Application Note

The Importance of Timing Accuracy in a Logic Analyzer

Introduction

The explosive growth in the communication infrastructure worldwide is driving demand for a whole new generation of higher-performance digital designs and the logic components used to implement them. Timing problems once only associated with expensive ECL-based designs are now becoming commonplace as low-voltage CMOS and Bi-CMOS displaces older and slower technologies in microprocessors, ASIC, memory, and discrete logic devices. These newer components often have sub-nanosecond rise times with typical setup-and-hold windows of just a few nanoseconds (ns). A direct result of this phenomena is that high-resolution timing measurements, test equipment probe loading, and transmission-line effects are becoming increasingly important, even critical, to hardware engineers involved in mainstream digital design and debug. Recent market research clearly indicates that a need for more timing resolution is the single greatest reason why engineers purchase a new logic analyzer.

Although digital hardware designers must develop an understanding of many different timing problems, this application note will focus on demonstrating the need for high-resolution timing by examining three common problems. We will also look at how a modern logic analyzer designed with these types of problems in mind can provide the necessary triggering and measurement tools to find and accurately display tough-to-detect timing problems with minimal intrusion on the design-under-test (DUT).

Timing Measurement Resolution Is Not Just a Function of Clock Speed

Although high-performance processors, ASICs, and buses are becoming more common in current mainstream design, designers are faced with other more

Tektronix®

fundamental, yet crucial, timing problems that are often unrelated to system clocking. The advanced technologies mentioned earlier are inherently faster and have much tighter specifications for setup-and-hold and propagation delay, than their predecessors. This allows them to be used in a broad range of applications with a wide variance in system clock speeds, which may also lead to some very difficult-to-detect timing problems.

Table 1 lists some sample specifications for a wide range of processors, memory devices, peripherals, programmable and discrete logic devices currently used in most embedded products. This shows that, while the clock speeds at which these devices can operate ranges from about 16 MHz to 200 MHz, the setup-and-hold window of almost all of them is in the 1 ns-to-5 ns range. Propagation-delay characteristics typically fall in the same 1 ns-to-5 ns range. This means that to effectively identify timing-related problems or to verify timing margins in common digital applications, designers must be able to measure the timing of large numbers of signals with sub-nanosecond resolution.

Unfortunately, as timing measurements have become more stringent and increasingly critical to the successful debug of embedded designs, the logic analyzers being used have changed little. Although high-speed timing analyzers are available today, they are often expensive and are usually used only in very high-end design applications. More common mainstream logic analyzers typically deliver a maximum timing resolution of only about 4 ns, which is insufficient to accurately measure or even detect the kind of subtle timing problems facing today's designers. This means that crucial details commonly go undetected, jeopardizing the integrity of the design.

Traditionally, designers would turn to an oscilloscope in such a situation. Although the oscilloscope is essential for examining signal integrity and gathering parametric information on individual channels, a logic analyzer remains the prime digital debug tool when the designer needs complete coverage of the design. Logic analyzers are designed to capture information on large numbers of signals simultaneously and can trigger on a wide variety of problems, including timing, state, and code execution violations. Using a scope alone, the designer would, in most cases, have to guess where to target the instrument – a time-consuming and often fruitless strategy for other than basic or well-understood designs. A more effective approach is to let the logic analyzer detect the problem and use the oscilloscope to zero in on the offending channel if more detailed analog information is required.

Table 1: Need for High-Speed Timing

Typical Digital Devices	Clock Rate	Setup	Hold
Altera Max 7000 PLD	200 MHz	2.5 ns	0.5 ns
Lattice GAL22V10-7 PAL	133 MHz	4.5 ns	0 ns
TI TMS320C549-40 DSP	80 MHz	5 ns	0 ns
Motorola MCM69D536 SRAM	66 MHz	3 ns	1 ns
Motorola MPC 860 Comm Controller	50 MHz	4 ns	2 ns
IDT 72420L20 Synchronous FIFO	50 MHz	5 ns	1 ns
Motorola Coldfire MCF 5206 Microcontroller	33 MHz	3 ns	3 ns
Motorola MC68332 Microcontroller	16 MHz	5 ns	0 ns
AMD 29DL800B Flash Memory	16 MHz	35 ns	0 ns
TI FN74LVC573A Octal Latch	n/a	2 ns	1.5 ns

How Does Probe Loading Affect Timing Measurements?

In addition to absolute timing resolution, the designer needs to be keenly aware of the effects of loading the device-under-test (DUT) with logic analyzer probes. Probes associated with many of today's common logic analyzers were not designed to deliver the signal accurately enough to support these stringent measurement requirements. These types of probes often have unacceptable capacitive loading factors (as high as 8 to 10 pf per channel) for high-accuracy measurements, which causes unacceptable distortion of the very signals they are trying to measure. It should be noted that the quality of the signals being measured is affected by the total load presented by the circuitry of the probe, not just parasitic probe-tip capacitance, as is specified by some vendors.

