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.

Find the *Keysight / Agilent E1431A* at our website: *Click HERE*

HP E1431A

25.6 kHz Eight Channel VXI Input Module

User's Guide

Part Number E1431-90005
Microfiche Number E1431-90205

Printed in U.S.A.
Print Date: June 1995

©Hewlett-Packard Company, 1994-1995. All rights reserved.
8600 Soper Hill Road Everett, Washington 98205-1298 U.S.A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

HEWLETT PACKARD

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MANUAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

WARRANTY

A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another language without the prior written consent of Hewlett-Packard Company. This information contained in this document is subject to change without notice.

Use of this manual and flexible disc(s) or tape cartridge(s) supplied for this pack is restricted
to this product only. Additional copies of the programs can be made for security and back-up purposes only.

© Copyright 1983, 1984, 1985, 1986, 1987, 1988 Hewlett-Packard Company.

© Copyright 1979 The Regents of the University of Colorado, a body corporate.

© Copyright 1979, 1980, 1983 The Regents of the University of California.

© Copyright 1980, 1984 AT&T Technologies. All Rights Reserved.

© Copyright 1986, 1987 Sun Microsystems, Inc.

© Copyright 1984, 1985 Productivity Products Intl.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the government is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

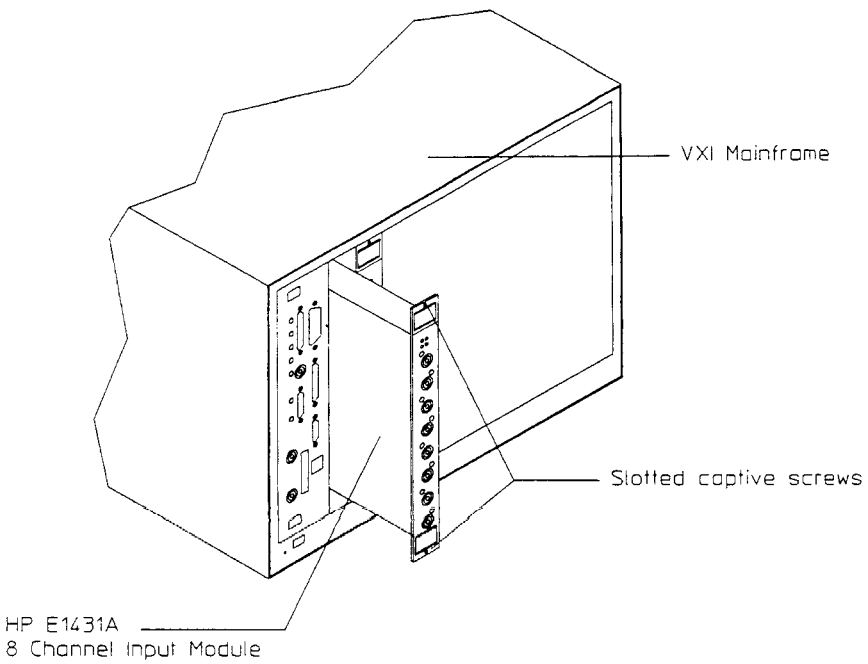
HEWLETT-PACKARD COMPANY
3000 Hanover St.
Palo Alto, CA 94303

Rights for non-DOD U.S. Government Departments and Agencies are set forth in FAR 52.227-19 (c) (1,2)

Copyright (c) 1994 Hewlett-Packard Company. All rights Reserved

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The HP E1431A at a Glance



Number of Channels	8
Type of Input	Differential
Input Bandwidths (alias protected)	0.39 Hz to 25.6 kHz
Samples per Second	1, 2, 4, 8, ... , 65,536
Voltage Ranges (1-2-5 sequence)	5 mV to 10 V peak
ICP Supply	4 mA
Number of ADC bits	16
Noise floor	–90 dB (typical)
Harmonic and intermodulation distortion	72 dB
Triggering	Input
VXI bus support	VME and Local Bus
VXI device type	Register-based
Size	C-sized, single slot

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

What you get with the HP E1431A

The following items are included with your HP E1431A:

Hardware:

- HP E1431A Input - C-size VXI module
- Software media:
  HP-UX tape
  MS-DOS® disks

Software:

HP-UX Tape

- The HP E1431A C Interface Library; including source files, HP-UX Series 300 C Library binaries and HP-UX Series 700 C Library binaries
- HP E1485A/B C Library binaries
- C-SCPI libraries for the HP E1431A (HP-UX Series 300 and HP-UX Series 700)
- SCPI downloadable for the HP E1405/06 Command Module
- HP VEE/ITG driver
- Example programs

Programming Software Disk

- The HP E1431A C Interface Library source files
- Example programs

SCPI Driver for Command Module Disk

- SCPI downloadable for the HP E1405/06 Command Module
- HP VIC downloader for HP E1405/06 Command Module
- DOS downloader for HP E1405/06 Command Module
- HP VEE/ITG driver

Documentation:

- HP E1431A User's Guide (this book)
- Online manual pages accessed via ptman (HP-UX only)

MS-DOS is a US registered trademark of Microsoft Corporation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

In This Book

The HP E1431A 25.6kHz Eight Channel VXI Input Module has built-in signal conditioning and anti-alias filters. The module plugs into a single C-size slot in a VXI mainframe.

This book documents the HP E1431A module. It provides:

- installation and service information
- operational information
- C Library software support reference material
- SCPI command reference materials

Associated with this product is the HP E1485A/B and the HP 35635T Programmer's Toolkit. The HP E1485 A/B is a VXI signal processing module. The 35635T Programmer's Toolkit consists of libraries and tools that form an application program development environment.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table of Contents

1 Installing the HP E1431A

Installing the HP E1431A 1-2
To inspect the HP E1431A 1-2
To install the HP E1431A 1-3
To install the HP VEE driver 1-6
To install the C library interface 1-7
To install the compiled SCPI (C-SCPI) driver 1-8
To install the downloadable SCPI driver 1-9
To store the module 1-11
To transport the module 1-11

2 Specifications

Specifications 2-2

3 Troubleshooting the HP E1431A

To troubleshoot using an RS-232 interface 3-2
To troubleshoot using an HP-IB interface 3-4

4 Replaceable Parts

Replaceable Parts 4-2
To remove the front panel 4-7

5 Circuit Description

Circuit Description 5-2

6 Backdating

Backdating 6-2

7 Using the HP E1431A

Front Panel Description 7-2
Input Mode: Voltage or ICP 7-3
VXI Backplane Connections 7-4
The HP E1431A's Measurement Process 7-6

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8 Programming the HP E1431A with the C Interface Libraries

Getting Started 8-3

C Libraries Example Program 1 8-7

HP E1485A/B C Libraries Example Program 2 8-10

C Libraries Quick Reference 8-14

9 C Interface Library Support Reference

Function Reference 9-2

Errors 9-93

10 Programming the HP E1431A with SCPI

Getting Started 10-2

Using the Status Registers 10-6

The HP E1431A Registers Sets 10-11

Addressing the HP E1431A 10-20

SCPI Quick Reference 10-25

11 SCPI Command Reference

Finding the Right Command 11-3

Command Syntax 11-4

Command Reference 11-8

Errors 11-80

Appendix: The VXI Registers

VXI Register Description A-2

Glossary

Index

Declaration of Conformity

Need Assistance?

About this Edition

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1

Installing the HP E1431A

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Installing the HP E1431A

This chapter contains instructions for installing the HP E1431A 25.6 kHz 8 Channel VXI Input Module and its drivers. This chapter also includes instructions for transporting and storing the module.

To inspect the HP E1431A

The HP E1431A 25.6 kHz 8 Channel VXI Input Module was carefully inspected both mechanically and electrically before shipment. It should be free of marks or scratches, and it should meet its published specifications upon receipt.

If the module was damaged in transit, do the following:

- Save all packing materials.
- File a claim with the carrier.
- Call your Hewlett-Packard sales and service office.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To install the HP E1431A

If you will be using the HP E1405/06 Command Module and an external computer with DOS based windows, use the HP VXI Installation Consultant (HP VIC) to install the HP E1431A module. Before starting HP VIC, insert the "SCPI Driver for Command Module" disk into your computer. HP VIC steps you through the installation procedure then tests the modules using the *TST? command. HP VIC may time out before the test is finished and display a "timed out" message. If this occurs, exit HP VIC and send the *TST? command. For instructions on sending the *TST? command, see chapter 3, "Troubleshooting the HP E1431A."

Caution

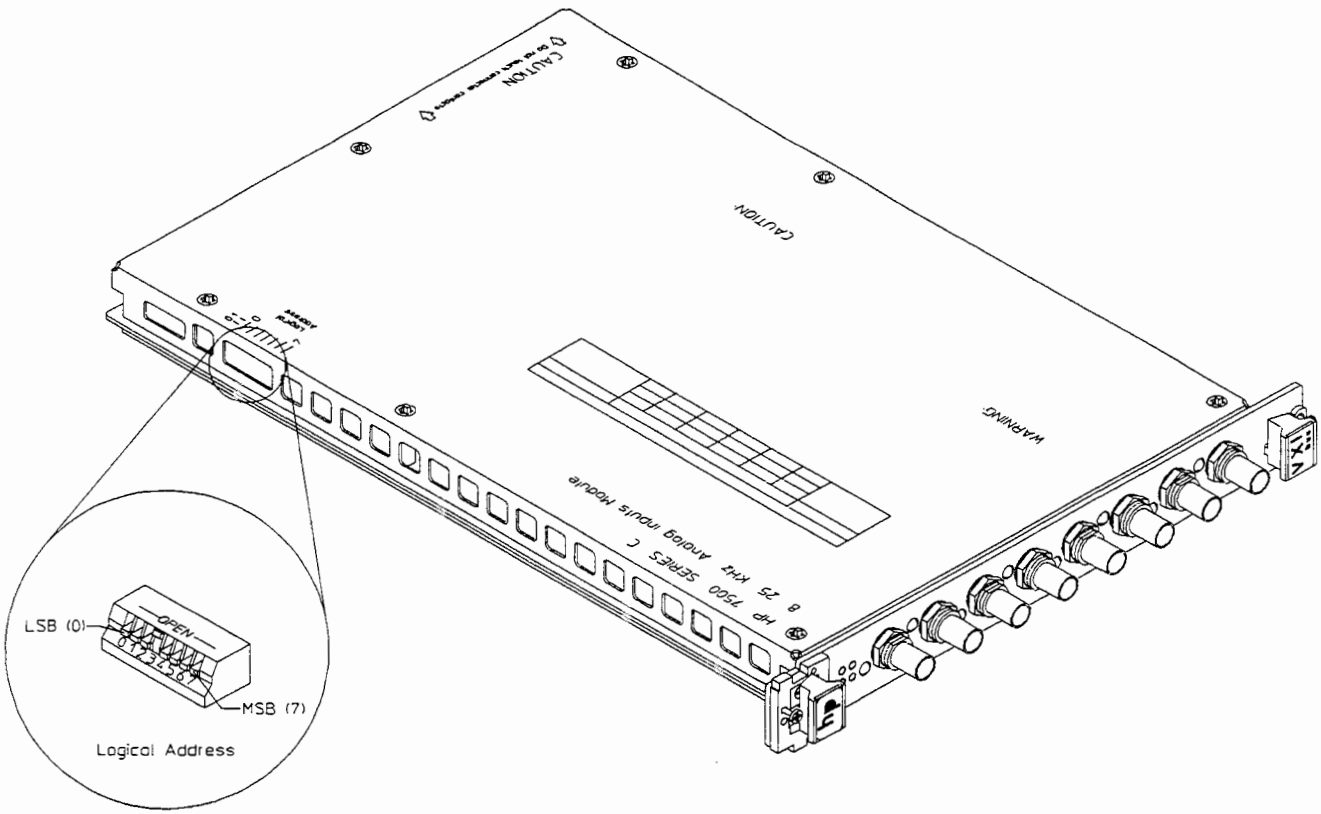
To protect circuits from static discharge, observe anti-static techniques whenever handling the HP E1431A 25.6 kHz 8 Channel VXI Input Module.

1 **Set up your VXI mainframe. See the installation guide for your mainframe.**

2 **Select a slot in the VXI mainframe for the HP E1431A module.**
The HP E1431A module's local bus receives ECL-level data from the module immediately to its left and outputs ECL-level data to the module immediately to its right. Every module using the local bus is keyed to prevent two modules from fitting next to each other unless they are compatible. If you will be using the local bus, select adjacent slots immediately to the left of the data-receiving module. The local bus can support up to five HP E1431A modules at full span at real time data rates. If the VXI Bus is used, maximum data rates will be reduced but the module can be placed in any available slot.

3 **Using a small screwdriver or similar tool, set the logical address configuration switch on the HP E1431A.**
Each module in the system must have a unique logical address. The factory default setting is 0000 1000 (8). If an HP E1485A Signal Processor Module will be controlling the HP E1431A module, select an address within the HP E1485A module's servant area. If an HP-IB command module will be controlling the HP E1431A module, select an address that is a multiple of 8.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



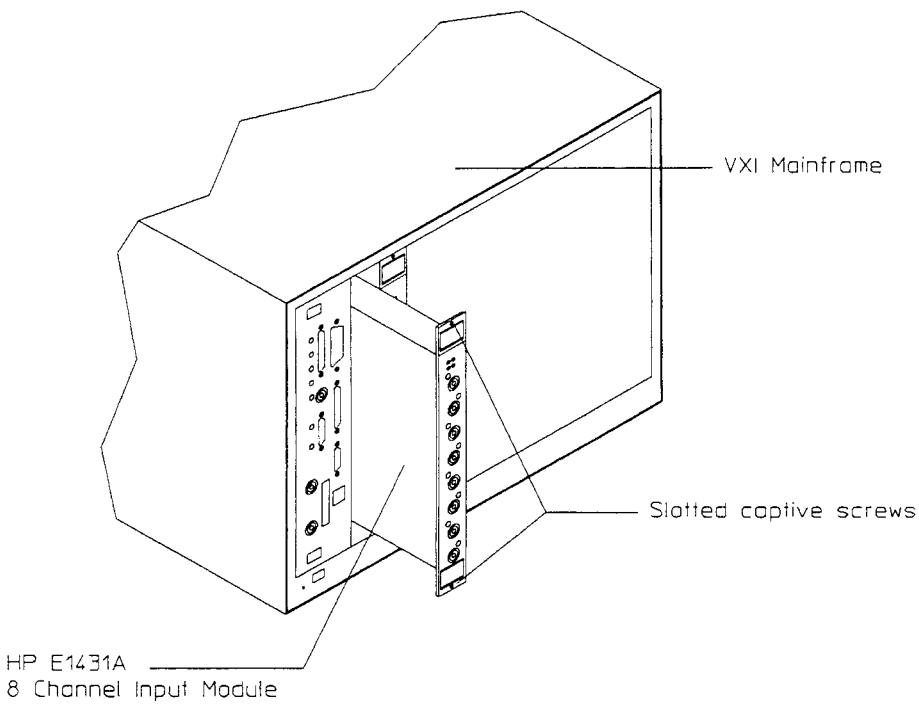
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4 Set the mainframe's power switch to standby (⏻).

5 Place the module's card edges (top and bottom) into the module guides in the slot.

6 Slide the module into the mainframe until the module connects firmly with the backplane connectors. Make sure the module slides in straight.

7 Attach the module's front panel to the mainframe chassis using the module's captive mounting screws.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To install the HP VEE driver

❑ Using an HP-UX operating system and one of the following:

- HP Series 300 Computer
- HP Series 700 Computer
- HP V/382 Embedded Computer

See the "Getting Started with HP VEE" manual for instructions on installing the HP VEE driver.

❑ Using a Windows operating system and one of the following:

- Personal Computer
- HP RADI-EPC7 Embedded Computer

The HP VEE driver is on the "E1431A Programming Software" disk. See the "Getting Started with HP VEE" manual for instructions on installing the HP VEE driver.

❑ HP VEE is not supported on the DOS operating system.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To install the C interface libraries

❑ Using an HP-UX operating system and one of the following:

- HP Series 300 Computer
- HP Series 700 Computer
- HP V/382 Embedded Computer

Do the following to install the C interface libraries:

1 Log in as root.
2 Insert the E1431A HP-UX tape into the tape drive.
3 Type `/etc/update`.

See the HP-UX Reference manual for information on the update command.

❑ Using a Windows or DOS operating system and one of the following:

- Personal Computer
- HP RADI-EPC7 Embedded Computer

The C library source files are on the "E1431A Programming Software" disk. The files are supplied for supplemental information.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To install the compiled SCPI (C-SCPI) driver

❑ Using an HP-UX operating system and one of the following:

- HP Series 300 Computer
- HP Series 700 Computer
- HP V/382 Embedded Computer

Three driver files are named E1431.o, but they are not identical. One file is for the C-SCPI preprocessor program and the other two are linked to the C-SCPI library. The files are placed in the /usr/e1431/cscpi/inst, /usr/e1431/cscpi/pic/inst, and /usr/e1431/cscpi/preproc directories during the update process.

After you install C-SCPI, do the following to install the C-SCPI driver:

1 Log in as root.
2 Insert the E1431A HP-UX tape into the tape drive.
3 To install the C library interface, type `/etc/update`.
  See the HP-UX Reference manual for information on the update command.
4 Type `/usr/e1431/bin/updatecscpi`.

❑ C-SCPI is not supported on the Windows or DOS operating system.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To install the downloadable SCPI driver

The SCPI driver is downloaded into the HP E1405/06 Command Module.

❑ Using an HP-UX operating system and one of the following:

- HP Series 300 Computer
- HP Series 700 Computer
- HP V/382 Embedded Computer

The downloadable version of the driver is named E1431. To accommodate older methods of downloading drivers, the driver files E1431.DU and E1431.DC are also included. The update process places these files in the /usr/e1431/scpi directory.

The utility, vxidldux, downloads the E1431 file into the Command Module. Either the Series 300 or the Series 700 version of this utility is taken off the update tape depending on the file set chosen.

Do the following to install the SCPI driver:

1 Set the HP E1405/06 Command Module's HP-IB address to 0000 1001 (9).
2 Connect an HP-IB cable from your computer to the HP E1405/06 Command Module.
3 Log in as root.
4 Insert the E1431A HP-UX tape into the tape drive.
5 To install the C library interface, type `/etc/update`.

