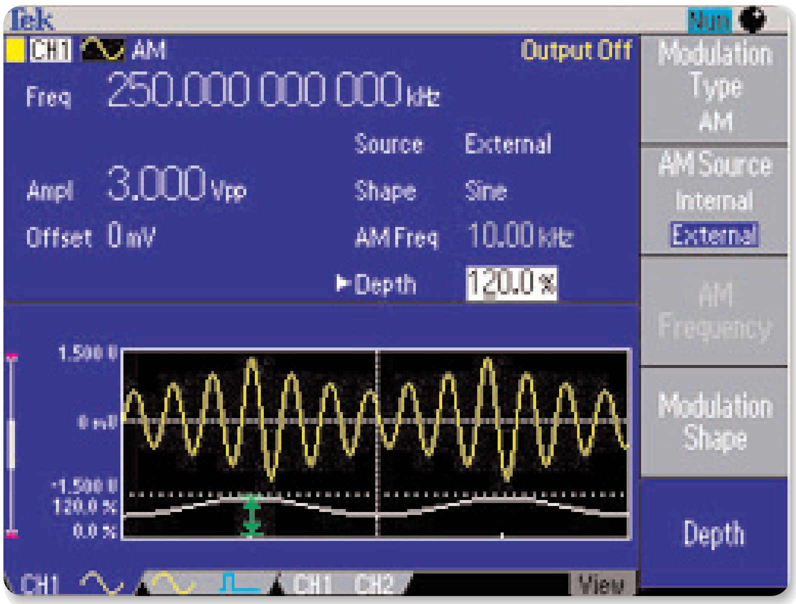
Understanding Signal Generation Methodologies

Choosing Between Direct Digital Synthesis (DDS) and Variable Clock Architecture ("True Arb")



In electronic test and measurement, more often than not, a signal source is required to generate signals that are not available unless externally provided.

A signal source can provide "known good" signals or it may add known, repeatable amounts and types of distortion (or errors) to the signal it delivers. This characteristic is one of the signal source's greatest virtues, since it is often impossible to create predictable distortion exactly when and where it is needed using only the circuit itself. Signal sources are used for hundreds of applications ranging from design verification to characterization and from stress and margin to compliance tests. Not surprisingly there are a choice of signal source architectures, each with certain strengths, capabilities and cost-effectiveness for particular applications. In this document we will compare two signal generation architectures: one used in Arbitrary/Function generators and one used in Arbitrary Waveform generators. The right choice very much depends on the application.

Arbitrary/function generators (AFG) create both function waveforms and arbitrary waveforms by reading the contents of an internal memory. Most modern AFG's use direct digital synthesis (DDS) technology to deliver signals over a broad range of frequencies.

Arbitrary waveform generators (AWG) based on a true variable clock architecture (commonly referred to as "true arbs*1") are favored for more complex waveform generation at all frequencies. AWGs, too, read the contents of an internal memory, but in a different way (described later). Designers working with advanced communications and computing elements choose the AWG to drive high-speed signals with complex modulation and anomalies. As a result, AWGs occupy the highest tier of research, development, and engineering applications.

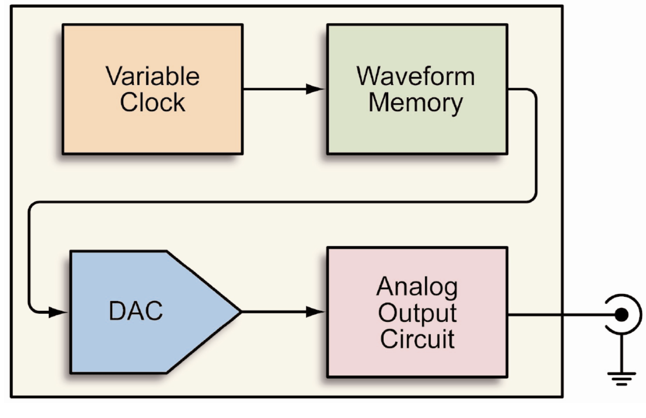
The two architectures differ greatly in terms of their waveform generation approach. This technical brief discusses the differences between variable-clock-based arbitrary waveform generators and DDS-based arbitrary/function generators.

Behind the Front Panel: Comparing the Platforms

AWG: Simple Concept, Maximum Flexibility

Though it is the more flexible of the two architectures, the underlying waveform generation technique of the AWG is straightforward. The AWG's playback scheme can be thought of as "sampling in reverse."