It is critical to consider the "total loading" on the circuit and carefully examine the effect of the entire probe circuit. It is important as well to look for an equivalent circuit schematic from the manufacturer. If the equivalent circuit is more complex than a single capacitor to ground and a series resistance to the input of the instrument, don't assume that a single value of parasitic tip capacitance adequately describes the load on your signals. For higher-speed applications where signal quality concerns lead you to perform analog simulation of the signal path in the design phase, be sure to consider which test equipment you will want to use. Also, make sure you re-run your signal path simulation with the probe's equivalent circuit included. This can ensure you will be able to effectively use your measurement tools during debug and verification.

Excessive probe loading can distort actual signal behavior by changing the rise times, delaying signals, injecting crosstalk, and even filtering out aberrant glitches. Tektronix has addressed this critical issue by using a unique "active attenuation" probe architecture in all its logic analyzers with a typical capacitive loading of just 2 pf. For a more detailed examination of this subject, refer to the Tektronix Technical Brief "Active Attenuator Probing" available on the web under "Resources For You" at: www.tektronix.com.

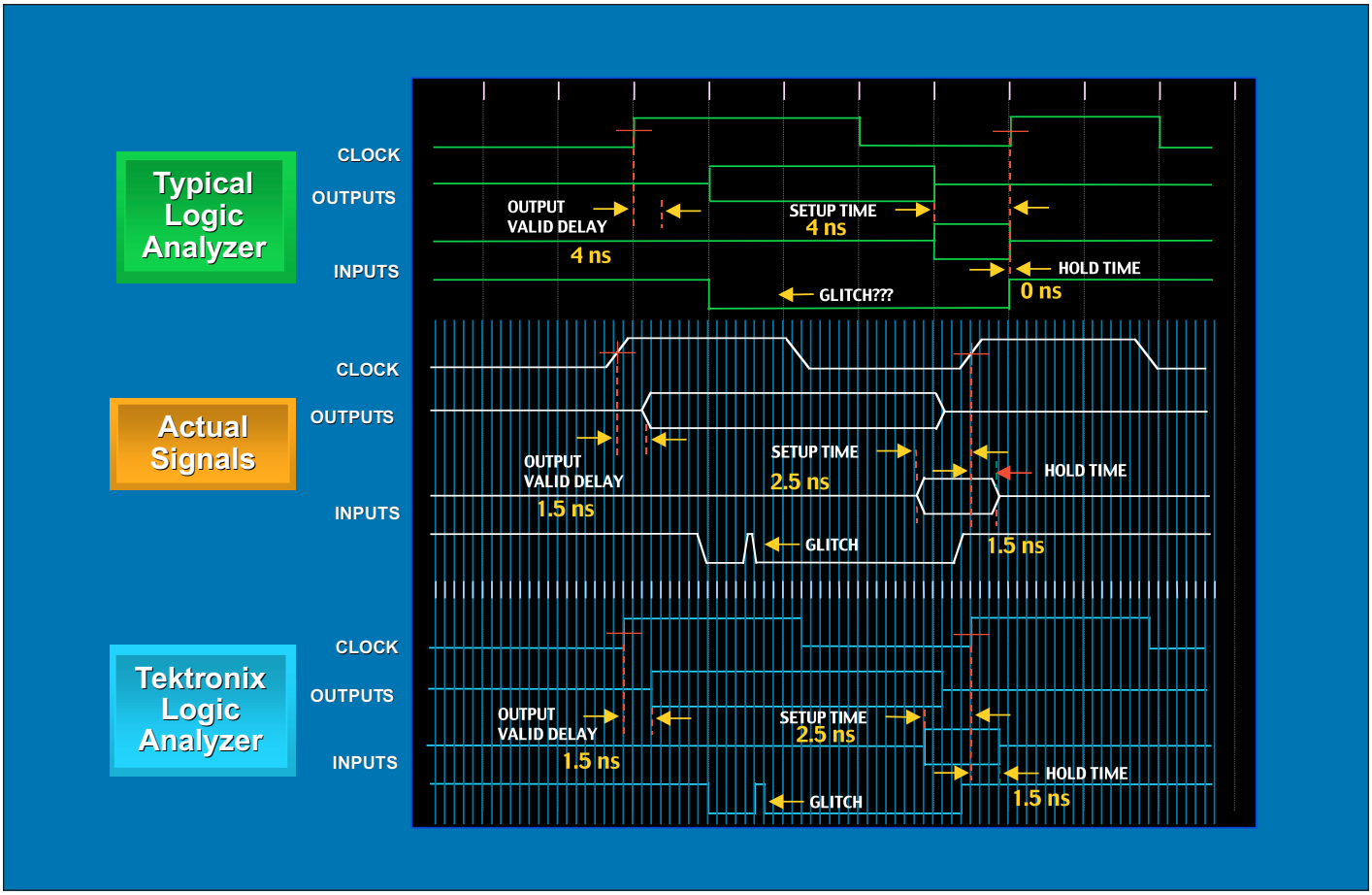


Figure 1: Truth in Timing. Comparison of actual signal detail revealed by 4 ns versus 500 ps sample resolution.

