

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

Operator's Manual



LeCroy LSA1000 *Signalyst*

Revision B — October 1998



Corporate Headquarters
700 Chestnut Ridge Road
Chestnut Ridge, NY 10977–6499
Tel: (914) 578 6020, Fax: (914) 578 5985

European Manufacturing
2, rue du Pré-de-la-Fontaine
1217 Meyrin 1/Geneva, Switzerland
Tel: (41) 22 719 21 11, Fax: (41) 22 782 39 15

Internet: www.lecroy.com

Copyright © October 1998, LeCroy. All rights reserved. Information in this publication supersedes all earlier versions. Specifications subject to change.

LeCroy, ProBus and SMART Trigger are registered trademarks of LeCroy Corporation. MathCad is a registered trademark of MATHSOFT Inc. Centronics is a registered trademark of Data Computer Corp. Epson is a registered trademark of Epson America Inc. PowerPC is a registered trademark of IBM Microelectronics. MATLAB is a registered trademark of The MathWorks, Inc. DeskJet, ThinkJet, QuietJet, LaserJet, PaintJet, HP 7470 and HP 7550 are registered trademarks of Hewlett-Packard Company. I^2C is a trademark of Philips.

LSA1000-OM Rev B 1098

Contents

Foreword

Explaining the LSA1000 v
LSA1000: Legacy of the Oscilloscope vi

Chapter 1 — Read This First!

Warranty and Product Support 1–1

Chapter 2 — About the LSA1000

What the LSA1000 Does and How 2–1
Block Diagram 2–2
Capture 2–5
Measure 2–9
Analyze 2–10
The LSA1000 — Front and Back 2–12

Chapter 3 — Installation and Safety

Installation for Safe and Efficient Operation 3–1

Chapter 4 — Connecting PC or Network

Connecting the LSA1000 to its Host 4–1

Chapter 5 — LSA1000 Software Tools

Introducing the LSA1000 Software 5–1
Installing and Launching Software 5–3
Operating the LSA1000 Using *Remote LSA* 5–6
Performance Verification with *Remote LSA* 5–11
Using *ActiveDSO* 5–13
Using *ScopeExplorer* 5–19
About NET_CON 5–21

Contents

Appendix A — Specifications 

Appendix B — NET_CON Source Code

Appendix C — Parameter Measurement

Appendix D — Program Examples

Appendix E — Suggestion Forms

Index



Contents

Explaining the LSA1000

There are two manuals that explain the LSA1000. This, the *Operator's Manual,* takes you through the initial steps and gets you started using the instrument. It explains basics such as how to connect to a PC and use the software tools supplied. Once familiar with the LSA1000's basic operation, use the accompanying *Remote Control Manual.* It contains detailed descriptions of all the remote commands used to operate the LSA1000 from the computer.

About This Manual...

- ***Chapter 1*** describes warranty, maintenance agreements, service and return procedure.
- ***Chapter 2*** covers the instrument's architecture, as well as the fundamental technical concepts behind the LSA1000, to show how these concepts have been integrated to deliver superior performance.
- ***Chapter 3*** gives operating environment and safety information.
- ***Chapter 4*** covers the connection requirements for LSA1000 and host PC, using the Ethernet.
- ***Chapter 5*** offers a step-by-step process to get you started using the LSA1000 and its software tools.
- ***Appendix A*** gives the product's detailed technical performance specifications of the product, and recommends optimum operating conditions.
- ***Appendix B*** is the source code for the Net-Con program.
- ***Appendix C*** offers a general explanation of how the instrument's standard parameters are computed.
- ***Appendix D*** provides several real-life program examples you can use with LSA1000.
- ***Appendix E*** provides suggestions forms that welcome your input for improving both the product and the manuals.

LSA1000: Legacy of the Oscilloscope

In this publication and its companion, the LSA1000 *Remote Control Manual,* references are to be found to functions not directly applicable to the LSA1000. Some examples are references to "time/div", "cursors", and "display".

Their presence in dedicated LSA1000 manuals is owing to the legacy of LeCroy DSOs (Digital Storage Oscilloscopes) in the development of the LSA1000. Although the current practicability of these functions may not immediately be apparent, the basic concepts to which they adhere remain valid for the LSA1000, and the functions are supported by remote control commands.

Moreover, in order to maintain compatibility, the LSA1000's remote commands have been made a subset of the commands for the LeCroy digital oscilloscopes.

Terminology borrowed from the oscilloscope should thus be understood to refer to the LSA1000 *conceptually*, as if it possessed an oscilloscope display. It should be noted, for example, that all commands that refer to "divisions" on a DSO are applicable to the LSA1000: there are eight divisions full scale in the vertical (voltage) direction, and 10 divisions in the horizontal (time) axis.



Foreword

Warranty and Product Support

It is recommended that you thoroughly inspect the contents of the LSA1000 packaging immediately upon receipt. Check all contents against the packing list/invoice copy shipped with the instrument. Unless LeCroy is notified promptly of any missing or damaged item, responsibility for its replacement cannot be accepted. Contact your nearest LeCroy Customer Service Center or national distributor immediately.

Warranty

LeCroy warrants this product for normal use and operation within specifications for a period of three years from the date of shipment. Calibration each year is recommended to ensure in-spec. performance. Spares, replacement parts and repairs are warranted for 90 days. The instrument's firmware has been thoroughly tested and is thought to be functional, but is supplied without warranty of any kind covering detailed performance. Products not made by LeCroy are covered solely by the warranty of the original equipment manufacturer.

Under the LeCroy warranty, LeCroy will repair or, at its option, replace any product returned within the warranty period to a LeCroy authorized service center. However, this will be done only if the product is determined after examination by LeCroy to be defective due to workmanship or materials, and not to have been caused by misuse, neglect or accident, or by abnormal conditions or operation.

Note: ***This warranty replaces all other warranties, expressed or implied, including but not limited to any implied warranty of merchantability, fitness, or adequacy for any particular purpose or use. LeCroy shall not be liable for any special, incidental, or consequential damages, whether in contract or otherwise. The client will be responsible for the transportation and insurance charges for the return of products to the service facility. LeCroy will return all products under warranty with transport prepaid.***

Maintenance Agreements

LeCroy provides a variety of customer support services under Maintenance Agreements. Such agreements give extended warranty and allow clients to budget maintenance costs after the initial three-year warranty has expired. Other services such as installation, training, enhancements and on-site repairs are available through special supplemental support agreements.

Staying Up to Date

LeCroy is dedicated to offering state-of-the-art instruments, by continually refining and improving the performance of LeCroy products. Because of the speed with which physical modifications may be implemented, this manual and related documentation may not agree in every detail with the products they describe. For example, there might be small discrepancies in the values of components affecting pulse shape, timing or offset, and — infrequently — minor logic changes. However, be assured the LSA1000 itself is in full order and incorporates the most up-to-date circuitry.

LeCroy frequently updates firmware and software during servicing to improve LSA1000 performance, free of charge during warranty. You will be kept informed of such changes, through new or revised manuals and other publications.

Nevertheless, you should retain this, the original manual, for future reference to your LSA1000's unchanged hardware specifications.

Service and Repair

Please return products requiring maintenance to the Customer Service Department in your country or to an authorized service facility. The customer is responsible for transportation charges to the factory, whereas all in-warranty products will be returned to you with transportation prepaid. Outside the warranty period, you will need to provide us with a purchase order number before we can repair your LeCroy product. You will be billed for parts and labor related to the repair work, and for shipping.

How to Return a Product

Contact the nearest LeCroy Service Center or office to find out where to return the product. All returned products should be identified by model and serial number. You should describe the defect or failure, and provide your name and contact number. In the case of a product returned to the factory, a Return Authorization Number (RAN) should be used. The RAN can be obtained by contacting the Customer Service Department.

Return shipments should be made prepaid. We cannot accept COD (Cash On Delivery) or Collect Return shipments. We recommend air-freighting.

It is important that the RAN be clearly shown on the outside of the shipping package for prompt redirection to the appropriate LeCroy department.

What Comes with LSA1000

The following items are shipped together with the standard configuration of the LSA1000:

- Getting Started Software CD-ROM
- AC Power Cord and Plug
- *Operator's Manual* (this manual)
- *Remote Control Manual*
- NIST Calibration Certificate
- Declaration of Conformity
- Warranty

Technical Assistance…

Help on installation, calibration, and the use of LeCroy equipment is available from the LeCroy Customer Service Center in your country (*see contact numbers following index*). *See also below.*

Hardware Assistance…

Before contacting us on hardware-related questions (*see below*), try the following:

- Review Chapter 2 of the *Operator's Manual*, which covers general technical concepts and functions such as acquisition, triggering, and memory.
- See Appendix A of the *Operator's Manual* for the LSA1000's technical specifications.

Software Assistance…

Before contacting us with software-related questions, try the following:

- Review Chapter 5 and Appendix D of the *Operator's Manual*, which contains several program examples that can be used as a starting point when writing application-specific software using the LSA1000.
- See the *Remote Control Manual* for questions on specific commands, or if unsure as to which command to use.
- Check *Frequently Asked Questions* at the LeCroy web site: www.lecroy.com

…Or Contact

LeCroy Embedded Signal Analysis:

E-mail: esa@lecroy.com

Fax: ++1 914 578 4485

Phone: (800) 5-LeCroy

1 Read This First!

What the LSA1000 Does and How

The LSA1000 is an ideal instrument for capturing, digitizing and analyzing high-speed electronic signals. Moreover, it has been optimized for system-integration applications with high throughput, ease of use and reliability. The host computer can easily control and retrieve data from the LSA1000 via a 10/100Base-T Ethernet interface. The LSA1000 is an intelligent digitizer that will save valuable development time by acquiring and analyzing complex, high-speed signals. *This chapter covers basic digital concepts and the benefits of using the instrument.*

Overview

The LSA1000 has 1 GHz analog bandwidth. It has two channels, and a 1 GS/s, eight-bit flash Analog-to-Digital Converter (ADC) for each channel. Faster sampling rates can be achieved by combining channels — up to a maximum of 2 GS/s on one channel. The waveform acquisition memory capacity is 100 000 data points per channel in the standard configuration, 500 000 with *Option M1*, 1 million points with *Option M2*, 2 million points with *Option M4*, and 4 million points with option *M8*. Acquisition memory can also be doubled by combining channels — to as much as 8 million data points.

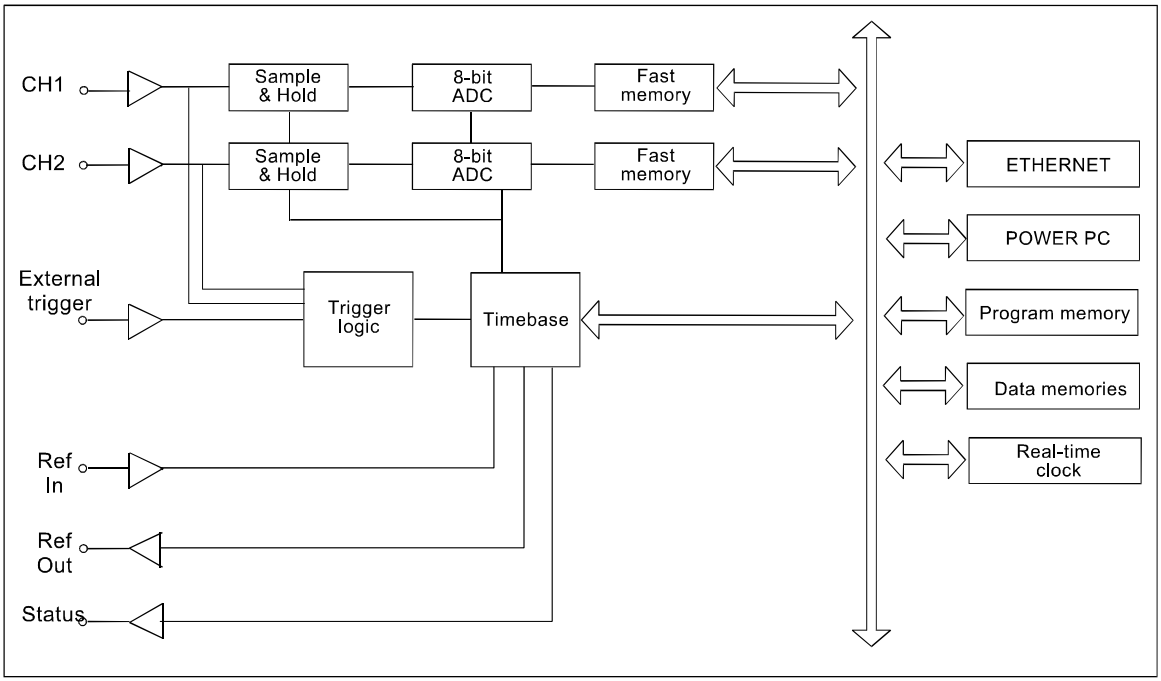
The instrument's central processor is a Motorola PowerPC 603e microprocessor, which performs waveform processing and controls the LSA1000's operation.

A 10/100Base-T Ethernet port is used to directly interface the LSA1000 to the computer for remote control and data transfer.

The LSA1000's entire functionality is via remote control commands, which are fully described in a separate, enclosed manual (*see Remote Control Manual*). Software tools are also provided for application-specific program development and integration. They are explained in the following chapters of the present manual.

Block Diagram

➢ LSA1000 *Signalyst*

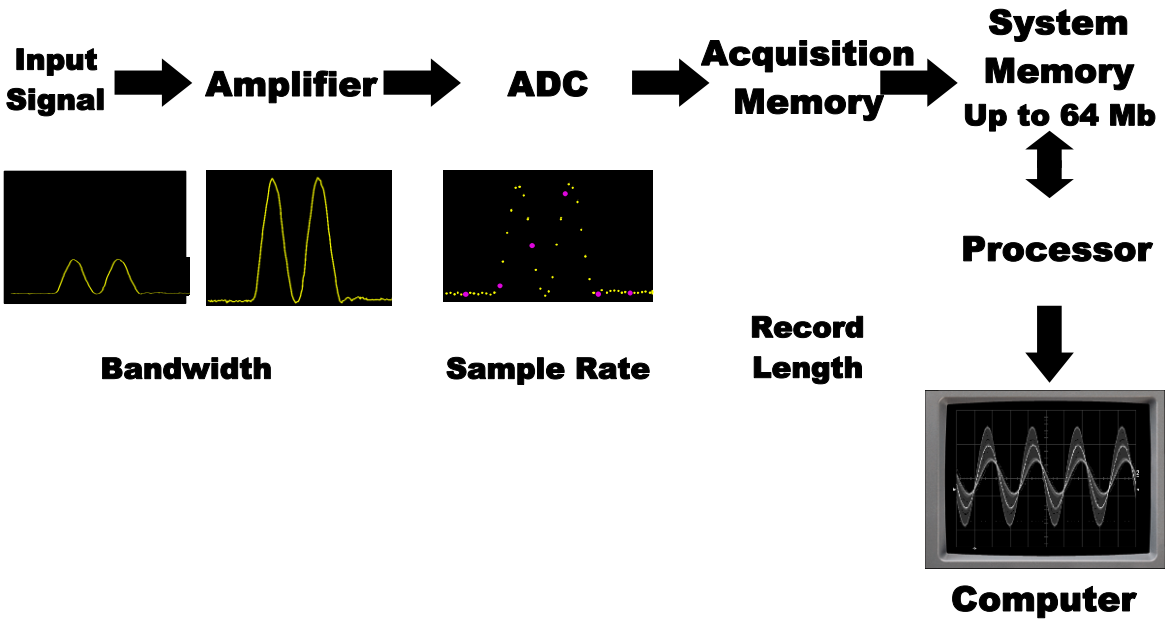


What Can it Do?

The LSA1000 delivers the same powerful performance as a high-end LeCroy DSO (Digital Storage Oscilloscope) in a design optimized for system integration, with:

- ***Capture of one-off (single-shot) events, stored in memory***
- ***On-board analysis of signals, and precise measurement results***
- ***Fast data transfer via Ethernet; remote control of all major functions.***

See this chapter's "Capture", "Measure", "Analyze" and "Transfer" sections for more



Basic Elements

These are the basic elements that make up the LSA1000:

- **Amplifier** — amplifies and conditions the input signal so that it can be measured most effectively
- **Analog-to-Digital Converter (ADC)** — converts the analog signal into digital form by translating it into a series of sample points that are then measured and transformed into digital codes representing the signal samples. Multiple ADC architecture ensures absolute amplitude and phase correlation, maximum ADC performance for multi-channel acquisitions, large record lengths and excellent time resolution.
- **Acquisition Memory System** — stores resulting digital data from ADCs. Standard 100K points per channel simplify transient capture by producing long waveform records that capture even when trigger-timing or signal-speed is uncertain. Combining two channels doubles the acquisition memory length. There are four memories for temporary storage, and four more for waveform zooming and processing.
- **System RAM Memory** of 16 Mbytes as standard. 64 Mbyte optional.

- **Processor** — a PowerPC 603e microprocessor controls the entire system and performs computations and special monitoring.
- **Interfaces** — Ethernet port transfers processed data to the host computer and receives remote control instructions.

Finding the "Right Fit"

For optimum results there must first be a good "fit" between the character of the signals under analysis and the primary specifications of the LSA1000. In assessing the overall signal representation, particular consideration needs to be given to these three cardinal parameters:

- **Bandwidth:** The bandwidth specification of an LSA1000 indicates the ability of its front-end amplifier to faithfully track an incoming signal. The LSA1000's bandwidth is defined as the frequency at which a sinusoidal input signal has been attenuated to about 70 % of its real amplitude. This point, at which the vertical amplitude error is about 30 %, is called the –3 dB point.
- **Sample Rate:** The higher the sample rate the better the signal resolution. This is particularly important for single-shot waveform capture and measurement. The danger with low sample rates is that important information may be lost between samples. A sample rate of at least twice the highest frequency component of the signal is required to avoid aliasing. A sample rate of at least five times is advisable for accurate waveform capture.
- **Memory Length:** Acquisition memory length determines the number of input signal samples that can be stored. The greater the capacity for stored samples, the faster the possible sample rate for a given waveform duration. The signal captured with a long-memory LSA1000 has a greater resolution, whereas in the case of other short-memory instruments, such high sample rates are only available on short waveform-capture time settings. Long-memory LSA1000s allow operation at the highest possible sample rate, and over a far longer capture time.

The LSA1000's fidelity in reconstructing the input signal is affected mainly by these three parameters. The bandwidth must be sufficient to allow all the signal components to pass through the signal-conditioning system. And the sample rate has to be high enough to provide a good definition of the signal. Long memory maintains the sample rate for large time windows.

Capture

In addition to the primary specifications and three parameters described above, capture techniques, the triggering system, channels and ADC specification all impact on LSA1000 performance.

Acquisition Technique

Single-shot acquisition is the LSA1000's basic acquisition technique, which makes the instrument very suitable for the study of signal phenomena that have a low-repetition rate or are not repeated — hence *single*-shot.

