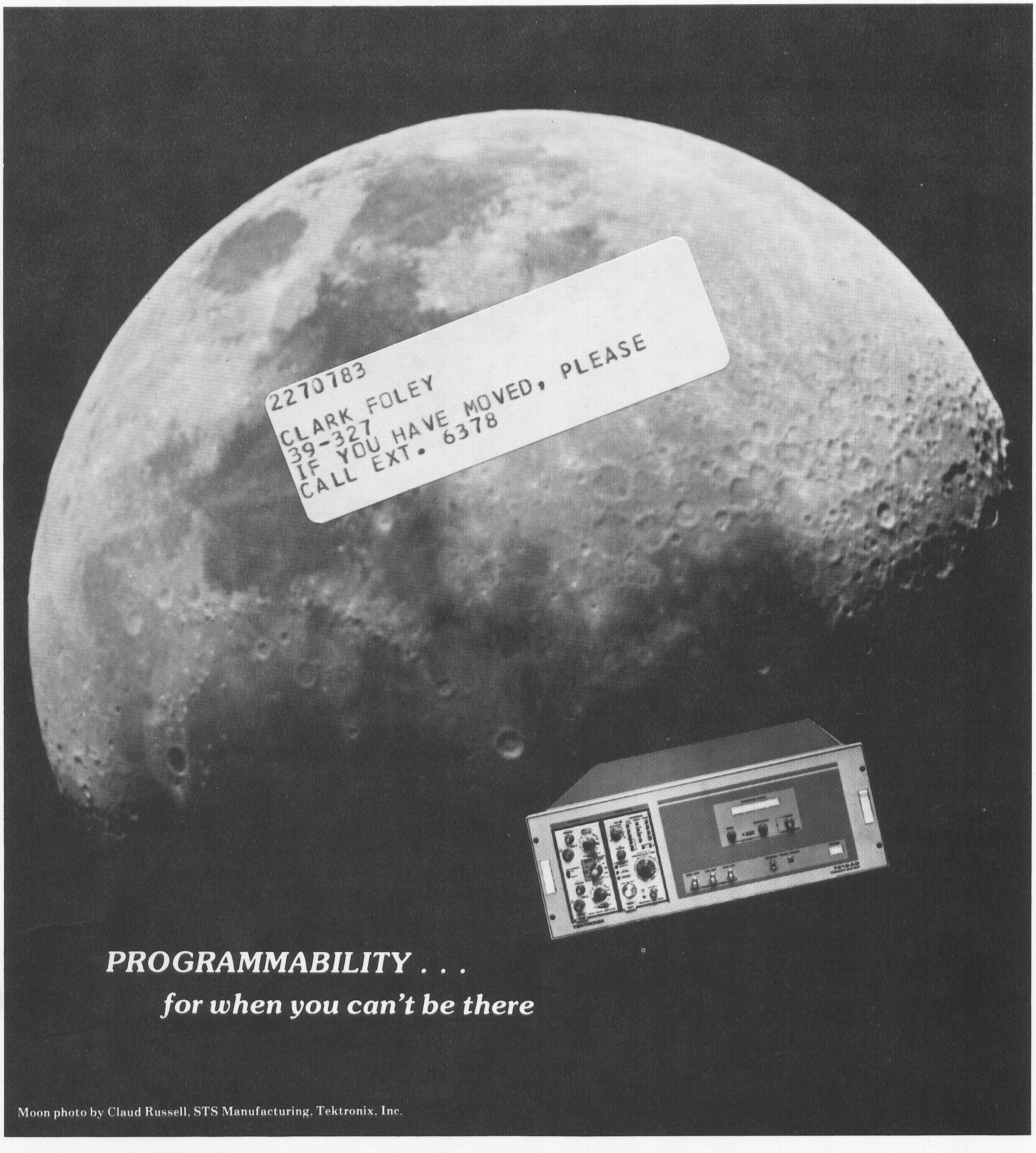
Vol. 4 No. 1 FALL 1978

Tektronix®
COMMITTED TO EXCELLENCE

HANDSHAKE

Newsletter of the Signal Processing Systems Users Group

INSIDE:



2270783
CLARK FOLEY
39-327
IF YOU HAVE MOVED, PLEASE
CALL EXT. 6378

Moon photo by Claud Russell, STS Manufacturing, Tektronix, Inc.

The Programmable 7912AD— A giant step toward remote test and measurement

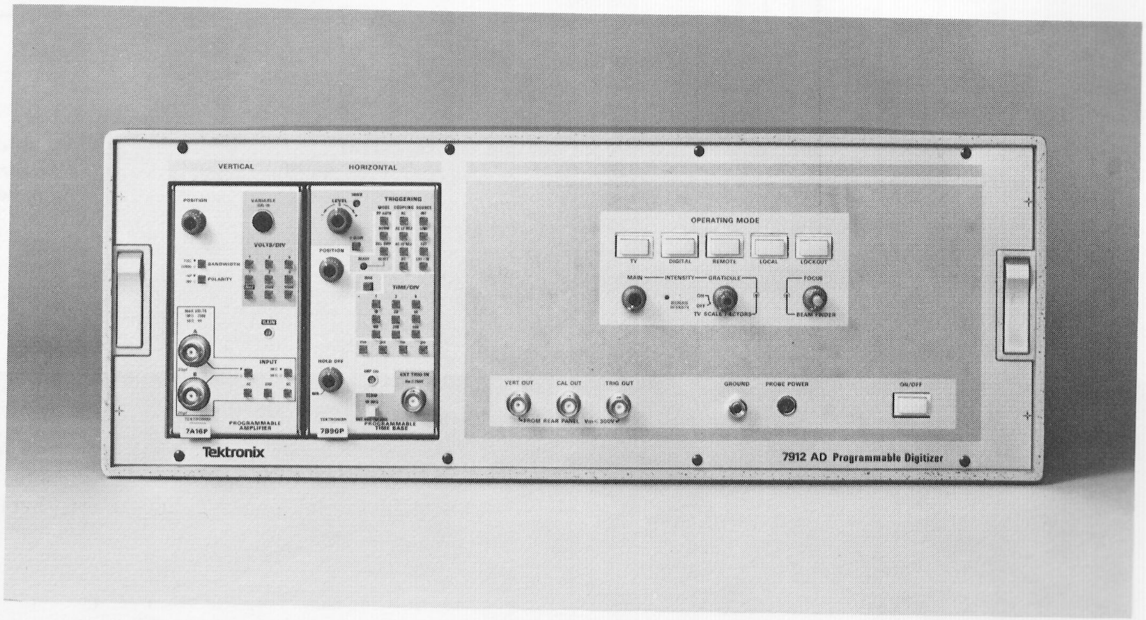


Fig. 1. A new breed of oscilloscope, the microprocessor-based 7912AD Programmable Digitizer. It's fast. Its display is remote, and it is programmable.

> remote /ri-'mōt/ adj 1: separated by great intervals 2: far removed in space, time, or relation (Webster's New Collegiate Dictionary)

How much is a great interval? How far is far removed? From here to the moon . . . 10 kilometers . . . 20 meters?

In many test and measurement situations, even an arm's length can be a great interval. Consider dangerous or unpredictable test techniques—surge testing distribution transformers or lightning arresters, for example. The surge voltages are often high, tens of thousands of volts and more. There's always the possibility of arcs or flashovers, not to mention the occasional explosion of a defective device under test. Indeed, devices are sometimes purposely tested to violent destruction. So testing is done in protective cages or conducted from bunkers. And the instrumentation, to avoid the ground loops and possible distortions of longer cable runs, is often placed in the testing area. It may only be an arm's length away from you, just the other side of a concrete wall, but it might as well be on the moon if you need to make an adjustment while testing is under way.

In this example and many others, remoteness is not so much a matter of being far removed. It's more controlling without touching.

Automation! Programmability!

Going remote . . .

As a good example of how to go remote, think about a reasonably challenging measurement you frequently make. Picture the instruments in place and the measurement in progress.

More than likely, one of the instruments you are picturing is an oscilloscope. Oscilloscopes are widely used, powerful, general purpose measurement tools. So let's zero in on the oscilloscope portion of your measurement as the starting point for going remote. Put it and your test behind a concrete wall, out of sight and out of reach.

Now, what would you have to do to that oscilloscope to operate it remotely?

Your first concern might be how to turn it on, but that is easily solved. Either turn it on when you initially connect the equipment, or control the power with an extension cord and a power switch on your side of the wall.

So now the oscilloscope is on. The test signal is being displayed. How do you see the signal through the concrete wall?

Somehow you have to separate the display from the oscilloscope and put it on your side of the wall.

In the past, that has been done with a video display on a TV monitor, and that is not a bad solution for the present.

But now that you've seen the signal on the monitor, imagine what comes next. Maybe the oscilloscope intensity is too low for a rise time to be visible, or the display is positioned partially off screen, or the vertical sensitivity is adjusted too low, or the time base isn't set right. Now is when you want to run around that wall and start adjusting knobs and pushing buttons. But you can't. Remember, you're going remote. So you have to have a way to control all of those things from your side of the wall.

