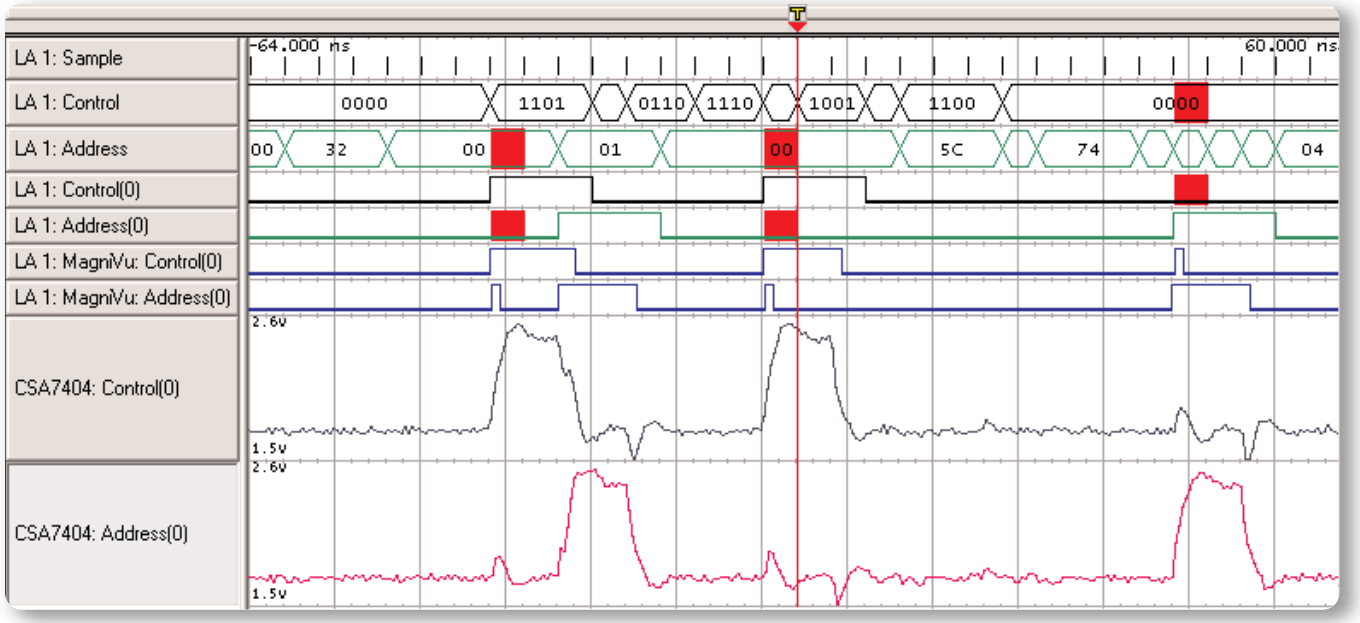
Timesaving Techniques for Digital Debugging



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.

Fast Edge Effects

Today's designs are harder to debug because there are more factors to 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. If your prototype uses high performance logic families – such as ECL (Emitter Coupled Logic), PECL (Positive Emitter Coupled Logic), or HSTL (High Speed Transistor Logic) – they 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 frequencies, 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. Before you can solve such problems, you have to find the effects, characterize them, and work back to determine their cause.

Tektronix
Enabling Innovation

Top/Down Methodology

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. Then begin focusing in on problems.

Glitch hunting is a good illustration. On the macro level, your Tektronix logic analyzer allows you to perform glitch triggering on buses that are hundreds of signals wide. The logic analyzer checks every signal at every sample point for glitches. Red bars on the bus timing diagrams show glitch locations for further analysis. On the micro level, your Tektronix oscilloscope can help characterize the problem by revealing exactly what the glitch looks like. Using the iLink™ Tool Set available on TLA700 Series logic analyzers, you can combine your logic analyzer and your oscilloscope into a single system and progressively "zoom in" on the problem. Here is a typical sequence:

1. Examine the bus

Focus on what works. Look globally for faults. Your logic analyzer's bus timing waveform will flag any glitches that occur between its 2 ns timing sample points.

2. Examine the lines

Then 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 occurs.

3. 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.

4. Examine the analog waveform

Discover what the glitch really looks like. Comparing the analog and digital perspectives will help characterize the problem.

