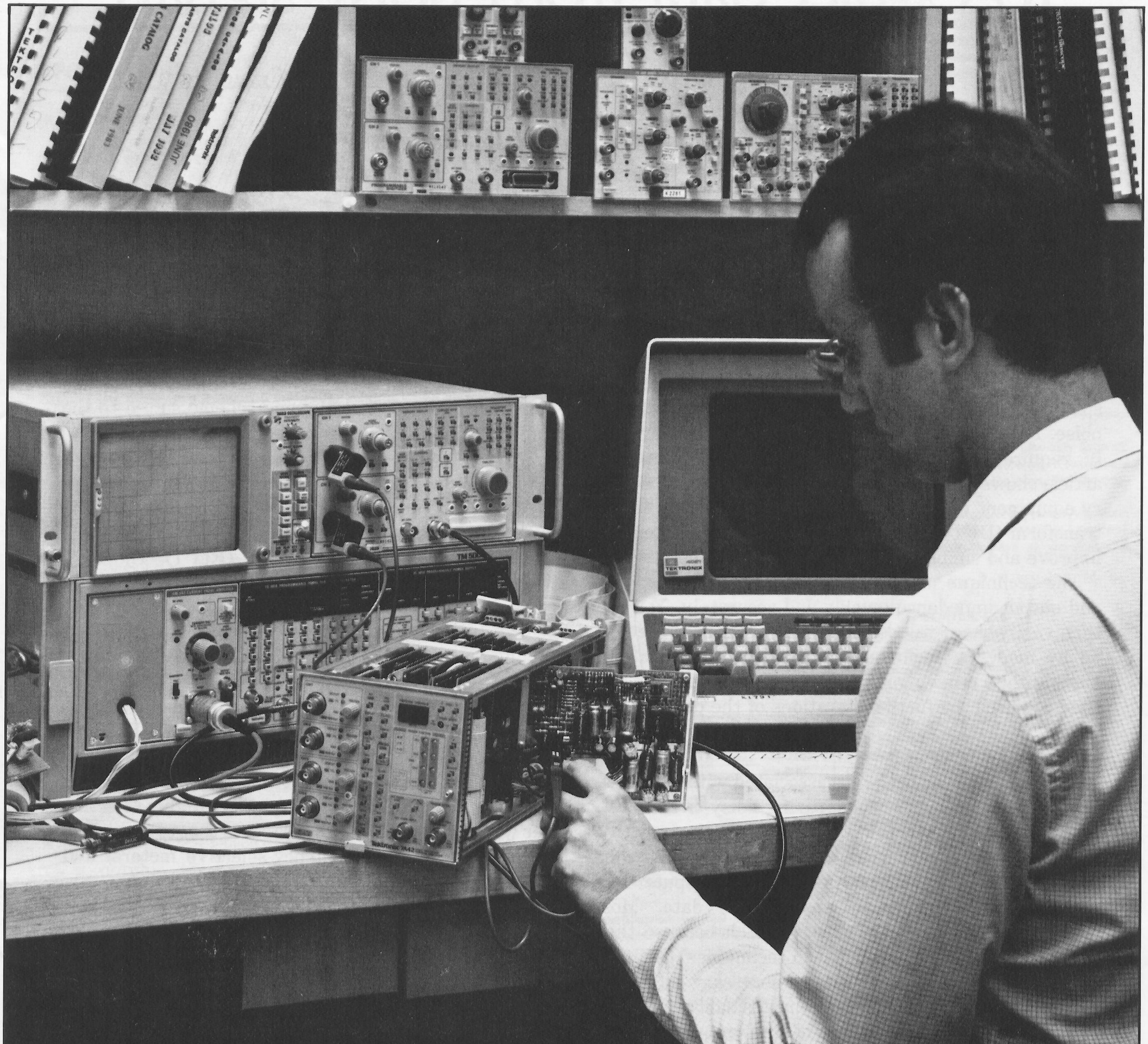
April 5, 1984

A CAHNERS PUBLICATION

EDN

EXCLUSIVELY FOR DESIGNERS AND DESIGN MANAGERS IN ELECTRONICS

SPECIAL ISSUE: TEST & MEASUREMENT



Use a personal computer and DFT to extract data from noisy signals

Evaluating a power-supply signal buried beneath 20 dB of noise is difficult, particularly when you don't have a tracking voltmeter. With a digital storage scope and a personal computer, you can make these sensitive measurements.

Clark Foley *and* **Jim Lamb,** *Tektronix, Inc*

When characterizing a switching power supply's design, you generally examine its loop gain and output impedance. To do so, though, you need to measure small signal variations buried in high levels of switcher noise. Detecting and measuring these signals previously required expensive special equipment, but this article shows how combining general-purpose laboratory equipment with a relatively simple discrete Fourier transform (DFT) running on a personal computer can improve and simplify the supplies' design and test.

The technique measures loop gain, phase response and output impedance quickly and accurately. You can then verify these values against the responses predicted by modeling techniques and software simulators. Having thus proven your models, you can shorten the time required for further iterations of the design.

Filtering eases design verification

You can avoid the problems associated with traditional equipment (see **box,** "Comparing noisy-circuit testing techniques") by making your measurements with a waveform digitizer or digital storage oscilloscope (DSO), feeding the waveform data to a computer and using digital filtering to analyze the data. Most engineers shy away from this approach because they find it difficult to determine such techniques' performance limitations and to create the necessary software.

Part of the problem is that most small computers' BASIC versions lack many of the required math functions, such as complex operations. Consequently, engineers must resort to more primitive functions and algorithms, which take more time to design and execute. The approach presented here, however, illustrates a straightforward technique that uses the lab equipment and a digital-filtering scheme suited to the computational capabilities and memory size of small computers.

Before moving on to implementation details, however, you must understand the fundamentals of the DFT—one of the most basic, yet poorly understood, frequency-transforming techniques **(Ref 1).** The DFT is especially useful for switching-power-supply measurements because you can extract information on just one frequency from a time-domain signal even if its remaining components have substantially higher energy. For example, consider the case in which you're looking for an injected test signal in a feedback loop with a signal-to-noise (S/N) ratio of 0.1 (−20 dB). Such a signal is hardly visible on an oscilloscope.

The DFT allows a computer to exclude all frequencies except the one of interest; it serves as a highly selective filter with performance comparable to that of expensive tracking voltmeters (frequency-selective meters) and impedance analyzers.

Why not use the more familiar FFT for this application? True, the FFT reduces the number of complex multiplications needed to yield a complete spectrum, but the algorithms aren't trivial; most people buy FFT programs rather than write them. Also, the FFT yields spectra at equal and linear increments from

High noise in switchers makes signal detection tough

Comparing noisy-circuit testing techniques

To find the frequency response of a circuit, designers have traditionally used an oscilloscope and sine-wave generator to measure gain and phase. Sometimes they also used a counter/timer to make phase measurements. This simple, straightforward approach is still valid for many applications.

In operation, to determine gain at a given frequency you simply inject a sine wave of the desired frequency and amplitude into the circuit and view that signal at two different points in the loop **(Fig A).** With an oscilloscope, you can easily measure the two signals' amplitudes and compute their ratio. To find phase, simply adjust the display of both signals for equal height and then examine the time difference between points on each waveform where they cross a common reference.

You can also use a counter/timer to compute automatically the phase difference once you've selected a crossing reference. In the presence of uncorrelated noise, however, this procedure begins to degrade and provide less accurate results. In fact, this technique is valid only as long as the S/N ratio is much greater than 10 (20 dB).

Even with an S/N ratio of approximately 10, you need some way to eliminate or substantially reduce noise. One aid is either a variable-persistence, bistable or digital storage scope with signal averaging. When using a counter/timer, you'll need adjustable trigger hysteresis and time-interval averaging. Of course, some form of signal filtering will also reduce the noise effects, but beware of introducing gain or phase errors over the measured spectrum. A spectrum analyzer combines filtering and signal stimulus, and its display of amplitude vs frequency presents perhaps the best gain information of the methods mentioned thus far. It can't give phase information, however.

What happens if the S/N ratio is extremely poor? With a ratio much less than unity (0 dB or

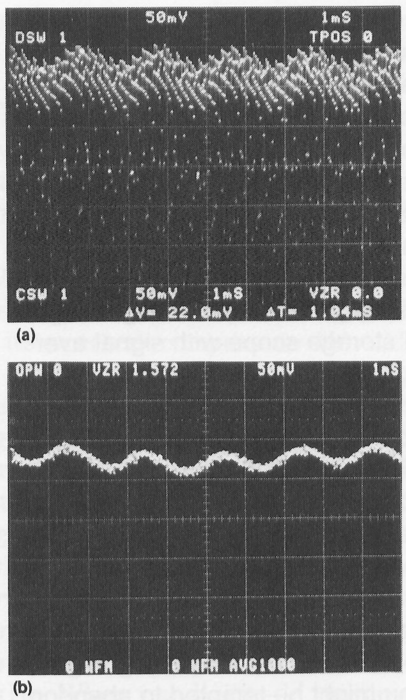
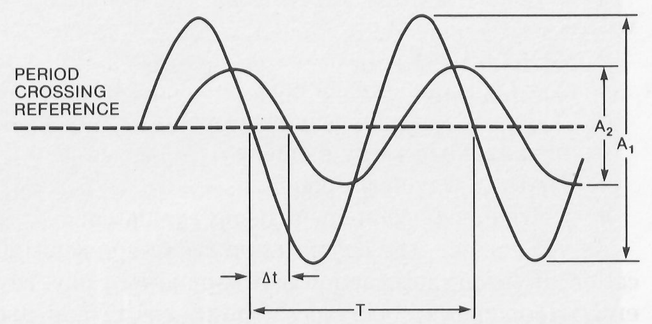


Fig B—Try to dig the signal from the noise in this trace (a), *which is typical of a sine wave plus some switching noise in the feedback loop of a switching power supply. Obviously, traditional methods such as that in* ***Fig A*** *won't work. When you average this signal 1000 times, you obtain the waveform in* ***(b),*** *which gives you more than 10% measurement error for magnitude and very poor phase information.*



$$\text{GAIN} = \frac{A_1}{A_2}$$

$$\text{PHASE} = \left(\frac{\Delta t}{T}\right) 360°$$

Fig A—Using an oscilloscope and function generator, *you can determine the characteristics of many circuits. After injecting a test signal and viewing it at a different point, you can read gain and phase changes directly from the scope's screen. Try this traditional method only when the S/N ratio is very high.*