   See the HP-UX Reference manual for information on the update command.

6 Type `/usr/e1431/bin/vxidldux -d/dev/hpib7 /usr/e1431/scpi/E1431`.

   See man pages on vxidldux or the HP E1401-90021 installation note if you have difficulty.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

❑ Using a Windows operating system and one of the following:

- Personal Computer
- HP RADI-EPC7 Embedded Computer

Do the following to install the SCPI driver using the HP VXI Installation Consultant (HP VIC).

1 Connect an RS-232 cable (HP 24542U) from your computer to the HP E1405/06 Command Module.
2 Insert the "SCPI Driver for Command Module" disk into your computer.
3 Start HP VIC.
4 Select the HP E1431 from the "HP INSTRUMENTS" list.

   If the HP E1431 did not appear in the list, make sure the "SCPI Driver for Command Module" disk is in the computer's disk drive and restart HP VIC.

5 Follow the instructions provided by HP VIC.

   HP VIC steps you through the installation procedure then tests the modules using the *TST? command. HP VIC may time out before before the test is finished and display a "timed out" message. If this occurs, exit HP VIC and send the *TST? command. For instructions on sending the *TST? command, see chapter 3, "Troubleshooting the HP E1431A."

❑ Using a DOS operating system and one of the following:

- Personal Computer
- HP RADI-EPC7 Embedded Computer

Do the following to install the SCPI driver:

1 Connect an RS-232 cable from your computer to the HP E1405/06 Command Module.
2 If you are using an HP terminal with RS-232 interface, set the baud rate to 9600 and check that the HP E1405\06 Command Module is configured as resource manager and slot 0 device (logical address = 0, system controller and slot 0 enabled, and servant area = 255).
3 If you are not using an HP terminal, see the HP E1405/06 Command Module's User's Guide for RS-232 setup information.
4 Insert the "SCPI Driver for Command Module" disk into drive A: on your computer.
5 Type `A:\VXIDLD`.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To store the module

Store the module in a clean, dry, and static free environment.

For other requirements, see storage and transport restrictions in chapter 2, "Specifications."

To transport the module

- **Package the module using the orginal factory packaging or packaging identical to the factory packaging.**
  Containers and materials identical to those used in factory packaging are available through Hewlett-Packard offices.
- **If returning the module to Hewlett-Packard for service, attach a tag describing the following:**
  - Type of service required
  - Return address
  - Model number
  - Full serial number

  In any correspondence, refer to the module by model number and full serial number.
- **Mark the container FRAGILE to ensure careful handling.**
- **If necessary to package the module in a container other than original packaging, observe the following (use of other packaging is not recommended):**
  - Wrap the module in heavy paper or anti-static plastic.
  - Protect the front panel with cardboard.
  - Use a double-wall carton made of at least 350-pound test material.
  - Cushion the module to prevent damage.

Caution

Do not use styrene pellets in any shape as packing material for the module. The pellets do not adequately cushion the module and do not prevent the module from shifting in the carton. In addition, the pellets create static electricity which can damage electronic components.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2

Specifications

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Specifications

To test the HP E1431A 25.6 kHz 8 Channel VXI Input Module's performance, contact your nearest Hewlett-Packard Service Center. Hewlett-Packard does not supply customer performance test software for the HP E1431A. The HP E1431A's recommended calibration cycle is every 24 months.

Instrument specifications apply after 1 hour warm up. Unless stated otherwise, all specifications apply under the following conditions: ICP disabled, the alias protection filter enabled, factory gain corrections enabled, and within 24 hours and ± 5 °C of an auto-zero operation and phase calibration operation.

Definitions

dBfs, %fs = Relative to full scale of a particular range. (0 dBfs = 100% of full scale)

dBVpk = dB relative to 1 Vpk: $dBV_{pk} = 20 \times \log_{10}\left(\frac{Vpk}{1\ Vpk}\right)$

FS, fs = Full scale; synonymous with input range. The full scale measure is a limit to the peak input voltage.

Real Time and **Online** = Collecting and displaying information with no dropouts or missing information.

Signal "high" side = Input BNC center pin.

Signal "low" side = Input BNC shell.

Typical = Typical, non-warranted, performance specification included to provide general product information.

Vpk = Peak of the ac voltage.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Frequency

Input frequency range	dc to 25.6 kHz
Frequency spans	25.6 kHz down to 0.39 Hz in 2 × steps
Sampling frequency	
Fixed internal sample rate	65,536 Hz
External sample rate	≤ 65,536 Hz (requires user-provided external alias protection for rates below 65,536 Hz)
External sample clock	3 × external sample rate
Internal sample clock accuracy	± 0.01% (± 100 ppm)
Block sizes	1 to 16384 samples, integer steps
Word sizes	16 or 32 bit word transfers

Input

Full scale input ranges	5 mVpk to 10 Vpk in 1, 2, 5 steps
Input impedance (dc coupled)	
Differential	2 MΩ nominal
High side to chassis – floating mode	1 MΩ, 35 pF nominal
Low side to chassis – floating mode	1 MΩ, 0.01 µF nominal
Low side to chassis – grounded mode	< 90 Ω typical
AC coupling 3 dB corner frequency	< 1 Hz
Common mode rejection ratio	
DC coupled	
< 67 Hz	≥ 65 dB
67 Hz to 1000 Hz	≥ 55 dB
AC coupled	
45 Hz to 1000 Hz	≥ 45 dB
Common mode voltage range	± 10 Vpk maximum

Maximum signal present on either the high or low side of the input connector relative to chassis ground.

Amplitude over-range detection

Over-range indication after any of:

Common mode overload	± 12.5 Vpk nominal
Signal over-range, broadband detector	+ 3 dBfs nominal (@100 kHz)
ADC overload	+ 3 dBfs nominal (≤ 25.6 kHz)
Residual dc (after auto-zero)	≤ 1% of full scale, or ≤ 1 mVdc

Amplitude

Absolute amplitude accuracy (FFT measurement)	± 1.5% of reading ± 0.035% of full scale

(A combination of full scale accuracy, full scale flatness, and amplitude linearity.)

Amplitude Accuracy at 1 kHz (A combination of full scale accuracy and amplitude linearity)	± 1% ± 0.035% of full scale
Accuracy at 1 kHz at 100% of full scale, factory gain corrections enabled	± 1% (± 0.09 dB)
Flatness relative to 1 kHz, at full scale	± 0.5% (± 0.043 dB)
Amplitude linearity at 1 kHz	± 0.035% of full scale

ICP is a registered trademark of Piezotronic Inc.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Cross Channel

Cross channel magnitude	± 3% (± 0.26% dB)

DC coupled, ac coupled above 10 Hz, full scale.

Cross channel phase match, phase delay match across HP E1431A channels in any module in the same VXI mainframe.

DC coupled, or ac coupled above 500 Hz

At 25.6 kHz span	± 1.0°
At 12.8 kHz span	± 0.5°
< 1 kHz	± 0.1°

Cross channel time delay match across HP E1431A channels in any module in the same VXI mainframe, dc coupled.

At 25.6 kHz span	± 200 ns
At 12.8 kHz span	± 100 ns
< 1 kHz	± 100 ns
Cross-mainframe time delay match (typical between first and last mainframe in daisy-chain)	70 ns + cable delay (6 ns/meter)

FFT Dynamic Range

Amplitude resolution	16 bits (less 2 dB overhead)
FFT noise floor	<– 90 dBfs

Hann window, 800 line resolution, 16 rms averages

Spurious free dynamic range	> 80 dB (harmonics ≤ 5 kHz)

(Includes spurs, harmonic distortion, crosstalk, intermodulation distortion, alias products)

Crosstalk between input channels, any range combination	<– 125 dB below signal or <– 80 dBfs of receiving channel, whichever response is greater in amplitude.
Input noise level (≥ 100 Hz)	$< 40\ ^{nV_{rms}/\sqrt{Hz}}$
Spurious and residual responses	<– 80 dBfs
Harmonic distortion (measurements ≤ 5 kHz)	<– 80 dBfs
Single tone (in band), ≤ 100% of full scale (measurements > 5 kHz)	<– 72 dBfs
Intermodulation distortion	<– 80 dBfs

Two tones (in-band), each ≤ 50% of full scale

Frequency alias responses, signal < 170 kHz	<– 90 dBfs

Trigger

Trigger modes	Input channel (same HP E1431A) Input channel (other HP E1431A)
Maximum trigger delay	
Pre trigger	16383 sample periods
Post trigger	> 10^6 sample periods
Input channel trigger level resolution	14 bits
Trigger uncertainity	± 0.5 × (1/65536 Hz*) * sample rate

ICP® current source

ICP® current source	
Current source	4.5 mA ± 0.5 mA
Open circuit voltage	≥ 24 Vdc
Intermodulation distortion	<– 75 dBfs

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

General Specifications

Safety standards	This product is designed for compliance to: UL1244, Fourth Edition IEC 348, 2nd Edition, 1978 CSA C22.2, No. 231
EMI/RFI Standards	CISPR 11, Group 1, Class A Note: Requires VXI Mainframe, PN HP E1401A OPT 918, and Single Slot Filler Panels, PN E1400-60202.
VXI bus standards	VXI (Rev. 1.4); Register based; A16/D16; VMEbus slave; requires 2 TTLTRG_ lines for multi-module synchronization

VXI power requirement	dc	Dynamic current
+ 5 V	1.05 A	0.27 A
– 5.2 V	0.66 A	0.03 A
– 2 V	0.23 A	0.15 A
+ 12 V	1.50 A	0.01 A
– 12 V	0.52 A	0.00 A
+ 24 V	0.22 A	0.01 A
– 24 V	0.24 A	0.00 A
+ 5 V Standby	0.00 A	0.00 A

VXI cooling requirement (10 °C rise)	3.5 liters/s 0.5 mm H_2O
Warm-up time	60 minutes
Weight	Net 2.1 kg (4.6 lbs) Shipping 5 kg (11 lbs)
Dimensions	Single slot, C-size VXI module

Environmental

Operating restrictions	
Ambient temperature	0° to 55 °C
Humidity, non-condensing	20% RH to 95% RH at 40 °C
Maximum altitude	2300 meters (7,500 feet)
Storage and transport restrictions	
Ambient temperature	– 40° to 65 °C
Humidity, non-condensing	20% RH to 90% RH at 65 °C
Maximum altitude	4600 meters (15,000 feet)

Benchmarks

HP E1431A module count for real time data transmission at full span

Local bus	5 modules
VXI bus (host dependent)	~ 1 module

Transmission rate

Local bus	2.62 Msamples/s
VXI bus (host dependent)	~ 1 Msamples/s

Timing

The following timing comparison on software version A.00.01 is between an HP E1406A Command Module and an HP 9000 Model 715/50 Workstation with a MXIbus interface using HP E1431A C Interface Libraries.

	HP E1406A	HP Model 715/50
Autozero		
1 module	1 min 06 s	15 s
2 modules	2 min 00 s	18 s
Autozero and Phase †		
1 module	6 min 00 s	19 s
2 modules	11 min 31 s	27 s
Autorange ‡ (1 kHz sine wave input)		
8 channels		
10 V input	1 s	130 ms
0.1 V input	1 s	80 ms
16 channels		
10 V input	4 s	380 ms
0.1 V input	2 s	160 ms
Self test		
1 module	1 min 18 s	26 s
2 modules	2 min 13 s	30 s

† Cal mode parameter = frequency; see e1431_auto_zero
‡ Wait time parameter = 0.01 s; see e1431_auto_range

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3

Troubleshooting the HP E1431A

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To troubleshoot using an RS-232 interface

Equipment Required: "C" size mainframe
HP E1405/06 Command Module with HP E1431A SCPI driver
HP terminal with RS-232 interface

❑ Step 1. Configure the system.

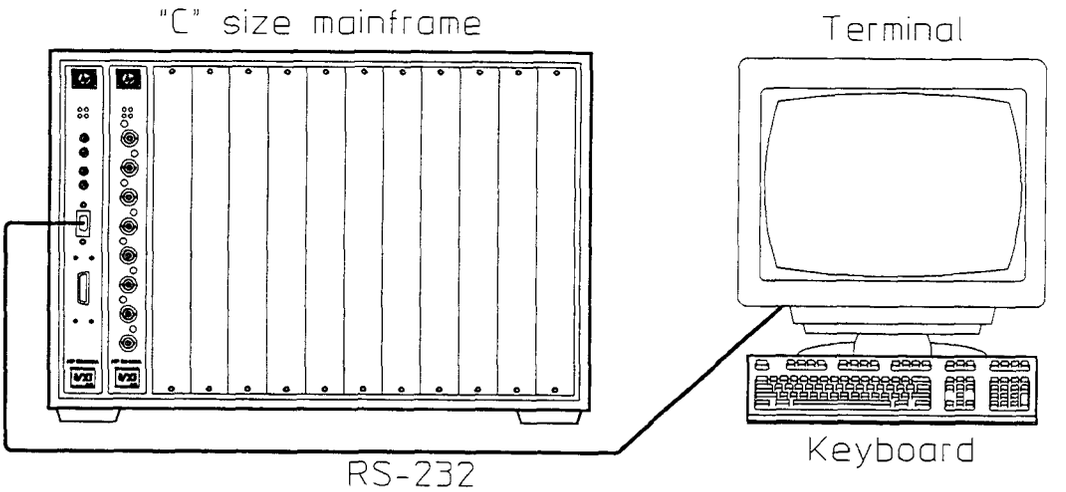
1 Set the mainframe's power switch to standby (⏻).

2 Check that the HP E1405/06 Command Module is configured as resouce manager and slot 0 device (logical address = 0, system controller and slot 0 enabled, and servant area = 255).

See the command module's documentation for the location of the switches and information on using its terminal interface.

3 Slide the command module into the mainframe until it connects firmly with the backplane connectors.

4 Connect an RS-232 cable from the terminal to the command module's RS-232 port.



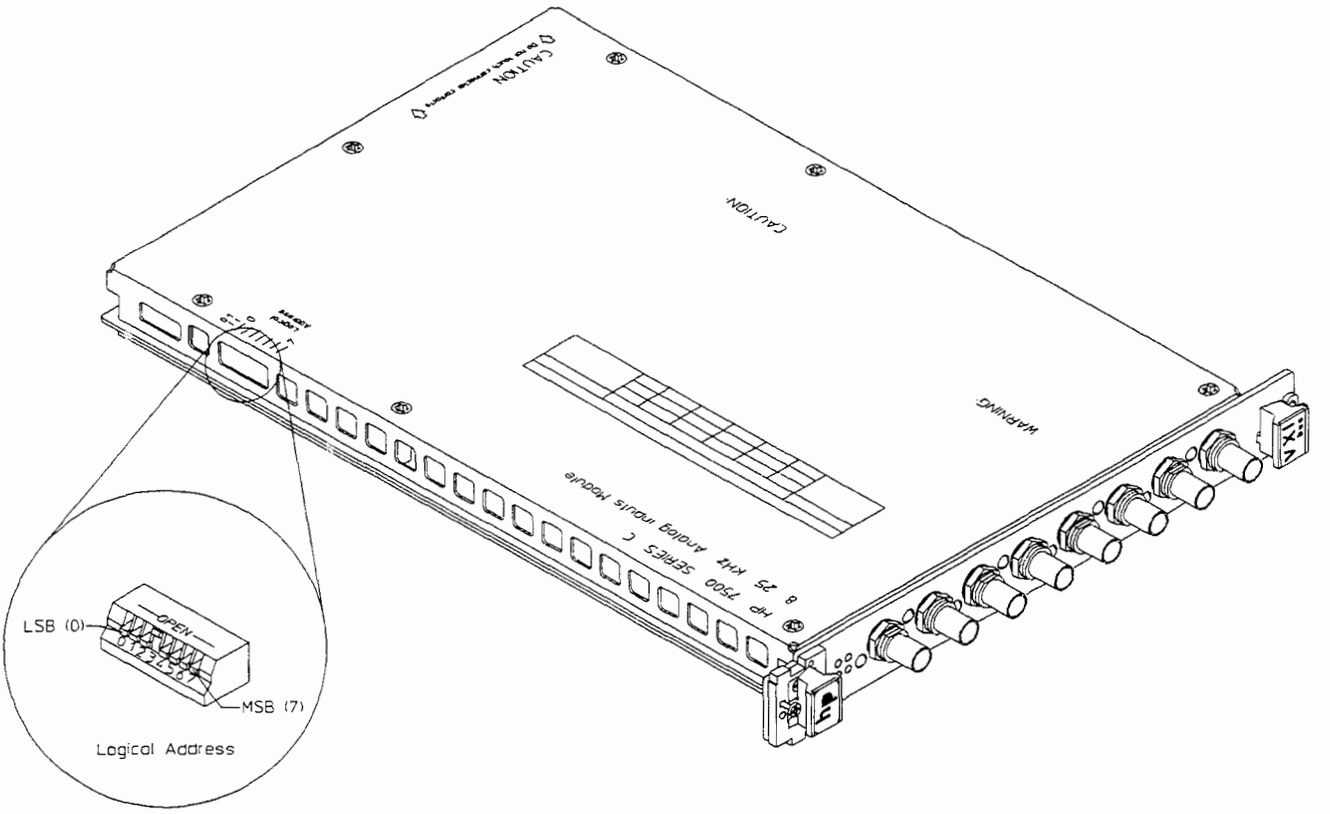
5 Set the speed of the communication port to match the command module (9600 baud).

6 Set the HP E1431A module's logical address to 0000 1000 (8).

7 Slide the HP E1431A module into the mainframe until it connects firmly with the backplane connectors.

8 Set the mainframe's power switch to on (|).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



❑ Step 2. Test the HP E1431A module.

1 Select the E1431 terminal softkey.

2 Type `*idn?` and press Enter.

The response should be "Hewlett-Packard, E1431A, *nnnnAnnnnn*, *A.nn.nn*" where *nnnnAnnnnn* is the module's serial number and *Ann.nn* is the module's software revision. If there is no response within 20 seconds, check your setup (cabling, address switch settings, baud rate) and run the program again.

