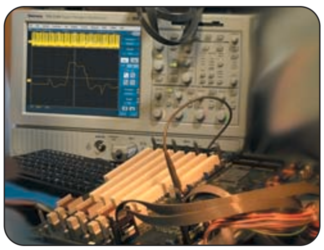
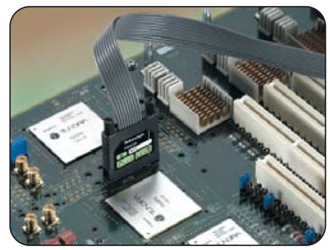
The XYZs of Logic Analyzers





COMPUTING
COMMUNICATIONS
VIDEO

Logic Analyzers

Tektronix
Enabling Innovation

The XYZs of Logic Analyzers

▶ Contents

Introduction

Like so many electronic test and measurement tools, a logic analyzer is a solution to a particular class of problems. It is a versatile tool that can help you with digital hardware debug, design verification and embedded software debug. The logic analyzer is an indispensable tool for engineers who design digital circuits.

Logic analyzers are used for digital measurements involving numerous signals or challenging trigger requirements. In this document, you will learn about logic analyzers and how they work.

In this introduction to logic analyzers, we will first look at the digital oscilloscope and the resulting evolution of the logic analyzer. Then you will be shown what comprises a basic logic analyzer. With this basic knowledge you'll then learn what capabilities of a logic analyzer are important and why they play a major part in choosing the correct tool for your particular application.

Where It All Began

Logic analyzers evolved about the same time that the earliest commercial microprocessors came to market. Engineers designing systems based on these new devices soon discovered that debugging microprocessor designs required more inputs than oscilloscopes could offer.

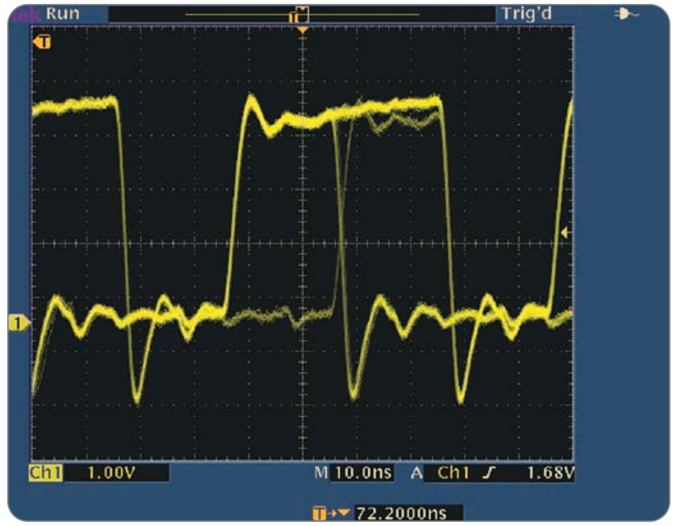
Logic analyzers, with their multiple inputs, solved this problem. These instruments have steadily increased both their acquisition rates and channel counts to keep pace with rapid advancements in digital technology. The logic analyzer is a key tool for the development of digital systems.

There are similarities and differences between oscilloscopes and logic analyzers. To better understand how the two instruments address their respective applications, it's useful to take a comparative look at their individual capabilities.

The Digital Oscilloscope

The digital oscilloscope is the fundamental tool for general-purpose signal viewing. Its high sample rate (up to 20 GS/s) and bandwidth enables it to capture many data points over a span of time, providing measurements of signal transitions (edges), transient events, and small time increments.

While the oscilloscope is certainly capable of looking at the same digital signals as a logic analyzer, most oscilloscope users are concerned with analog measurements such as rise- and fall-times, peak amplitudes, and the elapsed time between edges.



▶ ***Figure 1.*** *The oscilloscope reveals the details of signal amplitude, rise time, and other analog characteristics.*

A look at the waveform in Figure 1 illustrates the oscilloscope's strengths. The waveform, though taken from a digital circuit, reveals the analog characteristics of the signal, all of which can have an effect on the signal's ability to perform its function. Here, the oscilloscope has captured details revealing ringing, overshoot, rolloff in the rising edge, and other aberrations appearing periodically.

With the oscilloscope's built-in tools such as cursors and automated measurements, it's easy to track down the signal integrity problems that can impact your design. In addition, timing measurements such as propagation delay and setup-and-hold time are natural candidates for an oscilloscope. And of course, there are many purely analog signals – such as the output of a microphone or digital-to-analog converter – which must be viewed with an instrument that records analog details.

Oscilloscopes generally have up to four input channels. What happens when you need to measure five digital signals simultaneously – or a digital system with a 32-bit data bus and a 64-bit address bus? This points out the need for a tool with many more inputs – the logic analyzer.

When Should I Use an Oscilloscope?

If you need to measure the "analog" characteristics of a few signals at a time, the digital oscilloscope is the most effective solution. When you need to know specific signal amplitudes, power, current, or phase values, or edge measurements such as rise times, an oscilloscope is the right instrument.