. . . with intelligence

A common solution used to be to remove the front-panel controls and extend them to a control console with cable harnesses. With today's technology, however, there is a much more intelligent approach. Not only intelligent from the standpoint of using available resources, but intelligent from the standpoint of putting more of the work load on the instruments. Advanced integrated circuit technology, inexpensive minicomputers, specialized software, microprocessors with firmware, and a General Purpose Interface Bus (GPIB or IEEE 488 Standard Bus) all make it possible. And they are all part of a new breed of intelligent oscilloscopes, the programmables.

An example of this new breed is shown in Fig. 1. It is the TEKTRONIX 7912AD Programmable Digitizer. It's not even called an oscilloscope, and you'll notice it doesn't have an oscilloscope display, yet it works much the same. On its back panel it has X-Y-Z and video outputs so you can send an oscilloscope-like display to a distant X-Y-Z or TV monitor.

But there are more important differences than just the display. All the Operating Mode buttons on the front panel (with the exception of beam finder) are programmable. You don't have to touch them to make them work. For example, to monitor the signal being acquired by the 7912AD, all you have to do is send the message, MODE TV, and the 7912AD is automatically set to the TV mode of operation. Of course, you can do the same by pressing the TV button, but that is not remote operation.

What happens when you go to TV mode is that the waveform acquired by the 7912AD is scanned off a small diode target in a TV format. The waveform is originally written on the target by a high-velocity electron beam capable of writing at rates allowing capture of a single-shot 500 megahertz sinusoid of eight divisions equivalent amplitude or an eight division single-shot pulse with a one nanosecond rise time. This occurs in the same manner, essentially, as in a conventional oscilloscope. There are accelerators and deflectors that cause the beam to trace the waveform on the target. The target, however, is a digitizing array of diodes instead of the phosphor plate of an oscilloscope cathode-ray tube. Once the beam writes the waveform, as a series of dots on the diode target, another electron beam reads the series of dots off the target at a slower, easier-to-handle rate. In the TV mode, the reading is in the standard television format and can be viewed on a TV monitor.

And, just like your home television set, there are those times when the image brightness is not high enough for good viewing. Or maybe it's too bright. In either case, you can adjust this by turning the intensity knob, or, without leaving your seat, you can send a MAI 256 message. This message sets Main Intensity to a level of 256 out of 0 to 1023 levels of intensity, with 1023 being full bright. There's also a graticule intensity control, and it, too, can be adjusted either manually or by a GRI 200, where the 200 refers to an intensity setting on a scale of 0 to 255.

With the waveform and the graticule displayed on the TV monitor, you can make oscilloscope measurements in the conventional manner. However, the Digital Mode of operation is far more accurate, much faster, and of greater resolution.

To get to the Digital Mode, just send a MODE DIG to the 7912AD. The effect is the same as pushing the Digital Mode button.

In the Digital Mode, the waveform acquired by the 7912AD is read off the target in a digital format for transfer to a processing unit or even just for storage as part of a larger data base. In either case, the 512-element array format allows computer processing of waveforms for a variety of measurement information.

Before transfer of the waveform data, however, it is often advisable to go through some preprocessing to clean things up. Several approaches to preprocessing are included in the 7912AD in the form of a microprocessor with firmware. Signal averaging, a process for improving the signal-to-noise ratio of repetitive signals, is probably the most familiar of the operations included in the firmware repertoire.

But wait!

Here we are into data transfers and processing, and the waveform may not even be centered on the viewing area. Or worse, maybe a changing bias has shifted it completely off the display. Whatever repositioning might be necessary, you can do with program statements, remotely.

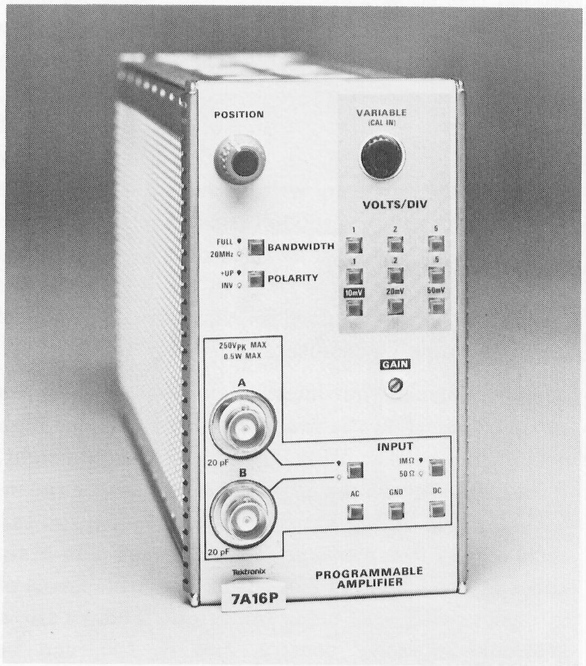


Fig. 2. The 7A16P Programmable Amplifier is a 200-megahertz, microprocessor-based vertical plug-in for use with the 7912AD Programmable Digitizer.

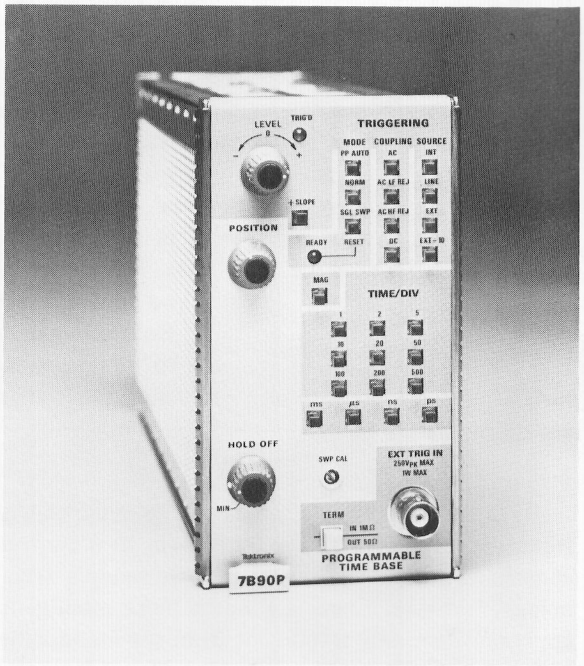


Fig. 3. The 7B90P Programmable Time Base is also microprocessor-based and is intended for use with the 7912AD Programmable Digitizer.

Programs control the vertical

Fig. 2 shows the 7A16P Programmable Amplifier that controls the vertical adjustment of the 7912AD. Notice that its controls are quite similar to the vertical controls of many oscilloscopes. These controls, except GAIN and VARIABLE, and selectors can be operated remotely, by simple program statements.

For example, let's say you need to move the waveform up two divisions in the display area. The message to do this amounts to POS 2. If, on the other hand, you need to move it down two divisions, use POS-2. The full range of available movement is from −10.22 to +10.24 divisions.

The same type of simple statement is used to select vertical sensitivity. For example, the message V/D .5 sets the vertical sensitivity to 0.5 volts per division, just the same as if you'd pressed the .5 button on the 7A16P plug-in.

With this capability for remote operation, there's really no need in most applications for you to worry about vertical adjustments again. Just work the sequence of adjustments out once. Then put them into a program, a string of messages for the microprocessors in the 7A16P and 7912AD, and let the instrument automatically set itself up for the best waveform display. An example of how this routine might be done for repetitive waveforms is covered in an article about autoranging the 7912AD vertical scale factor which follows in this issue.

Programs control the horizontal

Just as often, you'll need to adjust the horizontal variables of your waveform display. The 7B90P Programmable Time Base (Fig. 3) allows you to do this in the same manner as with any other type of oscilloscope. The difference is that the 7B90P contains a microprocessor and firmware that lets you also program all of its front-panel controls (except SWP CAL and TERM) from a remote location.

So, just as you would do for vertical setup, the horizontal setup of the instrument can be expressed as a sequence of simple messages. Put these with the vertical control messages, and you have complete, automatic instrument setup.

Build and control an entire automatic test system

Each operation of the 7912AD and its plug-ins is available singly or in combinations by sending simple messages—messages like MODE DIG, MAI 256, and so forth. But how do you send those messages?

The answer is via the software of any GPIB (IEEE 488) compatible computer, minicomputer, or calculator. The 7912AD operations and data transfers are all handled over the GPIB bus. This means that all of its operations and adjustments can be controlled by a computer program. Waveform data can, therefore, be processed for information like rise time, fall time, rms level, frequency spectrum, or subjected to complex analyses involving many waveforms and waveform parameters.