3 Type `*tst?` and press Enter.

This test should take approximately 2 minutes.

4 If the response is `+0`, the HP E1431A module is operating correctly.

5 If the response is `+1`, the HP E1431A module is faulty.

The HP E1431A module's repair strategy is module exchange. See chapter 4, "Replaceable Parts," for the replacement procedure and exchange module part number.

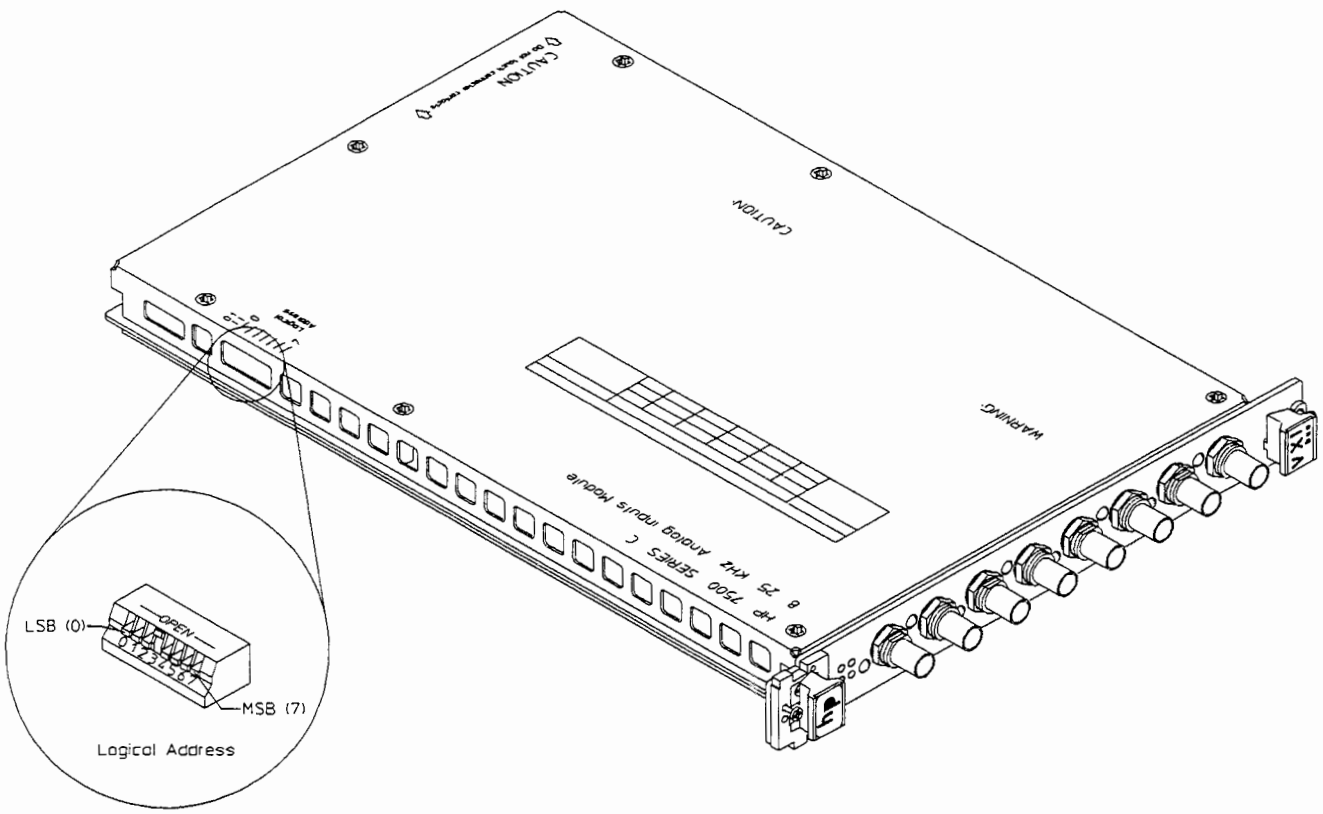
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To troubleshoot using an HP-IB interface

Equipment Required: "C" size mainframe
HP E1405/06 Command Module with HP E1431A SCPI driver
Series 200/300 computer with an HP-IB interface and RMBASIC

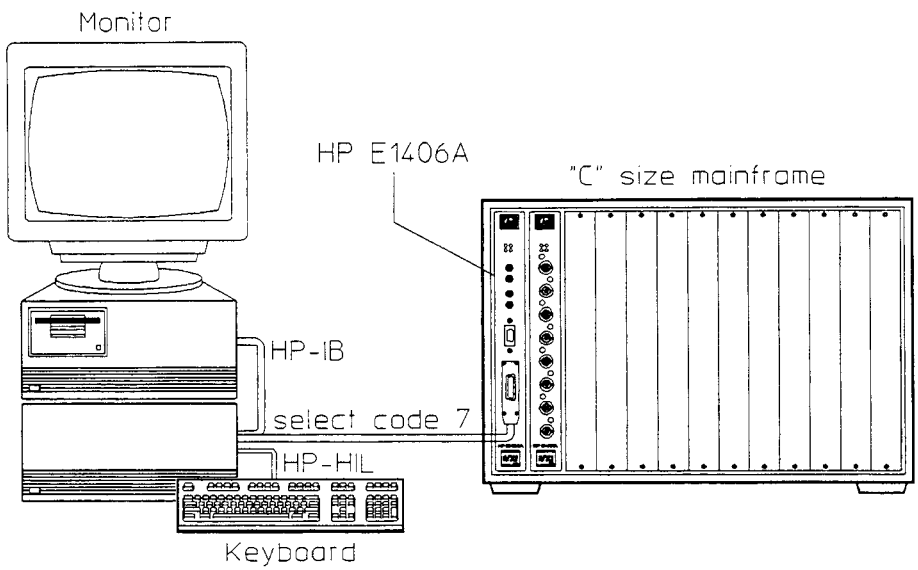
❑ Step 1. Configure the system.

1 Set the mainframe's power switch to standby (⏻).
2 Set the HP E1405/06 Command Module's HP-IB address to 0000 1001 (9).
3 Set the HP E1431A module's logical address to 0000 1000 (8).



4 Connect an HP-IB cable from the computer to the command module's HP-IB port.
5 Set the mainframe's power switch to on (|).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



❑ Step 2. Test the HP E1431A module.

1 Type in and run the following program:

```
10 DIM Response$[80]
20 OUTPUT 70901;"*idn?"
30 ENTER 70901;Response$
40 DISP Response$
50 END
```

The response should be "Hewlett-Packard, E1431A, *nnnnAnnnnn, A.nn.nn*" where *nnnnAnnnnn* is the module's serial number and *Ann.nn* is the module's software revision. If there is no response within 20 seconds, check your setup (cabling, address switch settings) and run the program again.

2 Type in and run the following program:

```
10 OUTPUT 70901;"*tst?"
20 WAIT 90
30 ENTER 70901;Response$
40 DISP Response$
50 END
```

This test should take approximately 2 minutes.

3 If the response is +0, the HP E1431A module is operating correctly.

4 If the response is +1, the HP E1431A module is faulty.

The HP E1431A module's repair strategy is module exchange. See chapter 4, "Replaceable Parts," for the replacement procedure and exchange module part number.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4

Replaceable Parts

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Replaceable Parts

The HP E1431A 25.6 kHz 8 Channel VXI Input Module's circuit assemblies can not be individually replaced. The assemblies must be matched and adjusted at the factory. Therefore, if the HP E1431A fails, order the exchange module. However, selected hardware can be replaced if damaged. Replacement parts are listed in the following three tables:

- Module
- Covers
- Front Panel

Caution The module is static sensitive. Use the appropriate precautions when removing, handling, and installing to avoid unnecessary damage.

Ordering Information

To order a part listed in one of the tables, quote the Hewlett-Packard part number (HP Part Number), the check digit (CD), indicate the quantity required, and address the order to the nearest Hewlett-Packard sales and service office (see the inside back cover of this guide). The check digit verifies that an order has been transmitted correctly, ensuring accurate and timely processing of the order. The first time a part is listed in the table, the quantity column (Qty) lists the total quantity of the part used in the module. For the corresponding name and address of the manufacturers' codes shown in the tables, see "Code Numbers."

Direct Mail Order System

Within the U.S.A., Hewlett-Packard can supply parts through a direct mail order system. Advantages of the Direct Mail Order System are:

- Direct ordering and shipment from the HP Parts Center.
- No maximum or minimum on any mail order. There is a minimum order for parts ordered through a local HP sales and service office when the orders require billing and invoicing.
- Transportation charges are prepaid. A small handling charge is added to each order.
- No invoicing. A check or money order must accompany each order.
- Mail order forms and specific ordering information are available through your local Hewlett-Packard sales and service office. See the inside back cover of this guide for a list of Hewlett-Packard sales and service office locations and addresses.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Code Numbers

The following table provides the name and address for the manufacturers' code numbers (Mfr Code) listed in the replaceable parts tables.

Mfr No.	Mfr Name	Address
12085	Schlegel Corp.	Rochester, NY 14692 U.S.A.
28480	Hewlett-Packard Company	Palo Alto, CA 94304 U.S.A.
30817	Instrument Specialties Co. Inc.	Placentia, CA 92670 U.S.A.
83486	Elco Industries Inc.	Rockford, IL 61101 U.S.A.

Module

Caution Before installing the HP E1431A module into the VXI mainframe, be sure to set the mainframe power switch to standby (⏻) or remove power from the mainframe. Inserting or removing the module with power on can damage the module or mainframe.

Caution To protect circuits from static discharge, remove or replace modules only at static-protected work stations.

Ref Des	HP Part Number	CD	Qty	Description	Mfr Code	Mfr Part Number
	E1431-69201	3	1	HP E1431A EXCHANGE MODULE †	28480	E1431-69201

† Do the following when you replace the HP E1431A module:

1 Write the faulty module's serial number on the exchange module's blank serial number tag using a fine point permanent marker.

2 To program the serial number using RS-232 commands, do the following:
   1 Configure the system. See page 3-2.
   2 Select the E1431 terminal softkey.
   3 Type `diag:bser "nnnnAnnnnn"` and press Enter.
   4 Type *idn? and press Enter.

   The response should be "Hewlett-Packard, E1431A, *nnnnAnnnnn, A.nn.nn*."

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3 To program the serial number using HP-IB commands, do the following:

1 Configure the system. See page 3-4.

2 Type in and run the following program:

```
10 DIM Response$[80]
20 OUTPUT 70901;"diag:bser ""nnnnAnnnnn"""
30 OUTPUT 70901;"*opc?"
40 ENTER 70901;Dummy
50 OUTPUT 70901;"*idn?"
60 ENTER 70901;Response$
70 DISP Response$
80 END
```

The response should be "Hewlett-Packard, E1431A, *nnnnAnnnnn, A.nn.nn*."

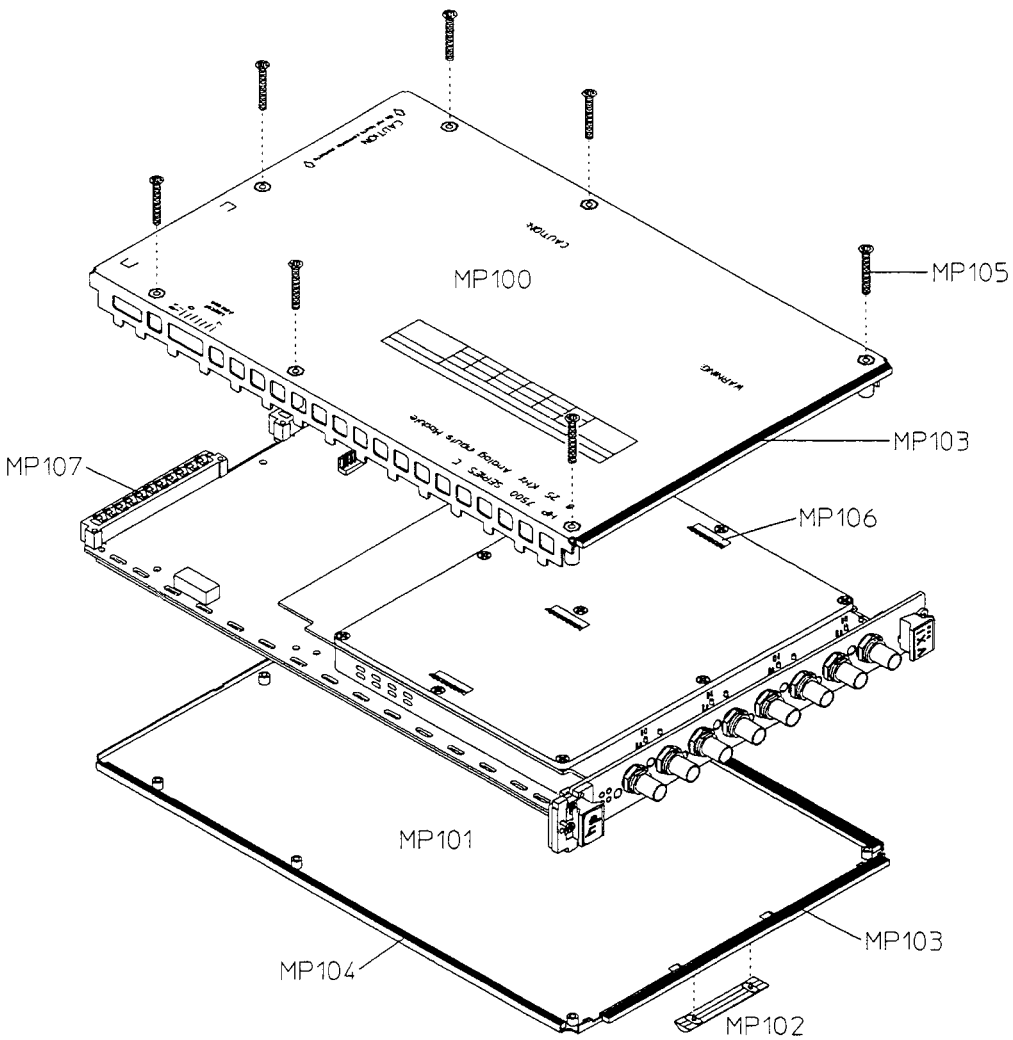
4 Write the faulty module's serial number on the exchange module's Certificate of Calibration.

5 Remove all customer labels from the faulty module and place on the exchange module.

6 Set the exchange module's logical address configuration switch to the faulty module's original logical address.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

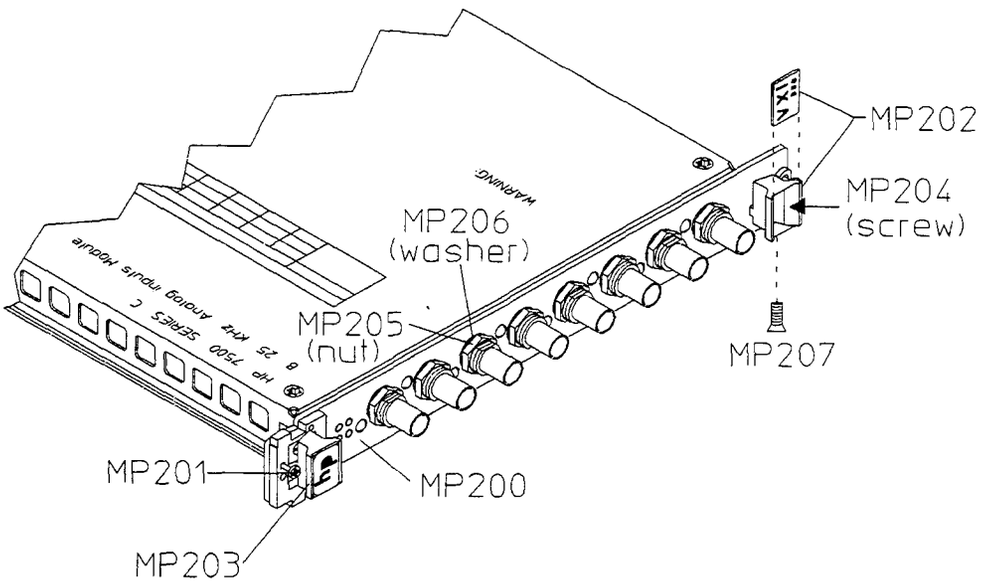
Covers



Ref Des	HP Part Number	CD	Qty	Description	Mfr Code	Mfr Part Number
MP100	E1431-00202	0	1	SHTF SHLD-TOP ALSK	28480	E1431-00202
MP101	E1431-00203	1	1	SHTF CVR-BTTM AL	28480	E1431-00203
MP102	8160-0686	6	1	STMP FNGRS-RFI STRP BECU	30817	786-185
MP103	E1485-40602	2	2	GSKT RFI-FRT PNL,ADH LG SD	12085	5774-191W-0
MP104	E1485-40601	1	2	GSKT-RFI,BTTM CVR ADH SHT SD	12085	5774-194W-0
MP105	0515-1135	7	7	SCREW-MACH M3 X 0.5 25MM-LG	28480	0515-1135
MP106	8160-0467	1	1	RFI STRIP-FINGERS BE-CU BRIGHT DIP	30817	97-555-A-X
MP107	E1450-01202	5	4	STMP SHLD-RFI GRND "VXI"	28480	E1450-01202