What does that mean? Imagine an oscilloscope, the ultimate sampling platform. It acquires a waveform by digitizing the analog signal's voltage value at a succession of points in time, with the frequency of the sampling being determined by the user-selected clock rate. The resulting samples end up in a memory.



▶ **Figure 1.** *Simplified block diagram of the AWG architecture.*

The AWG reverses the process. The AWG starts with a waveform already in its memory. The waveform occupies a designated number of memory locations. With every clock cycle, the instrument outputs another waveform sample from the memory. Because the number of samples representing the waveform is fixed, a faster clock rate will read through the waveform data points in memory more quickly, yielding a higher output frequency. In other words, the output signal frequency depends entirely on the clock frequency and the number of waveform samples in memory*2. The simplified block diagram in Figure 1 summarizes the AWG architecture.

The AWG's flexibility comes from the waveform stored in its memory. The waveform can take any shape; it can have any number of aberrations, or none at all. With the help of PC-based tools, users can develop literally any wave shape the mind can conceive (within the constraints of physics!). The samples can be read out of the memory at any clock frequency the instrument is capable of producing. The waveform will have the same shape whether the clock is running at 1 MHz or 1 GHz.

*1 Engineers often use the casual term "arb" when referring to any type of arbitrary waveform generation instrument.

*2 There is of course a maximum memory capacity for any AWG model. The waveform can occupy less depth than the full capacity.

AFG Takes Efficient Shortcuts to High Frequency

The AFG, too, uses a stored waveform as the basis of its output signal. A clock signal is involved in the readout of the samples. But here the similarity ends.

The AFG's clock runs at one fixed rate. Since the number of waveform samples is also fixed in memory, how can the AFG deliver the waveform at varying frequencies?

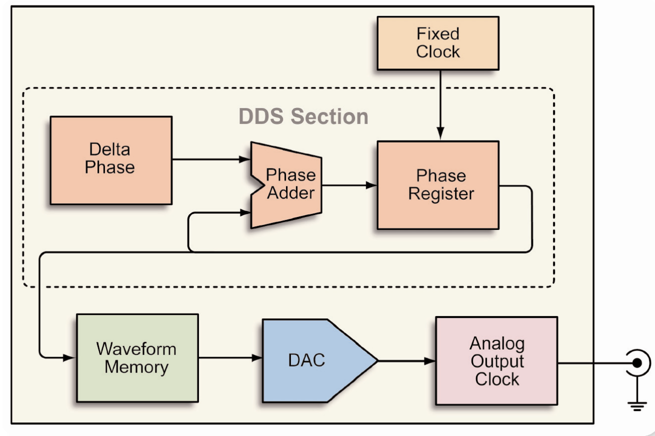
For example, imagine you were using an AFG with a stored waveform consisting of 1000 samples that are output at a fixed rate of 1 MHz. The period of the output signal would be fixed at exactly 1 ms (1 kHz).

Obviously a single-frequency signal source would be of limited use in most applications. So DDS technology offers a solution. Instead of reading every sample, a DDS-based instrument reconstructs the waveform by reading fewer than 1000 samples.

Figure 2 depicts a typical AFG architecture, including the DDS section, in simplified form. The output signal is formed by the clock, a stored binary number representing a phase value, and the contents of the waveform memory.

As already explained, the AFG maintains a fixed system clock frequency. The 360 degrees of a waveform cycle are spread across the full number of waveform samples, and the DDS section automatically determines the phase increments based on the waveform length and the frequency selected by the user.

High frequency settings result in large phase increments, causing the AFG to skip ahead through the 360-degree cycle quickly, delivering a high-frequency signal. Low frequency values lead to small increments triggering the phase accumulator to step through the waveform samples at a slower pace and even repeat individual samples to complete the 360 degrees, producing a lower-frequency waveform.



▶ **Figure 2.** *Simplified block diagram of an AFG architecture.*

The math behind all of this decision-making is beyond the scope of this discussion. Suffice to say that the AFG skips selected waveform data points based on its own internal algorithms. Because of the phase increment approach, it doesn't always skip the same samples in every cycle. The AFG offers an expedient way to produce varied waveforms and frequencies, but the end-user cannot control which data points are skipped.