Intermittent Challenges

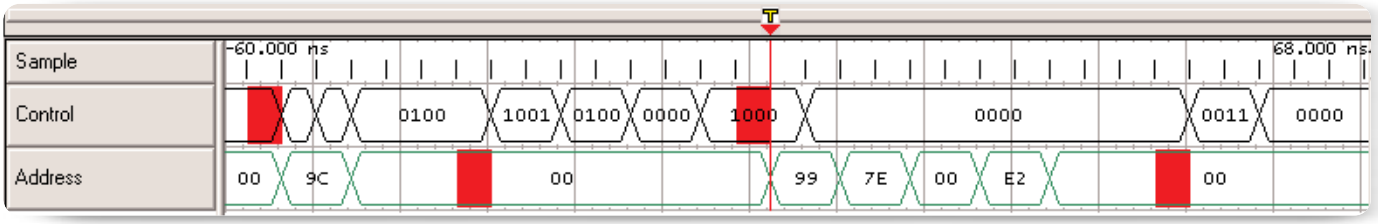
Other project-stopping culprits include timing violations, driver errors, and race conditions. All of these can create similar faults in state machine logic. Furthermore, they can show up in any circuit. Since they may, or may not occur when you are capturing signals, they are exceptionally difficult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

Tip #1: Look for Glitches

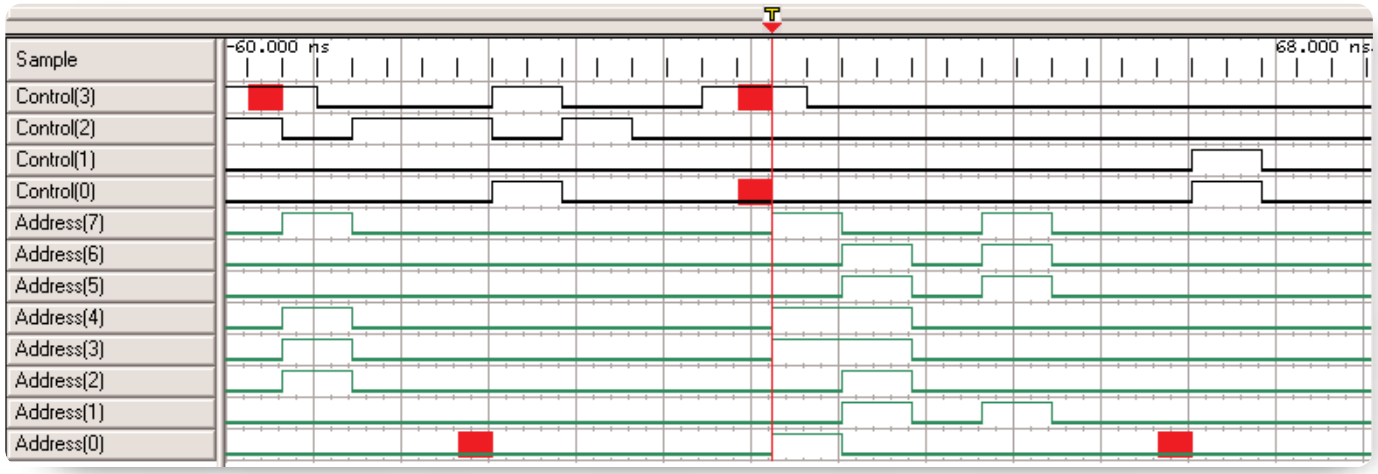
What is a glitch?

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 crosstalk, race conditions, termination errors, driver errors, and timing violations.

Since glitch effects are intermittent, they can be very hard to capture. Fortunately, logic analyzers can hold a steady stream of data in memory and look for a specified anomaly or set of events to use as a trigger. Once the logic analyzer triggers, it can share the information still held in its deep memory. These details can help you understand the cause of the glitch and how to correct it.



▶ **Figure 1.** *Sample ticks, Control bus and Address bus showing red glitch flags.*



▶ **Figure 2.** *4-bit Control bus and 8-bit Address bus expanded showing red glitch flags on individual signals.*