An acquired waveform consists of a series of measured voltage values sampled at a uniform rate on the input signal. The acquisition is typically stopped at a fixed time after the arrival of a trigger event as determined by the trigger delay. The acquisition consists of a single series of measured data values associated with one trigger event. The time of the trigger event is measured using the timebase clock. The timing information of a waveform is determined using the trigger event as the definition of time 0. Since each channel has its own ADC, the voltage on each of the input channels is sampled and measured at the same instant. This allows extremely reliable time measurements between channels.

Trigger delay can be selected anywhere within a range that allows the waveform to be sampled from well before the trigger event up to the moment it occurs (100% pretrigger), or at a time starting at the equivalent of 1,000 times of the memory size after the trigger.

The captured input data signal is transferred to memory for measurement and analysis.

Analog–Digital Conversion

High-speed LSA1000 performance is made possible by the use of advanced flash ADCs whose vertical resolution guarantee a clear representation of the signal. These ADCs measure the voltage level at evenly spaced intervals and store the digitized value in high-speed dedicated memory. The shorter the intervals, the faster the digitizing rate — and the finer the time structure able to be seen. The higher the resolution of the ADC, the better its sensitivity to small voltage changes. And the greater its memory capacity, the longer the recording time available.

Channels

The LSA1000 offers two channels for signal input. There are four additional BNC connectors for an external trigger channel, external reference clock in, external reference clock out and acquisition status output. The LSA1000 can be "stacked" to provide more channels and parallel processing with multiple units. And multiple units can be synchronized using external reference clock connections.

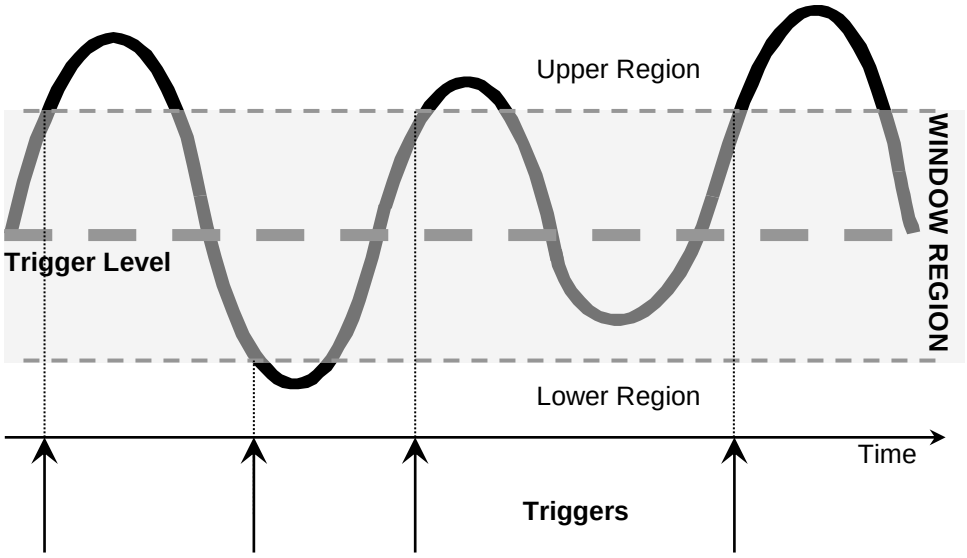
Triggering

Trigger rates of up to 1 GHz are possible with the LSA1000. Single Edge triggers are described by a source, coupling, slope, and level condition.

- **Source** is selected from:
  - CH1, CH2: the acquisition channel signal conditioned for the overall voltage gain, coupling, and bandwidth
  - EXT: the signal applied to the EXT TRIG IN BNC connector. It can be used to trigger the LSA1000 with an ECL signal level. The input impedance is 50 Ω, terminated to –2V. Optionally, this input can be configured to accept a TTL level signal. This configuration must be specified at time of purchase.
- **Coupling** refers to the type of signal coupling at the input of the trigger circuit. LSA1000 supports *DC* coupling on all sources, where all the signal's frequency components are allowed to pass through to the trigger circuit. This coupling mode is used for high-frequency bursts.
- **Slope** selects the direction of the trigger voltage transition that generates a trigger event. In addition to the two traditional choices of *Positive* and *Negative*, the *Window* mode is also included. In Window mode two trigger levels (one upper, one lower) are defined with the window region in between. Triggering occurs when the signal leaves the window region in either direction, *as shown in the diagram on the next page*.
- **Level** defines the source voltage at which the trigger circuit will generate an event (a change in the input signal that satisfies the trigger conditions). The selected level is associated with a trigger source in the same way as the coupling. Note that the trigger level is specified in volts and is normally unchanged when the vertical gain or offset is modified.

The amplitude and range of trigger signals and levels are limited as follows:

- ECL or TTL threshold crossing with EXT as trigger source.



Edge Window Trigger: trigger when the signal leaves the window region.

- With Ch1 or Ch2 as trigger ± 4 divisions. The trigger threshold on a channel is always normalized to the screen and is therefore affected by offset and Vrange. For default configuration of 1V FSR (125mV/div) and 0V offset, the achievable trigger levels are ± 0.5V.

Trigger Modes	
STOP	This command halts the acquisition in any of the three *re-arming* modes: Auto, Normal or Single. Issuing the STOP command prevents the LSA1000 acquiring a new signal. Issue STOP while a single-shot acquisition is under way and the last acquired signal will be kept.
AUTO	In Auto Mode: the LSA1000 automatically captures the signal without waiting for a trigger.
NORM	In Normal Mode, the LSA1000 will continuously capture the signal as long as a valid trigger is present. If not, the last signal is preserved.
SNGL	In Single-Shot Mode, the LSA1000 waits for a single trigger to occur, then captures the signal and stops acquiring.
DELAY	Used to adjust the pre- or post-trigger delay. Pre-trigger adjustment is available from zero to 100 % of the full time-scale in steps of 1 %. The pre-trigger delay is illustrated by the vertical arrow symbol at the bottom of the grid. Post-trigger adjustment is available from 0 to 10 000 divisions in increments of 0.1 of a division.
TIME/DIV	Selects the time per division. Each acquisition has ten divisions.
LEVEL	Adjusts the trigger threshold. The amplitude of trigger signals and the range of trigger levels is limited: ± 4 screen divisions with a channel as trigger source; the full scale range is divided into 8 divisions. Inactive with EXT as trigger source. The trigger sensitivity is better than a third-of-a-screen division.

Measure

Measuring signal characteristics and analyzing performance is at the heart of all test and measurement systems. Some instruments can measure parameters, but analyzing their statistics can be a time-consuming chore. LeCroy's parameter measurements and statistics allow worst-case analysis of performance and make the process much faster.

Parameters

The LSA1000 can determine certain signal properties automatically, using signal parameters. For common measurements on a signal, parameters can be measured in either in the amplitude or time domain. Most users measure the same waveform parameters, such as risetime, falltime, pulse width, overshoot, undershoot, peak voltage, peak-to-peak voltage, maximum, minimum, standard deviation, rms value, frequency, and period. The LSA1000's standard parameters are listed and described in Appendix C of this manual*.

Waveform Math

Waveform mathematics can yield final answers instead of raw data. For example, inputs from voltage and current transformers can be multiplied together to calculate power. An important LeCroy LSA1000 feature is the ability to 'daisy-chain' math functions: a power trace can be integrated to calculate energy, for instance.

* A wide range of additional parameters are available in the specialized software packages, such as WP03, described in those packages' Operator's Manuals.

Analyze

The LSA1000 has the added ability to process and analyze data *on-board*. Once the analog signal has been converted into digital data, the data can be analyzed either by the LSA1000's internal digitizer processor or by an external computer. Most current digitizers have a wide analysis spectrum built-in. Depending on its options, LeCroy LSA1000s offer a selection of the following analytical features.

Time Domain

The optional time domain waveform analysis package — includes summed averaging of up to one million sweeps, derivatives, logarithm, square root, absolute value, ratio, six digital filters and standard arithmetic operations.

Frequency Domain

Spectrum analysis greatly extends the LSA1000's processing power and offers two clear advantages over FFT (Fast Fourier Transform) analyzers: higher frequency components, and the ability to monitor both time and frequency information. Fourier transform converts sampled waveform information into a unique set of sine wave components. The data is usually plotted as amplitude vs frequency, exposing information not easily visible in the time domain (amplitude vs time). Ideally, it can be used for such analyses as measuring frequency components of communication signals and monitoring drift in an oscillator. The frequency resolution of an FFT is directly proportional to the number of time-domain points the FFT algorithm can handle.

Statistical Domain

The existence of measured waveforms in digital representations permits convenient use of the data inherent in those measurements. As well as analysis of signals in the frequency domain and the ability to perform mathematical operations and signal averaging on data, trends can be determined and *histograms* of the data analyzed.

The histogram is a graph of the distribution of values of a measured parameter. For example, when measuring the risetime of a repetitive signal for which all the measurements are equal, the histogram will be a straight vertical line without breadth. However, variations in risetimes create a plot with some horizontal structure, implying variations in the measurements. Your LeCroy LSA1000 is able not

only to create such histograms, but also to allow measurement of their characteristics. This is extremely useful when analyzing pulse amplitudes, widths or timing jitter.

Other LeCroy analysis software packages include *JTA (Jitter and Timing Analysis)* and the specialized testing for magnetic media applications, *DDM (Disk Drive Measurements)*, *PRML (Partial Response Maximum Likelihood)* and *ORM (Optical Recording Measurements)*.

Transfer, Remote Control and Software

A 10/100Base-T Ethernet port is provided for control and transfer of data to and from the LeCroy LSA1000. Remote Control is via a subset of the standard LeCroy DSO remote commands with enhancements specific to the LSA1000. TCP/IP protocol is used for Ethernet communication. See the *Remote Control Manual*.

LeCroy's Versatile Instrument Control Protocol (VICP) allows much of the behavior of GPIB to be emulated using Ethernet.

Users can exercise the full functionality of LSA1000 from the host PC using the following easy-to-use software tools. These tools also simplify users' programming development for own specific applications.

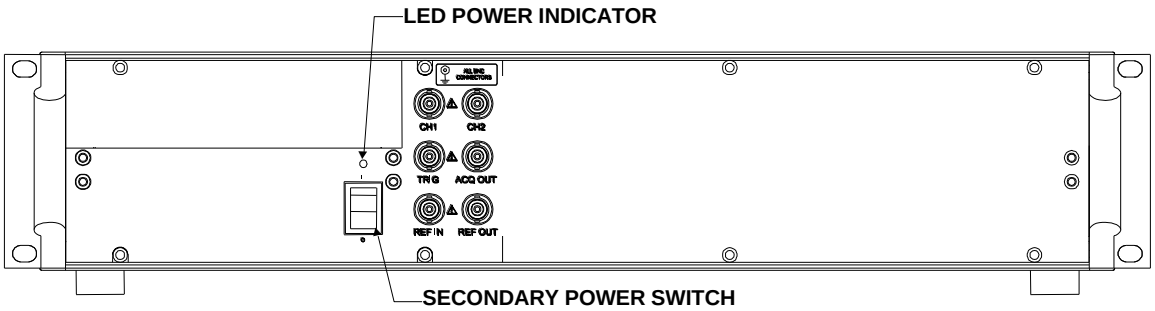
An ActiveX control provides the basic connectivity interface to allow Windows to be able to send remote commands from a PC to the LSA1000 in its native form. Application programs such as Microsoft Excel may use the ActiveX controls to control and retrieve waveform data from the LSA1000 with minimal coding.

Remote LSA program provides a scope-like user interface with live waveform display on a PC. This familiar user interface makes initial performance verification of LSA1000 simple, even for first-time users.

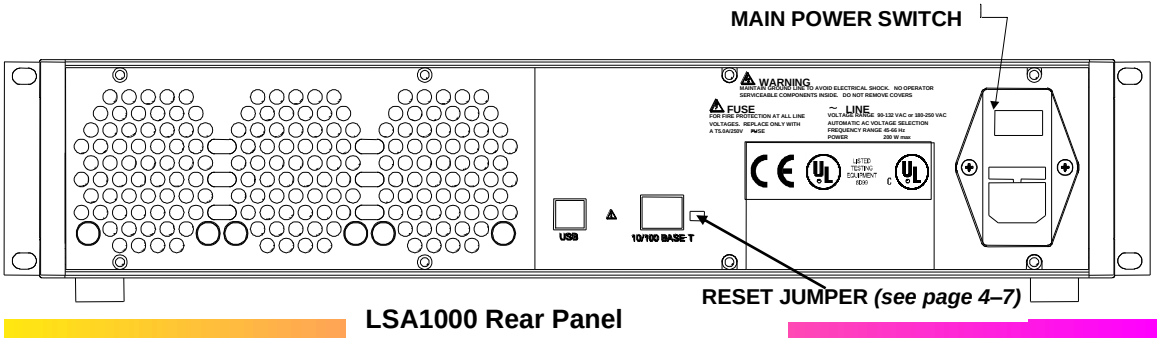
The LSA1000 — Front and Back

Front Panel

- **CH 1 and CH 2:** signal input channels
- **LED Power Indicator:** indicates power on/off status
- **Secondary Power Switch:** power on/off switching
- **External Reference Clock In:** allows the instrument to be synchronized to an external 10 MHz reference.
- **External Reference Clock Out:** can be used to synchronize multiple units of LSA1000.
- **External Trigger In:** ECL or TTL swing level external trigger input
- **Acquisition Status Indication:** indicates the state of the acquisition system in binary format. On receipt of a trigger, the output goes to the HIGH state until the acquisition is complete at which time it will go LOW.



LSA1000 Front Panel



LSA1000 Rear Panel

Rear Panel

- **Ethernet Interface Port:** to transfer data and to receive remote commands from the host computer
- **Main Power Switch**: power on/off switching
- **USB port:** Currently disabled. Provision for future enhancement.

Note: ***All six BNC connections may be located on the LSA1000 rear, rather than front, panel, depending on the instrument's mechanical configuration.***

2 About the LSA1000

Installation for Safe and Efficient Operation

For safe operation of the LSA1000 to its specifications, ensure that the operating environment is maintained within the following parameters:

Operating Environment

- Indoor use only
- Temperature ..... 5 to 40 °C (41 to 104 °F)
- Humidity ..... < 80 % RH non condensing
- Altitude ..... < 4600 m (15090 ft)

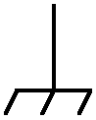
The LSA1000 has been qualified to the following EN61010-1 category:

- Protection Class.....I
- Installation (Overvoltage) Category.....II
- Pollution Degree 2

Safety Symbols

Where the following symbols or indications appear on the instrument's front or rear panels, or elsewhere in this manual, they alert the user to an aspect of safety.

Symbol	Meaning
	**CAUTION: Refer to accompanying documents (for Safety-related information).** ***See elsewhere in this manual wherever the symbol is present, as indicated in the Table of Contents.***
	**CAUTION: Risk of electric shock.**
	On (Supply)
	**Off (Supply)**

Symbol	Meaning
	Earth (Ground) Terminal
	Protective Conductor Terminal
	Alternating Current Only
	Chassis Terminal
	Earth (Ground) Terminal on BNC Connectors
WARNING	Denotes a hazard. If a WARNING is indicated on the instrument, do not proceed until its conditions are understood and met (*see also* CAUTION).



WARNING

Any use of this instrument in a manner not specified by the manufacturer may impair the instrument's safety protection.

The LSA1000 has *not* been designed to make direct measurements on the human body. Users who connect a LeCroy instrument directly to a person do so at their own risk.



Power Requirements

The LSA1000 operates from a 115 V (90–132 V) or 220 V (180–250 V) AC (~) power source at 45–66 Hz, and draws 200 W max power.

No voltage selection is required, since the instrument automatically adapts to the line voltage present.

Fuses

The power supply of the LSA1000 is protected against short-circuit and overload by means of two 5x20mm"T"-rated 5 amp, 250 V AC fuses, located above the mains plug on the rear panel.

Disconnect the power cord before inspecting or replacing a fuse. Open the fuse box by inserting a small screwdriver under the plastic cover and prying it open.



WARNING

For continued fire protection at all line voltages, replace only with fuses of the specified type and rating (see above).

For continued fire protection at all line voltages, replace only with fuses of the specified type and rating (*see above).*

Ground

The LSA1000 has been designed to operate from a single-phase power source, with one of the current-carrying conductors (neutral conductor) at ground (earth) potential. Maintain the ground line to avoid an electric shock.

None of the current-carrying conductors may exceed 250 V rms with respect to ground potential. The LSA1000 is provided with a three-wire electrical cord containing a three-terminal polarized plug for mains voltage and safety ground connection. The plug's ground terminal is connected directly to the frame of the unit. For adequate protection against electrical hazard, this plug must be inserted into a mating outlet containing a safety ground contact.

Cleaning and Maintenance

Maintenance and repairs should be carried out exclusively by a LeCroy technician (*see Chapter 1*). Cleaning should be limited to the exterior of the instrument only, using a damp, soft cloth. Do not use chemicals or abrasive elements. Under no circumstances should moisture be allowed to penetrate the instrument. To avoid electric shocks, disconnect the instrument from the power supply before cleaning.



CAUTION

Electric shock hazard: To avoid personal injury, do not remove covers. No user-serviceable parts inside. Leave repair to qualified personnel.

Power On

Connect the instrument to the power outlet and switch it on using the main On/Off switch, located on the rear panel followed by the

auxiliary On/Off switch, located on the front panel. After the instrument is switched on, auto-calibration is performed.



CAUTION

Do not exceed the maximum specified signal input voltage levels (see Appendix A for details).


3
Installation and Safety

Connecting the LSA1000 to its Host

This chapter describes connecting the LSA1000 to the host PC or network over the standard 10/100Base-T Ethernet. Windows NT and Windows 95 operating systems are supported. *For more about the LSA1000 and the Ethernet, see Chapter 3 of the accompanying Remote Control Manual.*

PC Requirements

For acceptable operational performance with the LSA1000, the following are the minimum PC requirements:

- Pentium class PC
- 32 MB RAM
- 10MB free disk space
- Windows 95 or Windows NT, and
- Ethernet Adapter supporting 10/100Base-T, with a standard RJ45 connection.

Ethernet Connection

The LSA1000 operates over a standard 10/100BaseT Ethernet connection. The instrument can be plugged into a network or operated from a direct connection to a host computer. A different type of cable is required for each of these connections. For a direct connection of LSA1000 to PC, a *cross-over* cable is required, whereas the network connection is made using a *straight* cable.

The LSA1000 has an IP address assigned to it at the factory: it *does not* support Dynamic Host Configuration Protocol (DHCP) or any other automatic address resolution scheme. This IP address may be changed by the user after the connection has been established.

