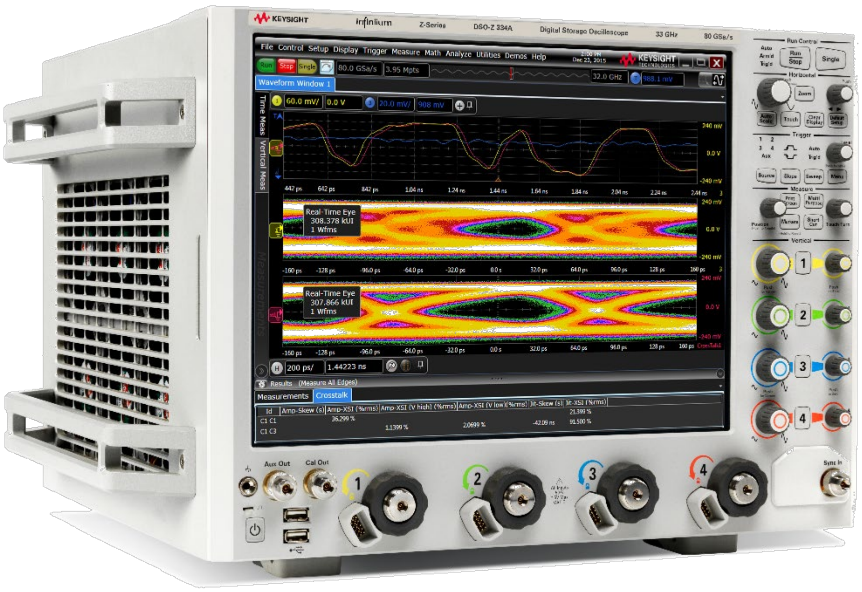
DATA SHEET

N8833A and N8833B Crosstalk Analysis Application for Real-Time Oscilloscopes



This document is discontinued and preserved for legacy customers.

The N8833A and N8833B have been replaced by D9020ASIA. Information herein, may refer to products/services no longer supported. We regret any inconvenience caused by obsolete information.

For the latest information go to: https://www.keysight.com/us/en/assets/7018-07031/data-sheets/5992-3383.pdf

Introduction

Increased data communication speeds as well as the increased circuit density of mobile devices has led to very high serial data rates on multiple lanes, placed very close together. This combination of higher bit rates, spaced tightly together, leads to an increased amount of crosstalk distortion between serial data signals. As a result, crosstalk is becoming a more important problem to diagnose and quantify. Integrated circuit power supplies are also increasingly susceptible to crosstalk at these higher serial data rates. Simultaneous switching noise (SSN) and ground bounce from serial data signals and switching power supplies can create perturbations on the supplies that distort the data lanes they drive in the form of noise and jitter.

The need to troubleshoot and characterize crosstalk is not new, but the legacy methods of measuring crosstalk in digital communications systems has relied on the process of selectively disabling some channels while enabling others. This necessarily requires measuring the crosstalk effects in the system while operating in special test modes, which means measuring them under abnormal conditions. Worse yet, some systems cannot even operate the necessary special modes.

Keysight Technologies, Inc. has developed a crosstalk analysis application to assist in the diagnosis and quantification of crosstalk that does not require the system under test to operate in any special test modes. The application works by constructing optimal models of the crosstalk mechanisms between multiple measured signals. The user, for example, can acquire up four signals simultaneously of a running system and then configure the application to calculate the optimal crosstalk models between them. Once these models are known, various victim signals can be displayed with and without the crosstalk from each aggressor. In this way, noise, jitter and eye-diagram measurements can be made on the victim with and without crosstalk to quantify the amount each aggressor contributes to each of these measurement values.

Features of the crosstalk analysis application include:

- Analysis of up to four signals (aggressors or victims) at once.
- No crosstalk simulation or model is required when performing crosstalk analysis.
- Identify and report the amount of crosstalk present on victims from each aggressor.
- Plot waveforms without crosstalk. Compare them with the original waveforms using scope tools such as eye diagram and jitter separation to see how much margin can be recovered.
- Setup wizard to guide user through setting up the aggressors and victims and running the crosstalk analysis.

In another scenario, the user could move around their running system by probing various suspected aggressor signals while monitoring their victim signal until they identify which suspected aggressor signals in their system are corrupting their victim signals.