Also, importantly, there is an ever expanding variety of GPIB-compatible instruments on the market. Voltmeters, power supplies, frequency sweepers, and many more are currently available for powering, stimulating, and monitoring special test parameters. By choosing instruments that are fully programmable and by interfacing them to a 7912AD system, entire tests can be run under TEK SPS BASIC software—without touching or even looking at an instrument. Just express the operations in a program, type RUN on the computer terminal, and sit back.

By Bob Ramirez, HANDSHAKE Staff

Step up to autoscaling

If you are thinking about remote test and measurement, you are probably thinking about remote instrument control, too. Fortunately, if you are using the 7912AD Programmable Digitizer and its programmable plug-ins, you get remote measurement and remote control as part of the same package. All important controls are programmable. In fact, with appropriate software, the 7912AD can be turned into a fully automatic oscilloscope.

Such an instrument could be put to work to track a repetitive waveform whose amplitude changes. When a change occurs, the 7A16P Programmable Amplifier, which is the 7912AD front end, may require adjustment to bring the waveform back within range of the 7912AD. This can be handled by a vertical scaling routine; see Fig. 1 for a flowchart of such a program and Fig. 2 for a listing of the program.

The routine operates by checking to see if the full waveform is acquired with reasonable resolution. If it isn't, as shown in Fig. 3, the routine climbs up or down the 7A16P attenuator ladder while offsetting the waveform with the position control. The idea is to change the scale factor and position only as needed to allow the greatest resolution possible. To do this, the waveform is tested for four things:

1) Is it too big?
2) Is it too high?
3) Is it too low?
4) Is it too small?

Failing any of these tests causes a change in 7A16P parameters to correct the display. The waveform is then digitized and tested again. When the waveform is finally acquired so that it is not too big, not too high, not too low, and not too small — in other words, just right — it is graphed and the 7912AD reset so the front panel can be changed, if desired, before the program is tried again.

Although these tests and control statements could be part of an autoranging subroutine working with a larger program, they are presented here as a complete program incorporating its own serial poll, set command, query command, and read-data subroutines. As such, it illustrates how to use TEK SPS BASIC in a CP4165 or other TEKTRONIX CP-Series controller to program the 7912AD and plug-ins. The serial poll routine is left enabled to handle any unusual conditions and interaction with the operator. The set and query command subroutines handle setting and reading 7912AD and plug-in operating parameters. The read-data subroutine handles 7912AD block-binary data transfers.

The program gets underway by loading the GPI (general-purpose interface) software driver and initializing the IEEE 488 bus and instrument addresses. The software is also set to a comfortable time-out value and told to respond to any service request (SRQ) on the bus.

To prevent double-talk on the REMOTE button (an operator interrupt, explained later), the interrupt is disabled by "REM OFF" once the program is underway. Graticule intensity (GRI) is set to zero so the graticule dots are not mixed into the waveform data. The value of the main intensity (MAI) is acquired at the same time to be used later. Also, because false data points could confound the logic to follow, the 7912AD is readied to reject scan-converter target defects with the "DIG DEF" command.

Step up to autoscaling

The program then proceeds to acquire data and test it. Main intensity is set to zero between digitize operations to prevent ghosting if the waveform is moved and digitized again (momentary ghosting can occur with repetitive waveforms due to target storage). The first test checks for a waveform that is too large. Full-scale for 7912AD data is 512 levels. A maximum peak-to-peak value of 480 is allowed by the program to provide some damping for the autoscaling — we don't want programmed dither, just programmed autoscaling.

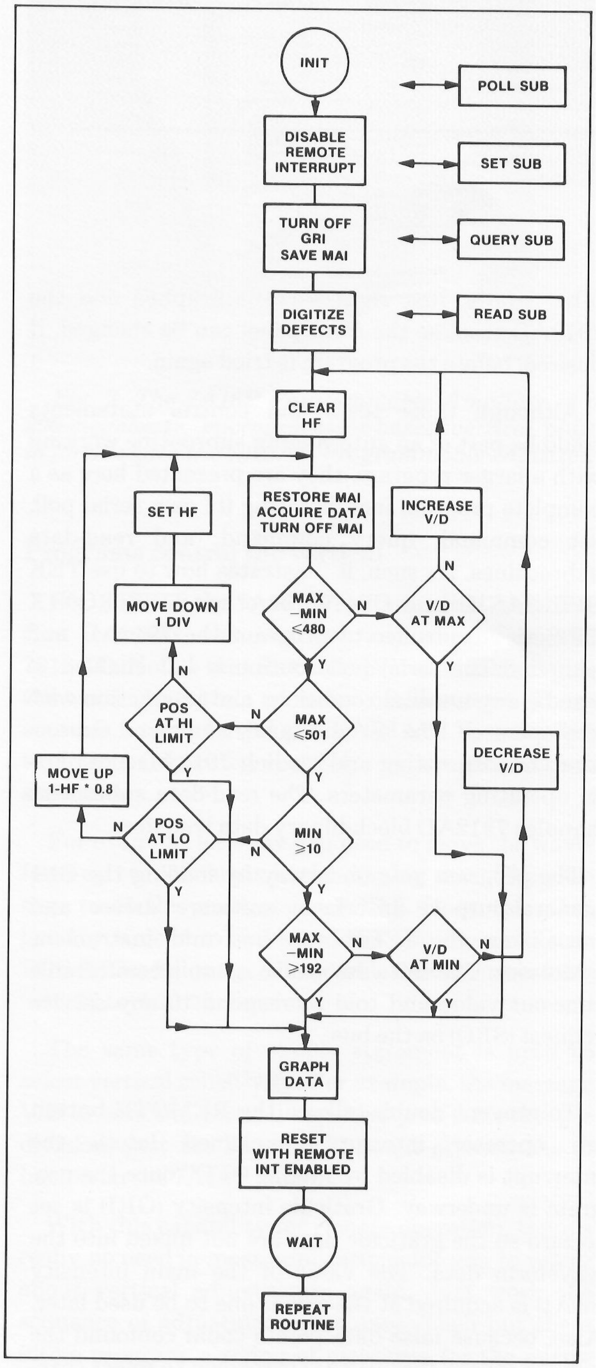


Fig. 1. Flow diagram of the 7912AD/7A16P autoscaling program.

If the waveform is too big, the program tries to bring it into range by increasing the volts/division. To do this, it looks for the uneven rung on the attenuator ladder's 1-2-5 sequence and adjusts for it before doubling the volts/division. A test is included when shifting either up or down to detect the attenuator end points, 5 and 0.01 volts/division, so neither is passed.

To sort out the scale-factor and position values from the 7A16P, TEK SPS BASIC string functions are called into action. Although the ASCII strings used by the 7A16P look like English to you and me, they are a foreign language to the controller. To make them into the numbers that the controller likes and then convert them back to ASCII strings for the 7A16P, TEK SPS BASIC string functions VAL, POS, LEN, and STR are used. For a full explanation of these, see your TEK SPS BASIC Software System manual.

With the waveform in range, the program checks to see that the top of the waveform is not too close to the top of the target (511). Again, a margin of safety is preserved. If it's too high, the waveform is moved down quickly (one division at a time); finer adjustment is left to the up test — that's the purpose of the hi-flag, HF. Both going up or going down, the program checks to avoid pushing beyond the range of the position (offset) control.

After satisfying the maximum value test, the program proceeds to assure that the waveform doesn't crowd the bottom of the target. If it does, it's moved up. However, because the waveform may have blundered off the target by marching blindly down one division at a time as part of the max-value test, HF is used to set the rate of upward repositioning. This rate, for waveforms that have already been moved down, is 0.2 division at a time.

After successfully passing the minimum test, the waveform is on target. But the program isn't satisfied until it obtains adequate resolution, defined to be 192 data levels, peak-to-peak. Why only 192? Because of the uneven rung in the 1-2-5 attenuator ladder. Of course if the scale factor is changed, the flag is reset, and positioning is again attempted.