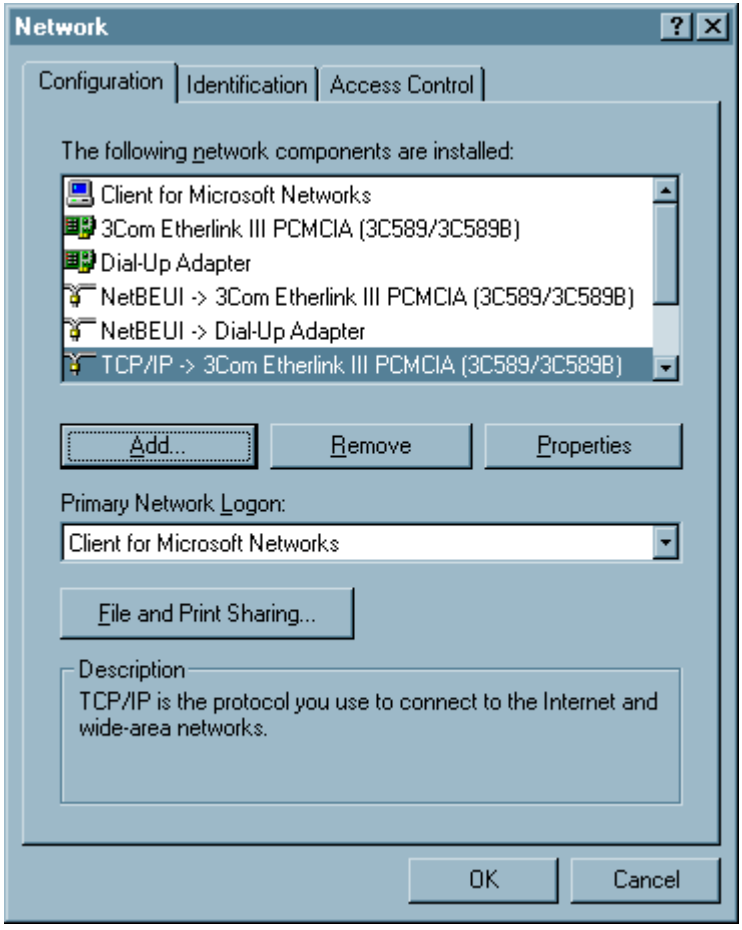
The LSA1000's factory-assigned IP address is: **172.25.1.2**
The LSA1000's factory-assigned Mask is: **255.255.0.0**
The LSA1000's factory-assigned Gateway is: **172.25.0.1**

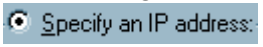
Configuring the PC

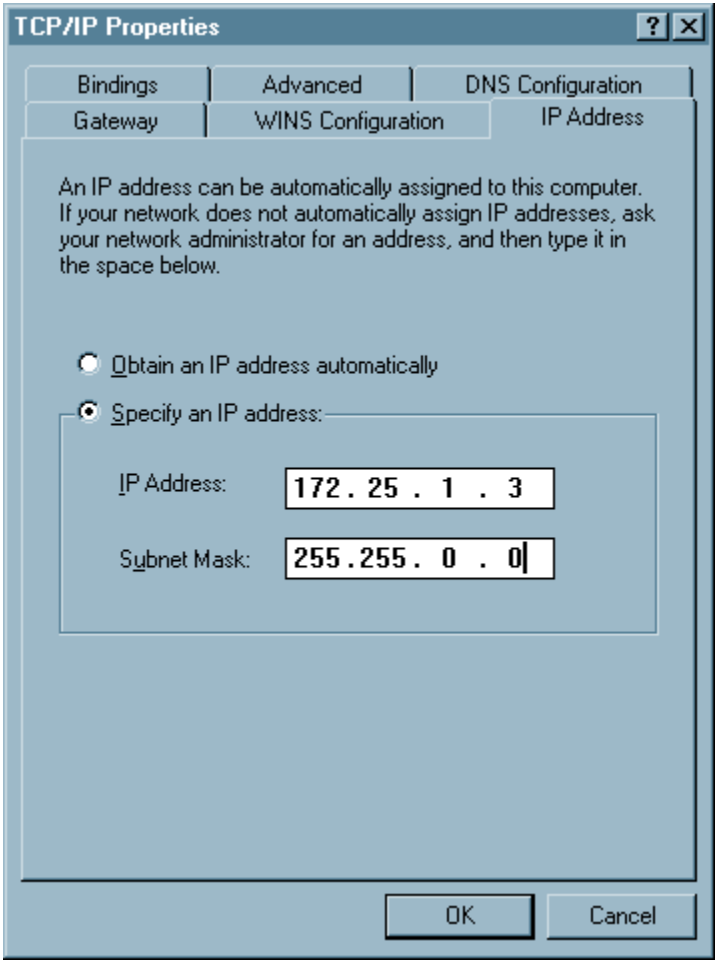
***Note:** The following examples assume that the host PC operates from Windows™ 95. The connection procedure for Windows NT is similar.*

Before establishing a direct connection between the LSA1000 and the host computer, the PC must first be properly configured. A specific TCP/IP address must be assigned — known as "static addressing". But this means that the PC *cannot* be set up to obtain its IP address from a DHCP server. To set the host PC's static address with Windows 95:

1. **Select: Start > Settings > Control Panel**
2. **Double-click on the Network icon in the Control Panel. A nework dialog box similar to this one appears.**



3. **Double-click on the**  **line. A dialog box similar to the one below appears. Select** 



4. **If this has already been selected, then the computer's static address is set and nothing more needs to be done. Cancel out of the TCP/IP and network dialog boxes and close the control panel.**

5. **However, if the address has not already been selected, fill in the IP address and subnet mask as shown above. The subnet mask for 172.25.x.x is 255.255.0.0. If the computer will not be plugged into a network, the above address (or almost any address within the chosen subnet) will do. The only address that will not work is the same one as that of the LSA1000 to be controlled.**

6. **Now click** 

in the TCP/IP Properties dialog box. Depending on the operating system and version, you may need to reboot the computer. If so, a dialog box should alert you to this.

Making Physical Connection

To make the physical connection between the LSA1000 and the host computer:

1. **Connect the LSA1000 to the PC using a cross-over cable (for direct connection).**

2. **Power the LSA1000 unit on. The green LED power indicator on the front panel will light up.**

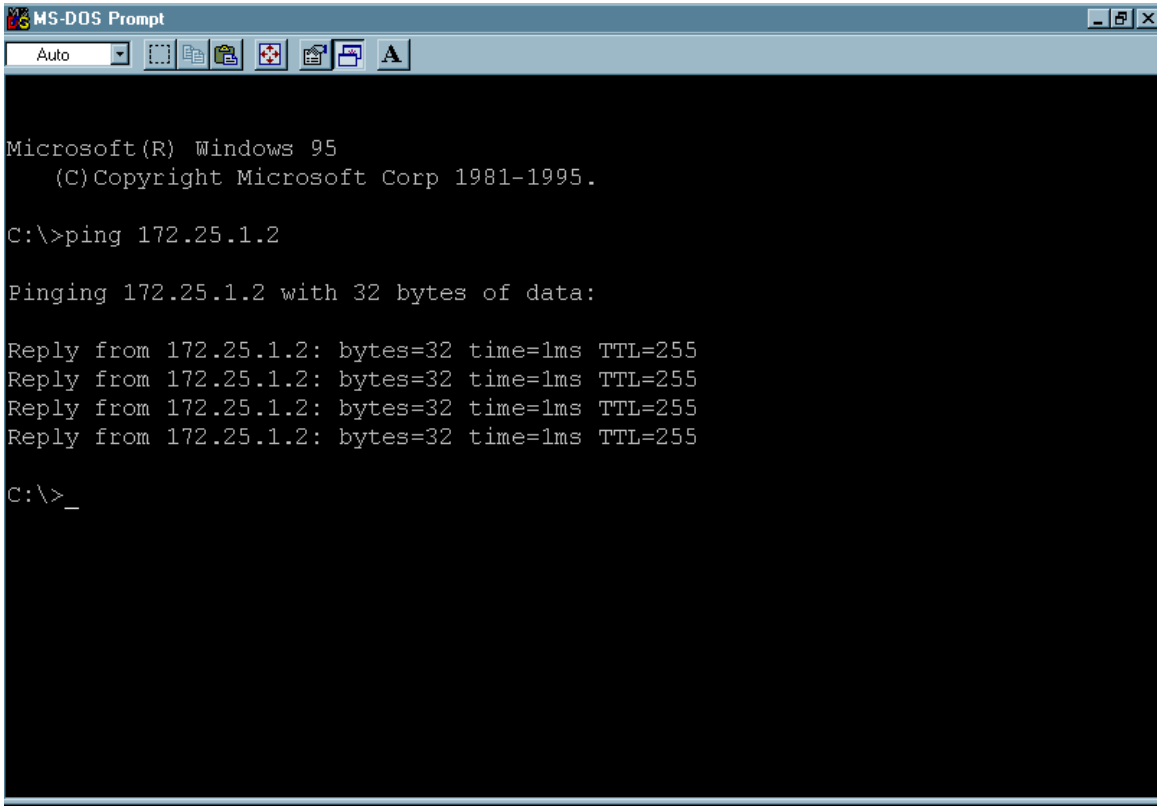
Verifying Connection

***Note:** If your PC does not have TCIP/IP, see the dedicated Windows™ 95 or NT User's Manuals for installation instructions.*

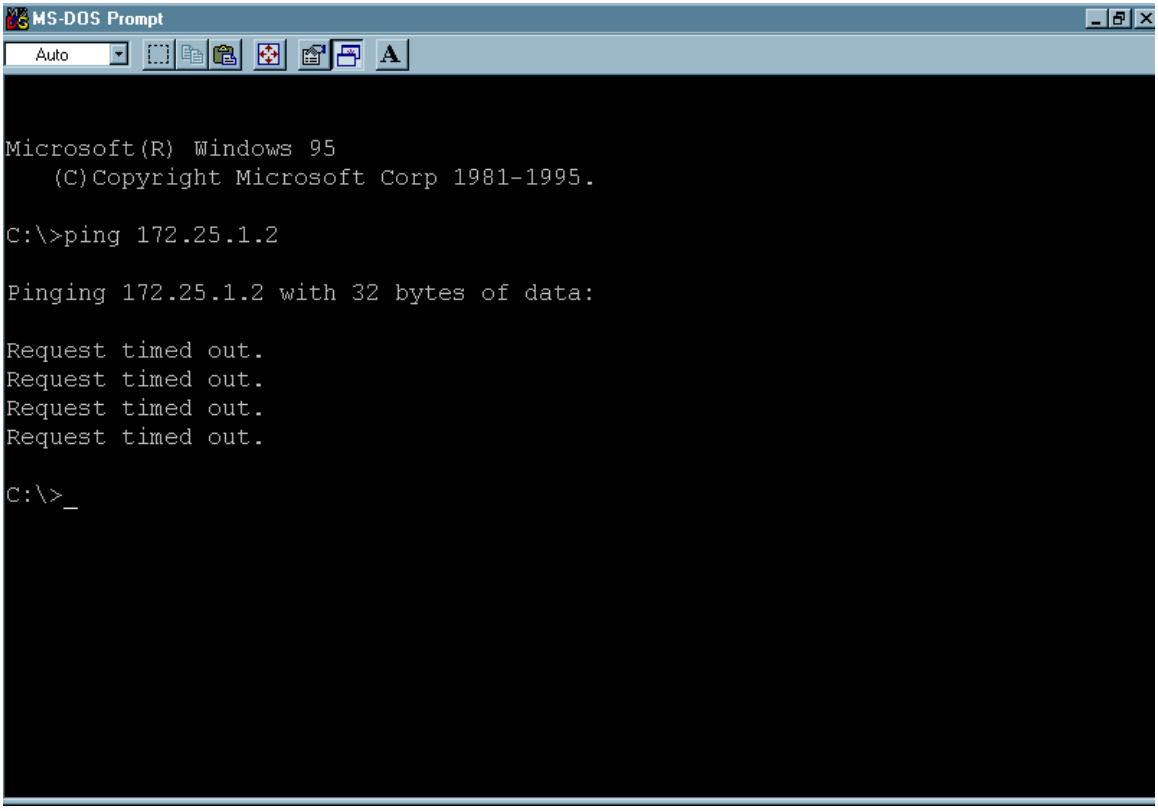
The physical connection and the PC's TCP/IP configuration can be verified using the "**ping**" command, available on both Windows™ 95 and Windows™ NT with TCP/IP network protocol installed. In order to check the network connection between the PC and the LSA1000:

3. **Start MS-DOS Prompt**

4. **Type `ping` <ip_address> where <ip_address> is the static address assigned to the LSA1000. The dialog box on the next page illustrates the result of a successful "ping", with the Ethernet connection shown established. The IP address of the LSA1000 in this case is 172.25.1.2, the factory default address.**



The ping command has sent a message to the instrument and waited for a response. But if a timeout occurs, as is shown in box on the next page, the IP address used for the destination (the LSA1000) is *incorrect* or not within the subnet mask of the PC's IP address.



Network Connection

Check with your network administrator before connecting the LSA1000 to a network. Incorrect addresses on a network can cause both the network and the LSA1000 to behave strangely! However, a network connection ought to be as simple as plugging the LSA1000 into the network. *Proper connection can be verified by following the instructions on verification in the previous section.*

For those users concerned mainly with system throughput, network connection is *not* recommended as the network traffic will slow down the LSA1000 data transfer rate.

Note: The default Gateway is assigned as "172.25.0.1" unless your network has this Gateway available, you must ensure the computer and the LSA1000 are on the same subnet.

Changing IP Address

Once communication between the LSA1000 and host PC has been verified, the IP address can be easily reset to any address using the *Scope Explorer* remote control software tool provided in the Getting Started CD-ROM. To change the IP address, initiate the following remote commands from *ScopeExplorer*'s Terminal. These are described in the *Remote Control Manual (see Chapter 5 of the present manual).*

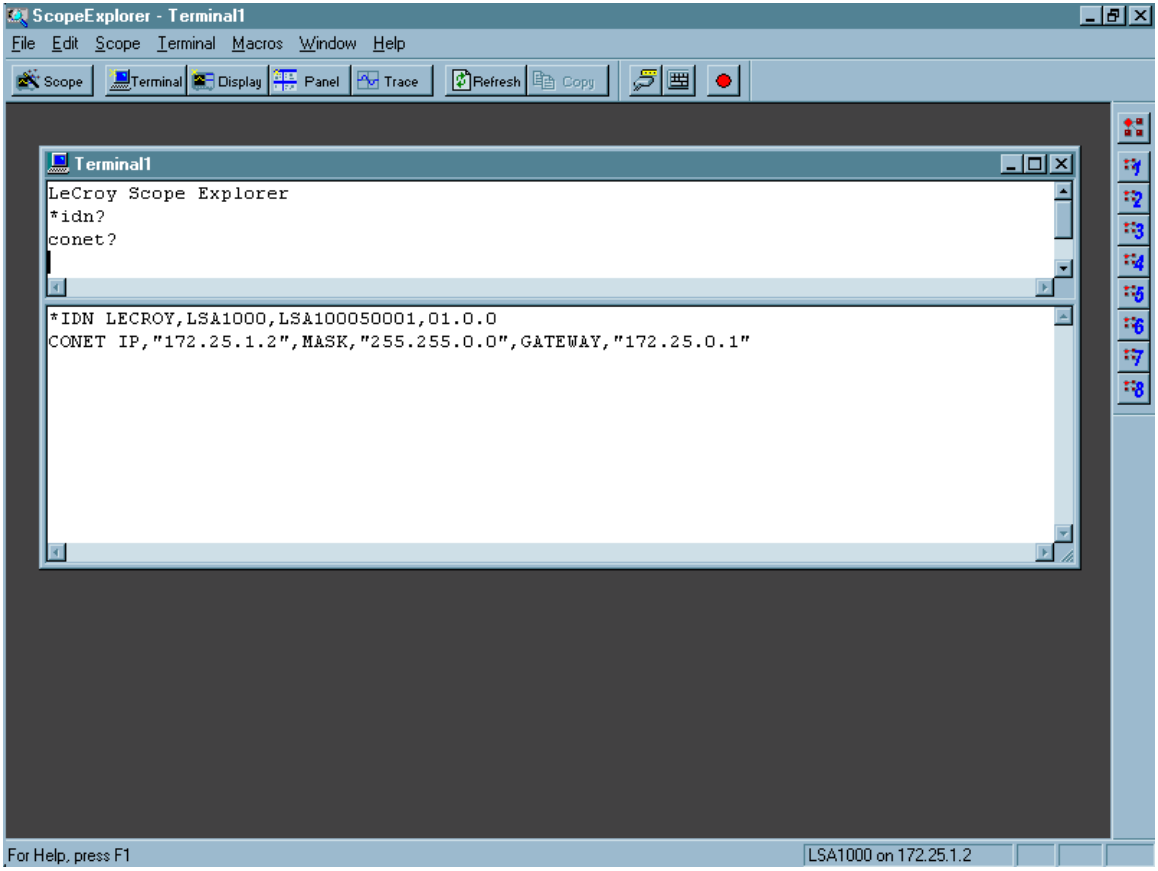
- **`*IDN?`**: Retrieves the unit's name, serial number, and software revision level.
- **`CONET?`**: Returns the current IP address configuration including the mask and gateway. Default is IP, "172.25.1.2", Mask, "255.255.0.0", Gateway, "172.25.0.1".
- **`CONET IP,"172.25.1.5",MASK,"255.255.0.0",GATEWAY,"172.25.0.1"`**: Sets the IP address. This example changes the unit's IP address to 172.25.1.5.

Once the IP address is changed, the unit will no longer respond to the original IP.

If the network settings are unknown or accidentally set to invalid values, the factory default network settings can be recovered by following the procedure below.

Restoring Default Network Settings

The factory default network settings can be restored by powering-up the unit with the Ethernet reset jumper closed. (The reset jumper is accessible through the small hole next to the Ethernet connector).



Example of responses for * IDN and CONET? queries.

4 Connecting to PC or Network

Introducing the LSA1000 Software...

The LSA1000 software tools, provided separately with the instrument, allow you to develop your own application-specific programs, quickly and easily. These tools are: *Remote LSA*, *ActiveDSO*, *ScopeExplorer*, and NET_CON. The files for all software described here are to be found on the CD-ROM.

Remote LSA

The LSA1000 offers a scope-like graphical user interface on the host PC for the Windows 95 or NT environment. This program, *Remote LSA*, provides a live waveform display and the ability to control the basic functions of the LSA1000 remotely from the computer *without programming*. *Remote LSA*'s user interface broadly simulates a LeCroy oscilloscope's front panel. The program is intended as a initial performance verification, and a visual diagnostic tool for use in program development with the LSA1000. *See pages 5–6 and 5–11.*

ActiveDSO

Based on Microsoft's ActiveX control technology, *ActiveDSO* gives leverage to widely-available Microsoft software tools and makes programming within the Microsoft environment easier. *ActiveDSO* simplifies the computer's interface with the LSA1000 and programming within Visual C++, Visual Basic or any other ActiveX-compatible applications. For example, Microsoft Excel can even be used to control and retrieve data directly from the LSA1000. This tool becomes part of the target application and provides seamless access to the full power of the LSA1000. *See page 5–13.*

ScopeExplorer

This PC-based connectivity tool integrates LeCroy instruments with Windows 95 or NT PCs. Connected to the PC by the Ethernet port, the LSA1000 using *ScopeExplorer* allows data and images to be transferred from the instrument and stored in the computer. And because it is designed specifically for use with LeCroy instruments, *ScopeExplorer* allows these tasks to be completed with only a few keystrokes or mouse clicks. *See page 5–19*.