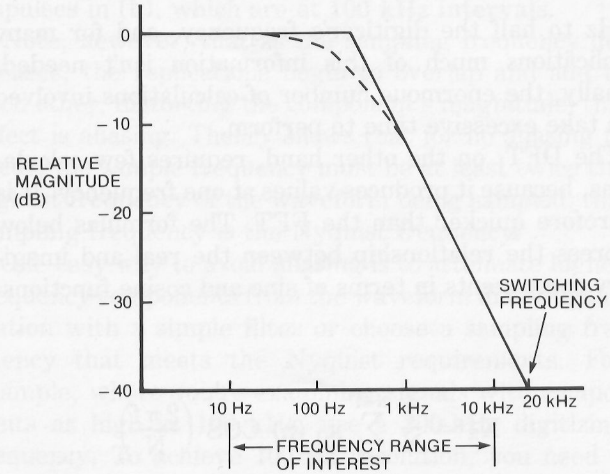


Fig C—You can try to filter out high-frequency noise *to improve the S/N ratio, but a simple filter can also detrimentally affect part of the spectrum you want to investigate. Here you see how a 2-pole filter will attenuate desired signals that are still significantly lower than a power supply's 20-kHz switching frequency.*

worse), as in switching power supplies using sampled-data feedack techniques, attempting to measure gain and phase with these approaches is an exercise in futility. What happens, for instance, if you're trying to measure the amplitude of a signal with an S/N ratio of 0.1 (−20 dB) (**Fig B,** part **(a)**)? Without filtering, a digital storage scope with signal averaging would have to average 1000 times the trace shown in **(b)** to approach an S/N ratio of 3 (9.5 dB) and 10,000 times to reach 10 (20 dB). Even then you can make measurements with only approximately 10% error. At best, these are marginal results that can take from 20 minutes to several hours to attain. After examining **Fig B,** you might be tempted to abandon these approaches.

Thus, filtering deserves further investigation. For signals with an S/N ratio much less than unity, the simple 1- or 2-pole filter commonly included in most instruments won't provide significant rejection without serious effects over the measured spectrum. For example, **Fig C** shows that such filters can affect two-thirds of the desired spectrum. At first glance, these effects might not seem significant, because you're interested in relative magnitude and phase differences, and the filter's influences cancel out. Although this is true, the filter will greatly attenuate the signals you're looking for, thereby reducing the S/N ratio, again producing an unacceptable noise level.

Obviously, you need a more sophisticated filtering method—one that's more selective with the same or better rejection, but with less influence on the spectrum of interest. An adjustable and steep roll-off bandpass or notch filter performs nicely. However, you'll have trouble finding such filters in most test and measurement instruments. In fact, very few instrumentation companies provide such filters, which perhaps indicates the limited number of applications for such low-frequency filters.

Another useful type of filter comes equipped with a voltmeter and is often referred to as a frequency-tracking voltmeter. When combined with a counter/timer, the instrument can easily gather accurate magnitude and phase information. The most advanced form of this combination is the impedance analyzer. Although quite different internally, impedance analyzers and frequency-tracking voltmeters use the same concepts and have the same limitations.

These limitations aren't in the equipment's ability to measure gain and phase; either unit provides the best approach for these types of measurements. Instead, impedance analyzers and frequency-tracking voltmeters are not general-purpose test instruments and are rarely found in most test environments. Their infrequent use and high cost usually preclude their purchase.

0 Hz to half the digitizer's frequency, and for many applications much of this information isn't needed. Finally, the enormous number of calculations involved can take excessive time to perform.

The DFT, on the other hand, requires few calculations, because it produces values at one frequency. It is therefore quicker than the FFT. The formulas below express the relationship between the real and imaginary components in terms of sine and cosine functions:

$$X_R = \frac{1}{N}\sum_{n=0}^{N-1} x_O(n)\ \mathrm{COS}\left(\frac{2\pi f}{N}\right)$$

$$X_I = -\frac{1}{N}\sum_{n=0}^{N-1} x_O(n)\ \mathrm{SIN}\left(\frac{2\pi f}{N}\right)$$

$$\mathrm{MAGNITUDE} = \sqrt{X_R^2 + X_I^2}$$

$$\mathrm{ANGLE} = \mathrm{TAN}^{-1}(X_I/X_R),$$

where:

- X_R=DFT's real part
- X_I=DFT's imaginary part
- $X_O(n)$=time-series waveform value
- n=waveform array element
- N=total waveform points
- f=frequency component being evaluated.

As you can see, the computation reduces to multiplication, division, subtraction and summation, plus several trigonometric operators (sin(x), cos(x) and arctan(x)). You'll find these operations on almost all personal computers.

This algorithm operates on a series of digitized samples of the desired waveform. To ensure accurate results, keep this waveform data free of any aliased frequency components. Don't undersample the signal during digitization; the minimum sampling frequency should equal twice the highest frequency component composing the signal under examination. You can often ignore higher components of negligible amplitude when

Traditional filtering techniques often compromise your data

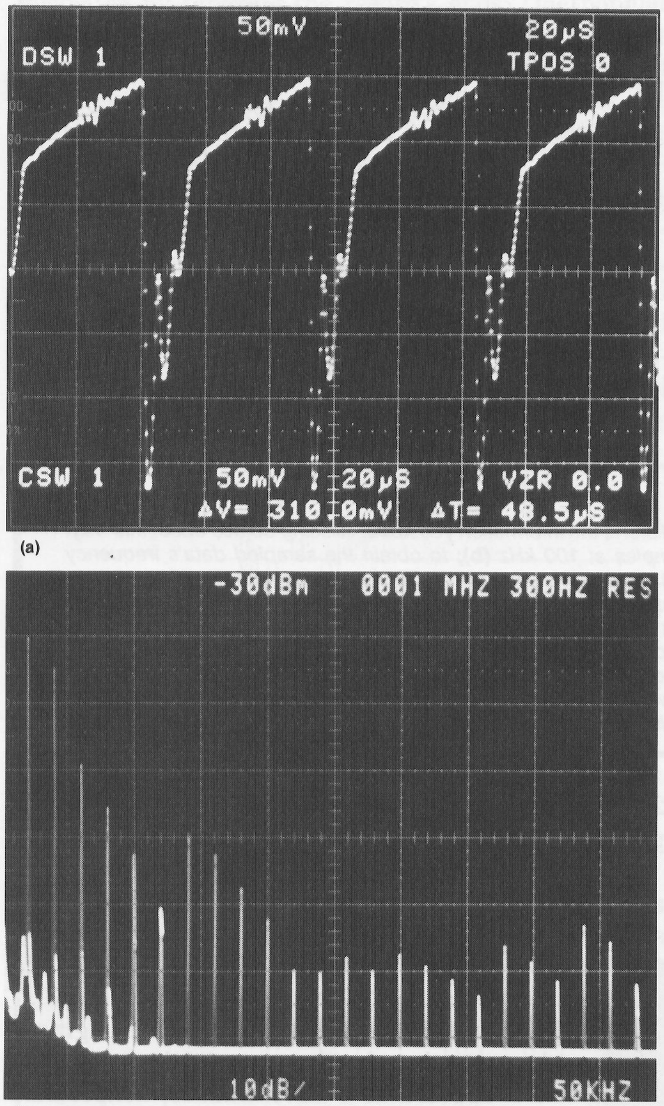
selecting the sample frequency, thereby reducing the lab equipment's performance requirements.

In setting up the DFT for this application, you can avoid several major problems by taking into account a few fundamentals of DFT theory. First, consider spectral resolution. To find the components of a specific frequency, you must ensure sufficient spectral resolution because the transform produces results only at discrete frequencies. Select the sampling speed and waveform-record length such that the increments between these discrete frequency values are small enough so that you don't lose vital data in the gaps.

The minimum frequency comes from the formula F_{MIN}=(1/sample interval)/(record length); thus, for example, if you acquire 1000 points in a waveform, each at a 10-μsec interval, the minimum frequency increment in the spectrum is expressed as follows:

$$F_{MIN}=1/10^{-5}\text{sec}/1000=100\text{ Hz}.$$

In this case, you can examine frequencies only in multiples of 100 Hz. If you're interested in frequencies lower than 100 Hz, you must either increase the record length or increase the sample interval.



(a)

(b)

Fig 1—A switching power supply's feedback signal (a) *consists of a series of pulses that produce a wide frequency spectrum* **(b).** *This signal, in the loop of a flyback converter, contains components above several hundred kilohertz. You can ignore these components because of their relatively small magnitude (less than −30 dB).*

To illustrate the importance of sampling speed, consider a supply with a 20-kHz switching frequency and note that this signal's components extend well beyond 20 kHz. **Fig 1** shows a signal in the loop of the flyback converter **(a)** and its spectral components **(b).** Although the highest frequency component is greater than several hundred kilohertz, those above 100 kHz have very small magnitudes and can be safely ignored.

In selecting your sampling speed, however, you should take care not to set that value so low that you produce aliasing. **Fig 2** shows how aliasing comes about: **(a)** shows 20-kHz switching noise (left) and its associated frequency spectrum (right). Because the time-domain signal is continuous and periodic, the frequency spectrum is discrete and its components occur at integer multiples of 20 kHz. When you digitize in the time domain, here with a sample rate of 100 kHz ((**b**), left), you essentially multiply the left sides of **(a)** and **(b)** to achieve the result on the left side of **(c)**, the digitized waveform. However, multiplication in one domain transforms to convolution in the other domain, so convolve the right sides of **(a)** and **(b)** to produce the digitized waveform's frequency spectrum on the right side of **(c).** As you can see, convolution has the effect of replicating the original spectrum of **(a)** about the impulses in **(b)**, which are at **100 kHz intervals.**

Note, however, that as the sampling frequency decreases, the replications begin to overlap and add to each other, distorting the component's magnitude; this effect is aliasing. Theory shows that for no aliasing to occur, the sample frequency must be at least twice the highest frequency of the waveform being sampled; this sampling frequency is the Nyquist frequency.

One easy way to avoid aliasing is to attenuate higher frequency components from the waveform under examination with a simple filter or choose a sampling frequency that meets the Nyquist requirements. For example, where you're examining signals with components as high as 100 kHz, use a 200-kHz digitizing frequency. To achieve 100-Hz resolution, you need a 2000-point waveform **sampled at 200 kHz. This resolution** allows you to examine the loop's frequency response from 100 Hz to 100 kHz in 100-Hz steps; the resolution proves more than adequate for measuring feedback stability.