How Much Timing Resolution Is Required?

In order to reasonably evaluate critical timing parameters such as setup-and-hold, or gate propagation delay where time intervals are less than 5 ns, digital designers require a resolution of at least 1 ns or better. Tektronix' logic analyzers address this issue by providing a sampling resolution of 0.5 ns across all channels. This is more than eight times greater than what is currently available in today's prevalent logic analyzers.

Without this level of timing resolution, digital designers will often miss important timing details in their hardware and software debug efforts as illustrated in Figure 1. As shown here, the signal timing reported by a typical logic analyzer, sampling at 4 ns, differs greatly from the actual signal. With 500-ps resolution, the Tektronix logic analyzer shows timing that much more precisely represents the actual signals. Additional details such as the glitch, which most likely occurred as a result of a setup violation, can now be reliably observed and measured, and then corrected.

Having this precision in time-interval measurements for signal-to-signal edges or pulse/event widths can make the difference between success and failure when it comes to identifying the cause of elusive problems that threaten your ability to meet product development and time-to-market goals.

Sample Timing Problems

This application note examines three common, yet critical, timing problems to illustrate the importance of timing accuracy and resolution. They are setup-and-hold time violations, unexpected propagation delays, and glitches. The first two may interact to produce unwanted signals or glitches, resulting in system errors or failure.

Setup-and-Hold Violations

Setup time is the time a digital signal needs to be valid at a device input prior to that device receiving a clock edge at the clock input. Hold time is the time that the same digital signal needs to remain valid at the input of a device after receipt of the clock edge. Both parameters are typically 0 to 5 ns for most common parts (see Table 1).

Most digital designs include clocked logic. Setup-and-hold timing violations in these devices can mean serious complications. All clocked circuits read their inputs or place data on their outputs with respect to a clock signal. The engineer must ensure that the input data is valid (i.e., not changing) for a specified amount of time before the clock (setup time) and that it remains valid for some specified time after the clock (hold time).

For the flip-flop depicted in Figure 2, the setup time is measured from the time where the data inputs settle (point A) to the active edge of the clock or strobe signal (point B) used to latch those signals into the clocked device. The minimum setup time specified by the manufacturer for this example is 3 ns. The hold time is measured from the active edge of the clock or strobe signal (point B) to the time where the data changes again (point C). The minimum hold-time specification for this example is 1 ns.

The waveform in Figure 2 shows the actual setup time of a circuit using this kind of flip-flop as acquired by a Tektronix logic analyzer with 500-ps timing resolution. It indicates precisely when the data input changes prior to the active clock edge. The TLA family of logic analyzers not only provides the timing resolution necessary to reveal this problem, but includes a powerful capability to trigger on the setup-and-hold time on any number of channels. This permits important signal behavior to be captured and measured, thereby creating a more complete view of the design. This advanced triggering capability is made possible by unique acquisition technology that samples every channel with 500-ps resolution, then digitally processes this pre-sampled data to identify such violations in real time. It can further qualify the trigger definition to identify setup-and-hold violations that are associated with specific address ranges and cycle types. It is also able to flag violations only when control signals indicate that data is really being sampled by the device that is clocking the data.