NET_CON

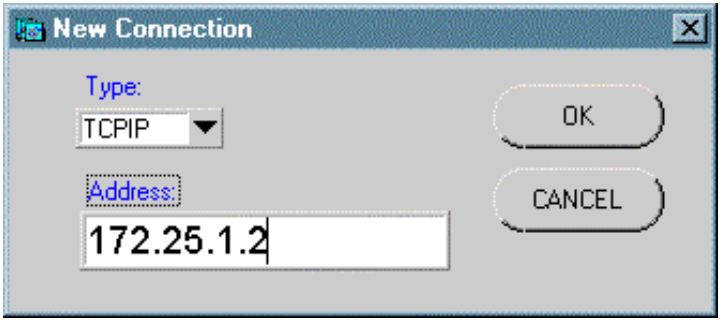
When complied on a target system, this C program demonstrates the implementation of a BSD socket under Microsoft Developer Studio 5.0. The TCP/IP protocol is accessed using this “socket”, which is platform-independent. For non-Windows systems such as the UNIX-based workstation, the “NET_CON” sockets software can be used as the baseline implementation of the LSA1000 network protocol. *The NET_CON source code is provided in Appendix B of this manual. See also page 5–22.*

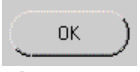
Installing and Launching the Software

Included with the LSA1000 software tools on the CD-ROM are the source codes for all the programs. Provided as reference materials for software development, depending on the application they can be directly copied into your program.

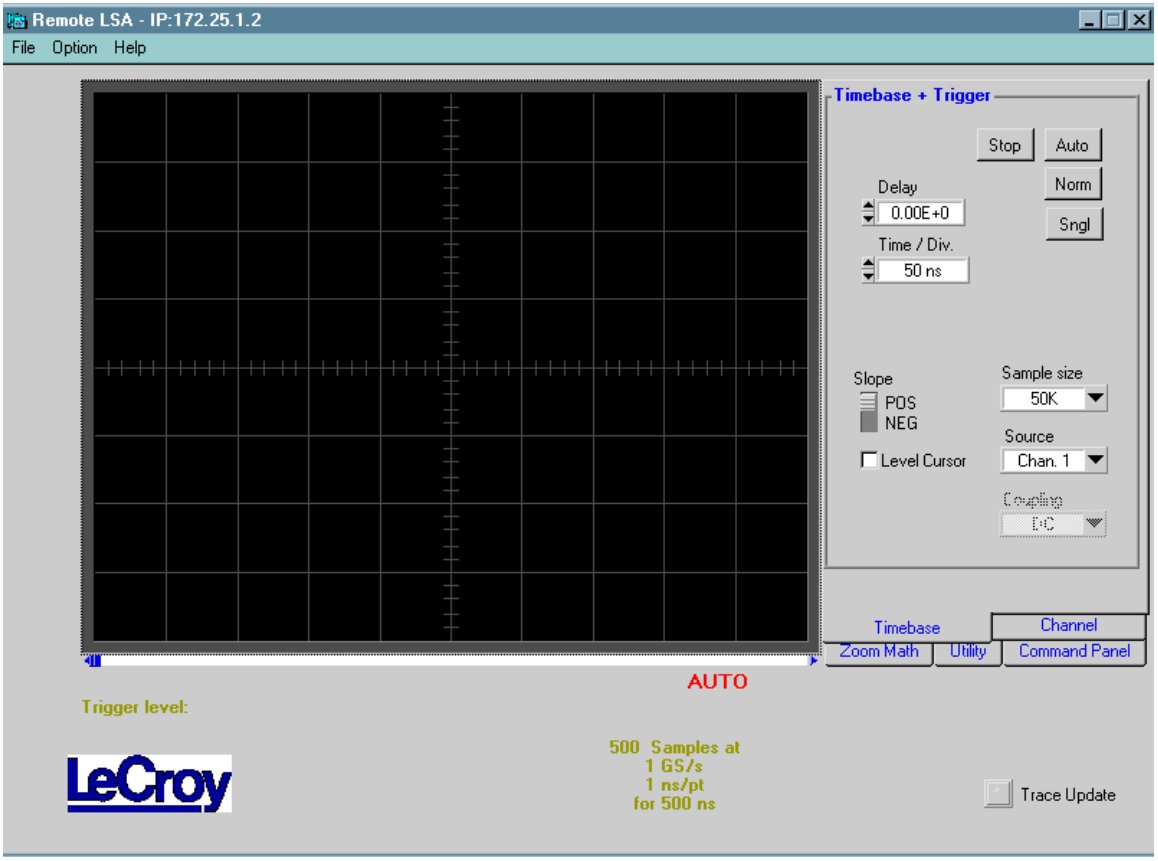
Note: ***Because of the similarities between Windows™ 95 and Windows™ NT, the software tools provided should work in both system with minimal configuration effort. For PCs with Windows™ 3.1, NET_CON should be used as the base line communication protocol implementation on 16-bit compilers.***

1. **Insert the CD into a Windows 95 or NT computer. Executable files (.exe) are available for *Remote LSA*, *ActiveDSO* and *ScopeExplorer,* and each of their program folders contains either a "setupex.exe" or "setup.exe" file.**
2. **Run the desired program's "setupex.exe" file.**
3. **Follow the on-screen instructions to complete the installation.**
4. **Ensure that the Ethernet connection between the PC and LSA1000 is established (*see Chapter 4*).**
5. **Start *Remote LSA*.**
6. **After starting the program, you will be asked for the IP address of the LSA1000, as shown in the following dialog box. Select TCPIP under Type and enter the IP address of the your LSA1000 unit.**



7. **Click** 

Remote LSA **will be launched and the panel shown on the next page opened.**

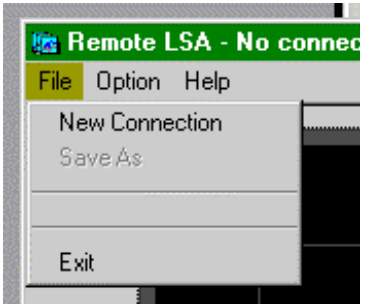


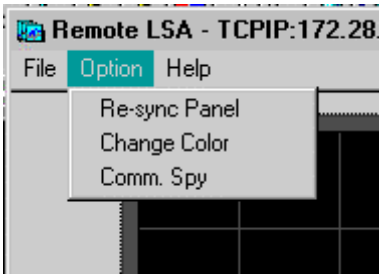
Remote LSA Panel. The Timebase + Trigger control panel is displayed here (see following sections).

Remote LSA Menu

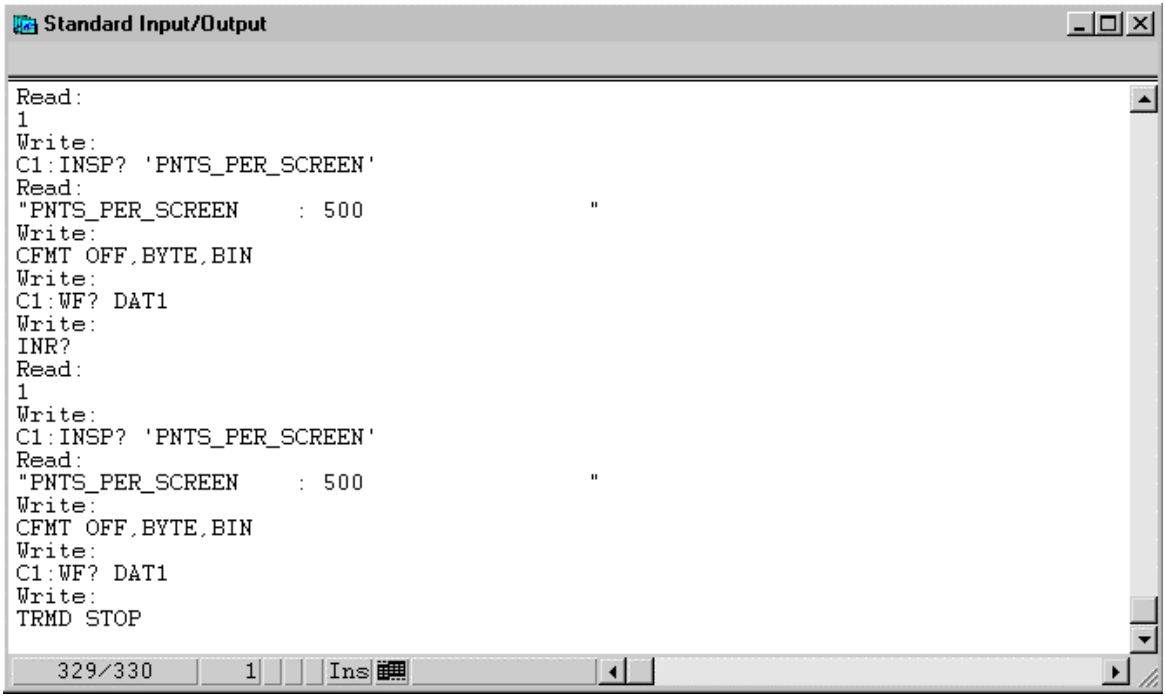
The standard menus are described here:

- **File New Connection** creates a new connection.
- **File Exit** returns to the operating system.





- **Option Re-sync Panel** loads all the current values from the instrument.
- **Option Change Color** opens a window that allows the user to change the color of waveform traces, the grid, and the screen. Chosen colors replace the default colors.
- **Option Comm. Spy** opens a window that shows the ASCII transfer that takes place between the LSA1000 and the host computer. An example of the Comm. Spy window is shown in the figure below. Data from the Comm. Spy can be cut and pasted into other applications to make script files.



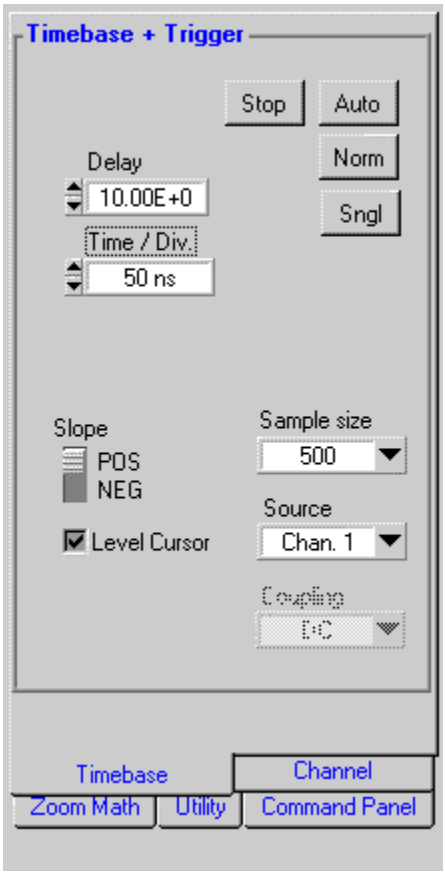
"Comm. Spy" Window.

Operating the LSA1000 Using *Remote LSA*

Remote LSA offers control panels for operating all the major LSA1000 functions, simulating front-panel oscilloscope controls. Five control panels provide this scope-like graphical user interface: "Timebase", "Channel", "Zoom Math", "Utility", and "Command Panel". The desired panel is selected by clicking on its tab.

Timebase + Trigger

This control panel is accessed by clicking on 





halts acquisitions.



starts an acquisition that continuously updates the screen, taking into account possible trigger occurrence or, after a short time interval, if no trigger has been detected.



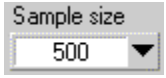
(normal) continuously updates the screen as long as a valid trigger is present.



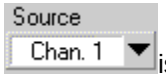
(single) arms the LSA1000 and acquires one trigger.



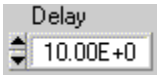
selects the signal transition direction required to qualify the trigger.



is a fly-out menu providing valid record-size options. .



is a fly-out menu for selection of the trigger source. ...



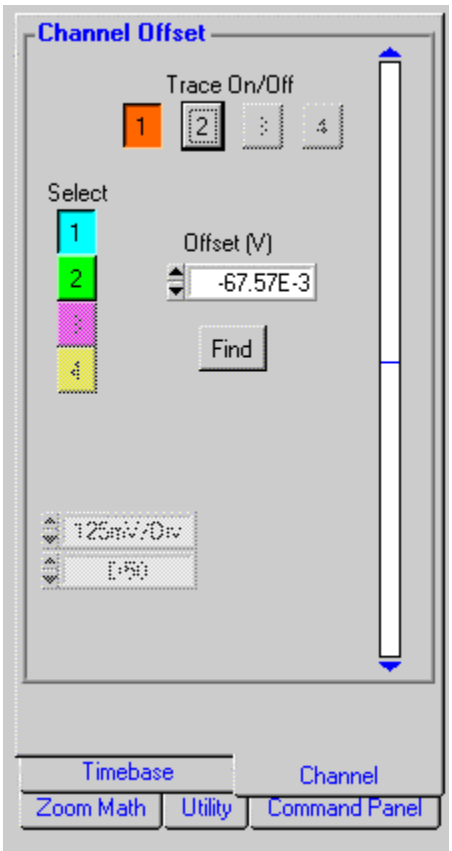
is a spin button used to adjust pre- or post-trigger delay.



is a fly-out menu giving the valid sweep rates.

Channel Offset This control panel is accessed by clicking on 

.

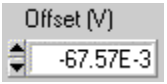




are toggle buttons that change the display status of the corresponding channel.



toggle buttons provide a mechanism for changing the "active" channel without modifying the display status. … .

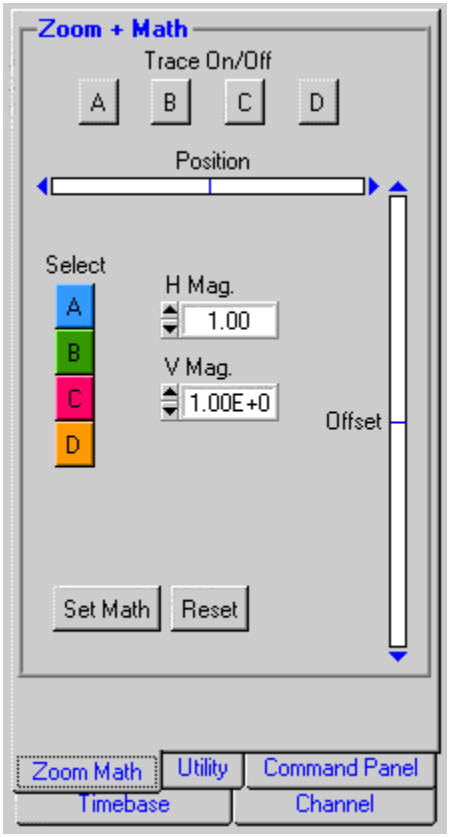


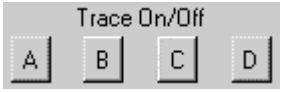
sets the vertical position of the active channel. It is calibrated in volts, ±0.5 V. And the offset scroll bar to its right offers offset control using mouse.



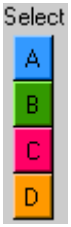
automatically adjusts the offset to display the active channel.

Zoom + Math This control panel is accessed by clicking on Zoom Math.





enables or disables the display status of the corresponding memory.



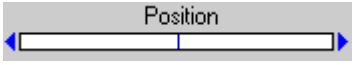
toggle buttons to choose the active memory trace.



spin buttons allow the active memory trace to be expanded horizontally.



spin buttons allow the active memory trace to be expanded vertically.



scroll bar allows the active memory trace to be positioned horizontally on the screen.
.

While the 

scroll bar allows the active memory trace to be positioned vertically on the screen.

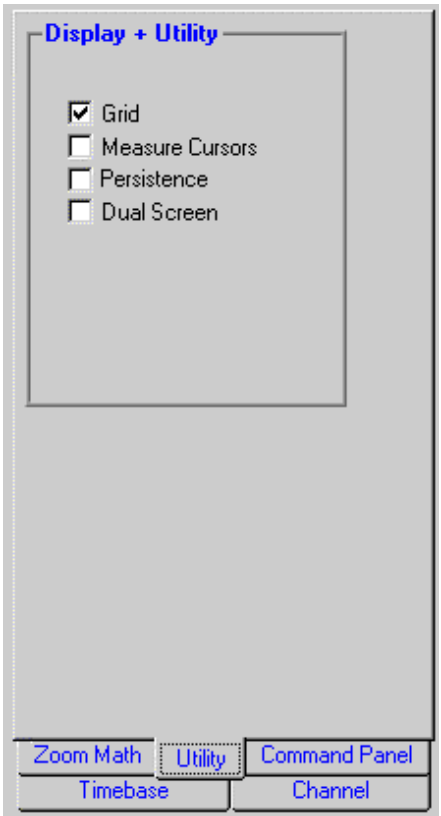


invokes a menu for loading of selected math functions.



restores all traces to the normal ZOOM function.

Display + Utility This control panel is accessed by clicking on Utility



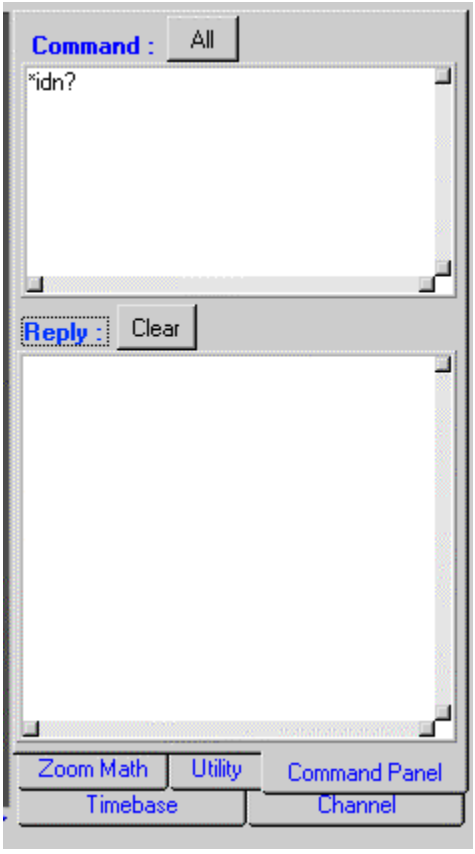
Grid turns the 8 x 10 division display grid on or Grid off.

Measure Cursors enables or Measure Cursors disables a pair of cursors on each active trace. The time and amplitude difference between the cursors on the active trace is displayed at the bottom of the screen.

Persistence enables or Persistence disables Persistence Display Mode.

Dual Screen enables or Dual Screen disables dual display.

Command/Reply This control panel is accessed by clicking on Command Panel



Command : is the window for remote command input to the LSA1000. *Refer to the Remote Control Manual.*

All executes all the commands listed in the window.

Reply : displays the text messages returned by the LSA1000.

Clear erases the reply window.