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Front Panel

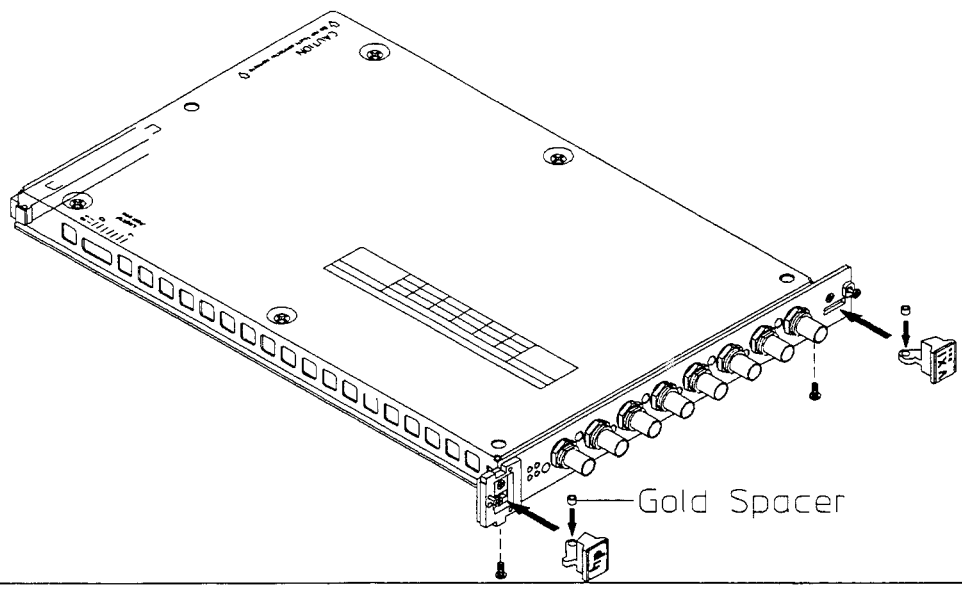


Ref Des	HP Part Number	CD	Qty	Description	Mfr Code	Mfr Part Number
MP200	E1431-00204	2	1	PNL-FRT"E1431A" VXI ALPT	28480	E1431-00204
MP201	0515-1968	4	2	SCR-MCH M2.5 11MMLG PHSPS SST	28480	0515-1968
MP202	E1400-84105	1	1	MOLD KIT-BTTM EXTR HDL"VXI"	28480	E1400-84105
MP203	E1400-84106	2	1	MOLD KIT-TOP EXTR HNDL"HP"	28480	E1400-84106
MP204	0515-1375	7	2	SCR-MCH M2.5 6MMLG FHTX SST	83486	343-300-02506
MP205	2950-0154	2	8	NUT-HEX-DBL-CHAM 1/2-28-THD .078-IN-THK	28480	2950-0154
MP206	2190-0068	5	8	WASHER-LK INTL T 1/2 IN .505-IN-ID	28480	2190-0068
MP207	0515-0368	6	2	SCREW-MACHINE ASSEMPLY M2.5 X 0.45	28480	0515-0368

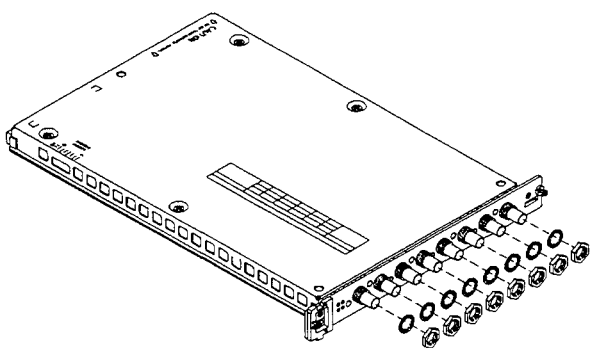
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To remove the front panel

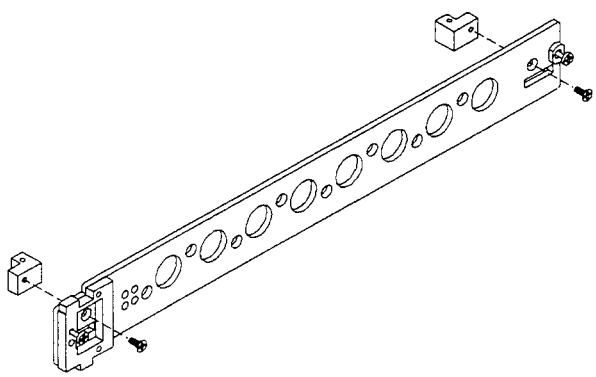
1 Using a T-8 torx driver, remove the screws that attach the handles. Be careful not to lose the gold spacers.



2 Using a 9/16-inch open-end wrench, remove the washers and nuts from the BNC connectors.



3 To replace the front panel, remove the side brackets using a T-8 torx driver.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5

Circuit Description

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Circuit Description

The HP E1431A 25.6 kHz 8 Channel VXI Input Module provides eight analog input channels in a single slot, C-size, register based, VXI module. The module supports the VME bus and an ECL-level local bus. A module can be used alone or with a VXI signal processing module. Each input channel is a true differential input that conditions input signals from dc to 25.6 kHz. Each input channel provides common and differential overload detection, alias protection, 16-bit analog-to-digital conversion, and a 4 mA, 28 V (nominal) floating ICP supply.

ICP Source — Supplies power to transducers such as accelerometers when enabled. The ICP Source is a 4.5 ±1 mA floating current source. When disabled, the source is disconnected by a relay and the switching power supply is disabled.

Common-Mode Overload Detection — Detects common mode overloads.

Differential Amp — Provides attenuators for highest ranges.

Range Selection — Provides a gain selectable amplifier. The full scale input range is from 5 mVpk to 10 Vpk.

Differential Overload Detection — Detects differential overloads.

Offset DAC — Provides dc offset compensation.

Analog Anti-Alias Filter — Provides alias protection up to 25.6 kHz.

Sample and Hold — Holds a voltage sample for the period of time required by the Oversampling ADC to digitize the voltage.

Oversampling ADC — Converts the input signal to a 16-bit digital word at the sample clock rate.

A/D Overload Detection — Detects overloads at its input.

Digital Filter and Decimation — Provides digital alias protection, sample rate reduction, and trigger detection. The trigger slope and level may be set to positive or negative. Pre- or post-trigger delay, which is computed digitally, can be defined for all eight channels.

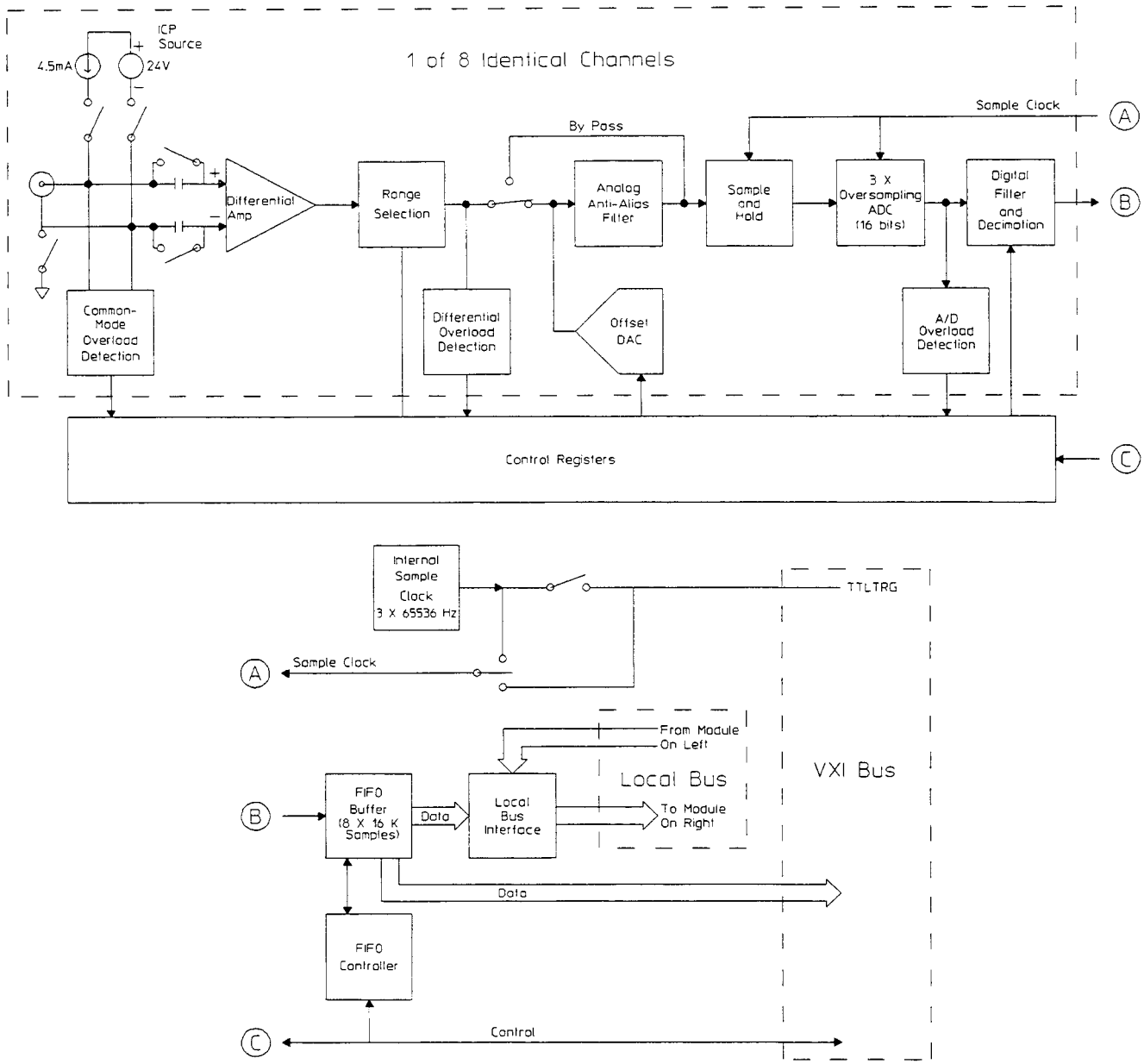
Control Registers — Provides the interface to the VXI bus, via registers.

Internal Sample Clock — Generates a 65.536 kHz clock with 3 times oversampling which results in a 196.608 kHz internal sample clock. The sample clock is either the internal sample clock or the external sample clock. The external sample clock's maximum frequency is 196.608 kHz. TTLTRIG inputs the external sample clock or outputs the internal sample clock.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FIFO Buffer/Controller Provides 16 K-samples for each input channel.

Local Bus Interface Provides high speed access to the local bus. Signals on the local bus are ECL-level.



HP E1431A Block Diagram

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6

Backdating

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Backdating

This chapter will document modules that differ from those currently being produced. This information will allow this guide to be modified so that it applies to any earlier version or configuration of the module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7

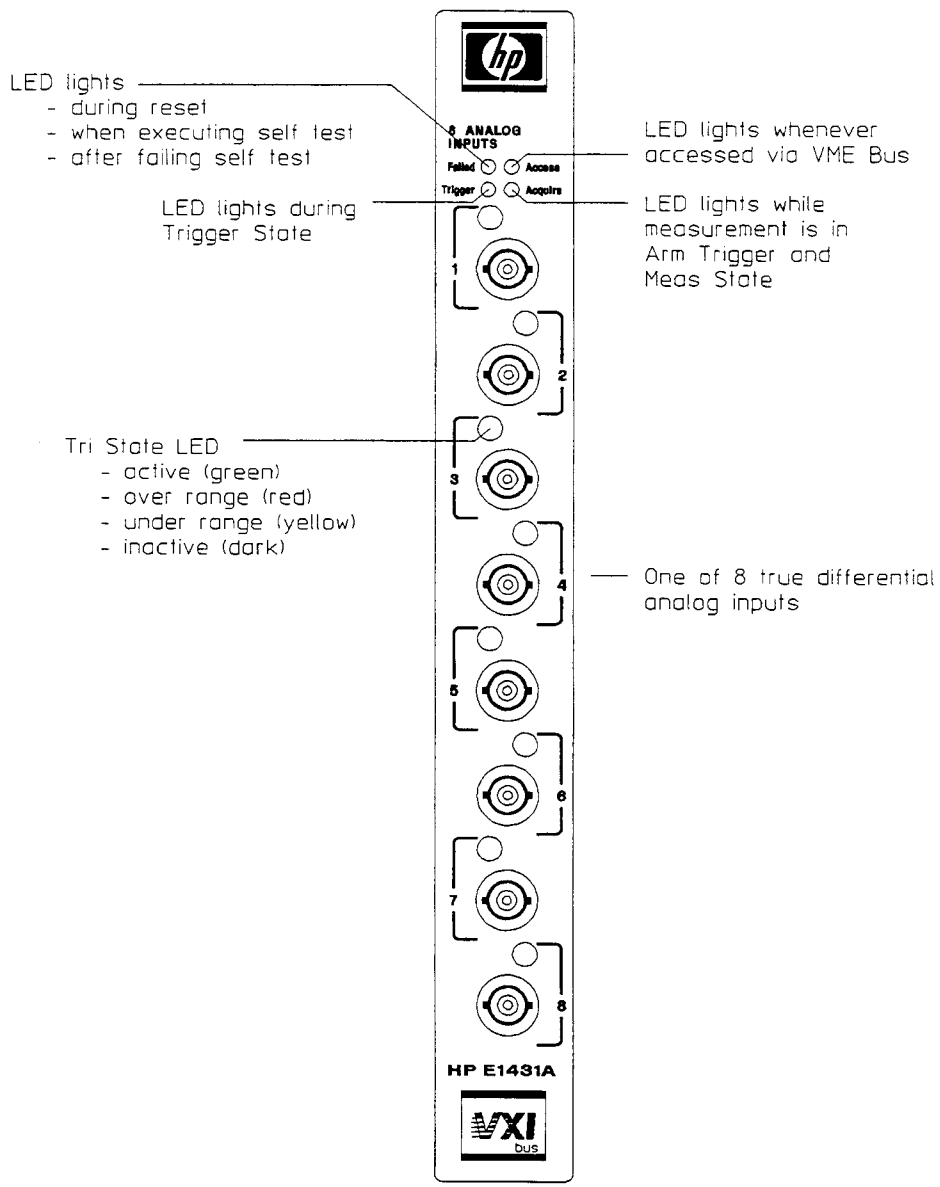
Using the HP E1431A

The HP E1431A is an eight-channel analog input module with built-in signal conditioning and anti-alias filters. Its 25.6 kHz bandwidth supports general vibration analysis in mechanical test and machine condition monitoring. When used with the HP E1485A/B signal processing module, the HP E1431A can perform computed order tracking for runup/rundown testing of piston engines, turbines, and other rotating machines.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Front Panel Description

The HP E1431A front panel contains the following:



The HP E1431A Front Panel

A set of hardware switches allow set-up of the VXI logical address of this module. See step 3 of "To Install the HP E1431A" in Chapter 1. There are no other configuration switches; trigger lines and interrupt lines are configured by software.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Input Mode: Voltage or ICP®

Voltage mode is used with accelerometers having external signal conditioning or in situations where signal conditioning does not need to be addressed.

ICP input mode provides power to accelerometers with internal signal conditioning. ICP mode supports a standard voltage (24 to 30 V) and current (≈ 4.5 mA ± 1mA). Input channels with ICP power can be either grounded or floating.

ICP transducers contribute a DC offset (roughly +10 volts to +15 volts), and consequently, products generally should be AC coupled when measuring these types of signals.

ICP is a registered trademark of Piezotronic Inc.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VXI Backplane Connections

Power Supplies and Ground

The HP E1431A conforms to the VME and VXI specifications for pin assignment. The current drawn from each supply is given in the specifications chapter.

Data Transfer Bus

The HP E1431A conforms to the VME and VXI specifications for pin assignment and protocol. Only A16/D16 data transfers are supported. Thus the upper address and data bits are ignored.

DTB Arbitration Bus

The HP E1431A module is not capable of requesting bus control. Thus it does not use the Arbitration bus. To conform to the VME and VXI specifications, it passes the bus lines through.

Priority Interrupt Bus

The HP E1431A generates interrupts by applying a programmable mask to its status bits. The priority of the interrupt is determined by the interrupt priority setting in the control register.

Utility Bus

The VME specification provides a set of lines collectively called the utility bus. Of these lines, the HP E1431A only uses the SYSRESET* line.

Pulling the SYSRESET* line low (a hardware reset) has the same effect as setting the reset bit in the Control Register (a software reset), with one exception--the Control Register itself does not reset.

Local Bus

The VXI specification includes a 12-wire Local Bus between adjacent module slots. Using the Local Bus, Hewlett-Packard has defined a standard byte-wide ECL protocol which can transfer data from left to right at up to 100 Mbytes/sec. The HP E1431A can be programmed to output its data using this high speed port instead of the VME data output register. The Data Port Control register determines which output port is used.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Trigger Lines

The VXI specification provides 8 TTL and 2 ECL trigger lines which can be used for module-specific signaling. There are four (4) pairs of TTLTRG lines: 01, 23, 45, and 67. When programmed in a multiple-module configuration, the HP E1431A uses one pair of the TTLTRG trigger-lines, designating one as a SYNC line and the other as the ADC sample clock (CLOCK). Only one module in a multiple module configuration can drive the same pair of trigger lines.

The CLOCK line is the master ADC clock for a synchronous system of multiple HP E1431A modules. Only one HP E1431A module in each mainframe is allowed to drive this line.

The SYNC line is used to send timing signals among HP E1431A modules in a multiple-module system. Any module which drives this line must do so synchronously with CLOCK so that transitions on SYNC do not occur near the rising edge of CLOCK. This ensures that all modules with a synchronous state machine clocked on CLOCK will interpret SYNC in a consistent manner for each cycle of the state machine. SYNC is used for synchronizing, arming, and triggering signals between HP E1431A modules. The interpretation of the SYNC line is dependent on the states of the module described in the Measurement Loop section which follows.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The HP E1431A's Measurement Process

Measurement Setup and Control

The measurement consists of two phases, the measurement initialization, and the measurement loop. Each of these phases consists of several states, through which the measurement progresses.

The transition from one state to the next is tied to a transition in the synchronization line, called the *SYNC line* (one of the TTLTRG lines on the VXI backplane). This SYNC line is "wired-ORed" such that all HP E1431As in a multiple-module system must release it for it to go high. Only one HP E1431A is required to pull the SYNC line low. In a single HP E1431A system, the SYNC line is local to the module and is not tied into a TTLTRG line on the VXI backplane.

State transitions depend on the OPCODE value (field of the control register) and are caused by changes in the module's Sync line which are synchronous with the sample clock. The control register OPCODES may be found in the Appendix. In multiple-module synchronous systems all the ADC clocks and SYNC lines are driven from the VXI backplane, allowing for synchronous state transitions among all modules.