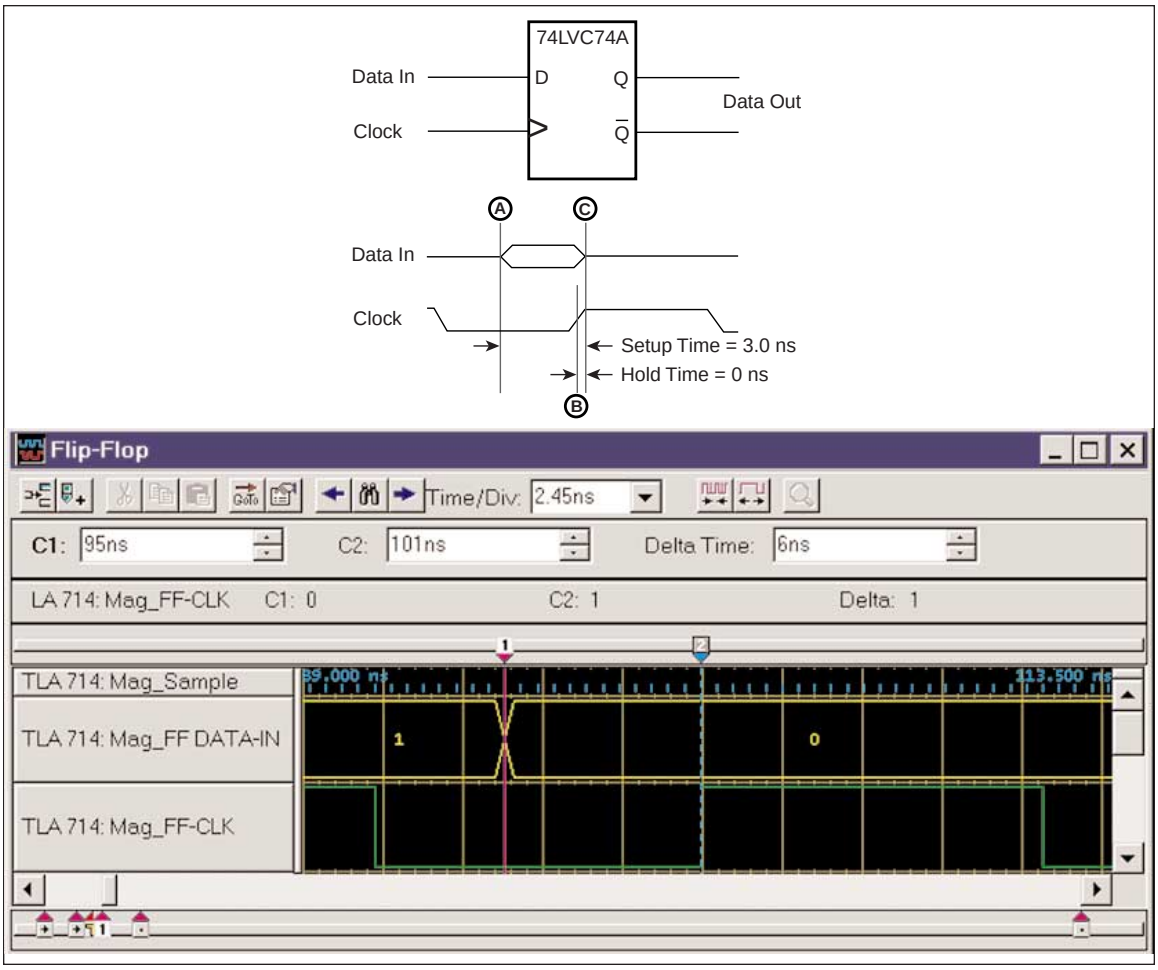


Figure 2: Schematic of a common flip-flop with timing diagram (top) with the measurements from the TLA 600 showing that the measured setup time for the flip-flop is 6.0 ns (bottom).

Propagation Delay

Propagation delay is another important element of digital signal timing. Just as clocked inputs have setup-and-hold time specifications, clocked outputs have propagation delay specifications. Actual propagation delay can vary and even fall outside the vendors' specified range when the actual application does not match the test conditions used to specify the devices. Signal loading, signal path design, variations in supply voltage, and sources of external noise can all affect the actual timing of digital outputs.

Digital signals require a finite amount of time to propagate through logic devices as well as travel from point-to-point in a system. Device-related propagation delay generally falls into one of two different categories. Gated propagation delay is the time it takes the signal to travel from the input of a circuit to its output. Clocked propagation delay is the time it takes for the active clock edge to effect a change of state at the output of the circuit. This is referred to as the CLK-to-Q propagation delay or, in some cases, data-valid delay.

In either case, the resulting time is the propagation time or propagation delay of the signal path. It is not uncommon for a circuit to have two different propagation delay specifications depending on the polarity of the signal change. Excessive propagation delay can lead to a number of problems including downstream setup violations or severely skewed signals. The flip-flop in Figure 3 has a maximum propagation delay of 5.2 ns from the CLK edge to the Q output. The actual variance is 1 ns minimum to 5.2 ns maximum, with a supply voltage of 3.3 V and a capacitive load of 30 pf. This wide variance can cause a range of problems, race conditions among them.

Inspecting this problem with a conventional logic analyzer would give the designer a misleading or erroneous picture of the signal (see Figure 1). Because of the coarse resolution capabilities of traditional tools, designers cannot get an accurate look at the real performance of their design. A typical logic analyzer will round off all timing measurements to a multiple of 4 ns. Tektronix logic analyzers deliver 500-ps resolution, allowing designers to see these finer design details, such as propagation delay, and identify the source of anomalies (see Figure 3).