Another conflict between DFT theory and reality

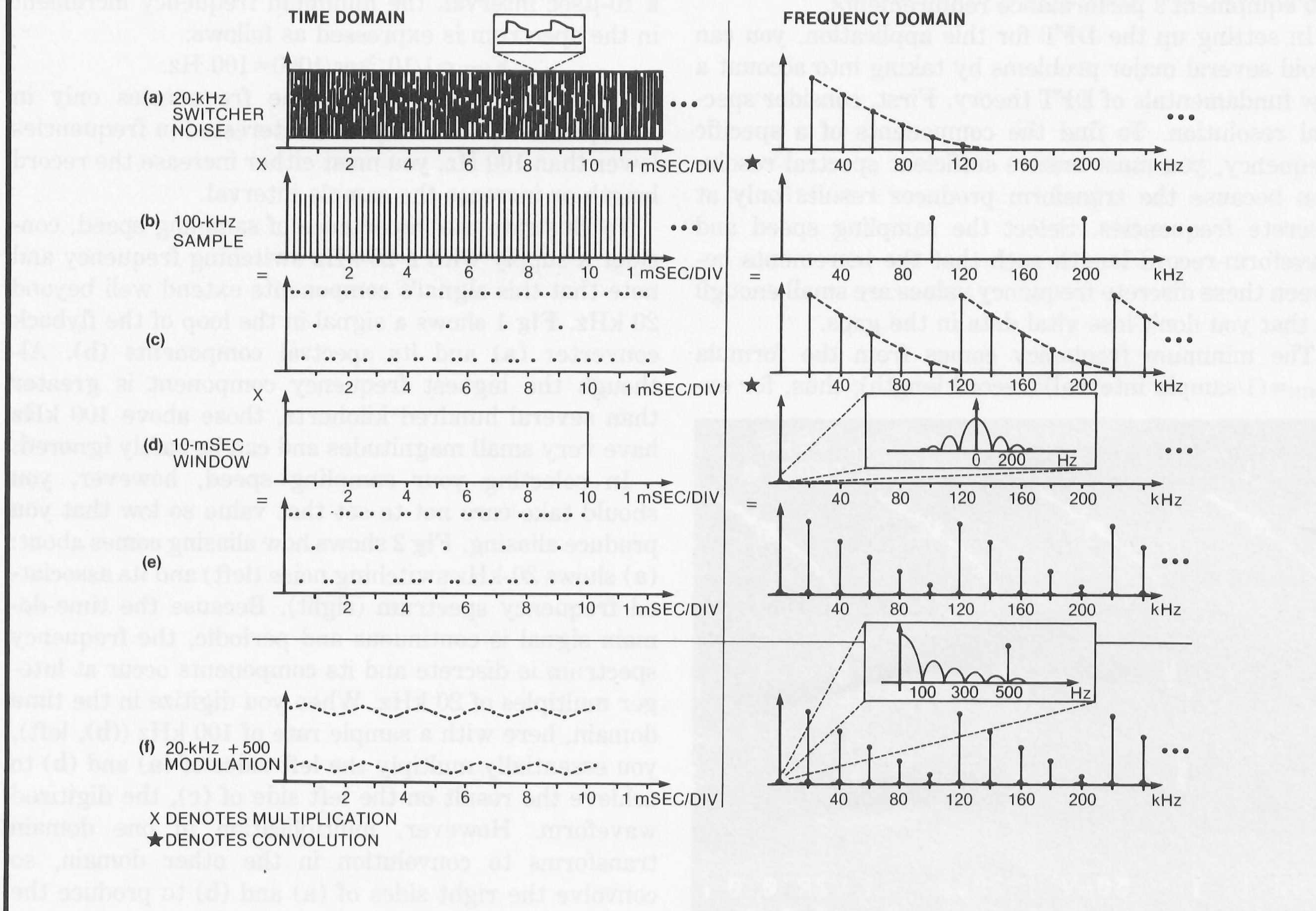


Fig 2—Sampling speeds that are too low cause aliasing, *which distorts the information you need. Aliasing comes about this way: Suppose you sample 20-kHz switcher noise* ***(a)*** *with discrete samples at 100 kHz* ***(b);*** *to obtain the sampled data's frequency spectrum* ***(c),*** *convolve the frequency-domain curves of* ***(a)*** *and* ***(b).*** *Note that convolution replicates the original signal's spectrum about impulses located at multiples of the sampling rate. Digitizers store only a finite number of samples, however, so you effectively multiply the sampled data by a finite window. To see what effect this has on the spectral data available for later analysis, first look at the window's spectrum* ***(d)*** *(the sin(x)/x, or sinc(x), function) and convolve that with the sampled data's spectrum. You'll get the spectrum in* ***(e),*** *which has a series of sinc(x) functions centered on impulses at 20-kHz intervals, the multiples of the switching speed. Modulating the 20-kHz switcher noise with a 500-Hz sine wave that has an integer number of cycles within the digitized record* ***(f)*** *produces spectral information that occurs at the zero values of the sinc(x) function, thus eliminating window errors.*

arises because your equipment can store only a finite number of samples; this limitation can create "windowing" errors. Examine the effect of windowing the infinite sample sequence in **Fig 2.** Think of the window as a rectangular pulse that you multiply with the infinite time sequence. The window itself ((**d**), left) has unit amplitude and is as long as the digitizer's record length (10 msec). The window's transform ((**d**), right) is a sin(x)/x=sinc(x) function, with zeros at integer multiples of 0.1 msec or 100 Hz. Multiplying the window with the infinite sample sequence in the time domain results in the 10-msec stored record on the left of (**e**). Convolving the window's spectrum (the sinc(x) function) with that of the infinite sample series then produces a repetitive sinc(x) function centered on impulses with 20-kHz intervals ((**e**), right)—a considerably different pattern from the transform of the infinite signal alone.

Your measurement task would be easier if you could analyze only continuous waveforms, but the power-supply application adds some complications. To characterize the supply's feedback loop, you need to modulate the feedback signal with an injected test signal at various frequencies. The waveform that the digitizer captures, however, is a short segment of the modulated signal, and it appears as a discontinuous, nonperiodic event with its own frequency spectrum (**Fig 3a**).

But because the DFT assumes that the data sequence represents a periodic sequence, unexpected results can occur if you don't pay attention to how the digitized record represents the true signal. Consider a sampled

Slow sampling can create errors caused by aliasing

0.25-Hz cosine **(Fig 3a),** which doesn't have an integer number of cycles within the digitized record. When replicated end to end, the waveform no longer appears as a continuous cosine **(Fig 3c).** With a record length of 10 sec **(Fig 3b,** left), only frequency components of 0.1-Hz(**$^1/_{10}$ sec**) multiples are reported. Thus, the actual 0.25-Hz signal isn't reported faithfully. **Fig 3d** illustrates how such a signal appears to the DFT based on its finite digitized representation. The discontinuity in this repetitive waveform creates unwanted frequency components.

There are several options for reducing or eliminating these errors. First, some applications allow you to reshape the time record to minimize the end-to-end mismatch by tapering the record's ends. One popular technique for this reshaping is to multiply the time record by a window with the shape of an inverted offset cosine wave **(Fig 4).** This multiplication alters the relationship between the higher frequency components and the low-frequency components.

To obtain the information you need from the waveform, you need its low-frequency components, so reshaping with inverted cosine waves won't work. There's another way, however, to alter the sampled information that minimizes erroneous information and yet maintains low-frequency information: Simply ensure that critical information (such as the injected signal you later measure to check loop response) always has an integer number of cycles within the waveform record's time frame **(Fig 5a).**

If you imagine that this signal repeats infinitely and its endpoints match, then it will appear continuous and have the correct period **(d),** and it's windowed transform will be evaluated at the zero values of the sinc(x) function **(c).** (To see the effects of injecting a sine wave into the power-supply loop, see **(f).**) This technique eliminates two key problems: leakage that distorts frequency components, and the need for special windows that require additional computations.

With this knowledge of the DFT and its subtleties, you're prepared to select the equipment needed to perform the tests, implement the DFT algorithm and evaluate the data. The basic operating principle is this: You inject a test signal into the power supply's feedback loop at the frequency you want to test, and then look at this signal after it's traveled through the loop. By doing this at various frequencies, you can compute the supply's **magnitude** and phase-response curves. Furthermore, you can check the unit's output-impedance characteristics. To perform these tests, you need a computer, a sine-wave or function generator, a waveform digitizer or digital storage scope, and some voltage and current probes.

To ensure that measured results are repeatable, try to automate the process wherever possible. This approach also reduces the time needed to monitor the