It is easier to manage the measurement sequence by using the C Library functions or SCPI commands. However, you can force the measurement to enter the tested and idle states by writing to the control register (i.e. the OPCODE field). You may also drive the SYNC line low via the same register in order to force transitions from idle to arm, or from trigger to measure (Pull Sync bit). For more information, see "The VXI Registers" in Appendix A.

Be sure to perform an autozero operation and phase calibration after power up or as required by specifications (see Chapter 2). The autozero operation nulls the DC offset of the input amplifier of the ADC. Specifications for cross-channel time delay match and cross-mainframe phase accuracy are dependent upon calibrating the module with these operations.

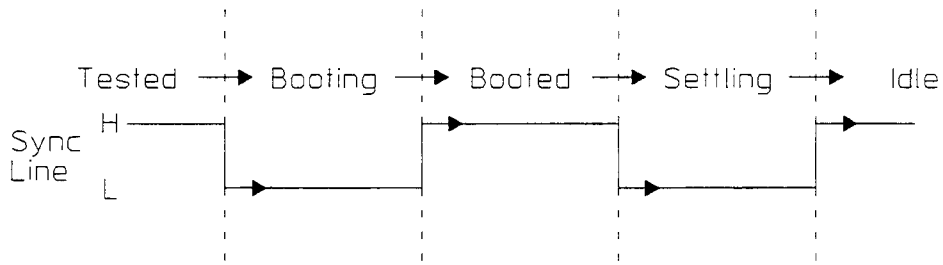
C Library Function Perform an autozero operation with a call to the function *e1431A_auto_zero*. You can perform an autozero operation and calibrate phase accuracy with a call to the function *e1431A_auto_zero_and phase*.

SCPI Perform an autozero operation with the INPut:OFFSet:AUTO command. You can perform an autozero operation and calibrate phase accuracy with the INPut:PHASe:AUTO command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Measurement Initialization

The measurement initialization states, and the corresponding SYNC line transitions are illustrated below:



Measurement Initialization States

The *tested state* is reached after a reset. In this state, all of the module parameters may be set. An HP E1431A stays in the *tested state* until it sees a high to low transition of the SYNC line.

In the *booting state*, the digital processors of the module load their parameters, and their program. Once done, the module releases the SYNC line. When all modules in a multiple-module system have released the SYNC line they all move to the *booted state*. An HP E1431A stays in the boot state until it sees a high-to-low transition of the SYNC line, meaning that all HP E1431As in the system have booted and the processors are to synchronize.

In the *settling* state, the digital filters are synchronized and the digital filter output is *settled* (for example, waiting N number of samples before outputting any data). Once the module is settled, it releases the SYNC line. When all modules in a multiple-module system have released the SYNC line, they all advance to the *idle* state.

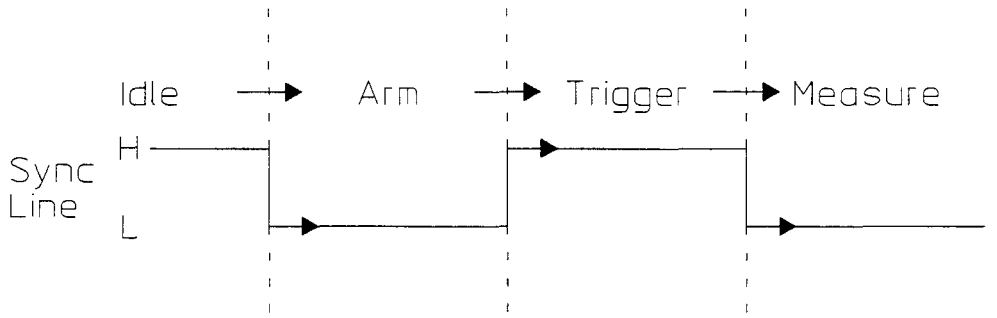
C Library Function This complete measurement sequence initialization, from tested, through booting, booted, settling and ultimately idle, can be performed with a call to the function *e1431A_init_measure*.

SCPI This complete measurement sequence initialization, from tested, through booting, booted, settling and ultimately idle, can be performed with the INITiate:IMMediate command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Measurement Loop

The progression of measurement loop states and the corresponding SYNC line transitions are:



Measurement Loop States

In the *idle state* there is not data in the FIFO. The HP E1431A does not write data into the FIFO. An HP E1431A remains in the idle state until it sees a high-to-low transition of the SYNC line.

Upon entering the *arm state* the HP E1431A starts saving new data in its FIFO. It remains in the arm state until the SYNC signal goes high. If an HP E1431A is programmed with a pre-trigger delay, it collects enough data samples to satisfy this pre-trigger delay, and then releases the SYNC line. If no pre-trigger delay has been programmed, the SYNC line goes high immediately. When all HP E1431As in a system have released the SYNC line allowing it to go high, a transition to the *trigger state* occurs.

Upon entering the *trigger state* an HP E1431A continues collecting data into the FIFO, discarding any data prior to the pre-trigger delay. An HP E1431A remains in the trigger state until it sees a high-to-low transition of the SYNC line. The SYNC line is pulled low by any HP E1431A which encounters a trigger condition and is programmed to pull the SYNC line.

Upon entering the *measure state* an HP E1431A continues to collect data. The HP E1431A holds the SYNC line low as long as it is actively collecting data, or has data to read in FIFO. After the block of data has been acquired the HP E1431A presents the first data to the selected output port.

In *block mode*, the HP E1431A stops taking data as soon as a block of data has been collected, including any programmed pre- or post-trigger delays. After the FIFO has been emptied, the HP E1431A releases the SYNC line (only in block mode). When all HP E1431As are finished and the SYNC line goes high, the HP E1431A goes into the *idle state* again.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

In *continuous mode*, the HP E1431A does not stop taking data, unless the FIFO overflows. The intention is that data will be read fast enough from the FIFO to prevent FIFO overflow. When data collection stops, the HP E1431A releases the SYNC line. When all HP E1431As are finished and the SYNC line goes high, the HP E1431 goes into the idle state again.

C Library Functions

Auto-arming / Arming

To program any of the HP E1431As for auto-arming, use the function, *e1431_set_auto_arm*. With auto-arming, the SYNC line immediately goes low, exiting the idle state to the arm state. The HP E1431A may also be moved to the arm state by an explicit call to the function, *e1431_arm_measure*. If the function is called in a multiple-module system, all modules go to the arm state.

Auto-triggering / Triggering

To program any HP E1431A for auto-triggering, use the *e1431_set_auto_trigger*. With auto-triggering, the SYNC line goes low immediately, exiting the trigger state. The SYNC line may also be pulled low by an explicit call to the function, *e1431_trigger_measure*. If the function is called in a multiple-module system, all modules go to the trigger state.

Interrupting a measurement

The measurement initialization and loop may be interrupted at any time with a call to *e1431_init_measure*, *e1431_reset_measure*, or *e1431_finish_measure*. The measurement resets to the idle state.

SCPI

Auto-arming / Arming

To program any of the HP E1431As for auto-arming, use the ARM:SOURce IMMediate command. With auto-arming, the SYNC line immediately goes low, exiting the idle state to the arm state.

Free-run triggering / Triggering

To program any HP E1431A for free-run triggering (also called auto-triggering), use the TRIGger:SOURce IMMediate command. With free-run triggering, the SYNC line goes low immediately, exiting the trigger state.

Interrupting a measurement

The measurement initialization and loop may be interrupted at any time with the INITialize[:IMMediate] command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Transferring Data

You can transfer data from the HP E1431A two different ways; via the VMEbus or via the Local Bus.

The VMEbus is the universal data bus for VXI architecture. It provides flexibility and versatility in transferring data. Transfers over the VMEbus can be 16 or 32 bits wide.

The Local Bus supports faster transfer rates than the VMEbus. For example, if you are transferring data from the HP E1431A to the HP E1485A/B, the Local Bus provides a direct pipeline to the HP E1485's DSPs.

In addition, the Local Bus allows you to transfer data from a maximum of 5 HP E1431A card sets at full span in real time. The maximum transfer rate is 2.62 M samples per second.

Using the Local Bus, you can transfer data in the background while processing data in a signal-processing module.

All Local Bus data-transfers, originate in the HP E1431A and move towards a signal processing module to the right of the HP E1431A. If other modules generate data to the left of the input module, the HP E1431A will pass the data to its right and append its own data at the end of the frame.

C Library Function

Use the *e1431_set_data_port* to specify which bus to transmit data.

SCPI

Use the VINStrument[:CONFigure]:PORT command to specify which bus to transmit data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Data Formats

You must specify the data format to be used in all data transfers. The type of format depends of the programming environment you are using. The specified data format applies to all eight channels.

C Library Function

Use the *e1431_read_raw_data* to specify unscaled data, 32-bit floating point data or 64-bit floating point data. Maximum block size is 16K real samples per channel. Minimum block size is one sample.

SCPI

Use the DATA:FORMAT command to specify ascii, real or packed data. The maximum block size is 16384 samples.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Using the Sample Clock

The HP E1431A uses an oversampling ADC to minimize the effects of the analog anti-alias filter on channel phase. This oversampling requires that the ADC sample clock be 3 times the desired effective sample rate. A digital decimation filter reduces the 3 times oversampled data rate to the effective sample rate. This 3 times oversampling is completely transparent if you are using the HP E1431A's internal sample clock. Its fixed 196.608 kHz sample rate is decreased to the 65,536 sample rate by the digital decimation filter.

Digital filters are used to further reduce the frequency spans. Digital filters can reduce the frequency spans from 25.6 kHz down to 0.39 kHz in factor-of-two steps. Frequency spans and effective sample rates are related as follows:

frequency span = effective sample rate / 2.56

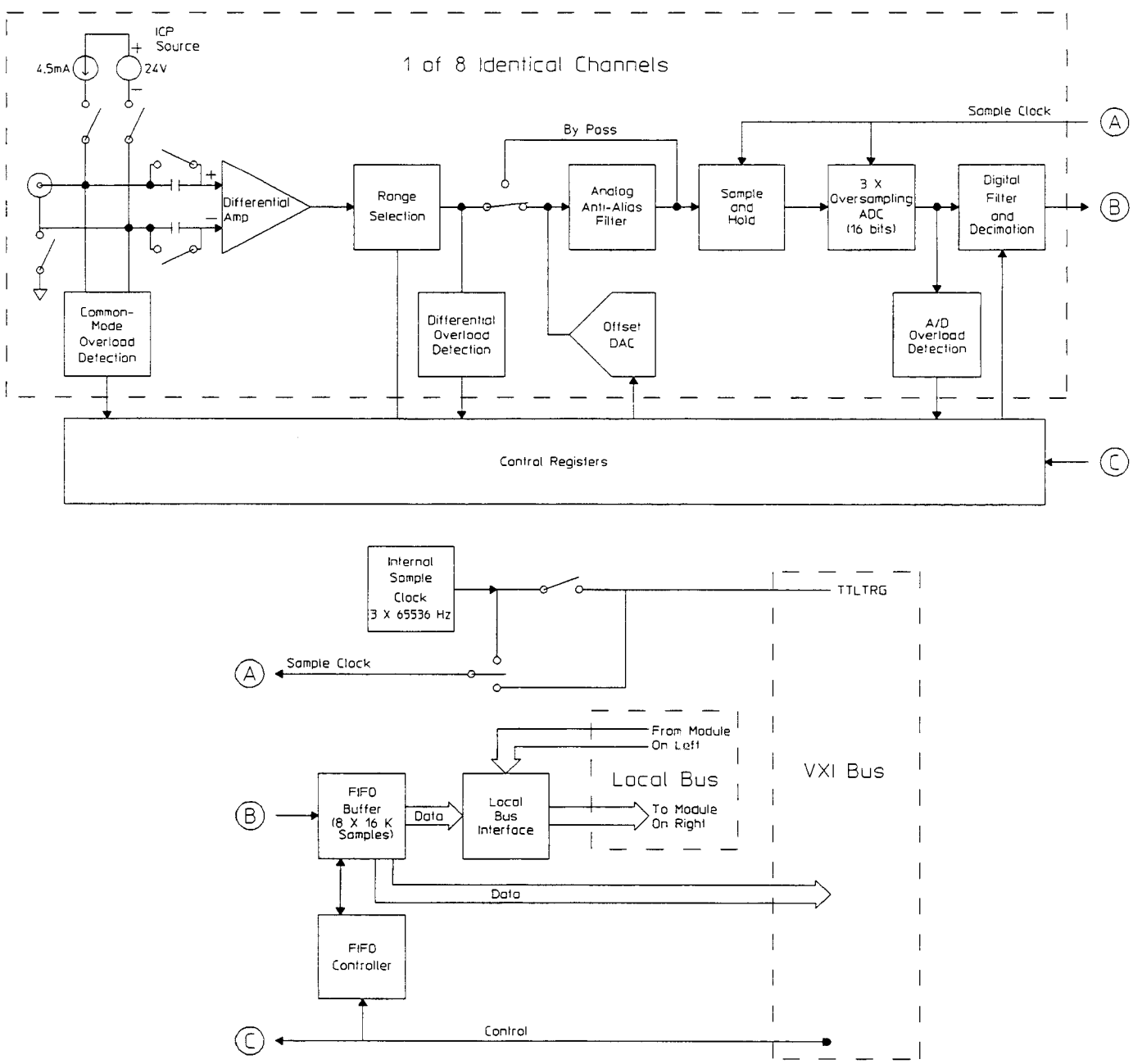
Providing an External Sample Clock

You can program the HP E1431A to accept an external sample clock from the VXI bus (TTLTRG line). If you are providing an external sample clock, its rate must be at 3 times the desired effective sample rate. For example, to achieve an effective sample rate of 50 kHz, a 150 kHz sample rate must be supplied on TTLTRG. The digital filters are still functional, providing effective sample rates from 50 kHz down to 0.76 Hz in factor-of-two steps. All sampling is done simultaneously and is not multiplexed. See the block diagram on the next page.

Connecting an External Sample Clock

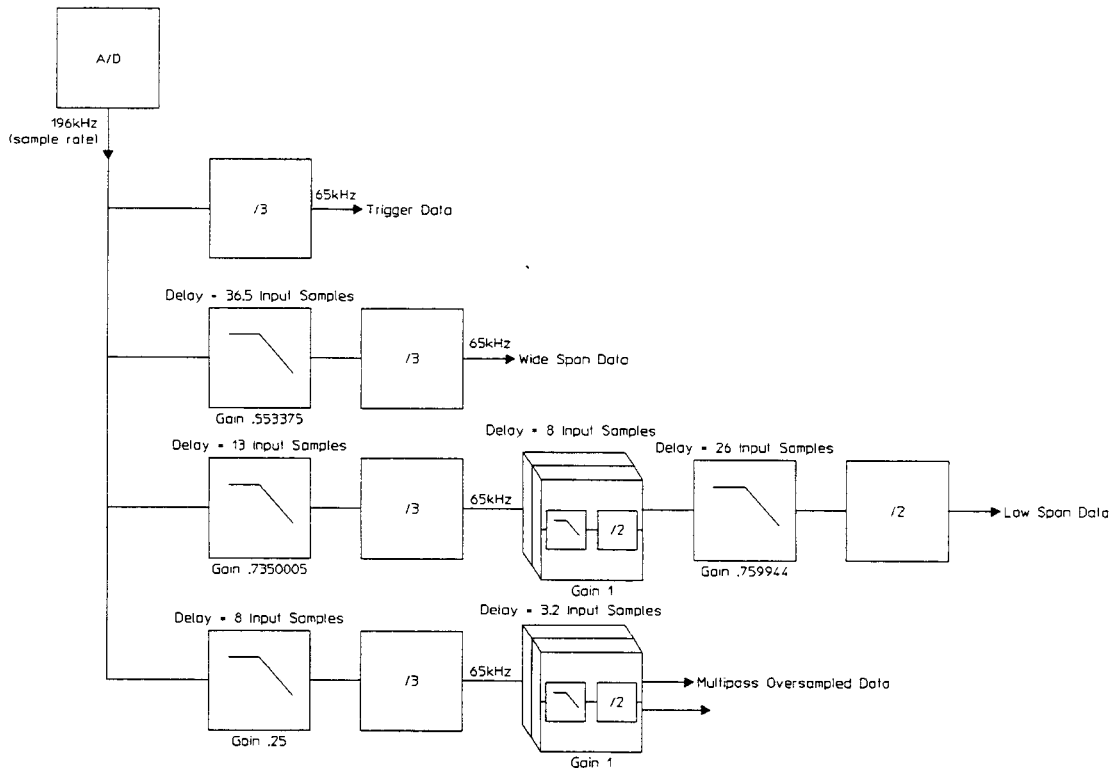
Use the trigger input (TRIG IN) on the VXI slot 0 controller to input the external sample clock.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



HP E1431A Block Diagram

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



Decimation Filter Block Diagram

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Replaying Data

Replay is the re-processing of previously-captured-throughput data It is a continuous operation with no trigger detection or delays. Setting parameters is the same as in real-time operation. The measurement loop state is the same *except for the measure state*.

In replay, the HP E1431A goes through the idle state, arm state and trigger state as it does in real-time processing. (See "Measurement Loop" earlier in this chapter for more information about the measurement loop states.) In replay, however, the DSPs read data from RAM and reprocess the data accoring to the span and multi-pass setup. The output is written to the FIFO as in real-time measurements.

In replay, you can perform a zoom operation. Sometimes called band selectable analysis, zooming reduces the frequency span while maintaining a constant center frequency. This allows you to select a frequency span around a specified center frequency so you can focus on a specific frequency band.

For additional information, see the C Library functions, *e1431_replay_data* and *e1431_init_replay*, in Chapter 9.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8

Programming the HP E1431A with the C Interface Libraries

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The C Interface Libraries for the HP E1431A are a set of functions that allow you to program the register-based HP E1431A at a higher level than register reads and writes. The libraries allow groups of HP E1431As to be set up and programmed as if they were one entity. The current state of each HP E1431A in a system is maintained in your host computer because you cannot read from many of the HP E 1431A's registers you can write to. These states can be saved and restored. The libraries include routines to perform auto-ranging and autozeroing, routines to aid debugging and hardware diagnostic routines. In addition, there are low-level routines to allow direct register access, but with the added protection of bus error trapping.