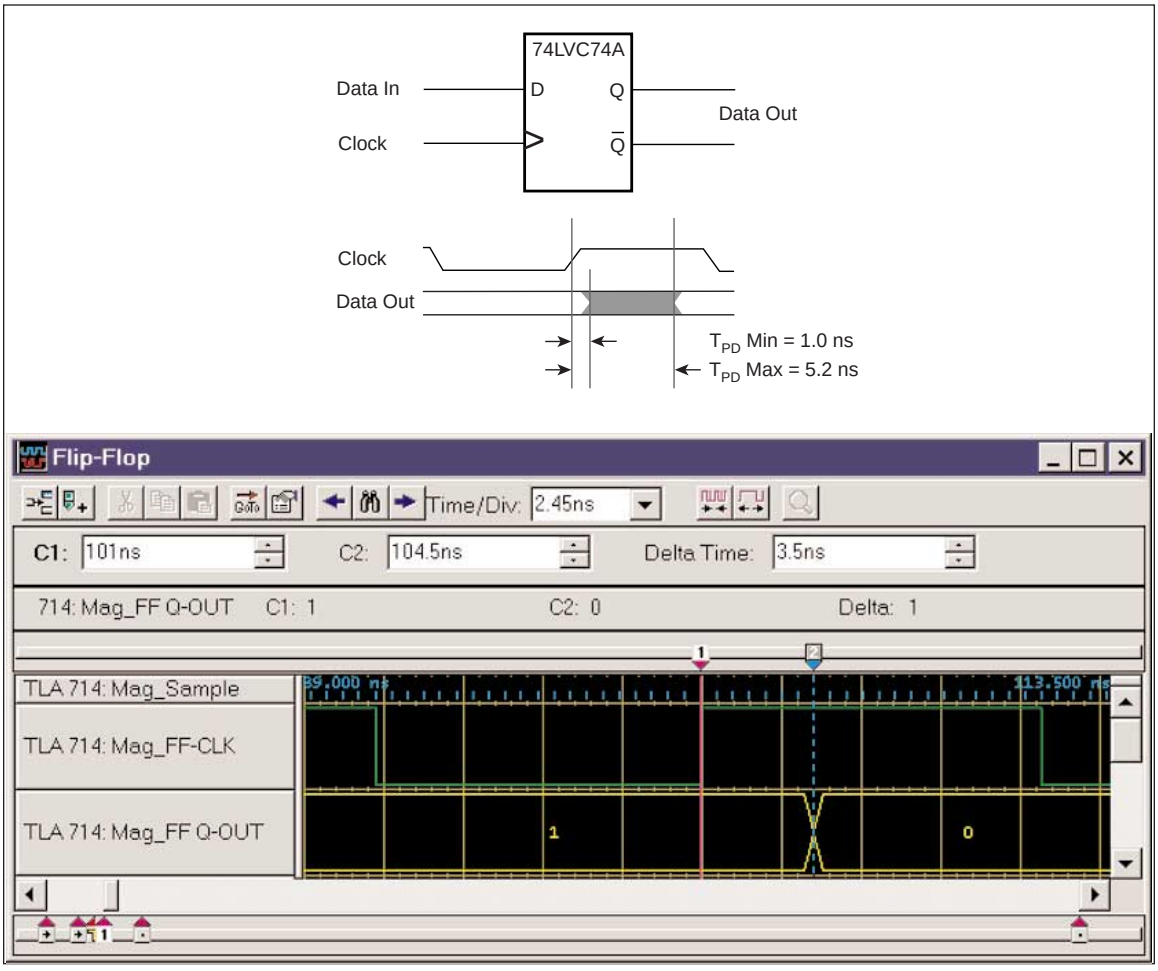


Figure 3: Flip-flop schematic and timing diagram showing clocked propagation delay (top) with a TLA 600 screen capture showing propagation delay measurement of 3.5 ns (bottom).

System Timing Example

Figure 4 depicts how excessive propagation delay in a circuit can lead to a setup-time violation and create unexpected results on the output of the circuit. This could result in system timing problems, seriously impacting system performance. In this example, there are two clocked circuits, both flip-flops. The circuits are separated by two gated circuits, both AND gates. The specifications associated with the flip-flops that will affect the system timing are setup time, hold time, and the CLK-to-Q propagation delay. The propagation delay of the AND gates will affect the system timing.

The design goal in this scenario is to have the system operate at 66 MHz. This requires that the data moving along the path highlighted in Figure 4 has a maximum of 15 ns (one cycle) to propagate from the input of the first flip-flop to the input of the second flip-flop. In order to achieve a proper setup time of 3.0 ns, the signal propagation delay should be no longer than 12 ns. Table 1 shows the manufacturer's specifications for these devices. The propagation delay (T_{PD}) for the 74LVC08A AND gate is between 1 ns and 4.1 ns. For the 74LVC74A flip-flops, the maximum T_{PD}, CLK-to-Q propagation delay, is between 1 ns and 5.2 ns. The flip-flop at the end also has a minimum setup time specification of 3.0 ns.

The propagation time of the total circuit from the edge of the clock until the output of the first flip-flop arrives at the input of the second flip-flop is the sum of the propagation delays of the first three devices, plus the delays contributed by the circuit board traces. Even if the devices are in close proximity and the trace lengths are negligible, the total delay will be