Use a Digital Oscilloscope When You Need to:

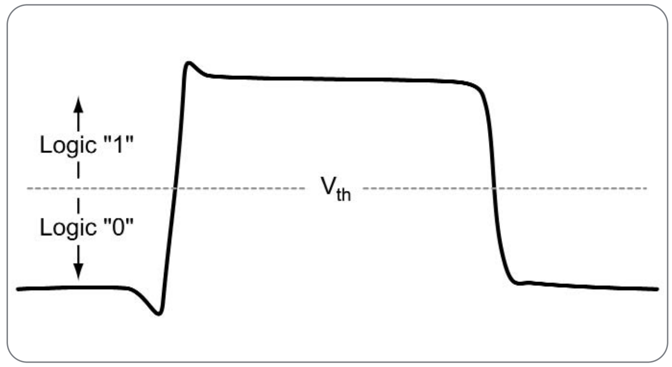
- Characterize signal integrity (such as rise time, overshoot, and ringing) during verification of analog and digital devices
- Characterize signal stability (such as jitter and jitter spectrum) on up to four signals at once
- Measure signal edges and voltages to evaluate timing margins such as setup/hold, propagation delay
- Detect transient faults such as glitches, runt pulses, metastable transitions
- Measure amplitude and timing parameters on a few signals at a time

When Should I Use a Logic Analyzer?

A logic analyzer is an excellent tool for verifying and debugging digital designs. A logic analyzer verifies that the digital circuit is working and helps you troubleshoot problems that arise. The logic analyzer captures and displays many signals at once, and analyzes their timing relationships. For debugging elusive, intermittent problems, some logic analyzers can detect glitches, as well as setup-and-hold time violations. During software/hardware integration, logic analyzers trace the execution of the embedded software and analyze the efficiency of the program's execution. Some logic analyzers correlate the source code with specific hardware activities in your design.

Use a Logic Analyzer When You Need to:

- Debug and verify digital system operation
- Trace and correlate many digital signals simultaneously
- Detect and analyze timing violations and transients on buses
- Trace embedded software execution



▶ ***Figure 2.*** *A logic analyzer determines logic values relative to a threshold voltage level.*

The Logic Analyzer

The logic analyzer has different capabilities than the oscilloscope. The most obvious difference between the two instruments is the number of channels (inputs). Typical digital oscilloscopes have up to four signal inputs. Logic analyzers have between 34 and 136 channels. Each channel inputs one digital signal. Some complex system designs require thousands of input channels. Appropriately-scaled logic analyzers are available for those tasks as well.

A logic analyzer measures and analyzes signals differently than an oscilloscope. The logic analyzer doesn't measure analog details. Instead, it detects logic threshold levels. When you connect a logic analyzer to a digital circuit, you're only concerned with the logic state of the signal. A logic analyzer looks for just two logic levels, as shown in Figure 2.

When the input is above the threshold voltage (V) the level is said to be "high" or "1;" conversely, the level below V_{th} is a "low" or "0." When a logic analyzer samples input, it stores a "1" or a "0" depending on the level of the signal relative to the voltage threshold.

A logic analyzer's waveform timing display is similar to that of a timing diagram found in a data sheet or produced by a simulator. All of the signals are time-correlated, so that setup-and-hold time, pulse width, extraneous or missing data can be viewed. In addition to their high channel count, logic analyzers offer important features that support digital design verification and debugging. Among these are:

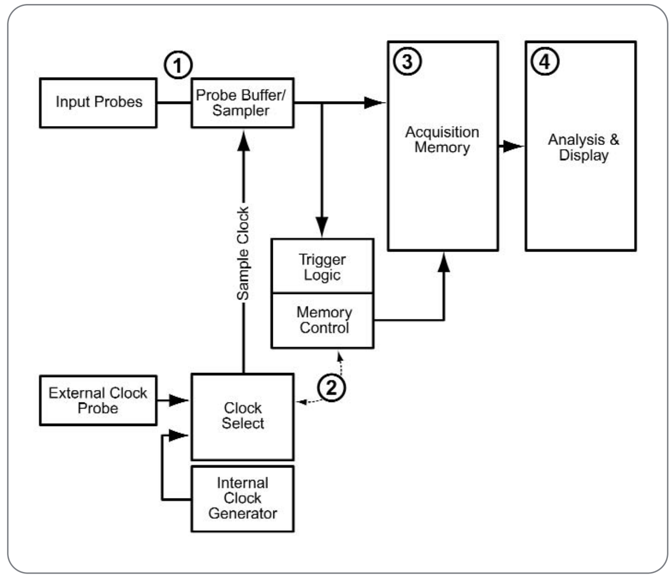
- Sophisticated triggering that lets you specify the conditions under which the logic analyzer acquires data
- High-density probes and adapters that simplify connection to the system under test (SUT)
- Analysis capabilities that translate captured data into processor instructions and correlate it to source code

Logic Analyzer Architecture and Operation

The logic analyzer connects to, acquires, and analyzes digital signals. These are the four steps to using a logic analyzer:

1. Probe: Connect to the System Under Test – SUT
2. Setup (clock mode and triggering)
3. Acquire
4. Analyze and display

Figure 3 is a simple logic analyzer block diagram. Each block symbolizes several hardware and/or software elements. The block numbers correspond to the four steps listed above.