When all tests are satisfied, the program graphs the waveform and returns control to the 7912AD front panel. Then the program waits by looping with an IF statement until it detects remote-request status (193). When it detects a remote request, the routine goes through autoscaling again. This wait is

```
10 REM INIT ROUTINE
15 LOAD "GPI"
20 ATTACH #1 AS GPI0:
25 PRINT "ENTER PRIMARY ADDRESS"
30 INPUT PA
35 LA=32+PA
40 TA=64+PA
45 PRINT "ENTER SECONDARY ADDRESS"
50 INPUT SE
55 SA=96+SE
60 SIFTO #1,3000
65 WHEN #1 HAS "SRQ" AT 60 GOSUB 1000
100 REM DISABLE REMOTE REQUEST
105 A$="REM OFF"\D=0
110 GOSUB 2000
115 REM TURN OFF GRI, SAVE MAI
120 A$='GRI 0;MAI'\D=0
125 GOSUB 3000
130 M$=A$
135 REM ACQUIRE DEFECTS
140 A$="DIG DEF,199"\D=0
145 GOSUB 2000
150 REM INIT FLAG
155 HF=0
160 REM RESTORE MAI, ACQUIRE DATA
165 A$=M$\D=0
170 GOSUB 2000
175 WAIT 33
180 A$="DIG DAT;MAI 0;DEF ON;ATC;READ ATC"
185 D=0
190 GOSUB 2000
195 GOSUB 4000
200 REM TEST MAX-MIN >>>
205 IF MAX(DA)-MIN(DA)<=480 THEN 265
210 A$="V/D"\D=1
215 GOSUB 3000
220 VD=VAL(SEG(A$,6,LEN(A$)))
225 REM TEST V/D
230 IF VD=5 THEN 460
235 REM INCREASE V/D
240 IF POS(A$,"2",1)<>6 THEN 250
245 VD=VD+VD/4
250 A$="V/D "&STR(VD*2)\D=1
255 GOSUB 2000
260 GOTO 150
265 REM TEST MAX VALUE
270 IF MAX(DA)<501 THEN 330
275 A$="POS"\D=1
280 GOSUB 3000
285 PO=VAL(SEG(A$,5,LEN(A$)))
290 REM TEST POS RANGE
295 IF PO<-9.22 THEN 470
300 REM MOVE WAVEFORM DOWN
305 A$="POS "&STR(PO-1)\D=1
310 GOSUB 2000
315 REM SET FLAG
320 HF=1
325 GOTO 160
330 REM TEST MIN VALUE
335 IF MIN(DA)>10 THEN 390
340 A$="POS"\D=1
345 GOSUB 3000
350 PO=VAL(SEG(A$,5,LEN(A$)))
355 REM TEST POS RANGE
360 IF PO>9.24+HF*.8 THEN 470
365 REM MOVE WAVEFORM UP
370 PO=PO+1-HF*.8
375 A$="POS "&STR(PO)\D=1
380 GOSUB 2000
385 GOTO 160
390 REM TEST MAX-MIN <<<
395 IF MAX(DA)-MIN(DA)>200 THEN 450
400 A$="V/D"\D=1
405 GOSUB 3000
410 VD=VAL(SEG(A$,6,LEN(A$)))
415 REM TEST V/D FOR MIN
420 IF VD=.01 THEN 480
425 IF POS(A$,"5",1)<>6 THEN 435
430 VD=VD-VD/5
435 A$="V/D "&STR(VD/2)\D=1
440 GOSUB 2000
445 GOTO 150
450 A$="SUCCESS"
455 GOTO 485
460 A$="WAVEFORM EXCEEDS MAX INPUT"
465 GOTO 485
470 A$="WAVEFORM OUTSIDE POSITION RANGE"
475 GOTO 485
480 A$="WAVEFORM TOO SMALL FOR FULL RESOLUTION"
485 REM GRAPH WAVEFORM
490 PAGE
495 WINDOW 0,511,0,511
500 SETGR WIND,TICS 10,8
505 GRAPH DA
510 MOVE 0,550
515 PRINT A$;"--PRESS REMOTE FOR ANOTHER WAVEFORM"
520 REM RESET INSTRUMENT
525 SB=0
530 A$="MODE TV;REM ON"\D=0
535 GOSUB 2000
540 SIFCOM #1,LA,SA,"GTL"
545 IF SB<>193 THEN 545
550 GOTO 100
1000 REM SERIAL POLL SUB
1010 POLL #1,SB,PS,SS;TA,SA;TA,SA+1;TA,SA+2
1020 IF SB+PS+SS=0 THEN 1040
1030 PRINT "DEVICE";SS-96;" STATUS =";SB
1040 RETURN
2000 REM SET COMMAND SUB
2010 PUT A$ INTO #1,LA,SA+D
2020 RETURN
3000 REM QUERY COMMAND SUB
3010 PUT A$&"?" INTO #1,LA,SA+D
3020 GET A$ FROM #1,TA,SA+D
3030 RETURN
4000 REM READ AND GRAPH DATA SUB
4010 IFDTM #1,"UNP"
4020 GET X FROM #1,TA,SA
4030 IF CHR(X)<>"%" THEN 4180
4040 IFDTM #1,"PAK","HBF"
4050 GET BC FROM #1,TA,SA
4060 DI=(BC-1)/2-1
4070 DELETE DA
4080 INTEGER DA(DI)
4090 GET DA FROM #1,TA,SA
4100 DA=DA/2
4110 IFDTM #1,"UNP"
4120 GET X1 FROM #1,TA,SA
4130 GET X2 FROM #1,TA,SA
4140 IF CHR(X2)<>";" THEN 4160
4150 RETURN
4160 PRINT "MESSAGE DELIMITER ERROR"
4170 GOTO 4190
4180 PRINT "BINARY PARSER ERROR"
4190 PRINT "PROGRAM ABORTED"\STOP
5000 REM STATUS SUB
5010 PRINT "ENTER DEVICE #"
5020 INPUT D
5030 GETSTA #1,SN,TA,SA+D
5040 PRINT "STATUS OF DEVICE";D;" IS";SN
5050 STOP
6000 REM DEVICE CLEAR SUB
6010 SIFCOM #1,"DCL"
6020 STOP
7000 REM GOTO LOCAL SUB
7010 SIFCOM #1,LA,SA,"GTL"
7020 STOP
8000 REM RESET SUB
8010 A$="MODE TV"
8020 D=0
8030 GOSUB 2000
8040 STOP
```

Fig. 2 TEK SPS BASIC listing of the 7912AD/7A16P autoscaling program.

continued on page 13

Beating bandwidth specs with software compensation

When selecting a waveform digitizer, most people rely heavily on a bandwidth specification to indicate instrument performance. This is particularly true when the digitizer is being considered for acquiring transient waveforms since high-frequency content determines much of their structural detail (rise time, aberrations, etc.). If the digitizer lacks sufficient bandwidth, then the detail of the acquired transient is smoothed or filtered out.

So bandwidth is a key performance factor. Yet most reputable instrument manufacturers underspecify it. Rather than touting a mean or "typical" value, they specify and guarantee a lower bandwidth value so their instruments all meet or exceed the published specification. And therein lies the means for you, along with the help of software, to get more bandwidth than you bargained for.

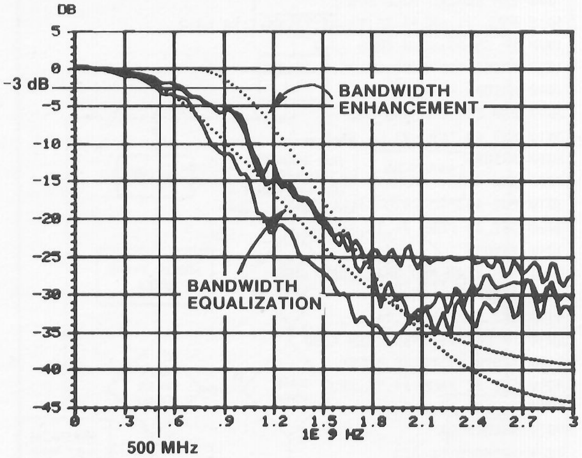


Fig. 1. Measured frequency-response amplitudes (solid lines) for three transient digitizers show significant variations between instruments above the specified 500 megahertz bandwidth. Software can, in effect, extend and equalize these bandwidths as indicated by the dotted lines. In this example, using TEKTRONIX R7912 Transient Digitizers with 7A19 plug-in amplifiers, software enhancement extends 500-megahertz bandwidth to one gigahertz.

Cashing in on conservatism

This bandwidth situation is graphically portrayed in Fig. 1. The solid curves show measured frequency-response amplitudes for three different R7912 Transient Digitizers equipped with 7A19 plug-in amplifiers. The Tektronix Catalog specifies this as a 500-megahertz combination. But, as shown in Fig. 1, only one of the tested instruments fell close to the 500-megahertz specification. The other two exceed the specification by having 3-dB points at 600 megahertz plus.

This means that on some instruments you already have an extra 100 megahertz to play with thanks to conservative specification writing. However, you can get even more by using software enhancement techniques. In the case of the R7912-7A19 combinations used for Fig. 1, software can double the bandwidth—from 500 megahertz to one gigahertz!

For other instruments and plug-ins, the 7912AD Programmable Waveform Digitizer for example, the situation is the same. Only the numbers are different.

Rolling up roll-off

In Fig. 1, the proposed bandwidth enhancement is indicated with a dotted line (also indicated is a bandwidth equalization curve, but more on that later). To accomplish the indicated enhancement, which amounts to applying a correction filter to all measurements, you must first determine the frequency response of the digitizer. The specific frequency-response function is then used in determining correction factors to be applied to subsequent measurements made with that particular instrument. Fig. 2 blocks out the general process.

In Fig. 2, each digitizer is first characterized by acquiring a fast step signal. Any pulse or step generator with good stability, flat amplitude, and linear phase through one gigahertz is acceptable, with the TEKTRONIX S-52 Pulse Generator Head being a good choice. The step is fed to the instrument input and used to estimate its frequency response, H_i as H_{ei}. Then, using a frequency response that you would like to be the standard (H_s in Fig. 2 and indicated by dotted lines in Fig. 1), H_i is decomposed into two subsystems consisting of H_s and H_{di} such that $H_i=H_{di}H_s$. From this, H_{di} is estimated with software as $H_{edi}=H_{ei}/H_s$ and saved for use in correcting future measurements to the desired standard, H_s.

