7854 APPLICATION PROGRAMS

7854

Written and compiled by
Mike Mraz
Gary Kirchberger

July 28, 1981

Programs also contributed by
Clark Foley
Roger Loop

Tektronix®
COMMITTED TO EXCELLENCE

INTRODUCTION

This software is provided for use in demonstrating the power of the 7854 oscilloscope. The programs range from simple to somewhat complex, although an effort was made to make each easy to use.

We encourage you to investigate each program. While some programs may appear on the surface to have no applicability to a particular user's requirements, this book still has value. Each program demonstrates various and different techniques to make the 7854 perform. Tasks that may appear to be difficult or impossible at first glance are often handled quite simply. The programming techniques are the real value.

We also encourage you to forward to us any 7854 programs you may have. They will be considered for publication in future volumes. Please include some documentation. Programs may be sent to:

Lab Scopes Marketing
Tektronix, Inc.
P.O. Box 500
D/S 39-327
Beaverton, Oregon 97077

This software is not for sale nor is it supportable.

TABLE OF CONTENTS

7854 Sales Aid Reference Guide

1. AMPLITUDE COMPARISON IN dB

I. PROGRAM LISTING

```
000 STORED OFF CLS NEXT
001 STOP 1 0 AVG . 0 5 SMOOTH P-P NEXT
002 STOP 1 0 AVG . 0 5 SMOOTH NEXT
003 X<>Y P-P X<>Y / NEXT
004 LN 1 0 LN / 2 0 * NEXT
005 1 GOTO NEXT
```

II. DESCRIPTION

This program calculates the difference, in dB, between an input and an output waveform. The program is arranged so that only one probe is used to connect the signal to the 7854, thus minimizing probe-to-probe errors. After acquiring the input voltage, the program stops; after repositioning the probe, the program is restarted by again pressing RUN. At end of execution, magnitude difference (in dB) is displayed in the X register.

III. ALGORITHM or PROCEDURE

To use this program, first press START (f RUN). This initializes 7854 cursors, registers, etc. Then, connect the scope probe to amplifier-under-test input, and press RUN. Move the probe to amplifier output and press RUN again. Gain is displayed in the X register, in dB. To measure the next amplifier, simply press RUN after connecting probe to input.

IV. INPUT: Waveforms (both input and output) are acquired into Ø WFM.

OUTPUT: X register = $20 \log \left(\frac{V_{in}}{V_{out}}\right)$

V. RESTRICTIONS

None

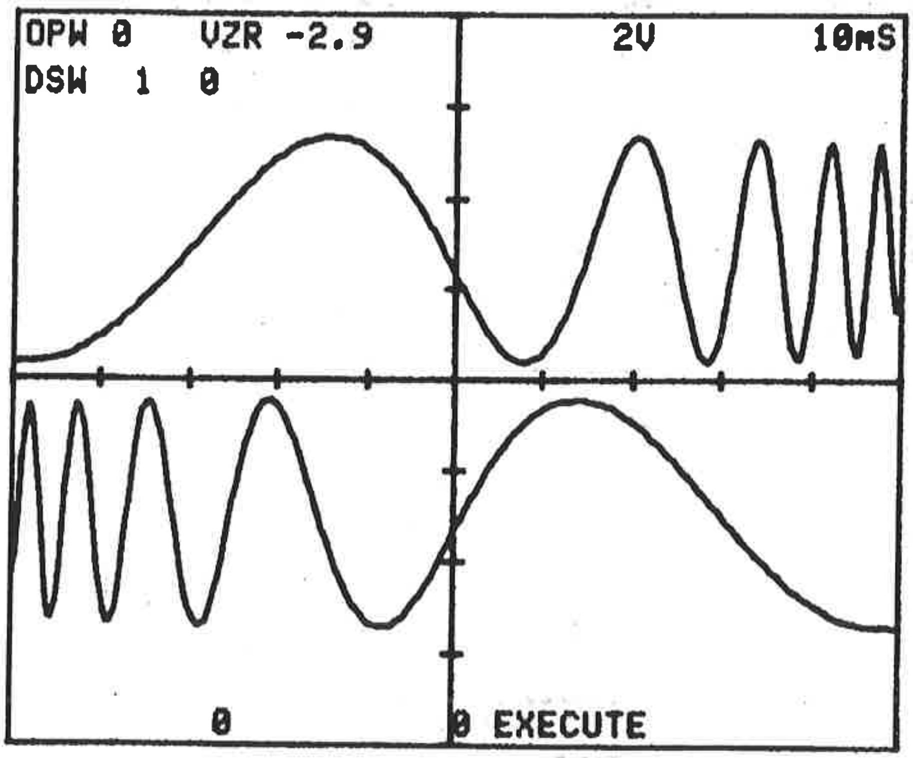
VI. COMMENTS

000 Entry point-initialize
001 Wait for user to press RUN; acquire V_{in}; find P-P
002 Wait for user to press RUN; acquire V_{out}
003 Divide V_{in} by V_{out}
004 Convert to dB
005 Wait for next user input

2. REVERSE WAVEFORM IN TIME

I. PROGRAM LISTING

```
000 OFF CRS1 CLS >HCRD NEXT
001 P/W 2 0 >CNS NEXT
002 LNN 0 1 1 WFM VCRD 1 0 >CNS NEXT
003 CRS1> 0 WFM NEXT
004 1 0 CNS 2 0 CNS 1 - 2 0 >CNS >PNT NEXT
005 2 0 CNS 0 IFX=Y STOP NEXT
006 0 1 LBL GOTO NEXT
```



II. DESCRIPTION

Occasionally, it may be desirable to reverse a waveform in time for display or calculation. This program may be used to reverse a waveform with respect to time.

III. ALGORITHM or PROCEDURE

The original waveform, stored in 1 WFM, is transferred in reverse order to Ø WFM.

IV. INPUT: Original waveform = 1 WFM.

OUTPUT: Reversed waveform = Ø WFM.

V. RESTRICTIONS

None

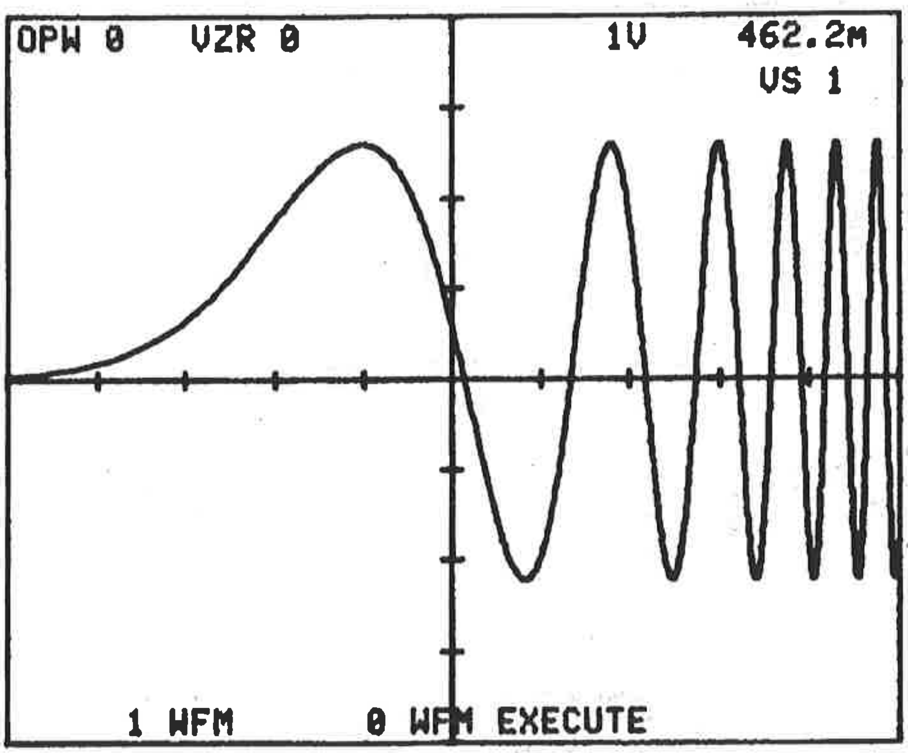
VI. COMMENTS

000 Initialize
001 Store P/W in 20 CNS (counter for Ø WFM points)
002 (Label 1) Retrieve VCRD, store in 10 CNS
003 Move cursor one point to right
004 Create point on Ø WFM; decrement 20 CNS
005 Test 20 CNS for last point
006 Go to Label 1

3. LOGARITHMIC SWEEP DISPLAY

I. PROGRAM LISTING

```
000 OFF 0 WFM ENTER - 1 + NEXT
001 INTG MAX 1 0 0 / + NEXT
002 LN MID - 1 0 P-P VSCL / / VXPD NEXT
003 1 >WFM NEXT
004 AQR . 2 5 SMOOTH 1 VS STOP NEXT
```



II. DESCRIPTION

This program displays a waveform with respect to log-time, as in a log-frequency plot. The program generates a waveform which is the logarithm of a linear ramp, then displays the input waveform versus the log ramp. Although the log ramp is not calibrated, actual time information can be obtained using the cursors and the HCRD command.