This is bound to have some impact on output waveform fidelity. Waveforms with continuous shapes—sines, triangles, etc.—are usually not a problem but signals with fast transitions like pulses and transients common in today's digital environment can be affected. Suppose, for example, you are doing stress tests on a new component for a telecom switch. The test waveform is a series of binary pulses, one of which has a transient on the rising edge. At some frequencies the DDS phase increments might skip right past the transient without clocking it out as part of the signal. To the device under test (DUT), the signal looks like an undisturbed pulse stream and the stress test, lacking any actual "stress," is invalid.

	AFG (DDS)	AWG
Sample Clock Rate	Fixed	Variable
Sample Increment	Varies automatically, depending on output frequency setting	Fixed, 1 point per clock
Memory Depth	Fixed or variable	Variable

▶ **Table 1.** *AFG vs. AWG sampling characteristics.*

The AFG architecture is less expensive to implement than a full-featured AWG toolset. Consequently it is affordable enough to be allocated to individual engineers and researchers. And the AFG has some unique performance advantages of its own. Some leading models have the best frequency agility—the ability to switch smoothly among different frequencies without producing discontinuities in the signal—of any waveform-generating platform.

Table 1 summarizes the clock and memory characteristics of the AFG and AWG platforms.

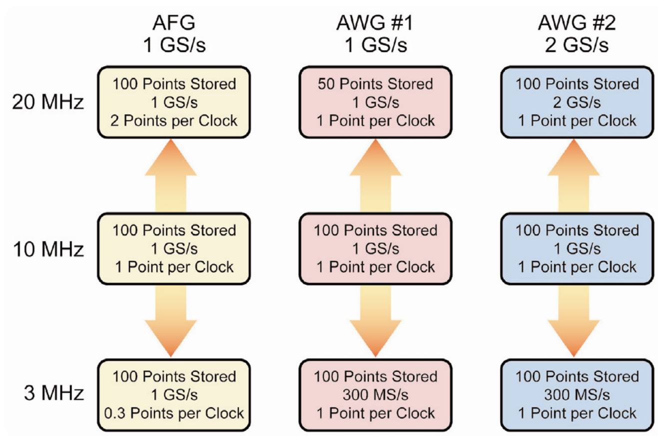
Drilling Down to the Details

To better understand the contrast between AWG and AFG architectures, a brief "case study" is in order. We will look at the way in which the two platforms handle the sample points that define the output waveforms.

There are three instruments in this comparison: an AFG with a maximum sample rate of 1 GS/s; AWG #1 with a maximum sample rate of 1 GS/s; and AWG #2 with a maximum rate of 2 GS/s.

The object is to produce a sine wave at frequencies ranging from 3 MHz to 20 MHz. Both of the AWGs and the AFG are loaded with one cycle of the sine wave in 100 points of sample memory. Figure 3 shows how the attributes of the three platforms affect the way they handle the task.

All three tools can read through the 100 points at a sample rate of 1 GS/s to produce the 10 MHz sine wave (middle row in Figure 3):



▶ **Figure 3.** *Three approaches to managing the frequency of the output signal.*

- Instructed to deliver 10 MHz at the output, the AFG's DDS element calculates an increment of 1 point for each tick of the 1 GS/s clock. It touches every one of the 100 sample points.
- The clocks in both of the AWG are manually set to 1 GS/s, and they too read the 100 points to generate the 10 MHz waveform.

The methods diverge when you set the output frequency to 3 MHz (bottom row):

- The AFG's clock still runs at its fixed 1 GS/s rate. But now the DDS automatically sets the increments to 0.3 points per clock tick; that is, individual data points are repeated three or four times.
- The clock frequencies in both AWGs must be reduced manually to 300 MS/s. The clock now reads through the points more slowly, yielding the 3 MHz output frequency.