The application not only detects and quantifies the presence of crosstalk, but it can also determine the relative magnitude of error each aggressor imparts to the victim. It can also go one step further by actually removing the crosstalk from the victim so you can visually compare the original waveform with the clean waveform side-by-side. You can compare the “before” and “after” waveforms directly on the scope display or by comparing the results from other scope analysis tools such as real-time eye diagrams or jitter analysis. This approach gives you a direct way of quantifying the amount of improvement you can expect by mitigating the different sources of crosstalk.

The crosstalk analysis application can provide a lot of valuable insight into your design. For instance, it can help engineers determine the margins the design would recover without the crosstalk. It can also help determine if a signal that fails design specification can now pass without the crosstalk. This approach can lead to important design decisions on whether it is worth the time and effort to improve the crosstalk effect and where in the board to make improvements.

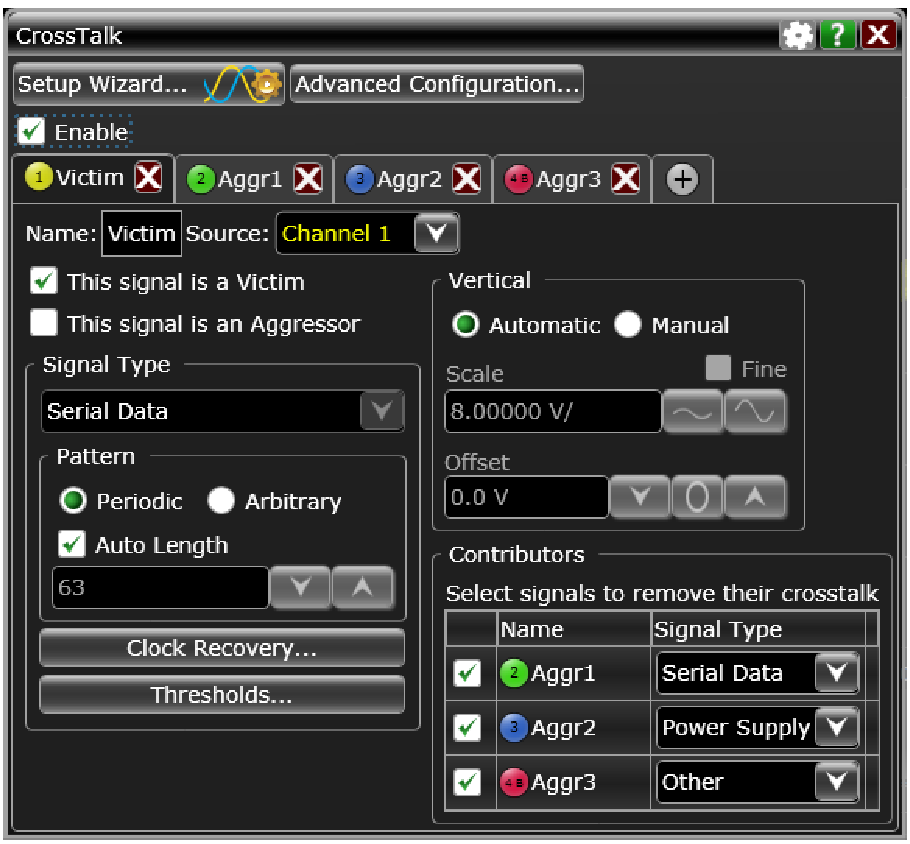


Figure 1. Keysight crosstalk analysis application for analyzing serial data, power supply and other sources of crosstalk effect to the victim signal. Up to four signals can be analyzed at once.

Types of Crosstalk

Transmission line crosstalk

Parallel transmission lines are prime candidates for crosstalk. The mutual inductance and capacitance between them allows energy to couple from one lane into another. The voltage generated by capacitive coupling creates a current that travels in both directions, whereas the inductive coupling creates a current that travels only in the reverse direction. When added together, the two currents reinforce each other in the reverse direction but cancel each other out (at least to some extent) in the forward direction. The reverse traveling wave returns back to the transmitter end and is called “near-end” crosstalk (NEXT), while the forward traveling wave arrives at the receiver end and is called “far-end” crosstalk (FEXT). The magnitude and shape of these waveforms are quite different from each other.

Power supply aggressor crosstalk

Power supplies can also be a significant source of interference on a data line, creating both noise and jitter. This type of interference may simply be referred to as “power supply noise,” but its effects are similar to traditional crosstalk, and so it is straightforward to think of a power supply as just another source of crosstalk.