III. ALGORITHM or PROCEDURE

Using the INTG command, a linear ramp is generated in Ø WFM, then LN converts it to a logarithmic ramp which is scaled to give a full 10-division logarithmic ramp.

IV. INPUT: Waveform to be displayed is acquired into Ø WFM.

OUTPUT: Log-sweep waveform displayed on screen.

V. RESTRICTIONS

(1) AQR restrictions

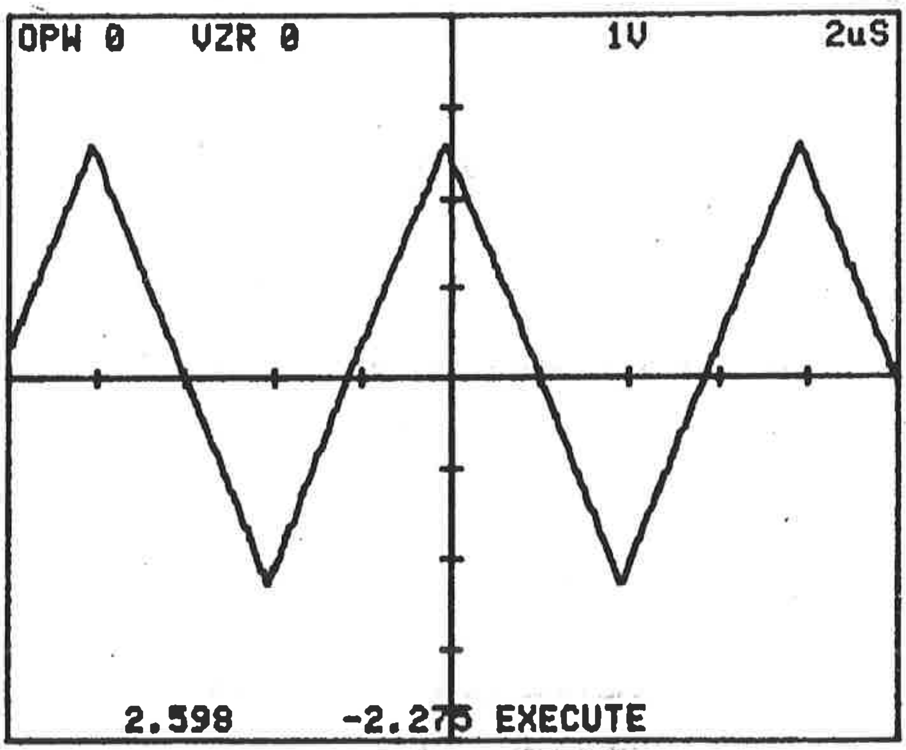
VI. COMMENTS

000 Clear Ø WFM; add 1 to Ø WFM
001 Integrate to form ramp; add $\frac{MID}{100}$ to avoid ln (Ø)
002 Take LN; scale to create 10-division log ramp
003 Store log ramp in 1 WFM
004 AQR input; display vs. 1 WFM (log ramp)

4. RUNNING MAXIMA AND MINIMA

I. PROGRAM LISTING

```
000 CLS OFF 1 >CNS 2 >CNS 3 >CNS NEXT
001 STORED NEXT
002 AQR MAX 1 CNS NEXT
003 IFY>X 1 LBL GOTO NEXT
004 LNN 0 2 MIN 2 CNS X<>Y NEXT
005 IFY>X 3 LBL GOTO NEXT
006 LNN 0 4 3 CNS 1 + 3 >CNS 1 GOTO NEXT
007 LNN 0 1 X<>Y 1 >CNS 2 LBL GOTO NEXT
008 LNN 0 3 2 >CNS 4 LBL GOTO NEXT
```



II. DESCRIPTION

This program continually acquires a signal, finding the minimum and maximum value and updating max and min stored in 1 CNS and 2 CNS. In other words, the program keeps a running maximum and minimum of the input waveform. The program will loop indefinitely until stopped manually.

APPLICATIONS: (1) Unattended monitoring of a signal.

III. ALGORITHM or PROCEDURE

MAX and MIN commands are performed on Ø WFM after acquisition. If new MAX > old MAX, new MAX is stored; if new MIN < old MIN, new MIN is stored. When program is stopped, number of acquisitions is stored in 3 CNS.

IV. INPUTS: Signal acquired into Ø WFM.

OUTPUTS:

MAX	1 CNS
MIN	2 CNS
COUNTER	3 CNS

V. RESTRICTIONS

(1) AQR restrictions

(2) 3 CNS overflows after 65536

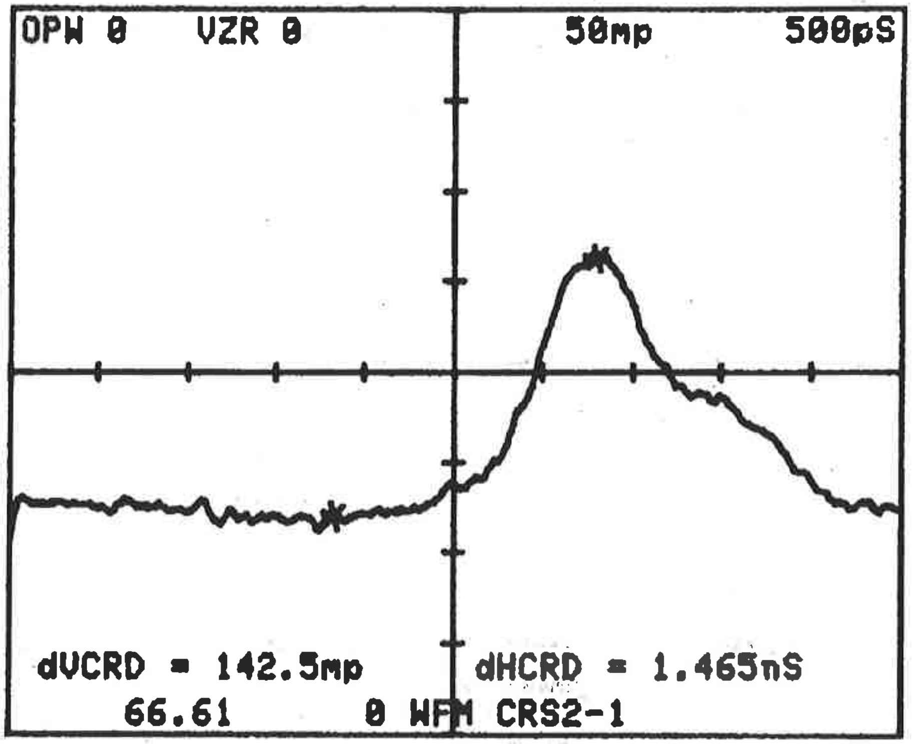
V. COMMENTS

000 Initialize
001 Initialize
002 Acquire; find MAX; recall old MAX (1 CNS)
003 Test new MAX
004 Find MIN; recall old MIN (2 CNS)
005 Test new MIN
006 Increment counter (3 CNS)
007 Replace MAX if new MAX > old MAX
008 Replace MIN if new MIN < old MIN

5. CHARACTERISTIC IMPEDANCE (Zo) FROM TDR DISPLAY

I. PROGRAM LISTING

```
000 STOP VCRD 1 + NEXT
001 VCRD 1 X<>Y - NEXT
002 / 5 0 * 0 GOTO NEXT
```



II. DESCRIPTION

This simple program calculates the characteristic impedance, Z , of a transmission line from its reflection coefficient (p). Reflection coefficient is determined by using a 7S12 Time Domain Reflectometer (TDR)plug-in with the 7854. Cursor 1 defines the reference impedance (50Ω) and cursor 2, the unknown impedance.

III. ALGORITHM or PROCEDURE

Cursors 1 and 2 are manually positioned to the desired point on the TDR display. When RUN is pressed, the program calculates Z_o using the formula:

$$Z = Z \frac{1 + \Delta p}{1 - \Delta p},$$

where Z = 50Ω

Δp = change in reflection coefficient, given by ΔVCRD.

IV. INPUTS: TDR display in Ø WFM

CRS 1 and CRS 2 positioned manually

OUTPUTS: Z_o in X-register

V. RESTRICTIONS

Ø WFM is a TDR waveform, with vertical values representing reflection coefficient (p).