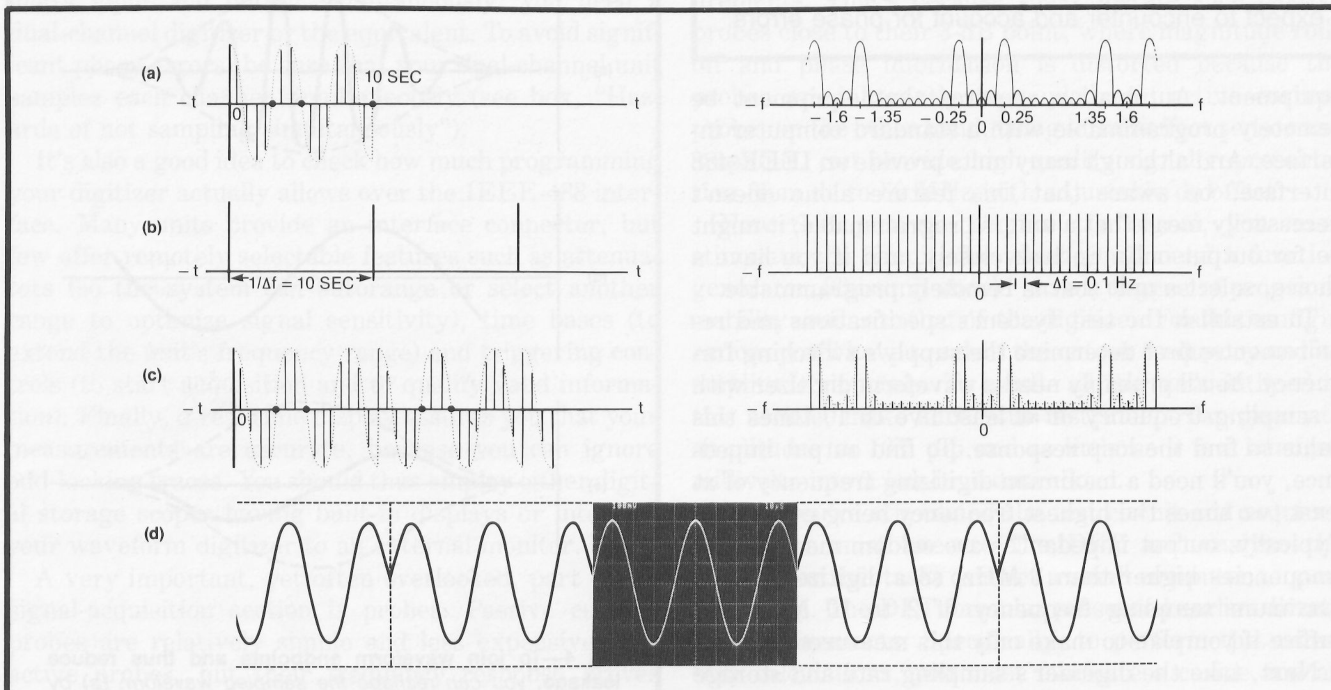


Fig 3—Improper windowing of a signal *can produce erroneous information. When sampled* **(a)** *and windowed* **(b),** *a modulated cosine with a noninteger number of cycles produces a spectrum* **(c)** *with unexpected frequency components. This comes about because the DFT assumes that the sampled record is periodic. If you match the endpoints of the sampled waveform* **(d),** *you can clearly see the discontinuities that create these unwanted frequency components.*

Hazards of not sampling simultaneously

Simultaneous sampling, or a digitizer's ability to sample two channels of input signals, is an important consideration in switching-power-supply measurements. To eliminate significant phase errors, both inputs must be sampled at the same instant.

To do so is no minor accomplishment. In fact, most digitizer manufacturers don't describe clearly how their instruments acquire two channels. Look for such phrases as "dual-simultaneous" acquisition, and don't confuse them with "dual-channel" acquisition.

What's the difference? Often, the unit alternately samples inputs so that the two records are out of phase by one sample interval. In addition, most instruments lower their sampling frequency and record length to half that of single-channel operation in order to capture two signals. As this article shows, such schemes can lead to serious problems.

To be sure you're getting what you need, look for the words "dual-simultaneous" or "true-dual" channel digitizing. Such capabilities are generally highlighted on data sheets and ads. If you can't find or can't afford a "true-dual" unit, note that some digitizers allow capturing of a waveform one channel at a time. Be careful, however, of trigger jitter, and expect to encounter and account for phase errors.

equipment. Automation requires that equipment be remotely programmable with a standard computer interface. And although many units provide an IEEE-488 interface, be aware that this feature alone doesn't necessarily mean that a unit is programmable; it might be for output only. So look closely, and if you have a choice, select a unit that is remotely programmable.

To establish the test system's specifications and requirements, first determine the supply's switching frequency. You'll probably need a waveform digitizer with a sampling frequency of at least five to 10 times this value to find the loop response. To find output impedance, you'll need a maximum digitizing frequency of at least two times the highest frequency being evaluated. Typically, output impedances are seldom measured at frequencies higher than 1 MHz, so a digitizer with a maximum sampling frequency of 2 to 10 MHz will suffice if you plan to make only this measurement.

Next, take the digitizer's sampling rate and storage capacity. This will help you determine the record length when used with the formula provided earlier for frequency resolution and minimum allowed sampling frequency. For example, if you operate the digitizer at a 100-kHz sampling frequency and want to examine loop response from 100 Hz and higher, 1000 points per record are required. Now select each test frequency in the spectrum that you'll inject into the loop for evaluation, making sure that these frequencies put an integer number of cycles into the digitized records. That's a simple task: Because record length and sampling frequency define the frequency resolution, you make the test frequencies a multiple of this minimum frequency. The minimum frequency also has exactly one cycle for the full record, so any integer multiple of it results in an integer number of cycles within the record.

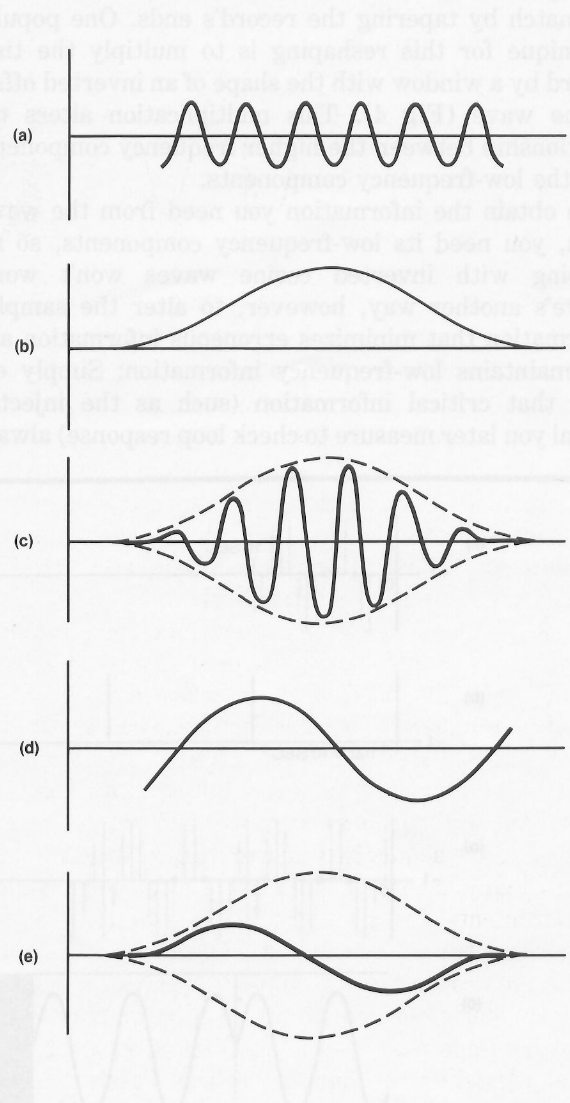


Fig 4—To join waveform endpoints and thus reduce leakage, *you can reshape the sampled waveform* ***(a)*** *by multiplying it with an inverted offset cosine wave* ***(b).*** *This technique preserves most of the high-frequency components* ***(c),*** *but when dealing with low-frequency values* ***(d),*** *it distorts both the amplitude and the shape* ***(e).***

Windowing can add unexpected frequencies to your results

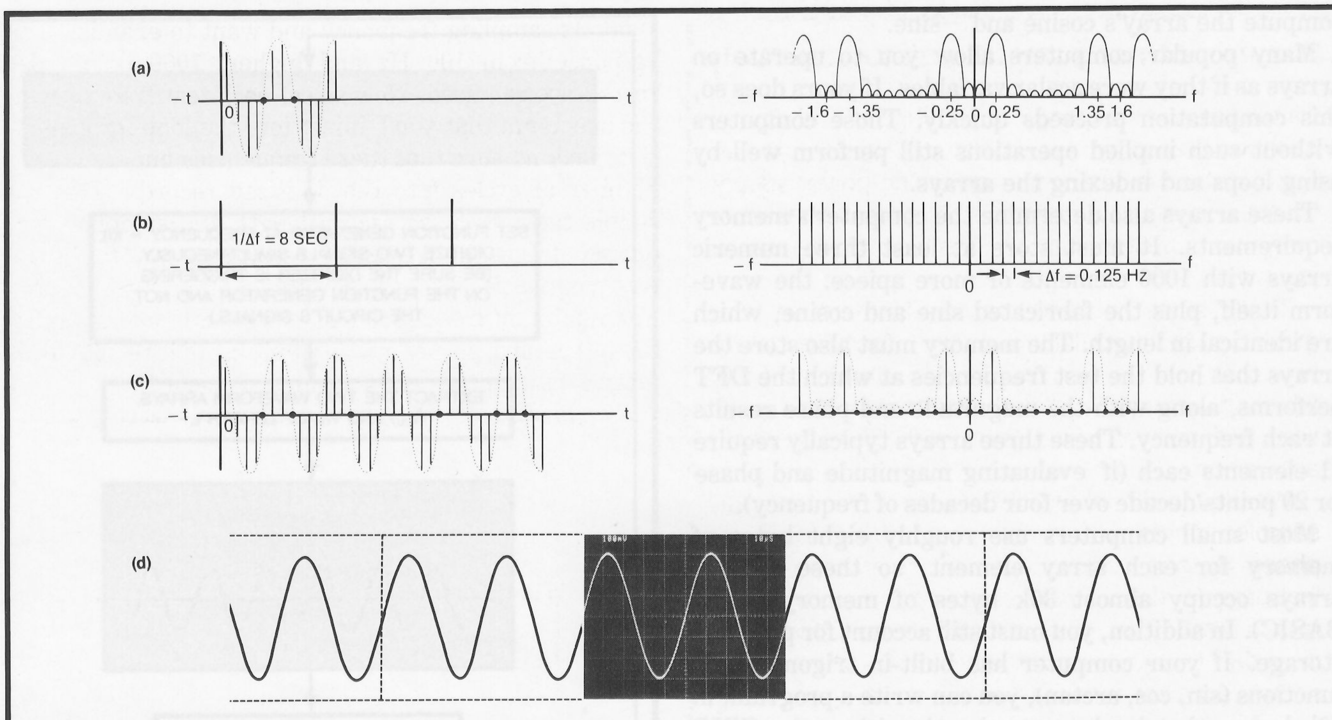


Fig 5—Put an integer number of cycles in the waveform (a) *to eliminate windowing errors. For the window length in* **(b)**, *its resulting spectrum* **(c)** *recovers only the frequencies expected.*

There are other aspects of your waveform digitizer that could affect system performance. For example, because you have to sample the test frequency at the loop's input and output simultaneously, you need a dual-channel digitizer or the equivalent. To avoid significant phase errors, be sure that your dual-channel unit samples each channel simultaneously (see **box**, "Hazards of not sampling simultaneously").

