



# **CompuScope Software Development Kit (SDK) for LabVIEW for Windows**

## **User's Guide**

**CompuScope Driver Version 4.10+**

**For LabVIEW 6.1+**

**P/N: 0045022**

**Reorder #: MKT-SWM-CSLV10**

**1210**

© Copyright Gage Applied Technologies 2005, 2006, 2007, 2008, 2009.2010

Tel: 1-800-567-GAGE or (514) 633-7447. Fax: 1-800-780-8411 or (514) 633-0770

COMPUSCOPE, GAGESCOPE, AND COMPUGEN are trademarks or registered trademarks of Gage Applied Technologies.

Microsoft Windows is a trademark or registered trademark of Microsoft Corporation.

LabVIEW is a trademark or registered trademark of National Instruments Corp.

Changes are periodically made to the information herein; these changes will be incorporated into new editions of the publication. Gage Applied Technologies may make improvements and/or changes in the products described in this publication at any time.

Copyright © 2005, 2006, 2007, 2008, 2009 Gage Applied Technologies. All Rights Reserved, including those to reproduce this publication or parts thereof in any form without permission in writing from Gage Applied Technologies.

***How to reach GaGe Product Support***

Toll-free phone: (800) 567-4243

Toll-free fax: (800) 780-8411

***To reach GaGe Product Support from outside North America***

Tel: +1-514-633-7447

Fax: +1-514-633-0770

**E-mail:** [prodinfo@gage-applied.com](mailto:prodinfo@gage-applied.com)

**Web site:** <http://www.gage-applied.com>

**On-line Support Request Form:** [www.gage-applied.com/support/support\\_form.php](http://www.gage-applied.com/support/support_form.php)

# Table of Contents

<b>PREFACE</b> .....	<b>4</b>
<b>CHAPTER 1: INSTALLATION OF COMPUSCOPE LABVIEW SDK</b> .....	<b>5</b>
<b>CHAPTER 2: OVERVIEW OF COMPUSCOPE LABVIEW SDK</b> .....	<b>6</b>
OVERVIEW OF COMPUSCOPE MAIN VIS.....	7
OVERVIEW OF COMPUSCOPE CsTOOL VIS.....	9
OVERVIEW OF COMPUSCOPE CSLV DLL VIS.....	10
<b>CHAPTER 3: DETAILED DESCRIPTION OF COMPUSCOPE MAIN VIS</b> .....	<b>11</b>
<i>GageSimple.vi</i> .....	11
<i>GageAcquire.vi</i> .....	12
<i>GageCoerce.vi</i> .....	16
<i>GageMultipleRecord.vi</i> .....	18
<i>GageDeepAcquisition.vi</i> .....	20
<i>GageComplexTriggerAcquire.vi</i> .....	22
<i>GageTwoIndependentSystemsAcquire.vi</i> .....	23
<i>GageDigitalCapture.vi</i> .....	25
<i>Advanced Main VIs</i> .....	26
<b>CHAPTER 4: SPECIAL TOPICS</b> .....	<b>27</b>
<i>Converting from CompuScope ADC Code to Voltages</i> .....	27
<i>Depth and Segment Size</i> .....	27
<i>Trigger Holdoff</i> .....	28
<i>Trigger Delay</i> .....	28
<i>CompuScope Acquisition Timing Diagram</i> .....	29
<i>Representative Acquisition Sequences</i> .....	30
<b>TECHNICAL SUPPORT</b> .....	<b>31</b>
<b>GAGE PRODUCTS</b> .....	<b>32</b>

## Preface

This manual is meant to serve as an aid to engineers using the CompuScope series of high-performance data acquisition cards in the LabVIEW 6.1+ for Windows environment.

Throughout this manual, it is assumed that you are familiar with the LabVIEW graphical programming environment. If you do not feel comfortable with LabVIEW, it is highly recommended that you go through the **Getting Started with LabVIEW for Windows** manual supplied to you by National Instruments, before starting any program development for the CompuScope cards.

It is also assumed that you have correctly installed the CompuScope Windows drivers.

The CompuScope LabVIEW SDK supports all PC-based GaGe CompuScope cards – PCI and CompactPCI/PXI. Specific hardware features that are available from the SDK VIs, however, may not be supported by your CompuScope hardware. Please refer to the CompuScope Hardware manual for information specific to your CompuScope card in order to determine the capabilities of your CompuScope.

It is also assumed that you are familiar with PCs and Microsoft Windows. This manual will use the terms “CompuScope SDK for LabVIEW for Windows” and “CompuScope LabVIEW SDK” interchangeably.

## **Chapter 1: Installation of CompuScope LabVIEW SDK**

If you purchased the CompuScope LabVIEW SDK you will have been shipped a software key that allows installation of the SDK from the GaGe CompuScope CD. Simply select the installation of the CompuScope LabVIEW SDK from the CompuScope CD and enter the software key when prompted.

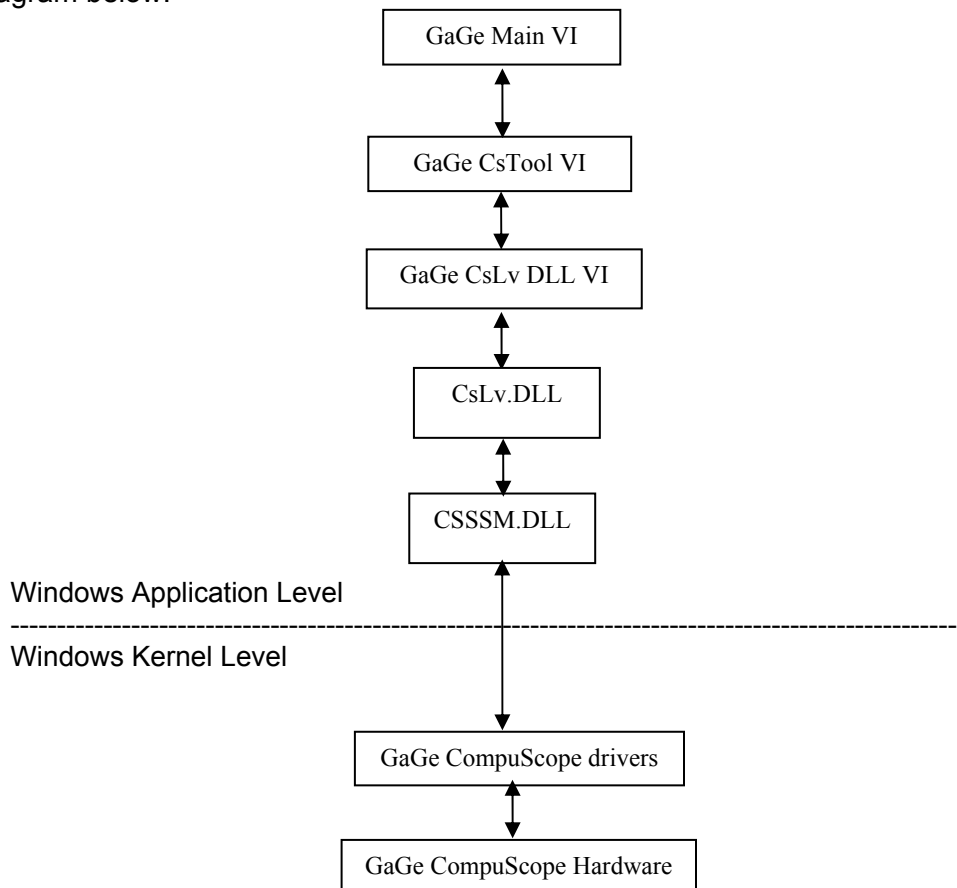
By default, the CompuScope LabVIEW SDK will install itself in:  
O/S system drive:\Program Files\Gage\CompuScope\CompuScope LabVIEW SDK. It is recommended that you use the default installation location.

If you require more detailed installation instructions, please refer to the GaGe CompuScope Startup Guide, which was shipped with your order.

## Chapter 2: Overview of CompuScope LabVIEW SDK

### Structure of CompuScope LabVIEW SDK

The overall structure of the CompuScope LabVIEW SDK and its relation to the GaGe CompuScope Hardware is best described from the bottom up with reference to the diagram below.



At the lowest level is the CompuScope hardware, which is installed within a slot that is connected to the host PC. The CompuScope hardware is directly controlled by the CompuScope Windows drivers. The drivers reside at the Windows Kernel level, which allows direct low-level access to CompuScope hardware registers and to physical PC RAM. The drivers communicate with Windows Applications through a Dynamically Linked Library (DLL) called C\$SSM.DLL.

Communications through C\$SSM.DLL uses the CompuScope Applications Programming Interface (API), which is a set of C subroutine calls that allows control of all CompuScope functionality. Above C\$SSM.DLL is a small intermediate DLL called CsLV.DLL that packages the API calls so that they may be easily called from LabVIEW. Above CsLV.DLL sits three layers of LabVIEW VIs, called *CsLv DLL VIs*, *CsTool VIs* and *Main VIs*, that will be completely described in this manual.

The Main VIs are top-level application VIs that can be executed directly under LabVIEW.

Each Main VI illustrates usage of the CompuScope hardware in different operating modes. The Main VIs are intended as convenient starting points for CompuScope users to develop into their own LabVIEW applications. Consequently, the controls and indicators of the Main VIs are not wired to the VI Icons and the Main VIs should not be called as sub-VIs by other VIs.

The CsTool VIs are an easy-to-use complete set of sub-VIs for calling the GaGe hardware. All Main VIs are constructed using only CsTool VIs. While they may not provide the best possible performance, CsTool VIs are extremely easy to understand and use.