VI. COMMENTS

000 Add 1 to Δp
001 Subtract 1 from Δp
002 Divide and normalize to 50Ω; restart

6. AREA OF X-Y DISPLAY

I. PROGRAM LISTING

```
000 TIME OFF STORED NEXT
001 VMDL 1 0 AVG 2 >WFM NEXT
002 VMDR 1 0 AVG 1 >WFM NEXT
003 VMDALT NEXT
004 MEAN PER NEXT
005 2 * NEXT
006 HSCL 1 0 * X<>Y / 1 >CNS NEXT
007 1 WFM 1 CNS HXPD 1 >WFM NEXT
008 2 WFM 1 CNS HXPD 2 >WFM NEXT
009 2 WFM 1 VS NEXT
010 1 WFM DIFF 2 WFM * AREA NEXT
011 ABS NEXT
012 2 / NEXT
013 2 WFM CLX NEXT
014 STOP 0 GOTO NEXT
```

II. DESCRIPTION

This program computes the area of an X-Y display from two time-related waveforms. Since the standard AREA command calculates only the area under the operational waveform (OPW), a program such as this one is necessary to compute area of X-Y displays.

III. ALGORITHM or PROCEDURE

To reduce potentially large errors, the program uses two cycles of the input waveforms for calculation. The mathematical basis for X-Y area is:

$$\text{Area} = \int_{V_1}^{V_2} P\,dv = \int_{t_1}^{t_2} \left(P\,\frac{dv}{dt}\right) dt$$

Then, letting $\frac{dv}{dt}$ = DIFF (1 WFM)

and $\left(P\frac{dv}{dt}\right) = \left[\text{DIFF (1 WFM)}\right] \times \text{(2 WFM)}$,

simply finding the area of this quantity will perform the AREA integral above.

IV. INPUTS: 1 WFM - X (independent) variable

2 WFM = Y (dependent) variable

OUTPUTS: XY waveform on screen

X register = area

V. RESTRICTIONS

At least two cycles of input waveform must be acquired.

VI. COMMENTS

000 Initialize
001, 002 } AVG left and right channels; store
003 Display realtime in ATL mode
004 Compute period of input
005 Select two periods for calculation
006, 007, 008 } Horizontally expand over two cycles
009 Display X vs. Y
010 Compute area
011 Take absolute value to adjust for polarity
012 Normalize to one cycle of input
013 Push 2 WFM to retain 2 WFM vs. 1 WFM
014 Stop; return to beginning of program

7. TIME JITTER MEASUREMENT

I. PROGRAM LISTING

```
000 OFF STORED 1 0 2 4 >P/W NEXT
001 AQR NEXT
002 2 >WFM NEXT
003 MID - SGN NEXT
004 DIFF ABS NEXT
005 SGN 1 >WFM NEXT
006 CRS1 0 >HCRD NEXT
007 1 >VCRD NEXT
008 0 WFM INTG NEXT
009 MAX VCRD - CRS2-1 >VCRD NEXT
010 HCRD NEXT
011 STOP 0 GOTO NEXT
```

II. DESCRIPTION

This program measures the amount of time jitter contained in a positive or negative transition (step).

III. ALGORITHM or PROCEDURE

After acquiring the input waveform, the SGN, DIFF, and INTG functions are used to generate a waveform which represents each transition in the jitter region as an increment in voltage. The cursors are then positioned to convert the total voltage to time, giving a measure of total jitter.

IV. INPUT: Waveform acquired into Ø WFM

OUTPUTS: X register = time jitter value

Ø WFM = integral of 1 WFM

1 WFM = waveform representing jitter by 0 to +1 transitions

2 WFM = input step

V. RESTRICTIONS

(1) Only one transition may be acquired at a time.

(2) AQR restrictions

(3) Option 2D required for 1024 P/W. Lower resolution (e.g., 512 P/W) reduces accuracy

VI. COMMENTS

000 Initialize
001 Acquire step
002 Store input waveform in 2 WFM
003 Convert input to ±1V equivalent waveform with SGN
004 Differentiate to locate transitions
005 Convert to 1V equivalent (SGN); store in 1 WFM
006 Initialize CRS 1
007 Position CRS 1 to beginning of jitter region
008 Integrate 1 WFM to create voltage step
009 Position CRS 2 to end of jitter region
010 Convert to time
011 Stop; restart

8. PERIOD MEASUREMENTS OF EIGHT CYCLES OF WAVEFORM

I. PROGRAM LISTING

```
000 1 0 2 4 >P/W TIME STORED NEXT
001 CLS >CNS CRS1 >HCRD CRS2-1 >HCRD NEXT
002 CRS1 AQS . 0 5 SMOOTH NEXT
003 CRS1> CRS1> CRS1> CRS1> VCRD OFF NEXT
004 PER 0 >CNS 1 >CNS NEXT
005 CRS2-1 >HCRD NEXT
006 2 ENTER 1 0 >CNS NEXT
007 LNN 0 0 0 >VCRD >VCRD NEXT
008 HCRD 0 CNS - 1 0 CNS >CNS NEXT
009 HCRD 0 >CNS NEXT
010 1 0 CNS 1 + 1 0 >CNS NEXT
011 8 ENTER IFY>X STOP NEXT
012 0 0 LBL GOTO NEXT
```

II. DESCRIPTION

This program was written to measure the period of eight consecutive cycles of an input waveform. Each value is stored in the corresponding constant register, i.e., first period value in 1 CNS, second value in 2 CNS, etc.

APPLICATIONS: Checking frequency stability

Observing frequency-shift keying (FSK)

III. ALGORITHM or PROCEDURE

Zero-crossings and cursors are used to find period of eight consecutive cycles of a waveform. The program takes advantage of the fact that the cursor can search for a given voltage level by repeatedly executing the >VCRD command.

IV. INPUTS: Waveform acquired into Ø WFM by AQS (may also use AQR or AVG by changing program line 2).

OUTPUTS: 1 CNS = period of first cycle

2 CNS = period of second cycle

. .
. .
. .

8 CNS = period of eight cycle

V. RESTRICTIONS

(1) If AQS is used, the AQS sweep speed restrictions must be observed.

(2) At least eight cycles of the input waveform must appear on-screen; otherwise, cursors will recycle to beginning of waveform and repeat measurements.

VI. COMMENTS

000 Set P/W, initialize
001 Zero-initialize register and cursors
002 AQS and SMOOTH waveform
003 Find Ø-crossing level for period calculation
004 Find first period-store in Ø and 1 CNS
005 Position CRS 2-1 to first period
006 Set counter for loop (1- CNS)
007 Search for next Ø-crossing with same slope
008 Find new HCRD; subtract Ø CNS to get next period; store result in next register (from 10 CNS)
009 Store new HCRD in Ø CNS
010 Increment counter
011 Test counter; stop after 8th period
012 End of loop

9. RISETIME WITH CURSOR POSITIONING

I. PROGRAM LISTING

```
000 STORED 4 AVG NEXT
001 CRS1 0 >HCRD NEXT
002 MID >VCRD NEXT
003 HSCL 2 * NEXT
004 HCRD X<>Y - >HCRD NEXT
005 CRS2-1 NEXT
006 HSCL 4 * >HCRD NEXT
007 RISE STOP NEXT
```

10. PULSE WIDTH WITH CURSOR POSITIONING

I. PROGRAM LISTING

```
000 STORED 4 AVG NEXT
001 CRS1 0 >HCRD NEXT
002 MID >VCRD NEXT
003 HSCL 2 * NEXT
004 HCRD X<>Y - >HCRD NEXT
005 CRS2-1 NEXT
006 HSCL 4 * >HCRD NEXT
007 WIDTH STOP NEXT
```

II. DESCRIPTION

Both of these programs are very similar in that they make a pulse measurement by positioning the cursors at 2 divisions before and after the first transition of the input waveform.

III. ALGORITHM or PROCEDURE

The program first positions the cursors to ±2 divisions with respect to the first transition midpoint on the input signal, as determined by the MID >VCRD command sequence. Then, the parameter (either rise-time or pulse width) is measured.

IV. INPUT: Waveform acquired (AVG) into Ø WFM

OUTPUT: X register = measured parameter

V. RESTRICTIONS

(1) First transition must occur to the right of +2 horizontal divisions from left side of screen.

(2) AVG restrictions

VI. COMMENTS (both programs)

000 Acquire waveform (AVG)
001 Initialize CRS 1
002 Locate CRS 1 to midpoint of first transition
003 Find time equivalent of 2 divisions
004 Position CRS 1 to -2 div. from midpoint
005 Turn on CRS 2
006 Position CRS 2 to +2 div. from midpoint
007 Measure risetime (or pulse width); stop

11. RISETIME, 20% TO 80% LEVELS

I. PROGRAM LISTING

```
000 STORED 4 AVG NEXT
001 CRS1 0 >HCRD VCRD 1 >CNS NEXT
002 CRS2-1 HSCL 1 0 * >HCRD NEXT
003 VCRD 2 >CNS NEXT
004 . 2 * 1 CNS + CRS1 >VCRD NEXT
005 2 CNS . 6 * CRS2-1 >VCRD NEXT
006 HCRD STOP NEXT
```

II. DESCRIPTION

This program positions the cursors to the 20% and 80% points on an input step waveform. It is useful for applications which require 20%-80% risetime instead of the 7854's built in 10%-90% RISE command.