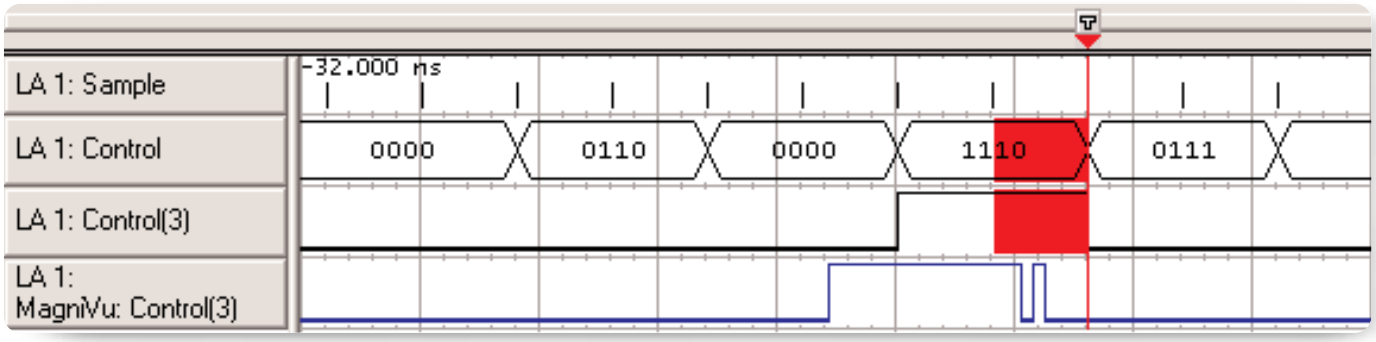
Use a Top/Down Approach

Step 1: Examine the bus

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 64 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 1, the top waveform is Sample, which shows the sample ticks that represent the logic analyzer's deep timing sampling rate of up to 2 ns (500 MHz). 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 2: Examine the lines

Then expand the logic analyzer's deep timing signal waveform, which can also be up to 64 Mb deep. In Figure 2, 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 1 are now resolved as glitch flags on signal lines Control (3) and Control (0), and as two glitches on the Address (0) signal line.



▶ **Figure 3.** *MagniVu waveform of Control (3) showing glitch.*

Step 3: Take a closer look

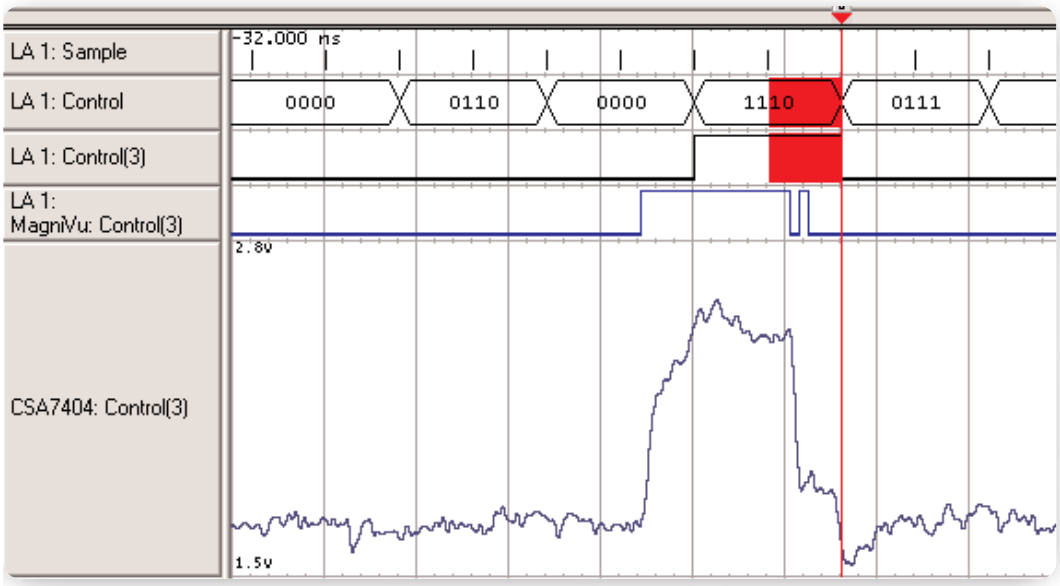
For this example, focus on the Control (3) signal line and show the MagniVu™ trace for the Control (3) signal. (We will look at the other glitches in the next example.) In addition to deep timing, Tektronix logic analyzers have high-resolution MagniVu 125 ps (8 GHz) timing capability that runs simultaneously with the deep timing capability. MagniVu 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.

Figure 3 shows that because of its higher resolution, MagniVu 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.

Step 4: Examine the analog waveform

Now use your oscilloscope and the logic analyzer's iView capability. iView™ integrated view enables you to perform time-correlated, integrated logic analyzer and oscilloscope tests and see the waveforms on the logic analyzer's display. Figure 4 shows iView's analog display of the glitch on signal line Control (3). iView allows the logic analyzer to trigger the oscilloscope at exactly the right time to capture the glitch. With iView, the logic analyzer also time-correlates the data and displays both the analog and digital waveforms on the logic analyzer's display.