It's also a good idea to check how much programming your digitizer actually allows over the IEEE-488 interface. Many units provide an interface connector, but few offer remotely selectable features such as attenuators (so the system can autorange or select another range to optimize signal sensitivity), time bases (to extend the unit's frequency range) and triggering controls (to start acquisition and to qualify valid information). Finally, a real-time display assures you that your measurements are accurate, because you can ignore odd-looking traces. You should thus employ either digital storage scopes having built-in displays or interface your waveform digitizer to an external monitor.

A very important, yet often overlooked, part of the signal-acquisition section is probes. Passive current probes are relatively simple and less expensive than active probes, but their frequency response proves inadequate for this application. They typically operate as high as 120 MHz, have a low-frequency limit of approximately 100 Hz and can be compensated to 15 or 20 Hz. For output-impedance checks, however, you want measurements as low as several hertz. And even if you're not concerned about output impedance, such probes can throw off your magnitude and phase measurements. That's because you'd operate such passive probes close to their 3-dB point, where magnitude rolls off and phase information is distorted because the probes are inherently ac-coupled. An active current probe, on the other hand, using a Hall-effect sensor and amplifier, can provide high-quality signal representation from dc to 50 MHz with adjustable dc offset.

Now that you can acquire the signal, you must stimulate the circuit, for which you'll need a function generator. This unit should have a degree of programmability similar to that of the digitizer. For this configuration, you'll have to select the output frequency and its amplitude remotely. To handle all the tests outlined in this article, make sure your instrument can generate amplitudes to 10V p-p and as little as a few hundred millivolts into a low-impedance load.

Once you've stimulated the test circuit and acquired data, you must execute the DFT and analyze the results; for this task, connect a small computer to your test setup. The DFT formulas presented earlier dictate the requirements. Specifically, to perform the necessary calculations, you must fabricate a cosine and an inverted sine wave with the same frequency as that in **the waveform record. You can quickly generate them by creating an array of values from 0 to 360° in linear increments multiplied by the number of cycles and**

The injected waveform should have an integer number of cycles

compute the array's cosine and −sine.

Many popular computers allow you to operate on arrays as if they were scalar variables. If yours does so, this computation proceeds quickly. Those computers without such implied operations still perform well by using loops and indexing the arrays.

These arrays also determine the computer's memory requirements. It must store at least three numeric arrays with 1000 elements or more apiece: the waveform itself, plus the fabricated sine and cosine, which are identical in length. The memory must also store the arrays that hold the test frequencies at which the DFT performs, along with the magnitude and phase results at each frequency. These three arrays typically require 81 elements each (if evaluating magnitude and phase for 20 points/decade over four decades of frequency).

Most small computers use roughly eight bytes of memory for each array element, so these various arrays occupy almost 30k bytes of memory (using BASIC). In addition, you must still account for program storage. If your computer has built-in trigonometric functions (sin, cos, arctan), you can write a program, in 30k bytes, that implements the algorithm, the GPIB drivers for the instruments, operator prompting and rudimentary graphics. Consequently, if you have a 64k machine, the program and data will fit tightly. If you have access to additional memory, use it. Finally, you'll probably want to graph the results, so choose a computer or software that supports graphics operations.

The program that calculates the DFTs **(Fig 6)** begins by multiplying the first waveform record by the fabricated cosine and summing the products; this value represents the real part of the frequency component. To compute the imaginary part, calculate the sum of the products by multiplying the waveform record by the −sine array. Magnitude and phase information follow easily: Phase is the arctangent of the imaginary/real ratio, and magnitude is the square root of the sum of the squared real and imaginary parts.

You repeat these calculations for the second waveform record. To find the loop's gain, compute the ratio of the two magnitudes; to find loop phase, compute the difference between the two phases. Repeat this process for the two waveform records at other frequencies within the measured spectrum and plot them to view the circuit's behavior.

Verify a supply's performance

Armed with this setup, you can now examine the performance of switching power supplies. To summarize briefly, set general circuit topology and choose a computer model. Then make the measurements necessary to verify the model. After you gain a high degree of confidence in the design, use software simulations to

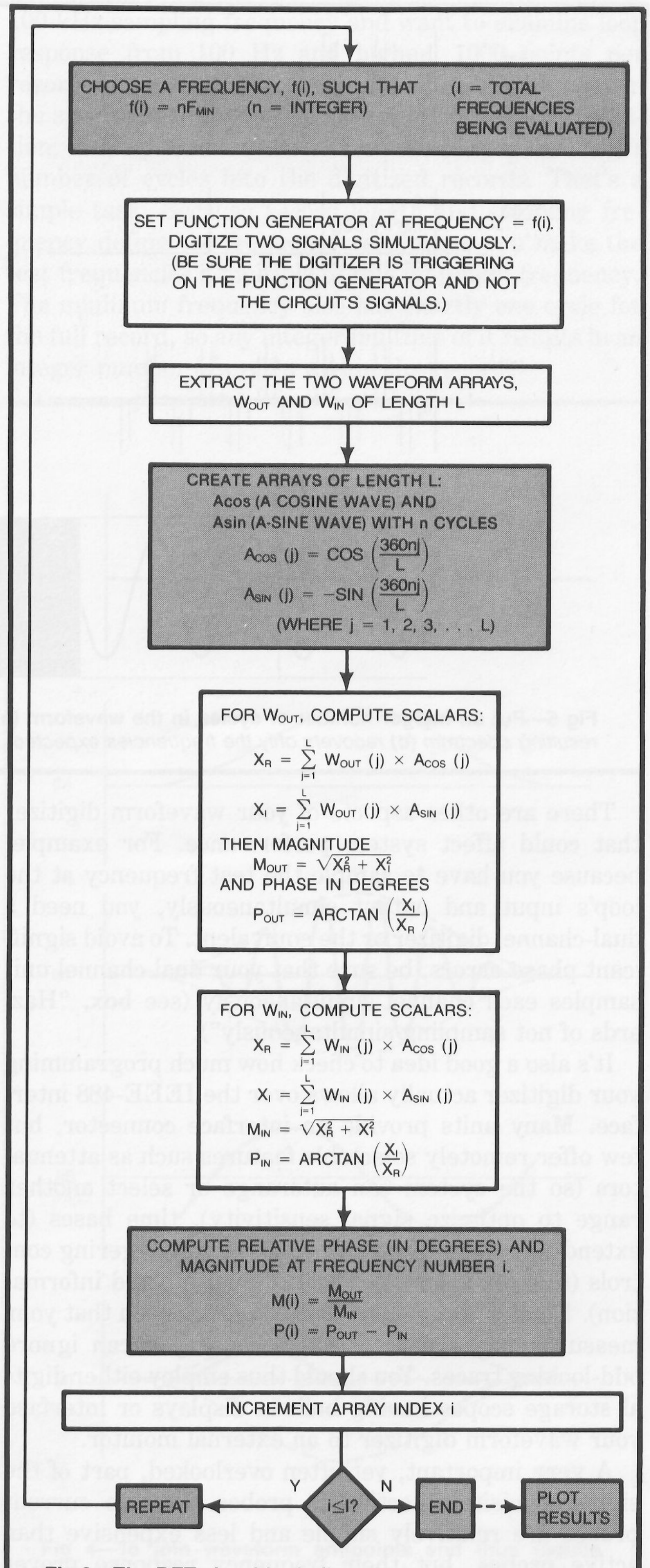


Fig 6—The DFT algorithm's flow diagram *uses two arrays, M(i) and P(i), to store the results of magnitude and phase calculations at selected frequencies (f(i)). When the algorithm is complete, you might want to plot the results vs frequency.*

optimize the design's performance and investigate its response to parameter changes.

The following example illustrates the entire process: The power supply necessary to convert ±50V into 5V for ECL circuitry in the 7A42 (a logic-triggered vertical-amplifier module for 7000 Series oscilloscopes) is designed and verified. The supply's simplified diagram (**Fig 7**) shows all the components that contribute significantly to the open-loop frequency response.

This configuration, a "buck-boost" or flyback type, operates in the discontinuous mode whereby the flyback transformer's secondary current falls to 0 before reconnecting the primary to the ±50V supply through switch Q_1. (Continuous vs discontinuous mode becomes an important distinction later when you're creating the linearized model for computer simulation.)

This supply's heart consists of a standard pulse-width-modulation (PWM) controller IC, which contains a high-gain transconductance error amplifier, a band-gap voltage reference, a pulse-width modulator and shutdown circuitry. This IC determines the On/Off duty cycle at the base of Q_1, a high-voltage power transitor that switches the primary of the flyback transformer to the ±50V input supply. Additional base-drive circuitry minimizes Q_1's turn-on and turn-off times to ensure that it remains in its specified safe operating area. This extra circuitry also increases the supply's efficiency.

During On time, while the switch is closed, the primary ramps up linearly, diode D_1 is reverse biased and no secondary current flows. After a time determined by the PWM IC, the switch opens, primary current falls to 0A and the collector voltage increases until it's clamped through the transformer by D_1. During this time, while the transistor is off, energy is supplied to the secondary. The duty cycle is directly related to several factors: the inductance of the flyback transformer's primary winding, the primary's input voltage, the regulated secondary-output voltage, the switching frequency and the secondary's load.

Given this configuration, where's the best place to insert a test voltage and make loop gain measurements? For stability reasons, the controller IC senses the converter's output voltage at the input to the π filter rather than at the filter's output.