Noise and voltage drift in a power supply can affect the timing of the serial data waveforms they are driving. Timing errors can occur through a number of different mechanisms, such as phase changes which are caused by voltage-dependent driver impedances and frequency changes caused by voltage-controlled oscillators. The resulting jitter is called power supply induced jitter (PSIJ).

In another case, the power supply may be directly connected to the transmission line when the logic level switches to that value. Noise and bias on the voltage rail can therefore transfer directly to the bit stream. While this seems straightforward, it is more complicated than the parallel transmission line case. The interference may be present only when the bit stream is at a particular logic level (for Vcc aggressors it might present only when the logic level is high, whereas for GND aggressors it might present only when the logic level is low). It is therefore possible to have the high voltage bits experience a lot of noise while the low bits have little (or vice versa). This is a non-linear (or voltage-dependent) type of interference.

Power supply victim crosstalk

The signal integrity of power supply voltages (including ground) can be affected by the circuits they are driving. One common example of this is simultaneous switching noise (SSN), which can produce ground bounce (the Vcc rail can also "bounce" and may be referred to as Vcc sag). SSN can occur as a result of parasitic inductances that lie between the device (chip) ground and the system (board) ground.

As the voltage on the output changes state, it draws a current through the switching transistors. As that current flows to ground, it causes a voltage drop across the parasitic inductances. That voltage drop in turn changes the voltage you would measure at the device ground. The voltage measured there may bounce up and down in correlation with data transitions. The effect is amplified when more than one line is switching states at the same time since this will draw more current. In addition depending on the impedance and the various switching delays, the ground bounce may appear to have a ringing effect as well.

Crosstalk Network Models

The crosstalk application constructs different types of crosstalk models from the measured waveforms depending on the victim and aggressor signal types. In the simplest case, when both victim and aggressor are serial data signals, the model is a linear n-port transfer function model. Figure 2 is simple example of a 2-channel serial data system. Physically, the transmission system consists of two transmitters, two receivers and a 4-port linear network model connected between them. Probing the two serial data channels provides the input waveforms to the crosstalk application.

The application begins analyzing the waveforms by extracting the serial clock and data pattern of each. It then uses the ideal data pattern as input to the 4-port linear crosstalk model shown in figure 3. It computes the best-fit transfer function model, which transforms the ideal waveforms into the actual measured waveforms. Within the model, h_{11} and h_{22} represent the signal's ISI while h_{21} and h_{12} represent the coupling or crosstalk between the two signals. This approach allows the crosstalk analysis application to remove the effect of either the crosstalk, ISI or both crosstalk and ISI from the measured waveforms.

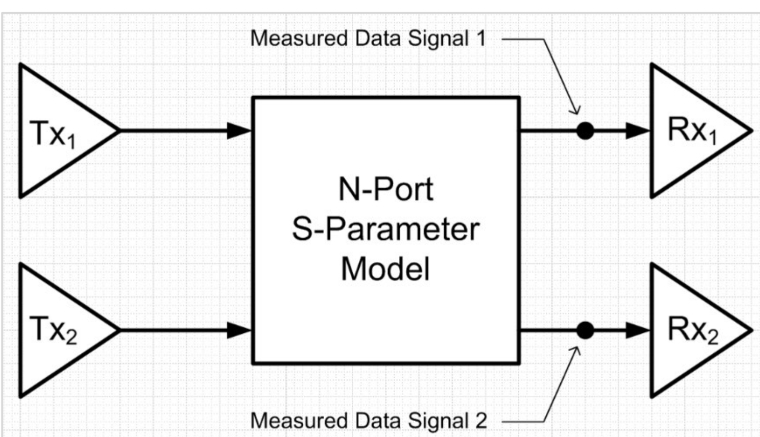


Figure 2. A 4-port network model for two parallel transmission lines. The black dots show possible probe locations at the far end.