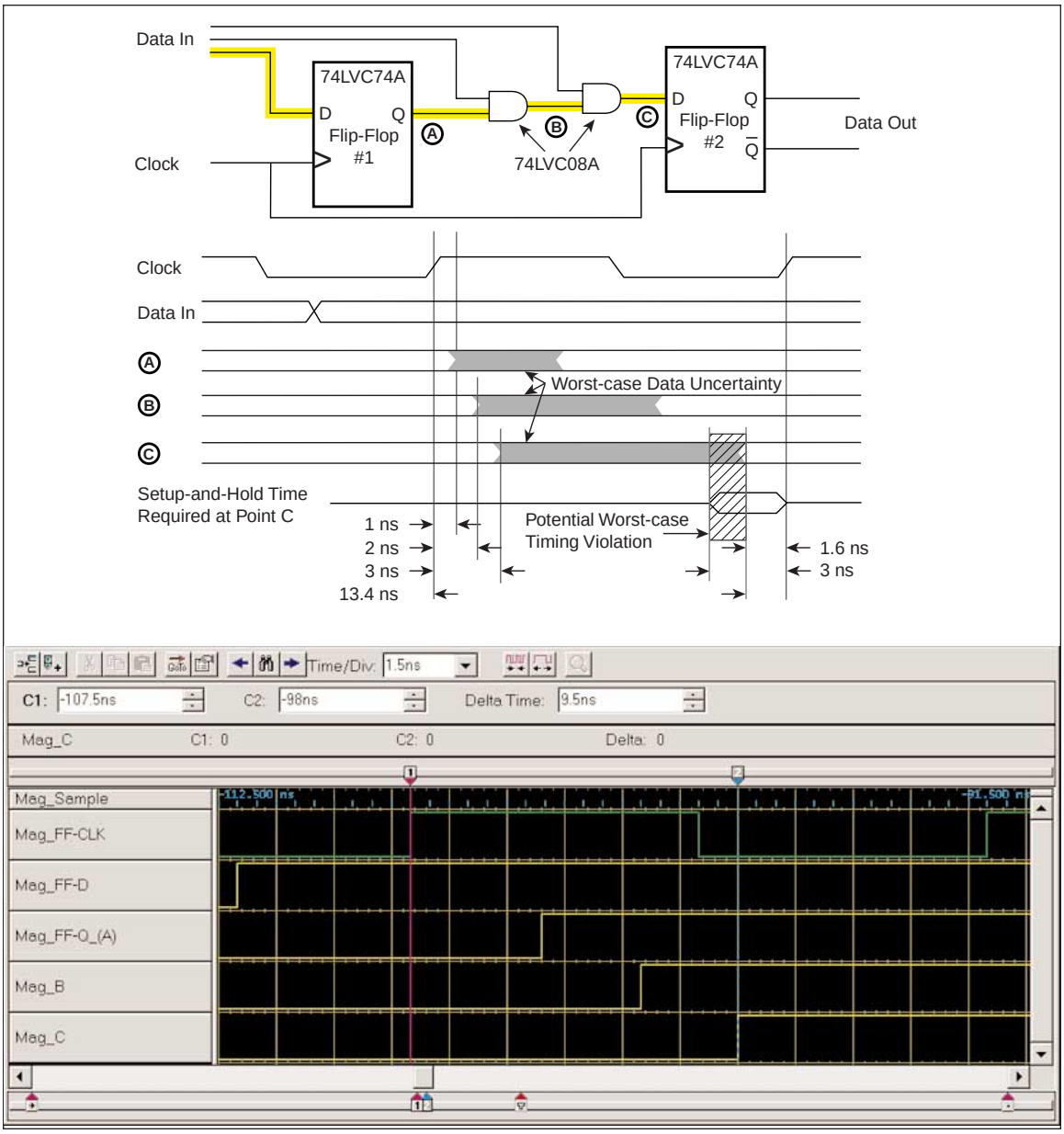


Figure 4: Two flip-flop configuration with timing diagram showing a worst-case timing violation at the input of the second flip-flop. The TLA 600 screen capture shows actual circuit timing. Notice that the signal arrives at point "C" 8.5 ns after the clock edge, providing 6.5 ns setup time to the input of the second flip-flop. The actual propagation delays observed are 3.5 ns for the flip-flop and 2.5 ns each for the AND gates.

between 3 ns (the sum of the minimum T_{PD} values) and 13.4 ns (the sum of the maximum T_{PD} values), as shown in the timing diagram in Figure 4. The worst-case maximum would leave only 1.6 ns (15-ns clock period minus 13.4-ns delay) of setup time for the next flip-flop and violate its 3 ns requirement.

This simple example illustrates how signals that pass through several devices with a broad range of T_{PD} characteristics can exhibit an extremely wide range of actual timing at the end of the line. Obviously, this can be minimized if the number of gates can be reduced. But if the gates are needed to provide the functionality, the total delay can vary greatly, requiring actual measurement to characterize behavior and determine whether the setup time is actually being violated and whether the design has sufficient margin for reliable operation.

The circuit timing measurement illustrated in Figure 4 shows that the actual delay of each gate is less than the worst-case maximum, and that the setup time observed at the input of the second flip-flop (9.5 ns) is more than adequate. The next question to answer is: Does the design offer sufficient margin to operate properly over a range of operating conditions and with component variations encountered in manufacturing over the product's life cycle?

Glitches

Another major threat to the viability of a digital design are signal anomalies such as glitches. Glitches are short pulses on digital signals that can be induced by a wide variety of logic-design problems such as metastability and race conditions, or by external factors such as inductive coupling, signal crosstalk, ground bounce, or electrostatic discharge. Glitches are generally unwanted and seemingly random anomalies that occur on digital signals.