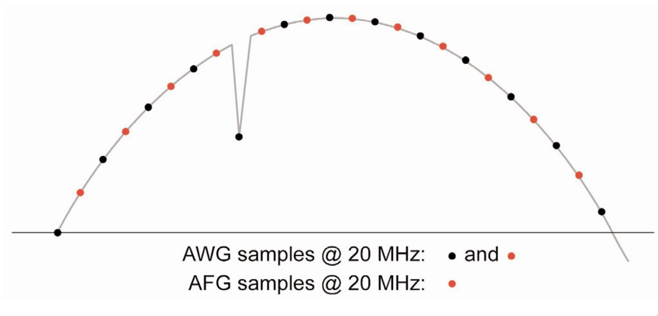
Now the output frequency must be increased to 20 MHz. Here all three platforms handle the challenge differently:

- The AFG's DDS element sets the sample increment to two samples. It reads every other sample, using a total of 50 points to define the waveform. This takes only half as long as reading 100 points. The result is a 20 MHz output signal.
- AWG #1 reads one sample per clock tick, as do all AWGs at any frequency setting. But because its maximum sample rate is 1 GS/s, it can't read 100 samples within the 50 ns period of a 20 MHz sine wave cycle. Therefore, the stored waveform image must be decreased by deliberate user intervention to a total of 50 samples. The result is a 20 MHz output signal.

  Software tools are available to help users edit the sample count when required, and some instruments have built-in features for the purpose. When using external tools, the modified waveform must be reloaded into the AWG.
- AWG #2 reads one sample per clock tick, but the clock rate is doubled to 2 GS/s. The instrument reads through its 100-point memory twice as fast. The result is a 20 MHz output signal.

At first glance, it would appear that AWG #1 is limited to the same waveform resolution as the AFG. But there is a critical difference. At a 20 MHz output frequency, the AWG is reading every single one of the 50 samples in the sine wave. The AFG is skipping samples.

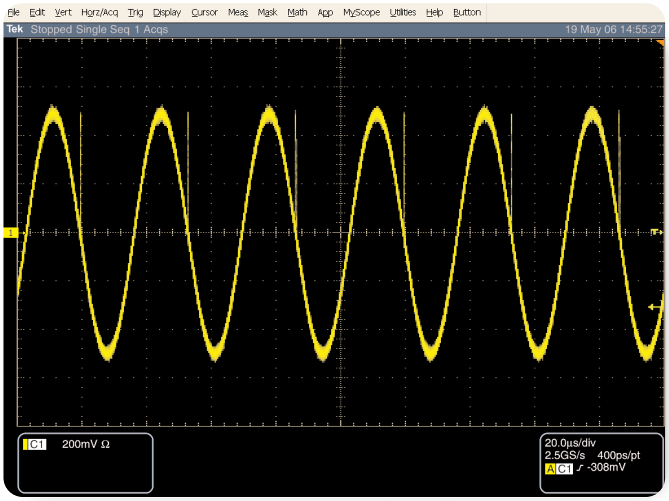
Figure 4 illustrates the basic dichotomy between the AFG/DDS and AWG approaches. The image depicts one half-cycle of a sine wave consisting of 25 points, including an aberration added to simulate a momentary dropout on a DAC.



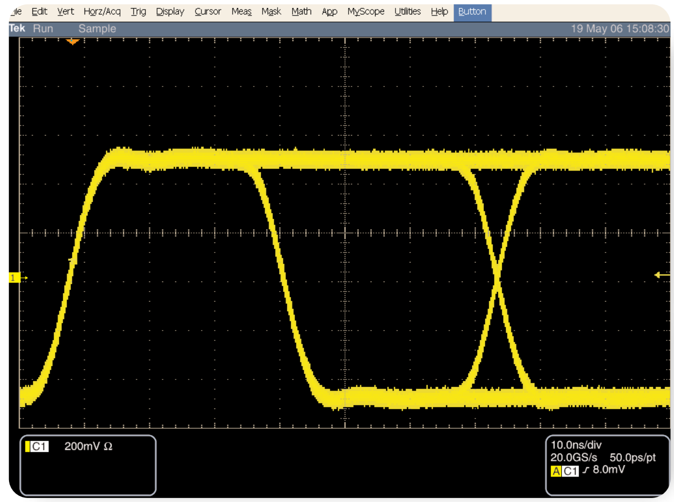
▶ **Figure 4.** *The AFG skips samples to increase its output frequency. Individual signal details can be overlooked at some frequencies.*

The AWG reads every point, red or black, irrespective of the output frequency setting. If the output frequency is set to 10 MHz, the AWG reads 25 points. If it is set to 20 MHz, the AWG still reads 25 points. If the maximum clock rate within the AWG isn't high enough to produce the desired frequency by reading all of the points, then the number of points can be reduced. Assuming that the user pays attention to preserving the desired waveform characteristics when trimming the AWG's sample count, the instrument will reliably deliver the glitch once in every cycle.