For open-loop gain measurements, replace the strap between points A and B in **Fig 7** with a floating ac

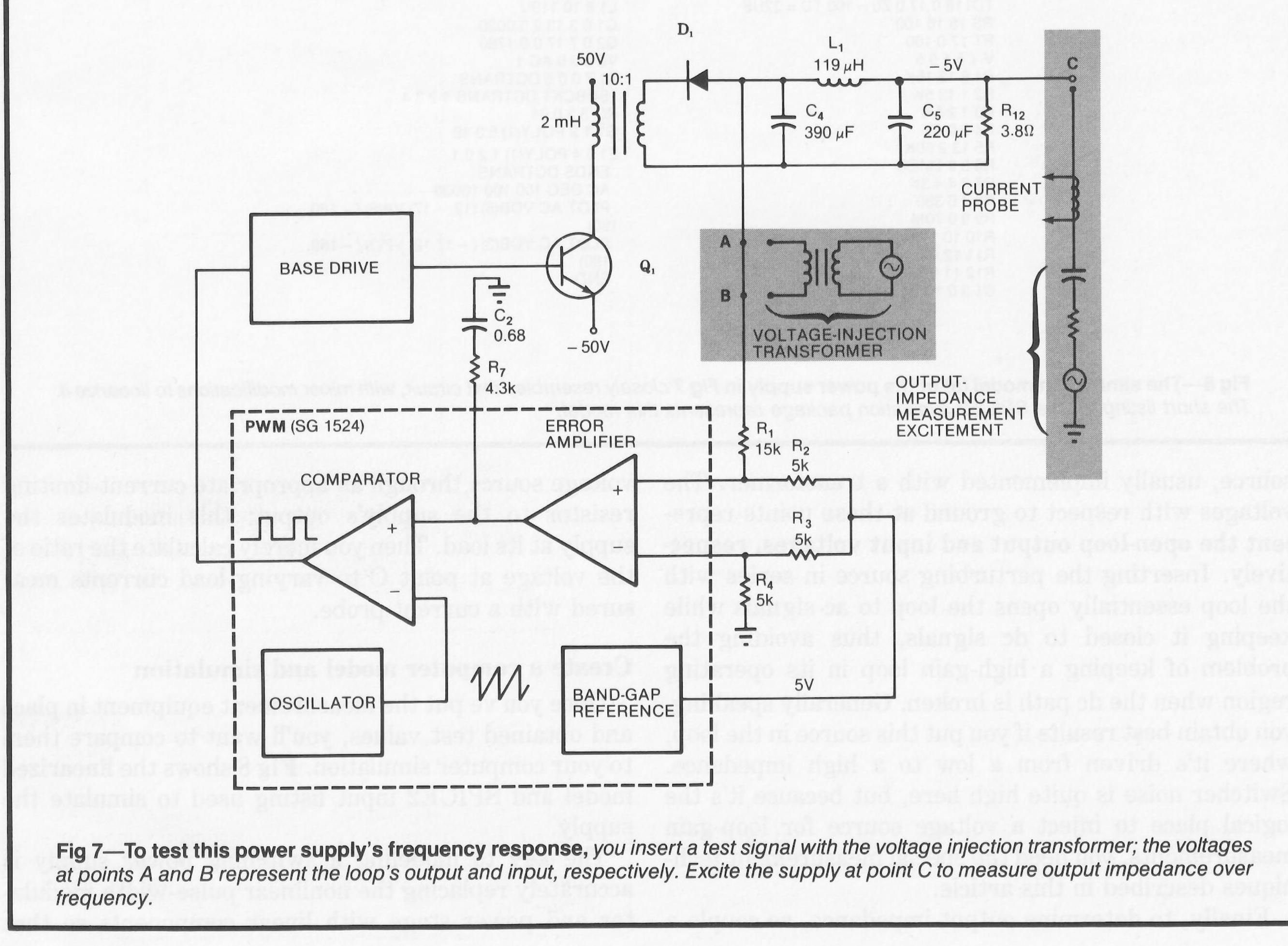
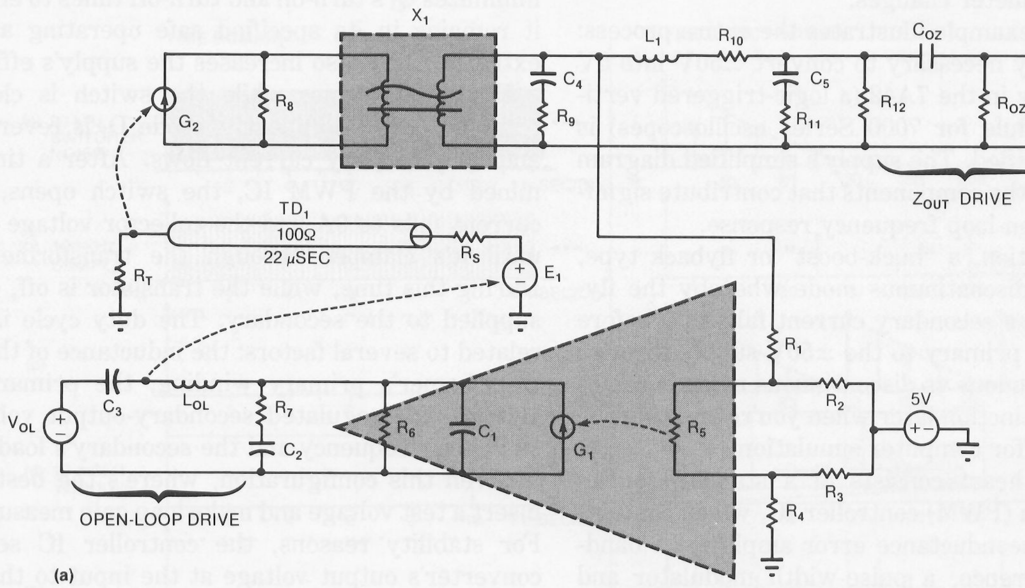


Fig 7—To test this power supply's frequency response, *you insert a test signal with the voltage injection transformer; the voltages at points A and B represent the loop's output and input, respectively. Excite the supply at point C to measure output impedance over frequency.*

Circuit measurements help you verify the computer model



(a)

```
7A42 P/S
* DISCOUNT. MODE - A CL*
*
E1, 15 0 5 0 2.00
TD1 16 0 17 0 Z0 = 100 TD = 22US
RS 15 16 100
RT 17 0 100
V + 5 1 0 5
R1 8 13 15K
R2 1 13 5K
R3 1 2 5K
R4 2 0 5K
R5 13 2 50K
R6 3 0 5MEG
R7 3 4 4.3K
R8 7 0 380
R9 9 0 70M
R10 10 11 0.125
R11 12 0 0.18
R12 11 0 3.8
C1 3 0 100P
C2 4 0 0.68U
C3 6 5 1K
C4 8 9 410U
C5 11 12 220U
LOL 3 5 1G
L1 8 10 119U
G1 0 3 13 2 0.0020
G2 0 7 17 0 0.1780
VOL 6 0 AC 1
X1 7 0 0 8 DCTRANS
. SUBCKT DCTRANS 1 2 3 4
RO 5 3 0.01
G1 1 2 POLY(1) 5 3 10
E1 5 4 POLY(1) 1 2 0.1
. ENDS DCTRANS
. AC DEC 100 100 10000
. PLOT AC VDB(8) (12, – 17) VP(8) ( – 180,
180)
. PLOT AC VDB(3) ( – 17,12) VP(3) ( – 180,
180)
. END
```

(b)

Fig 8—The simulation model (a) of the power supply in ***Fig 7*** *closely resembles that circuit, with minor modifications to linearize it. The short listing for the SPICE2 simulation package represents this model.*

source, usually implemented with a transformer. The voltages with respect to ground at these points represent the open-loop output and input voltages, respectively. Inserting the perturbing source in series with the loop essentially opens the loop to ac signals while keeping it closed to dc signals, thus avoiding the problem of keeping a high-gain loop in its operating region when the dc path is broken. Generally speaking, you obtain best results if you put this source in the loop, where it's driven from a low to a high impedance. Switcher noise is quite high here, but because it's the logical place to inject a voltage source for loop-gain measurements, you need the special measurement techniques described in this article.

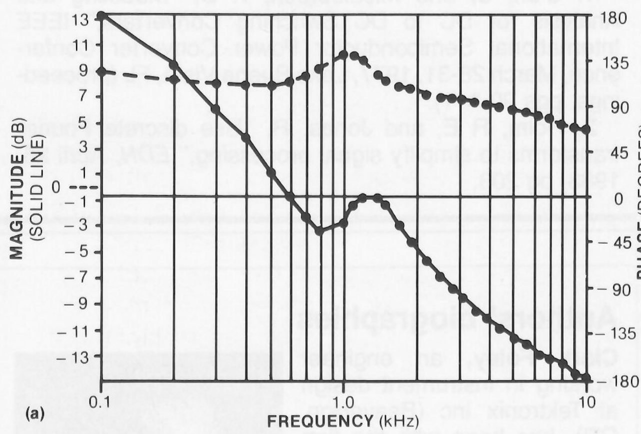
Finally, to determine output impedance, ac couple a voltage source through an appropriate current-limiting resistor to the supply's output; this modulates the supply at its load. Then you merely calculate the ratio of the voltage at point C to varying load currents measured with a current probe.

Create a computer model and simulation

Once you've put the measurement equipment in place and obtained test values, you'll want to compare them to your computer simulation. **Fig 8** shows the linearized model and SPICE2 input listing used to simulate the supply.

The key to modeling a switching power supply is accurately replacing the nonlinear pulse-width modulator and power stage with linear components so that

A verified model provides a production quality-control system



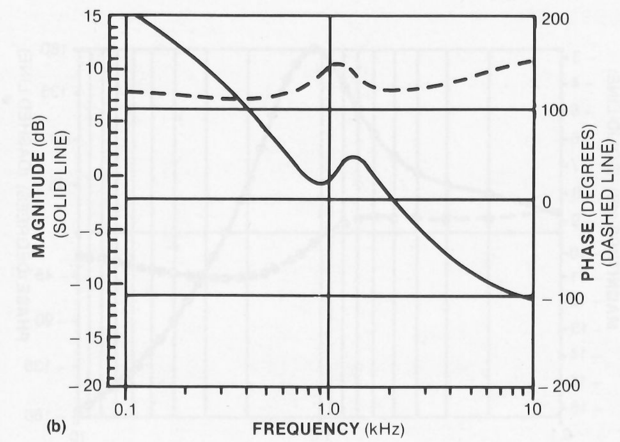
SPICE can perform small-signal analysis (**Ref 1**). Most of the components in **Fig 8** also appear in the simplified block diagram in **Fig 7**, but you must pay attention to some of their more obscure elements. For example, in the output filter, R_{10} is the dc coil resistance of inductor L_1. R_9 and R_{11} are the equivalent series resistances (ESRs) of C_4 and C_5. In some cases, ESR can be a strong function of frequency and thus difficult to model. In this situation, ESR is relatively flat, and you can use a nominal value at 1 kHz.

R_5, R_6, C_1 and G_1 together represent the transconductance error amplifier. These values come from the PWM IC's spec sheet. R_7 and C_2 are external components used for gain setting and frequency compensation. The next step is to replace the pulse-width modulator and power stage with circuit elements G_2, R_8 and subcircuit X_1, which acts as a dc transformer. You can find formulas to determine their values in **Ref 1.**

For open-loop simulations, make L_{OL} and C_3 very large; V_{OL} becomes the ac input source to the loop. The open-loop gain then becomes the ratio of the voltages on either side of L_{OL}. For output-impedance simulations, greatly reduce the values of L_{OL} and C_3 to close the loop. By setting the magnitude of ac source I_{OZ} to unity and its phase to 0, you can directly interpret the voltage at node 11 as the switching supply's output impedance.