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 4.** *iView oscilloscope trace showing analog representation of Control(3) signal.*

To help focus in on problems, you can also use the TLA700 Series logic analyzer's iCapture™ digital-analog probing capability. With iCapture, you can route any four of the bus signals probed by the logic analyzer to an oscilloscope for an analog domain inspection. This saves setup time and avoids loading the signals with additional probes. You will be able to see what is actually there, not artifacts of the testing process.

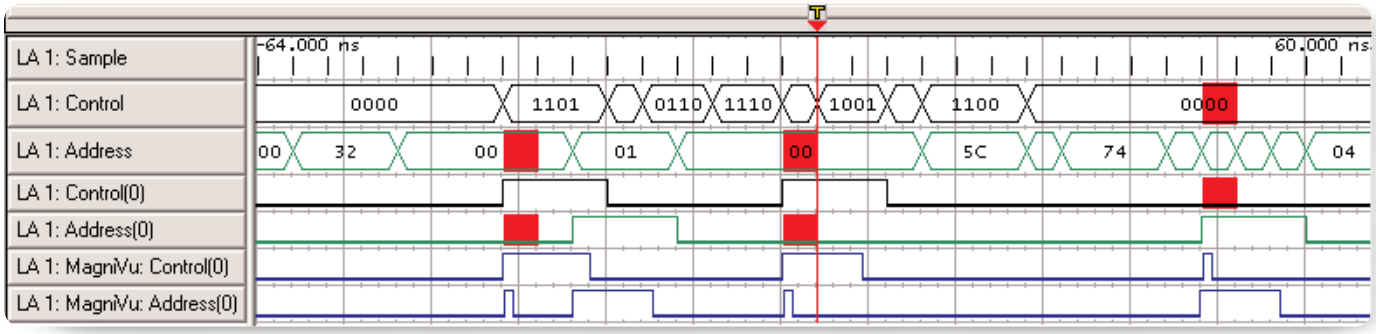
Tip #2: Look for Crosstalk

Another common problem is crosstalk. 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.

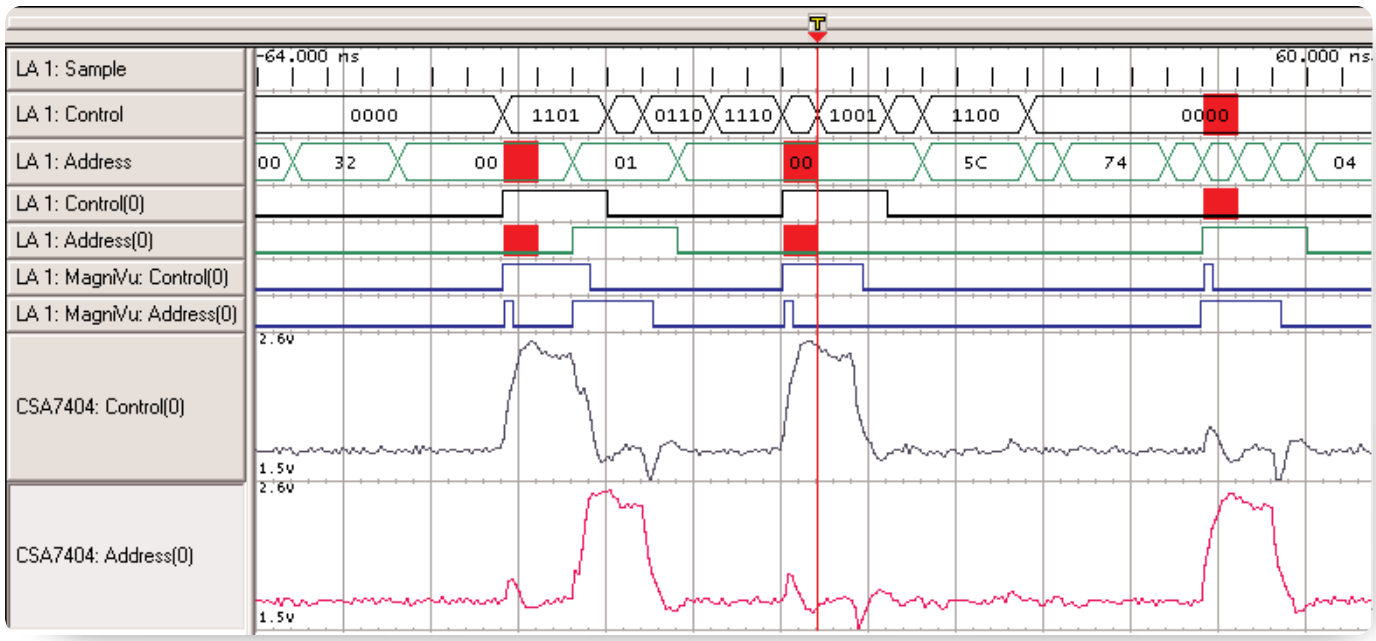
This tip continues the setup from the previous example. There you resolved the termination glitch on Control (3). That left the glitches in the Control (0) and Address (0) signal lines for this example. “Examining the Bus” in Step 1 and “Examining the Lines” in Step 2 have already taken you half way to the solution. When you resume focusing in, remember that crosstalk errors have effects that occur on more than one line. Resume your troubleshooting at Step 3.