Performance Verification with *Remote LSA*

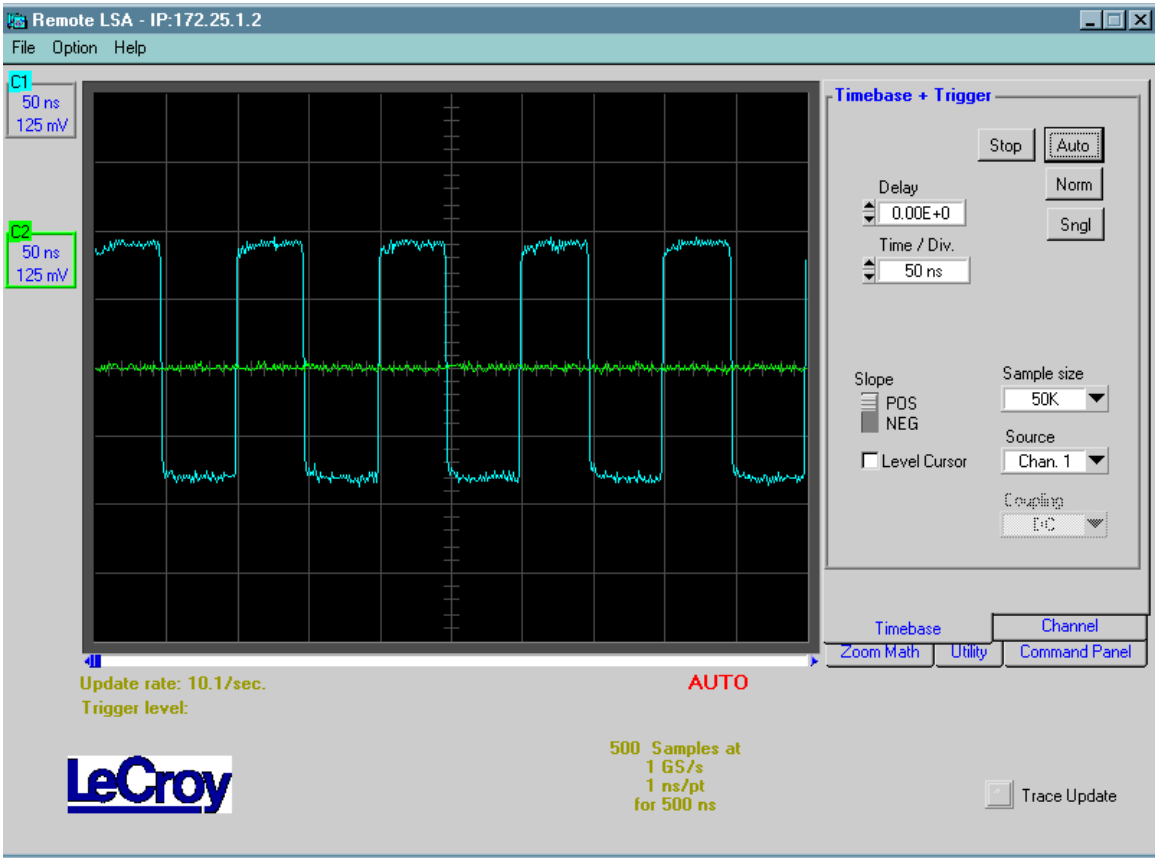
***Remote LSA* is the simplest way to run the LSA1000's basic operations — without the need for remote control commands. Thus we recommend that *Remote LSA* be used for the initial performance verification of the LSA1000.**

1. **Ensure that your PC has the Ethernet connection established with the LSA1000 (*see Chapter 4*).**
2. **Connect the LSA1000's "REF OUT" to "CH 1".**
3. **Start *Remote LSA*.**
4. **In the "Timbase + Trigger" panel click on** 

to set the trigger to automatic.
5. **Select channel** 

from the "Channel" panel.
6. **The program should display a 10MHz square waveform of approximately 400mV amplitude, *as shown on the next page*.**

This process using *Remote LSA* is a simple means of verifiing that the LSA1000 is working properly when first received. *Remote LSA* is also recommended as a visual diagnostic tool during LSA1000 program development.



Remote LSA display of LSA1000's External Reference Out Waveform.

Using *ActiveDSO*

***ActiveDSO* is highly suitable for fast program development in the Microsoft environment. This LSA1000 program is a control of ActiveX, the software technology developed by Microsoft as a subset of its COM model.**

ActiveDSO facilitates the programming with the LSA1000 by providing a ready interface between the instrument and the host computer. Programs such as Visual C++, Visual Basic or Visual Basic for Applications (VBA) can be used under remote control without concern for interfacing complications. *ActiveDSO* acts as the key design structure allowing effective integration of software from the different manufacturers supporting ActiveX containment.

Control Instantiation

This ActiveX component can be instantiated more than once by using the Visual Basic function CreateObject. Once the object is created, it can be initialized by invoking the connection method. ActiveDSO enables control of the LSA1000 from a variety of PC desktop applications. And the complexities of programming using Ethernet are fully encapsulated in this control. For example, with less than 10 lines of VBA code in an Excel macro the spreadsheet can recover pre-scaled waveform data from the LSA1000 (*see the on-line Excel example in ActiveDSO*).

ActiveDSO control can be used in two fundamental ways:

1. As a visible object embedded in an OLE Automation-compatible Client — PowerPoint, for example — showing a captured LSA1000 display image. *See the Embedded Control example below for more details.*
2. As an invisible object accessed via a scripting language — VBA, for example — to remotely control the LSA1000. *See VBA example below for more details.*

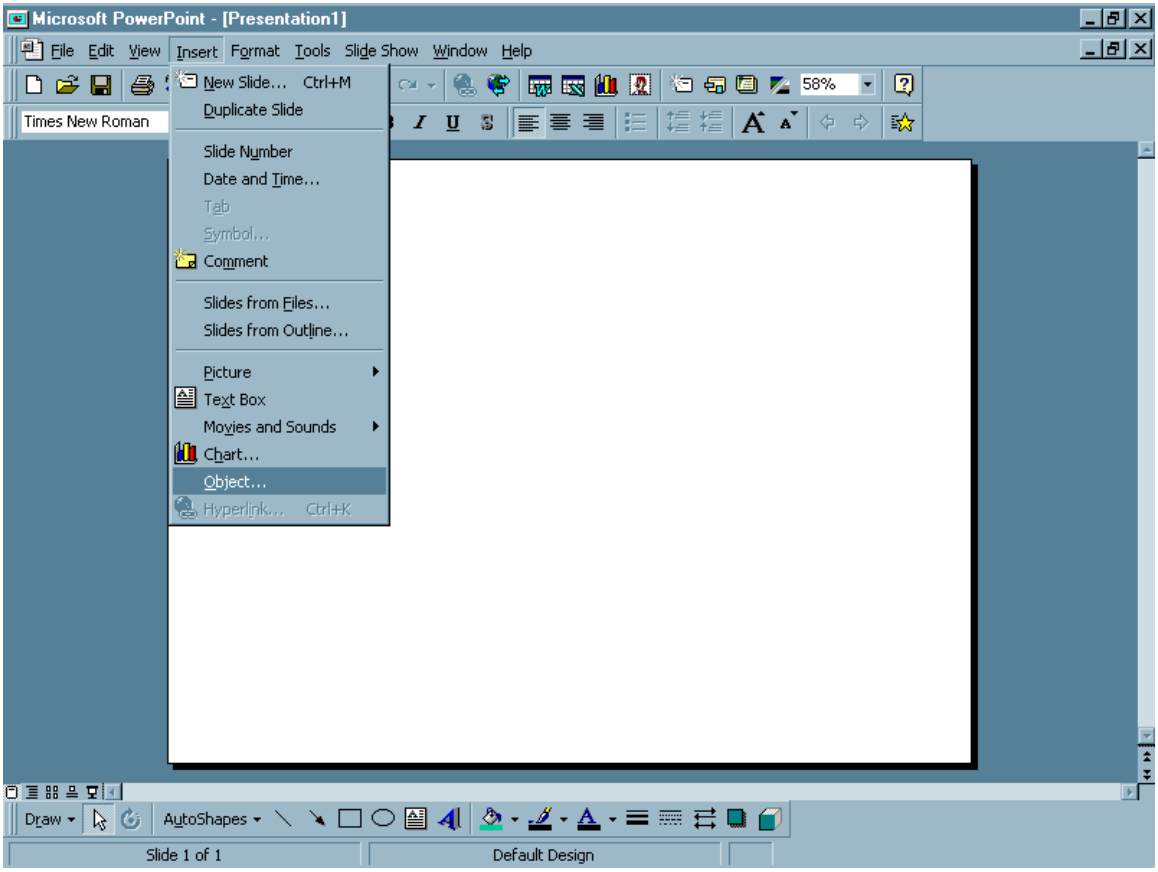
The *ActiveDSO* control may be embedded in any ActiveX containment-capable client and used manually without the need for any programming or scripting.

Example: PowerPoint

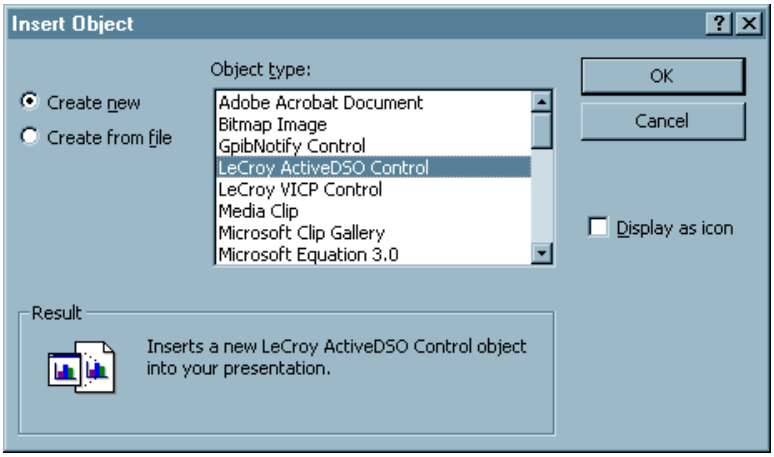
***Note:** this example assumes that PowerPoint 97 is being used. Earlier versions may or may not behave in the same manner.*

This example shows the control being embedded into a Microsoft PowerPoint slide. The waveform captured by the LSA1000 can be simply imported into PowerPoint with just a few mouse clicks:

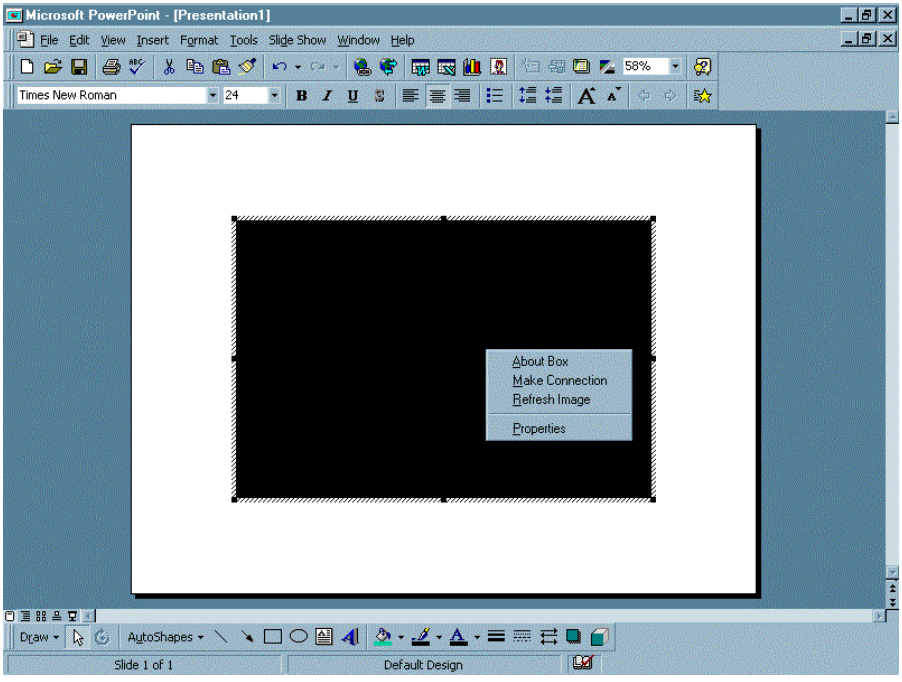
1. **Ensure that the *ActiveDSO* files from the CD-ROM are installed on the PC.**
2. **Check that the PC and LSA1000 are properly connection via the Ethernet.**
3. **Open PowerPoint with a new blank presentation.**
4. **Select "Insert" and then "Object", *as shown here:***



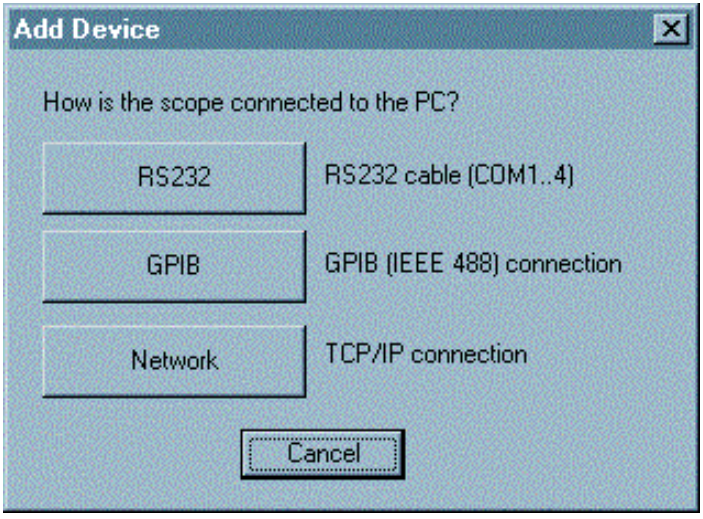
5. **From the pop-up window, select LeCroy ActiveDSO object *as shown here:***



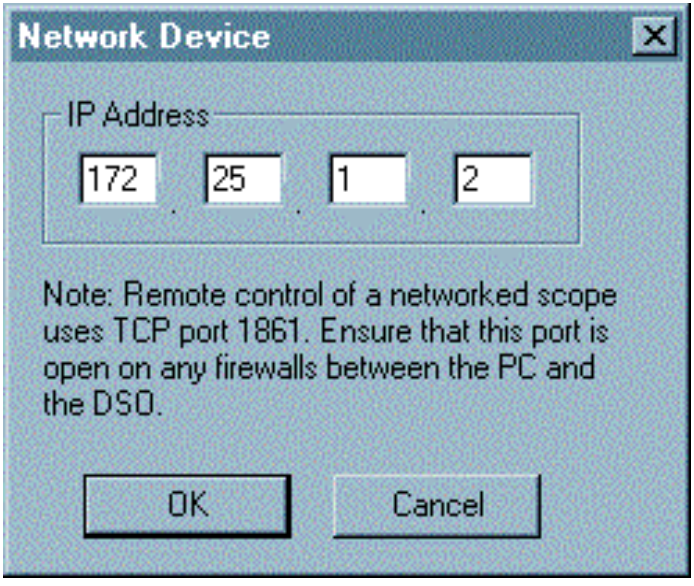
6. **Right-click on the object and select “Make Connection”.**



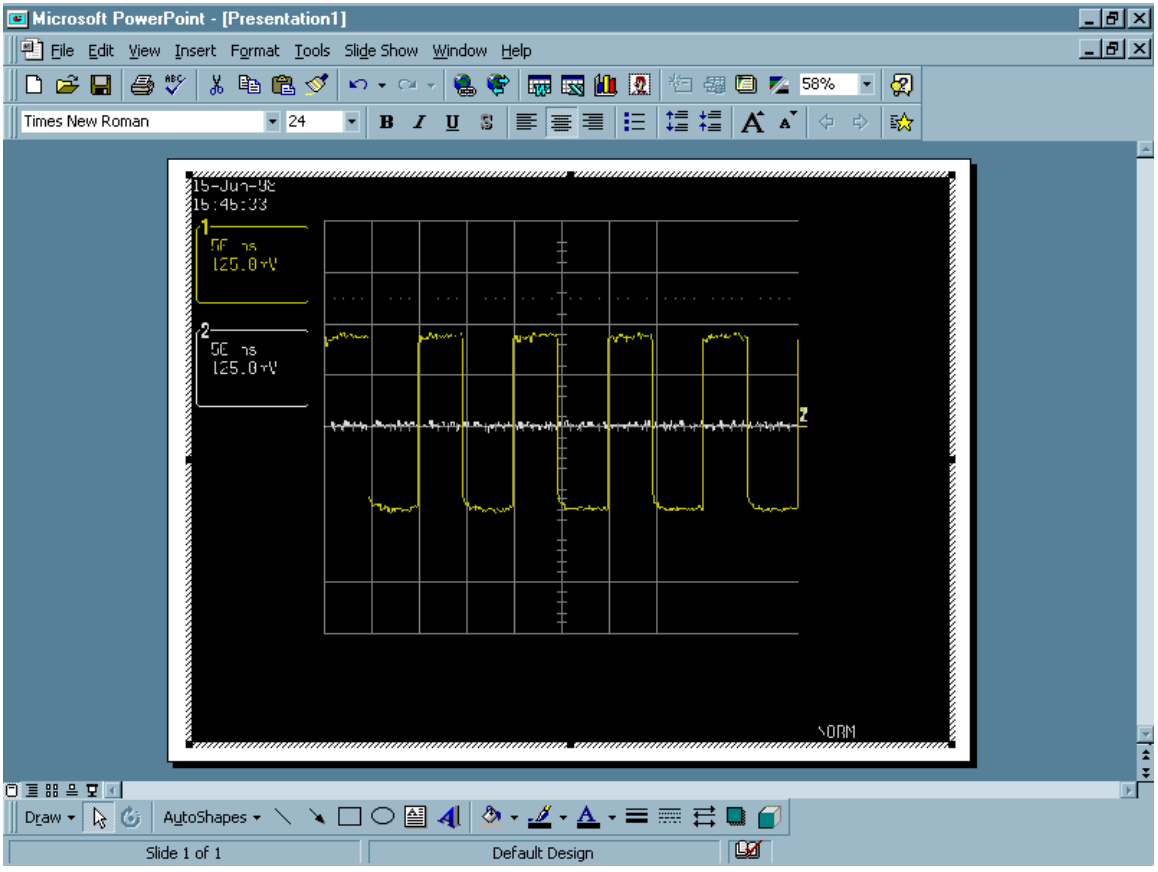
7. **Select "Network" "TCP/IP connection" *as shown here* (for "scope" read LSA1000):**



8. **Enter the LSA1000's IP address and click "OK".**



9. **Right-click on the object again and select the "Refresh Image" menu item. A captured waveform will be displayed *similar the one shown here*:**



LSA1000's captured waveform imported into PowerPoint.

Once the *ActiveDSO*™ object has been properly set within the application, macro script can be created utilizing an object method such as SendString() to send `DISP ON, C1:TRA ON, TRMD AUTO` (*see the Remote Control Manual*). Then RefreshImage() method can be used to update the screen.