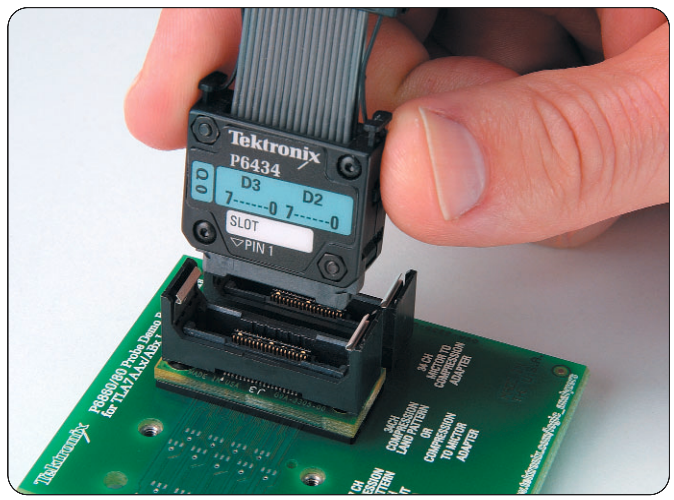


▶ ***Figure 3.*** *Simplified logic analyzer block diagram.*

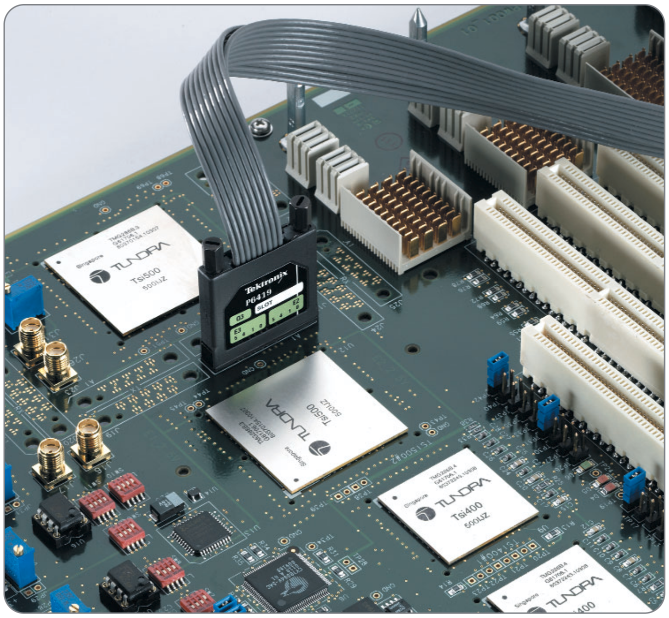
Connect to the System Under Test

Probe

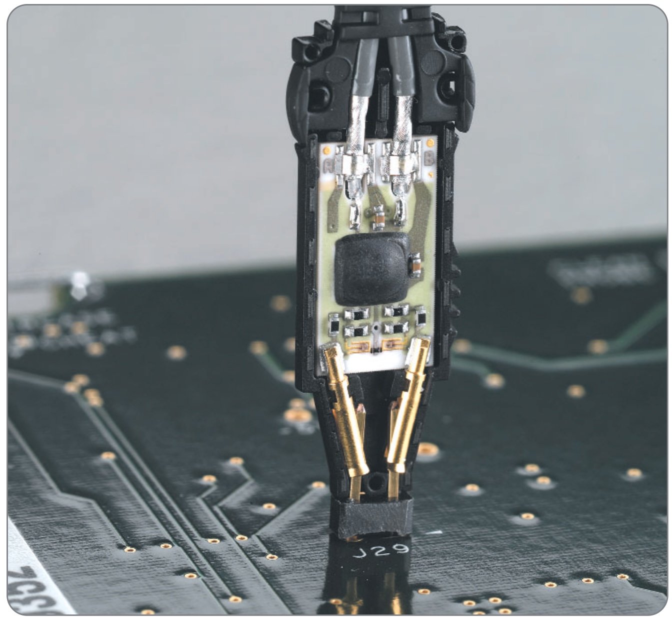
The large number of signals that can be captured at one time by the logic analyzer is what sets it apart from the oscilloscope. The acquisition probes connect to the SUT. The probe's internal comparator is where the input voltage is compared against the threshold voltage (V_{th}), and where the decision about the signal's logic state (1 or 0) is made. The threshold value is set by the user, ranging from TTL levels to, CMOS, ECL, and user-definable.



▶ ***Figure 4.*** *High-density, multi-channel logic analyzer probe.*



▶ ***Figure 5.*** *Compression probe.*



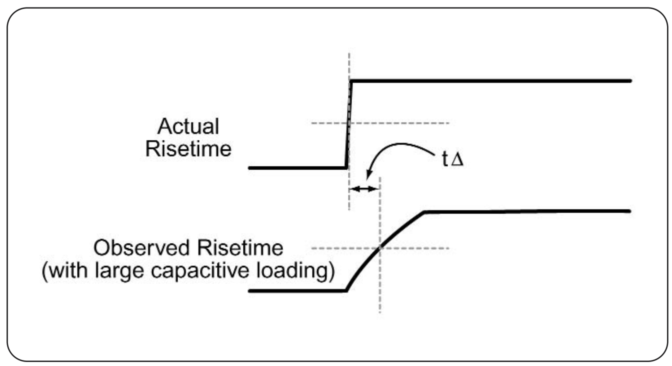
▶ ***Figure 6.*** *General purpose probe with square pin adapter.*