Step 3: Take a closer look

Use MagniVu high-resolution timing to examine the two remaining flagged lines, Control (0) and Address (0). MagniVu's perspective of the signals appears in Figure 5. Since MagniVu is 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.



▶ **Figure 5.** *Control (0) and Address (0) lines with MagniVu traces showing glitches caused by crosstalk.*

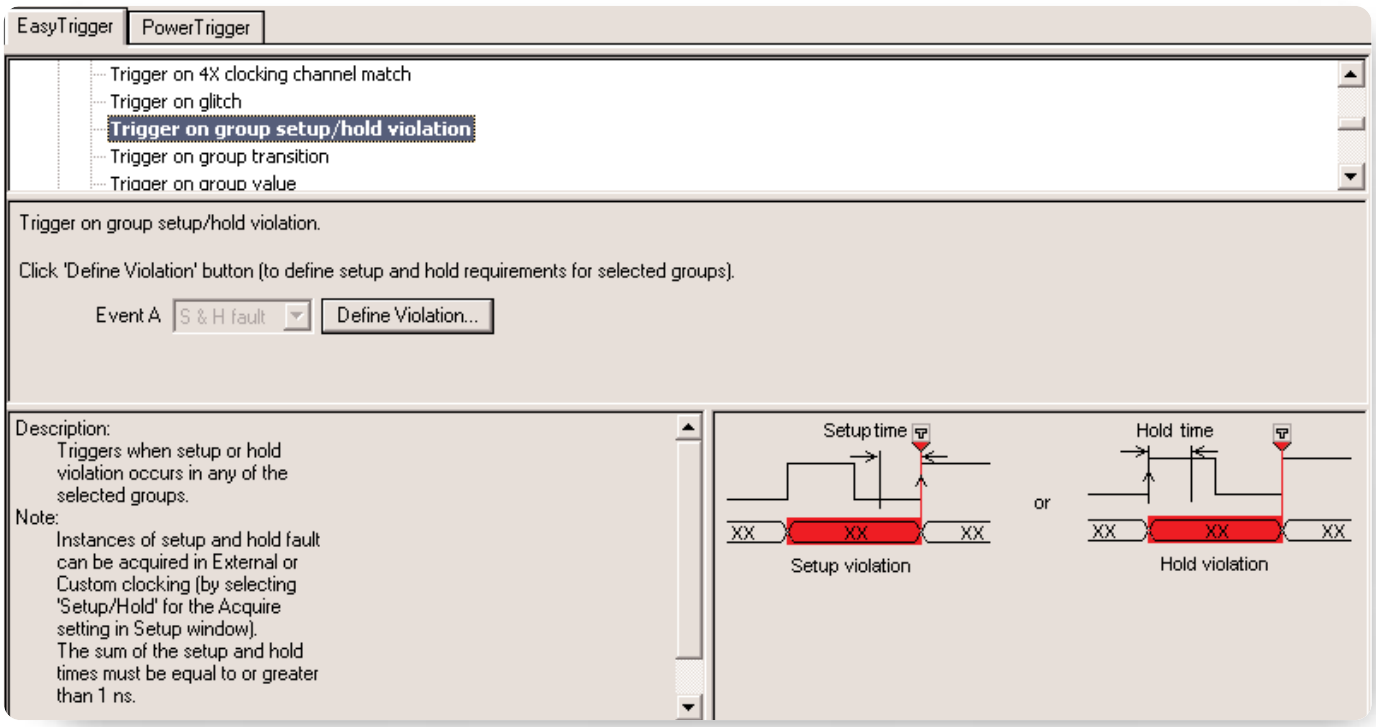


▶ **Figure 6.** *Crosstalk between Control (0) and Address (0) shown using iView.*

Step 4: Examine the analog waveform

To test the hypothesis, use iView measurements to make a time-correlated comparison of both MagniVu's digital waveform and your oscilloscope's analog waveform. Figure 6 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. Now you know to look for possible high frequency connection points between the two traces on the circuit board.