The CsLv DLL VIs are simple wrapper VIs that make direct Library calls to CsLV.DLL, which is the link from LabVIEW to the CompuScope drivers. All CsTool VIs are constructed from CsLv DLL VIs. Direct usage of the CsLv DLL VIs is intended only for advanced LabVIEW users who wish to improve upon the hardware performance that is possible using the CsTool VIs.

## **CompuScope Systems**

A CompuScope system is defined as a single CompuScope digitizer board or a group of CompuScope boards configured as Master/Slave CompuScope system. A Master/Slave CompuScope system samples and triggers simultaneously on all channels and is considered to be one multi-channel CompuScope system. For instance, a Master/Slave system composed of four CompuScope two-channel boards will be considered as a single CompuScope system with eight available input channels.

By structuring the drivers to consider CompuScope systems, a single PC can be equipped with almost any imaginable combinations of CompuScope hardware. For instance, a PC could be equipped with two separate Master/Slave systems of four channels each and then an additional single CompuScope system for a total of three CompuScope systems.

CompuScope systems are addressed from LabVIEW by first acquiring a *Handle* for the system. After usage of the system is complete, the user must release the Handle so that it is free for usage by other processes. By obtaining handles for different systems, a single LabVIEW VI may simultaneously operate different CompuScope systems. Alternately, separate LabVIEW VIs may operate independently and simultaneously by calling handles for separate CompuScope systems. Different VIs may even access the same hardware as long as one VI frees the system handle before the other VI obtains it, since different applications may not simultaneously access the same CompuScope system.

While most Main example VIs access only a single CompuScope system, understanding the CompuScope system structure easily allows users to extend these Main VIs to multiple system operation.

## **Overview of CompuScope Main VIs**

The Main VIs are intended to be convenient starting points around which users can develop customized LabVIEW software for their digitizer application. Users can construct

more complex VIs themselves by using the existing VIs as a guide for combining their functionalities.

The basic algorithm for all of the Main VIs is the following:

1. *Initialize the CompuScope driver and the CompuScope hardware.*
2. *Obtain the handle to the first available CompuScope system in the PC.*
3. *Pass the desired configuration settings to the driver using CsTool-Configure VIs*
4. *Pass the configured settings to the CompuScope hardware using the CsTool-Commit VI.*
5. *Start the CompuScope hardware to digitize data into its on-board memory and await a trigger event.*
6. *Continuously check to see if the CompuScope hardware has finished capturing the current record.*
7. *Download the data of interest from the CompuScope hardware to LabVIEW array variables.*
8. *Plot the data.*

A list of all CompuScope Main VIs is given below with a brief description of the functionalities.

GageSimple.vi: A simple VI used to verify correct CompuScope, driver and LabVIEW SDK operation. No adjustments are possible and default CompuScope settings are used.

GageAcquire.vi: The simplest VI with full configuration control of CompuScope settings that illustrates usage of CompuScope Single Record Mode. An error occurs upon entry of invalid settings.

GageCoerce.vi: A VI that is just like GageAcquire.vi, except that invalid settings are coerced to the closest valid values. A panel appears that indicates the actual coerced settings.

GageMultipleRecord.vi: A VI that acquires and displays captured records when operating a CompuScope system in Multiple Record Mode.

GageDeepAcquisition.vi: A VI that acquires and manages large data captures from CompuScope Hardware (> 16 MB).

GageComplexTriggerAcquire.vi: A VI that illustrates usage of complex triggering using multiple on-board trigger engines, if available.

GageTwoIndependentSystemsAcquire.vi: A VI that operates two independent CompuScope systems, each with their own setting controls and display charts.

GageDigitalCapture.vi: A VI that is designed to capture and display data from digital input CompuScope cards, like the CompuScope 3200.



Not all possible CompuScope operation configurations are covered by the Main VIs. For instance, there is a VI that handles deep acquisitions and a VI that handles acquisitions from two CompuScope systems. However, there is not a VI that handles deep acquisitions from two CompuScope systems. For both these functionalities, the user must study GageDeepAcquisition.vi and GageTwoIndependentSystemsAcquire.vi and then appropriately combine them to create a VI that meets the requirement.

All Main VIs should always be stopped using the STOP button provided. Stopping an SDK in another fashion will not correctly free CompuScope handles and other processes will be unable to use them. Such locked handles can be freed by invoking the CsTool-FreeSystem or CsLv\_FreeSystem VIs if the handle value is known. If the locked handle value is not known, CsLv\_FreeAllSystems.vi can be used to free all CompuScope systems. Finally, locked handles may be freed from outside of LabVIEW by doing a “Refresh” from the CompuScope Manager utility after you have closed down LabVIEW.

## **Overview of CompuScope CsTool VIs**

The CsTool VIs are easy-to-use building blocks from which the Main VIs are built. A LabVIEW user is able to develop a custom LabVIEW VI for their requirement using only the CsTool VIs, with no modifications to them. All CsTool VIs are extensively documented in the LabVIEW Context Help. For convenience, a listing of all the CsTool descriptions is also given in a file called CsTool-VI\_Guide.pdf that is installed with the LabVIEW SDK.

Most CsTool VIs, require a Handle Cluster input, which identifies the CompuScope system on which the CsTool VI will operate. The Handle Cluster contains static information about the CompuScope system. All CsTool VIs return error status values, which may be translated into descriptive error message strings using CsTool-ErrorHandler.vi.

Where appropriate, CsTool VIs use descriptive string variables rather than index values for VI readability and easy programming. (By contrast, lower level CsLv VIs generally use indices,) Note that valid string values are not case-sensitive, but must the exact spelling must be used. For example, “8-Bit”, “8-BIT”, or “8-bIT” are acceptable as Sample Bits inputs, however, “8 bit” is not.

CsTool VIs are divided into six different types. Each CsTool type is associated with its own distinctive LabVIEW icon. The six CsTool VI types listed below



#### Contact VIs

These CsTool VIs are used to establish or break communication with a specific CompuScope system.



#### Get VIs

These CsTool VIs are used to obtain capability information about the specified CompuScope system.



#### Configuration VIs

These CsTool VIs are used to adjust configuration settings in the drivers for the specified CompuScope system. The settings are not actually passed to the CompuScope System until a call to CsTool-Commit.vi is made.



#### Queries VIs

These CsTool VIs are used to obtain the actual configuration settings on the specified CompuScope system.



#### Action VIs

These CsTool VIs are used perform actions on the specified CompuScope system, such as loading settings, starting acquisitions and transferring data.



#### Error handler VI

This VI is used to decode error number codes and display descriptive error messages.

## **Overview of CompuScope CsLv DLL VIs**

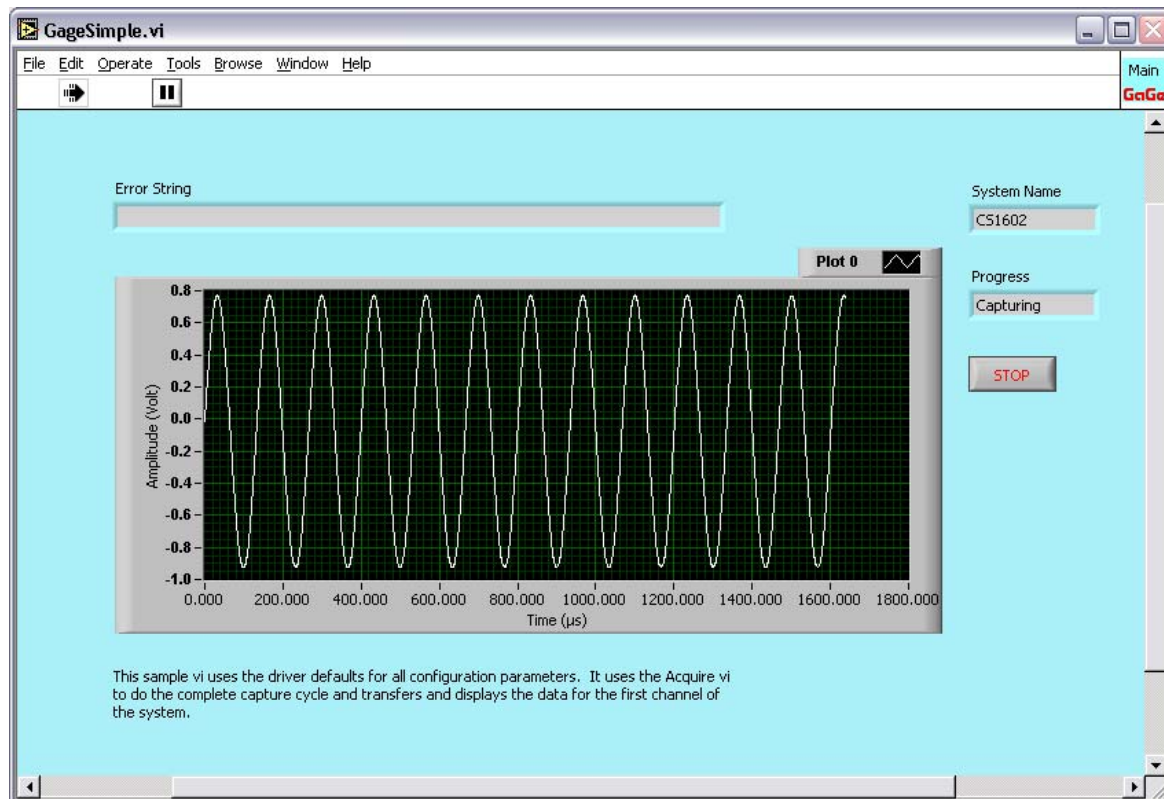
The CsLv DLL VIs are the lowest level of SDK VIs. They are simple wrapper VIs that make LabVIEW Library calls to an intermediate DLL called CsLv.DLL. This DLL simply translates calls into CompuScope C API calls that are passed along to CSSSM.DLL – the CompuScope drivers DLL.