The libraries are designed to work both in the Series 300 and Series 700 UNIX environment, in the Series 700 UNIX/MXI environment, and in the HP E1485A/B environment. *The library must be used in one environment or the other, but not both at the same time*. This restriction is necessary since the libraries maintain internal state information about the HP E1431A that could become corrupted if two sources are changing the HP E 1431A's registers.

UNIX is a registered trademark of UNIX System Laboratories Inc. in the U.S.A. and other countries

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Getting Started

For instructions on how to install the C Interface Libraries, see Chapter 1, "Installing the HP E1431A."

Compiling and Linking Your Program

How you compile and link the libraries to your program depends on the operating environment.

HP-UX Environment

The following listing is an example of a makefile for the HP-UX environment. The program requires an HP E1431A at address 16 in a VXI mainframe controlled by an embedded computer or MXI computer.

```
----------------- example makefile ----------------------------------
E1485_LIB=      /usr/e1485/lib
E1485_INC=      /usr/e1485/include
E1431_LIB=      /usr/e1431/lib
E1431_INC=      /usr/e1431/include
CC=               cc
HPUX_CFLAGS =    -D_HPUX_SOURCE -c -Aa -O
example : $(E1431_LIB)/lib1431.a example.o
       $(CC) example.o $(E1431_LIB)/lib1431.a -lsicl -lc -lm -o example
example.o : example.c
       $(CC) $(HPUX_CFLAGS) -I $(E1431_INC) -I $(E1485_INC) example.c
-------------------------------------------------------------
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

HP E1485A/B Environment

The following listing is an example of a makefile for the HP E1485A/B environment:

```
----------------- example makefile ----------------------------------
E1485_LIB=      /usr/e1485/lib
E1485_INC=      /usr/e1485/include
E1431_LIB=      /usr/e1431/lib
E1431_INC=      /usr/e1431/include
CC=               cc
HPUX_CFLAGS =    -D_HPUX_SOURCE -D_DEBUG_ -D__STDC__ -c -Aa -O
DLD_CFLAGS=    -Aa -c -O -W2,-x -DE1485_SOURCE $(EFLAG)       -I $(E1485_INC) \
  -I $(E1431_INC)
DLD_LIBS=      $(E1431_LIB)/libd1431.a
all:      example example.dld
### series 300 and 700 unix build
example.o : example.c $(E1431_INC)/e1431.h $(CC) $(HPUX_CFLAGS) -I       $(E1431_INC) -I
$(E1485_INC) example.cexample   :  $(E1431_LIB)/lib1431.a
example.o     $(CC) example.o $(E1431_LIB)/lib1431.a -lsicl -lc -lm -o example
### E1485 build
example.dld:  dexample.o
      ld -dr -N $(E1485_LIB)/libspil.o dexample.o $(DLD_LIBS) \
      -o example.dld
      nm -u example.dld
      cp example.dld ../scope
dexample.o:      example.c $(E1431_LIB)
      cc $(DLD_CFLAGS) example.c -o dexample.o
-------------------------------------------------------------------- */
```

Debugging

There are several levels of debugging aids provided with the libraries. First, you should check the return value of all functions. Usually, a non-zero value denotes an error.

The *e1431_print_errors* function can be called to enable/disable an error printing mechanism. If error printing is enabled, an error message will be printed by any function returning an error. If the libraries are used in a host computer environment, the errors are output to stdout (normally the console screen). If the libraries are used in the HP E1485A/B environment, the error messages will be output to a terminal connected to the RS-232-C port available on this module. It is normal while developing code to include a call to *e1431_print_errors* enabling error printing early in the code. Once the code has been fully debugged, you should to remove this call.

Another debugging method is *call tracing*, which is controlled with *e1431_trace_level*. This mechanism prints a line of information each time a library function is entered and displays the parameters of the call, which can often indicate errors.

There are functions for dumping individual registers or the complete state of a group of HP E1431As in an easy-to-read format. See "Debugging Your Program" in the C Libraries Quick Reference which appears at the end of this chapter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The function, *e1431_debug_level*, is used to print out a message each time a register write occurs to the HP E1431A. The message includes the register number and new contents being written. This function and the material in Appendix A, "The VXI Registers," allows detailed examination of sequence of register writes as well as the contents of the HP E1431A registers at the bit level.

Parameter Settings

Many parameters are channel-dependent, meaning each channel can be set independently. Other parameters are module-dependent; *changing a module-dependent parameter for a channel will change it for all channels in that module*. For example, changing *blocksize*, a module-dependent parameter, for channel 3 will also change the blocksize for channels 1 through 8.

All parameters settings are stored in RAM in module structures until the *e1431_init_measure* function is called. At that time, the settings in RAM direct the set up of the modules. Parameters can be changed while a measurement is running, but the new settings will not be taken into account until the next call to *e1431_init_measure*. Some calls that read the data from the HP E1431A, use the RAM settings to interpret that data, so it is not wise to change parameter values while acquiring data.

Grouping of channels / modules

The interface libraries for the HP E1431A are designed to allow programming of several channels in one or several distinct modules, as if they were one entity. Each HP E1431A module has eight (8) channels, and the libraries may control up to a maximum of 255 HP E1431As or 2,040 channels.

When initializing the interface libraries, all module logical addresses are passed in the call to *e1431_assign_channel_numbers*, in order to associate a unique module identifier, *modID*, and eight unique channel identifiers, *chanIDs* to each of the HP E1431As in the VXI system. From there on, library functions use these identifiers rather than the logical address.

Any number of groups of channels, possibly overlapping, may be declared, and uniquely identified by a *groupID*, through a call to *e1431_create_channel_group*.

The "target" of a library function is either a channel, a group, or more seldom a module, depending upon the nature of the call. When the same library function may be called with either a channel or a group identifier, it is shown by a parameter named *ID*.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Programming Multiple Channels

A channel group that spans more than one module has to be configured to use the TTL trigger lines on the VXI bus for inter-module communications. The *e1431_init_measure()* call automatically configures the channel group unless you defeat the configuration *e1431_set_auto_group_meas()* function.

The following outlines what *init_measure()* does automatically. It also details what you must do if you are using the *e1431_set_auto_group_meas()* call to bypass auto-configuration.

There are four (4) pairs of VXIBUS TTL lines that can be used for multi-module synchronization. The pair is selected using the *e1431_set_ttltrg_lines()* function call. Calling *e1431_set_clock_source()* with the group ID will set all modules to the same pair.

All modules need to be set to use the shared SYNC line rather than the default setting of internal SYNC line. This can be done with the *e1431_set_multi_sync()* function call again using the group ID.

All modules need to be set to use the VXIBUS TTL lines as the clock source. To use the VXIBUS TTL lines as the clock source, use the *e1431_set_clock_source()* function call.

The system module needs to be set to output the clock with the *e1431_set_clock_master()* function call.

All system SYNC pulses will now come from the designated system module and will drive the measurement state machines on all boards in the group.

Warning

If active modules are left after a completed measurement, conflicting conditions may result.

For example, "module A" is driving the TTL trigger lines. A different group is started which also drives the TTL trigger lines, but this group does not include "module A." "Module A" will conflict and prevent the different group from functioning. In this case, a call to *e1431_finish_measure* (old group ID with A) will turn off module A, allowing the new group to function.

If the new group includes all modules of the old group, the conflict will not occur since *e1431_init_measure()* will reset all modules as needed. Single module groups do not drive the TTL trigger lines, therefore single modules groups do not experience this conflict.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

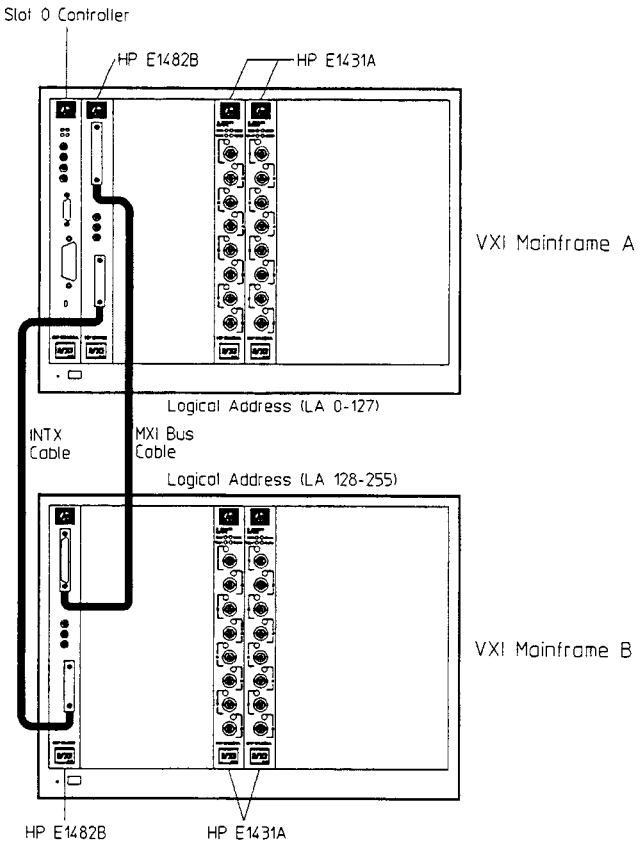
Managing Multiple Mainframe Measurements

In a single mainframe measurement, the HP E1431A communicates with other HP E1431As through the TTLTRG lines. However, when using the VXI-MXI bus extender modules, the TTLTRG lines, which carry the group synchronization pulse and sample clock, are extended only in one direction. This unidirectional signal connection restricts the types of measurements you can make in a multiple mainframe environment.

You cannot perform the following types of multiple mainframe measurements:

- unequal pre-trigger delay settings between mainframes
- channel triggering by channels in Mainframe B
- lower spans or longer blocksizes in Mainframe B
- different digital filter settling times between HP E1431A modules

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



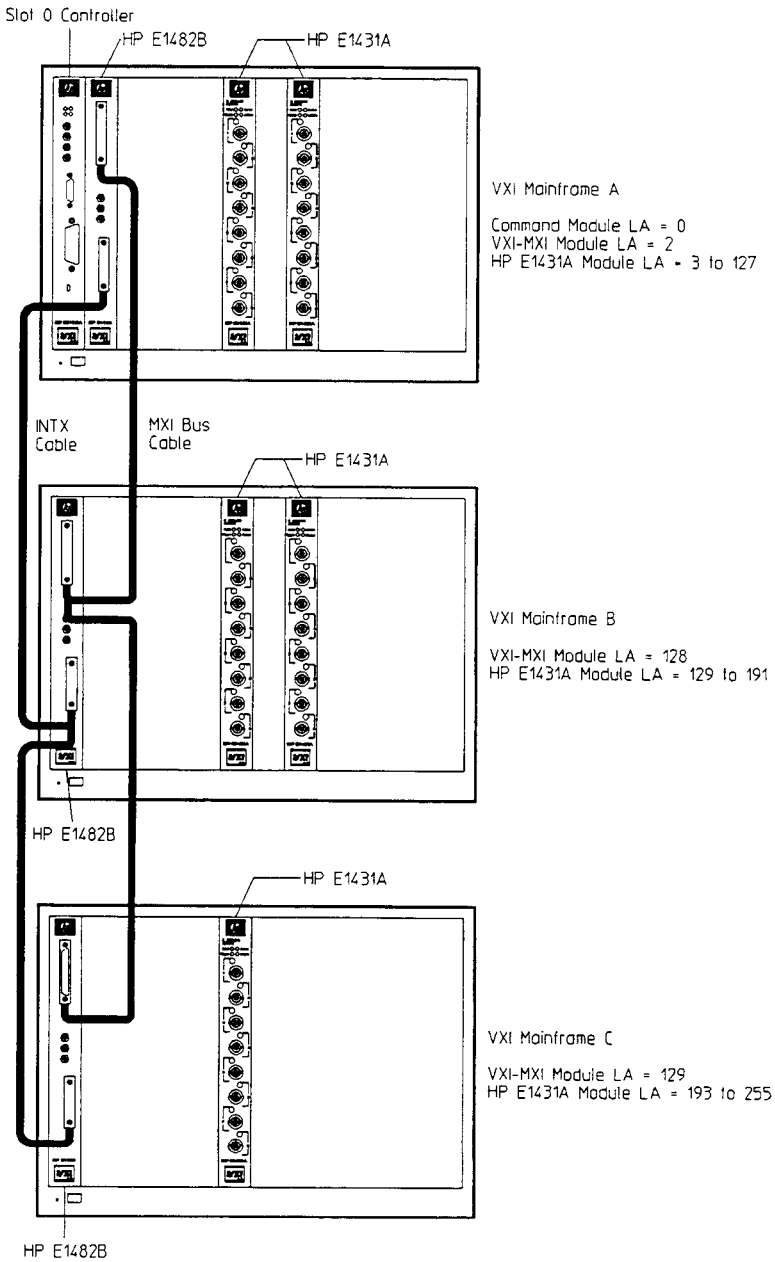
In the example above, Mainframe A contains the Slot 0 Controller for a multiple mainframe system. Mainframe A is connected to Mainframe B with a VXI-MXI interface, HP E1482B. To successfully manage this multiple mainframe environment, use the following guidelines.

- Locate modules with logical addresses less than 128 in Mainframe A.
- Locate modules with logical addresses greater than 127 in Mainframe B.
- Locate the highest-numbered channels in Mainframe A.
- Locate the last module in the module list specified in the call to *e1431_assign_channels()* in Mainframe A.
- Locate the module that generates the group synchronization pulse in Mainframe A.
- Locate the channels performing channel triggering in Mainframe A.
- Locate the module with the shared sample clock in Mainframe A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

- If you do not use a groupID with the call *e1431_read_data()*, you must empty the HP E1431As' FIFOs in Mainframe B before Mainframe A. In other words, do not empty the FIFOs in Mainframe A unless you have emptied the FIFOs in Mainframe B. For more information about groupID see "Grouping of Channels/Modules" on page 8-.5

- If more than two mainframes are needed, daisy-chain them together. Treat each mainframe after the first as a Mainframe B. See the example below.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Phase Performance in Multiple Mainframe Measurements

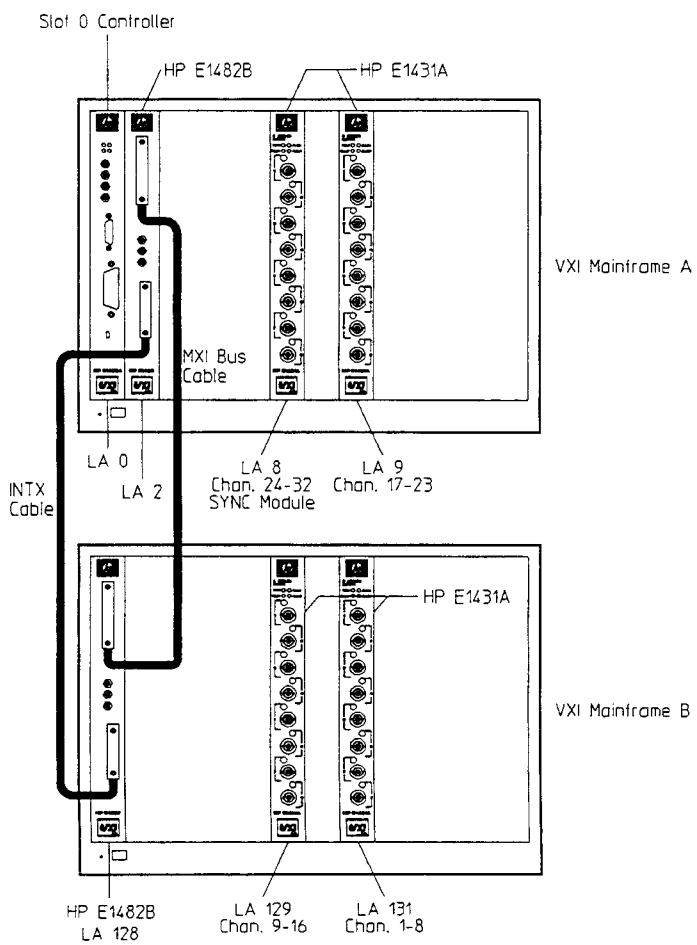
Phase specifications are degraded by the delay that the inter-mainframe interface gives the sample clock. This delay is insignificant for many low-frequency applications because the phase error is proportional to frequency.

A system with two VXI-MXI modules and a 1 meter cable, typically has a 76 nanosecond (ns) sample clock delay in Mainframe B. This corresponds to an additional 0.007 degree phase error at 256 Hz and an additional 0.7 degree phase error at 25.6 kHz.

A 4 meter cable adds approximately 18 ns of delay for a total of 94 ns clock delay in Mainframe B. This corresponds to an additional 0.0087 degree phase error at 256 Hz and an additional 0.87 degree phase error at 25.6 kHz.

The cable adds approximately 6 ns per meter of cable.

Each daisy-chained mainframe adds another increment of delay, but only for the additional cabling length.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Synchronization in Multiple Mainframe Measurements

A TTLTrigger line between HP E1431As making group measurements keeps all modules synchronized. This is an open-collector line where each module holds the one designated as the SYNC line low until the module is ready to advance to the next measurement state. Another TTL Trigger line is designated to carry the sample clock to all modules. This shared sample clock may come from any HP E1431A module in Mainframe A or from an external signal routed through the Slot 0 Commander in Mainframe A.

One module is responsible for pulling the SYNC line low to start each group's state transition. Then, each module holds the line low until it is ready. When all modules are ready, the SYNC line drifts high. The unidirectional line prevents modules in Mainframe B from holding-off modules in Mainframe A.

The lowest logical address must be in Mainframe A because of VXI-MXI and Resource Manager (RM) constraints. Group constraints with the C-Library force modules in Mainframe A to have their FIFOs emptied last. The C-Library reads data in channel order, so the highest channel is read last. To get this to work automatically, the call to *e1431_assign_channels()* must list the logical addresses in descending order.

Channel triggering must be done only by modules in Mainframe A. A trigger in any other mainframe would not be communicated back on the SYNC line to Mainframe A. The C-Library itself selects the HP E1431A with the highest channel number for synchronization.

VXI-MXI Module Setup and System Configuration

To setup your multiple mainframe system, follow the "Hardware Installation Rules" which appear in Chapter 2 of the *HP E1482B VXI-MXI Bus Extender User's Manual*. This allows the Resource Manager to configure your system.

The VXI-MXI Module setup in Mainframe A needs to be changed from those set by the factory. The VXI-MXI module is not the Slot 0 Controller for Mainframe A. See "Table 2-1. Configuration Settings" in the *HP E1482B VXI-MXI Bus Extender User's Manual*. This requires changing several switch settings.

- Set the module as not being the Slot 0 Controller.
- Set the VME timeout to 200 μs.
- Set the VME BTO chain position to 1 extenter, non-slot0.
- Do not source CLK10.
- Set the proper logical address.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The VXI-MXI module is the Slot 0 Controller for Mainframe B. You must set the Logical Address for each VXI-MXI module. For modules not at the ends of the daisy chain, remove six VME terminators and 4 INTX terminators.

See "Hardware Installation Rules" which appears in Chapter 2 of the *HP E1482B VXI-MXI Bus Extender User's Manual* for additional information.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C Libraries Example Program 1