When subsequent measurements are made with that digitizer, they are corrected by one of two algorithms indicated in Fig. 2. In the case of measuring pulse-like signals, the measurement is modified to appear as if it had come from a standard instrument, having H_s as its frequency response, by direct filtering with a software filter function having a frequency response of $1/H_{edi}$. When the measured signal is step-like, it is first converted to a pulse by differentiation, then filtered with $1/H_{edi}$, and finally converted back to an enhanced step-like signal by

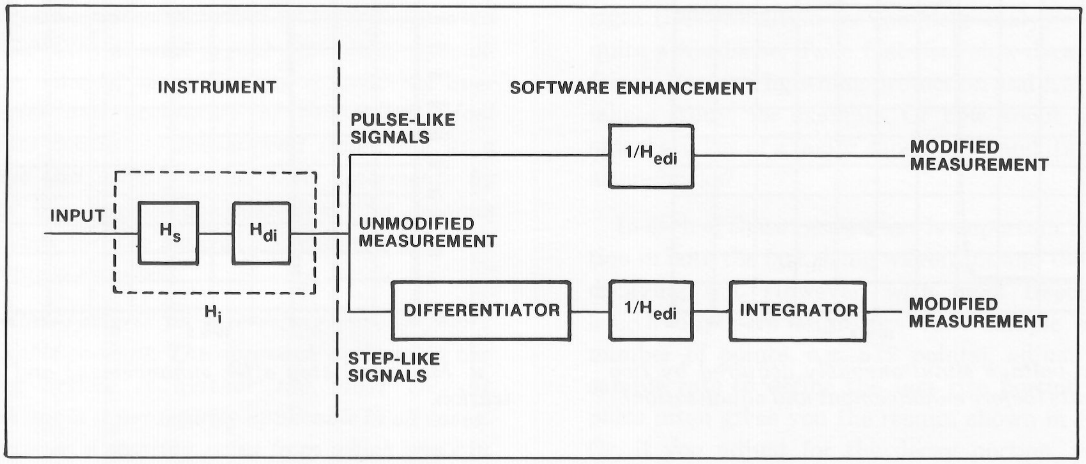


Fig. 2. Bandwidth enhancement and equalization procedure for TEKTRONIX R7912 Transient Digitizers and 7912AD Programmable Digitizers.

integration. Some sample results, consisting of **before** and **after** measurements with their corresponding amplitude and phase spectra, are shown in Fig. 3.

All things being equal

Of particular importance in Fig. 3 is the fact that measurements from two instruments are displayed. The step voltage measured by each is from the same source and measured simultaneously, so differences seen in the **before** measurements (Fig. 3a for example) come from instrument differences. Notice, also, that the instrument differences seen in Fig. 3a are not due to passband variations, but to differences existing beyond the passband, in the roll-off region. This is shown more clearly in the **before** amplitude spectra in Fig. 3b. There, the measured step's spectral estimates remain as one up to 500 megahertz. From that point on, the spectra vary according to the roll-offs of the individual instruments.

Notice in the **after** portion of Fig. 3b that not only has the measurement bandwidth of each instrument been increased, but the instruments have also been equalized. That is, when each instrument's correction filter is applied to the measurements, instrument deviations from the selected standard, H_s, are reduced. As a result, measurements made by different instruments appear almost as if they had been made by a single, "standard" instrument (Fig. 3d for example).

This equalization feature is important when you must deal with multiple instrument systems that are measuring signals having possible components above the specified instrument bandwidth. If your instruments don't look the same, then measurement ambiguities can result. Remember, though, this applies only to cases where you are forced to exceed bandwidth specifications. Properly designed instruments have uniform responses within their passbands and need equalization only when the bandwidth is exceeded.

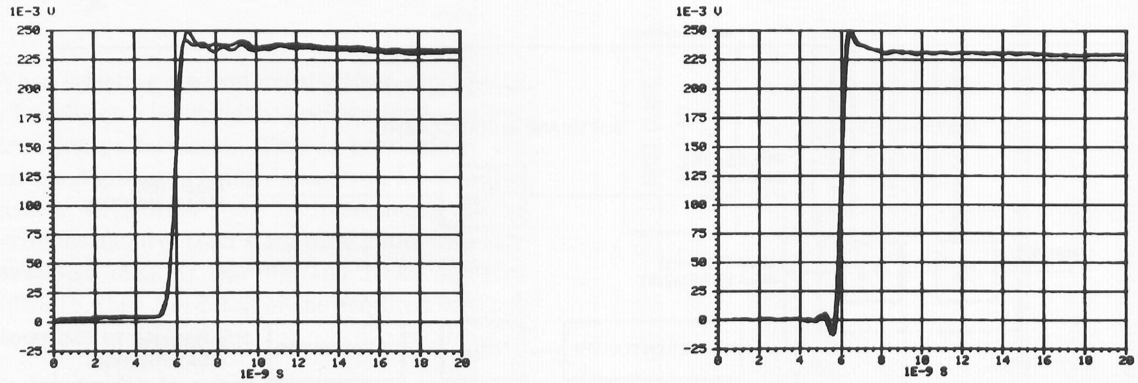
Some things to think about

Before you try to get results such as shown in Fig. 3 from your digitizer, there are some aspects of the instruments and process that you should know about.

First of all, the digitizer that you plan to enhance must be of a type that oversamples. This is true of both the TEKTRONIX R7912 and 7912AD. This oversampling pushes the Nyquist frequency well beyond the instrument's bandwidth limit. This allows instrument roll-off to be measured, as shown in Fig. 1. If the digitizer doesn't oversample, then you cannot define its roll-off and create the correction filter.

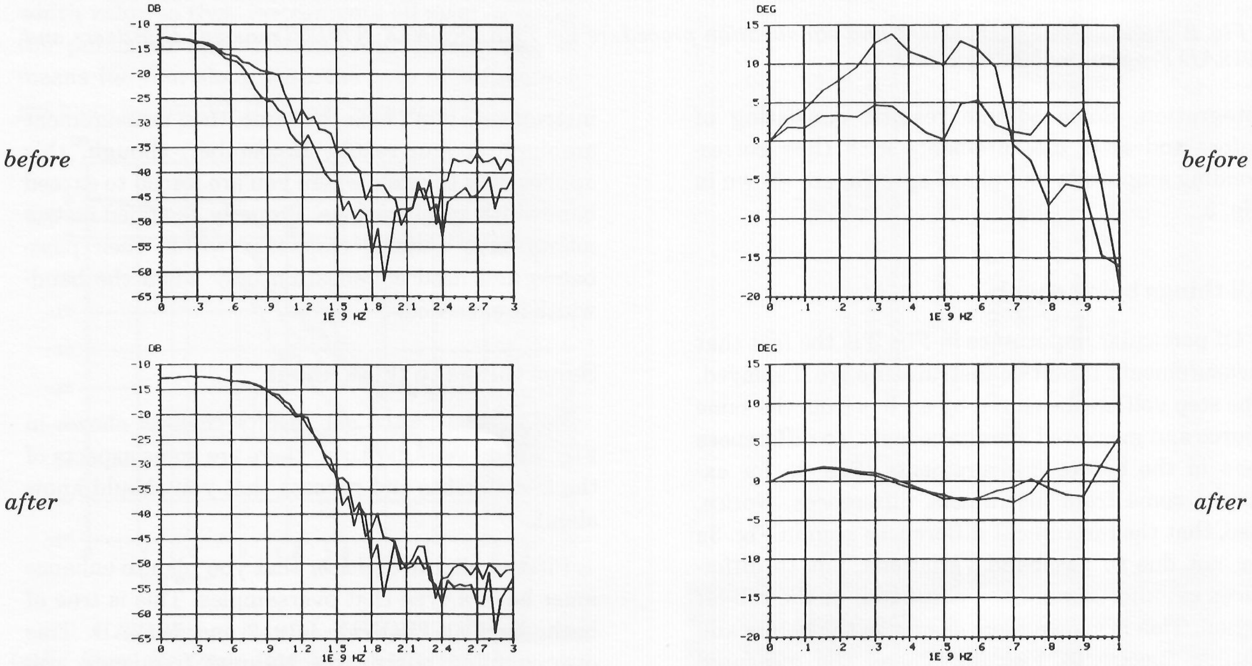
Also, the process blocked out in Fig. 2 is for transient signals only. Repetitive signals that do not reach or return to a steady state within the digitizing window, such as sinusoids, require additional processing to reduce time truncation and other endpoint problems.