All CsLv DLL VIs are extensively documented in the LabVIEW Context Help. For convenience, a listing of all the CsLv DLL VI descriptions is also given in a file called CsLv-VI\_Guide.pdf that is installed with the LabVIEW SDK.

Direct usage of CsLv DLL VIs is intended only for the advanced LabVIEW user. Advanced users can develop or by-pass CsTool VIs or directly use CsLv DLL VIs in order to achieve the best possible hardware performance under LabVIEW. For instance, an advanced user might pull out the CsLv DLL data transfer VI from the CsTool-Transfer VI and forego conversion of the raw data into Volts in order to achieve the best repetitive capture performance.

## Chapter 3: Detailed Description of CompuScope *Main VIs*

### *GageSimple.vi:*



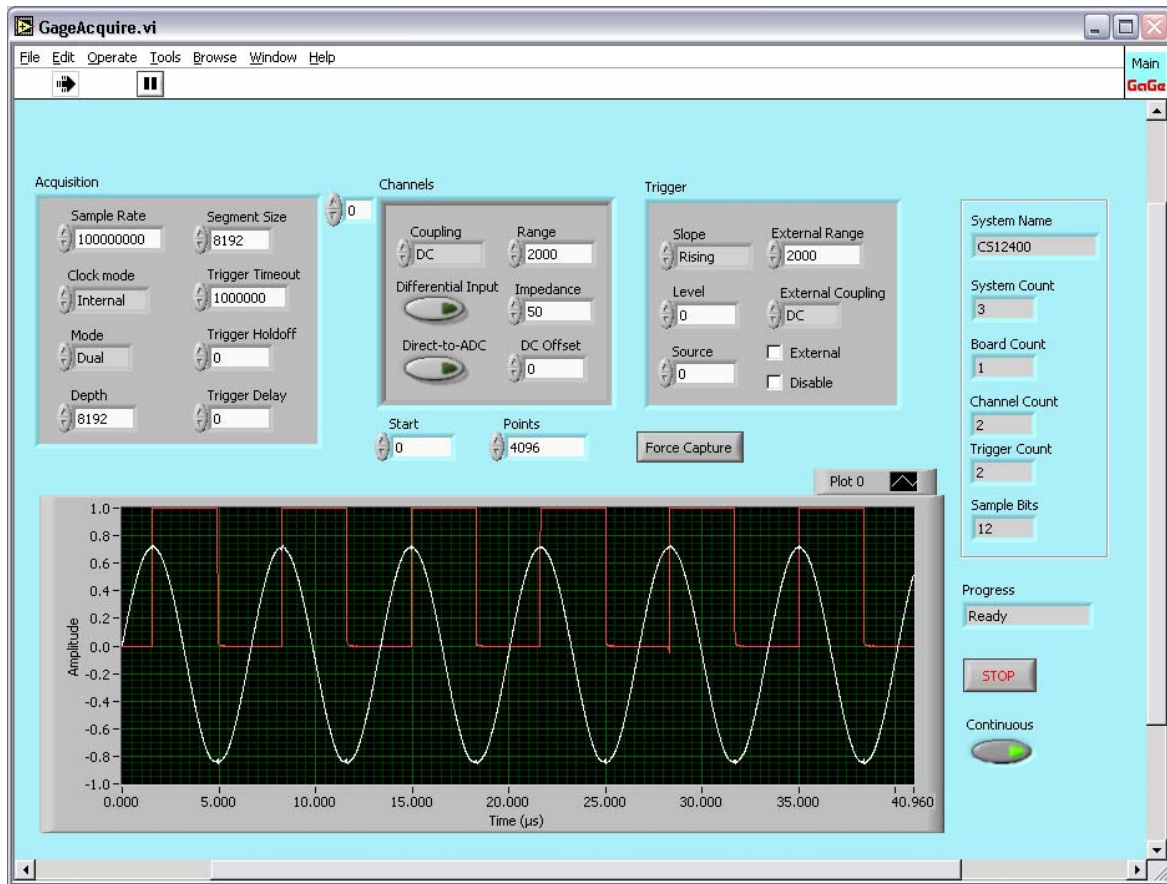
GageSimple.vi is not intended as a starting point for customer applications but is a simple test program to confirm correct operation of the CompuScope hardware, Windows drivers, and LabVIEW SDK.

First, GageSimple.vi grabs the handle to the first available CompuScope system. In the event of a Master/Slave CompuScope system, however, GageSimple.vi only operates upon the first CompuScope card within that system. Using this CompuScope card, GageSimple.vi then does repetitive acquisitions using the driver defaults for all configuration parameters. The driver defaults are a set of default configuration parameters that are tailored to each CompuScope hardware model. The System Name indicator displays the model of the CompuScope hardware in the system.

This VI operates only on a single CompuScope card. In the event of a multi-CompuScope system, the VI only addresses the first card in the first system. Except for a STOP button, the VI has no front-panel controls. The VI will repetitively display the acquired waveform from Channel 1 until the STOP button is pressed.

Again, GageSimple.vi should only be used to test hardware and software integrity and should not be developed by the customer into a custom application VI, since there are better Main VIs for this purpose.

## GageAcquire.vi



GageAcquire.vi is the Main VI for Single Record capture from a single CompuScope system. In the event of Multiple CompuScope systems, the VI only operates the first CompuScope system in the PC.

The VI accepts configuration settings from the configuration controls and uses them to operate the CompuScope system. Acquisition operations occur sequentially in a LabVIEW sequence. Data are displayed in the large display plot. The VI can be configured to operate continuously by pressing the Continuous button. Press the STOP button to terminate the VI. The Progress indicator shows the current state of the CompuScope system.

Waveform data are displayed as Amplitude in Volts against Time in microseconds. The time axis does not use Auto-Scaling since small fluctuations in the actual start of the data may lead to a fluctuating x-axis minimum.

If any of the configuration settings are determined by the VI to be invalid for the active CompuScope system, the VI will STOP and show a descriptive error string. Users who prefer the VI to continue operating upon invalid setting entry should use GageCoerce.vi, which will coerce invalid entries to valid entries and continue running.

Please note that when the Channels index is changed from the default value (0), the controls will appear grayed out. This does not mean that you cannot change the control settings. You simply have to click on any of the grayed-out controls to activate them.

The VI grabs the handle to the first available CompuScope system and displays information about the system in the boxed indicator at the top right of the front panel. The displayed information is shown below:

- System Name – A string indicating the CompuScope model name of the hardware in the system.
- System Count – The total number of detected CompuScope systems.
- Board Count – The total number of CompuScope boards in the active CompuScope system.
- Channel Count – The total number of input channels in the active CompuScope system.
- Trigger Count – The total number of trigger engines in the active CompuScope system.
- Sample bits – The vertical resolution of the hardware in the active CompuScope system.

The type and entries for all controls and indicators are listed below:

Acquisition Control: Cluster of the Acquisition Setting Controls listed below:

- Sample Rate – The Sampling rate in Hz.
- Clock Mode – A ring control that specifies the clock mode for the CompuScope system. The available modes are:
  - Internal clock (use the built-in internal sampling oscillator of the CompuScope system)
  - External clock (use a clock signal connected to the external clock input of the CompuScope system as the sampling clock)
  - Reference clock (use a 10 MHz external clock signal connected to the external clock input of the CompuScope system to synchronize the internal CompuScope sampling oscillator).
- Mode – Set to “Single”, “Dual”, “Quad” or “Octal” for single-channel, dual-channel, four-channel or eight-channel acquisition mode.
- Depth – The post-trigger depth, which is the number of samples to acquire after the trigger event.
- Segment Size – Controls the total amount of memory allocated to the acquisition. The maximum possible amount of pre-trigger data that can be acquired, therefore, is (Segment Size - Depth).
- Trigger Timeout – Trigger Timeout in microseconds, which is the amount of time to wait before forcing a trigger event.
- Trigger Holdoff – The time in samples during which trigger events will be ignored after the CompuScope hardware begins acquiring and awaiting a trigger event. The function is useful for ensuring the accumulation of a specified amount of pre-trigger data that is equal to the Trigger Holdoff value.
- Trigger Delay – The time in samples between the logging of the trigger event and beginning of the count-down of the post-trigger depth counter. This function is

useful for waveforms in which the signal feature of interest occurs long after the trigger event.

Channel Control: An array of clusters of the Channel Setting Controls listed below. The Array index determines the channel number, starting from 0.

- Coupling – A ring control that contains “DC” and “AC” for the two input coupling modes.
- Differential – A button that activates Differential input mode, for CompuScope hardware that supports it.
- Direct-to-ADC – A button that activates Direct-to-ADC input mode, for CompuScope hardware that supports it.
- Input Range – The full scale input range in millivolts. For instance, for the +/-1 Volt range, enter 2000.
- Impedance – An integer control containing 1000000 (1 MOhm) and 50 (50 Ohms) for the two possible input terminations.
- DC Offset – The DC offset in millivolts that will be subtracted from the input signal, changing the effective input range. For instance, a 200 mV DC offset will change the extrema of the +/-1 Volt input range to 0.8 V and -1.2 V. DC offset is not supported by all CompuScope models.