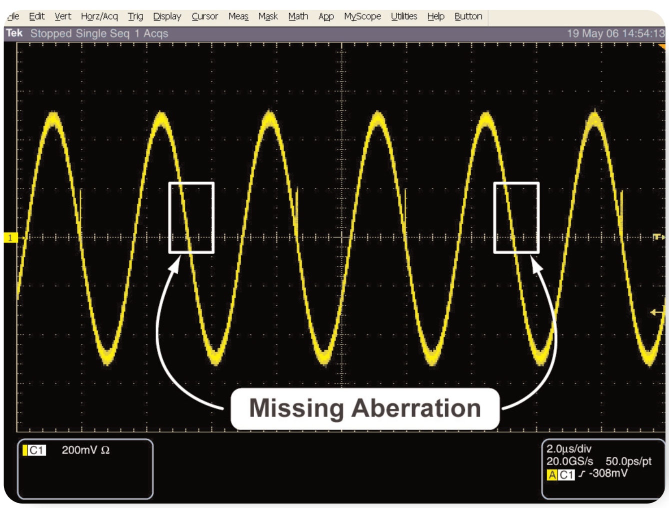
Now consider the AFG. If the output frequency is set to 10 MHz, it reads every point. If it is set to 20 MHz, it reads every second point, nominally. These DDS points are shown in red. *Notice that the AFG completely bypasses the glitch*. It skips the very sample that defines the dropout. The waveform goes out as a clean sine wave. The device under test doesn't receive the aberration.



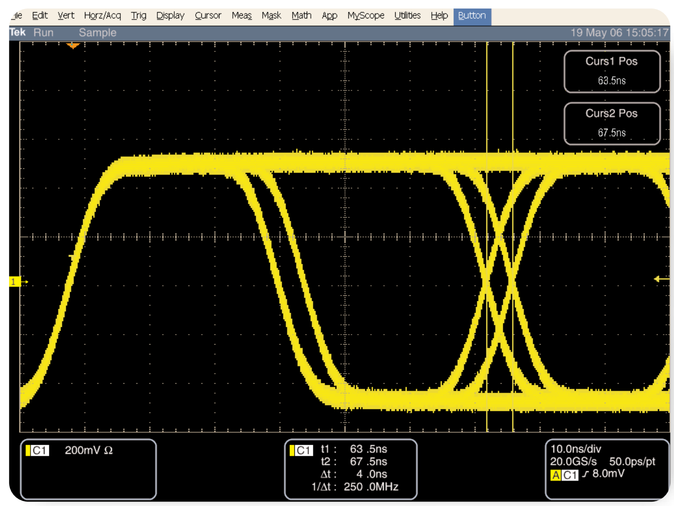
▶ **Figure 5.** *This sine wave signal from an AWG shows an aberration in every cycle. The AWG is reading every sample in the memory, which ensures that the aberration will repeat itself consistently.*



▶ **Figure 7.** *30 Mbps random pattern from an AWG running at 30 MS/s.*



▶ **Figure 6.** *This sine wave from an AFG fails to reproduce the aberration on some cycles, because it is skipping the samples that define the transient.*



▶ **Figure 8.** *30 Mbps random pattern from an AFG running at 250 MS/s. The jitter value is the reciprocal of 250 MS/s., that is, 4 ns.*

Signals that Include Aberrations

Figure 4 is strictly a "textbook" example. Depending on the algorithm and the frequencies involved, the DDS will select different points to skip, so the dichotomy between the red and black samples will not apply in every case. Figures 5 and 6 are actual screen shots that underscore the differences in the two sampling and waveform reconstruction architectures.

Pseudo-random Bit Stream (PRBS) Pattern Generation

Jitter is an issue when generating pseudo-random bit stream (PRBS) pattern using a DDS-based AFG and its fixed sample rate. Simply stated, the AFG tends to apply one sample period worth of jitter to fast-changing pulse edges both rising and falling*3. If, for example, the AFG's sample rate is 250 MS/s, then 4 ns of jitter will appear on the signal edges. The jitter value is the same as the sampling period of the AFG.

The jitter appears because the AFG has a fixed sampling rate, which is not a multiple of the data rate. Here again, the AWG is not subject to this limitation (although any real-world signal source will produce some jitter).

*3 Sine waves and other signals with slower transktions are not affected.