A glitch is generally defined by a logic analyzer as any signal that changes logic state more than once between subsequent data samples. Depending on when they occur in a logic circuit, glitches may cause logic errors and other circuit or system malfunctions.

A logic analyzer that provides both glitch triggering and glitch storage makes it easy to trigger on glitches anywhere in the system and identify when they occur and on which signals. This enables the designer to quickly measure and characterize them so they can identify and correct the cause.

Some logic analyzers provide glitch triggering, but not glitch storage. Although this limited capability is still somewhat useful, the lack of storage makes the whole process both much more cumbersome and more prone to missing problems, thereby misleading the user. The big drawback of not having glitch storage is that the

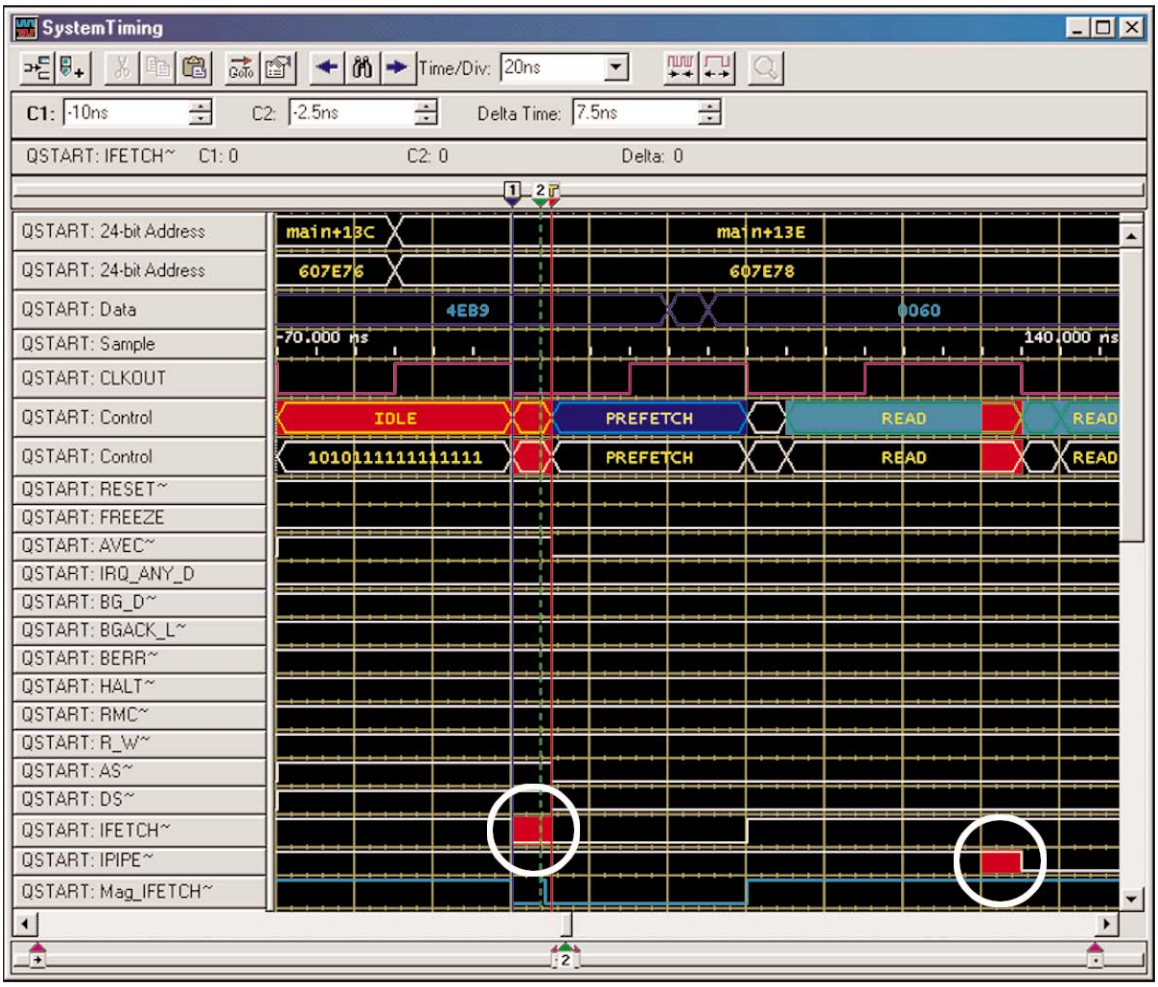


Figure 5: This TLA 600 screen capture shows data acquired from a microprocessor system containing glitches in the control group on the IFETCH and IPIPE signals.

analyzer will see a glitch and trigger on it, but has no way to show the user when or where the glitch occurred. The user is also unable to determine how many signals might have a glitch unless they repeat the acquisition many times, with the analyzer set to trigger on glitches one channel at a time.

Once the glitches are triggered on and captured, the timing resolution of the analyzer will determine how effectively it can measure the width and position of the glitches that are acquired. If the analyzer uses a separate module or even separate circuitry to sample high-speed timing, it is also quite possible that the trigger and the timing acquisition will disagree on whether the glitch was really there.