Another important point to consider is the type of standard response, H_s, to construct for your instruments. Two possibilities are shown in Fig. 1. The first pushes bandwidth out but has a steep roll-off. The other passes through the instrument's specified 3-dB point and has a much gentler roll-off. With the steeper roll-off, you get extended bandwidth, but



a. Step voltage simultaneously acquired by two instruments (before enhancement and equalization).

d. Measured step after enhancement and equalization.



b. Amplitude spectra of step before and after enhancement equalization.

c. Phase spectra of step before and after enhancement and equalization.

Fig. 3. Before and after waveforms showing the effects of bandwidth enhancement and equalization on a measurement exceeding the instrument bandwidth.

your transient response is subject to more preshoot and overshoot. By sacrificing a little bandwidth, the roll-off can be slower and a better transient response is obtained.

In a bandwidth enhancement program developed at Tektronix, Inc., both types of standard responses were included for flexibility and convenience. If you would like to see how this was done, a copy of the program and the documentation explaining many of the procedural details is available. To get your copy, consult the "Signal Processing System users' application program library" column in this issue of HANDSHAKE. Ordering information is given there, and the program abstract is given under the title of "System Frequency Response Compensation."

Condensed by HANDSHAKE Staff
from program documentation by Paul McClellan,
Signal Analysis Group, Tektronix, Inc.

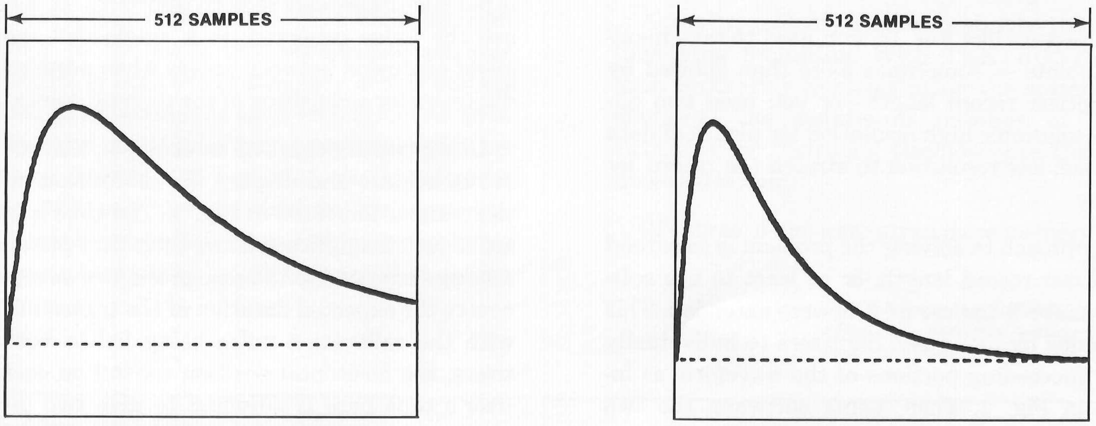
Time-Tying: Rx for short record lengths

Editor's Note: After receiving information for the following article, the HANDSHAKE staff discussed time-tying with several signal processing specialists. Surprisingly, there is some controversy over the merits and accuracies of time-tying. Most specialists cautiously recommend time-tying as a technique and usually hedge their statements by pointing out special considerations for various signal types and the general uncertainties that can be introduced by noise.

Still, time-tying is an interesting and, in many cases, usable concept. The approach outlined in the following article is neither the most elegant approach nor is it necessarily applicable in all cases. It is, however, a starting point from which you can develop specialized approaches reflecting your own needs. Or, as has happened in several instances, it may be sufficient as it stands.

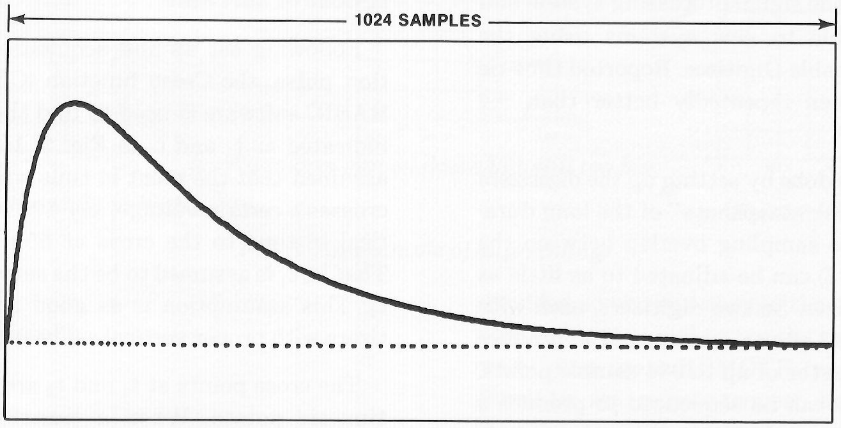
Digitizing certain types of waveforms with the right time resolution for complete analysis can be quite a headache. Take fast-rise, slow-decay waveforms, the kind lightning-protection and EMP specialists study, for example. Or how about seismological studies of a major shock wave and its trailing after-shocks?

In each of these cases, there is important information in both the fast rising wavefront and the slowly decaying tail. However, with most fixed-record-length digitizers (digitizers that sample a fixed number of points, e.g. 512 points), adjusting the sample rate to define the fast rise portion of the pulse often gives you the results shown in Fig. 1a. Or, if you adjust for the decay portion, you get something similar to Fig. 1b. While all along, Fig. 1c is what you'd like to get.



a. A fixed record length with the sampling rate set for wavefront definition. Decay information is truncated.

b. A fixed record length with the sampling rate adjusted to capture decay information. Wavefront resolution is lost.



c. A double-length record allows the sampling rate to be set for good definition of both the wavefront and the decaying tail.

Fig. 1. Digitizer record length is sort of like computer memory — no matter how much you have, you'll need more sooner or later. With digitizers, transient waveforms and random series are the major signal types that sometimes require expanded record lengths.

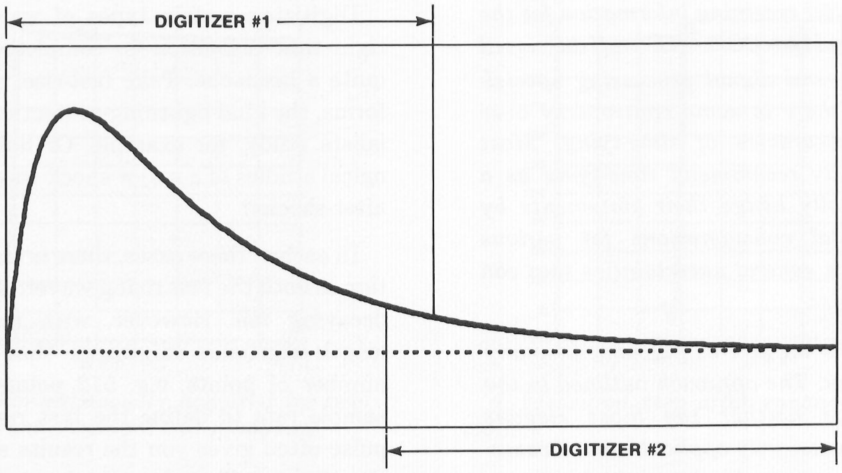


Fig. 2. Record length can be nearly doubled by using two digitizers to acquire successive portions of the waveform. More digitizers can be used to extend the record length.

Take two digitizers

For a picture like Fig. 1c, you need to take many sample points — sometimes more than allowed by your digitizer record length—or you need two different resolutions: high resolution for plenty of data on the rise; low resolution to stretch the record for the decay.

One approach to solving the problem is to extend the digitizer record length, or at least to use software to make it appear as if it were extended. This can be done by using two digitizers to individually acquire succeeding portions of the waveform as indicated in Fig. 2. Then, using software, the two record lengths can be time-tied to produce a near double-length record of the waveform. This approach has been successfully tried with two TEKTRONIX R7912 Transient Digitizers operating off the same signal processing system and is equally applicable to new systems using the 7912AD Programmable Digitizer. Reported time-tie accuracies have been repeatedly better than ±2 sample points.

The time-tying is done by setting up the digitizers to capture sequential "snapshots" of the long duration waveform. The sampling overlap between the digitizers (see Fig. 2) can be adjusted to as little as 20 points per digitizer. So two digitizers, each with record lengths of 512 points, can be used to produce time-tied record lengths of up to 984 sample points. And four digitizers can be sequenced to produce a time-tied record of around 1900 sample points.

The digitizers must be equipped with time base plug-ins having the X10 sweep magnification feature. Also, a PG502 250 MHz Pulse Generator is required to supply the calibration pulse. This pulse is fed simultaneously to both digitizers. It is best to use the pulse generator in a single-shot mode to avoid confusion between pulses when adjusting the digitizers for acquisition of the time-tie points.