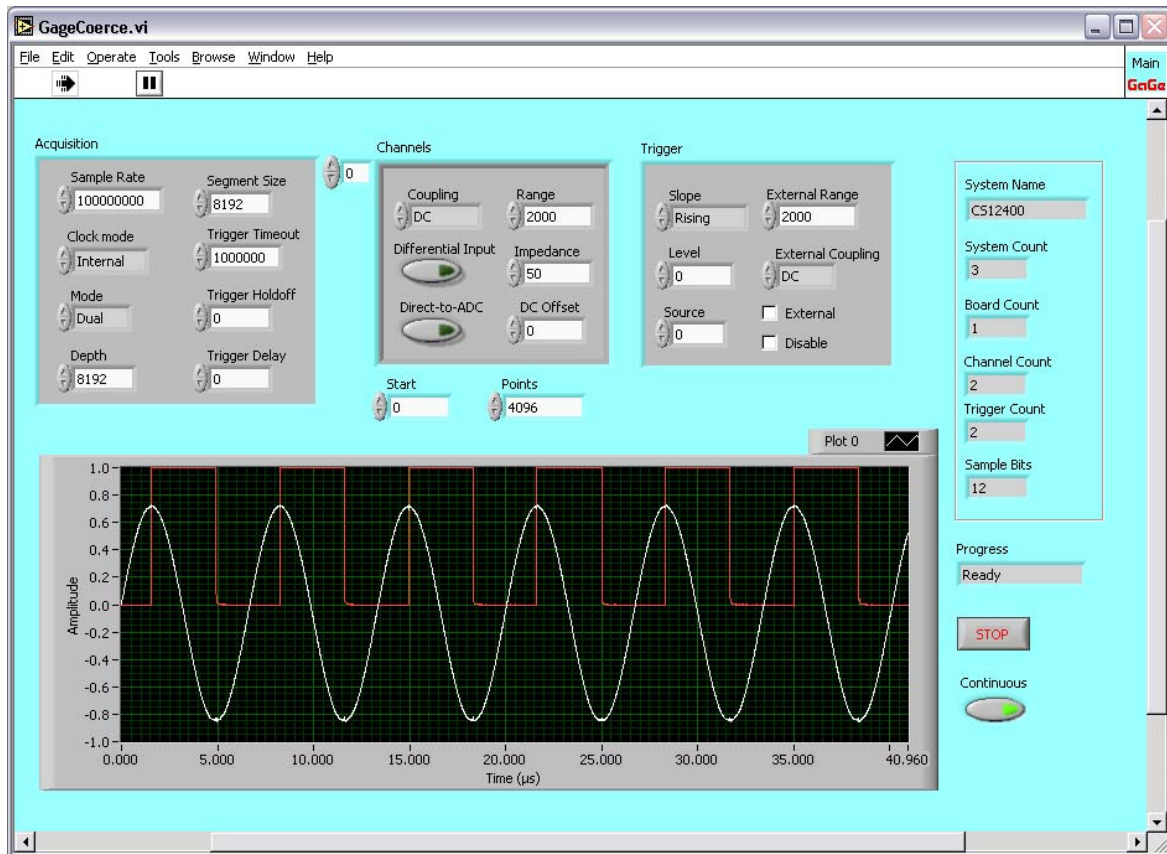
Trigger Control: A Cluster of Trigger Setting Controls

- Source – An integer which sets the channel trigger source if one is required. Channel trigger sources begin at 0 (for Channel 1) and can continue up to the number of channels available in the system. The Source is used if both the External and Disable buttons are not selected.
- External – A checkbox that sets the trigger source to be an external trigger. If this box is selected, the Source field is ignored and the External Range and External Coupling fields are used. Selecting both the External and Disable checkboxes will cause an error.
- Disable – A checkbox that disables the trigger source for the system. If this box is selected, the Source field is ignored and the system will not trigger unless CsTool-ForceCapture.vi is called. Selecting both the External and Disable checkboxes will cause an error.
- Level – The Trigger Level between +/-100%. The percentage is with respect to the input range of the trigger source.
- Slope – A ring control that sets the trigger slope to “RISING” or “FALLING”.
- External Range – List Box – An integer control that sets the full scale external trigger input range in millivolts. For instance, for the +/- 5 Volt range, select 10000.
- External Coupling – A ring control that contains “DC” and “AC” for the two external trigger coupling modes.

Transfer Control: Two variables that determine the data to be transferred, once captured:

- Start – The point from which to start data transfer. Zero indicates the trigger address. Enter a negative value in order to download pre-trigger data.
- Points – The number of points to transfer, starting from the Start point.

## GageCoerce.vi

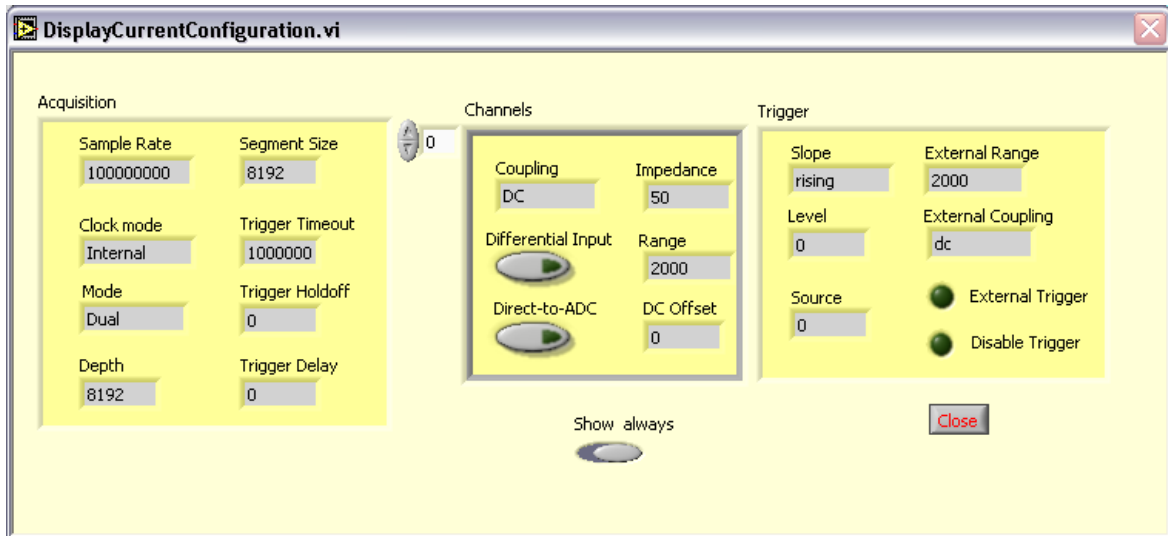


GageCoerce.vi is identical to GageAcquire.vi, except that in the event of invalid configuration setting entries, the VI will not stop with an error message. Instead of stopping, if invalid configuration settings are entered, the VI will coerce the settings to the closest values that are available on the current CompuScope system. When coercion occurs, a VI called DisplayCurrentConfiguration.vi, displayed below, pops-up displaying the actual current settings. DisplayCurrentConfiguration.vi uses the CsTool-Query VIs in order to obtain the actual configuration settings. If the user employs configuration settings values for future calculations, then it is very important that the values obtained by the CsTool-Query VIs be used and not the initially requested settings.

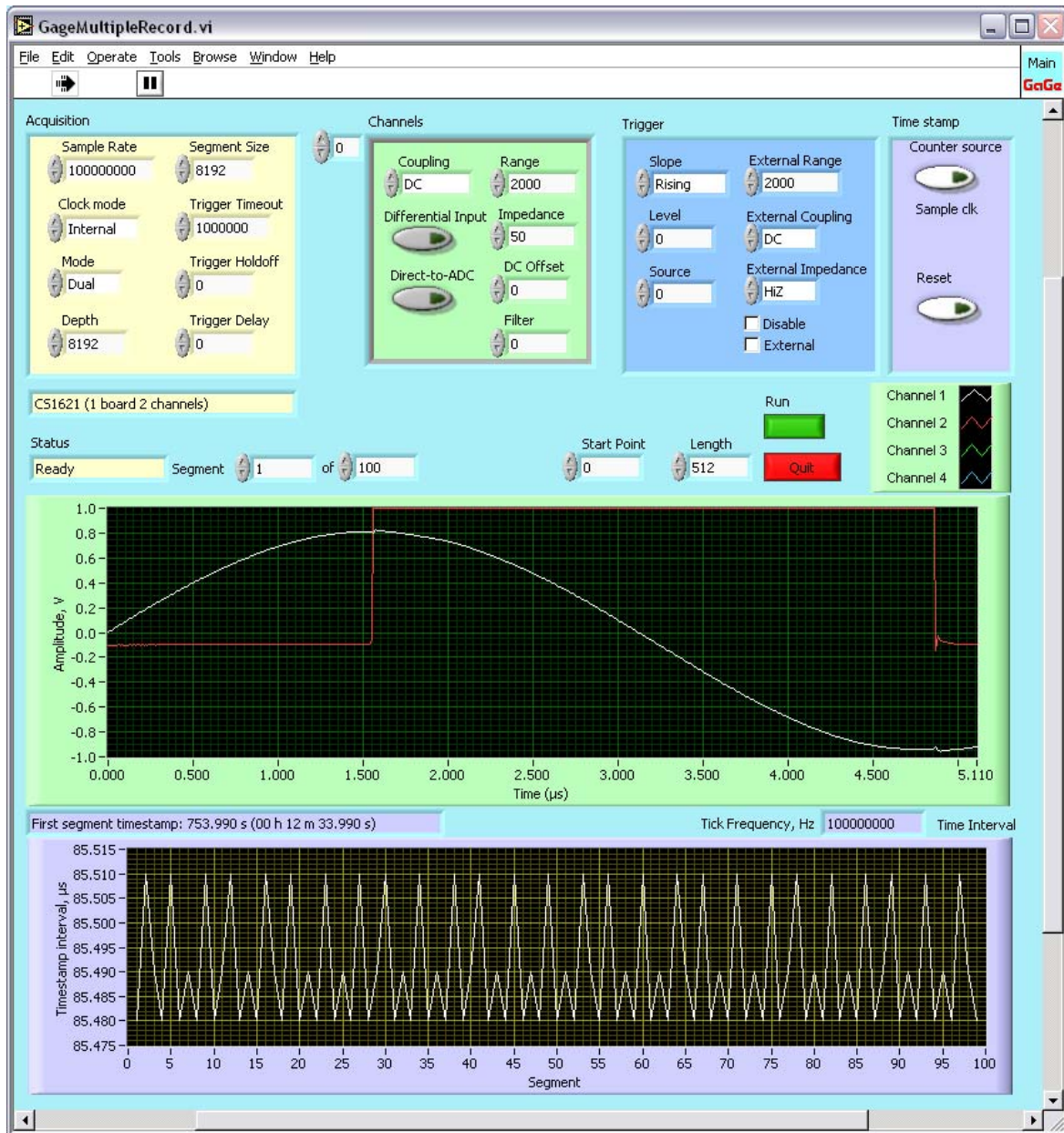
Coercion is actually executed by the CsTool-Commit.vi upon request. The coercion procedure varies for the different types of configuration settings. For the internal sampling rate, for instance, the coercion procedure chooses the closest available sampling rate. For example, if the user enters 6600000 as an internal sampling rate and only 50 MS/s and 10 MS/s are available, the coercion procedure will choose 50 MS/s.