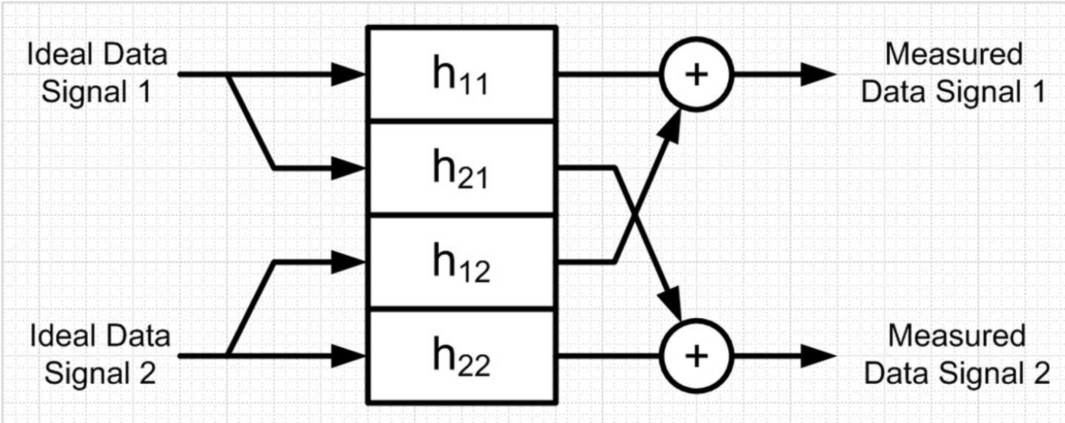


Figure 3. The crosstalk analysis application computes a 4-port linear crosstalk model, which transforms the ideal waveforms into the actual measured waveforms.

Crosstalk Analysis Challenges Solved by Keysight Application

One way to uncover the network model behavior and estimate its parameters is to selectively turn off the victim and/or aggressor waveforms. For example, you could visualize the crosstalk that channel 1 receives by turning off channel 1 and turning on channel 2. The resulting waveform on channel 1 would be pure crosstalk (and noise). Alternatively, you could do the reverse by turning on channel 1 and turning off channel 2. The resulting waveform on channel 2 would represent a clean output without any crosstalk from channel 1. To quantify the relationships, you could do measurements on the different waveforms, such as peak-to-peak voltage measurements or eye diagrams.

While straightforward, there are a number of problems with this approach. First, it may not be possible or convenient to selectively enable or disable the different data streams. Second, the approach is tedious and time consuming. In a 4-lane system, for example, there are 16 victim-aggressor pairs (if you include ISI), which leads to a large number of test setups, measurements and results that are seen separately and must be manually recorded. Third, because the measurements are not simultaneous, any constructive or deconstructive interference effects will be lost. Fourth, the approach simply would not work if the source of interference comes from outside the network, such as from a power-supply that cannot be disabled.

The Keysight crosstalk tool solves these problems by estimating the network model from a single test setup using a simultaneous acquisition of all data lanes. The model can be constructed by simply probing each transmission line using the oscilloscope and then enabling the application. After constructing the model, the application will display some measurement results, such as the magnitude of the crosstalk for each victim-aggressor pair, and will also display a crosstalk-removed or “clean” waveform. The crosstalk-removed waveform simulates what the waveform would look like if the specified aggressors were disabled. The waveform is treated like any other waveform on the scope and can therefore be used as an input to any other oscilloscope measurement or analysis. You can easily see, for example, the before and after eye diagrams or compare the before and after jitter components using EZJIT Plus.

Test Setup

1. Probing

To create a test setup for crosstalk, simply probe all of the transmission lanes and power supplies that are to be included in the analysis. The application currently supports a total of four input waveforms.

For serial data signals, it is best to probe all the victim lanes at the receiver end. This captures the actual waveforms that are delivered to the receiver. For perfectly-terminated transmission lines, this probing scheme will capture primarily FEXT since NEXT travels back toward the receiver end. However, it's possible for a received waveform to contain elements of both FEXT and NEXT. To specifically analyze NEXT, one can probe at the transmitter end. In the crosstalk tool you do not need to specify a NEXT or FEXT measurement; The application will measure the crosstalk that is present regardless of what type it is.

For power supply aggressor crosstalk, it is also best to probe the victim lanes at the receiver end. The power supply aggressor can be probed anywhere. Keep in mind that supplies of both the transmitter and receiver can contribute to crosstalk.

For power supply victim crosstalk, it does not matter where you probe the victim or the aggressors in this case. It is possible to have up to three aggressors, but all of them must be serial data aggressors. This requirement is because ground bounce is a result of the current drawn from a supply during data transitions, so it is necessary to find those transitions. Only one power supply victim can be analyzed at a time due to the complexity of the algorithm.

2. Synchronization

Another consideration in the test setup is synchronization. Because the application is looking for correlations between victims and aggressors, all victims and aggressors need to be acquired simultaneously. That application accommodates time skews between victims and aggressors. However, minimizing this skew can improve calculation time measurement results.