Using two digitizers as an example, they are both set to acquire and display the calibration pulse in the manner indicated in Fig. 3. The time bases are set to X10 magnification and their time-per-division settings adjusted so the sum of the two sweep times covers the expected duration of the transient. Then, with the calibration pulse being fed to both digitizers, the horizontal position control on each digitizer's time base is adjusted to give two displays similar to those indicated in Fig. 3. (Note: With the time bases operating in X10 magnification, there is great latitude in horizontal position adjustment.)

A dose of software

Following set up and acquisition of the calibration pulse, the Cross function (CRS) of TEK SPS BASIC software is used to find the crossing points indicated as t_1 and t_2 in Fig. 3. In doing this, it is assumed that the point in time where the fall of #1 crosses a certain voltage, say 500 millivolts, is identical in time to the cross at 500 millivolts on #2. That is, t_1 is assumed to be the same point in time as t_2. This assumption is as good as fact when digitizers with proper vertical calibration are used.

The cross points at t_1 and t_2 are then used as the time-tie points. When a transient waveform is acquired with the first half from digitizer #1 and the second half from digitizer #2, the resulting arrays of stored data can be tied togeter in one longer array having the same resolution as either of the shorter arrays.

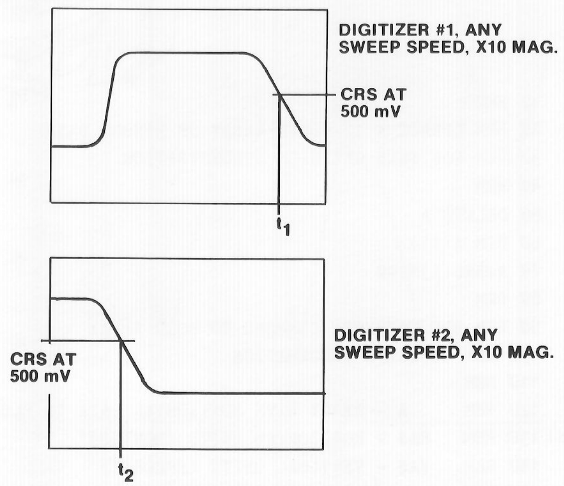


Fig. 3. Calibration pulse positioning used in setting up and selecting time-tie points.

Normally, time-tying is done with both digitizers operating on the same time-per-division settings. This results in both the preliminary arrays and the time-tied array having the same resolution. As an interesting and sometimes useful variation, digitizer #1 can be operated on a shorter time-per-division setting than used for digitizer #2. This provides greater resolution on fast rises, and lower resolution on decays where record length is generally more important than time resolution. Of course, such resolution changes must be taken into account or at least kept track of if the time-tied array is to be processed further.

And watch your calibration

Limitations to the time-tying technique are few, and most can be taken care of either through hardware calibration or software compensation. Having the digitizers matched both vertically and horizontally is extremely important and is usually accomplished through careful instrument calibration. However, digitizer frequency response at the high end can only be adjusted to a certain point. Any remaining variation between digitizers shows up there as amplitude errors in the time-tied results, but most variations can be equalized by software. An approach to such bandwidth equalization is discussed elsewhere in this issue of HANDSHAKE.

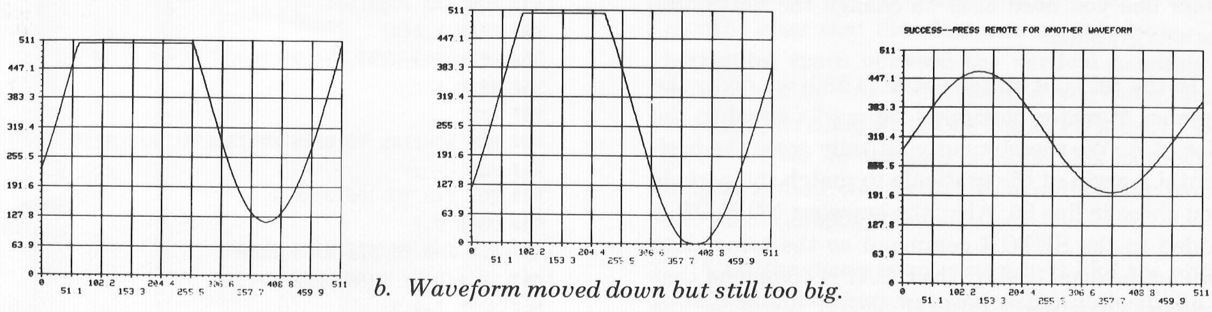
Actually, memory size is probably the greatest time-tying limitation. With TEKTRONIX Waveform Processing Systems using the normal 28K of memory, the maximum number of R7912s, 7912ADs, or Digitizing Oscilloscopes that can be time-tied is four.

Of course, if one were given more memory . . .

By Bob Ramirez, HANDSHAKE Staff, based on a Field Memo from Dean Turnbaugh, Tektronix Field Office, Rockville, MD.

---

continued from page 7



a. Waveform too big and too high.

b. Waveform moved down but still too big.

c. Final sizing and positioning.

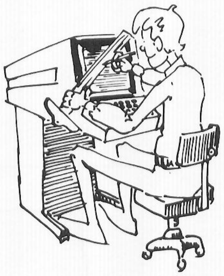
Fig. 3. An example of autoscaling.

provided so you can change the volts/division, position, or input signal to try the routine again. When ready, just press REMOTE on the 7912AD front panel to set remote-request status. To be fair, though, give the instrument and software something to work with — some part of the waveform must be visible on the TV monitor.

By Jim Kimball, HANDSHAKE Staff.

Getting the most out of TEK BASIC graphics

Build bars — to see better



It can be frustrating to gather information, arrange it in a meaningful manner for one of those inevitable reports that attempts to communicate what is being done and how well it's being done, and have the mechanics of the presentation — the graphics — hinder instead of enhance the message. Bars — bar graphs, that is — are tools you can use to add clarity to your reports.

Graphs, generally, are effective tools for communications. That's why you see them used so often. But, for certain information, some graphs work better than others. The line graph you get when you use the GRAPH command may not always be the best presentation of your data. Sometimes, a bar graph works better.

The program listing in Fig. 1 contains some easy-to-use routines that will give you the bar graphs shown in Figs. 3, 4, or 5. You pick the graph style that best suits your information, and include those routines that will do the job for you.

The graphs included as examples here use "month" as the horizontal units. We must first, then, dimension the data array to the number of periods (months) we want to show. Since array elements are numbered from zero, dimensioning the array to 11 gives us 12 periods, or 12 months on the graph. If you want a 24 month scale graphed, as in Fig. 5, dimension A to 23. Line 60 is the only program line you need alter to change the horizontal periods graphed.

In the SETGR statement (line 280) we make the number of major horizontal tic marks equal to the size of A. We thereby automatically scale the horizontal margin of the graticule to match the periods you chose in line 60. Also, the keyword NOPLOT is added to the SETGR command so the array data will not be displayed by the GRAPH command that follows. Omitting NOPLOT would result in the normal line graph of the data shown in Fig. 2.

The three routines included in Fig. 1 can be retained, deleted, or altered at your option.

The first routine draws the outline bar graph (Figs. 3 and 4). Line 340 moves the cursor to the left-hand corner of the graticule where we find the zero X value and the first Y value of the array data. MOVE was used instead of SMOVE because MOVE employs the user units per the current window; SMOVE uses screen coordinates. Line 360 draws the vertical lines between the months, con-

```
10 REM
20 REM CREATE A 12-MONTH ARRAY OF RANDOM DATA
30 REM FOR THIS GRAPHICS DEMONSTRATION
40 REM
50 DELETE A
60 DIM A(11)
70 A=RND(A)*100
80 REM
90 REM USE WAVEFORM COMMAND TO HOLD ARRAY
100 REM LABELING INFORMATION
110 REM
120 REM     A - ARRAY WITH HORIZONTAL DATA TO PLOT
130 REM   HA$ - HORIZONTAL UNITS (MONTHS)
140 REM   VA$ - VERTICAL UNITS (PERCENT)
150 REM
160 WAVEFORM WA IS A,SA,HA$,VA$
170 SA=1
180 HA$="MONTH"
190 VA$="PERCENT"
200 GOSUB 250
210 END
220 REM
230 REM ERASE 4010/4014 SCREEN AND DRAW GRATICULE
240 REM
250 PAGE
260 REM
270 WINDOW 0,SIZ(A),0,100
280 SETGR GRAT 2,2.NOPLOT,WIND,TICS SIZ(A),5,2,2
290 GRAPH WA
300 REM
310 REM ROUTINE TO MOVE TO FIRST ARRAY LEVEL
320 REM AND DRAW OUTLINE BAR GRAPH
330 REM
340 MOVE 0,A(0)
350 FOR X=0 TO SIZ(A)-1
360 DRAW X,A(X)
370 DRAW X+1,A(X),X+1,0
380 NEXT X
390 REM
400 REM ROUTINE TO DRAW HORIZONTAL HATCHING
410 REM
420 FOR Y=2 TO 100 STEP 2
430 MOVE 0,Y
440 FOR X=0 TO SIZ(A)-1 STEP 2
450 IF A(X)>Y THEN DRAW X+1,Y
460 MOVE X+2,Y
470 NEXT X
480 NEXT Y
490 REM
500 REM ROUTINE TO DRAW VERTICAL HATCHING
510 REM
520 FOR X=1 TO SIZ(A) 1 STEP 2
530 FOR I=.2 TO .8 STEP .2
540 MOVE X+I,0
550 DRAW X+I,A(X)
560 NEXT I
570 NEXT X
580 RETURN
```

Fig. 1. Bar graph program listing.