***Example:* VBA**

VBA is the programming language built in to many of the more recent Windows applications. It is a subset of Visual Basic that makes using OLE Automation Servers and ActiveX Controls very simple. The following VBA subroutine demonstrates how easy it is to connect to an LSA1000 and send remote commands to it.

```
Sub LeCroyDSOTest()
      Dim o As Object

      Set                    o                    =
CreateObject("LeCroy.ActiveDSO.1")

      Call o.AboutBox          Present         the
control's About box
      Call     o.MakeConnection("IP:172.25.1.2")
      Connect to the unit
      Call o.WriteString("DISP ON", 1)
      Enable the internal display routine
      Call o.WriteString("TRMD AUTO", 1)  Set
the trigger mode to AUTO
End Sub
```

To enter the VBA editor in members of the Microsoft Office suite:

1. **Select Tools > Macro >Visual Basic Editor menu item.**
2. **When the VBA window appears, select the "Insert > Module" menu item**
3. **Copy the above example into the editor window that appears.**

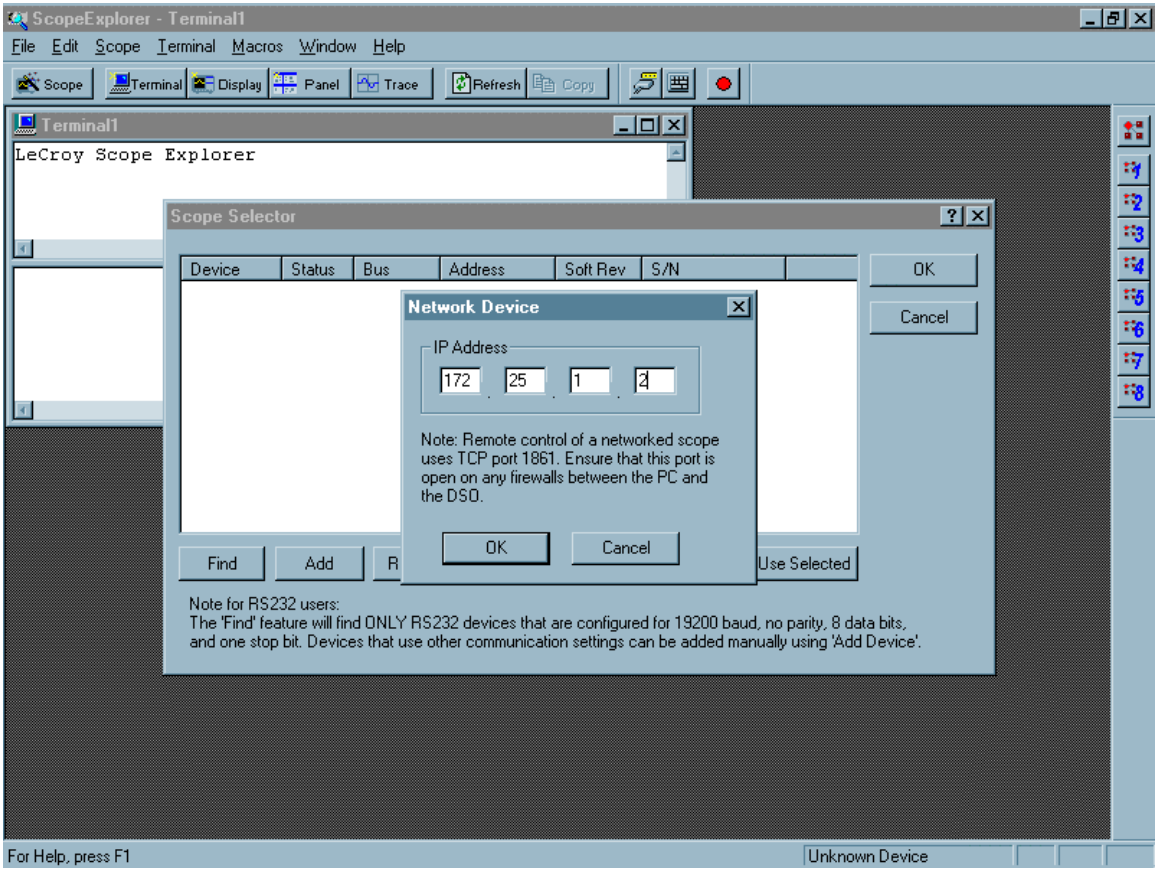
And to execute:

4. **Position the text cursor within the subroutine**
5. **Either select the Run > Run Sub/UserForm or press function key F5.**

***Note:* For more information, see the ActiveDSO on-line Help. On-line Help contains VisualC++ example, explanations of ActiveDSO Methods and Properties.**

Using *ScopeExplorer*

1. Start *ScopeExplorer*.
2. Click on “Scope > Scope Finder”.
3. In the “Scope Selector” window, click “OK”, *as below*.
4. When the ADD DEVICE window opens, select "Network". (If you don’t see “Network” button, press “ALT + N” simultaneously.)
5. A “Network Device” window will appear, *as here*:


ScopeExplorer - Terminal1
File Edit Scope Terminal Macros Window Help
Scope Terminal Display Panel Trace Refresh Copy
Terminal1
LeCroy Scope Explorer
Scope Selector
Device Status Bus Address Soft Rev S/N
OK
Cancel
Network Device
IP Address
172 25 1 2
Note: Remote control of a networked scope uses TCP port 1861. Ensure that this port is open on any firewalls between the PC and the DSO.
OK Cancel
Find Add Use Selected
Note for RS232 users:
The 'Find' feature will find ONLY RS232 devices that are configured for 19200 baud, no parity, 8 data bits, and one stop bit. Devices that use other communication settings can be added manually using 'Add Device'.
For Help, press F1
Unknown Device

6. **Enter the IP address of LSA1000 in the "NETWORK DEVICE" window.**

ScopeExplorer offers:

- ***Terminal: Remote control commands can be sent and data can be retrieved using Terminal.***
- ***Image capture: Internal scope-like "screen" representation of the acquired waveform can be viewed with Display button. Use the Refresh button to refresh the "screen" dump image.***
- ***IP address change: ScopeExplorer can be used to change the IP address of the LSA1000. See Chapter 4 for details.***

ScopeExplorer is supported for all LeCroy instruments and additional information can found on the LeCroy web site: www.lecroy.com

About NET_CON

A C program demonstrating the construction of a TCP/IP socket, NET_CON is an example that can be used as a reference for program development in either a Windows or non-Windows environment. The program is written using the Microsoft Developer Studio version. 5.0. All source code for the Windows environment can be found on the CD and in Appendix B this manual. When the program is run, the LSA1000 will return the ID string "LECROY, LSA1000, LSA1000000000, 01.0.0".

For Unix based workstation, the NET_CON sample program provides the baseline implementation of a stable TCP/IP communication between the LSA1000 and the computer system.

This API (Applications Programming Interface) sits above the TCP/IP protocol stack in all UNIX systems and is also available in Windows 95 and Windows NT. It is for the most part platform-independent, and should allow the same source code to compile and run on each of the supported systems.

The more commonly known WinSock API is derived from the original BSD sockets API and may be used to communicate with BSD sockets based systems. WinSock is the API used by Netscape, Internet Explorer, Telnet, FTP, among others, for PC communication with the Internet.


5
LSA1000 Software Tools

Signal Capture



Acquisition System

Bandwidth (-3 dB):
@50 Ω: DC to 1 GHz
Number of Channels: 2
Number of Digitizers: 2
Sampling Rate:
2 ch. in use: 1 MS/s to 1 GS/s
1 ch. in use: 2 MS/s to 2 GS/s
Sensitivity:

- 250 mV – 2.5 V or 100 mV – 1.0 V Full Scale Range (FSR), depending on specific configuration ordered.
- Higher sensitivity achievable with lower bandwidth

Scale Factors: Attenuator selected for proper FSR, adjustable over 10:1 range.
Offset Range: ≥ FSR
DC Accuracy: typically 1%
Vertical Resolution: 8 bits (up to 11 bits with Enhanced Resolution (ERES) of Option WP01)
Bandwidth Limiter:

- 25 MHz, 200 MHz

Input Coupling: DC
Input Impedance: 50 Ω ± 2 %
Max. Input: 2.5 V rms, and ±5 V peak
Acquisition status output: ECL level output. Indicates acquisition is complete.

Acquisition Modes

Single shot: For transient and repetitive signals

Sequence: Stores multiple events, each time-stamped, in segmented acquisition memories

Deadtime Between Segments: Typically <40 μs, max. 60 μs

Number of Segments Available:

Memory per ch				Segments
100k				500
500k	1M	2M	4M	2000

Timebase System

Capture Time Window:

Acquisition Memory: 100kpts/ch on 2 active channels; 200kpts/ch on 1 active channel. Larger memory options available

Memory	@ 1GS/s	@ 1MS/s
100K	0.1 ms	100 ms
500K	0.5 ms	500 ms
1M	1 ms	1 s
2M	2 ms	2 s
4M	4 ms	4 s

Clock Accuracy: ≤ 10 ppm
Interpolator resolution: 2ns or sample size, whichever is larger
External reference: 10MHz square wave input and output. The instrument can be synchronized to an external reference. Multiple units can also be synchronized.

Triggering System

Modes: Normal, Auto, Single, Stop
Sources: CH1, CH2, Ext;
Slope:
CH1, CH2: Positive, Negative, Window
Ext: Positive, Negative
Coupling: DC
Pre-trigger Recording: 0–100 % of record size adjustable in 1 % increments
Post-trigger Delay: 0–1,000 times record size adjustable in 1% increments
Internal Trigger Range: ± FSR setting
EXT Trigger: ECL trigger standard. Optional TTL trigger in lieu of ECL.
EXT Trigger Max Input: ECL: 0.0V to –4.0V,
TTL: 7.5V to –11.5V
EXT Trigger Range: ECL or TTL signal swing
Trigger Timing: Trigger Date and Time available

Signal Analysis



Waveform Processing

Processing functions: Add, Subtract, Multiply, Divide, Negate, Identity, Summation Averaging, and Sine x/x; four functions performable at one time

Average: Summed averaging of up to 1000 waveforms in the basic instrument; up to 10^6 averages possible with optional WP01 Advanced Waveform Math Package

Extrema: Roof, Floor or Envelope values of from 1 to 10^6 waveforms — with WP01 Option

ERES: Low-Pass digital filter provides up to 11 bits vertical resolution; sampled data always available, even when trace turned off; any of above modes usable without destroying data — with WP01 Option

FFT: Spectral Analysis with five windowing functions and FFT averaging, with optional WP02 Spectrum Analysis Package

Histogramming and Trending: With optional WP03 Parameter Analysis Package, in-depth diagnostics on waveform parameters

Internal Memory

Waveform Memory: Up to four 16-bit Memories.

Processing Memory: Up to four 16-bit Waveform Processing Memories.

Cursor Measurements

Relative Time: Provides time and voltage differences relative to each other

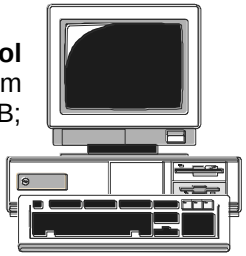
Relative Voltage: Measures voltage differences up to ±0.2% of full-scale

Absolute Time: Measures time relative to trigger and voltage with respect to ground

Absolute Voltage: Measures voltage with respect to ground

Interfacing

Remote Control: By Ethernet for all controls, internal functions
Ethernet Port: 10/100Base-T Ethernet
Ethernet Protocol: TCP/IP
Versatile Instrument Control Protocol (VICP): Protocol that allows Ethernet medium to emulate much of the behavior of GPIB; Remote command set conforms to the IEEE 488.2 standard to ensure compatibility with existing software.



General

Humidity: <80% RH (non-condensing)
Auto-calibration: Ensures specified DC and timing accuracy
Temperature: 5 to 40 °C (41 to 104 °F) rated accuracy 0 to 45°C (32° to 113°F) operating.
Power: 90-132VAC or 180-250VAC; 45-66Hz; 200W (max)
Altitude: Up to 4600 m (15090 ft) operating, 40 °C max
Shock and Vibration: Conforms to selected sections of MIL-PRF-28800F, Class 3
Dimensions: (HWD) 3.5 x 19.0 x 17.5 inches
Weight: 10.5 kg (23 lb.) net, 15 kg (33 lb.) shipping
Warranty: Three years

Conformity



CE Conformity:
EMC: Conforms to EN50081-1 (Emissions) and EN50082-1 (Immunity)
Safety: Conforms to EN61010-1; Protection Class I, Installation (over-voltage) Category II, Pollution Degree 2
(See Declaration of Conformity for further details.)

UL and cUL Listed:
UL Standard: UL3111-1
cUL Canadian Standard: CSA-C22.2 No.1010.1-92



Appendix A: Specifications



Appendix B: NET_CON Source Code

NET_CON.CPP

```
/*
 *                      LSA1000 Sample Network Connection
 *
 *                      Copyright (c) 1998 LeCroy Corporation
 *
 *                      Written By: Ricardo Palacio
 *
 *                      April, 1998
 *
 *
 */


/*
 * $Header:$
 * $Log:$
 *
 */


#include <windows.h>
#include <winsock.h>
#include <stdio.h>
#include <time.h>
#include "net_con.h"


static int                              hSocket;
static int                              sTimeout = 3;
static int                              sWinsockInitFlag = FALSE;
static char                             sCurrentAddress[256];
static int                              sConnectedFlag = FALSE;


#define CMD_BUF_LEN             8192
static char                     sCommandBuffer[CMD_BUF_LEN];



int TCP_Connect(char *ip_address)
{
SOCKADDR_IN    serverAddr;
int sockAddrSize = sizeof (SOCKADDR), result;
WORD wVersionRequested;
WSADATA wsaData;
const int resp = 1;
char tmpStr[512];
fd_set wr_set = {1, {0}};
TIMEVAL tval;
```

```
unsigned long argp;



        if (sConnectedFlag==TRUE)
                return -1;

        strcpy(sCurrentAddress, ip_address);

        tval.tv_sec = sTimeout;
        tval.tv_usec = 0;



        if (!sWinsockInitFlag)
        {
                wVersionRequested = MAKEWORD(1, 1);
                if (WSAStartup(wVersionRequested, &wsaData) != 0)
                {
MessageBox(0, "Unable to initialize the Windows socket environment.", "ERROR", MB_OK);
                        return -1;
                }
                sWinsockInitFlag = TRUE;
        }

    /* build server socket address */
        serverAddr.sin_family = AF_INET;
        serverAddr.sin_port = htons (SERVER_PORT);

        if ((serverAddr.sin_addr.s_addr = inet_addr(ip_address)) == -1)
        {
            MessageBox(0, "Bad server address", "ERROR", MB_OK);
            return -1;
        }

    /* create client's socket */
if ((hSocket = socket(AF_INET, SOCK_STREAM, 0)) == INVALID_SOCKET)
        {
MessageBox(0, "Unable to create client's socket.", "ERROR", MB_OK);
            return -1;
        }

    if (setsockopt(hSocket, IPPROTO_TCP, TCP_NODELAY, (char*)&resp, sizeof(resp)) != 0)
        {
                MessageBox(0, "Unable to set socket option to TCP_NODELAY", "ERROR",
MB_OK);
                return -1;
        }


        wr_set.fd_array[0] = hSocket;
```

```
        argp = 1;//non blocking mode
        ioctlsocket(hSocket, FIONBIO, &argp);

        connect(hSocket, (SOCKADDR FAR *) &serverAddr, sockAddrSize);

        result = select(hSocket, NULL, &wr_set, NULL, &tval);

        argp = 0;//blocking mode
        ioctlsocket(hSocket, FIONBIO, &argp);

        /* connect to server (scope) */
        if (result < 1)
        {
            sprintf(tmpStr, "Unable to make connection to IP:%s", ip_address);
               MessageBox(0, tmpStr, "TIMEOUT", MB_OK);
            return -1;
        }

        sConnectedFlag = TRUE;

        return 0;
}
```

```
int TCP_Disconnect(void)
{

if (sConnectedFlag != TRUE)
               return -1;

closesocket(hSocket);
sConnectedFlag = FALSE;

return 0;
}
```

```
int TCP_WriteDevice(char *buf, int len, BOOL eoi_flag)
{

        TCP_HEADER header;
```

```
        int result, bytes_more, bytes_xferd;
        char *idxPtr;



        if (sConnectedFlag != TRUE)
                return -1;

        if (len < CMD_BUF_LEN)
                strcpy(sCommandBuffer, buf);

        // set the header info
        header.bEOI_Flag = DATA_FLAG;
        header.bEOI_Flag |= (eoi_flag)? EOI_FLAG:0;
        header.reserved[0] = 1;
        header.reserved[1] = 0;
        header.reserved[2] = 0;
        header.iLength = htonl(len);


        // write the header first
        if (send(hSocket, (char *) &header, sizeof(TCP_HEADER), 0) != sizeof(TCP_HEADER))
        {
            MessageBox(0, "Unable to send header info to the server.", "ERROR", MB_OK);
            return -1;
        }


        bytes_more = len;
        idxPtr = buf;
        bytes_xferd = 0;
        while (1)
        {
                // then write the rest of the block
                idxPtr = buf + bytes_xferd;


                if ((result = send(hSocket, (char *) idxPtr, bytes_more, 0)) < 0)
                {
                        MessageBox(0, "Unable to send data to the server.", "ERROR",
MB_OK);
                        return -1;
                }

                bytes_xferd += result;
                bytes_more -= result;
                if (bytes_more <= 0)
                        break;
        }

        return 0;
}
```

```
int TCP_ClearDevice(void)
{

        if (sConnectedFlag != TRUE)
                return -1;

        TCP_Disconnect();
        TCP_Connect(sCurrentAddress);
        return 0;
}
```

```
int TCP_ReadDevice(char *buf, int len, int *recv_count)
{
        TCP_HEADER header;
        char tmpStr[512];
        int result, accum, space_left, bytes_more, buf_count;
        char *idxPtr;
        fd_set rd_set = {1, {0}};
        TIMEVAL tval;



        if (sConnectedFlag != TRUE)
                return -1;

        *recv_count = 0;

        if (buf==NULL)
                return -1;

        rd_set.fd_array[0] = hSocket;
        tval.tv_sec = sTimeout;
        tval.tv_usec = 0;



        memset(buf, 0, len);
        buf_count = 0;
```

```
        space_left = len;

        while (1)
        {
                // block here until data is received of timeout expires
                result = select(hSocket, &rd_set, NULL, NULL, &tval);
                if (result < 1)
                {
                        TCP_ClearDevice();
                        MessageBox(0, sCommandBuffer, "Read timeout", MB_OK);
                        return -1;
                }
                // get the header info first
                accum = 0;
                while (1)
                {
                        memset(&header, 0, sizeof(TCP_HEADER));

if ((result = recv(hSocket, (char *) &header + accum, sizeof(header) - accum, 0)) < 0)
                        {
                                TCP_ClearDevice();
MessageBox(0, "Unable to receive header info from the server.", "ERROR", MB_OK);
                                return -1;
                        }

                        accum += result;
                        if (accum>=sizeof(header))
                                break;
                }

                header.iLength = ntohl(header.iLength);
                if (header.iLength < 1)
                        return 0;

                // only read to len amount
                if (header.iLength > space_left)
                {
                        header.iLength = space_left;
                        sprintf(tmpStr, "Read buffer size (%d bytes) is too small", len);
                        MessageBox(0, tmpStr, "ERROR", MB_OK);
                }

                // read the rest of the block
                accum = 0;
                while (1)
                {
                        idxPtr = buf + (buf_count + accum);
                        bytes_more = header.iLength - accum;
                        if ((space_left-accum) < TCP_MINIMUM_PACKET_SIZE)
                        {
                                TCP_ClearDevice();
```

```
sprintf(tmpStr, "Read buffer needs to be adjusted, must be minimum of %d bytes",
TCP_MINIMUM_PACKET_SIZE);
                                MessageBox(0, tmpStr, "ERROR", MB_OK);
                                return -1;
                        }

if ((result = recv(hSocket, (char *) idxPtr, (bytes_more>2048)?2048:bytes_more, 0)) < 0)
                        {
                                TCP_ClearDevice();
MessageBox(0, "Unable to receive data from the server.", "ERROR", MB_OK);
                                return -1;
                        }


                        accum += result;
                        if (accum >= header.iLength)
                                break;
                        if ((accum + buf_count) >= len)
                                break;
                }
                buf_count += accum;
                space_left -= accum;

                if (header.bEOI_Flag & EOI_FLAG)
                        break;
                if (space_left <= 0)
                        break;
        }

        *recv_count = buf_count;

        return 0;
}




int TCP_SetTimeout(int seconds)
{

        sTimeout = seconds;
        return 0;
}




int main(int argc, char *argv[])
{
```

```
    char replyBuf[512];
        int read;


        if (argc < 2)
        {
                printf("\nEXAMPLE: net_con 172.28.11.22\n");
                return 0;
        }


    if (TCP_Connect(argv[1]))
                return 0;


        if (TCP_WriteDevice("*idn?\n", 6, TRUE))
        {
                TCP_Disconnect();
                return 0;
        }


        if (TCP_ReadDevice(replyBuf, 512, &read))
        {
                TCP_Disconnect();
                return 0;
        }

        if (TCP_Disconnect())
                return 0;

    printf("Scope's reply: %s\n", replyBuf);

    return 0;
}
```

```
                                    #define SERVER_PORT                 1861

#define EOI_FLAG                    0x01
#define SRQ_FLAG                    0x08
#define CLEAR_FLAG                  0x10
#define LOCKOUT_FLAG                0x20
#define REMOTE_FLAG                 0x40
#define DATA_FLAG                   0x80

#define READ_TIME_OUT               10
```

```
#define TCP_MINIMUM_PACKET_SIZE       64

typedef struct
{
        unsigned char  bEOI_Flag;
        unsigned char  reserved[3];
        int            iLength;
} TCP_HEADER;
```



Appendix B: NET_CON Source Code

Parameters and How They Work

In this Appendix, a general explanation of how the instrument's standard parameters are computed (*see below*) is followed by a table listing, defining and describing those parameters (*page C–1*).