Point/Counterpoint

As always, the ultimate choice of the tools depends on the application. There is always a temptation to go for the "best numbers," which when applied to sample rate and memory depth means the biggest numbers. Astute users will instead make a choice compatible with the application's actual signal requirements.

For example, certain mid-range AFG's offer 1 GS/s sample rates while some AWGs in the same class are limited to 600 MS/s. But when the application requires reliable delivery of small signal details at a wide range of frequencies, the AWG is the preferred tool. Because the AWG reads every sample point on the stored waveform, you can be sure that transients, edge risetimes, and even noise effects will be reproduced accurately.

Moreover, the AWG is the better tool for sourcing low-jitter digital waveforms such as pseudo-random bit streams (PRBS). That makes it the best solution for many serial bus measurement applications.

Inevitably there are a few tradeoffs. Editing the number of samples to increase the output frequency, as in the case of AWG#1 described earlier, is less convenient than the AFG method of changing one setting to alter the frequency.

And because the AWG architecture relies on one variable master clock across all of its channels, generating differing frequencies simultaneously across several channels requires storing a different waveform file behind each channel.

If there is a need, for example, to generate a 10 MHz sine from Channel 1 and at the same time a 20 MHz sine wave from Channel 2, then the Channel 2 waveform memory must be loaded with two cycles. Therefore, when the clock steps through the memory, two cycles will emerge from Channel 2 for every single cycle from Channel 1, doubling the output frequency. This process grows more complicated when the differing frequencies are not simple multiples of the base frequency.

The AFG offers a different set of strong points. Its phase noise specifications and frequency agility tend to be superior to those in AWGs. In some leading AFG models, the master clock is manipulated independently by the DDS element in each channel, making it easy to deliver multiple frequencies at once. And AFG's are usually the most affordable solution among the available choices. Arbitrary function generators have become the mainstay of general-purpose signal sources.

The AFG is less suited to applications requiring low jitter and very narrow transients. The platform may not suffice for PRBS applications since the innately higher jitter on its output waveform can cause an erratic response in DUT receiving elements. And for stress tests requiring predictable signal distortions, the AFG's sample-skipping technique can produce misleading results at some frequencies.

Conclusion

As is often the case, the choice between the AFG and the AWG is one of selecting the most appropriate of two strong contenders.

- Choose the AFG when the application calls for clean, regular waveforms, and/or fast switching from frequency to frequency, or when multiple channels must deliver differing frequencies simultaneously.
- Choose the AWG for the most complex signals: PRBS streams, modulated RF signals, and more. When the source must reliably produce aberrations, controlled jitter, and noise in every operational cycle at every available frequency, the AWG is the better tool.

Contact Tektronix:

ASEAN / Australasia (65) 6356 3900
Austria +41 52 675 3777
Balkan, Israel, South Africa and other ISE Countries +41 52 675 3777
Belgium 07 81 60166
Brazil & South America 55 (11) 3741-8360
Canada 1 (800) 661-5625
Central East Europe, Ukraine and the Baltics +41 52 675 3777
Central Europe & Greece +41 52 675 3777
Denmark +45 80 88 1401
Finland +41 52 675 3777
France +33 (0) 1 69 86 81 81
Germany +49 (221) 94 77 400
Hong Kong (852) 2585-6688
India (91) 80-22275577
Italy +39 (02) 25086 1
Japan 81 (3) 6714-3010
Luxembourg +44 (0) 1344 392400
Mexico, Central America & Caribbean 52 (55) 5424700
Middle East, Asia and North Africa +41 52 675 3777
The Netherlands 090 02 021797
Norway 800 16098
People's Republic of China 86 (10) 6235 1230
Poland +41 52 675 3777
Portugal 80 08 12370
Republic of Korea 82 (2) 528-5299
Russia & CIS +7 (495) 7484900
South Africa +27 11 254 8360
Spain (+34) 901 988 054
Sweden 020 08 80371
Switzerland +41 52 675 3777
Taiwan 886 (2) 2722-9622
United Kingdom & Eire +44 (0) 1344 392400
USA 1 (800) 426-2200
For other areas contact Tektronix, Inc. at: 1 (503) 627-7111
Updated 12 May 2006

Our most up-to-date product information is available at: **www.tektronix.com**

7/06 JS/WOW 76W-19764-1

Tektronix
Enabling Innovation