Although the buses in these two examples were quite narrow, logic analyzer glitch triggering can be used on buses with hundreds of signals. At every sample point, 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.

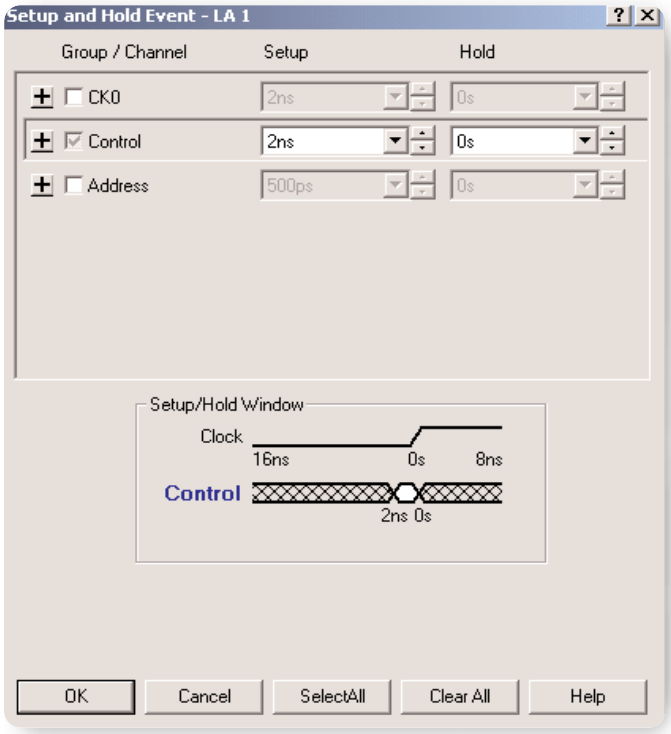


▶ **Figure 7.** *EasyTrigger setup/hold violation trigger definition screen.*

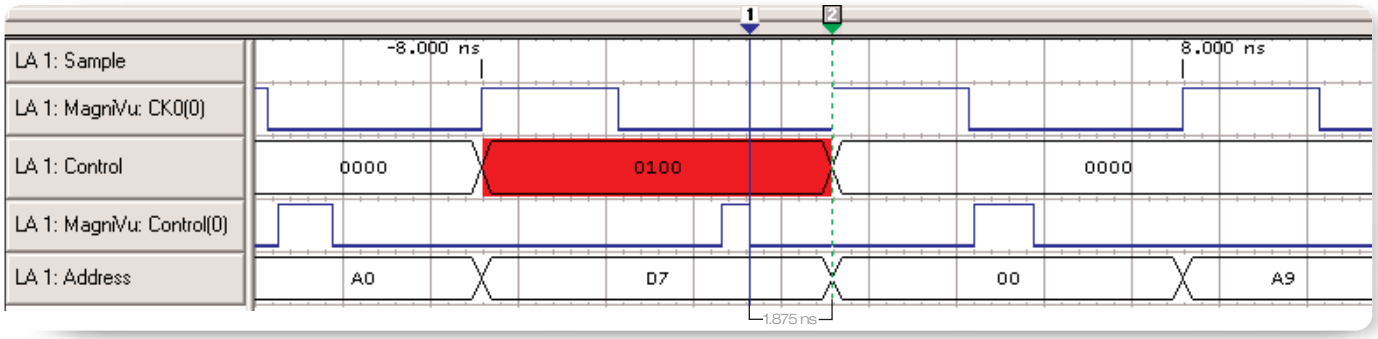
Tip #3: Look for Setup/Hold Violations

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 7 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 8.** *Setup/hold violation trigger parameter selection dialog.*



▶ **Figure 9.** *Trigger display of setup/hold violation on Control (0).*

Figure 8 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.