Determining Top and Base Lines

Proper determination of the *top* and *base* reference lines is fundamental for ensuring correct parameter calculations. The analysis begins by computing a histogram of the waveform data over the time interval spanned by the left and right time cursors. For example, the histogram of a waveform transitioning in two states will contain two peaks (*Fig. C–1*). The analysis will attempt to identify the two clusters that contain the largest data density. Then the most probable state (centroids) associated with these two clusters will be computed to determine the *top* and *base* reference levels: the *top* line corresponds to the top and the *base* line to the bottom centroid.

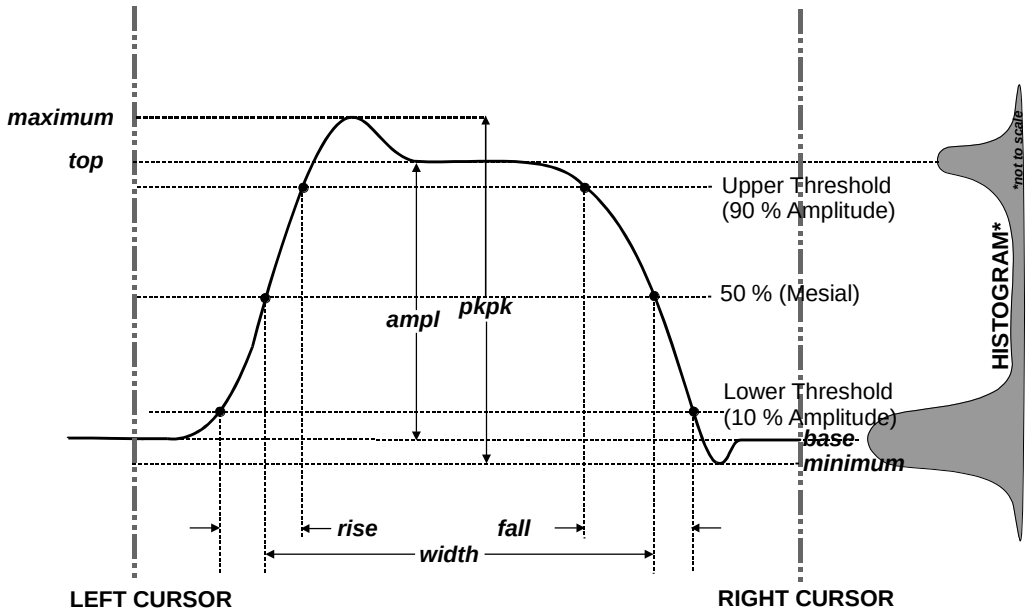


Figure C–1

Determining Rise and Fall Times

Once *top* and *base* are estimated, calculation of the *rise* and *fall* times is easily done (*Fig.1*). The 90 % and 10 % threshold levels are automatically determined by the LSA1000, using the amplitude (*ampl*) parameter.

Threshold levels for *rise* or *fall* time can also be selected using absolute or relative settings (*r@level, fall@level*). If absolute settings are chosen, the *rise* or *fall* time is measured as the time interval separating the two crossing points on a rising or falling edge. But when relative settings are chosen, the vertical interval spanned between the *base* and *top* lines is subdivided into a percentile scale (base = 0 %, top = 100 %) to determine the vertical position of the crossing points.

The time interval separating the points on the rising or falling edges is then estimated to yield the rise or fall time. These results are averaged over the number of transition edges that occur within the observation window.

Rising Edge Duration	$\frac{1}{Mr}\sum_{i=1}^{Mr}\left(Tr_i^{90}-Tr_i^{10}\right)$
Falling Edge Duration	$\frac{1}{Mf}\sum_{i=1}^{Mf}\left(Tf_i^{10}-Tf_i^{90}\right)$
Where *Mr* is the number of leading edges found, *Mf* the number of trailing edges found, Tr_i^X the time when rising edge *i* crosses the x % level, and Tf_i^X the time when falling edge *i* crosses the x % level.	

Determining Time Parameters

Time parameter measurements such as *width*, *period* and *delay* are carried out with respect to the mesial reference level (*Fig. C–2*), located halfway (50 %) between the top and base reference lines.

Time-parameter estimation depends on the number of cycles included within the observation window. If the number of cycles is not an integer, parameter measurements such as *rms* or *mean* will be biased.

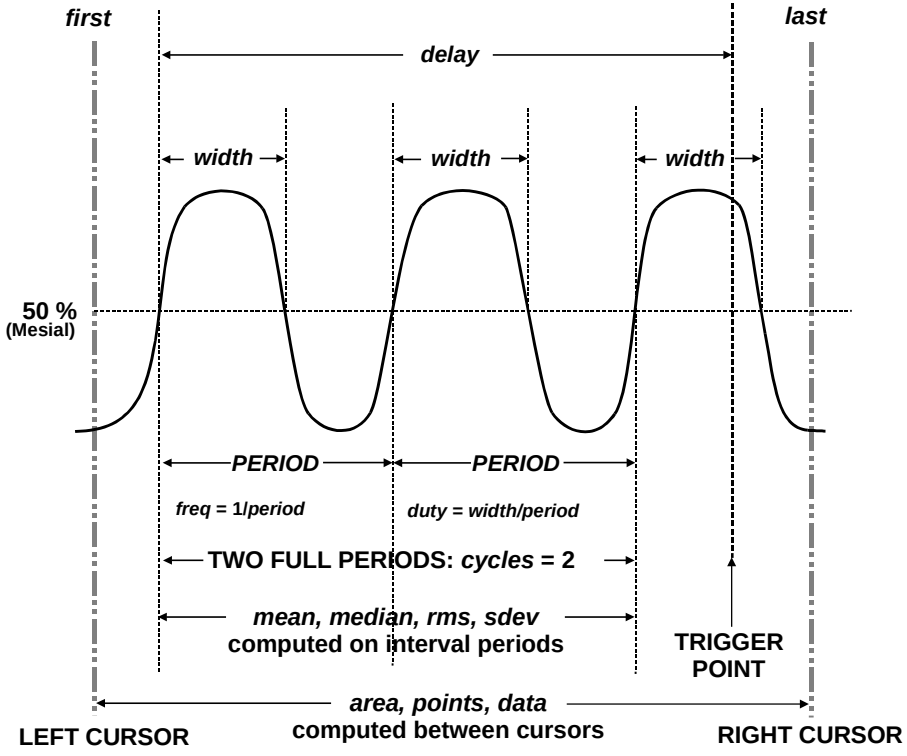


Figure C–2

To avoid these bias effects, the instrument uses cyclic parameters, including *crms* and *cmean*, that restrict the calculation to an integer number of cycles.

Determining Differential Time Measurements

The LSA1000 enables accurate differential time measurements between two traces — for example, propagation, setup and hold delays (*Fig. C–3*).

Parameters such as *Δc2d±* require the transition polarity of the clock and data signals to be specified.

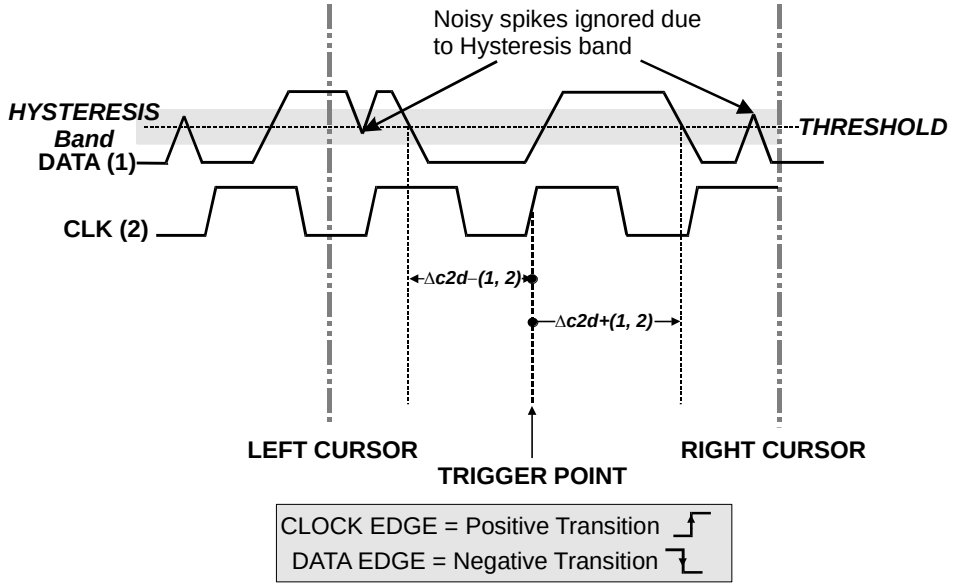


Figure C–3

Moreover, a hysteresis range may be specified to ignore any spurious transition that does not exceed the boundaries of the hysteresis interval. In Figure 3, $\Delta c2d-$ (1, 2) measures the time interval separating the rising edge of the clock (trigger) from the first negative transition of the data signal. Similarly, $\Delta c2d+$ (1, 2) measures the time interval between the trigger and the next transition of the data signal.

Parameter — Description		Definition	Notes
ampl	Amplitude: Measures difference between upper and lower levels in two-level signals. Differs from *pkpk* in that noise, overshoot, undershoot, and ringing do NOT affect measurement.	*top - base* *(See Fig. C–1)*	*On signals NOT having two major levels (such as triangle or saw-tooth waves), returns same value as* pkpk.
area	Integral of data: Computes area of waveform between cursors relative to zero level. Values greater than zero contribute positively to the area; values less than zero negatively.	Sum from *first* to *last* of data multiplied by horizontal time between points *(See Fig. C–2)*	
base	Lower of two most probable states (higher is *top*). Measures lower level in two-level signals. Differs from *min* in that noise, overshoot, undershoot, and ringing do NOT affect measurement.	Value of most probable lower state *(See Fig. C–1)*	*On signals NOT having two major levels (triangle or saw-tooth waves, for example), returns same value as* min.
cycles	Determines number of cycles of a periodic waveform lying between cursors. First cycle begins at first transition after the left cursor. Transition may be positive- or negative-going.	Number of cycles of periodic waveform *(See Fig. C–2)*	
cmean	Cyclic mean: Computes the average of waveform data. Contrary to *mean*, computes average over an integral number of cycles, eliminating bias caused by fractional intervals.	Average of data values of an integral number of periods	
cmedian	Cyclic median: Computes average of base and top values over an integral number of cycles, contrary to *median*, eliminating bias caused by fractional intervals.	Data value for which 50 % of values are above and 50 % below	
crms	Cyclic root mean square: Computes square root of sum of squares of data values divided by number of points. Contrary to *rms*, calculation performed over integral number of cycles, eliminating bias caused by fractional intervals.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i)^2}$	Where: v_i denotes measured sample values, and N = number of data points within the periods found up to maximum of 100 periods.

Parameter — Description		Definition	Notes
csdev	Cyclic standard deviation: Standard deviation of data values from mean value over integral number of periods. Contrary to *sdev*, calculation performed over integral number of cycles, eliminating bias caused by fractional intervals.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i - mean)^2}$	*Where: v_i denotes measured sample values, and N = number of data points within the periods found up to maximum of 100 periods.*
data	Returns average of all data points.	All data values in analyzed region *(See Fig. C–2)*	*Multi-value parameter especially valuable for histograms and trends.*
delay	Time from trigger to transition: Measures time between trigger and first 50 % crossing after left cursor. Can measure propagation delay between two signals by triggering on one and determining delay of other.	Time between trigger and first 50 % crossing after left cursor *(See Fig. C–2)*	
Δdly	Δdelay: Computes time between 50 % level of two sources.	Time between midpoint transition of two sources	
Δt@lv	Δt at level: Computes transition between selected levels or sources.	Time between transition levels of two sources, or from trigger to transition level of a single source	*Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data.*
Δc2d±	Δclock to data ±: Computes difference in time from clock threshold crossing to either the next (*Δc2d+*) or previous (*Δc2d–*) data threshold crossing.	Time from clock threshold crossing to next or previous edge *(See Fig. C–3)*	*Threshold levels of clock and data signals, and edge-transition polarity can be selected. Hysteresis argument used to differentiate peaks from noise in data, with good hysteresis value between half expected peak–to–peak value of signal and twice expected peak–to–peak value of noise.*

<table>
<tr><th>Parameter — Description</th><th>Definition</th><th>Notes</th></tr>
<tr><td>dur</td><td>Duration of acquisition: For single sweep waveforms, dur is 0; for sequence waveforms: time from first to last segment's trigger; for single segments of sequence waveforms: time from previous segment's to current segment's trigger; for waveforms produced by a history function: time from first to last accumulated waveform's trigger.</td><td>Time from first acquisition — for average, histogram or sequence waveforms</td><td></td></tr>
<tr><td>duty</td><td>Duty cycle: Width as percentage of period.</td><td>width/period
(See Fig. C–2)</td><td></td></tr>
<tr><td>f80–20%</td><td>Fall 80–20 %: Duration of pulse waveform's falling transition from 80% to 20%, averaged for all falling transitions between the cursors.</td><td>Average duration of falling 80–20 % transition</td><td>On signals NOT having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.</td></tr>
<tr><td>f@level</td><td>Fall at level: Duration of pulse waveform's falling edges between transition levels.</td><td>Duration of falling edge between transition levels</td><td>On signals NOT having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.</td></tr>
<tr><td>fall</td><td>Fall time: Measures time between two specified values on falling edges of a waveform. Fall times for each edge are averaged to produce final result.
Arguments
<table>
<tr><th>Threshold</th><th>Remote</th><th>Lower Limit</th><th>Upper Limit</th><th>Default</th></tr>
<tr><td>Lower</td><td>low</td><td>1 %</td><td>45 %</td><td>10 %</td></tr>
<tr><td>Upper</td><td>high</td><td>55 %</td><td>99 %</td><td>90 %</td></tr>
</table>
Threshold arguments specify two vertical values on each edge used to compute fall time.
Formulas for upper and lower values:
$\text{lower value} = \text{lower threshold} \times \frac{amp}{100} + base$
$\text{upper value} = \text{upper threshold} \times \frac{amp}{100} + base$</td><td>Time at lower threshold - Time at upper threshold averaged over each falling edge

(See Fig. C–1)</td><td>On signals NOT having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.</td></tr>
</table>

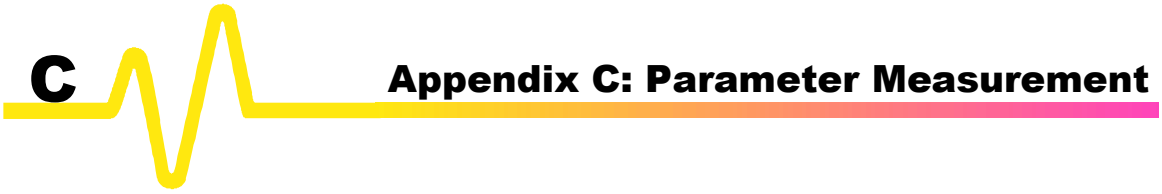
Parameter — Description		Definition	Notes
first	Indicates value of horizontal axis at left cursor.	Horizontal axis value at left cursor *(See Fig. C–2)*	*Indicates location of left cursor. Cursors are interchangeable: for example, the left cursor may be moved to the right of the right cursor and* first *will give the location of the cursor formerly on the right, now on left.*
freq	Frequency: Period of cyclic signal measured as time between every other pair of 50 % crossings. Starting with first transition after left cursor, the period is measured for each transition pair. Values then averaged and reciprocal used to give frequency.	1/*period* *(See Fig. C–2)*	
last	Time from trigger to last (rightmost) cursor.	Time from trigger to last cursor *(See Fig. C–2)*	*Indicates location of right cursor. Cursors are interchangeable: for example, the right cursor may be moved to the left of the left cursor and* first *will give the location of the cursor formerly on the left, now on right.*
maximum	Measures highest point in waveform. Unlike *top*, does NOT assume waveform has two levels.	Highest value in waveform between cursors *(See Fig. C–1)*	*Gives similar result when applied to time domain waveform or histogram of* data *of same waveform. But with histograms, result may include contributions from more than one acquisition. Computes horizontal axis location of rightmost non-zero bin of histogram — not to be confused with* maxp.
mean	Average of *data* for time domain waveform. Computed as centroid of distribution for a histogram. But when input is periodic time domain waveform, computed on an integral number of periods.	Average of *data* *(See Fig. C–2)*	*Gives similar result when applied to time domain waveform or histogram of* data *of same waveform. But with histograms, result may include contributions from more than one acquisition.*