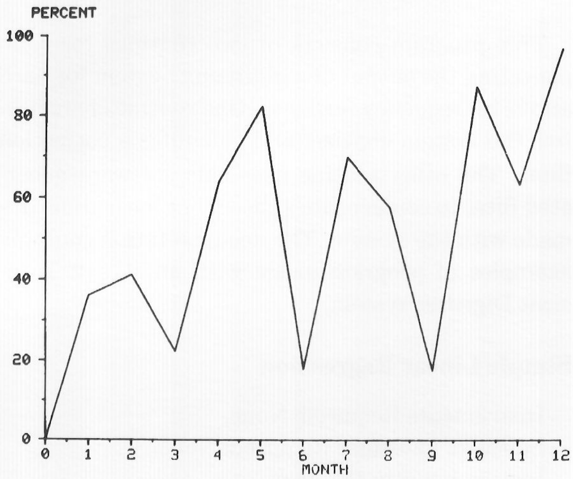


Fig. 2. Normal line graph.

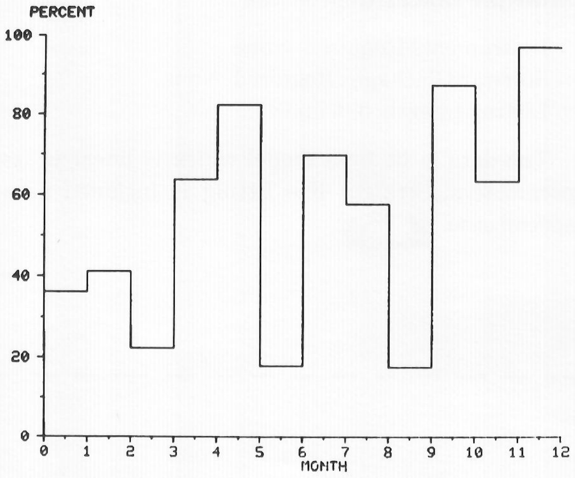


Fig. 3. Outline bar graph; one style.

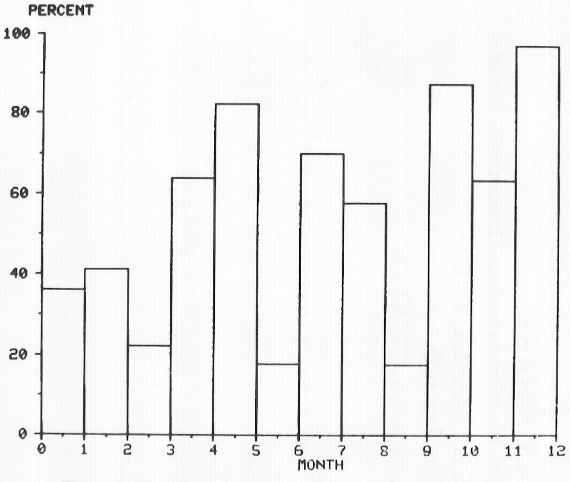


Fig. 4. Outline bar graph; another style.

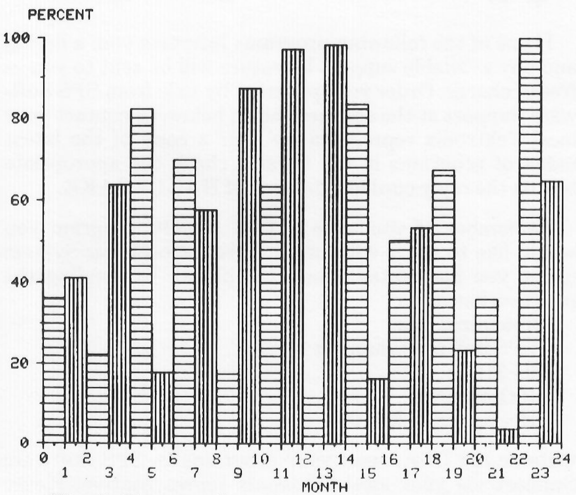


Fig. 5. Bar graph with contrasting hatching.

necting the varying Y values (Fig. 3). The vertical lines that drop to the bottom of the graph (Y=0) are created by the ",X+1,0" at the end of line 370 (Fig. 4). The first pair of coordinates in line 370 draw the horizontal lines (top of the bars). The bar graph in Fig. 5 requires routines 2 and 3 to draw the hatching within the bars.

The second routine draws the horizontal hatching, beginning with the bar for the first month and for every other month after that (line 440 sets up this loop). Line 420 determines the incremental spacing of the hatching. If you want the horizontal hatching lines closer together, change line 420 to FOR Y=1 TO 100.

The third routine draws the vertical hatching, beginning with the second month and for alternate months after that (line 520 sets up this loop). The horizontal space between the vertical hatching is determined by line 530. Again, if you want closer spacing, you could write this line as FOR I=.1 TO .9 STEP .1. We do not step through this loop beginning at zero or ending at 1 to avoid writing over a line that is already there.

If, on the other hand, you would like the entire area within the bars hatched either horizontally or vertically instead of the alternate patterns in Fig. 5, this is easily done by first deleting the routine you do not want to use, then by changing the following program lines:

All vertical hatching: make line 520 FOR X=0 TO SIZ (A)−1

All horizontal hatching: make line 440 FOR X=0 TO SIZ (A)−1, and make line 460 MOVE X+1, Y

By Walt Robatzek,
HANDSHAKE Staff
With significant contributions by David Stubbs, SPS Software Engineer and by Joyce Ferriss, HANDSHAKE Staff.

Signal Processing Systems users' application program library

If one of the following programs interests you, a listing and any available support literature will be sent to you — free of charge. Order your program by title from SPS Software Support at the address shown below, or contact your local Tektronix representative. For a copy of the latest index of programs in the library, check the appropriate box on the reply card in this issue of HANDSHAKE.

Remember, if you have a TEK BASIC program you would like to share with other Signal Processing System users, you may enter it into the library by sending the program listing to:

Tektronix, Inc.
SPS Software Support
94-319
P.O. Box 500
Beaverton, OR 97077

Outside the USA, send your programs to SPS Software Support via your local Tektronix representative. Please include with your program a short description of what it does and how it does it. We would also like to know about any special data conditions, instruments, or software package requirements. The memory requirements for running the program would also be very helpful.

TEK SPS BASIC PROGRAM ABSTRACTS

System Frequency Response Compensation

Instruments Required: Stored instrument data is required, and recommendations for acquiring that data are given in the documentation.

Software Packages Required: Signal Processing.

Listing Length: 630 lines.

This program consists of two routines for compensating the output of a digitizing system for variations in frequency response. One routine characterizes the system digitizer and generates a correction filter. The other routine uses this software generated filter to compensate subsequent measurements made with the system. The documentation contains examples of program usage with an R7912 Transient Digitizer system.

Simple Linear Regression

Instruments Required: None.
Software Packages Required: None.
Listing Length: 187 lines.

This program is a simple linear regression that fits a straight line to the data.

Multiple Linear Regression

Instruments Required: None.
Software Packages Required: None.
Listing Length: 336 lines.

The data is fit by a model which is linear in its parameters. Straight line fitting is included as a special case.

Edited by Walt Robatzek Graphics by Bernard Chalumeau

HANDSHAKE is published quarterly by the SPS Documentation Group. Permission to reprint material in this publication may be obtained by writing to:

HANDSHAKE Editor
Group 157 (94-384)
Tektronix, Inc.
P.O. Box 500
Beaverton, Oregon 97077

HANDSHAKE is provided free of charge by Tektronix, Inc., as a forum for people interested in digital signal processing. As a free forum, statements and opinions of the authors should be taken as just that — statements and opinions of the individual authors. Material published in HANDSHAKE should not be taken as or interpreted as statement of Tektronix policy or opinion unless specifically stated to be such.

Also, neither HANDSHAKE nor Tektronix, Inc., can be held responsible for errors in HANDSHAKE or the effects of these errors. The material in HANDSHAKE comes from a wide variety of sources, and, although the material is edited and believed to be correct, the accuracy of the source material cannot be fully guaranteed. Nor can HANDSHAKE or Tektronix, Inc., make guarantees against typographical or human errors, and accordingly no responsibility is assumed to any person using the material published in HANDSHAKE.

AX 4088

HANDSHAKE
SPS Documentation Group MS 94-384
TEKTRONIX, INC.
P.O. Box 500
Beaverton, OR 97005