All controls and indicators for GageCoerce.vi are identical to those for GageAcquire.vi. When executed, most Main VIs other than GageCoerce.vi will stop with an error upon invalid setting entry. The user can easily modify other VIs to coerce settings simply by following the logic illustrated in GageCoerce.vi





## GageMultipleRecord.vi



GageMultipleRecord.vi is the Main VI for Multiple Record capture from a single CompuScope system. Multiple Record mode allows multiple waveforms to be rapidly acquired and stacked in on-board CompuScope memory. For instance, a CompuScope card with 32 MegaSamples of on-board memory may acquire 16,000 records of 2000 samples each in Multiple Record mode. (Strictly speaking, depending on the CompuScope model, this relation may not exactly apply because of slight inter-record padding requirements.)

Between successive acquisitions, the CompuScope acquisition engine is re-armed in by the hardware with no CPU interaction required. Consequently, in Multiple Record mode, a CompuScope is capable of capturing bursts of trigger that repeat at rates of 100,000 triggers per second and more.

The Number of Records setting is used to enter the number of records desired. If this exceeds the maximum possible number of records, which is roughly equal to the available acquisition memory divided by the Depth, then an “Invalid Segment Count” error will occur. (If coercion had been used in this case, then the actual number of records would have been set equal to its maximum possible value.)

The Segment Size control is used to control the size of each Multiple Record segment. For newer CompuScope hardware, this may be set larger than the Depth so that pre-trigger data may be acquired. Older CompuScope models do not support pre-trigger data in Multiple Record mode. Consequently, setting the Segment Size to a value different from the Depth will cause an error. Please refer to your CompuScope Hardware Manual for information specific to your CompuScope model’s capabilities.

When executed, GageMultipleRecord.vi first initializes and configures the CompuScope system, as in GageAcquire.vi. Next, a single Multiple Record Acquisition is performed.

After acquisition, the VI goes into a loop that allows the user to select and display any one of the acquired records. Multiple records are displayed in a fashion that is identical to the MulRec Playback feature in GageScope. The front panel contains a string that reads “Record XX of YY”, where YY is the Number of Records. XX is the Segment control value that may be used to select the record to be downloaded and displayed. Since, with each change of XX, the VI downloads and displays a new record, only one record is resident in the VI at a time. The user is able to flip up and down through the records until the STOP button is pressed. The user may easily modify the VI so that the records are analyzed and/or stored, instead of being displayed.

Newer CompuScope models such as the CS14200, CS14105 and CS12400, support Time-Stamping, where each Multiple Record trigger event stores a counter value that indicates its time of occurrence. If the CompuScope system is equipped with on-board Time-Stamping, GageMultipleRecord.vi displays the Time Stamp for the displayed record in microseconds. Otherwise, 0’s are displayed as the Time Stamp value.

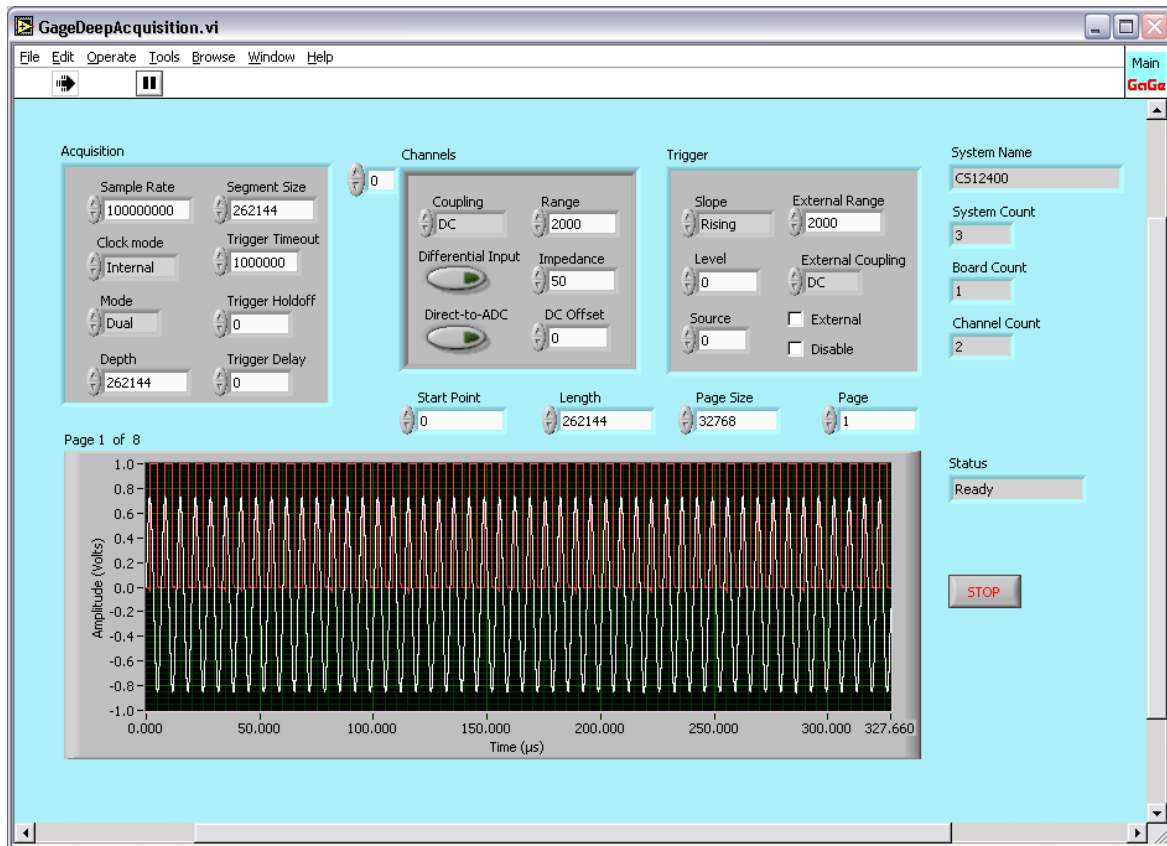
Time zero is the time at which the time-stamping counter was last reset. For convenience, the Time Interval display shows the difference between the Time Stamps for the current record and the previous record.

The Time stamp Reset button, when activated, will reset the time stamp counter at the beginning of each Multiple Record acquisition. Otherwise, the counter is never reset.

The Time stamp Clock source button selects the clock source of the time stamp counter: Fixed or Sample. “Fixed” selects an internal fixed oscillator that is asynchronous to the sampling. “Sample” selects a source that is synchronous to the sampling clock. “Sample” generally provides the greatest accuracy.

All remaining controls and indicators for GageMultipleRecord.vi are identical to those for GageAcquire.vi.

## GageDeepAcquisition.vi



GageDeepAcquisition.vi is the Main VI for large acquisitions from a single CompuScope system. The definition of large varies with System configuration but is roughly of order 16 MegaBytes. For small acquisitions, LabVIEW is fully capable of downloading the entire acquisition into PC RAM. For larger sizes, the host PC will begin to use “virtual RAM”, which is a section of the hard drive that Windows treats like PC RAM. Usage of virtual RAM causes the hard drive to spin and slows down execution. For large enough data sets, LabVIEW will simply refuse to download the data at all.