Logic analyzer probes come in many physical forms:

- "Clip-on" probes intended for point-by-point troubleshooting
- High-density, multi-channel probes that require dedicated connectors on the circuit board as shown in Figure 6. The probes are capable of acquiring high-quality signals, and have a minimal impact on the SUT
- High-density compression probes that use a connector-less probe attach as shown in Figure 5. This type of probe is recommended for those applications that require higher signal density or a connector-less probe attach mechanism for quick and reliable connections to your system under test

The impedance of the logic analyzer's probes (capacitance, resistance, and inductance) becomes part of the overall load on the circuit being tested. All probes exhibit loading characteristics. The logic analyzer probe should introduce minimal loading on the SUT, and provide an accurate signal to the logic analyzer.

Probe capacitance tends to "roll off" the edges of signal transitions, as shown in Figure 7. This roll off slows down the edge transition by an amount of time represented as "tΔ" in Figure 7. Why is this important? Because a slower edge crosses the logic threshold of the circuit later, introducing timing errors in the SUT. This is a problem that becomes more severe as clock rates increase. In high-speed systems, excessive probe capacitance can potentially prevent the SUT from working! It is always critical to choose a probe with the lowest possible total capacitance.



▶ ***Figure 7.*** *The impedance of the logic analyzer's probe can affect signal rise times and measured timing relationships.*

It's also important to note that probe clips and lead sets increase capacitive loading on the circuits that they are connected to. Use a properly compensated adapter whenever possible.

Set Up

Set Up Clock Modes

Clock Mode Selection

Logic analyzers are designed to capture data from multi-pin devices and buses. The term “capture rate” refers to how often the inputs are sampled. It is the same function as the time base in an oscilloscope. Note that the terms “sample,” “acquire,” and “capture” are often used interchangeably when describing logic analyzer operations.

There are two types of data acquisition, or clock modes:

Timing acquisition captures signal timing information. In this mode, a clock internal to the logic analyzer is used to sample data. The faster that data is sampled, the higher will be the resolution of the measurement. There is no fixed timing relationship between the target device and the data acquired by the logic analyzer. This acquisition mode is primarily used when the timing relationship between SUT signals is of primary importance.

State acquisition is used to acquire the “state” of the SUT. A signal from the SUT defines the sample point (when and how often data will be acquired). The signal used to clock the acquisition may be the system clock, a control signal on the bus, or a signal that causes the SUT to change states. Data is sampled on the active edge and it represents the condition of the SUT when the logic signals are stable. The logic analyzer samples when, and only when, the chosen signals are valid. What transpires between clock events is not of interest here.

What determines which type of acquisition is used? The way you want to look at your data. If you want to capture a long, contiguous record of timing details, then timing acquisition, the internal (or asynchronous) clock, is right for the job.

Alternatively, you may want to acquire data exactly as the SUT sees it. In this case, you would choose state (synchronous) acquisition. With state acquisition, each successive state of the SUT is displayed sequentially in a Listing window. The external clock signal used for state acquisition may be any relevant signal.

Clock Mode Setup Tips

There are some general guidelines to follow in setting up a logic analyzer to acquire data:

1. Timing (asynchronous) acquisition: The sample clock rate plays an important role in determining the resolution of the acquisition. The timing accuracy of any measurement will always be one sample interval plus other errors specified by the manufacture. As an example, when the sample clock rate is 2 ns, a new data sample is stored into the acquisition memory every 2 ns. Data that changes after that sample clock is not captured until the next sample clock. Because the exact time when the data changed during this 2 ns period cannot be known, the net resolution is 2 ns.
2. State acquisition: When acquiring state information, the logic analyzer, like any synchronous device, must have stable data present at the inputs prior to and after the sample clock to assure that the correct data is captured.

Set Up Triggering

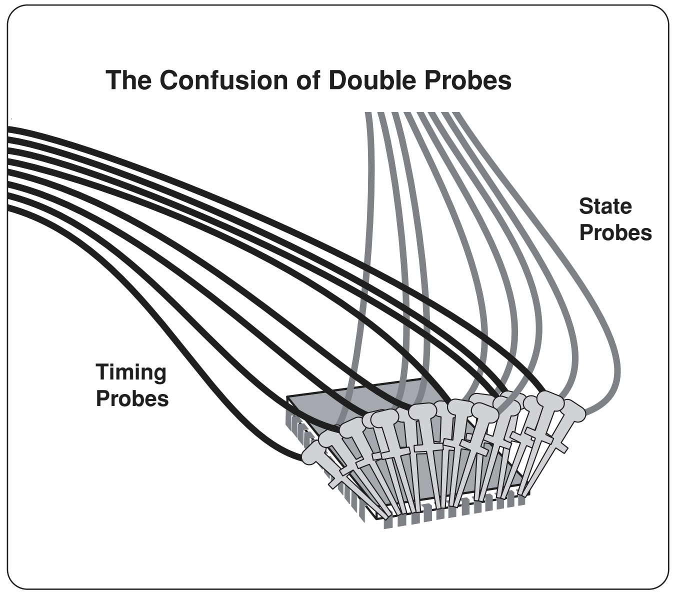
Triggering is another capability that differentiates the logic analyzer from an oscilloscope. Oscilloscopes have triggers, but they have relatively limited ability to respond to binary conditions. In contrast, a variety of logical (Boolean) conditions can be evaluated to determine when the logic analyzer triggers. The purpose of the trigger is to select which data is captured by the logic analyzer. The logic analyzer can track SUT logic states and trigger when a user-defined event occurs in the SUT.