3. Test patterns

The crosstalk application will work with arbitrary serial data patterns, though periodic patterns produce better results with a shorter calculation time. Note also that in order to distinguish between distortions cause by ISI and the various aggressors, each serial data channel must transmit different patterns or run at different data rates. Note that longer data patterns are not necessarily helpful or desirable in the crosstalk analysis tool. One reason is simply practicality. A jitter analysis tool looks only at timing errors in the data, so it only needs to process one data point per data edge (transition), whereas the crosstalk tool uses all of the waveform data. Therefore, moderate lengths up to $2^{11} - 1$ or less are recommended.

Examples of Crosstalk Analysis Results

1. Transmission line crosstalk result example

Figure 4 shows the victim and aggressor waveforms (in green and orange) in the top plot along with the crosstalk-removed waveform in red. The middle plot shows the eye diagram of the victim waveform with the tall crosstalk peak or bulging on the left side of the eye indicative of FEXT. The bottom plot shows the eye diagram for the crosstalk-removed waveform.

In this particular example, the crosstalk magnitude of the victim-aggressor (M2 M3, Amp-XSI (%rms)), which is the RMS value of the crosstalk divided by the RMS value of the measured waveform reported, shown at the bottom of the figure is 14%. The eye opening is significantly improved when the crosstalk is removed from the victim waveform.

The victim ISI magnitude (M2 M2, Amp-XSI (%rms) shows about 2%, and it is not a significant contributor to the victim waveform distortion.

Figure 4. Before and after view of a victim waveform with FEXT. The top plot shows the victim waveform in green, the aggressor in orange, and the crosstalk removed waveform in red. The middle plot is the eye diagram of the victim with the bulging indicative of FEXT, and the bottom plot is the eye diagram of the victim after removing crosstalk.

2. Power supply aggressor crosstalk result example

The following example illustrates the effect a noisy power supply can have on the jitter of a serial data signal. Figure 5 shows the serial data victim and power supply aggressor waveforms (in yellow and blue) in the top grid along with the crosstalk-removed waveform in red. The power supply is creating excessive jitter on the victim, as seen in the upper eye diagram. In this case, the perturbations in the power supply change quite slowly relative to the data signal's unit interval, but by scrolling through the waveform, you can see that as the power supply increases in voltage, the data signal is delayed in time. As the power supply decreases in voltage, the data signal advances in time. In this example, the application created a model for the jitter in the data signal that is attributed directly to the voltage variation of the power supply and removed that crosstalk error from the victim in the lower eye diagram. The eye diagram of the crosstalk-removed waveform at the bottom is considerably improved from the one above it, showing that the power supply aggressor was responsible for most of that jitter.

This particular example does not have a lot of amplitude crosstalk from the power supply, as shown by the low values of C1 C3, Amp-XSI (V high) and C1 C3, Amp-XSI (V low). These values represent the relative magnitudes of crosstalk distortion caused by the aggressor power supply on the serial data victim. There is, however, a significant amount of jitter in the data signals caused by the power supply, as shown by the large C1 C3, Jit-XSI value.