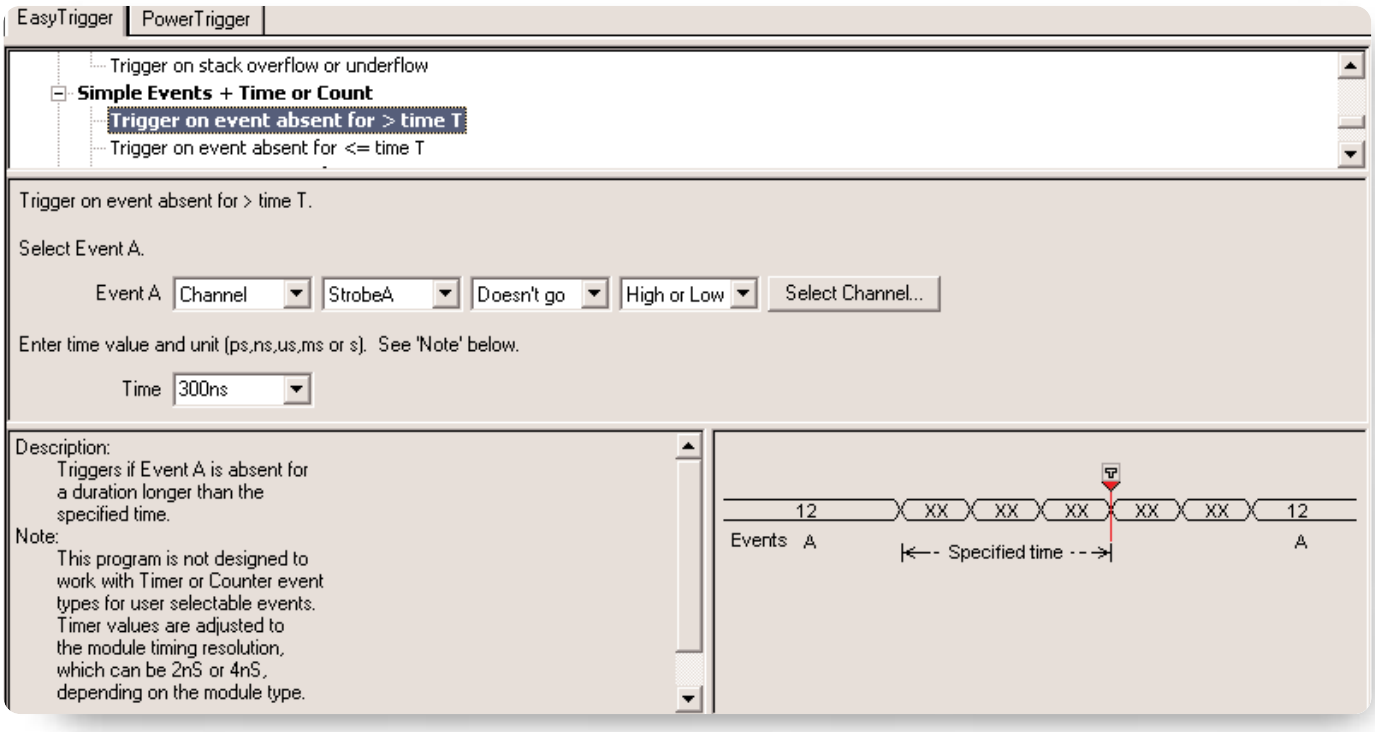
Figure 9 shows the logic analyzer triggering on a setup violation of 1.875 ns before the clock edge. Now that you have identified the problem, you can work toward a solution.

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 an easy test to setup.

Tip #4: 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. Fortunately, it is very easy to set your logic analyzer to trigger on “nothing” and to give you a deep picture of the state of the system.



