Appendix K

HP-Basic ADC_FOURIER Code

```
      !*********************************
      !  ADC CARD SPECTRUM ANALYSIS PROGRAM
      Array_size=8192
      ALLOCATE Array(Array_size)
      MAT Array=(0)
      PEN 1
      CALL Get_data(Array(*),Array_size)
      Isign=1
      CALL Fourier(Array(*),Array_size,Isign)
       CALL Plot_transform(Array(*),Array_size,1,120,10,50)
      END
      !*********************************************************************
      SUB Fourier(Dat(*),Array_size,Isign)
      FOR K=1 TO 16
         IF Array_size=2^K THEN GOTO Size_ok
      NEXT K
         !
       ! IF Array_size<2^K THEN
       !      Length=2^K
       !      ALLOCATE Dat(2*Length)
       !      N=2*Length
         OUTPUT 1;"SIZE NOT RIGHT.  PROGRAM PAUSED."
         PAUSE
         STOP
Size_ok:            ! SIZE OF ARRAY IS 2^N
         N=Array_size
       ! ALLOCATE Dat(N)
         J=1
         FOR I=1 TO N STEP 2
              IF J>I THEN
                   Tempr=Dat(J)
                   Tempi=Dat(J+1)
                   Dat(J)=Dat(I)
                   Dat(J+1)=Dat(I+1)
                   Dat(I)=Tempr
                   Dat(I+1)=Tempi
              END IF
              M=N/2
              IF M>=2 AND J>M THEN
                   J=J-M
                   M=M/2
                   GOTO 490
              END IF
              J=J+M
         NEXT I
         Mmax=2
         IF N>Mmax THEN
              Istep=2*Mmax
              Theta=2*PI/(Isign*Mmax)
              Wpr=-2*(SIN(.5*Theta))^2
              Wpi=SIN(Theta)
              Wr=1.0
              Wi=0.
              FOR M=1 TO Mmax STEP 2
                   FOR I=M TO N STEP Istep
                        J=I+Mmax
                        Tempr=Wr*Dat(J)-Wi*Dat(J+1)
                        Tempi=Wr*Dat(J+1)+Wi*Dat(J)
                        Dat(J)=Dat(I)-Tempr
                        Dat(J+1)=Dat(I+1)-Tempi
                        Dat(I)=Dat(I)+Tempr
                        Dat(I+1)=Dat(I+1)+Tempi
                   NEXT I
                   Wtemp=Wr
                   Wr=Wr*Wpr-Wi*Wpi+Wr
                   Wi=Wi*Wpr+Wtemp*Wpi+Wi
              NEXT M
              Mmax=Istep
              GOTO 570
         END IF
         SUBEND
       !*************************************************************
       SUB Get_data(A(*),Array_size)
       PRINT
       PRINT
```

```
       PRINT
       PRINT
       PRINT
       PRINT
       PRINT
       INTEGER Rate,I
       PRINT "ADC Card Communication/Spectrum Analysis Program"
       PRINT "Please enter the sampling rate you desire."
User_input:              ! get user input
       INPUT "0 (10 kHz) or 1 (100 kHz)",Rate
       IF Rate<>0 AND Rate<>1 THEN GOTO User_input
       OUTPUT 728 USING "#,B";Rate
       INTEGER Adc(8191)
       WAIT 1
       ENTER 728 USING "8192(#,W)";Adc(*)
       FOR I=0 TO 8191
       A(I)=Adc(I)
       NEXT I
       A(8192)=0
       CLEAR SCREEN
           CALL Plot_data(A(*),Array_size,0,120,60,90)
       SUBEND
       !****************************************************************
       SUB Plot_data(A(*),Array_size,V1,V2,V3,V4)
            VIEWPORT V1,V2,V3,V4
         !  WINDOW 0,Array_size/2,-1,20
            Maximum=MAX(MAX(A(*)),ABS(MIN(A(*))))
            WINDOW 0,Array_size,-Maximum,Maximum
            FRAME
            MOVE 1,A(I)
            FOR I=1 TO Array_size STEP 2
                 PLOT I,A(I)
            NEXT I
            PENUP
       SUBEND
       !****************************************************************
       SUB Plot_transform(A(*),Array_size,V1,V2,V3,V4)
            VIEWPORT V1,V2,V3,V4
            Maximum=MAX(A(*))
        !   WINDOW 0,(Array_size/2)/8,-Maximum,Maximum
            Bottom=0
            WINDOW 0,(Array_size/2)/1,Bottom,LGT(MAX(ABS(MIN(A(*))),ABS(MAX(A(*)))))
         !  FRAME
            MOVE 1,A(1)
            FOR I=1 TO Array_size/2+1 STEP 2
                  IF ABS(A(I))=0 THEN
                       PLOT I,Bottom
                     ELSE
                       IF LGT(ABS(A(I)))>Bottom THEN PLOT I,LGT((ABS(A(I))))
          !            IF LGT(ABS(A(I)))>Bottom THEN PRINT I,A(I)
                       IF LGT(ABS(A(I)))<=Bottom THEN PLOT I,Bottom
                  END IF
            NEXT I
            PENUP
            J=1  ! FIRST POINT SHOULD FALL ON 3 (1/(N*DELTA))
            MOVE J,A(Array_size-1)
            FOR I=Array_size-1 TO Array_size/2 STEP -2
                 J=J+2
                 IF ABS(A(I))=0 THEN
                      PLOT J,Bottom
                    ELSE
                       IF LGT(ABS(A(I)))>Bottom THEN PLOT J,LGT((ABS(A(I))))
          !            IF LGT(ABS(A(I)))>Bottom THEN PRINT J,A(I)
                       IF LGT(ABS(A(I)))<=Bottom THEN PLOT J,Bottom
                  END IF
            NEXT I
            PENUP
            ! NOW PLOT THE COMPLEX PART
            MOVE 2,A(2)
            FOR I=2 TO Array_size/2+1 STEP 2
                  IF ABS(A(I))=0 THEN
                       PLOT I,Bottom
                     ELSE
                       IF LGT(ABS(A(I)))>Bottom THEN PLOT I,LGT((ABS(A(I))))
           !           IF LGT(ABS(A(I)))>Bottom THEN PRINT I,A(I)
                       IF LGT(ABS(A(I)))<=Bottom THEN PLOT I,Bottom
                  END IF
            NEXT I
            PENUP
            J=2  ! FIRST POINT SHOULD FALL ON 3 (1/(N*DELTA))
            MOVE J,A(Array_size)
```

```
     FOR I=Array_size TO Array_size/2 STEP -2
          J=J+2
          IF ABS(A(I))=0 THEN
               PLOT J,Bottom
              ELSE
                IF LGT(ABS(A(I)))>Bottom THEN PLOT J,LGT((ABS(A(I))))
  !             IF LGT(ABS(A(I)))>Bottom THEN PRINT J,A(I)
                IF LGT(ABS(A(I)))<=Bottom THEN PLOT J,Bottom
           END IF
     NEXT I
     PENUP
     SUBEXIT  ! TREMOVE THIS
     MOVE 2,A(2)
     FOR I=2 TO Array_size/2 STEP 2
          PLOT I-1,A(I)
     NEXT I
     PENUP
SUBEND
     !************************************************************
```