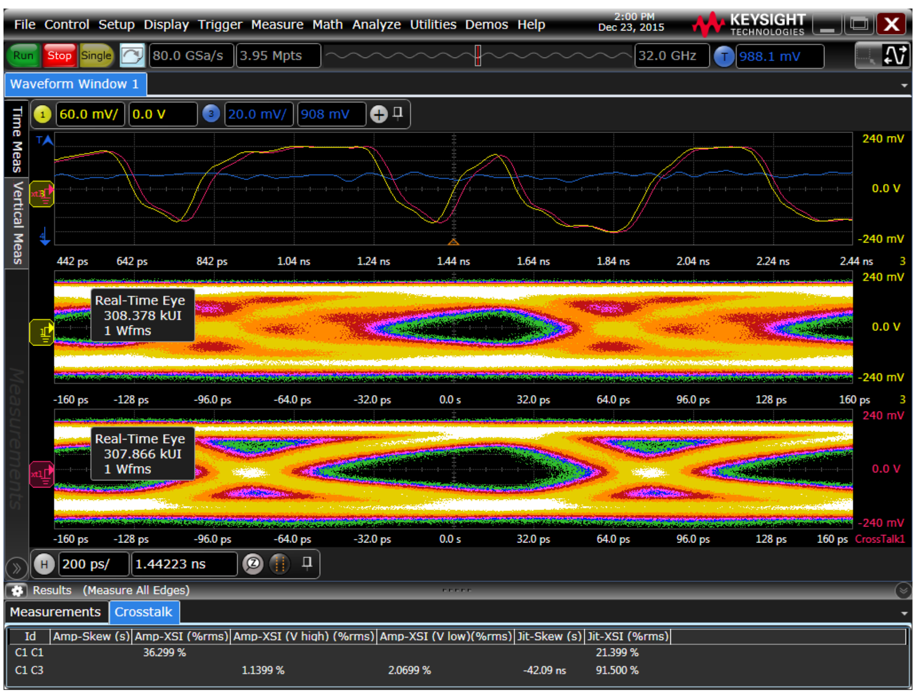


Figure 5. Crosstalk example of a power supply aggressor generating jitter on a serial data victim.

Taking the analysis a step further, EZJIT Plus can be used to quantify the timing improvement by comparing the jitter statistics of the before and after waveforms. Figure 6 shows a capture of the RJPJ histogram (bottom) for the original serial data waveform. The power supply has a strong periodic component that is showing up primarily as a large PJdd value in the jitter statistics (at the very bottom). The large PJdd value also creates large PJrms and DJdd values.

Figure 6. Jitter analysis results using EZJIT, showing the large amount of jitter in the original serial data waveform.

Figure 7 shows the same results for the crosstalk-removed waveform. Note the improvement in the RJPJ histogram and the large reduction in PJdd, and related statistics, such as DJdd and PJrms.

Figure 7. Jitter analysis results using EZJIT, demonstrating the improvement in jitter statistics after removing the jitter generated by the power supply.

3. Power supply victim crosstalk result example

This example demonstrates the crosstalk analysis application's ability to identify any crosstalk corruption of a power supply. Although this example uses the Vcc rail, the methods and results can apply to either supply rail. Figure 8 shows a power supply waveform (in yellow) that is a victim of crosstalk from a clock signal aggressor (in green). Note the large ringing in the power supply that is correlated with the edges of the clock. The red signal is the power supply signal after removing the crosstalk generated by the clock edges. The ringing has been removed. The crosstalk magnitude of the victim-aggressor (C1 C2, Amp-XSI), which is the RMS value of the crosstalk divided by the RMS value of the measured waveform reported, shown at the bottom of the figure is more than 95%, which indicates a strong correlation of the clock edges aggressing at the power supply.

Figure 8. Example of a power supply being treated as a victim. Victim waveform (yellow), aggressor (green) and crosstalk-removed waveform (red).

Oscilloscope Compatibility

Oscilloscopes	Software revision
Z-Series	5.60 or higher
V-Series	5.60 or higher
S-Series	5.60 or higher
90000 Series	5.60 or higher
9000 Series	5.60 or higher

Related Literature

Publication title	Publication number
Infiniium Z-Series Oscilloscopes - Data Sheet	5991-3868EN
Infiniium V-Series Oscilloscopes - Data Sheet	5992-0425EN
Infiniium S-Series High-Definition Oscilloscopes - Data Sheet	5991-3904EN
Infiniium 90000A Series Oscilloscopes - Data Sheet	5989-7819EN
Infiniium 9000 Series Oscilloscopes - Data Sheet	5990-3746EN
E2688A, N5384A High-Speed Serial Data Analysis and Clock Recovery Software For Infiniium Oscilloscopes - Data Sheet	5989-0108EN
EZJIT Complete Jitter and Vertical Noise Analysis Software for Infiniium Oscilloscopes - Data Sheet	5991-0523EN
N5461A Serial Data Equalization Software for Infiniium Series Oscilloscopes - Data Sheet	5990-3330EN
D9020ASIA Advanced Signal Integrity Software for Infiniium Oscilloscopes - Data Sheet	5992-3383EN
D9010/20JITA Timing, Vertical, and Phase Noise Jitter Analysis - Data Sheet	5992-3376EN
Infiniium UXR-Series Oscilloscopes - Data Sheet	5992-3132EN
Infiniium MXR-Series Oscilloscopes - Data Sheet	7120-1115EN

For copies of this literature, contact your Keysight representative or visit www.keysight.com/find/scope-apps.

Learn more at: www.keysight.com

For more information on Keysight Technologies' products, applications or services, please contact your local Keysight office. The complete list is available at: www.keysight.com/find/contactus