(1)<WX(1) THEN GOTO 520
470 FOR CC%=SE% TO 61
480 IF I(1)>WX(CC%) THEN GOTO 500
481 CF=(EC(CC%-1)^2-EC(CC%)^2)/(WX(CC%-1)-WX(CC%)):DF=CF*WX(CC%-1)-EC(CC%-1)^2
482 X(J%,I%)=SQR(CF*I(1)-DF):CC%=61
500 NEXT CC%
510 PRINT X(J%,I%),Y(J%,I%)
520     IF FIN% THEN XMAX=X(J%,I%):XMIN=X(J%,I%):YMAX=Y(J%,I%):YMIN=Y(J%,I%):
FIN%=0:GOTO 550
530     IF X(J%,I%)<XMIN THEN XMIN=X(J%,I%)
ELSE IF X(J%,I%)>XMAX THEN XMAX=X(J%,I%)
540     IF Y(J%,I%)<YMIN THEN YMIN=Y(J%,I%)
ELSE IF Y(J%,I%)>YMAX THEN YMAX=Y(J%,I%)
550 NEXT I%
560 NEXT J%
570 IX=(XMAX-XMIN)/10:M=10^INT(LOG(IX)/LOG(10))
580 FD=INT(IX/M+5.000000E-01)
590 FD=VAL(MID$(" 1 2 5 5 51010101010",FD*2-1,2))
600 IXF=FD*M
610 T%=INT(LOG(ABS(IXF))/LOG(10)):IXF$=STR$(INT(IXF*10^(-T%)+1.000000E-01))+ "E"+
MID$("+-",1-(T%<0),1)+RIGHT$(STR$(T%),LEN(STR$(T%))-1)
620 XMIN=INT(XMIN/IXF)*IXF
630 IF XMIN=0 THEN XMIN$="0" ELSE T%=INT(LOG(ABS(XMIN))/LOG(10))-4:
XMIN$=STR$(INT(XMIN*10^(-T%)+1.000000E-01))+ "E"+MID$("+-",1-(T%<0),1)+
RIGHT$(STR$(T%),LEN(STR$(T%))-1)
640 XMAX=INT(XMAX/IXF+1)*IXF
650 IF XMAX=0 THEN XMAX$="0" ELSE T%=INT(LOG(ABS(XMAX))/LOG(10))-4:
XMAX$=STR$(INT(XMAX*10^(-T%)+1.000000E-01))+ "E"+MID$("+-",1-(T%<0),1)+
RIGHT$(STR$(T%),LEN(STR$(T%))-1)
660 IY=(YMAX-YMIN)/10:M=10^INT(LOG(IY)/LOG(10))
670 FD=INT(IY/M+5.000000E-01)
680 FD=VAL(MID$(" 1 2 5 5 51010101010",FD*2-1,2))
690 IYF=FD*M
700 T%=INT(LOG(ABS(IYF))/LOG(10)):IYF$=STR$(INT(IYF*10^(-T%)+1.000000E-01))+ "E"+
MID$("+-",1-(T%<0),1)+RIGHT$(STR$(T%),LEN(STR$(T%))-1)
710 YMIN=INT(YMIN/IYF)*IYF
720 IF YMIN=0 THEN YMIN$="0" ELSE T%=INT(LOG(ABS(YMIN))/LOG(10))-4:
YMIN$=STR$(INT(YMIN*10^(-T%)+1.000000E-01))+ "E"+MID$("+-",1-(T%<0),1)+
RIGHT$(STR$(T%),LEN(STR$(T%))-1)
730 YMAX=INT(YMAX/IYF+1)*IYF
740 IF YMAX=0 THEN YMAX$="0" ELSE T%=INT(LOG(ABS(YMAX))/LOG(10))-4:
YMAX$=STR$(INT(YMAX*10^(-T%)+1.000000E-01))+ "E"+MID$("+-",1-(T%<0),1)+
RIGHT$(STR$(T%),LEN(STR$(T%))-1)
750 FOR TYR=1 TO 20:BEEP:NEXT TYR
760 PRINT #2,TITLE$

```

```

770 PRINT #2,PROG$
780 PRINT #2,VAXL$:PRINT #2,HAXL$
790 PRINT #2,CF%;" , ";NG%
800 PRINT #2,TYP$
810 PRINT #2,MS%;" , ";LO%
820 PRINT #2,XMIN%;" , ";XMAX%;" , ";IXF$
830 FOR J%=1 TO NG%
840     PRINT #2,NIT%(J%);" , ";DOT%(J%);" , ";O%(J%);" , ";SSR%(J%);" , ";SSC%(J%)
850     PRINT #2,SUB$(J%)
860     PRINT #2,YMIN%;" , ";YMAX%;" , ";IYF$
870     FOR I%=1 TO NIT%(J%)
880         PRINT #2,X(J%,I%);" , ";Y(J%,I%)
890     NEXT I%
900 NEXT J%
910 PRINT #2,""
920 FOR I%=1 TO 10
930     PRINT #2,"0"
940 NEXT I%
950 CLOSE #2,#3
960 INPUT "DO YOU WANT TO MANIPULATE ANOTHER FILE (Y/N) ";ANS$
970 IF (ANS$="Y" OR ANS$="y") THEN RUN
980 INPUT "DO YOU WANT TO RUN ANOTHER TEST (Y/N) ";ANS$
990 IF (ANS$="Y" OR ANS$="y") THEN RUN "C:\DAS\1-GDA6.BAS"
1000 INPUT "HIT <RETURN> TO RETURN TO DOS MENU",ANS$
1010 SYSTEM
1020 IF ERR=61 THEN PRINT "DISK FULL":RESUME
1030 IF ERR=71 THEN PRINT "DISK NOT READY":RESUME
1040 IF ERR=70 THEN PRINT "DISK WRITE PROTECTED":RESUME
1050 IF ERR=53 THEN PRINT "NO FILES":RESUME
1060 IF ERR=64 THEN PRINT "BAD FILE NAME":RESUME
1070 IF ERR=76 THEN PRINT "PATH NOT FOUND":RESUME
1080 PRINT "ERROR #";ERR;" IN LINE";ERL
1090 INPUT "HIT <RETURN> TO RETURN TO DOS MENU",ANS$
1100 SYSTEM

```

□