When discussing logic analyzers, it's important to understand the term "event." It has several meanings. It may be a simple transition, intentional or otherwise, on a single signal line. If you are looking for a glitch, then that is the "event" of interest. An event may be the moment when a particular signal such as Increment or Enable becomes valid. Or an event may be the defined logical condition that results from a combination of signal transitions across a whole bus. Note that in all instances, though, the event is something that appears when signals change from one cycle to the next.

Many conditions can be used to trigger a logic analyzer. For example, the logic analyzer can recognize a specific binary value on a bus or counter output. Other triggering choices include:

- Words: specific logic patterns defined in binary, hexadecimal, etc.
- Ranges: events that occur between a low and high value
- Counter: the user-programmed number of events tracked by a counter
- Signal: an external signal such as a system reset
- Glitches: pulses that occur between acquisitions
- Timer: the elapsed time between two events or the duration of a single event, tracked by a timer
- Analog: use an oscilloscope to trigger on an analog characteristic and to cross-trigger the logic analyzer

With all these trigger conditions available, it is possible to track down system errors using a broad search for state failures, then refining the search with increasingly explicit triggering conditions.



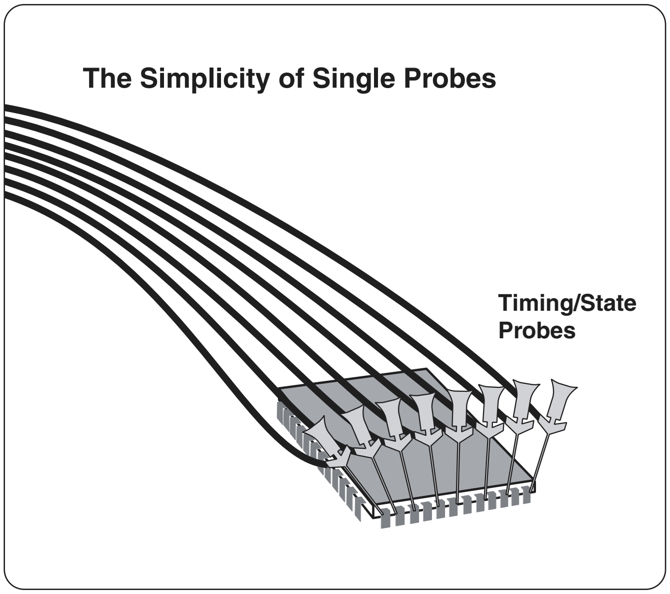
▶ ***Figure 8.*** *Double-probing requires two probes on each test point, decreasing the quality of the measurement.*

Acquisition

Simultaneous State and Timing

During hardware and software debug (system integration), it's helpful to have correlated state and timing information. A problem may initially be detected as an invalid state on the bus. This may be caused by a problem such as a setup and hold timing violation. If the logic analyzer cannot capture both timing and state data simultaneously, isolating the problem becomes difficult and time-consuming.

Some logic analyzers require connecting a separate timing probe to acquire the timing information and use separate acquisition hardware. These instruments require you to connect two types of probes to the SUT at once, as shown in Figure 8. One probe connects the SUT to a Timing module, while a second probe connects the same test points to a State module. This is known as "double-probing." It's an arrangement that can compromise the impedance environment of your signals. Using two probes at once will load down the signal, degrading the SUT's rise and fall times, amplitude, and noise performance. Note that Figure 8 is a simplified illustration showing only a few representative connections. In an actual measurement, there might be four, eight, or more multi-conductor cables attached.



▶ ***Figure 9.*** *Simultaneous probing provides state and timing acquisition through the same probe, for a simpler, cleaner measurement environment.*

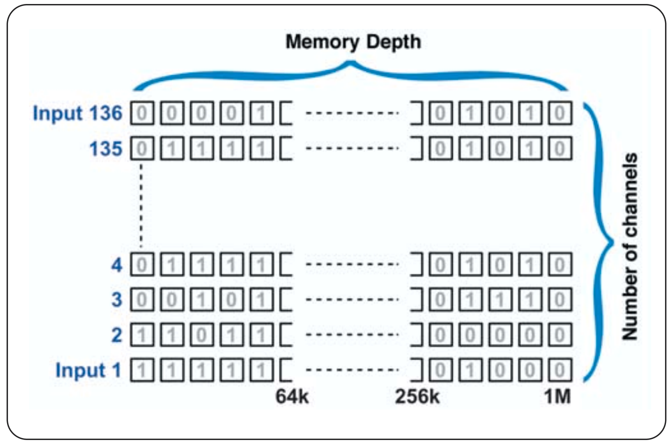
It is best to acquire timing and state data simultaneously, through the same probe at the same time, as shown in Figure 9. One connection, one setup, and one acquisition provide both timing and state data. This simplifies the mechanical connection of the probes and reduces problems.