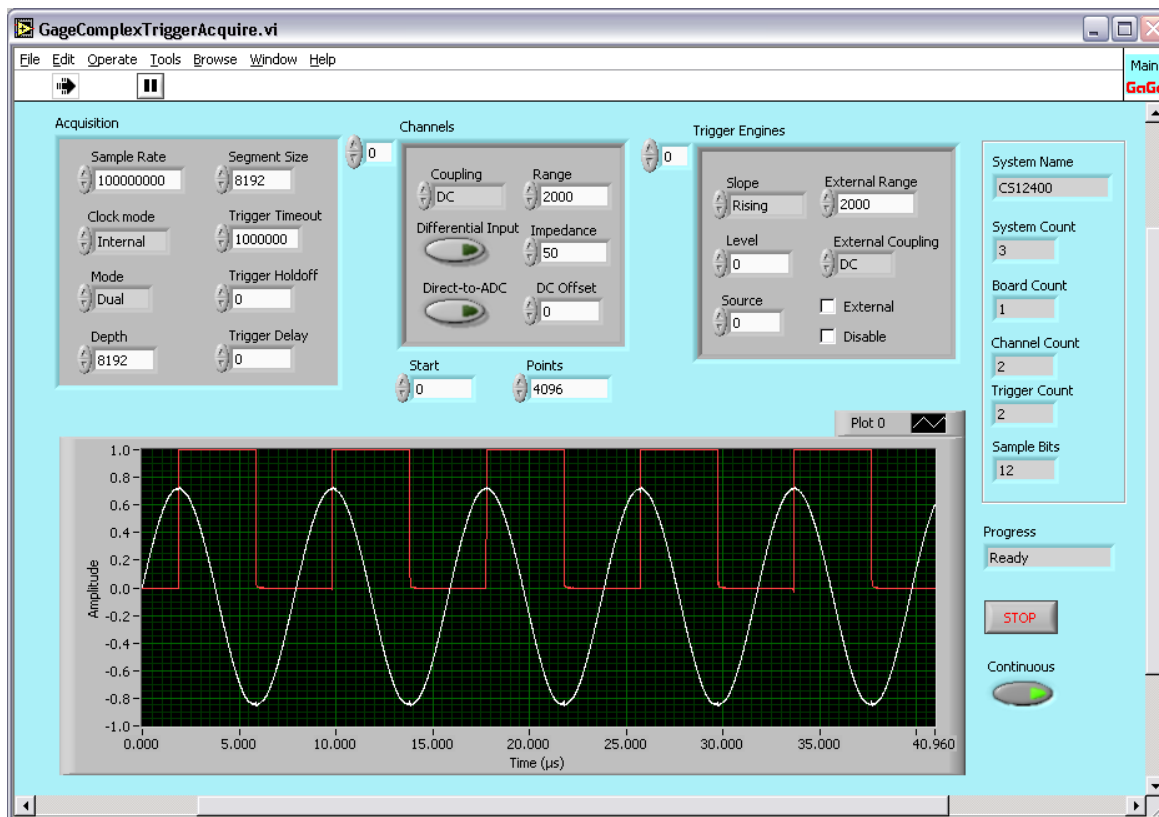
In order to avoid trying to keep large data sets in LabVIEW at one time, GageDeepAcquisition.vi manages deep acquisitions by dividing them up into more manageable data *pages*. The VI does initialization, configuration setting and acquisitions as usual. After the acquisition, however, only a single data page of the acquisition is downloaded and displayed.

After doing the acquisition, the VI goes into a loop that allows the user to select and display any page from the large acquisition. There is a control/indicator pair that shows “Page XX of YY”, where XX is the value in the adjustable Page control. YY is the total number of Pages that is equal to (Length/Page Size). With each change of Page control value, the VI downloads and displays a new page, so that only one page is resident in the VI at a time. The adjustable Page Size default is 32768. The time x-axis is offset to indicate the page position in time. For example, if the user displays Page 11 of a capture taken at 100 MS/s, the lower x-axis limit would be  $10 \times 32k / 100 \text{ MS/s} = 326.78$  milliseconds.

The user is able to flip up and down through the pages until the STOP button is pressed. The user can easily modify the VI so that the pages are analyzed and/or stored instead of being displayed.

All remaining controls and indicators for GageDeepAcquisition.vi are identical to those for GageAcquire.vi.

## GageComplexTriggerAcquire.vi



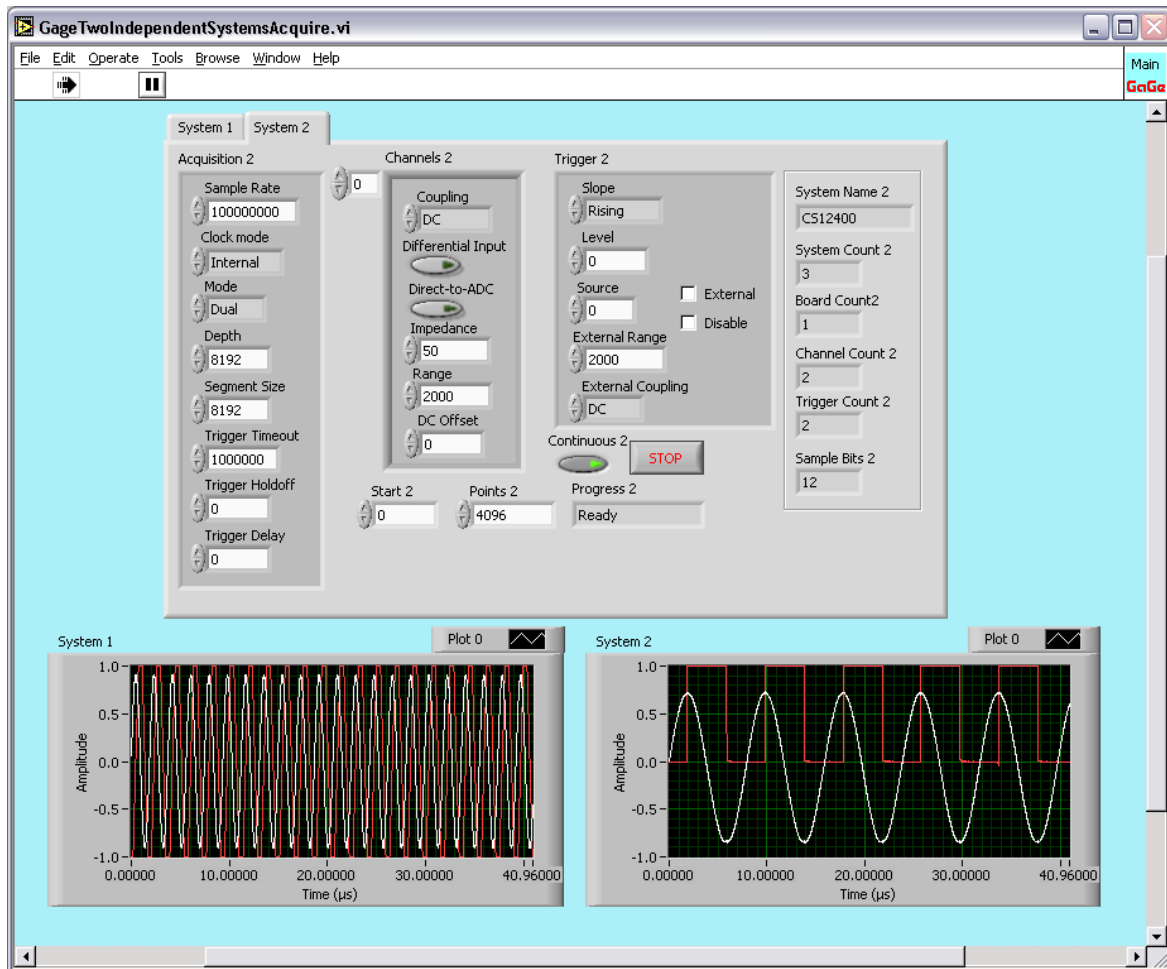
GageComplexTriggerAcquire.vi is a main VI that illustrates complex triggering. Some CompuScope models are equipped with two on-board trigger engines that can be used for complex triggering. On these models, the two engines can be configured independently and their outputs are Boolean ORed together so that either engine may cause a trigger event. For simple triggering, the second engine is disabled.

The Triggers array control allows the separate configuration of each trigger engine. For instance, by setting the two engine sources to 0 and 1, the user may configure the system to trigger on a pulse that occurs on either channel. (In CompuScope LabVIEW systems, channels start at 0). Alternately, by setting both engine sources to 0 but selecting different levels and slopes for each, the user may configure the system to do windowed triggering, where the system triggers if the input level leaves a specified voltage range.

Please note that when the Trigger Engines index is changed from the default value (0), the controls will appear grayed out. This does not mean that you cannot change the control settings. You simply have to click on any of the grayed-out controls in order to activate them.

All remaining controls and indicators for GageComplexTriggerAcquire.vi are identical to those for GageAcquire.vi.

## GageTwoIndependentSystemsAcquire.vi:



GageTwoIndependentSystemsAcquire.vi is the main VI for Multiple Independent CompuScope Systems. The VI is identical to GageAcquire.vi, except that there are two identical sets of controls and indicators for the two CompuScope systems. The two sets of controls and indicators for the two systems may be accessed by selecting the “System 1” or the “System 2” tabs. The VI operates two systems, but can be easily extendable to three or more Systems by the user.

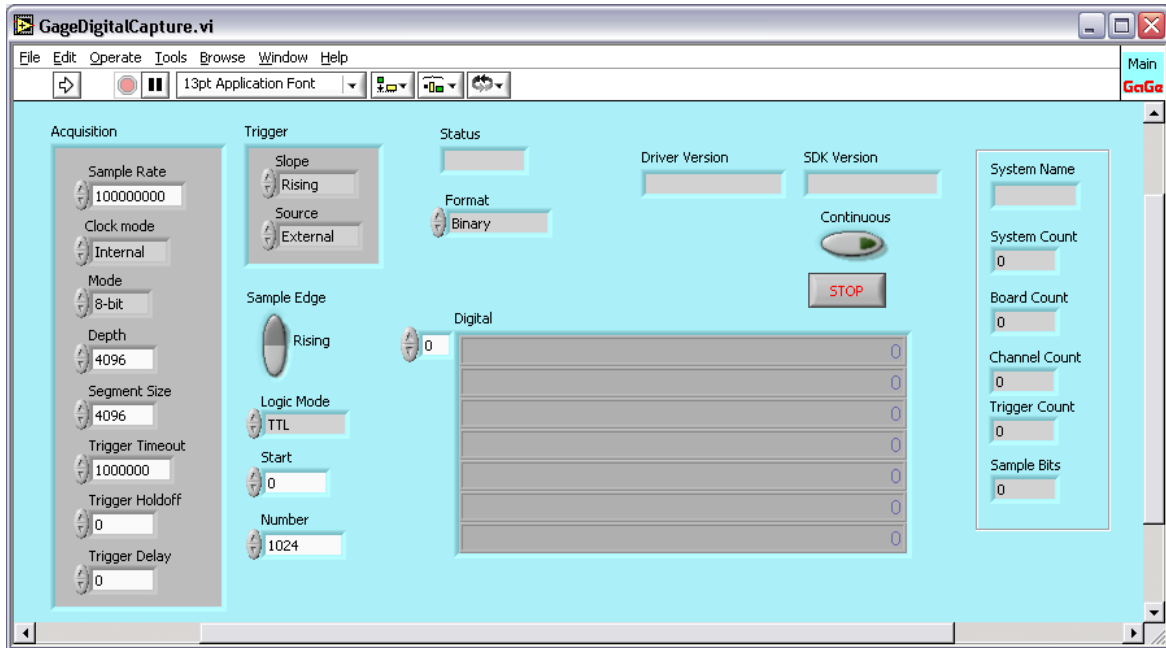
GageTwoIndependentSystemsAcquire.vi acquires waveforms from the two CompuScope systems in a completely asynchronous fashion. Waveform acquisition on the two systems is controlled from two independent loops between which LabVIEW’s built-in multi-tasking engine toggles in an indeterminate fashion. The VI, therefore, is designed to operate two CompuScope systems that are acquiring unrelated signals and triggers. In order to trigger both CompuScope systems simultaneously, the user must connect the same external trigger signal to both systems. Furthermore, the user must ensure that external trigger signals are inhibited until both cards are capturing. Otherwise, an external trigger may trigger one system and not the other.

Note: This VI will coerce all configuration values to be valid available settings for all CompuScope systems, as does GageCoerce.vi. Unlike GageCoerce.vi, however, GageTwoIndependentSystemsAcquire.vi does not display the actual coerced settings. Consequently, the user should bear in mind that the settings that were requested may not actually have been implemented.

All remaining controls and indicators for GageComplexTriggerAcquire.vi are identical to those for GageAcquire.vi



## GageDigitalCapture.vi



GageDigitalCapture.vi performs acquisitions from digital input CompuScope cards, such as the CS3200 and CS3200C. The Mode control determines the digital input sample width, which can be selected as 8-bit, 16-bit or 32-bit. The Clock Edge control allows selection of the edge of the clock on which the CompuScope will sample the data (Rising or Falling). The Logic Mode allows selection of the desired logic protocol (TTL or CMOS).

Captured digital data are displayed as a numerical array. The user may choose to display the data in Binary, Decimal or Hexadecimal by making a selection from the Format control.

All remaining controls and indicators for GageDigitalCapture.vi are similar to those for GageAcquire.vi.

### ***Advanced Main VIs***

The LabVIEW SDK contains “Advanced” VIs in addition to the documented Main VIs. Usage of some of these files may require special CompuScope hardware options, on-board firmware processing images or special driver versions. These VIs are provided as-is with limited documentation in the form of on-line LabVIEW Help and/or an accompanying explanatory text file.

## Chapter 4: Special Topics

### **Converting from CompuScope ADC Code to Voltages**

SDK sample programs are configured to save or display CompuScope waveform data that have been scaled so that the sample values are in Volts. The user may want to bypass the voltage conversion step, however, in order to achieve the best repetitive capture performance. Voltage conversion may then be done at leisure in post-processing.

Raw ADC code waveform data may be converted to voltage data for all CompuScope models using the following equation:

$$Voltage = \frac{Offset - ADC\_Code}{Resolution} \times \frac{(Full\ Scale\ Input\ Voltage)}{2} + DC\_Offset$$

The *Offset* and *Resolution* for the CompuScope system may be obtained programmatically using various SDK tools. The DC offset settings, supported only on newer CompuScope models, must be added.

For instance, for the CompuScope 82G, *Offset*=127 and *Resolution*=128. Let us assume that the user has selected the +/-1 Volt Input Range, for a Full Scale Input Range of 2 Volts. Let us further assume that the user has applied a 200 mV offset. For this example, therefore, the voltage conversion equation becomes:

$$Voltage = \frac{127 - ADC\_Code}{128} \times 1\ Volts + 0.2\ Volts$$

### **Depth and Segment Size**

On all CompuScope models, on-board acquisition memory is arranged as a circular buffer for Single Record acquisitions. This means that when the CompuScope is storing acquired data and it reaches the end of the memory, the memory counter rolls over. Data storage wraps around the memory and starts digitizing into the beginning of memory. The end of an acquisition is always initiated by the trigger event, after which the requested number of post-trigger data points is acquired and then the acquisition terminates. This circular memory architecture allows the CompuScope to accumulate an amount of pre-trigger data of up to the amount of CompuScope acquisition memory per channel, less the amount of post-trigger data.

Newer CompuScope models, such as the CS82G, CS12400, CS14200 and CS14105, allow for the accumulation of pre-trigger data in Multiple Record Mode. On these models, memory is divided into multiple circular buffers, the number of which is equal to the number of records specified. The Segment Size control allows the user to specify the size of each circular buffer. This way, the user may select the amount of memory reserved for pre-trigger data accumulation. For instance, if the user selects a post-trigger Depth of 2048 and a Multiple Record Segment Size of 8192, then up to 6144 Samples of pre-trigger data may be acquired. Increasing the Segment Size beyond the Depth will accordingly reduce the maximum possible number of Multiple Records.

Only newer CompuScope models such as the CS82G, CS12400, CS14200 and CS14105, support pre-trigger Multiple Record Mode. On older CompuScope models without pre-trigger data in Multiple Record Mode, the Segment Size setting will have no effect on a Multiple Record acquisition. The value is ignored and is forced to be equal to the Depth by the drivers.

### ***Trigger Holdoff***

Trigger Holdoff is a feature that is useful for ensuring the accumulation of a specified amount of pre-trigger data. The Trigger Holdoff setting specifies the amount of time, in Samples during which the CompuScope hardware will ignore trigger events after acquisition has begun and pre-trigger data are being acquired.

Without a non-zero Trigger Holdoff value, there is no guarantee that a given number of pre-trigger samples will be acquired. This is because a trigger event may occur immediately after acquisition, leading to a very small number of pre-trigger points. By ignoring trigger events for a time equal to the specified Trigger Holdoff value, the accumulation of a number of pre-trigger points equal to the Trigger Holdoff setting is guaranteed.

The downside of ensuring pre-trigger data with a non-zero Trigger Holdoff value is that triggers are ignored, so that important trigger events may be missed. For instance, in lightning monitoring applications, researchers usually want to acquire pre-trigger data. These data give information about signal behavior immediately preceding the lightning strike, which triggers the CompuScope hardware. Lightning strikes may occur very close together in time, however, and missing a lightning pulse is much worse than missing pre-trigger data. Consequently, lightning testers should not use Trigger Holdoff but should simply accept acquisitions of only the amount of pre-trigger data that naturally occur between triggers. Generally speaking, therefore, the user must decide whether to use Trigger Holdoff, based on the application.

### ***Trigger Delay***

New-generation PCI CompuScope models such as the CS14200, CS14105 and CS12400, support a feature called Trigger Delay. This feature is useful for situations where the signal portion of interest occurs long after the trigger event.

Normally, with a Trigger Delay of zero, the trigger event activates count-down of the post-trigger depth counter, which was preloaded with the post-trigger depth. The counter is decremented by one count for each sample acquired after the trigger event. When the counter value reaches 0, the acquisition is terminated. In this way, the CompuScope acquires a number of samples equal to the depth after the trigger event.

A non-zero Trigger Delay value is used to delay the beginning of the countdown of the post-trigger depth counter. The Trigger Delay value sets the number of samples that the CompuScope hardware will wait after the trigger event occurs before beginning the count-down of the depth counters. The SegmentSize may be set equal to the depth so that the CompuScope hardware need not waste memory by storing data that are not of interest.