Make sure the model is accurate

How closely do simulation-model results come to actual circuit measurements? **Fig 9a** shows the actual measured data of open-loop gain for the supply under test; **(b)** plots the simulated open-loop gain predicted by SPICE2. Predicted and measured curves agree closely, except on the phase curve, where the model shows



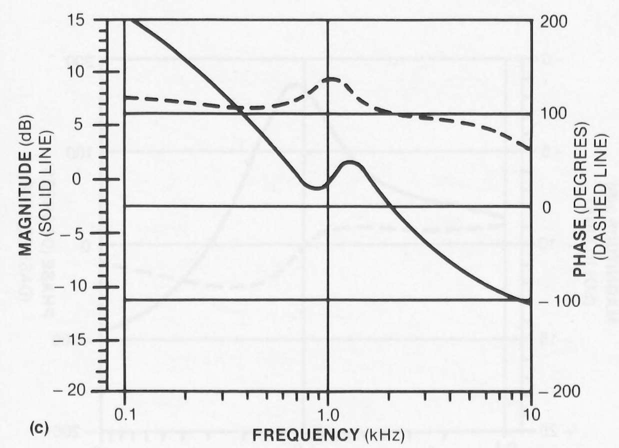


Fig 9—How does measured response compare to simulated values? *Quite well:* ***(a)*** *shows the plotted values from the measurement setup for the power supply in* ***Fig 7****, and* ***(b)*** *gives the values from the first computer simulation. There's a slight discrepancy because phase values taper upwards in the simulation. Adding a delay line in the simulation to correct for this factor produces the closely matched plot in* ***(c).***

phase increasing with frequency. A closer look reveals that the phase difference between predicted and measured values is linear, suggesting that somewhere in the loop there is a fixed time delay that the model doesn't account for.

This deviation turns out to be the time from the end of the On portion of the duty cycle to the time that the packet of energy is delivered to the load. In other words, the error voltage is sampled at roughly the end of the transistor's On time, and the load isn't satisfied until that energy is fully transferred. You can add this fixed time delay to the SPICE2 model with a transmis-

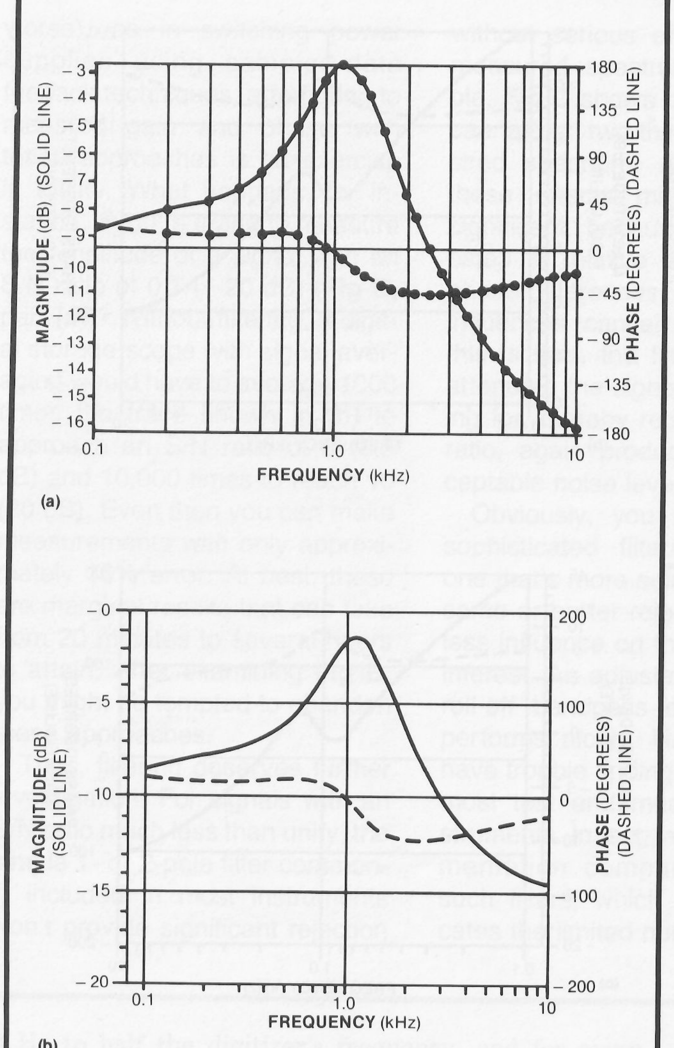


Fig 10—Actual (a) and simulated (b) output-impedance values *match quite closely. The simulation was run on the modified linearized circuit developed for the loop-gain plot in Fig 9c.*

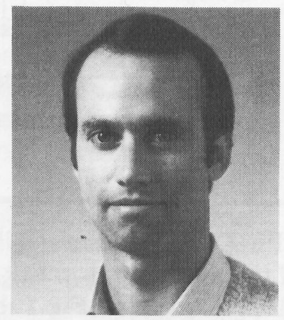
sion line specified to have the same time as that delay. **Fig 9c** shows the results of a second iteration on the model taking this effect into account. Finally, the plots in **Fig 10** show the measured and simulated output impedances using the refined model for comparison; they match very closely.

With this verified model, you can use a computer to evaluate the circuit's resiliency to component tolerances. Also, in building this test system and verifying the model, you now have an accurate quality-control system that can check the performance of supplies that leave manufacturing. **EDN**

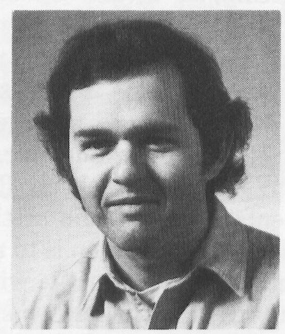
References

1. C'uk, S, and Middlebrook, R D, "Modeling and Analysis for DC to DC Switching Converters," IEEE International Semiconductor Power Converter Conference, March 28-31, 1977, Lake Buena Vista, FL (proceedings, pgs 90-111).
2. Holm, R E, and Jones, R, "Use discrete Fourier transforms to simplify signal processing," *EDN*, April 28, 1983, pg 203.

Authors' biographies



Clark Foley, an engineer working in instrument design at Tektronix Inc (Beaverton, OR), has been with the firm eight years. During that time, he spent six years in Laboratory Instrument Marketing, where he was involved in the development of new-product introductions and ideas, customer and sales training, and the writing of application articles. Clark earned his BSEE from Arizona State University (Tempe). His hobbies include water and downhill skiing, golf, attending cultural events and frequenting Portland's restaurants.



Jim Lamb has spent seven years with Tektronix Inc and now works as a design engineer involved in high-speed analog-IC design. He received his BSEE and MSEE degrees from Washington State University (Pullman). If Jim had any spare time, he'd enjoy water and snow skiing as well as cycling in the Portland area.

Reprinted from EDN April 5, 1984

GPIB IEEE-488 **7D20** ***NEW* 7D20T**

The 7D20 and 7D20T comply with IEEE Standard 488-1978, and with Tektronix *Standard Codes and Formats*.

Digital Storage for 7000 Series Mainframe (7D20)

70 MHz Bandwidth for Repetitive Signals

10 MHz Single-Shot Bandwidth

Two Channels Simultaneous Acquisition

Totally Programmable

Storage of Six Independent Waveforms

Enveloping and Signal Averaging

Cursor Measurements

Pretrigger and Posttrigger

APPLICATIONS

* **Ultrasonics**
* **Digital Design**
* **RF Modulation**
* **Automated Production Testing**

The 7D20 brings state-of-the-art digital performance to Tektronix 7000 Series mainframes and rackmounts.

The 7D20 is a GPIB programmable plug-in that is compatible with all 7000 Series mainframes (including the USM 281C) except the 7104. When combined with a 7000 Series mainframe, this plug-in creates a fully programmable, digitizing oscilloscope.

The 7D20T is the ideal high performance digitizer in automated systems applications where visual display of the acquired signals is not required.

The 7D20T is supplied with its own power module, but without a display. Rear panel connectors provide X, Y, and Z output data for use with an external X-Y monitor, if desired.

Four feedthrough cables permit routing of input/output signals from the front panel of the 7D20T to the rear in rack-mounted applications. The GPIB cable may be connected to either the front or the rear of the 7D20T.

The capabilities and characteristics described here for the 7D20 also apply to the 7D20T.

For those users who already own a 7D20 and would like to convert this 7000 Series plug-in into the 7D20T configuration, the power module itself is available as a 7D20T Option 01.

The 7D20/7D20T can accurately measure the amplitude of a 50 ns wide transient event. Dual samplers simultaneously acquire two channels as if it were a "dual-beam" scope.

Beyond basic acquisition, the 7D20/7D20T offers signal averaging to reduce uncorrelated noise, envelope displays to compare dynamic characteristics of changing signals, pretrigger for viewing prior to the trigger event, storage of six independent waveforms plus a reference waveform, cursors for more accurate two-dot measurements, and user prompting and menu displays to improve user interface effectiveness.

Cursor Measurements

Accurate amplitude measurements (referenced to ground) and time measurements (referenced to trigger position) are made using one cursor. Point-to-point difference (Δ) measurements are made using two cursors.

Digital Storage

A 40 MHz maximum sampling rate provides approximately 10 MHz single-shot bandwidth and up to 70 MHz bandwidth with repetitive signals.

Storage and Recall Front Panel Settings

Up to six different front panel set-ups can be stored and recalled as desired. These settings, plus the last panel setup, are saved in nonvolatile memory and are restored automatically when power is applied.

Fully Automated Measurements

Since the 7D20/7D20T is completely programmable, fully automated measurement and testing is possible. Tektronix programmable signal sources, multi-function interface, and RF scanner provide and control the test signals while the 7D20/7D20T acquires waveforms for the computer or controller.

Hands Off Operation With Probe Identify Feature

Recommended for use in interactive, computer-coordinated tasks, the Tektronix P6053B Probe allows computer routines to be sequentially activated at the 7D20's probe tip. This probe's "Identify" button signals the GPIB Interface via an input channel coded request. This capability allows the operator to work at a short distance from the 7D20 without the need to touch front panel controls. Two such probes may be used, one for each vertical channel.