Figure 5 shows that the control group contains glitches (the shaded highlighting on the lower busform display). The individual signals below the busform show that the glitches are actually occurring on the ~IFETCH and ~IPIPE signals. The bottom trace shows the ~IFETCH signal acquired with 500-ps resolution. This glitch is not only clearly visible now, but the user can easily measure the width and position of the glitch, which turns out to be ringing on a trace that is not adequately terminated.

Logic Analyzer Timing Resolution Is the Key to Addressing Many Digital Design Needs

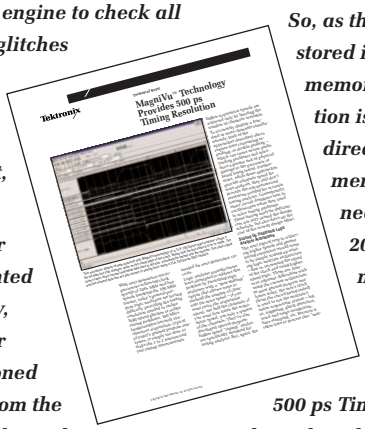
The TLA family of logic-analyzer products has changed the way digital designers approach timing-related problems. Until just a few years ago, logic analyzers were great tools for state analysis, but only provided very coarse timing resolution or required separate modules optimized for high-speed timing. Separate timing modules could usually provide good resolution, though only on a limited number of channels and with a very high cost-per-channel.

TLA logic analyzers, all with MagniVu technology, enable digital designers to capture and investigate a much broader range of problems than was previously possible. This technology, providing 500-ps timing resolution with up to 200-MS/s state acquisition simultaneously on all channels, allows designers to focus directly on correcting the faults, saving them from resorting to the time-consuming "needle-in-a-haystack" approach to finding problems.

MagniVu: A Powerful Asynchronous Oversampling Architecture

To create a logic analyzer with improved timing resolution across all channels, Tektronix leveraged a technology that has been successfully used in its high-speed portable oscilloscopes. The result is MagniVu, a fast asynchronous oversampling technology that enables the Tektronix TLA family of logic analyzers to deliver 500-ps timing resolution on all channels. MagniVu acquires all input signals, including clocks, asynchronously with a full-custom, high-speed digital sampling front end.

Selected samples of the data are extracted by high-speed digital logic and fed to the triggering and storage mechanisms based on user-defined clocking. The storage machine writes the data into a large off-chip memory array that stores synchronously acquired data at up to 200 MS/s. The oversampled data is also processed simultaneously by the triggering engine to check all channels for transitions, glitches and setup-and-hold time violations, with 500-ps resolution, to provide triggering on fast, complex activity.



At the same time, another faster memory, implemented on-chip as a custom array, directly stores 2 kbits per channel of the unconditioned stream of data straight from the 2-GHz sampler for each channel. This high-speed memory is large enough to store eight to 50 bus cycles worth of information for today's leading microprocessors.

So, as the overall state activity is stored in the larger, slower memory, complete timing information is simultaneously and directly captured in this faster memory. The result is simultaneous 2-GHz timing and 200-MHz state analysis. For more details on MagniVu, refer to the Tektronix Technical Brief "MagniVu Technology Provides 500 ps Timing Resolution" available on the web under "Resources For You" at: www.tektronix.com/LA.

For further information, contact Tektronix:

Worldwide Web: for the most up-to-date product information visit our web site at: www.tektronix.com/LA

ASEAN Countries (65) 356-3900; Australia & New Zealand 61 (2) 9888-0100; Austria, Central Eastern Europe, Greece, Turkey, Malta,& Cyprus +43 2236 8092 0; Belgium +32 (2) 715 89 70; Brazil and South America 55 (11) 3741-8360; Canada 1 (800) 661-5625; Denmark +45 (44) 850 700; Finland +358 (9) 4783 400; France & North Africa +33 1 69 86 81 81; Germany + 49 (221) 94 77 400; Hong Kong (852) 2585-6688; India (91) 80-2275577; Italy +39 (2) 25086 501; Japan (Sony/Tektronix Corporation) 81 (3) 3448-3111; Mexico, Central America, & Caribbean 52 (5) 666-6333; The Netherlands +31 23 56 95555; Norway +44 22 07 07 00; People's Republic of China 86 (10) 6235 1230; Republic of Korea 82 (2) 528-5299; South Africa (27 11)651-5222; Spain & Portugal +34 91 372 6000; Sweden +46 8 477 65 00; Switzerland +41 (41) 729 36 40; Taiwan 886 (2) 2722-9622; United Kingdom & Eire +44 (0)1344 392000; USA 1 (800) 426-2200.

ISO 9001

From other areas, contact: Tektronix, Inc. Export Sales, P.O. Box 500, M/S 50-255, Beaverton, Oregon 97077-0001, USA 1 (503) 627-1916.

0600 TD/XBS 52W-13771-0

Tektronix