As an example, consider a signal where the feature of interest is 20,000 samples long but begins 100,000 samples after the trigger event. In this case, the SegmentSize and Depth

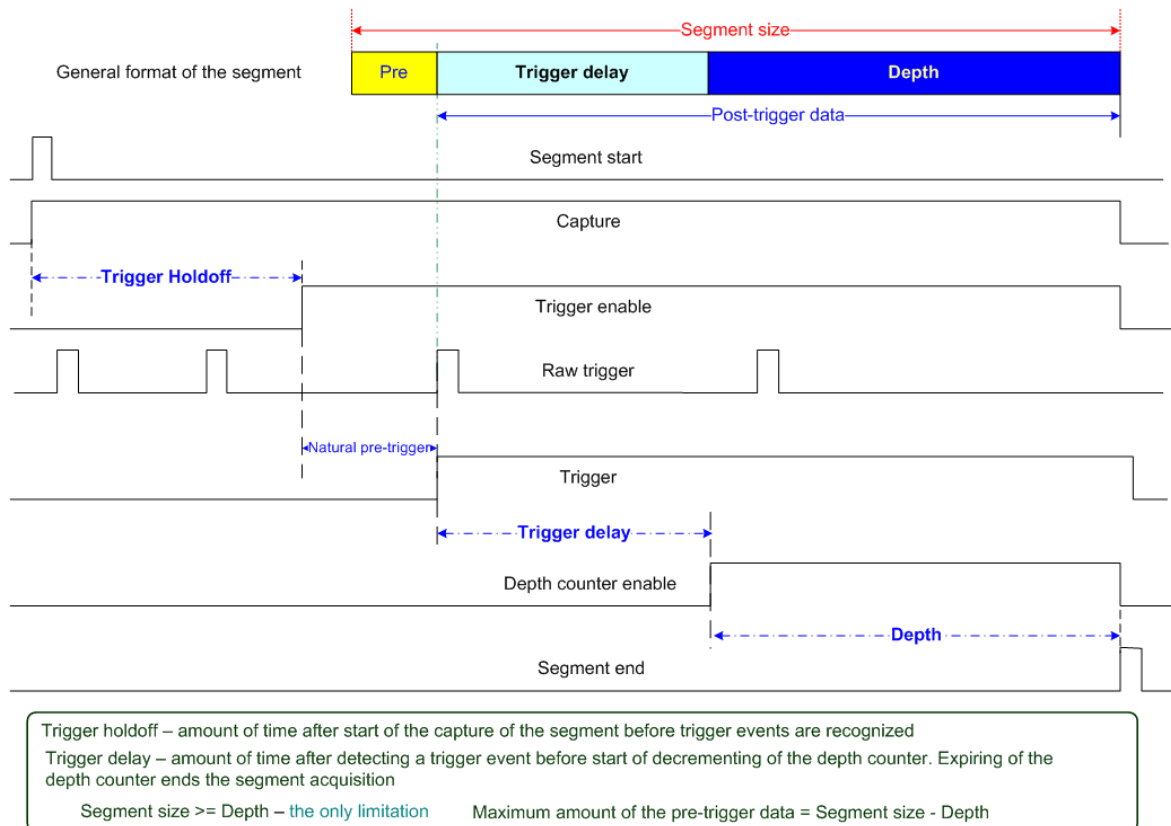
should be set to 20,000. The Trigger Delay value should be set to 100,000. Without the Trigger Delay feature, the user would be forced to set the Depth to 120,000 Samples and waste 100,000 Samples of memory, even though only the last 20,000 were downloaded. Using the Trigger Delay feature, however, only the data of interest are retained in CompuScope memory.

### CompuScope Acquisition Timing Diagram

The timing diagram below is provided as an aid for understanding the timing of events during the acquisition of a segment or record. The pseudo—signals are indicated as HIGH when the labeled function is active.

A CompuScope system always acquires from the beginning of the Segment Start pulse and continues until the Depth counter's count-down has expired. The memory allotted to the acquisition is equal to the Segment Size and is arranged in a circular fashion. Notice that raw trigger events are ignored until the Trigger Holdoff time has elapsed.

Note also that, although pre-trigger data are acquired throughout the time between Segment Start and the Trigger Event, most of the acquired pre-trigger data have been overwritten in the diagram below and only a small fraction are available for download.



## Representative Acquisition Sequences

The diagram below illustrates important representative acquisition sequences that are presented for key acquisition requirements. The first sequence shows a generalized acquisition with data collected during all acquisitions phases illustrated: overwritten pre-trigger data, available pre-trigger data, data acquired during the Trigger Delay and post-trigger data.

The second sequence shows an acquisition using a trigger delay where the Segment Size is made equal to the Depth. This acquisition sequence is useful for applications where the signal region of interest occurs long after the trigger event, as is often the case for Time Domain Reflectometry methods such as ultrasonics, radar and lidar.

The third sequence shows the acquisition in the case of a rapidly occurring trigger. In this case, the Segment size has been set bigger than the Depth. However, the trigger event occurred so rapidly after the start of the acquisitions that there was insufficient time to fill all of the memory allotted for pre-trigger data, which is (Segment Size – Depth) samples. Consequently, there is less preceding valid trigger data than the maximum possible amount of pre-trigger data. If transfer of invalid data were requested, the driver would return an actual start address that is equal to the address of the first valid pre-trigger point. More pre-trigger data could have been acquired using Trigger Holdoff at the expense ignoring (and so possibly missing) triggers.

The final sequence shows acquisition in the case of a trigger that occurs long after Segment Start so that all memory allotted for pre-trigger data has been filled up. In fact, the sequence shows that still more pre-trigger data were acquired but were overwritten by post-trigger data.



## Technical Support

We offer technical support for all our Software Development Kits.

In order to serve you better, we have created a web-based technical support system that is available to you 24 hours a day.

By utilizing the internet to the fullest, we are able to provide you better than ever technical support without increasing our costs, thereby allowing us to provide you the **best possible product at the lowest possible price.**

To obtain technical support, simply visit:

[www.gage-applied.com/support/support\\_form.php](http://www.gage-applied.com/support/support_form.php)

Please complete this form and submit it. Our form processing system will intelligently route your request to the Technical Support Specialist (TSS) most familiar with the intricacies of your product. This TSS will be in contact with you within 24 hours of form submittal.

In the odd case that you have problems submitting the form on our web site, please e-mail us at

[tech-support@gage-applied.com](mailto:tech-support@gage-applied.com)

As opposed to automatic routing of technical support requests originating from the GaGe web site, support requests received via e-mail or telephone calls are routed manually by our staff. Providing you with high quality support may take an average of 2 to 3 days if you do not use the web-based technical support system.

**Please note that Technical Support Requests received  
via e-mail or by telephone will take an average of 2 to 3 days to process.  
It is faster to use the web site!**

When calling for support we ask that you have the following information available:

1. Version and type of your CompuScope SDK and drivers.  
(The version numbers are indicated in the About CD screen of the CompuScope CD.  
Version numbers can also be obtained by looking in the appropriate README.TXT files)
2. Type, version and memory depth of your CompuScope card.
3. Type and version of your operating system.
4. Type and speed of your computer and bus.
5. If possible, the file saved from the Information tab of the CompuScope Manager utility.
6. Any extra hardware peripherals (i.e. CD-ROM, joystick, network card, etc.)
7. Were you able to reproduce the problem with standalone GaGe Software (e.g. GageScope, GageBit)?

# GaGe Products

For ordering information, see Gage's product catalog, or visit our web site at <http://www.gage-applied.com>

<b>PCI Bus Digitizers</b>	EON CompuScope Family	One or two channels, 12 bit resolution at up to 2 GS/s
	CompuScope 1250X Razor CompuScope Family	One or two channels, 12 bit resolution at 500 MS/s Up to 4 channels on a single-slot PCI card, 12, 14, or 16-bit resolution, 100 to 200 MS/s
	Octopus multi-channel CompuScope Family	Up to 8 channels on a single-slot PCI card, 12, 14, or 16-bit resolution, 10 to 125 MS/s
	CobraMax CompuScope Family	One or two channels, 8-bit resolution, up to 4 GS/s
	Cobra CompuScope family	One or two channels, 8-bit resolution, up to 2 GS/s
	BASE-8 CompuScope	One channel, 8-bit resolution, 500 MS/s
	CompuScope 14105	14 bit, resolution at 105 MS/s – Transformer-coupled
	CompuScope 3200	32 bit, 100 MHz Digital Input Card for PCI Bus
<b>USB Digitizers</b>	CompuScope 144002U	Two channels, 12 bit resolution, 400 MS/s
	CompuScope 148001U	One channel, 12 bit resolution, 800 MS/s
	CompuScope 121G11U	One channel, 12 bit resolution, 1.1 GS/s
<b>CompuGen PCI</b>	CompuGen 4300/4302	12 bit, 4-channel, 300 MHz Analog Output Card
	CompuGen 8150/8152	12 bit, 8-channel, 150 MHz Analog Output Card
	CompuGen 11G/11G2	12 bit, 1 GHz Analog Output Card
	CompuGen 12G/12G2	12 bit, 2 GHz Analog Output Card
<b>CompuGen ISA</b>	CompuGen 1100	12 bit, 80 MS/s Analog Output Card
	CompuGen 3250	32 bit, 50 MHz Digital Output Card
<b>Application Software</b>	GageScope	World's Most Powerful Oscilloscope Software
	GageBit	Digital Input/Output Software
	CompuGen for Windows	Arbitrary Waveform Generator Software for Windows
<b>Software Development Kits</b>	CompuScope SDK for C/C#	
	CompuScope LabVIEW SDK	
	CompuScope MATLAB SDK	
	CompuGen SDK for C/C++	
	CompuGen SDK for LabVIEW	
	CompuGen SDK for MATLAB	
<b>Instrument Mainframes</b>	LapScope-1	1-slot PCI Expansion Chassis for CompuScope and CompuGen cards
	LapScope-2	2-slot PCI Expansion Chassis for CompuScope and CompuGen cards
	Instrument Mainframe 7500	4-slot Portable Instrument Mainframe for CompuScope and CompuGen cards
	Instrument Mainframe 2020E	18 PCI slot and 1 ISA slot Instrument Mainframe for CompuScope and CompuGen cards
	Instrument Mainframe 4000	5 PCI-X and 1 PCI slot Instrument Mainframe for CompuScope and CompuGen cards