▶ **Figure 10.** *EasyTrigger timeout trigger definition screen.*

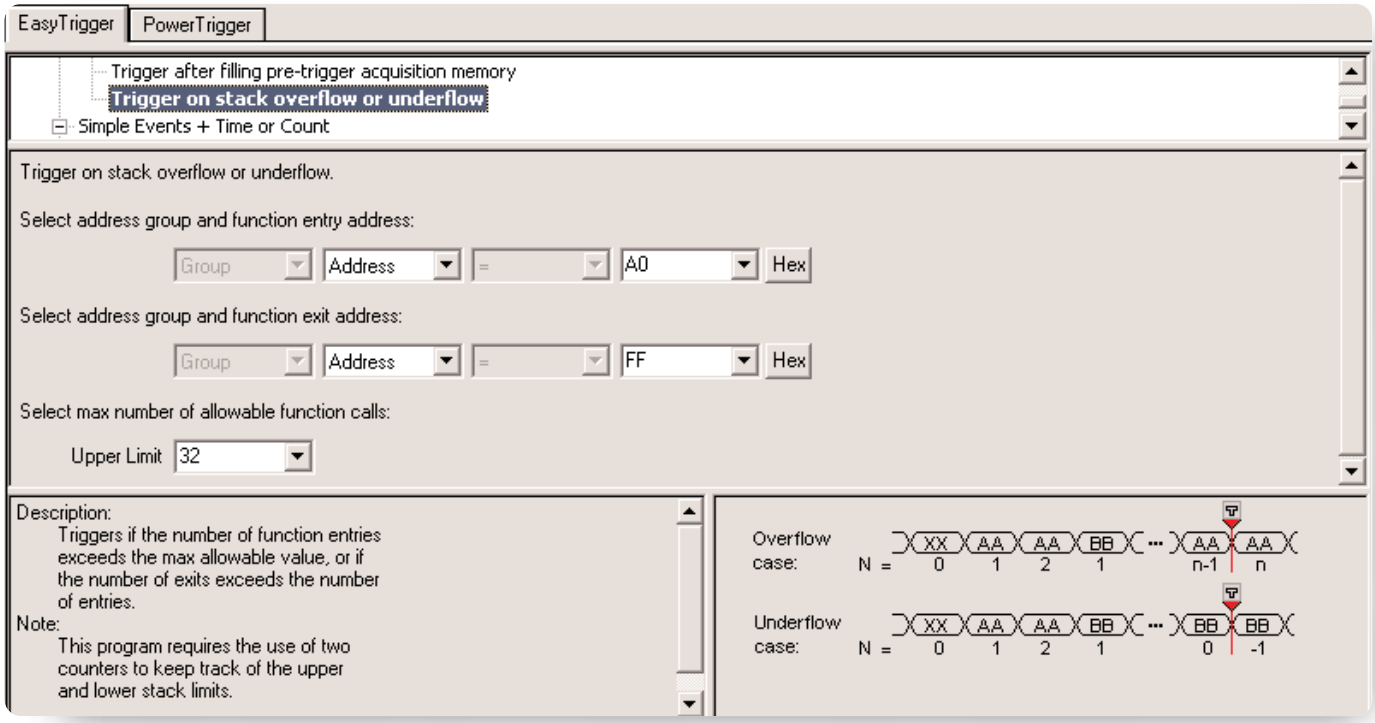
Triggering on the absence of activity is called Timeout Triggering. You can set the analyzer to watch a line or group of lines, if nothing happens – if there are no logic changes – in the period of time you specify, then the logic analyzer will trigger. You can also decide how deep your record of activity will be. Figure 10 shows the Timeout Trigger screen from the EasyTrigger menu. You can set it up in seconds.

The cause of the failure may take place well before the heartbeat actually stops. The system could stumble along 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.

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? How can you discover why they are not? Triggering with a counter is another useful capability of a logic analyzer.

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 11.** *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 if its many pre-defined triggers.

Figure 11 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.

Summary

Your logic analyzer can make your debugging process faster and more direct. Use a top/down methodology that takes advantage of your logic analyzer's special features. Work from a broad picture of how your prototype functions to a close-up view focused on the faults. Special triggers enable logic analyzers to quickly test for common problems. When you combine your logic analyzer's digital perspective with your oscilloscope's analog view, you make it much easier to characterize the source of the fault.

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 M deep, MagniVu 125 ps resolution, triggers designed for troubleshooting, and the ability to work together with Tektronix oscilloscopes. With the TLA5000 and TLA700 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 TLA700 Series logic analyzer and oscilloscope into a single troubleshooting system, use the iLink Tool Set. This Tektronix-exclusive capability provides comprehensive digital and analog insight to help you quickly find and characterize faults. The iLink Tool Set—available only with the TLA7Axx module – includes:

– iCapture™ simultaneous digital and analog acquisition through a single logic analyzer probe

– iView™ time-correlated, integrated logic analyzer and oscilloscope measurements on one display

– iVerify™ multi-channel bus analysis and validation testing using oscilloscope-generated eye diagrams

Contact Tektronix:

ASEAN / Australasia / Pakistan (65) 6356 3900
Austria +43 2236 8092 262
Belgium +32 (2) 715 89 70
Brazil & South America 55 (11) 3741-8360
Canada 1 (800) 661-5625
Central Europe & Greece +43 2236 8092 301
Denmark +45 44 850 700
Finland +358 (9) 4783 400
France & North Africa +33 (0) 1 69 86 80 34
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
Mexico, Central America & Caribbean 52 (55) 56666-333
The Netherlands +31 (0) 23 569 5555
Norway +47 22 07 07 00
People's Republic of China 86 (10) 6235 1230
Poland +48 (0) 22 521 53 40
Republic of Korea 82 (2) 528-5299
Russia, CIS & The Baltics +358 (9) 4783 400
South Africa +27 11 254 8360
Spain +34 (91) 372 6055
Sweden +46 8 477 6503/4
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
Updated 01 March 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**

06/04 FLG/WOW 57W-17683-0

Tektronix
Enabling Innovation