With simultaneous timing and state acquisition, the logic analyzer captures all the information needed to support both timing and state analysis. There is no second step, and therefore less chance of errors and mechanical damage that can occur with double probing. The single probe's effect on the circuit is lower, ensuring more accurate measurements and less impact on the circuit's operation.

The higher the timing resolution, the more details you can see and trigger on in your design, increasing your chance of finding problems.

Real-time Acquisition Memory

The logic analyzer's probing, triggering, and clocking systems exist to deliver data to the real-time acquisition memory. This memory is the heart of the instrument – the destination for all of the sampled data from the SUT, and the source for all of the instrument's analysis and display.



▶ ***Figure 10.*** *The logic analyzer stores acquisition data in a deep memory with one full-depth channel supporting each digital input.*

Logic analyzers have memory capable of storing data at the instrument's sample rate. This memory can be envisioned as a matrix having channel width and memory depth, as shown in Figure 10.

The instrument accumulates a record of all signal activity until a trigger event or the user tells it to stop. The result is an acquisition – essentially a multi-channel waveform display that lets you view the interaction of all the signals you've acquired, with a very high degree of timing precision.

Channel count and memory depth are key factors in choosing a logic analyzer. Following are some tips to help you determine your channel count and memory depth:

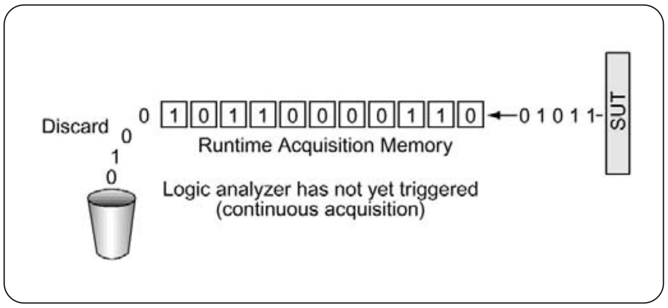
- **How many signals do you need to capture and analyze?**

  Your logic analyzer's channel count maps directly to the number of signals you want to capture. Digital system buses come in various widths, and there is often a need to probe other signals (clocks, enables, etc.) at the same time the full bus is being monitored. Be sure to consider all the buses and signals you will need to acquire simultaneously.

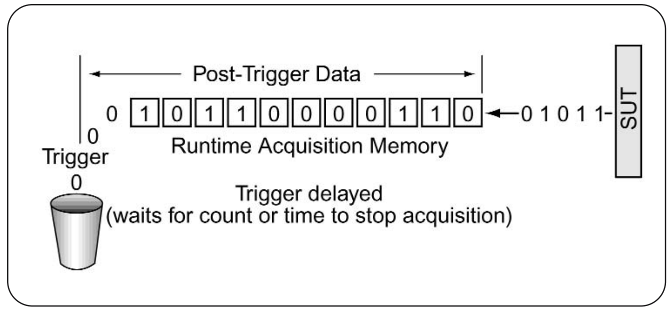
- **How much "time" do you need to acquire?**

  This determines the logic analyzer's memory depth requirement, and is especially important for asynchronous acquisition. For a given memory capacity, the total acquisition time decreases as the sample rate increases. For example, the data stored in a 1M memory spans 1 second of time when the sample rate is 1 ms. The same 1M memory spans only 10 ms of time for an acquisition clock period of 10 ns.

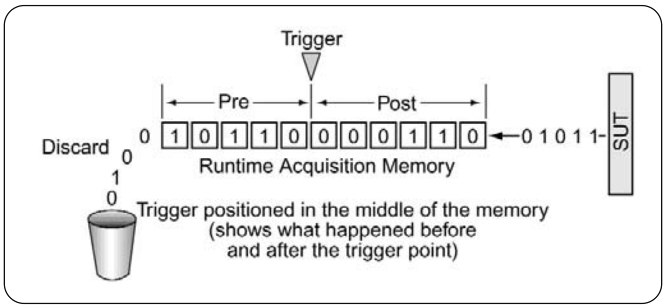
  Acquiring more samples (time) increases your chance of capturing both an error, and the fault that caused the error (see explanation which follows).



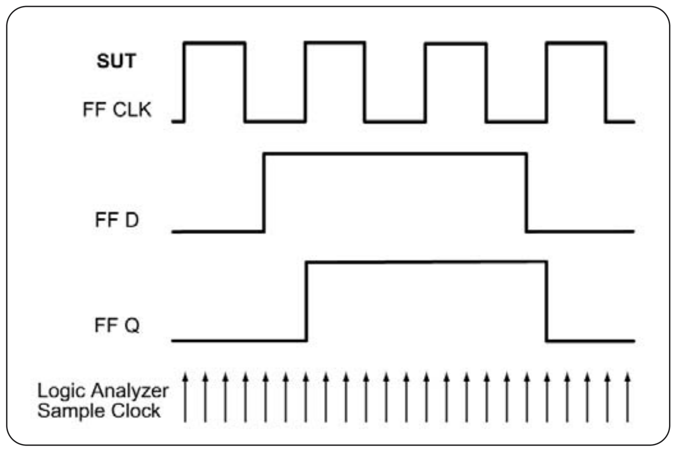
▶ ***Figure 11.*** *The logic analyzer captures and discards data on a first-in, first-out basis until a trigger event occurs.*



▶ ***Figure 13.*** *Capturing data that occurred a specific time or number of cycles later than the trigger.*



▶ ***Figure 12.*** *Capturing data around the trigger: Data to the left of the trigger point is "pre-trigger" data while data to the right is "post-trigger" data. The trigger can be positioned from 0% to 100% of memory.*



▶ ***Figure 14.*** *Logic analyzer waveform display (simplified).*

Logic analyzers continuously sample data, filling up the real-time acquisition memory, and discarding the overflow on a first-in, first-out basis as shown in Figure 11. Thus there is a constant flow of real-time data through the memory. When the trigger event occurs, the "halt" process begins, preserving the data in the memory.

The placement of the trigger in the memory is flexible, allowing you to capture and examine events that occurred before, after, and around the trigger event. This is a valuable troubleshooting feature. If you trigger on a symptom – usually an error of some kind – you can set up the logic analyzer to store data preceding the trigger (pre-trigger data) and capture the fault that caused the symptom. You can also set the logic analyzer to store a certain amount of data after the trigger (post-trigger data) to see what subsequent affects the error might have had. Other combinations of trigger placement are available, as depicted in Figures 12 and 13.

With probing, clocking, and triggering set up, the logic analyzer is ready to run. The result will be a real-time acquisition memory full of data that can be used to analyze the behavior of your SUT in several different ways.

Analysis and Display

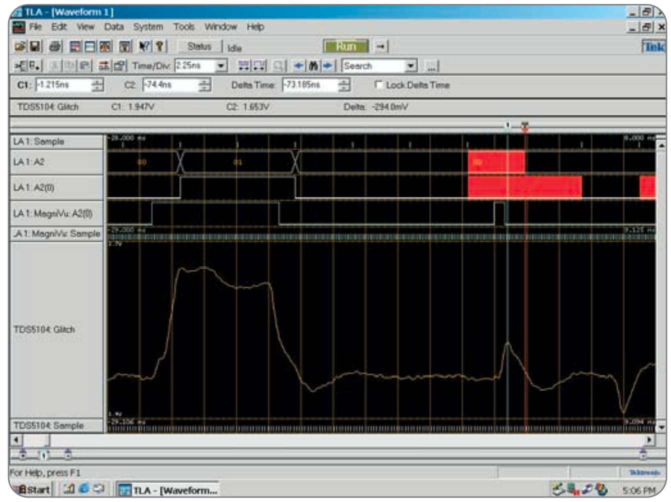
The data stored in the real-time acquisition memory can be used in a variety of display and analysis modes. Once the information is stored within the system, it can be viewed in formats ranging from timing waveforms to instruction mnemonics correlated to source code.

Waveform Display

The waveform display is a multi-channel detailed view that lets you see the time relationship of all the captured signals, much like the display of an oscilloscope. Figure 14 is a simplified waveform display. In this illustration, sample clock marks have been added to show the points at which samples were taken.

The waveform display is commonly used in timing analysis, and it is ideal for:

- Diagnosing timing problems in SUT hardware
- Verifying correct hardware operation by comparing the recorded results with simulator output or data sheet timing diagrams
- Measuring hardware timing-related characteristics:
  - Race conditions
  - Propagation delays
  - Absence or presence of pulses
- Analyzing glitches



▶ ***Figure 15.*** *A time-correlated analog-digital view of an anomaly.*

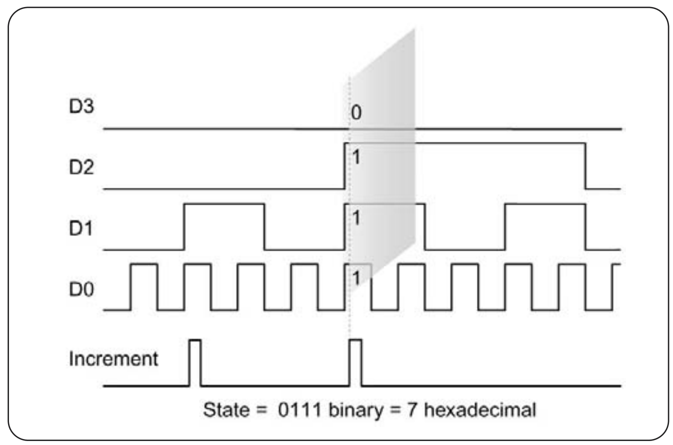
Integrated Analog-digital Display

Older digital systems could tolerate wide variances of digital signals. In newer systems with faster clock edges and data rates, this is no longer the case. As design margins decrease, analog characteristics of digital signals increasingly affect the integrity of your digital system. A time-correlated analog-digital display using a logic analyzer with an oscilloscope gives you the ability to see the analog characteristics of digital signals in relation to complex digital events in the circuit, so you can more easily find the source of anomalies.

Listing Display

The listing display provides state information in user-selectable alphanumeric form. The data values in the listing are developed from samples captured from an entire bus and can be represented in hexadecimal or other formats.

Imagine taking a vertical “slice” through all the waveforms on a bus, as shown in Figure 16. The slice through the four-bit bus represents a sample that is stored in the real-time acquisition memory. As Figure 16 shows, the numbers in the shaded slice are what the logic analyzer would display, typically in hexadecimal form.



▶ ***Figure 16.*** *State acquisition captures a “slice” of data across a bus when the external clock signal enables an acquisition.*

Sample	Counter	Counter	Timestamp
0	0111	7	0 ps
1	1111	F	114.000 ns
2	0000	0	228.000 ns
3	1000	8	342.000 ns
4	0100	4	457.000 ns
5	1100	C	570.500 ns
6	0010	2	685.000 ns
7	1010	A	799.000 ns

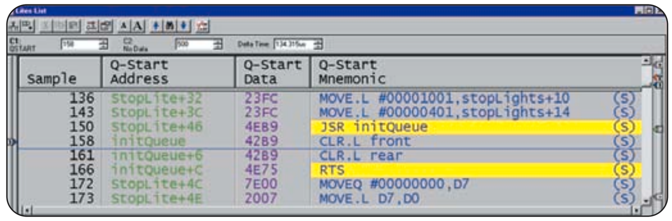
▶ ***Figure 17.*** *Listing display.*

The intent of the listing display is to show the state of the SUT. The listing display in Figure 17 lets you see the information flow exactly as the SUT sees it – a stream of data words.

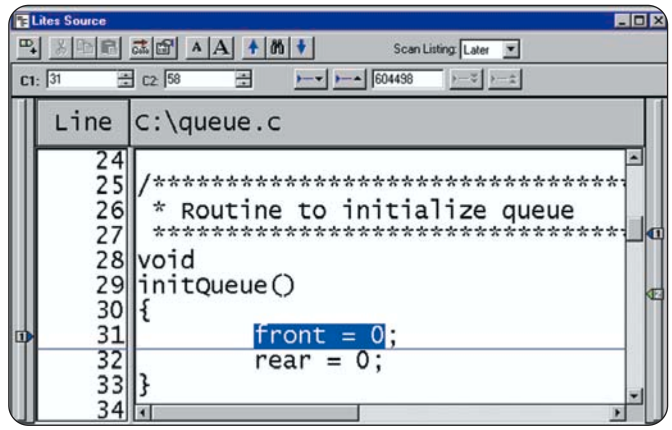
State data is displayed in several formats. The ***real-time instruction trace*** disassembles every bus transaction and determines exactly which instructions were read across the bus. It places the appropriate instruction mnemonic, along with its associated address, on the logic analyzer display. Figure 18 is an example of a real-time instruction trace display.

An additional display, the source code debug display, makes your debug work more efficient by correlating the source code to the instruction trace history. It provides instant visibility of what's actually going on when an instruction executes. Figure 19 is a source code display correlated to the Figure 18 real-time instruction trace.

With the aid of processor-specific support packages, state analysis data can be displayed in mnemonic form. This makes it easier to debug software problems in the SUT. Armed with this knowledge, you can go to a lower-level state display (such as a hexadecimal display) or to a timing diagram display to track down the error's origin.



▶ ***Figure 18.*** *Real-time instruction trace display.*



▶ ***Figure 19.*** *Source code display. Line 31 in this display is correlated with sample 158 in the instruction trace display of Figure 16.*

State analysis applications include:

- Parametric and margin analysis (e.g., setup & hold values)
- Detecting setup-and-hold timing violations
- Hardware/software integration and debug
- State machine debug
- System optimization
- Following data through a complete design

Summary

This document has introduced you to an essential tool for digital system verification and debug. Today's digital design engineers face daily pressures to speed new products to the marketplace. Tektronix logic analyzers answer the need with breakthrough solutions for the entire design team, providing the ability to quickly control, monitor, capture, and analyze real-time system operation in order to debug, verify, optimize, and validate digital systems.

For additional resources, visit **www.tektronix.com/logic_analyzers**

► **Logic Analyzers.** Tektronix logic analyzers provide a wide range of solutions for real-time digital systems analysis, so you can deal with the faster edge speeds and tighter timing margins of next-generation designs with confidence. From single synchronous-bus state and timing analysis to debug and verification of multiple-bus digital systems, there is a Tektronix logic analyzer to meet your needs.

► **Oscilloscopes.** All Tektronix oscilloscopes work with the Tektronix logic analyzers to provide iView™ time-correlated, integrated analog-digital display. The TDS5000, TDS6000 and TDS/CSA7000 Series work with the TLA700 Series to provide full iLink™ Tool Set oscilloscope-logic analyzer integration. iLink includes iConnect™ simultaneous analog-digital acquisition through a single probe, iView, and iVerify™ multi-channel 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) 80-2275577
Italy +39 (02) 25086 1
Japan 81 (3) 3448-3010
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 20 September 2002

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**

09/03 HB/TBD 52W-14266-1

Tektronix
Enabling Innovation