Parameter — Description		Definition	Notes
median	The average of base and top values.	Average of *base* and *top* *(See Fig. C–2)*	
minimum	Measures the lowest point in a waveform. Unlike *base*, does NOT assume waveform has two levels.	Lowest value in waveform between cursors *(See Fig. C–1)*	*Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition.*
over–	Overshoot negative: Amount of overshoot following a falling edge, as percentage of amplitude.	$\frac{(base - minimum)}{ampl} \times 100$ *(See Fig. C–2)*	*Waveform must contain at least one falling edge. On signals NOT having two major levels (triangle or saw-tooth waves, for example), may NOT give predictable results.*
over+	Overshoot positive: Amount of overshoot following a rising edge specified as percentage of amplitude.	$\frac{(maximum - top)}{ampl} \times 100$ *(See Fig. C–1)*	*Waveform must contain at least one rising edge. On signals NOT having two major levels (triangle or saw-tooth waves, for example), may NOT give predictable results.*
period	Period of a cyclic signal measured as time between every other pair of 50 % crossings. Starting with first transition after left cursor, period is measured for each transition pair, with values averaged to give final result.	$\frac{1}{Mr}\sum_{i=1}^{Mr}\left(Tr_i^{50} - Tr_i^{50}\right)$ *(See Fig. C–2)*	*Where:* Mr *is the number of leading edges found,* Mf *the number of trailing edges found,* Tr_i^x *the time when rising edge* i *crosses the x % level, and* Tf_i^x *the time when falling edge* i *crosses the x % level.*
pkpk	Peak–to–peak: Difference between highest and lowest points in waveform. Unlike ampl, does not assume the waveform has two levels.	*maximum - minimum* *(See Fig. C–1)*	*Gives a similar result when applied to time domain waveform or histogram of data of the same waveform. But with histograms, result may include contributions from more than one acquisition.*
phase	Phase difference between signal analyzed and signal used as reference.	Phase difference between signal and reference	
points	Number of points in the waveform between the cursors.	Number of points between cursors *(See Fig. C–2)*	

<table>
<tr><th>Parameter — Description</th><th>Definition</th><th>Notes</th></tr>
<tr><td>r20–80%</td><td>Rise 20 % to 80 %: Duration of pulse waveform's rising transition from 20% to 80%, averaged for all rising transitions between the cursors.</td><td>Average duration of rising 20–80 % transition</td><td>On signals NOT having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.</td></tr>
<tr><td>r@level</td><td>Rise at level: Duration of pulse waveform's rising edges between transition levels.</td><td>Duration of rising edges between transition levels</td><td>On signals NOT having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.</td></tr>
<tr><td>rise</td><td>Rise time: Measures time between two specified values on waveform's rising edge (10–90 %). Rise times for each edge averaged to give final result.
Arguments<table><tr><th>Threshold</th><th>Remote</th><th>Lower Limit</th><th>Upper Limit</th><th>Default</th></tr><tr><td>Lower</td><td>low</td><td>1 %</td><td>45 %</td><td>10 %</td></tr><tr><td>Upper</td><td>high</td><td>55 %</td><td>99 %</td><td>90 %</td></tr></table>Threshold arguments specify two vertical values on each edge used to compute rise time.
Formulas for upper and lower values:
$\text{lower value} = \text{lower threshold} \times \frac{amp}{100} + base$
$\text{upper value} = \text{upper threshold} \times \frac{amp}{100} + base$</td><td>Time at upper threshold - Time at lower threshold averaged over each rising edge

(See Fig. C–1)</td><td>On signals NOT having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.</td></tr>
</table>

Parameter — Description		Definition	Notes
rms	Root Mean Square of data between the cursors — about same as *sdev* for a zero-mean waveform.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i)^2}$ *(See Fig. C–2)*	*Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Where: v_i denotes measured sample values, and N = number of data points within the periods found up to maximum of 100 periods.*
sdev	Standard deviation of the data between the cursors — about the same as *rms* for a zero-mean waveform.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i - mean)^2}$ *(See Fig. C–2)*	*Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Where: v_i denotes measured sample values, and N = number of data points within the periods found up to maximum of 100 periods.*
t@level	Time at level: Time from trigger (t=0) to crossing at a specified level.	Time from trigger to crossing level	
top	Higher of two most probable states, the lower being *base*. This is characteristic of rectangular waveforms and represents the higher most probable state determined from the statistical distribution of *data* point values in the waveform.	Value of most probable higher state *(See Fig. C–1)*	*Gives similar result when applied to time domain waveform or histogram of* data *of same waveform. But with histograms, result may include contributions from more than one acquisition.*
width	Width of cyclic signal determined by examining 50 % crossings in data input. If first transmission after left cursor is a rising edge, waveform is considered to consist of positive pulses and *width* the time between adjacent rising and falling edges. Conversely, if falling edge, pulses are considered negative and *width* the time between adjacent falling and rising edges. For both cases, widths of all waveform pulses averaged for final result.	Width of first positive or negative pulse averaged for all similar pulses *(See Figs. 1, 2)*	*Similar to* fwhm*, which, however, applies only to histograms.*



Appendix C: Parameter Measurement

Using the LSA1000 Software Tools

The application program examples contained in this appendix demonstrate the practical use by remote control of the LSA1000 software tools in real-life applications. *Refer to the accompanying Remote Control Manual for the commands.*

Note: Memory allocation and error handling are not shown.

Example 1

Digitized Data Transfer from LSA1000 to PC: In many cases, the LSA1000 may be used solely as a high speed digitizer, for transferring large amount of digitized data to the host PC as fast as possible. The following is an example program for transferring 2 MB of digitized data from LSA1000 to PC with emphasis on speed of transfer.

Requirement

Sample Rate = 1.5GS/s
Capture Time = 50us
Number of Points = 75,000

LSA1000 Settings

Sample Rate = 2GS/s
Time per Div. = 5us
Memory Size = 100K

Program Code

```
IM_MakeConnection("IP:172.25.1.2");

//Initialize the unit
    IM_WriteDevice("CHDR OFF");
    IM_WriteDevice("TDIV 5us");
    IM_WriteDevice("MSIZ 100K");
    IM_WriteDevice("COMB 2");
    IM_WriteDevice("COMS C1");
```

```
IM_WriteDevice("WFSU NP, 75000, FP, 0");//transfer
only 75k points
IM_WriteDevice("CFMT OFF, BYTE, BIN");
IM_WriteDevice("TRLV 0");
IM_WriteDevice("TRSL POS");
IM_WriteDevice("TRSE EDGE, SR, C1");


IM_QueryDevice("C1:INSP? 'PNTS_PER_SCREEN'",
strBuf, sizeof(strBuf));
idxPtr = strchr(rspStr, ':');
sscanf(idxPtr+1, "%d", &BufferSize);


Loop
{
        IM_WriteDevice("ARM; WAIT; C1:WF? DAT1");
        IM_ReadDevice(Buffer, BufferSize);
}

        IM_Disconnect();
```

Example 2

Using Processing Power on LSA1000 Pulse Parameters: While Example 1 shows the LSA1000 as a "pure" high speed digitizer, the on-board processing capability of the LSA1000 can also be used. In fact, the LSA1000 has the same math functions and processing capability as a high-end LeCroy digital oscilloscope. This Example measures pulse widths of the waveforms that were captured by the LSA1000.

Requirement

Sample Rate = 1.5GS/s
Capture Time = 50us
Number of Points = 75,000

LSA1000 Settings

Sample Rate = 2GS/s
Time per Div. = 5us
Memory Size = 100K

Program Code

```
        IM_MakeConnection("IP:172.25.1.2");

// Initialize the unit
        IM_WriteDevice("CHDR OFF");
        IM_WriteDevice("TDIV 5us");
        IM_WriteDevice("MSIZ 100K");
        IM_WriteDevice("COMB 2");
        IM_WriteDevice("COMS C1");
        IM_WriteDevice("WFSU NP, 75000, FP, 0");
        IM_WriteDevice("CFMT OFF, BYTE, BIN");
        IM_WriteDevice("TRLV 0");
        IM_WriteDevice("TRSL POS");
        IM_WriteDevice("TRSE EDGE, SR, C1");

Loop
{
        IM_QueryDevice("ARM; WAIT; C1:PAVA? WID", strBuf,
        sizeof(strBuf));
}
        IM_Disconnect();
```

Example 3

Using Processing Power for FFT: This commands the LSA1000 to perform FFT (Fast Fourier Transfer) on the captured waveform and transfer FFT result to the host PC.

Requirement

Sample Rate = 1.5GS/s
Capture Time = 50us
Number of Points = 75,000

LSA1000 Settings

Sample Rate = 2GS/s
Time per Div. = 5us
Memory Size = 100K

Program Code

```
        IM_MakeConnection("IP:172.25.1.2");

// Initialize the unit
        IM_WriteDevice("CHDR OFF");
        IM_WriteDevice("TDIV 5us");
        IM_WriteDevice("MSIZ 100K");
        IM_WriteDevice("COMB 2");
        IM_WriteDevice("COMS C1");
        IM_WriteDevice("WFSU NP, 75000, FP, 0");
        IM_WriteDevice("CFMT OFF, BYTE, BIN");
        IM_WriteDevice("TRLV 0");
        IM_WriteDevice("TRSL POS");
        IM_WriteDevice("TRSE EDGE, SR, C1");


// define the math function
        IM_WriteDevice("TA:TRA ON");
        IM_WriteDevice("TA:DEF EQN, 'PS(FFT(C1))',
        MAXPTS, 2500, WINDOW, RECT, DCSUP, OFF");

Loop
{
        IM_WriteDevice("ARM; WAIT");

        // loop until math processing completes
        Loop
        {
                IM_QueryDevice("INR?", strBuf, sizeof(strBuf));
                If (strBuf indicates a flag is set)
                        Break;
        }
        IM_QueryDevice("TA:WF? DAT1", Buffer, BufferSize);
}
        IM_Disconnect();
```

Example 4

Using Processing Power for Averaging: This commands the LSA1000 to perform Averaging on the captured waveform and transfer averaging result to the host PC.

Requirement

Sample Rate = 1.5GS/s
Capture Time = 50us
Number of Points = 75,000

LSA1000 Settings

Sample Rate = 2GS/s
Time per Div. = 5us
Memory Size = 100K

Program Code

```
IM_MakeConnection("IP:172.25.1.2");

// Initialize the unit
IM_WriteDevice("CHDR OFF");
IM_WriteDevice("TDIV 5us");
IM_WriteDevice("MSIZ 100K");
IM_WriteDevice("COMB 2");
IM_WriteDevice("COMS C1");
IM_WriteDevice("WFSU NP, 75000, FP, 0");
IM_WriteDevice("CFMT OFF, BYTE, BIN");
IM_WriteDevice("TRLV 0");
IM_WriteDevice("TRSL POS");
IM_WriteDevice("TRSE EDGE, SR, C1");


// define the math function
IM_WriteDevice("TA:TRA ON; TB:TRA ON");
IM_WriteDevice("TA:DEF EQN, 'AVGS(C1)',  SWEEPS,
25
IM_WriteDevice("TB:DEF EQN, 'PS(FFT(TA))',
MAXPTS, 2500, WINDOW, RECT, DCSUP, OFF");

Loop
{
    // loop until math processing completes on trace A
    Loop
    {
        IM_WriteDevice("ARM; WAIT");
```

```
                IM_QueryDevice("INR?", strBuf, sizeof(strBuf));
                If (strBuf indicates a flag is set)
                        Break;
        }
        // loop until math processing completes on trace B
        Loop
        {
                IM_QueryDevice("INR?", strBuf, sizeof(strBuf));
                If (strBuf indicates a flag is set)
                        Break;
        }
        IM_QueryDevice("TB:WF? DAT1", Buffer, BufferSize);
}
        IM_Disconnect();
```



Appendix D: Program Examples

What Do *You* Think About the LSA1000*?*

These suggestion forms covering the LSA1000 and its manuals are a means by which you can communicate to us what you like, and what you think can be improved.

LeCroy welcomes your suggestions on product improvement or ideas for new products.

Name:

Title:

Company:

Address:

Phone: **Fax:**

Email:

Product(s) used:

Problem (s), if any:

Suggestions:

Mail to: LeCroy Corporation
Embedded Signal Analysis
700 Chestnut Ridge Road
Chestnut Ridge, NY 10977
USA

Fax to: Embedded Signal Analysis
++1 (914) 578 - 5985

LeCroy welcomes your comments on:
The LSA1000 *Signalyst Operator's Manual Rev A*, and *Remote Control Manual Rev A*.

Name:

Title:

Company:

Address:

Phone: **Fax:**

Email:

How would you like the manuals to be changed?

Did you find any errors in the manuals? If so, please record them below, including the name of the manual and page number(s) concerned.

Mail to: LeCroy Corporation
Embedded Signal Analysis
700 Chestnut Ridge Road
Chestnut Ridge, NY 10977
USA

Fax to: Embedded Signal Analysis
++1 (914) 578 - 5985



Appendix E: Suggestion Forms



Index

Index

A

Acquisition Memory, 2-3, 2-4, A-1
  increasing it by combining channels, 2-1
Acquisition Modes, A-1
Acquisition Status, 2-12
Acquisition Summary, 2-8
ActiveDSO, 5-1, 5-3, 5-13
ActiveX, 2-11, 5-13, 5-18
ADC (Analog–to–Digital Converter), 2-1, 2-3, 2-5
All, 5-10
Altitude, A-5
Amplifier, 2-3
Amplitude, C-5
API, 5-21
Area, C-5
Auto, 5-6
AUTO, 2-8
Auto-Calibration, A-5
Average, D-5

B

Bandwidth, 2-4, A-1
Bandwidth Limiter, A-1
Base, C-1, C-2, C-5
Block Diagram, 2-2

C

Cables, 4-1
Calibration, A-5
Centroids, C-1
Channel, 5-7, 5-11
Channels, 2-6, 2-12, A-1
Cleaning and Maintenance, 3-3
Clear, 5-10
Clock, A-2
Clock Edge, C-4
Combining Channels, 2-1
Command Panel, 5-10
Conformity, 3-1, A-5
Connections, 4-1
Cursors, A-4
Cycles, C-5
  in parameter measurements, C-2
Cyclic Mean, C-5
Cyclic Median, C-5
Cyclic Parameters, C-3
Cyclic Root Mean Square, C-5
Cyclic Standard Deviation, C-6

D

Data, C-6
Data density, C-1
Data Transfer
  for speed, D-1
DC Accuracy, A-1
Delay, 5-6, C-6
DELAY, 2-8
Differential Time Measurements, C-3
Digitizers, A-1
Display + Utility, 5-9
Dual Screen, 5-9
Duration, C-7
Duty, C-7

E

Ethernet, 2-1, 2-4, 2-11
  Address, 4-1, 4-2
  connecting to, 4-1
  connecting via, 4-1, 4-4
Ethernet Port, 2-13
External Reference, A-2
External Reference Clock In, 2-12
External Reference Clock Out, 2-12
External Trigger In, 2-12

F

Fall, C-2
Fall 80–20 %, C-7
Fall at Level, C-7
Fall time, C-2, C-7
FFT (Fast Fourier Transform), 2-10, D-3

Find, 5-7
First, C-8
Frequency, C-8
Frequency Domain, 2-10
Frequency Range, A-1
Front Panel, 2-12
Fuses, 3-3

G

Grid, 5-9

H

H Mag, 5-8
Histograms, 2-10, A-4, C-1
Humidity, 3-1
Hysteresis, C-4

I

Input Coupling, A-1
Interfacing, A-5
IP Address, 4-1, 4-2, 4-3, 4-4, 5-16, 5-20, 5-21
  resetting it, 4-7

L

Last, C-8
Leading Edge, C-2
LEVEL, 2-8

M

Maintenance, 1-2
Maximum, C-8
Maximum Input, A-1
Maximum Sample Rate, A-1
Mean, C-8
Median, C-8
Memories, 2-1, 2-3
Memory Length, 2-4
Mesial, C-2
Minimum, C-9

N

NET_CON, 5-2, 5-21, B-1
Network
  connection, 4-6
Norm, 5-6
NORM, 2-8

O

Offset (V), 5-7
Offset Range, A-1
OLE, 5-13
Operating Environment, 3-1, A-5
Over-, C-9
Over +, C-9
Overload, 3-3
Overvoltage, 3-1

P

Packing and Shipment, 1-3
Parameters, C-1, C-5, C-6, C-7, C-8, C-9, C-10, C-11
PC
  configuring it, 4-2
  connecting to, 4-1, 4-4
  operational requirements, 4-1
Peak–to–Peak, C-9
Period, C-9
Persistence, 5-9
Phase, C-9
Points, C-9
Pollution Degree, 3-1
Position scroll bar, 5-8
Post-Trigger, 2-8
Power, 3-3
Power On/Off, 2-12, 2-13
Pre-Trigger, 2-8
Processing Functions, A-4
Processor, 2-3
Processors, 2-1
Pulse Parameters, 2-9

R

RAM, 2-3
  amount needed for
    LSA1000, 4-1
Rear Panel, 2-12
Remote LSA, 2-11, 5-1, 5-3, 5-5, 5-11
Reply, 5-10
Reset, 5-8
Return, 1-3
Rise, C-2
Rise 20–80 %, C-10
Rise at Level, C-10
Rise time, C-2, C-10
Root Mean Square (rms), C-2, C-11

S

Safety, 3-1
Sample rate
  increasing it by combining
    channels, 2-1
Sample Rate, 2-4
Sample size, 5-6
Scale Factors, A-1
ScopeExplorer, 4-7, 5-1, 5-3, 5-19
Select, 5-7
Sensitivity, A-1
Service and Repair, 1-2
Set Math, 5-8
Single-Shot Acquisition, 2-5
Single-Shot Mode, 2-8
Size, A-5
Slope, 5-6
Sngl, 5-6
SNGL, 2-8
Software Tools
  examples of their use, D-1
Source, 5-6
Source Code
  NET_CON, B-1
Standard Deviation, C-11
Standard Parameters, C-1
Statistical Domain, 2-10
Stop, 5-6
System Memory, 2-3

T

TCP, 4-2, 4-3, 4-4, 5-21
TCP/IP, 2-11, 5-3, 5-16, 5-21
Temperature, 3-1
  LSA1000 Series, A-5
Time at Level, C-11
Time Domain, 2-10
Time intervals, C-2
Time Parameter measurements, C-2
TIME/DIV, 2-8
Time/Div., 5-6
Timebase + Trigger, 5-6, 5-11
Top, C-1, C-2, C-11
Trace On/Off, 5-7, 5-8
Trailing Edge, C-2
Trending, A-4
Trigger, 2-6
  Amplitude, 2-7
  Coupling, 2-6
  Level, 2-6, 2-8
  Range, 2-7
  Slope, 2-6
  Source, 2-6, 2-8
Trigger Coupling, A-3
Trigger Delay, 2-8, A-3
Trigger Maximum Input, A-3
Trigger Modes, 2-8
Trigger Range, A-3
Trigger Slope, A-3
Triggering, A-3

U

UBS Port, 2-13
Utility, 5-9

V

V Mag, 5-8
Vertical Resolution, A-1
VICP (Versatile Instrument Control Protocol, 2-11

W

Warnings, 3-2
Warranty, 1-1, A-5
Waveform Math, 2-9
Waveform Processing
  Averaging, D-5
  FFT, D-3
  pulse parameters, D-2
Weight, A-5
Width, C-11
Windows, 4-1, 4-4, 5-3, 5-21
WinSock, 5-21

Z

Zoom + Math, 5-8



Index

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP