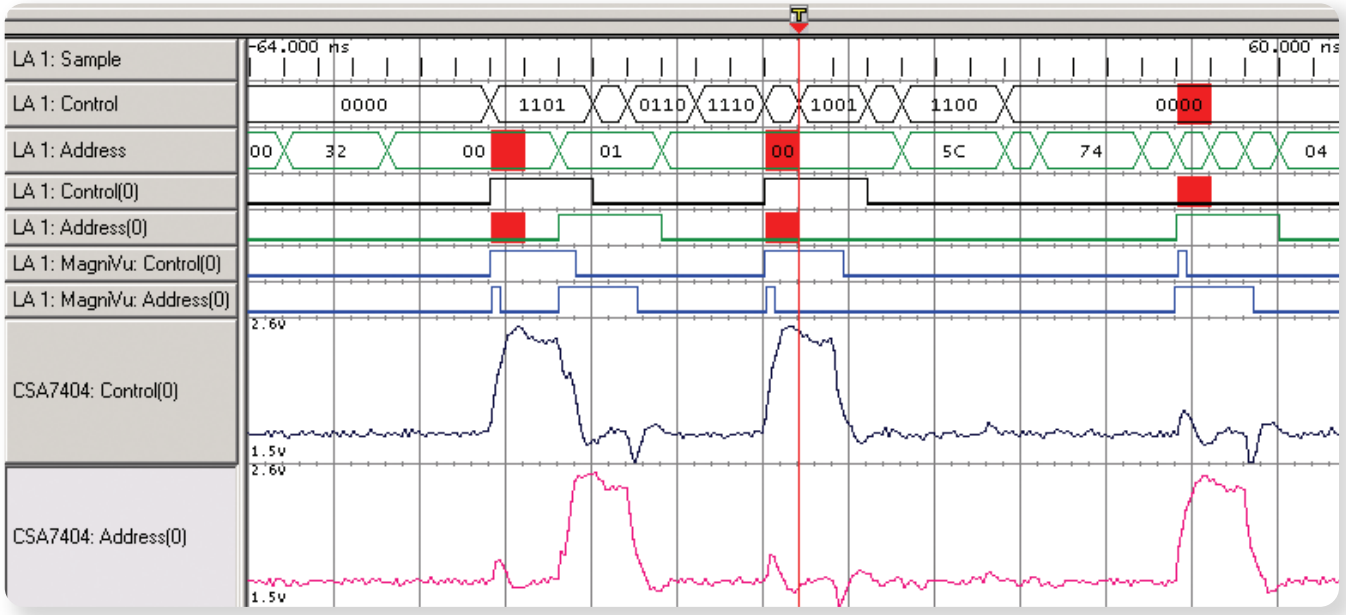
Timing Error Debugging



New Designs, New Headaches

New digital devices have become progressively more powerful by incorporating high-speed buses, subsystems, and logic families. They have also become more complex, more sensitive to signal quality, and more time consuming to troubleshoot. Tight schedules do not allow extra time for debugging. This note explains how to speed up troubleshooting by using more of the features in your logic analyzer and oscilloscope.

Introduction

Today's designs are harder to debug because there are more factors that can go wrong. Consider the emergence of high-frequency buses. Their high-speed digital edges are very sensitive to signal integrity. They can cause problems even when your device does not use faster clocks. Today's logic families can introduce high-speed digital edge rates even at slower clock rates.

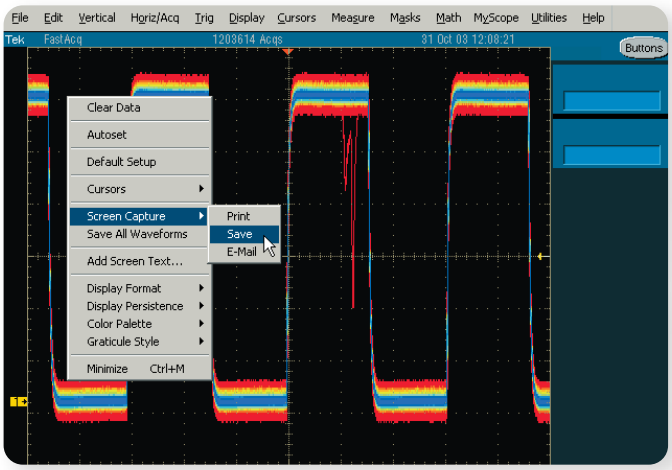
Fast edges can also increase crosstalk. On older designs, you could take the stability of circuit board traces for granted. High edge rates, however, can make them act as transmission lines, sending and receiving interference. Faster edges also create larger transient currents. Dynamic currents from these transients can induce ground bounce and power distribution artifacts. Fast edge faults usually appear in your signals as intermittent glitches.

Tektronix
Enabling Innovation

Before you can solve such problems, you have to find the effects, characterize them, and work back to determine their cause.

Other project-stopping culprits include timing violations, driver errors, and race conditions. All of these can create similar faults in state machine logic. Furthermore, they can show up in any circuit. Since they may, or may not occur when you are capturing signals, they are exceptionally difficult to resolve.

In this application note we will discuss several timesaving tips to help improve your productivity in the digital debugging stage as you address fast edge effects and intermittent challenges.



▶ **Figure 1.** *A Tektronix DPO Oscilloscope maximizes the probability of capturing elusive glitches and other infrequent events.*

▶ **Figure 2.** *Advanced and a wide selection of oscilloscope triggering increases efficiency to more quickly and easily capture different type of signals.*

Tip #1: Look for Glitches

What is a glitch?

If your device is malfunctioning, a good way to begin troubleshooting is to check for glitches. Glitches are very narrow pulses that your system may, or may not, interpret as logic changes. Most problems will appear as glitches in one or more of your signals. The effect of glitches on system operation is unpredictable. They can be your first sign of a wide variety of device faults, including race conditions, termination errors, driver errors, timing violations, and crosstalk.

Locating the problem

Since the problems that glitches cause are often intermittent, they can be very hard to resolve. A solid approach is to combine classic top/down troubleshooting with the specific advantages of your test instruments. Start wide, with a macro view of device operation, and then begin focusing in on problems.

Glitch hunting is a good illustration of this approach. One of the most powerful and easy to use glitch hunting tools when no more than 4 channels needs to be simultaneously investigated is using a DPO oscilloscope. A Digital Phosphor Oscilloscope (DPO) provides unmatched insight into signal behavior by displaying, storing and analyzing complex signals in real-time using three dimensions of signal information: amplitude, time and distribution of amplitude over time. With its continuous waveform capture rate of up to 100,000 it is the industry's most efficient tool to capture elusive signal glitches/intermittent signals and ideal for digital debug. A DPO oscilloscope is also extremely easy to use and in an instant you'll have a real time visual overview of what's really happening in the signal and you can then use the oscilloscopes advanced triggering capabilities to further characterize the problem. All Tektronix oscilloscopes offer extensive trigger possibilities that allow you to quickly and easily trigger on the event of interest and further analyze it.

If you need to perform glitch hunting on more than 4 channels, on the macro level your Tektronix logic analyzer allows you to perform glitch triggering on busses that are hundreds of signal wide. The logic analyzer checks every signal for glitches. Red bars on the bus timing diagrams show glitch locations for further analysis. You can then again use a Tektronix oscilloscope to help further characterize the problem by revealing exactly what the glitch looks like on a micro level. Using the iView™ measurements on the TLA5000 Series logic analyzers, you can combine your logic analyzer and your oscilloscope into a single system and progressively "zoom in" on the problem.

Using the top/down methodology, we can step through the debugging process to more easily find glitches and troubleshoot the problem. In the following four steps we will identify two different glitches and their likely source.

Step I: Examine the bus

Begin by focusing on what works and look globally for faults. Your logic analyzer's bus timing waveform will flag any glitches that occur.

When it comes to looking for intermittent effects, such as glitches, use a logic analyzer with a long record length. Tektronix logic analyzers can have up to 256 Mb of deep timing capability. The logic analyzer's bus timing waveform can examine all the signal lines of the bus at once. If the logic analyzer detects a glitch on any of the lines, it will flag the bus and the time location.

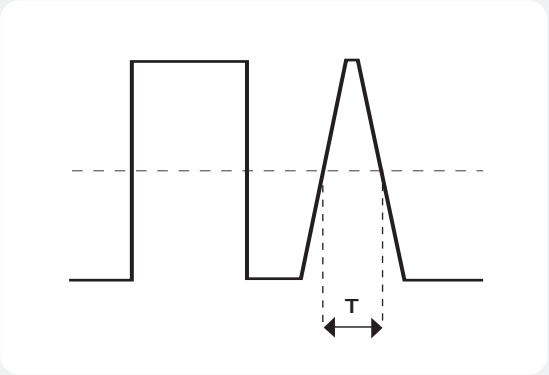
In Figure 3, the top waveform is Sample, which shows the sample ticks that represent the logic analyzer's deep timing sampling rate of up to 2 GHz (500ps). The next two lines are the bus waveforms – the 4-bit Control bus and the 8-bit Address bus. The red glitch flags that appear on both bus waveforms signify that there was more than one transition between the sample points at those locations.

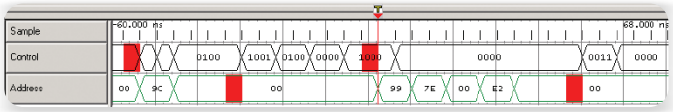
Step II: Examine the lines

Now focus on where the problems are. Use the logic analyzer's timing signal waveform to display the individual lines of the bus and flag where glitches occur. Expand the logic analyzer's deep timing signal waveform, which can also be up to 256 Mb deep.

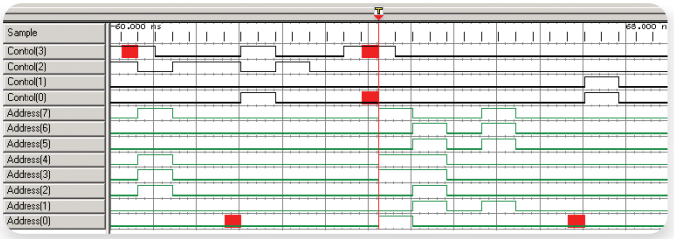
Glitch Triggering

Glitch triggering allows you to trigger or reject glitches of positive, negative or either polarity when they are shorter or longer than a user-defined time limit (TDS5000B minimum glitch width is 1.0 ns with 200 ps resolution). This trigger control enables you to examine the causes of even rare glitches and their effects on other signals.



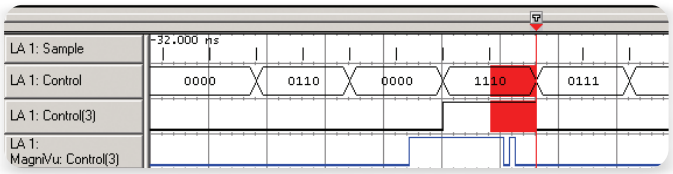


▶ **Figure 3.** *Sample ticks. Control bus and Address bus showing red glitch flags.*

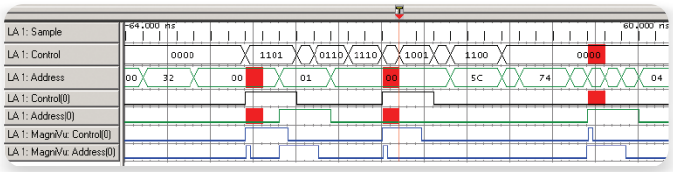


▶ **Figure 4.** *4-bit Control bus and 8-bit Address bus expanded showing red glitch flags on individual signals.*

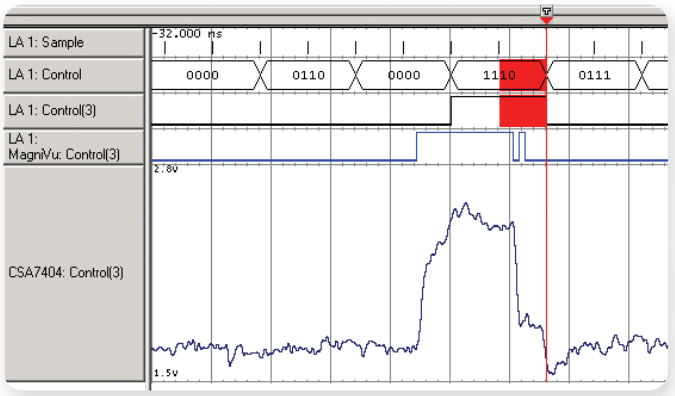
In Figure 4, the analyzer has expanded the Control bus into its four individual signals and the Address bus into its eight individual signals. The red glitch flags from the bus waveform in Figure 3 are now shown as glitch flags on signal lines Control (3) and Control (0), and as two glitches on the Address (0) signal line.



▶ **Figure 5.** *MagnuiVu™ waveform of Control (3) showing glitch.*



▶ **Figure 6.** *Control (0) and Address (0) lines with MagniVu traces showing glitches caused by crosstalk.*



▶ **Figure 7.** *iView oscilloscope trace showing analog representation of Control (3) signal.*

Step III: Take a closer look

Use a high-resolution timing view to examine the faults in fine detail. See how they relate to other events or faults. In addition to deep timing, Tektronix logic analyzers have high-resolution MagniVu™ 125 ps (8 GHz) timing capability that runs simultaneously with the deep timing capability. MagniVu waveforms can display all channels in high-resolution up to a 16 Kb memory depth. It is like having two logic analyzers in one: a deep timing logic analyzer and a high-resolution timing logic analyzer, both using the same probes.

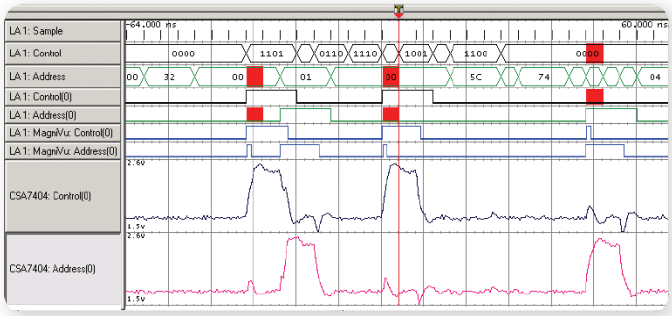
In this example, it appears there may be two different problems causing glitches. First, focus on the Control (3) signal line and show the MagniVu trace for the Control (3) signal. Figure 5 shows that because of its higher resolution, MagniVu waveforms can reveal that the glitch only appears at the end of a digital pulse – not at the beginning of the pulse and not by itself. This is an important clue to the cause of the fault. Proceeding to Step IV will reveal the likely cause.

Now focus on the second glitch identified in Control (0). Use MagniVu high-resolution timing to examine the two remaining flagged lines, Control (0) and Address (0). Figure 6 shows that since MagniVu waveforms are examining the signals at a much higher resolution, 125 ps, it is able to discern far narrower glitches on both lines. Note that the glitch and a pulse occur at the same time on both signal lines. That often indicates crosstalk between the two signals, but you need to take a different kind of close-up look to be sure. Proceeding to Step IV will also reveal more about this.

Step IV: Examine the analog waveform

Discover what the glitch really looks like by comparing the analog and digital perspectives using your oscilloscope and the logic analyzer's iView™ capability. iView capability allows the logic analyzer to trigger the oscilloscope at exactly the right time to capture the glitch. With iView measurements, the logic analyzer also time-correlates the data and displays both the analog and digital waveforms on the logic analyzer's display.

Looking at signal line Control (3), Figure 7 shows an analog iView display of the glitch. Considering both domains, it is obvious that something is distorting both the rising and falling edges of the pulse. The rising edge does not droop enough to trigger a logic transition and therefore has not appeared as a glitch. The falling edge, however, bounces high enough to pass through the logic threshold and sometimes act as a logic transition. Although the bus clock is not particularly fast, the LVPECL logic family that the circuit uses still introduces fast edges. The bouncing on the pulse edges suggests a termination problem on the circuit board magnified by the greater sensitivity of the fast edges.



▶ **Figure 8.** *Crosstalk between Control (0) and Address (0) shown using iView measurement.*

Testing the earlier crosstalk hypothesis on Control (0) and Address (0), Figure 8 shows that for every leading edge of one signal there is a corresponding positive voltage pulse on the other. This makes crosstalk between Control (0) and Address (0) the obvious diagnosis. Crosstalk can easily occur on adjacent runs or pins within the package. High frequency signals and clock edges have a greater susceptibility to crosstalk effects than lower frequency signals. This implies that even design practices that were consistently successful at slower frequencies can be a contributor to failures at higher frequencies.

Although the buses in these two examples were quite narrow, logic analyzer glitch triggering can be used on buses with hundreds of signals. The analyzer checks every signal line for glitches. If it flags a glitch, start focusing in on the problem until you determine the source of the glitch.

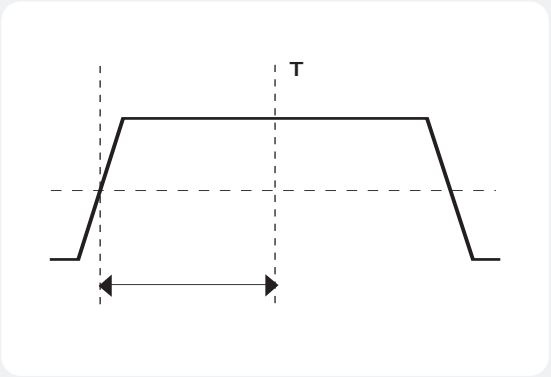
Tip #2: Use a Timeout Trigger to See What Isn't Happening

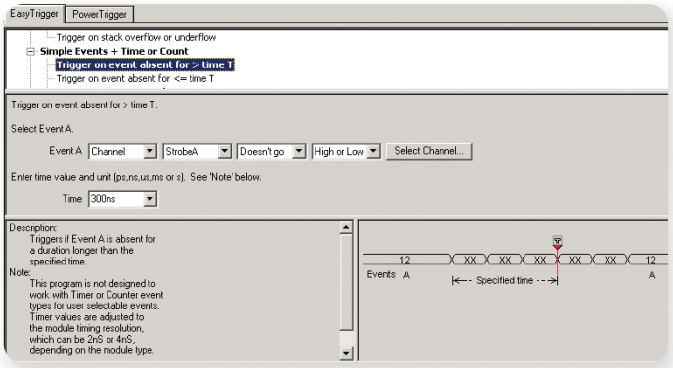
If your prototype's errors continue to elude you, go back to what is working. Consider the design's total operation. For example, you may have a signal that is supposed to do something periodically. Is it? This can be your key to capturing the problem, but only if your logic analyzer trigger can trigger on "nothing."

For example, you could have a strobe that provides a "local clock" function for a group of data lines. If the strobe does not act, or does not act often enough, then the device is not functioning as planned. Alternately, you could have embedded a "watchdog" or "heartbeat" pulse right into your system. As long as the heartbeat is pulsing then you know that the section is working. If the heartbeat stops, then you know when the failure became critical.

Time-out Triggering

Time-out triggering lets you trigger on an event which remains high, low or either without waiting for the trigger pulse to end, by triggering based on a specified time lapse.

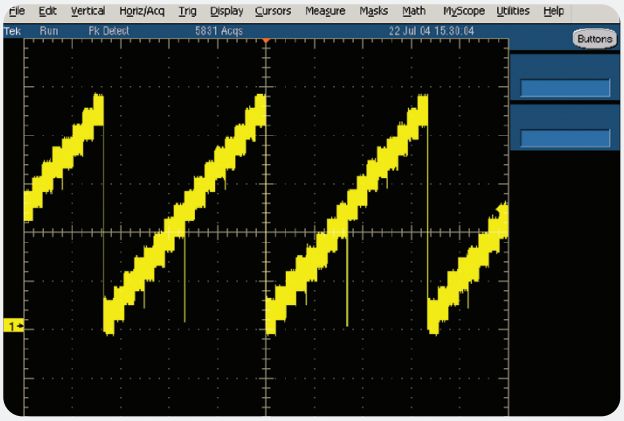




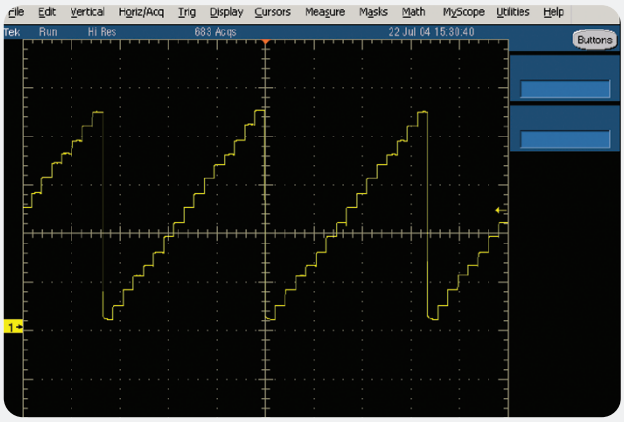
▶ **Figure 9.** *EasyTrigger timeout trigger definition screen.*

Fortunately, it is very easy to set your logic analyzer to trigger on "nothing" and to give you a detailed display of the state of the system.

Triggering on the absence of activity is called Timeout Triggering. You can set the analyzer to watch a line or group of lines; if nothing happens, if there are no logic changes in the period of time you specify, then the logic analyzer will trigger. You can also decide how deep your record of activity will be. Figure 9 shows the Timeout Trigger screen from the EasyTrigger menu in the logic analyzer. You can set it up in seconds.



▶ **Figure A.** *Peak Detect acquisition mode detects narrow glitches in slow signals.*



▶ **Figure B.** *HiRes acquisition mode shows low-level variations in slow signals.*

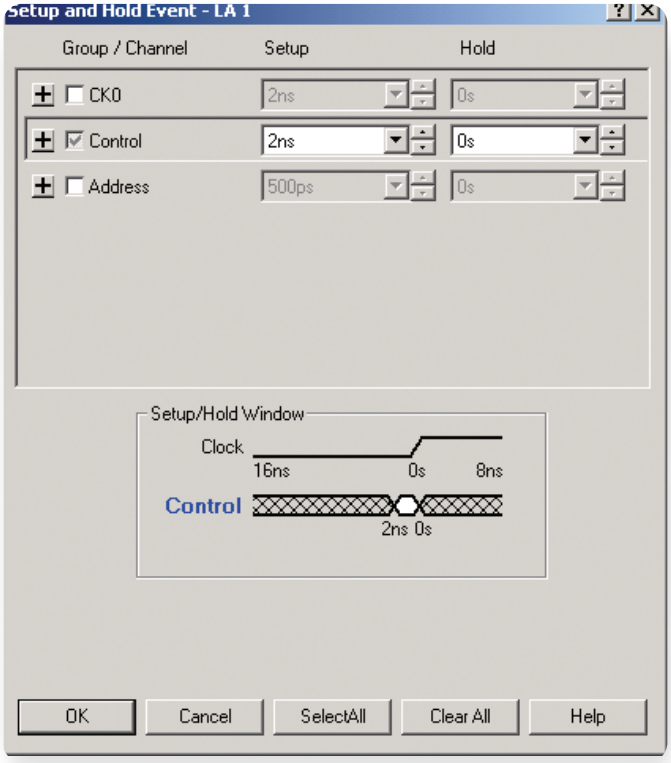
Searching for Glitches and Signal Variations in Slow Signals with an Oscilloscope.

Sometimes the signal itself can be relatively slow but have embedded glitches or low-level variations that are hard to detect. In the normal acquisition operating mode of a digital oscilloscope, the sample rate speed depends on the time/division setting. Consequently, when wanting to capture a certain time duration of a slow signal the oscilloscope also samples at low speed. This means that glitches can be undetected and not shown on the display. In other situations, low-level signal variations can be embedded in the signal and the normal vertical resolution is not sufficient to isolate this. In these situations rather than using a specific triggering mode, the use of specialized acquisition mode is more beneficial. The Tektronix oscilloscopes provide a suite of different acquisition modes to help gain insight into signal behavior such as described above.

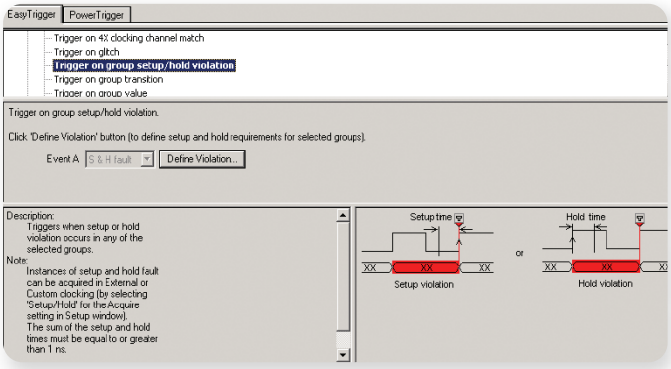
The peak detect acquisition mode, for example (Figure A), uses the instrument's maximum sample rate to reveal narrow glitches, even those less than 1 ns in width, at all sweep speeds. This provides an immediate solution to finding high-speed glitches in slow signal waveforms. Another acquisition mode is HiRes (Figure B) which stands for high resolution. This acquisition mode extends the oscilloscope's vertical resolution by filtering noise from low frequency signals and therefore shows signal behavior in more detail, at up to 12 bits resolution.

The cause of the failure may take place well before the heartbeat actually stops. The system could have continued on for some time before the effect of the fault became critical. By setting the trigger deep in the capture memory, you can acquire up to 64 Mb of pre-trigger information. Then you can analyze the record for possible causes.

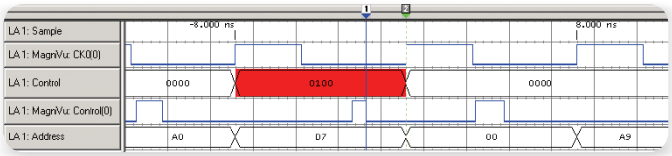
In a similar way to a logic analyzer, if the problem has already been isolated to a particular section/channel of the design an oscilloscope with timeout triggering can be used to trigger on an event which remains high, low or either, for a specified time period. In a Tektronix TDS5000B Series DPO the time period is selectable from 1 ns to 1 s with 200 ps resolution. Using an oscilloscope also provides further insight into exactly how the signals look.



▶ **Figure 10.** *EasyTrigger setup/hold violation trigger definition screen.*



▶ **Figure 11.** *Setup/hold violation trigger parameter selection dialog.*



▶ **Figure 12.** *Trigger display of set-up/hold violation on Control (0).*

Tip #3: Look for Setup/Hold Violations

Setup/hold compliance is one of the most crucial synchronous timing parameters, and a common source of errors. Searching for setup/hold violations can be very time consuming using the traditional approach of probing a clock and data line using an oscilloscope. The TLA logic analyzer can automate searching for setup/hold violations for you by triggering on and displaying any user-defined setup/hold violation on all your signals at once. Use the power of the TLA setup/hold violation trigger to watch all the signals in your system at the same time. The TLA will trigger on any violation and display all of the setup/hold violations in your system.

You can test for setup/hold violations directly by using your logic analyzer's Setup and Hold Violation trigger. Figure 10 shows the EasyTrigger setup/hold violation triggering setup menu. Using the logic analyzer's MagniVu high resolution of 125 ps, you can configure setup/hold window from 16 ns before the clock edge to 8 ns after the clock edge.

Figure 11 shows the dialog where you can specify the setup/hold violation parameters for the signals you would like to monitor. The TLA can monitor every signal in your system for setup/hold violations simultaneously.

Figure 12 shows the logic analyzer triggering on a setup violation of 1.875 ns before the clock edge. Now that you have identified the problem, you can work toward a solution.

All synchronous digital circuits have setup/hold requirements. Confirming setup/hold compliance should be part of your troubleshooting routine. Your logic analyzer's EasyTrigger Menu makes it a simple test to set up.

Similar to a logic analyzer, if the problem has already been isolated to a particular section/channel of the design an oscilloscope with setup/hold triggering can be used to trigger on violations of both setup time and hold time between clock and data present on any two input channels. Using an oscilloscope also provides further insight into exactly how the signals look.

Tip #4: Using an Oscilloscope and Specialized Jitter Analysis Software to Resolve Timing Problems

What About Jitter?

As seen in Tip #3, the logic analyzer or the oscilloscope can be used to identify any setup or hold violations in your system. This allows you to do a form of go/no go testing – if the logic analyzer or the oscilloscope triggered, then you know you had a violation. The next step is to better understand the source of these violations, and one potential cause is jitter.

Conceptually, jitter is the deviation of timing edges from their "correct" locations. In a timing-based system, timing jitter is the most obvious and direct form of non-idealness. As a form of noise, jitter should be treated as a random process and characterized in terms of its statistics.

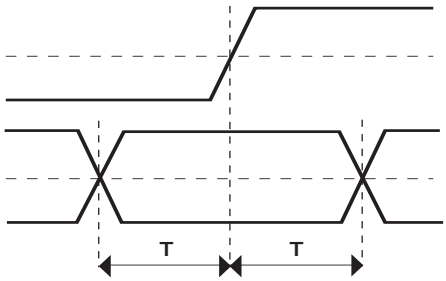
If you have a way to measure jitter statistics, you can compare components and systems to each other and to chosen limits. However, this alone will not allow you to efficiently refine and debug a design. Only by thoroughly analyzing jitter using an oscilloscope with specialized jitter analysis software is it possible for the root causes to be isolated, so that they can be reduced systematically rather than by trial and error.

Analyzing Jitter

In this example, the design has a Phase Lock Loop (PLL) oscillator configured as a "zero delay" clock source for its memory system. The PLL receives an external clock signal, locks on to the frequency, and retransmits the signal through a clock distribution network to the memory elements. In doing so, it corrects for all the known delays along the distribution path.

Setup-and-Hold Triggering

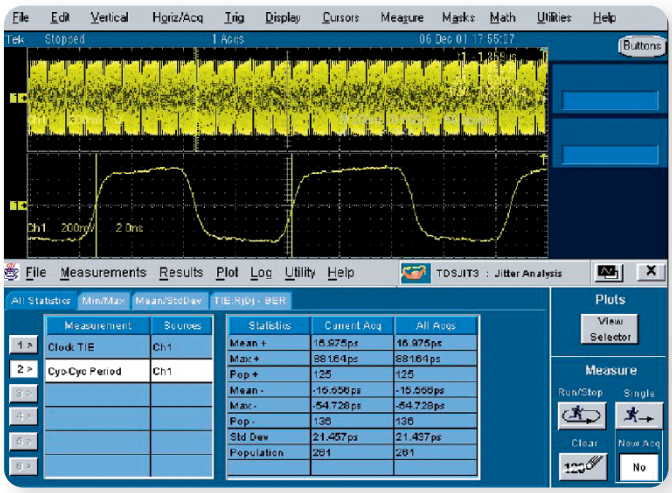
Only setup-and-hold triggering lets you deterministically trap a single violation of setup-and-hold time that would almost certainly be missed by using other trigger modes. This trigger mode makes it easy to capture specific signal quality and timing details when a synchronous data signal fails to meet setup-and-hold specifications.



However, the memory seems to store incorrect data occasionally. This is believed to be due to a timing error in the clock, which is causing data to get clocked into the memory at the wrong time—before all the data lines are "ready". What is the nature of the timing error?

What is its origin? The solution to a system-wide problem depends on theanswer to these questions.

After observing the digital errors with the logic analyzer, you suspect instability in the clock signal coming from the PLL oscillator. Though intermittent, the error does not appear to be completely random. A real-time jitter measurement is found to be the most productive solution. Connecting an oscilloscope to the clock signal, the engineer takes several measurements at a 20 GS/s sample rate and concatenates them in the on-board TDSJIT3 application.



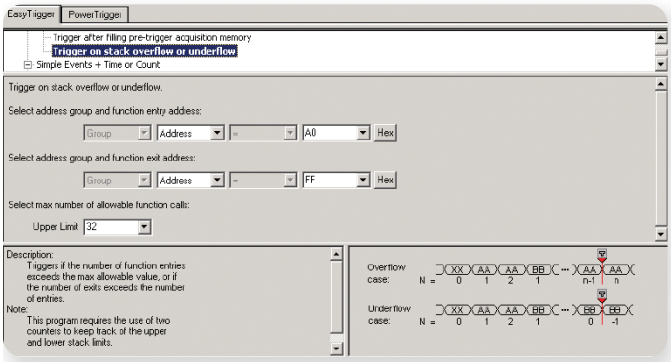
▶ **Figure 13.** *This TDSJIT3 measurement screen includes several concatenated jitter measurements taken at 20 GS/s. It reveals a PLL signal that has almost 1 ns of error within a 7.5 ns cycle.*

Comparing the sample on a cycle-to-cycle basis using the Cyc-Cyc Period measurement and the Cycle Trend function in TDSJIT3, it becomes clear that the PLL is staying within its frequency tolerance most of the time, but jumping ahead occasionally as though correcting a frequency drift. The error amounts to almost 1 ns within a 7.5 ns cycle.

A second TDSJIT3 function helps pinpoint the source of the PLL's behavior. Using the application's FFT tools, it is found that there is an unexpected energy peak at 120 kHz. A quick look at the system schematic reveals that this is the frequency of the system's switched power supply. From there, it is a simple matter to filter the offending frequency out of the PLL's supply connection.

Tip #5: Look for Overflow and Underflow Errors

For proper operation, some device events may need to occur N times, less than N times, or more than N times. How can you tell if they are occurring the correct number of times? How can you discover why they are not? Triggering with a counter is another useful capability of a logic analyzer.



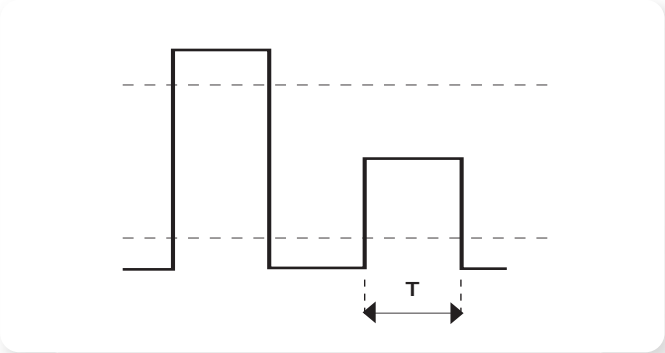
▶ **Figure 14.** *EasyTrigger stack overflow or underflow trigger definition screen.*

Another example could be a FIFO memory. If the system writes in data faster than it can be read out, the memory is being overrun. If the system locks up trying to pull data from an empty register, that would be an example of underrun.

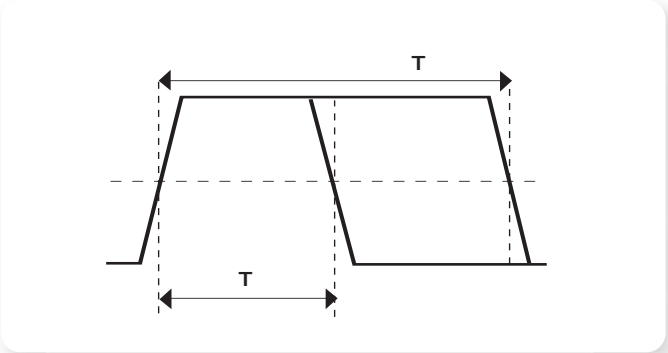
Triggering on overrun and underrun errors can be very simple to set up. Tektronix EasyTrigger includes "Trigger on stack overflow or underflow" as one of its many predefined triggers.

For example, consider the interrupt handler of a microprocessor. Interrupts are requests designed to take a processor out of its normal assignment and to address something from the periphery. The list of interrupts forms a stack in memory that waits for the processor to deal with them. If interrupts come faster than the processor can handle them, then the requests "overrun" the stack and are lost. The processor may end up in an unknown state or simply not do what it is supposed to.

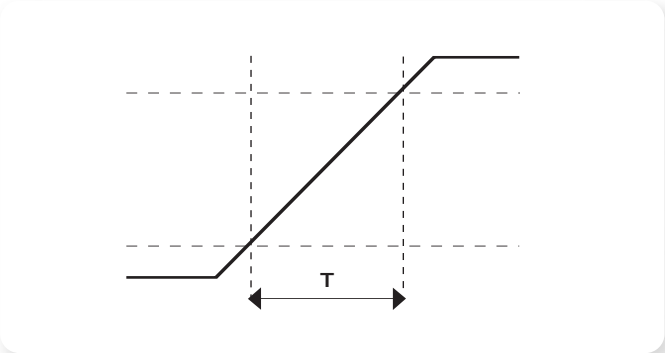
Figure 14 shows the setup screen for overflow or underflow trigger. It lets you specify what kind of event or events the logic analyzer should track. The trigger uses counters to track the number of events that are incrementing and decrementing the stack. Again, having a sufficiently long record length is important. The condition that caused the error may have occurred well before symptoms appear.



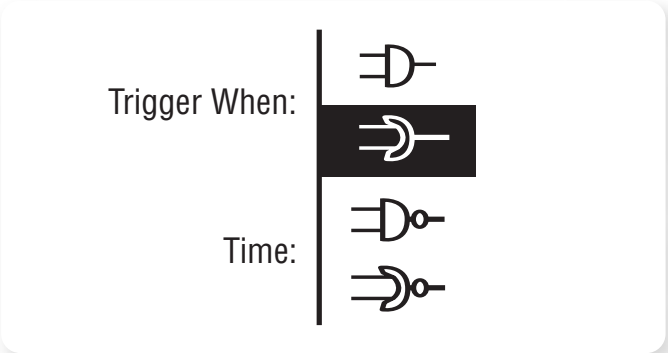
▶ **Figure 15.** *Runt Pulse Triggering*



▶ **Figure 17.** *Transition Triggering*



▶ **Figure 16.** *Pulse Width Triggering*



▶ **Figure 18.** *Logic Triggering*

Tip #6: Increase Troubleshooting Efficiency with Advanced Triggering

Earlier in this application note we mentioned some of the different oscilloscope triggering modes useful in hunting for glitches, triggering on timeout conditions and looking at setup/hold violations. Other trigger modes particularly useful for capturing timing errors and digital debug are:

Runt Pulse Triggering

Runt triggering allows you to capture and examine pulses that cross one logic threshold, but not both. The event can be time or logic qualified. Another trigger mode, similar to runt pulse triggering is Window Triggering. In Window Triggering you can trigger an event that enters or exits a window defined by two user-adjustable thresholds. The event can be time or logic qualified.

Pulse Width Triggering

Using pulse width triggering, you can monitor a signal indefinitely and trigger on the first occurence of a pulse, for example (either positive or negative going) whose duration (pulse width) is outside the allowable limits. In the TDS5000B the selectable time limits range from 1 ns to 1 s with 200 ps resolution.

Transition Triggering

With transition triggering, you trigger on pulse edge rates that are faster or slower than specified. Slope may be positive, negative or either.

Logic Triggering

Logic triggering allows you to trigger on any logic combination of available input channels – specifically useful in verifying digital logic.

Better Tools

Although logic analyzers and oscilloscopes have long been the tools of choice for digital troubleshooting, not every designer knows how capable they can be. Logic analyzers speed up debugging and verification by wading through the information stream to trigger on circuit faults and capture related events.

Oscilloscopes reveal how signal integrity problems can create false logic transformations by peering behind idealized digital timing diagrams and showing the raw, analog waveforms. Today's instruments, such as Tektronix logic analyzers, are even more powerful. They offer enhanced capabilities, including record lengths up to 64 Mb deep, MagniVu 125 ps resolution, triggers designed for troubleshooting, and the ability to work together with Tektronix oscilloscopes, such as the TDS5104B. The TDS5000B series digital phosphor oscilloscopes (DPO) deliver up to 1 GHz bandwidth, 5 GS/s real-time sample rate, 16 M record length, and a suite of advanced triggers, enabling you to capture and characterize even your most demanding signals. DPOs provide unmatched insight into signal behavior by delivering greater than 100,000 waveforms per second capture rates. In addition the TDS5000B Series are the easiest to use oscilloscopes offering the industry's first easily customizable oscilloscope user interface through the use of MyScope control windows and a comprehensive suite of context sensitive mouse right click menus.

In addition specialized jitter and timing analysis is available through:

- TDSJIT3 jitter and timing analysis software providing real time jitter measurements including random and deterministic separation (Rj/Dj) with Bit Error Ratio estimation.

With the TLA5000 Series' iView™ integrated digital-analog view, you can see time-correlated digital and analog signals on your logic analyzer display.

To fully integrate your Tektronix TLA5000 Series logic analyzer and oscilloscope into a single troubleshooting system, use the iView™ time-correlated, integrated logic analyzer and oscilloscope measurements on one display. This Tektronix-exclusive capability provides comprehensive digital and analog insight to help you quickly find and characterize faults.

Summary

Your oscilloscope and logic analyzer can help you find timing errors faster and more direct. Get a quick visual snapshot by using the oscilloscope or by using its advanced oscilloscope triggering capabilities. Or use your specialized logic analyzer triggering functionality to hunt for errors amongst hundreds of channels and then examine and characterize the faults in detail. Work from a broad picture of how your prototype functions to a close up view focused on the specific faults either with the oscilloscope or your logic analyzer. Or use them both at the same time with time synchronized cross triggering to combine the power of the simultaneous logic analyzer digital perspective and your oscilloscopes analog view. By combining these you make it much easier to characterize the source of the fault.

TDS5000B Series

The TDS5000B Series offers the industry's first easily customizable oscilloscope user interface. MyScope® control windows is a revolutionary new feature that allows you to build your own control windows with only the controls, features, and capabilities that you care about and are important in your job. For the first time, you can pull all the functionality you need from all the various parts of the oscilloscope into one control window, effectively creating your own personalized "toolbox" of oscilloscope features.

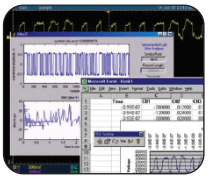
TLA5000 Series

The affordable TLA5000 Series logic analyzers make high-speed timing resolution, fast state acquisition and sophisticated triggering available to all digital designers. The TLA5000 Series are ideal for single-synchronous bus state and timing analysis. An intuitive user interface, set-up wizards, familiar Windows-based desktop and OpenChoice networking and analysis features make them easy to use and easy to network into the design environment.

iView

The iView software capability seamlessly integrates and automatically time-correlates data from the logic analyzer and oscilloscope, so you can transfer analog waveforms from the oscilloscope to the logic analyzer display with the click of a mouse. View time-correlated analog and digital signals side-by-side and pinpoint the source of elusive glitches and other problems in moments.

OpenChoice®



OpenChoice® is a collection of software libraries, utilities, samples, industry-standard protocols and interfaces offered with many Tektronix oscilloscopes and logic analyzers. From 60 MHz to 15 GHz, OpenChoice allows you to communicate with your oscilloscope or logic analyzer over a network, using numerous connectivity protocols and physical interfaces, such as GPIB, Ethernet, RS-232 and shared memory.

Contact Tektronix:

ASEAN / Australasia / Pakistan (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 Baltics +41 52 675 3777
Central Europe & Greece +41 52 675 3777
Denmark 80 88 1401
Finland +41 52 675 3777
France & North Africa +33 (0) 1 69 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) 56666-333
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 & The Baltics 7 095 775 1064
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
USA (Export Sales) 1 (503) 627-1916
For other areas contact Tektronix, Inc. at: 1 (503) 627-7111

Last Updated 3 November 2004

For Further Information

Tektronix maintains a comprehensive, constantly expanding collection of application notes, technical briefs and other resources to help engineers working on the cutting edge of technology. Please visit **www.tektronix.com**

12/04 DM/WOW 55W-18497-0

Tektronix
Enabling Innovation