III. ALGORITHM or PROCEDURE

The program assumes that the left and right edges of the graticule correspond to 0% and 100% points, respectively, for positive transitions, and 100% and 0% points for negative transitions. Cursors are then positioned accordingly.

IV. INPUT: Waveform is acquired into Ø WFM.

OUTPUT: X register = 20% - 80% time.

V. RESTRICTIONS

(1) Only one positive or negative transition is allowed on screen.

(2) AVG restrictions.

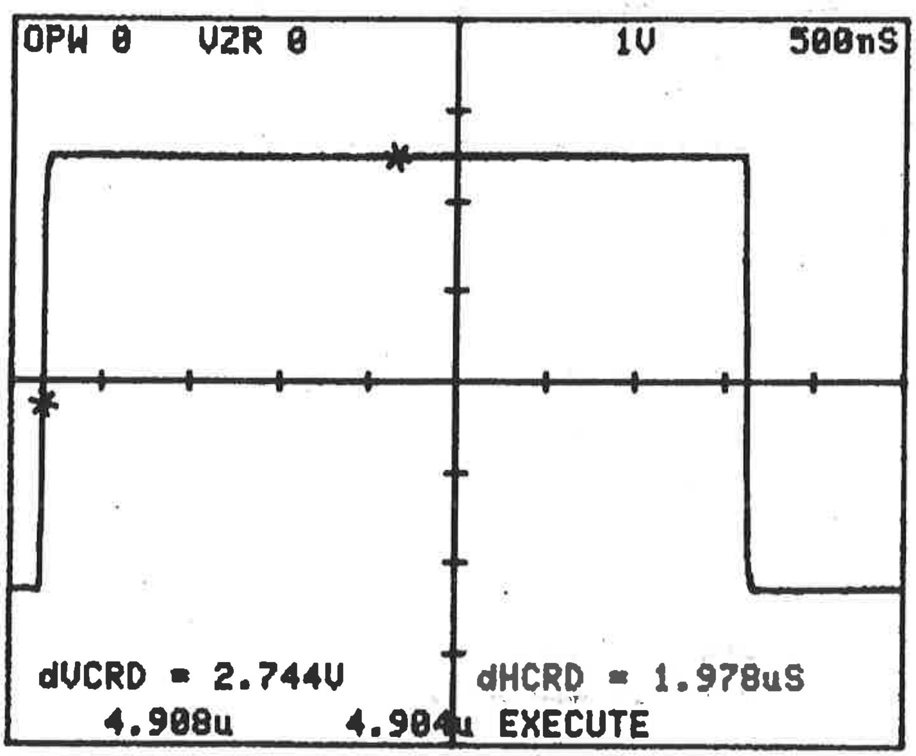
VI. COMMENTS

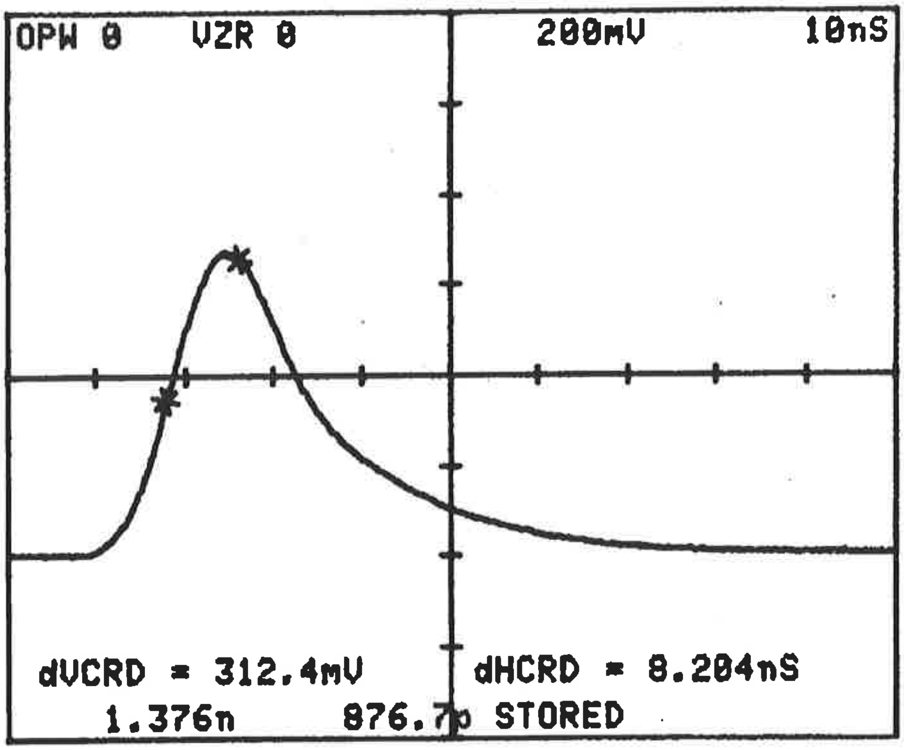
000 Acquire signal (AVG)
001 Find 0% point at left side of screen; store
002 Locate CRS 2 at right side of screen
003 Find 100% point; store
004 Locate CRS 1 at 20% point
005 Locate CRS 2 at 80% point
006 Display time (20%-80%) in X register

12. PULSE SYMMETRY MEASUREMENT

I. PROGRAM LISTING

```
000 4 AVG STORED NEXT
001 CRS1 0 >HCRD NEXT
002 OFF NEXT
003 MID CRS1 >VCRD NEXT
004 CRS2-1 0 >HCRD >VCRD NEXT
005 AREA 1 >CNS NEXT
006 HCRD 2 / >HCRD HCRD - NEXT
007 AREA 2 >CNS 1 CNS X<>Y - 3 >CNS NEXT
008 2 CNS X<>Y NEXT
009 STOP NEXT
```





II. DESCRIPTION

This program measures the area under a pulse, calculating both the area from peak to left 50% point and area from peak to right 50% point.

APPLICATION: Verifying symmetry of a pulse.

III. <u>ALGORITHM or PROCEDURE</u>

The cursors are used to divide the pulse area into left and right portions which are then measured by the AREA command.

IV. INPUT: Waveform is acquired (4 AVG) into Ø WFM

OUTPUTS: Total area - 1 CNS

Left area - 2 CNS

Right area - 3 CNS

V. RESTRICTIONS

(1) AVG restrictions

(2) Only one positive and one negative transition may be displayed.

VI. COMMENTS

000 Acquire waveform
001 Move CRS 1 to left screen
002 CRS 1 off
003 Move CRS 1 to midpoint of first transition
004 Move CRS 2 to midpoint of second transition
005 Find total area of pulse
006 Move CRS 2 to half pulse width
007 Calculate right and left areas
008 Put measurements in proper sequence
009 Stop

NOTE: At program STOP, Y-register = left area, X-register = right area.

13. BUILDING WAVEFORMS

I. PROGRAM LISTING

```
000 OFF BOTH NEXT
001 0 WFM 0 * 1 >VSCL P/W 1 0 / >HSCL NEXT
002 0 WFM 1 >WFM NEXT
003 0 DSW 1 DSW NEXT
004 1 WFM CRS1 NEXT
005 0 >HCRD 1 >CNS NEXT
006 STOP NEXT
007 LNN 0 1 NEXT
008 VZR HCRD 0 WFM CLX >PNT NEXT
009 1 CNS ITRP HCRD 1 >CNS 1 WFM NEXT
010 STOP NEXT
011 CRS1 1 WFM 1 LBL GOTO NEXT
```

II. DESCRIPTION

This program builds a waveform based on the positions of CRS 1 and VZR (vertical position), both of which are set manually. The program can be used to manually build a waveform based on a realtime waveform which is being displayed on screen. Each time the RUN button is pressed, the program draws a vector between the point defined by VZR and CRS 1 and the point at which RUN was previously pressed. See figures 13-1, 2, 3 for further description.

APPLICATIONS: (1) Building "limit" waveforms to compare with an acquired waveform.

(2) Building ideal waveforms to test programs with.

III. <u>ALGORITHM or PROCEDURE</u>

The ITRP (interpolate) command is used to generate the waveform by drawing vectors between two manually selected points.

IV. INPUTS: VZR (vertical position) and CRS 1 are positioned manually.

OUTPUT: Ø WFM contains the built waveform.

V. RESTRICTIONS

None

V. COMMENTS

000 Initialize; display both waveforms
001 Initialize Ø WFM to Ø; set scale factors
002 Set 1 WFM to same as Ø WFM
003 Display both Ø and 1 WFM
004 Place CRS 1 on 1 WFM
005 Set CRS 1 to start of 1 WFM; set 1 CNS = Ø
006 Await manual entry and RUN
007 Label 1
008 Place CRS 1 on proper point of Ø WFM
009 Interpolate from old horiz. value (1 CNS) to new point on Ø WFM
010 Await next manual entry
011 Return to 1 WFM for cursor; go to Label 1

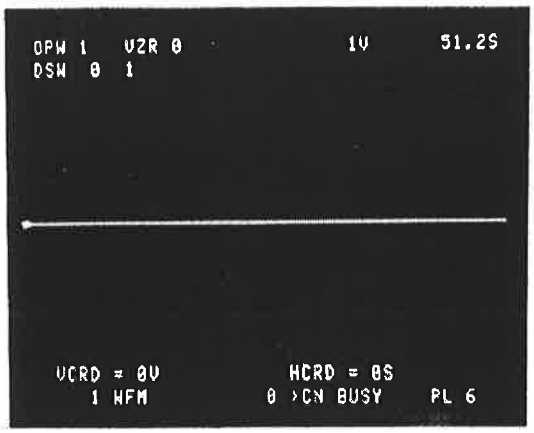


Fig. 13-1 Starting point; Note cursor position and that 1 WFM and Ø WFM are superimposed.

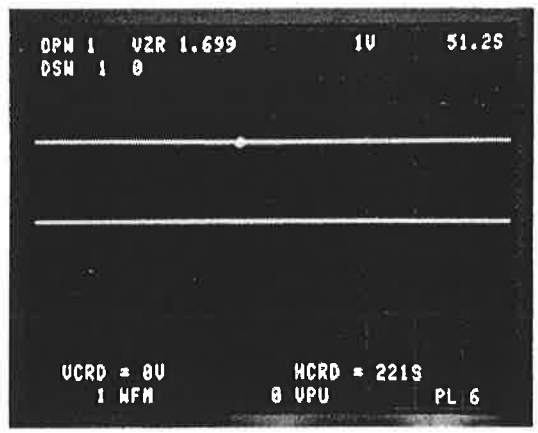


Fig 13-2 1 WFM and cursor have been positioned as shown.

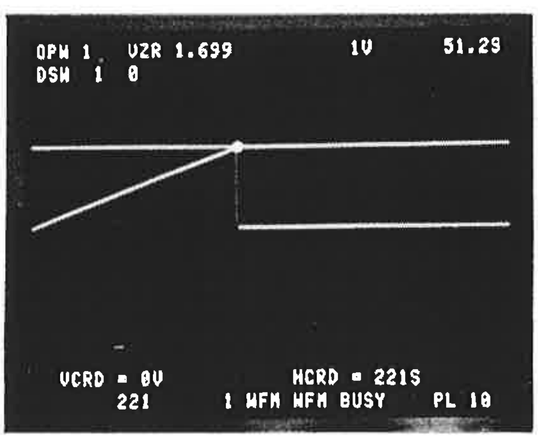


Fig. 13-3 After RUN is pressed, Ø WFM is interpolated between cursor 1 initial and current positions. For the next RUN, the cursor position shown will be the new initial position.

14. AVERAGING WAVEFORMS

I. PROGRAM LISTING

```
000 0 >CNS 1 + 1 >CNS NEXT
001 1 WFM 0 * 1 >WFM NEXT
002 LNN 0 1 NEXT
003 1 CNS 0 CNS - 2 >CNS NEXT
004 1 - 3 >CNS 1 WFM * 1 >WFM NEXT
005 AQR 2 >WFM NEXT
006 1 WFM + 2 CNS / 1 >WFM NEXT
007 0 CNS 1 NEXT
008 IFY>X - 0 >CNS 1 LBL GOTO NEXT
009 0 ENTER 2 CNS NEXT
010 STOP NEXT
```

II. DESCRIPTION

This program enables you to measure and process the actual waveforms that make up an "averaged" waveform. This is not possible by simply using the "AVG" key.

After each waveform is acquired, you have the opportunity to measure or process before the next acquisition. Applications include:

(1) Calculating information about the sample of waveforms making up an averaged waveform.

(2) Averaging processed waveforms rather than processing an averaged waveform.

III. ALGORITHM or PROCEDURE

A running average waveform is maintained. A weighted average is calculated from the previously averaged waveform and the newly acquired waveform.

IV. INPUTS: Enter n, the number of waveforms to be averaged, before pressing "f, START".

OUTPUTS: 1 WFM is the averaged waveform. The X register contains n.

V. RESTRICTIONS

Measurements or processing of the newly acquired waveform (2 WFM) must be accomplished between lines ØØ5 and ØØ6. You must then ensure that the newly acquired wavefcrm (2 WFM) is added to the previous average (1 WFM).

IV. COMMENTS

000 Set Ø CNS to n, 1 CNS to n + 1
001 Initialize 1 WFM to Ø
002 Label 1
003 Set 2 CNS = 1 CNS-Ø CNS, the value of i (i = 1, 2, ..., n)
004 Weight 1 WFM by i-1
005 Acquire new waveform and store as 2 WFM
006 Add new waveform to weighted 1 WFM and divide by i; store as 1 WFM
007 Y = Ø CNS, X = 1
008 Return to Label 1 n-1 times
009 Y = Ø, X = n

15. STANDARD NORMAL DISTRIBUTION AND ONE-SIDED PROBABILITY TEST

I. PROGRAM LISTING

```
000 - X<>Y / X<>Y SQRT * ABS 0 >CNS NEXT
001 1 >VSCL >HSCL NEXT
002 0 WFM 0 * 1 + NEXT
003 INTG 5 - ABS NEXT
004 ENTER * 2 / 1 CHS * EXP NEXT
005 2 . 5 0 7 / 1 >WFM NEXT
006 OFF CRS1 0 >HCRD NEXT
007 CRS2-1 NEXT
008 0 CNS 5 NEXT
009 IFY>X 0 ENTER OFF 1 LBL GOTO NEXT
010 X<>Y - >HCRD NEXT
011 AREA NEXT
012 LNN 0 1 NEXT
013 STOP NEXT
```

II. DESCRIPTION

This program allows you to perform a one-tail probability test for a sample of measurement observations. For example, a right tail test computes the probability that the sample mean would be as large as observed, given the value of the population mean (that you would have expected to observe).

Conversely, a left tail test computes the probability that the sample mean would be as small as observed.

A typical application would be manufacturing quality control where samples are periodically tested against a standard. The computation of the standard normal distribution (lines 001 thru 005) can be lifted out and the remaining program employed as a subroutine. You can also use the distribution to construct other statistical routines.

III. ALGORITHM or PROCEDURE

The Z value for the sample is computed. If Z > Ø, then Z is subtracted from 5. This means the left side of the distribution is always used.

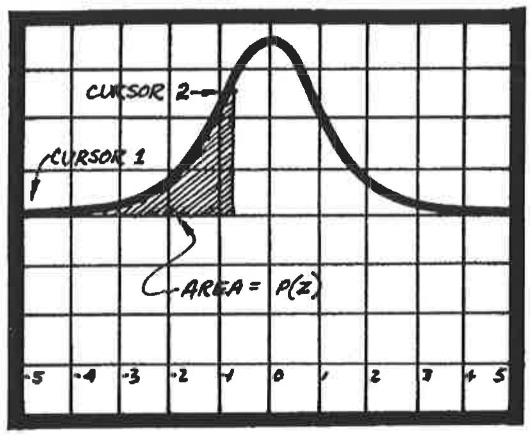


Fig. 15-1

Fig. 15-1: The area under the standard normal curve (1 WFM) to the left of cursor 2 equals the probability. Note the horizontal Z scale in the drawing.

Z is computed by:

$$Z = \frac{\overline{X}-\mu}{\sigma/\sqrt{n}}$$

where: $\overline{X}$ = observed sample mean

μ = population mean (standard or expected mean)

σ = standard deviation of the population

n = number of observations in the sample

The standard normal distribution is defined by:

$$P(Z) = \frac{1}{\sqrt{2\pi}} e^{-(\frac{1}{2})Z^2}$$

IV. INPUTS: Enter n, σ, $\overline{X}$, and μ (in that order) before pressing "f, START"

OUTPUTS: X = the probability P(Z) and the cursor is positioned on the curve at the -Z value. (Unless |Z| > 5)

V. RESTRICTIONS

The program always uses the left side of the distribution, even when Z > Ø.

The standard deviation of the sample (S) may be substituted for the standard deviation of the population (σ), if σ is unknown, but only for large n. Generally, for n ≤ 60, this program should not be used if "S" is used. You should then refer to a table of the "Students t" Distribution.

If n = 1, then the distribution of the population is assumed to be normal.