```
/* C Libraries example program *
 * This example program REQUIRES an E1431 at address 16 in
 * a VXI mainframe controlled by an embedded computer or MXI computer.
 *
 * This sample program creates a group containing two channels
 * and makes a block measurement with the group. *
----------------- example makefile ---------------------------------
E1485_LIB=     /usr/e1485/lib
E1485_INC=     /usr/e1485/include
E1431_LIB=     /usr/e1431/lib
E1431_INC=     /usr/e1431/include
CC=              cc
HPUX_CFLAGS =   -D_HPUX_SOURCE -c -Aa -O
example : $(E1431_LIB)/lib1431.a example.o
      $(CC) example.o $(E1431_LIB)/lib1431.a -lsicl -lc -lm -o example
example.o : example.c
      $(CC) $(HPUX_CFLAGS) -I $(E1431_INC) -I $(E1485_INC) example.c
------------------------------------------------------------------------ */
#include <stdio.h>
#include "e1431.h"
#include "machType.h"
#define RANGE         10.0          /* 10 volt range */
#define BLOCKSIZE      16           /* 16 points per block */
#define NUM_CHANS       2           /* two channels in group */
main()
{
SHORTSIZ16 modulelist[] = { 16 };
SHORTSIZ16 channellist[] = { 1,4 };
SHORTSIZ16 groupon;
SHORTSIZ16 error;
LONGSIZ32 actualsize;
SHORTSIZ16 buffer[NUM_CHANS][BLOCKSIZE];
SHORTSIZ16 i,b;
FLOATSIZ64 scale;

/* initialize the library */
 error=e1431_init_io_driver();
 if(error) printf("e1431_init_io_driver() failed, error: %d\n", error);

/* turn on debugging prints, this is a good idea while developing  */
 e1431_print_errors(1);

/* change this 0 to 1 to see call tracing */
 e1431_trace_level(0);

/* assign channels to modules */
 error=e1431_assign_channel_numbers(1,modulelist);
if(error) printf("e1431_assign_channel_numbers() failed,error:
%d\n",error);

/* get groupID of a two channel group */
 groupon=e1431_create_channel_group(NUM_CHANS,channellist);
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* check for bogus groupID */
 if(groupon=0) printf("e1431_create_channel_group() returns
error:%d\n",groupon);
/* set up parameters for reading a block */
 error=e1431_set_analog_input(groupon,E1431_INPUT_MODE_VOLT,
                  E1431_INPUT_SOURCE_BNC,
                  E1431_ANTI_ALIAS_ANALOG_ON,
                  E1431_INPUT_GROUNDED,
                  E1431_COUPLING_DC,
                  RANGE);
 if(error) printf("e1431_set_analog_input() failed, error: %d\n",
error);

/* no trigger needed for this example */
 error=e1431_set_trigger(groupon,E1431_CHANNEL_OFF,
                  0.0,0.0,0.0,E1431_TRIGGER_SLOPE_POS,
                  E1431_TRIGGER_MODE_LEVEL);
 if(error) printf("e1431_set_trigger() failed, error: %d\n", error);
 error=e1431_set_data_format(groupon,BLOCKSIZE,
                  E1431_DATA_SIZE_16,
                  E1431_BLOCK_MODE,
                  E1431_APPEND_STATUS_OFF);
if(error) printf("e1431_set_data_format() failed, error: %d\n", error);
 error=e1431_set_data_port(groupon,E1431_SEND_PORT_VME);
if(error) printf("e1431_set_data_port() failed, error: %d\n", error);

/* top span alias protected data */
 error=e1431_set_decimation_filter(groupon,
                  E1431_ONEPASS,
                  E1431_ANTI_ALIAS_DIGITAL_ON,
                  0);
if(error) printf("e1431_set_decimation_filter() failed, error:%d\n",
error);

/* set the span of the measurement */
/* this was also done by e1431_set_decimation_filter above, but is  */

/* done here to demoonstrate e1431_set_span() */
 error=e1431_set_span(groupon,25600.0);
 if(error) printf("e1431_set_span() failed, error:%d\n", error);

/* program will trigger and arm the measurement */
 error=e1431_set_auto_trigger(groupon,E1431_MANUAL_TRIGGER);
 if(error) printf("e1431_auto_trigger() failed, error: %d\n", error);
 error=e1431_set_auto_arm(groupon,E1431_MANUAL_ARM);
 if(error) printf("e1431_auto_arm() failed, error: %d\n", error);

/* prepare for collecting data */
 error=e1431_init_measure(groupon);
 if(error) printf("e1431_init_measure() failed, error: %d\n", error);

/* ARM */
 error=e1431_arm_measure(groupon);
 if(error) printf("e1431_arm_measure() failed, error: %d\n", error);

/* TRIGGER */
 error=e1431_trigger_measure(groupon);
 if(error) printf("e1431_trigger_measure() failed, error: %d\n",
error);
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* wait for group block ready */
 while(!(error=e1431_block_available(groupon)));
 ;
 if(error) printf("e1431_block_available() failed, error: %d\n",
error);

/* block available flag true means the all channels have a block ready
*/
/* read all blocks of data by using the group ID */
 error=e1431_read_raw_data(groupon,
                  buffer[0],      /* start of 2 channel buffer */
                  2*BLOCKSIZE,
                  &actualsize);
 if(error) printf("e1431_read_raw_data() failed, error: %d\n",error);
/* report if a full block wasn't read */
 if(actualsize!=BLOCKSIZE*NUM_CHANS) printf("group read acutalsize=%d
BLOCKSIZE=%d\n",
                  actualsize,BLOCKSIZE);

/* get the voltage scale factor for the raw data */
/* see e1431_read_float32_data() for an easier way to do this */
 error=e1431_get_scale(groupon,&scale);
 if(error) printf("e1431_get_scale() failed, error: %d\n", error);

/* print the collected data */
/* print header */
 printf("point ");
 for(i=0;i <NUM_CHANS;i++)
 {
                  printf("chan %1d          ",channellist[i]);
 }
 printf("\n");

/* print data */
 for(b=0;b<BLOCKSIZE;b++)
 {
     printf("%3d - ",b);
     for(i=0;i<NUM_CHANS;i++)
     {
          printf("0x%04x  %8.5f ",
          0xffff&(buffer[i][b]),
          (FLOATSIZ32)(scale*buffer[i][b]));/* convert to volts */
     }
   printf("\n");
 }
} /* end of sample program */
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