CHARACTERISTICS

VERTICAL SYSTEM

Input — Two channels, simultaneous sampling, BNC connectors.

Acquire Modes — CH 1, CH 2, Add, Both (dual channel).

Sensitivity — 5 mV to 5 V/div; 1-2-5 sequence.

Bandwidth — 70 MHz maximum. (Ac Coupled Low Frequency Response: 10 Hz or less.)

Step Response — 5 ns or less.

Input Impedance — 1 MΩ paralleled by ≈20 pF.

Maximum Input Voltage — Dc Coupled: 250 V, 1 kHz or less (dc + peak ac). Ac Coupled: 400 V, 1 kHz or less (dc + peak ac).

Signal Isolation — 100:1 dc to 20 MHz.

Vertical Resolution — 8 bits, 256 levels, 0.04 div/level.

Gain Ratio Accuracy — <2%. Maximum error throughout the V/div range with acquire gain calibrated at 10 mV/div. Measurement valid with Cursors or GPIB.

Noise — Mean value of 50 measurements taken at 0.02 div increments.

Volts/Div	Full Scale/RMS Noise	Percent of Full Scale
5 mV	52 dB	0.25
10 mV to 5 V	55 dB	0.18

NOTE: Full scale = 10.24 divisions.

Phase Match X-Y — <2° from dc to 10 MHz.

HORIZONTAL SYSTEM

Time Division Range — External Clock, 20 s/div to 50 ns/div in 1-2-5 sequence.

Digitizing Technique Versus Time/Division — Real Time (Rolling Display): External Clock, 20 s/div to 0.1 s/div. Real Time: 50 ms/div to 500 μs/div. Extended Real Time: 200 μs/div to 2 μs/div. Equivalent Time: 1 μs/div to 50 ns/div.

Note: Single events can be captured as fast as 2 μs/div. For 1 μs/div to 50 ns/div, repetitive events are required to build a complete waveform.

Time Measurement Accuracy — One Cursor: 0.1% of reading +0, −1 sample interval ±300 ps. Two Cursors: 0.1% of reading ±600 ps.

Horizontal Resolution

Time/Division	Points/ Waveform	Resolution Points/Division
External, 20 s to 500 μs	1024	100
200 μs to 2 μs	820*1	80*1
1 μs to 50 ns	1024	100

**1 Waveform interpolation to 1024 points is available for transfer over the GPIB Interface.*

Trigger Position

Pretrigger: 0 to 10 div in 1 div increments. Posttrigger (delay): 0 to 1500 div in 1 div increments (disabled during Roll with Envelope or Average).

	Frequency Range*1	Sensitivity Internal	Sensitivity External
Normal (Dc Coupling)	dc to 30 MHz	0.4 div	60 mV
	30 MHz to 70 MHz	1.0 div	150 mV
P-P and Auto	30 Hz to 200 Hz	2.0 div	300 mV
	200 Hz to 30 MHz	0.6 div	90 mV
	30 MHz to 70 MHz	1.2 div	200 mV

**1 The ac coupling low frequency limit is 30 Hz. In Time/Div settings of 1 μs to 50 ns, when using P-P or Auto, low-frequency limit is 300 Hz.*

SIGNAL PROCESSING

Cursors Readout — With one cursor (ΔOff), vertical and horizontal coordinate values are referenced to zero volts and the trigger position as zero time. With two cursors (ΔOn), vertical and horizontal coordinate values are the difference between the two cursors.

Signal Averaging

AVE N: A self-terminating, stable average processing "N" number of waveforms and then holds the result in memory. The "N" value may be selected using the SET N function (N = 8, 16, 32, 64, 128, 256).

AVE: A continuous, stable averaging process. N waveforms are averaged as in AVEN, then additional waveforms are weighted at 1/N. In Roll mode a running average (smooth) is available to provide high frequency filtering.

Enveloping

ENV N: A self-terminating recording of waveform maxima and minima. When N waveforms are processed, the result is held in memory.

ENV: A continuous (infinite) recording of waveform maxima and minima.

Waveform Modifiers

VPUP ↑ (Vertical Position Up), VPDN ↓ (Vertical Position Down): Provide vertical positioning control of any stored waveforms.

VCMP (Vertically Compress), VXPD (Vertically Expand): Provide vertical display expansion or compression. Two expansions or compressions in 1,2,5 calibrated steps, from the original V/div are available.

HMAG (Horizontal Magnify): Displays the cursor waveform horizontally magnified by a factor of 10. HMAG ALL (Horizontally Magnify All Waveforms): Displays all waveforms at 10 times horizontal magnification.

VS (Versus): Creates a Y versus X display of any two waveforms.

GPIB INTERFACE

Interface Function Subsets Implemented:

SH1	Complete source handshake
AH1	Complete acceptor handshake
T5	Complete talker — no secondary address
L3	Complete listener — no secondary address
SR1	Complete service request
RL1	Complete remote local
DC1	Complete device clear
DT1	Complete device trigger
PP∅	No parallel poll
C∅	No controller
E2	Three state

Programmable Functions — All instrument setting and operating modes are programmable except for Variable V/Div and Horizontal Position. However, these uncalibrated controls can be overridden and forced into the "CAL" position on command from the GPIB Interface. The display of Menu and ID is selectable from the front panel only.

Format — Device dependent commands in ASCII. Waveform data points selectable as BINARY or ASCII.

Waveform Output Time — 250 ms minimum for BINARY and 2.5 s minimum for ASCII. Actual transfer times depend upon the speed of the receiving device.

INPUTS

External Trigger (Front Panel) — Maximum Input Voltage: 250 V (dc + peak ac).

Signal Input Impedance — 1 MΩ, paralleled by ≈20 pF.

Hold Next (Mainframe Rear Panel) — Initiates Hold Next condition; connected to Single Sweep Reset connector.

OUTPUTS

Hold Next Ready — High level indicates unit is in Hold Next condition; output level remains low when unit is not in Hold Next condition; connected to Single Sweep Ready connector.

+Gate Out — Provides high level output signal for duration of waveform/character readout.

PLUG-IN COMPATIBILITY

The 7D20 is compatible with all 7000 Series mainframes with the exception of the 7104 mainframe. Use with the 7104 will void the 7104 warranty.

PHYSICAL CHARACTERISTICS

	7D20		7D20T	
Dimensions	mm	in	mm	in
Width	206	8.1	216	8.5
Height	127	5.0	183	7.2
Depth	371	14.6	566	22.3
Weight	kg	lb	kg	lb
Net	3.6	8.1	10.4	23.0
Shipping	8.0	17.8	12.0	26.6

Included Accessory — Instruction manual.

ORDERING INFORMATION

7D20 Programmable Digitizer (plug-in).
7D20T Programmable Digitizer.*1
7D20T Option 01 Power Module converts existing 7D20 to 7D20T.*1

*1 Price available upon request.

RECOMMENDED PROBE (7D20 and 7D20T)

P6053B Identify Probe — For remote service request via probes "Identify" button. 10X attenuation; 200 MHz bandwidth; scale factor coding; 6 ft.
Order 010-6053-13.

RECOMMENDED MAINFRAME FOR 7D20

R7603 Option 20 — The R7603 mainframe provides a 6-inch diagonal CRT display and three-wide plug-in compartment in a 5.25 inch high rackmount configuration. Option 20 permits rear panel access to the 7D20's GPIB Interface and includes cable (175-7151-00) required inside the 7D20.
Order R7603 Option 20 Mainframe.

Utility Software.

OPTIONAL ACCESSORY (R7603)

A field installable kit adds Option 20 to the standard R7603. Intended for use with a previously purchased R7603, this kit provides parts to connect the 7D20's GPIB Interface to the R7603 mainframe.
Order 040-1093-00.

INTERNATIONAL POWER CORDS AND PLUG OPTIONS (7D20T)

Option A1 — Universal Euro 220 V/16 A, 50 Hz
Option A2 — UK 240 V/13 A, 50 Hz
Option A3 — Australian 240 V/10 A, 50 Hz
Option A4 — North American 240 V/15 A, 60 Hz
Option A5 — Switzerland 220 V/10 A, 50 Hz

RACKMOUNTING 7D20T

Rackmounting kits are available for rackmounting the 7D20T with various other half rackwidth products from Tektronix. Please call your local Tektronix Sales Office for descriptions, part numbers, and prices.

For further information, contact:

U.S.A., Asia, Australia, Central & South America, Japan
Tektronix, Inc.
P.O. Box 1700
Beaverton, Oregon 97075
For additional literature, or the address and phone number of the Tektronix Sales Office nearest you, contact:
Phone: (800) 547-1512
Oregon only: (800) 452-1877
TWX: (910) 467-8708
TLX: 151754
Cable: TEKWSGT

Europe, Africa, Middle East
Tektronix Europe B.V.
European Headquarters
Postbox 827
1180 AV Amstelveen
The Netherlands
Phone: (20) 471146
Telex: 18312 - 18328

Canada
Tektronix Canada Inc.
P.O. Box 6500
Barrie, Ontario L4M 4V3
Phone: (705) 737-2700

Tektronix sales and service offices around the world:
Albania, Algeria, Angola, Argentina, Australia, Austria, Bangladesh, Belgium, Bolivia, Brazil, Bulgaria, Canada, Peoples Republic of China, Chile, Colombia, Costa Rica, Czechoslovakia, Denmark, East Africa, Ecuador, Egypt, Federal Republic of Germany, Fiji AWA New Zeland, Finland, France, Greece, Hong Kong, Hungary, Iceland, India, Indonesia, Ireland, Israel, Italy, Ivory Coast, Japan, Jordan, Korea, Kuwait, Lebanon, Malaysia, Mexico, Morocco, The Netherlands, New Zealand, Nigeria, Norway, Pakistan, Panama, Peru, Philippines, Poland, Portugal, Qatar, Republic of South Africa, Romania, Saudi Arabia, Singapore, Spain, Sri Lanka, Sudan, Sweden, Switzerland, Syria, Taiwan, Thailand, Turkey, Tunisia, United Arab Emirates, United Kingdom, Uruguay, USSR, Venezuela, Yugoslavia, Zambia, Zimbabwe.

Tektronix
COMMITTED TO EXCELLENCE

42W-5646-1