VI. COMMENTS

```
000  Set Ø CNS to the absolute value of Z               }
001  HSCL, VSCL = 1                                     }
002  Set Ø WFM = 1                                      }  Computes Standard
003  Set Ø WFM = |5-t|                                  }  Normal Distribution, P(Z)
004  Set Ø WFM = e^(-(½)(t-5)^2)                        }
005  Set 1 WFM = P(t-5)=(2.507)·(Ø WFM)                 }
006  Turns on cursor 1 at Z= -5 (actually t=Ø)          }
007  Turns on cursor 2
008  Y = |Z|, X = 5
009  Go to Label 1 if |Z| > 5, after setting X = Ø
010  Set cursor 2 at -|Z| (actually t=5-|Z|)
011  X = AREA = P(Z)
012  Label 1
```

16. STORING DATA AS A WAVEFORM AND STANDARD DEVIATION

I. PROGRAM LISTING

```
000 P/W NEXT
001 IFY>X X<>Y NEXT
002 CLX NEXT
003 0 >CNS 1 >CNS 1 - 2 >CNS NEXT
004 0 WFM 0 * 1 >WFM NEXT
005 CRS1 0 >HCRD OFF NEXT
006 LNN 0 1 NEXT
007 AQR P-P NEXT
008 1 WFM X<>Y 0 CNS 1 - >PNT NEXT
009 0 CNS 1 NEXT
010 IFY>X - 0 >CNS 1 LBL GOTO NEXT
011 1 CNS P/W NEXT
012 IFX=Y 2 LBL GOTO NEXT
013 0 PNT 1 CNS >PNT NEXT
014 CRS2-1 1 CNS 1 0 * NEXT
015 P/W / HSCL * >HCRD NEXT
016 LNN 0 2 NEXT
017 0 PNT P/W 1 - >PNT NEXT
018 1 WFM MEAN - RMS NEXT
019 1 CNS 2 CNS / SQRT * NEXT
020 1 WFM X<>Y MEAN NEXT
021 STOP NEXT
```

II. DESCRIPTION

The standard deviation (s) is a measure of the spread of a distribution about its mean. This program computes "s" for a sample of n measurements, which have been stored as 1 WFM. In this particular program, the measurement of interest is P-P. It could, however, be any desired parameter.

APPLICATIONS: (1) Describing in statistical terms the spread of measurements from a sample of waveforms.

(2) Establishing limits for testing purposes.

III. PROCEDURE

The measurements are accumulated as 1 WFM. Because each point is discrete and not part of a continuous waveform, the last point to be included in the calculations must be equal to the first point (Ø PNT). This compensates for the MEAN and RMS algorithms which take only one half of the first and last points.

The formula for standard deviation is:

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (X_i - \overline{X})^2}$$

where X_i is the ith sample measurement and $\overline{X}$ is the sample mean. This is approximated by using the MEAN and RMS algorithms.

A correction factor is applied to the RMS (line Ø19) to ensure that the sums of the squares of the deviations about the mean are multiplied by $\frac{1}{n-1}$. This is necessary because of the extra data point made equal to Ø PNT for reasons previously described.

IV. INPUTS: Enter n, the number of measurements to be made, before pressing "f, START".

OUTPUTS: 1 WFM is the record of measurements. The Y register contains "s" and the X register contains the measurement mean. Ø WFM is the deviations of 1 WFM about its mean.

V. RESTRICTIONS

The default value of n is P/W if n exceeds P/W.

If cursors are used to obtain the measurement, "CRS 1 Ø > HCRD OFF" must be inserted prior to line Ø14.

"n" should not be equal to or less than 2, obviously.

The program is general purpose with regard to n, and may be shortened if the values n can take are known. If n is very small, it might be easier to use constants rather than construct a waveform.

VI. COMMENTS

000	Set X = P/W	Initialize and store data as waveform (000–010)
001	Roll X and Y if n > P/W	
002	Clear X and proceed with X=n or X=P/W (default)	
003	Set Ø CNS, 1 CNS = n; set 2 CNS = n-1	
004	Initialize 1 WFM = Ø	
005	Initialize cursor 1 at Ø point and turn off	
006	Label 1	
007	Acquire waveform and perform measurement	
008	Set the next point of 1 WFM to the measurement value	
009	Y = Ø CNS, X = 1	
010	Return to Label 1 n-1 times	
011	Y = n, X = P/W	
012	Go to Label 2 if n = P/W	Accessing only actual data points of wave-form (012–017)
013	Set the n point equal to the Ø PNT value	
014	Turn on cursors; set X = 10_n	
015	Position cursor 2 at n point	
016	Label 2	
017	Set the last point equal to the Ø PNT value (relevant to calcu-lations only for n = P/W)	
018	Compute the RMS of the deviations from the mean	Calculate standard deviation (018–019)
019	$\sqrt{n/(n-1)}$ correction factor for RMS algorithm gives "s"	
020	Display 1 WFM and calculate MEAN	

17. LEAST SQUARES ESTIMATION

I. PROGRAM LISTING

```
000 OFF NEXT
001 P/W 1 - 0 >CNS NEXT
002 1 WFM 0 PNT 0 CNS >PNT NEXT
003 MEAN 1 >CNS NEXT
004 2 WFM 0 PNT 0 CNS >PNT NEXT
005 MEAN 2 >CNS NEXT
006 2 WFM X<>Y - 5 >WFM NEXT
007 3 WFM 0 PNT 0 CNS >PNT NEXT
008 3 WFM RMS MEAN IFX=Y 1 LBL GOTO NEXT
009 8 >CNS NEXT
010 3 WFM X<>Y - 6 >WFM NEXT
011 1 WFM * MEAN 9 >CNS NEXT
012 1 WFM 5 WFM * MEAN 1 0 >CNS NEXT
013 5 WFM ENTER * MEAN 1 1 >CNS NEXT
014 6 WFM ENTER * MEAN 1 2 >CNS NEXT
015 5 WFM 6 WFM * MEAN 1 3 >CNS NEXT
016 1 3 CNS * 1 1 CNS 1 2 CNS * - 1 4 >CNS NEXT
017 9 CNS 1 3 CNS * 1 0 CNS 1 2 CNS * - NEXT
018 X<>Y / 4 >CNS NEXT
019 1 0 CNS 1 3 CNS * 9 CNS 1 1 CNS * - NEXT
020 1 4 CNS / 1 5 >CNS NEXT
021 2 LBL GOTO NEXT
022 LNN 0 1 NEXT
023 0 ENTER 1 5 >CNS NEXT
024 1 WFM 5 WFM * MEAN NEXT
025 5 WFM ENTER * X<>Y MEAN / 4 >CNS NEXT
026 LNN 0 2 NEXT
027 2 WFM 4 CNS * 7 >WFM 3 WFM 1 5 CNS NEXT
028 * + 4 >WFM NEXT
029 1 CNS 4 CNS 2 CNS * - 1 5 CNS 8 CNS NEXT
030 * - 5 >CNS + 4 >WFM NEXT
031 1 WFM - RMS NEXT
032 P/W 0 CNS / SQRT * 6 >CNS NEXT
033 4 WFM 1 CNS - ENERGY NEXT
034 1 WFM 1 CNS - X<>Y ENERGY / 7 >CNS NEXT
035 4 WFM 7 CNS 6 CNS 5 CNS 4 CNS NEXT
036 1 5 CNS NEXT
037 STOP NEXT
```

II. DESCRIPTION

This multiple regression program computes a least squares estimation of a dependent variable (1 WFM) as a function of a linear combination of two independent variables (2 WFM, 3 WFM). The resulting equation takes the form:

$$1\ WFM = a + b(2\ WFM) + c(3\ WFM)$$

Simple regression is possible if 3 WFM is a constant. The program will then estimate 1 WFM as a function of 2 WFM.

Applications include understanding the relationship between phenomena, and predicting or estimating a dependent phenomenon.

III. ALGORITHM or PROCEDURE

The estimated a, b, and c are chosen such that the sum of the squared deviations between 1 WFM and 4 WFM (the estimated 1 WFM) are minimized. If

$$4\ WFM = \overline{a} + \overline{b}(2\ WFM) + \overline{c}(3\ WFM)$$

where $\overline{a}$ is the estimate of a, and so forth for $\overline{b}$ and $\overline{c}$, then the program chooses $\overline{a}$, $\overline{b}$, and $\overline{c}$ such that it minimizes:

$$\sum_{i=0}^{P/W-1} \left(1\ WFM_{\text{point } i} - 4\ WFM_{\text{point } i}\right)^2$$

For a more detailed understanding, refer to any text that covers statistics.

The program also calculates r^2, the coefficient of determination, which is the proportion of the variation in 1 WFM explained by the estimated waveform 4 WFM. r^2 varies between Ø and 1. The standard error of the regression, s, is also calculated.

The program can be streamlined for simple (1 WFM on 2 WFM) regression. Perform the following:

(1) Change line 028 to "2 WFM 2 CNS*4>WFM"
(2) Change line 030 to "1 CNS 4 CNS 2 CNS"
(3) Delete lines 007 through 023, and lines 026, 029 and 036

IV. INPUTS: 1 WFM, 2 WFM and 3 WFM (if used) must be stored prior to execution.

OUTPUTS: 4 WFM is the estimated 1 WFM. The stack contents are as follows:

W = r^2, coefficient of determination

T = s, standard error of the regression

Z = a, constant

Y = b, 2 WFM coefficient

X = c, 3 WFM coefficient

V. RESTRICTIONS

The waveform displayed is 4 WFM vs time. To observe the typical straight-line presentation of a simple regression, you must display 1 WFM vs 2 WFM and display 4 WFM.

Ignore the last points of any waveform. The original points were lost when set to the value of the first point. This is a correction for the MEAN, RMS, and ENERGY algorithms, since the program algebra treats each waveform as a set of discrete points (rather than continuous phenomena).

t-values for the coefficients and constant are not computed due to complexity.

Regression on more than two independent variables (waveforms) is best handled by a 4052. 7854 program length becomes unmanageable.

Horizontal scales of 1, 2, and 3 WFMs must be equal.

VI. COMMENTS

```
000  Turns off cursors, if on
001  Ø CNS = P/W-1
002  Last point = first point in 1 WFM
003  1 CNS = MEAN of 1 WFM
004  Last point = first point in 2 WFM
005  2 CNS = MEAN of 2 WFM
006  5 WFM = 2 WFM - 2 CNS } deviations about 2 WFM MEAN
007  Last point = first point in 3 WFM
008  Go to Label 1 if 3 WFM = constant
009  8 CNS = MEAN of 3 WFM
```

```
010  6 WFM = 3 WFM - 8 CNS }  deviations about 3 WFM MEAN
011  9 CNS = MEAN of (1 WFM · 6 WFM)            \
012  10 CNS = MEAN of (1 WFM · 5 WFM)            |
013  11 CNS = MEAN of (5 WFM)^2                  |  constants necessary to
014  12 CNS = MEAN of (6 WFM)^2                  |  calculate a, b, c
015  13 CNS = MEAN of (5 WFM · 6 WFM)            |
016  14 CNS = (13 CNS)^2 - (11 CNS · 12 CNS)     /
017  (9 CNS · 13 CNS) - (10 CNS · 12 CNS)  \  computes b estimate
018  4 CNS = (Line 017)/14 CNS             /
019  (10 CNS · 13 CNS) - (9 CNS · 11 CNS)  \  computes c estimate
020  15 CNS = (Line 019)/14 CNS            /
021  Go to Label 2
022  Label 1
023  15 CNS = Ø, the c estimate if 3 WFM = constant
024  X = MEAN of (1 WFM · 5 WFM)           \  computes b estimate if
025  4 CNS = Line 025/(MEAN of 5 WFM^2)    /  3 WFM = constant
026  Label 2
027  Z = 7 WFM = (2 WFM · 4 CNS), Y = 3 WFM, X = 15 CNS  \  4 WFM computed, but
028  4 WFM = Y · X + Z                                   /  still missing constant a
029  Z = 1 CNS - (4 CNS · 2 CNS), Y = 15 CNS, X = 8 CNS
030  5 CNS = a estimate; 4 WFM = 1 WFM estimate
031  RMS of (4 WFM - 1 WFM)                   \  computes s statistic
032  6 CNS = (Line 031) · (correction factor) /
033  ENERGY of (4 WFM - 1 CNS)                      \  computes r^2 statistic
034  7 CNS = (ENERGY of (4 WFM - 1 CNS))/line 33    /
035  4 WFM displayed, W = 7 CNS, T = 6 CNS, Z = 5 CNS, Y = 4 CNS  \  sets up display
036  X = 15 CNS                                                   /  & stack contents
```

18. ENVELOPE MODE

I. PROGRAM LISTING

```
000 2 ENTER 0 >CNS CLD NEXT
001 0 WFM 0 * 1 >WFM 3 >WFM NEXT
002 STOP NEXT
003 AQR 2 >WFM NEXT
004 LNN 0 1 NEXT
005 STOP NEXT
006 AQR 2 WFM - 4 >WFM NEXT
007 ENTER ENTER ENTER NEXT
008 ABS + 2 / 1 WFM - 5 >WFM ENTER NEXT
009 ABS + 2 / 1 WFM + 1 >WFM CLX NEXT
010 ABS - 2 / 3 WFM - 5 >WFM ENTER NEXT
011 ABS - 2 / 3 WFM + 3 >WFM NEXT
012 0 CNS 1 IFY>X - 0 >CNS 1 LBL GOTO NEXT
013 2 WFM 1 WFM + 6 WFM * 4 >WFM NEXT
014 2 WFM 3 WFM + 7 WFM * + 4 >WFM NEXT
015 2 DSW NEXT
016 STOP NEXT
017 1 2 8 >P/W AQR NEXT
018 0 WFM 0 * P/W 1 - 1 >CNS NEXT
019 LNN 0 2 NEXT
020 1 ENTER 1 CNS >PNT NEXT
021 1 CNS 2 IFY>X - 1 >CNS 2 LBL GOTO NEXT
022 0 WFM 6 >WFM 7 >WFM NEXT
023 CRS1 0 >HCRD CRS1> HCRD OFF HPRGT NEXT
024 START NEXT
```

II. DESCRIPTION

This program is an envelope mode for the 7854. It is suitable for recording maximum and minimum voltage excursions from a reference waveform over time.

The program is written in a fashion specifically for demonstrating its capability. Suggested modifications are to be found under "RESTRICTIONS" below. Some simple changes will make it a more useful program.

III. ALGORITHM or PROCEDURE

To use this program, first press 17 GOTO RUN. This causes the program to create two "envelope coefficient waveforms" that are composed of, alternately, 0 volt points and 1 volt points. One waveform is shifted one point with respect to the other. Later, when these coefficient waveforms are multiplied by the "maximum positive and minimum (greatest) negative waveforms" and the results added, the envelope waveform is created (4 WFM). (Half of the points of each waveform are lost, and the remaining points are combined).

The program then stops to allow you to check that the waveform you will acquire is the one you wish to be the reference waveform (2 WFM). If so, press RUN. The reference waveform is stored and the program stops. You may now modify the realtime waveform.

You are now about to make the first of two passes through a loop that records the maximum and minimum voltages. After the first pass, the program will stop again. This allows you to modify the realtime waveform a second time.

After each modification, press RUN. At the conclusion of the second pass, the program will end with the envelope and reference waveforms displayed.

To repeat, press f, START (or simply press RUN if you wish to calculate the envelope coefficient waveforms again. This is not necessary unless P/W or horizontal scales change). You will again stop prior to acquiring a new reference waveform.

IV. INPUT:
(a) Press 17 GOTO RUN
(b) (after first STOP) Press RUN to acquire reference waveform
(c) (after second STOP) Modify realtime waveform and press RUN
(d) (after third STOP) Modify realtime waveform and press RUN

OUTPUT: Operational waveform = envelope waveform, and the reference waveform is displayed.

1 WFM = maximum voltage recorded
2 WFM = reference waveform
3 WFM = minimum voltage recorded
4 WFM = envelope waveform
5 WFM = last stored negative deviations
6 WFM = } envelope coefficient waveforms
7 WFM = }

V. RESTRICTIONS

P/W are set at 128 for speed. 256 is possible, but to do 512 requires Option 2D. This is because you need to store 7 waveforms.

Note that autoscaling may cause different reference and envelope vertical scales.

Transients and rapid changes in the realtime waveform may not be recorded since the envelope is accomplished via program control. The time to loop through a succeeding acquisition will also depend on the sweep speed. Therefore, this program is most useful for slow changes or longer periods of time.

If you change sweep speeds, you should ensure that the horizontal scales of 6 WFM and 7 WFM reflect the change.

To run the envelope for more than 2 acquisitions, you should change line 000 to "n ENTER Ø > CNS CLD" and line 024 to "STOP". Then each time you simply enter n into the X register and press f, START (since you already did a 17 GOTO RUN). You can also delete line ØØ5.

To run the envelope loop continuously, change line 012 to "1 LBL GOTO". Then to leave the envelope loop, the operator simply presses STOP 13 GOTO RUN, and the envelope results will be displayed. Delete line ØØ5.

VI. COMMENTS

000 Set Ø CNS = 2, the number of acquisitions, and clear display
001 Initializes 1 WFM = 3 WFM = Ø
002 Stops, allowing for reference waveform setup before pressing "RUN"
003 Acquires and sets 2 WFM = reference waveform

004	Label 1	envelope recording
005	Stops, allowing changes in the realtime waveform before pressing "RUN"	
006	Acquires and stores deviations from reference = 4 WFM	
007	Loads stack with 4 WFM	
008	Stores positive deviations as 5 WFM	
009	Stores maximum positive deviations as 1 WFM	
010	Stores negative deviations as 5 WFM	
011	Stores maximum (greatest) negative deviations as 2 WFM	
012	Decrements Ø CNS and returns to Label 1	
013	Maximum (top) of envelope created	actual envelope creation
014	Minimum (bottom) of envelope added to create envelope	
015	Displays reference waveform	
016	Stops	
017	Sets P/W = 128 and acquires to get correct horizontal scale	creates envelope coefficient waveforms
018	Initialize Ø WFM = Ø and sets 1 CNS = P/W-1	
019	Label 2	
020	Sets the 1 CNS PNT = 1	
021	Decrements 1 CNS and returns to Label 2	
022	Sets 6 WFM = 7 WFM = Ø WFM	
023	Shifts 7 WFM one point to the right	
024	Go to line 000 and run	

19. SIMULTANEOUS XY AND YT

I. PROGRAM LISTING

```
000 LNN 0 1 TIME DOTS NEXT
001 HSCL 3 WFM CLX >HSCL 4 WFM CLX NEXT
002 >HSCL CLX 4 WFM * NEXT
003 VSCL 1 >CNS 3 WFM X<>Y >VSCL CLX NEXT
004 + VSCL 1 CNS / VXPD NEXT
005 CLX 0 VS HSCL NEXT
006 STOP 1 LBL GOTO NEXT
007 TIME 1 WFM 0 * P/W 1 - 1 >CNS NEXT
008 LNN 0 2 NEXT
009 1 ENTER 1 CNS >PNT NEXT
010 1 CNS 2 IFY>X - 1 >CNS 2 LBL GOTO NEXT
011 0 WFM 4 >WFM NEXT
012 0 * 1 + INTG MID - 4 WFM * NEXT
013 1 0 ENTER P-P VSCL / / VXPD 3 >WFM NEXT
014 4 WFM NEXT
015 CRS1 0 >HCRD CRS1> HCRD OFF HPLFT NEXT
016 6 GOTO NEXT
```

II. DESCRIPTION

You can easily display a waveform versus another in XY on the 7854. This program, however, will allow you to display the waveform in time simultaneously.

If you turn on one or both cursors, each will alternate between the XY and YT waveforms as the cursor(s) are moved. You can then see <u>where</u> on the YT waveform you are on the XY (within one point).

The cursor readout will function in its normal XY fashion. Time measurements on the YT waveform have to be done visually as long as you remain in XY, or you can press HCRD.

III. ALGORITHM or PROCEDURE

To use this program, press 7 GOTO RUN. This causes the program to create a (1,0) coefficient waveform composed of alternating 1V and ØV points. A ramp is also created and multiplied by the (1,0) waveform. The (1,0) waveform is shifted one point to the left for later use.

After the program stops, you can display XY and YT. To display 1 WFM vs 2 WFM, enter 1 WFM, 2 WFM and press RUN.

2 WFM will be multiplied by the (0,1) waveform and the ramp added to it. This makes Ø WFM a composite of 2 WFM and the ramp waveform. In XY mode, every other point (i.e. ramp points) will cause 1 WFM to be displayed as if in time.

You can also display 2 WFM vs 1 WFM by entering 2 WFM, 1 WFM and pressing RUN.

IV. INPUT: (a) Press 7 GOTO RUN
(b) (after STOP) Press 1 WFM 2 WFM RUN (or 2 WFM 1 WFM RUN. You can use other waveforms, but not 3 WFM or 4 WFM).

OUTPUT: Operational waveform displayed YT by use of alternate dots. X register contains the horizontal scale of the YT display.

3 WFM = ramp
4 WFM = (Ø,1) waveform

V. RESTRICTIONS

None. Note that the XY and YT waveforms are superimposed and cannot be independently moved.

VI. COMMENTS

000 Label 1, change to time and dots format
001 Set the HSCL of 3 and 4 WFMs = HSCL of 2 WFM
002 Multiply the " vs waveform" by (1,0) coefficient waveform
003 Set VSCL of ramp waveform equal to "vs waveform" VSCL
004 Add ramp to Ø WFM and scale properly
005 Display XY and YT and set X register = YT horizontal scale
006 Stop; go to Label 1 after RUN is pressed
007 Initialize Ø WFM = Ø with VSCL = 1 WFM VSCL, set 1 CNS = P/W-1
008 Label 2
009 Set the 1 CNS point = 1
010 Decrement 1 CNS by 2 and go to Label 2
011 Store (0,1) coefficient waveform as 4 WFM
012 Create ramp and multiply by (0,1) coefficient waveform
013 Scale ramp waveform
014 Operational waveform = (0,1) coefficient waveform = 4 WFM
015 Shifts 4 WFM one point to the left
016 Go to line 006

20. GAIN AND PHASE MEASUREMENT

1. PROGRAM LISTING

```
000 VMDL 4 AVG 3 >WFM NEXT
001 CRS1 HCRD 7 >CNS CRS2-1 NEXT
002 0 WFM MID PER 0 >CNS NEXT
003 3 6 0 / 1 >CNS NEXT
004 CRS1 MID >VCRD HCRD 5 >CNS NEXT
005 OFF P-P 2 >CNS NEXT
006 VMDR 4 AVG 4 >WFM NEXT
007 P-P 2 CNS / 3 >CNS NEXT
008 CRS1 7 CNS >HCRD NEXT
009 MID >VCRD HCRD 4 >CNS NEXT
010 5 CNS - 1 CNS / 6 >CNS NEXT
011 CRS1 7 CNS >HCRD CRS2-1 NEXT
012 3 DSW 3 CNS 6 CNS NEXT
013 STOP NEXT
```

II. DESCRIPTION

This program measures the gain and phase differences between an input and output signal of an amplifier.

III. ALGORITHM or PROCEDURE

You must first connect your input and output signals to the left and right vertical amplifiers respectively. Manually acquire the input signal and place the cursors for a proper period measurement.

Now press f START. The program will acquire the input waveform and measure the period, calculate time per degree, and measure P-P amplitude. It then acquires the output waveform and goes on to calculate the gain and phase.

$$\text{gain} = \frac{\text{output amplitude}}{\text{input amplitude}} \qquad \text{phase} = \frac{t_2 - t_1}{\text{period}/360^\circ}$$

IV. INPUT:
- (a) Input and output signals connected to left and right vertical amplifiers.
- (b) Cursors positioned for proper period measurement on input waveform.
- (c) Press f START

OUTPUT:
X register = phase = 6 CNS
Y register = gain = 3 CNS
3 WFM = input waveform
4 WFM = output waveform = Ø WFM

V. RESTRICTIONS

Period (PER) restrictions.

Input/output amplifier hookup as previously stated.

The maximum possible phase measurement is a function of waveform symmetry. For example, phase will never be measured beyond 180^{o} for a symmetrical waveform. Other waveforms may allow greater or smaller phase measurements.

VI. COMMENTS

000 Acquire averaged input waveform = 3 WFM
001 Store cursor 1 position = 7 CNS; turn on cursors
002 Set Ø CNS = period of 3 WFM (Ø WFM)
003 Calculate time per degree
004 Set 5 CNS = t_1
005 Set 2 CNS = input amplitude
006 Acquire averaged output waveform = 4 WFM
007 Calculate gain = 3 CNS
008 Reposition cursor 1 to 7 CNS HCRD
009 Set 4 CNS = t_2
010 Calculate phase = 6 CNS
011 Reposition cursor 1 to 7 CNS HCRD; turn on cursors
012 Display input waveform, Y = 3 CNS, X = 6 CNS

7854 SALES AID REFERENCE GUIDE

The 7854 is, as you well know, a very exciting product. Along with that excitement comes the reality that the 7854 is also complex. To address this complexity, many supportive aids have been developed. The following is a list of those aids. You should have these aids in your office. If not, call us.

Roger Loop
Mktg. Product Line Mgr.
July 18, 1981

1. Ringmaster #1 - "The 7854 Intelligent Oscilloscope"

2. Ringmaster #2 - "The 7854 Sales Slide Presentation" "Company Confidential"

3. Video Tape - 7854 Operation (068-0085-00)

4. Brochures - WP1310 AX-4382
   7854 A-4166
   7854 Specs AX-4178

5. Application Note - "GPIB Communication with the 7854" - AX-4416
   - "Introduction to 7854 Oscilloscope Measurement and Programming Techniques" - 42AX-4682
   - "Measurement Variety. An Engineering Challenge Featuring the 7854" - AX-4281

6. Tekscope - Digital waveform processing in a high performance 7000 series oscilloscope, Volume 12, Number 3.

7. 7854 Oscilloscope Operators Manual - 070-2873-00

8. Digital Oscilloscope Fundamentals - Concept Training for Sales Engineers

9. 7854 Sales Guide - "Company Confidential"

10. Reprint - Electronics March 13, 1980 - Introducing the 7854

11. WP1310 Utility Software - 020-0626-00

12. National Advertising and the Sales Leads That Result

13. Trade Shows and Leads That Result
    Wescon
    Electro
    Powercon

14. Catalog - page #12 Digital Storage
    page #103-105

15. Mailers featuring 7854, 7612D, 7912AD

    A - "Simplifying EMI Emissions Testing"
    B - "The Automatic Answer to Your Measurement Needs"
    C - "The Automatic Answer to Your Peripheral Testing Needs"
    D - "The Automatic Answer to Your Component Testing Needs"

16. Handshake - New Digitizers for a New Decade, Vol. 5, #1