HP E1485A/B C Libraries Example Program 2

```
/* C Libraries example program *
 *       This example program REQUIRES an E1431 at address 16 in
 *       a VXI mainframe controlled by an embedded computer
 *       or MXI computer.
 *
 *       This sample program creates a group containing two channels
 *       and makes a block measurement with the group. *
----------------- example makefile ---------------------------------
E1485_LIB=    /usr/e1485/lib
E1485_INC=    /usr/e1485/include
E1431_LIB=    /usr/e1431/lib
E1431_INC=    /usr/e1431/include
CC=        cc
HPUX_CFLAGS =    -D_HPUX_SOURCE -D_DEBUG_ -D__STDC__ -c -Aa -O
DLD_CFLAGS=    -Aa -c -O -W2,-x -DE1485_SOURCE $(EFLAG) -I $(E1485_INC) \
         -I $(E1431_INC)
DLD_LIBS=    $(E1431_LIB)/libd1431.a
all:        example example.dld
### series 300 and 700 unix build
example.o    :    example.c  $(E1431_INC)/e1431.h
        $(CC) $(HPUX_CFLAGS) -I $(E1431_INC) -I $(E1485_INC) example.c
example    :    $(E1431_LIB)/lib1431.a example.o
        $(CC) example.o
$(E1431_LIB)/lib1431.a -lsicl -lc -lm -o example
### E1485 build
example.dld:    dexample.o
        ld -dr -N $(E1485_LIB)/libspil.o dexample.o $(DLD_LIBS) \
           -o example.dld
        nm -u example.dld
        cp example.dld ../scope
dexample.o:    example.c $(E1431_LIB)
        cc $(DLD_CFLAGS) example.c -o dexample.o
------------------------------------------------------------------- */
#include <stdio.h>
#include "e1431.h"
#include "machType.h"

#define RANGE    10.0              /* 10 volt range */
#define BLOCKSIZE    16            /* 16 points per block */
#define NUM_CHANS    2             /* two channels in group */
main()
{
SHORTSIZ16 modulelist[] = { 16 };
SHORTSIZ16 channellist[] = { 1,4 };
SHORTSIZ16 groupon;
SHORTSIZ16 error;
LONGSIZ32 actualsize;
SHORTSIZ16 buffer[NUM_CHANS][BLOCKSIZE];
SHORTSIZ16 i,b;
FLOATSIZ64 scale;
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* initialize the library */
  error=e1431_init_io_driver();
  if(error) printf("e1431_init_io_driver() failed, error: %d\n",
error);
/* turn on debugging prints, this is a good idea while developing  */
  e1431_print_errors(1);
/* change this 0 to 1 to see call tracing */
  e1431_trace_level(0);
/* assign channels to modules */
  error=e1431_assign_channel_numbers(1,modulelist);
  if(error) printf("e1431_assign_channel_numbers() failed,error:
%d\n",error);
/* get groupID of a two channel group */
  groupon=e1431_create_channel_group(NUM_CHANS,channellist);
/* check for bogus groupID */
  if(groupon=0)
  printf("e1431_create_channel_group() returns error:%d\n",groupon);
/* set up parameters for reading a block */
  error=e1431_set_analog_input(groupon,E1431_INPUT_MODE_VOLT,
                E1431_INPUT_SOURCE_BNC,
                E1431_ANTI_ALIAS_ANALOG_ON,
                E1431_INPUT_GROUNDED,
                E1431_COUPLING_DC,
                RANGE);
   if(error) printf("e1431_set_analog_input() failed, error: %d\n",
error);
/* no trigger needed for this example */
  error=e1431_set_trigger(groupon,E1431_CHANNEL_OFF,
                0.0,0.0,0.0,E1431_TRIGGER_SLOPE_POS,
                E1431_TRIGGER_MODE_LEVEL);
  if(error) printf("e1431_set_trigger() failed, error: %d\n", error);

  error=e1431_set_data_format(groupon,BLOCKSIZE,
                E1431_DATA_SIZE_16,
                E1431_BLOCK_MODE,
                E1431_APPEND_STATUS_OFF);
  if(error) printf("e1431_set_data_format() failed, error: %d\n",
error);

  error=e1431_set_data_port(groupon,E1431_SEND_PORT_VME);    if(error)
printf("e1431_set_data_port() failed, error: %d\n", error);

/* top span alias protected data */
  error=e1431_set_decimation_filter(groupon,
                E1431_ONEPASS,
                E1431_ANTI_ALIAS_DIGITAL_ON,
                0);
  if(error) printf("e1431_set_decimation_filter() failed, error:%d\n",
error);
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* set the span of the measurement */
/*
  this was also done by e1431_set_decimation_filter above, but is  */
/*   done here to demoonstrate e1431_set_span() */
  error=e1431_set_span(groupon,25600.0);
  if(error) printf("e1431_set_span() failed, error:%d\n", error);
/* program will trigger and arm the measurement */
  error=e1431_set_auto_trigger(groupon,E1431_MANUAL_TRIGGER);
  if(error) printf("e1431_auto_trigger() failed, error: %d\n", error);

  error=e1431_set_auto_arm(groupon,E1431_MANUAL_ARM);
  if(error) printf("e1431_auto_arm() failed, error: %d\n", error);
/* prepare for collecting data */
  error=e1431_init_measure(groupon);
  if(error) printf("e1431_init_measure() failed, error: %d\n", error);
/* ARM */
  error=e1431_arm_measure(groupon);
  if(error) printf("e1431_arm_measure() failed, error: %d\n", error);
/* TRIGGER */
  error=e1431_trigger_measure(groupon);
  if(error) printf("e1431_trigger_measure() failed, error: %d\n",
error);
/* wait for group block ready */
  while(!(error=e1431_block_available(groupon)));
      ;
  if(error<0) printf("e1431_block_available() failed, error: %d\n",
error);
/* block available flag true means the all channels have a block ready
*/
/* read all blocks of data by using the group ID */
  error=e1431_read_raw_data(groupon,
            buffer[0],
/* start of 2 channel buffer */
            2*BLOCKSIZE,
            &actualsize);
  if(error) printf("e1431_read_raw_data() failed, error: %d\n",error);

/* report if a full block wasn't read */
  if(actualsize!=BLOCKSIZE*NUM_CHANS)
    printf("group read acutalsize=%d  BLOCKSIZE=%d\n",
            actualsize,BLOCKSIZE);
/* get the voltage scale factor for the raw data */
/* see e1431_read_float32_data() for an easier way to do this */
  error=e1431_get_scale(groupon,&scale);
  if(error) printf("e1431_get_scale() failed, error: %d\n", error);
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* print the collected data */
/* print header */
  printf("point    ");
  for(i=0;i<NUM_CHANS;i++)
  {
    printf("chan %1d      ",channellist[i]);
  }
  printf("\n");
/* print data */
  for(b=0;b<BLOCKSIZE;b++)
  {
    printf("%3d -    ",b);
    for(i=0;i<NUM_CHANS;i++)
    {          printf("0x%04x  %8.5f    ",
0xffff&(buffer[i][b]),
          (FLOATSIZ32)(scale*buffer[i][b])); /* convert to volts */
    }
    printf("\n");
    }
  } /* end of sample program */
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C Libraries Quick Reference

Initializing the HP E1431A

Function	Description	Page Number
e1431_init_io_driver	Initialize I/O driver	9-21
e1431_set_interface_addr	Set SICL interface address	9-21
e1431_assign_channel_numbers	Assign IDs to E1431s, and reset them	9-3
e1431_create_channel_group	Create a group of E1431 channels	9-12
e1431_delete_channel_group	Delete a group of HP E1431A channels	9-14
e1431_delete_all_channel_groups	Delete all groups of HP E1431A channels	9-14
e1431_free_all_memory	Free all memory and un-initialize library	9-17
e1431_preset	Preset values	9-25
e1431_reset	Reset (group of) E1431 modules	9-32
e1431_set_active	Enable/disable E1431 channels or groups	9-37
e1431_get_active	Get current state of E1431 channel(s)	9-37

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Configuring the Analog Inputs

Function	Description	Page Number
e1431_set_AC_settling	Select AC settling flag	9-21
e1431_get_AC_settling	Get AC settling flag	9-21
e1431_set_analog_input	Set all analog input parameters	9-38
e1431_set_anti_alias_analog	Enable/disable analog anti-alias filter	9-40
e1431_get_anti_alias_analog	Get current state of analog anti-alias filter	9-40
e1431_auto_range	Set range at level so that no overload occurs	9-4
e1431_auto_zero	Null out input DC offset	9-5
e1431_auto_zero_and_phase	Do autozero, followed by phase calibration	9-5
e1431_auto_zero_get_timestamp	Return time of last autozero	9-5
e1431_set_coupling	Set input coupling to AC or DC	9-52
e1431_get_coupling	Get current state of input coupling	9-52
e1431_set_front_end	Control common front end bus selection	9-61
e1431_get_front_end	Get current front end bus selection	9-61
e1431_set_input_high	Set source of input signal to ADC	9-62
e1431_get_input_high	Get current source of input signal to ADC	9-62
e1431_set_input_low	Set input connector shield to float or ground	9-63
e1431_get_input_low	Get current state of input connector shield	9-63
e1431_set_input_mode	Set input mode to volt or ICP	9-64
e1431_get_input_mode	Get current state of input mode	9-64
e1431_set_input_offset	Set value of the input offset DACs	9-65
e1431_get_input_offset	Get value of the input offset DACs	9-65
e1431_set_range	Set range of E1431	9-73
e1431_get_range	Get current range of E1431	9-73

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Formatting Data

Function	Description	Page Number
e1431_set_append_status	Enable/Disable appending status onto data	9-42
e1431_get_append_status	Get current state of append status switch	9-42
e1431_set_blocksize	Set measurement blocksize	9-47
e1431_get_blocksize	Get current measurement blocksize	9-47
e1431_set_data_format	Set all data format parameters, except data port	9-53
e1431_set_data_mode	Set data collection to block or continuous mode	9-54
e1431_get_data_mode	Get current data collection mode	9-54
e1431_set_data_port	Set data port to VME or Local Bus	9-55
e1431_get_data_port	Get current data port	9-55
e1431_set_data_size	Set size of samples to 16 or 32 bits	9-56
e1431_get_data_size	Get current value of data size	9-56
e1431_set_lbus_mode	Set local bus mode	9-70
e1431_get_lbus_mode	Get local bus mode	9-70
e1431_reset_lbus	Reset or enable Local Bus	9-22

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Configuring Digital Processing

Function	Description	Page Number
e1431_set_anti_alias_digital	Enable/disable digital anti-alias filter	9-41
e1431_get_anti_alias_digital	Get current state of digital anti-alias filter	9-41
e1431_set_center_freq	Set zoom center frequency	9-49
e1431_get_center_freq	Get zoom center frequency	9-49
e1431_set_decimation_bandwidth	Set data decimation bandwidth	9-57
e1431_get_decimation_bandwidth	Get current data decimation bandwidth	9-57
e1431_set_decimation_filter	Set all decimation filter parameters, except settling time	9-58
e1431_set_decimation_output	Set single or multi-pass filter output	9-59
e1431_get_decimation_output	Get current state of filter output	9-59
e1431_set_filter_settling_time	Change default filter settling time	9-60
e1431_get_filter_settling_time	Get current settling time	9-60
e1431_set_replay_data_size	Set size of replay input data	9-74
e1431_get_replay_data_size	Get current replay data size	9-74
e1431_set_span	Set measurement span	9-75
e1431_get_span	Get measurement span	9-75
e1431_set_time_offset	Set input time offset correction value	9-76
e1431_get_time_offset	Get input time offset correction value	9-76
e1431_set_zoom	Set zoom state for replay only	9-87
e1431_get_zoom	Get current zoom state	9-87

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Controlling Measurements

Function	Description	Page Number
e1431_arm_measure	Manually arm	9-2
e1431_finish_measure	Cleans up the VXI bus after a measurement is no longer needed	9-16
e1431_init_measure	Initiate measurement, move E1431s to IDLE state	9-22
e1431_init_measure_to_booted	Initiate measurement to booted stage	9-22
e1431_init_measure_finish	Move measurement from booted stage to completion	9-22
e1431_init_replay	Initialize replay	9-24
e1431_reset_measure	Reset current measure, move E1431s to TESTED	9-32
e1431_set_auto_group_meas	Control automatic module grouping	9-45
e1431_get_auto_group_meas	Read automatic module grouping	9-45
e1431_trigger_measure	Manually trigger	9-89

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Triggering

Function	Description	Page Number
e1431_set_auto_arm	Set auto arm state	9-44
e1431_get_auto_arm	Get current auto arm state	9-44
e1431_set_auto_trigger	Set auto trigger state	9-46
e1431_get_auto_trigger	Get current auto trigger state	9-46
e1431_set_trigger	Set all trigger parameters, except auto arm and auto trigger	9-77
e1431_set_trigger_channel	Set a channel as trigger source	9-79
e1431_get_trigger_channel	Get current trigger source	9-79
e1431_set_trigger_delay	Set trigger delay	9-80
e1431_get_trigger_delay	Get current trigger delay	9-80
e1431_set_trigger_level	Set trigger level	9-81
e1431_get_trigger_level	Get current trigger level	9-81
e1431_set_trigger_mode	Set trigger mode (i.e. level or bounded)	9-82
e1431_get_trigger_mode	Get current trigger mode	9-82
e1431_set_trigger_slope	Set slope of trigger	9-84
e1431_get_trigger_slope	Get current slope of trigger	9-84
e1431_trigger_measure	Manually trigger	9-89

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Reading Data

Function	Description	Page Number
e1431_block_available	Return true if data is ready	9-7
e1431_check_overloads	Check for overloads for the last data block	9-11
e1431_read_raw_data	Read raw data from E1431 channel	9-27
e1431_read_float32_data	Read scaled float data from HP E1431A channel	9-27
e1431_read_float64_data	Read scaled float data from HP E1431A channel	9-27
e1431_replay_data	Send data for replay processing	9-26
e1431_replay_data_wanted	Returns status of DSPs	9-26
e1431_get_scale	Get a voltage scale factor	9-20

Controlling Multiple Modules

Function	Description	Page Number
e1431_set_clock_master	Set sample clock master	9-50
e1431_get_clock_master	Get sample clock master	9-50
e1431_set_clock_source	Set sample clock source	9-51
e1431_get_clock_source	Get current value of sample clock source	9-51
e1431_set_multi_sync	Set multi-module system synchronization	9-71
e1431_get_multi_sync	Get current multi-module system sync state	9-71
e1431_set_ttltrg_lines	Select a pair of sync/clock lines	9-86
e1431_get_ttltrg_lines	Get current selection of sync/clock lines	9-86

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Programming Interrupts

Function	Description	Page Number
e1431_reenable_interrupt	Reenable interrupt hardware	9-30
e1431_set_interrupt	Set all interrupt parameters	9-66
e1431_set_interrupt_mask	Set interrupt mask	9-67
e1431_get_interrupt_mask	Get current interrupt mask	9-67
e1431_set_interrupt_priority	Set interrupt priority	9-69
e1431_get_interrupt_priority	Get current interrupt priority	9-69

Reading and Writing to Registers

Function	Description	Page Number
e1431_burn_eerom	Program "flash EEROM" on a module	9-10
e1431_read_eerom	Reads a block from the "flash eerom" on a module	9-10
e1431_read_register	Read register from E1431	9-29
e1431_write_register	Write register to E1431	9-29
e1431_get_register_address	Get memory mapped address of E1431 register	9-19

Calibration

Function	Description	Page Number
e1431_burn_calibration	Store calibrations in EEROM	9-9
e1431_read_calibration	Fills out cal structure	9-9
e1431_set_calibration	Set calibrations on or off	9-48
e1431_get_calibration	Get current calibration setting	9-48
e1431_trigger_time_correction	Get time correction of trigger event	9-90

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Debugging Your Program

Function	Description	Page Number
e1431_debug_level	Enable/disable register write printout	9-13
e1431_display_state	Dump E1431 state in easy to read format	9-15
e1431_print_errors	Enable/disable function error printout	9-26
e1431_get_error_string	Return last error string	9-18
e1431_selftest	Perform self test of hardware	9-35
e1431_set_ramp	Set ramp state	9-72
e1431_get_ramp	Get current ramp state	9-72
e1431_set_try_recover	Control signal trapping	9-28
e1431_trace_level	Enable/disable call tracing printout	9-88

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

9

C Interface Library Support Reference

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_arm_measure

Manually arm, move HP E1431As from IDLE to ARM state

Synopsis

SHORTSIZ16 e1431_arm_measure(SHORTSIZ16 ID)

Description

e1431_arm_measure moves all modules in the group from the IDLE state to the ARM state. This function performs a "manual arm", and does not need to be called when the group is set to "auto-arm". This function is called for the first time after *e1431_init_measure*, and then, when in block mode, after each data block has been read out of the module. See the "Measurement setup and control" section earlier in this manual, for a detailed description of the measurement states.

This function waits for all modules to be in the IDLE state, before proceeding further, and it will return an error if this state is not reached after a limited time. After the call to *e1431_arm_measure* completes successfully, the measurement will proceed further, as the trigger event occurs (see *e1431_trigger_measure*).

ID is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_init_measure

e1431_trigger_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_assign_channel_numbers

Assign IDs to HP E1431As, and reset them to power-up state

Synopsis

SHORTSIZ16 e1431_assign_channel_numbers(SHORTSIZ16 modCount,
SHORTSIZ16 *logAddrList)

Description

e1431_assign_channel_numbers must be called once, following the driver initialization, in order to declare to the library the logical addresses of the HP E1431A modules that will be used. All modules in the list are also reset to their power-up state, with *e1431_reset*.

This function checks the existence of an HP E1431A module at each of the logical addresses given in *logAddrList*, and allocates logical channel identifiers, *chanID* (or *ID*), and module identifiers, *modID*, for each logical address, starting at 1. For example, if two logical addresses (*la*) are passed to the function, the module at first *la* is assigned *modID* #1, and *chanID* #1 to #8, and the module at the second *la* in the list, is assigned *modID* #2, and *chanID* #9 to #16.

This function dynamically allocates memory, in order to store the internal structures kept for each channel and each module. These structures are first removed, and then created, each time that this function is called. It is the most recent call to this function which takes precedence over previous ones. It is also possible to free the memory allocated to these internal structures, by calling *e1431_delete_all_chan_groups*.

There are some restrictions when programming multiple mainframes connected by the MXI interface. The TTL trigger lines that are used by the HP E1431A to synchronize multiple-module measurements are uni-directional, left-to-right, through the MXI interface. The module that drives the TTL trigger lines must be in the left mainframe. The array of logical addresses sent to *e1431_assign_channel_numbers* must be in descending order. The lowest-logical address module must be in the left mainframe. (The lowest-logical address module is assigned the highest channel numbers by *e1431_assign_channel_numbers*.) Channel triggering must be done by channels in the left mainframe. Group *e1431_read_XXX()* should use the group ID form. For more information, see "Managing Multiple Mainframe Measurements" in Chapter 8.

modCount is the count of logical addresses passed in the second parameter, **logAddrList*. This number should be between 1 and 255.

**logAddrList* is the pointer to the list of logical addresses to be known by the library. Logical addresses have values ranging between 1 and 255.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_create_channel_group
e1431_delete_all_chan_groups
e1431_reset

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_auto_range

Set range at level so that no overload occurs

Synopsis

SHORTSIZ16 e1431_auto_range(SHORTSIZ16 ID, FLOATSIZ64 time)

Description

e1431_auto_range sets the range of a group of channels to the lowest value that does not cause an ADC overload to occur. The algorithm starts at the lowest range and checks the input for the given *time*, or until an overload occurs. If an overload occurs, the range is bumped up and is checked for another *time* duration.

ID is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

time is the time in seconds that the autorange algorithm waits at each range setting checking for an overload. Specifying a time of 0.0 will cause the function to use a default time related to blocksize and span and equal to ((1/span) * blocksize * 0.1 sec.).

Reset Value

After a reset, *range* is set to 10.0 volts.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_auto_zero
e1431_auto_zero_and_phase
e1431_auto_zero_get_timestamp

Corrects the input DC offset
Perform autozero, followed by phase calibration
Return time of last autozero

Synopsis

SHORTSIZ16 e1431_auto_zero(SHORTSIZ16 ID)

SHORTSIZ16 e1431_auto_zero_and_phase(SHORTSIZ16 ID,
SHORTSIZ16 calmode)

SHORTSIZ16 e1431_auto_zero_get_timestamp(SHORTSIZ16 ID,
LONGSIZ32 *seconds, char *date);

Description

e1431_auto_zero sets the input offset DAC of a single channel to a value that nulls out the offset of the input amplifier of the ADC. It does this by grounding the amplifier and going through all ranges, nulling the offset voltage. Each of these values of offset voltage are stored in the internal state structure for the channel. Whenever a future range change occurs, the offset value for that range is placed in the offset DAC to zero the input offset.

If a measurement is in progress while calling this function, the measurement is aborted. Then, the current state of each channel is saved, and some of the settings are changed as appropriate. Various measurements are performed calling successively *e1431_set_active_channel*, *e1431_init_measure*, *e1431_arm_measure*, and *e1431_trigger_measure*. Once done, and the offset saved for the range and the channel, the previous state of the channel is restored.

Autozero data is also stored on the HP E1431A in RAM and will be read during *e1431_init_io*. This data will be lost if power is removed.

e1431_auto_zero_and_phase will perform an autozero and also calculate the time offsets needed to correct inter-channel phase error. These time offsets are stored and used just like the autozero data.

e1431_auto_zero_get_timestamp returns the time of the last autozero performed, or a 0 if no autozero data can be found. Autozero data is kept in RAM on the HP E1431A and will remain until power is lost.

ID is the ID of a single of channel that was obtained with a call to *e1431_create_channel_group*.

seconds is a pointer which receives the time of the last autozero. This time is the number of seconds since Jan 1 1970. If the last autozero has been lost (power has been cycled) *seconds* will return a zero. This zero can be used by a program to decide whether or not to autozero the HP E1431A. The *seconds* pointer can be null if the preceding information is not wanted.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

date is similar to the *seconds* pointer, except it returns a 26 element character array of the autozero date (example: "Fri Jan 28 09:12:40 1994\\n\\0"). This string is generated by the unix call "ctime". A null string "" is returned for a lost autozero. The *date* pointer can be null if the preceding information is not wanted.

calmode is either E1431_TIME_DOMAIN or E1431_FREQ_DOMAIN. These modes minimize either the time error or the phase error over the frequency span. The difference between these two modes is quite small. E1431_TIME_DOMAIN will have the phase error at 25.6kHz be a few tenths higher than E1431_FREQ_DOMAIN.

Reset Value

Offset values are unaffected by reset.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_get_input_offset

e1431_get_time_offset

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_block_available

Return status of data FIFO

Synopsis

SHORTSIZ16 e1431_block_available(SHORTSIZ16 ID)

Description

e1431_block_available returns one of three status conditions:

- A positive number (TRUE) if a block of data is available to be read using *e1431_read_raw_data* or *e1431_read_float32_data*.
- Zero if a block of data is not available.
- A negative error number if an error occurred.

The HP E1431A holds a maximum of 16384 samples of data for each channel in an on-board FIFO. When the HP E1431A is set to block mode, *e1431_set_data_mode* is E1431_BLOCK_MODE, the channel stops acquiring data after one block of data has accumulated in the FIFO. When the HP E1431A is set to continuous mode, *e1431_set_data_mode* is E1431_CONTINUOUS_MODE, the channel continues to acquire data until the FIFO overruns because data is not read out fast enough or because the HP E1431A is initialized with *e1431_init_measure*.

If the FIFO overruns, data acquistion stops. Data remaining in the FIFO is valid and can be read. *e1431_block_available* returns TRUE (positive) as long as some data remains in the FIFO. Calls to *e1431_block_available* when the FIFO is empty because of a data overrun, will return the error, -ERR1431_FIFO_OVERRUN.

The usage of *e1431_block_available* varies depending upon whether a group ID or channel ID is used. The easiest way to get data is to use only group IDs. If a group ID is used, *e1431_block_available* will return positive when all blocks on all modules are ready for reading by either the VMEbus or the LBUS.

Example

```
/* wait for data, handle errors */
while(error=e1431_block_available(groupID,chanID))
      ;
  if(error < 0)
   call_error_handler_routine();
  else
  e1431_read_raw_data(groupID,*buffer,size,&actualCount);
```

buffer must be large enough to hold all channels of data in the groupID case.

If a channel ID is used, *e1431_block_available* returns positive when all blocks on that one module are ready to be read. After *e1431_block_available* returns positive for a channel, you should read all channels of data from that module. Other channels on other modules in that group will have to be checked for the FIFO data status separately.

Using a group ID frees you from managing module issues.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Reset Value Not applicable.

Return Value Return positive if successful, 0 if no data available, a negative error number if error

See Also e1431_read_raw_data

e1431_read_float32_data

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_burn_calibration
e1431_read_calibration

Store calibrations in EEROM

Synopsis

SHORTSIZ16 e1431_burn_calibration(SHORTSIZ16 ID,E1431UserCal *cal, SHORTSIZ16 where,char *passwd)

SHORTSIZ16 e1431_read_calibration(SHORTSIZ16 ID,E1431UserCal *cal, SHORTSIZ16 where)

Description

e1431_burn_calibration writes the calibration data into the "flash eerom" on a module.

e1431_read_calibration reads the calibration data from the "flash eerom" on a module and fills out the calibration structure.

chanID is the channel used to identify an HP E1431A module. Any channel of a module will do. The channel must be in an group set up by *e131_create_channel_groups()*.

cal is a pointer to a filled-out HP E1431 UserCal structure. See the include file e1431.h for the names of the variable for each range.

where is either E1431_USER or E1431_FACTORY. Using E1431_FACTORY requires a password.

passwd is the password needed to write factory calibration.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_burn_eerom

Program "flash EEROM" on a module

Synopsis

```
SHORTSIZ16  e1431_burn_eerom(SHORTSIZ16 chanID,
                             SHORTSIZ16 block,
                             char *data,char *passwd)
SHORTSIZ16  e1431_read_eerom(SHORTSIZ16 chanID,
                             SHORTSIZ16 block,
                             char *data)
```

Description

e1431_burn_eerom writes a block into the "flash eerom" on a module. This function is used by *e1431_write_calibration()*. Two of the blocks mentioned below CAL10 and CAL20 are open for use. The other two BOOT, MAIN are reserved and are password protected.

e1431_read_eerom reads a block from the "flash eerom" on a module.

chanID is the channel used to identify an HP E1431A module. Any channel of a module will do. The channel must be in an group set up by *e131_create_channel_groups()*.

block is one of the four blocks available: E1431_ROM_CALIB10, E1431_ROM_CALIB20, E1431_ROM_BOOT, E1431_ROM_MAIN.

Caution

If you write to E1431_ROM_BOOT you will overwrite the factory calibrations.

If you write to E1431_ROM_MAIN you will destroy the E1431A module.

BLOCK	SIZE	USE
E1431_ROM_CALIB10	4K (4096 decimal)	open for customer
E1431_ROM_CALIB20	4K (4096 decimal)	open for customer
E1431_ROM_BOOT	8K (8192 decimal)	factory calibration
E1431_ROM_MAIN	112K (114688 decimal)	reserved

data is a character buffer large enough to hold the complete block of data being burned into the eerom or read out of the eerom.

passwd is the password needed to write to E1431_ROM_MAIN or E1431_ROM_BOOT.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_check_overloads

Check for overloads for the last data block

Synopsis

```
SHORTSIZ16  e1431_check_overloads(SHORTSIZ16 ID,
                                  SHORTSIZ16 *any,
                                  SHORTSIZ16 *common,
                                  SHORTSIZ16 *diff,
                                  SHORTSIZ16 *ADC)
```

Description

e1431_check_overloads queries the overload registers and returns status information using four pointers. The first pointer location "any" will be true (non-zero) if any over load occured. The other three pointer locations will be filled with arrays of TRUE/FALSE flags, one per channel in the original channel list used in e1431_create_channel_group(). Any of the four pointers can be NULL if that information is not wanted. Overload flags are reset by either a call to *e1431_check_overloads* or a call to *init_measure()*.

ID is the ID of a group or single channel.

Example

```
---SHORTSIZ16any;SHORTSIZ16common[5];SHORTSIZ16diff[5];SHORTSIZ16ADC[5];
/* assuming groupID is a 5 channel group */
error=e1431_check_overloads(groupID,&any,common,diff,ADC);
if(error)
 printf("OVERLOAD error=%d
%s\n",error,e1431_get_error_string());
```

any is a pointer to SHORTSIZ16 that is set by any overload. "*any*" can be NULL if the information is not desired.

common is a pointer to SHORTSIZ16 that is set by a common mode input overload. overload. "common" can be NULL if the information is not desired.

diff is a pointer to SHORTSIZ16 that is set by a differential mode input overload. "diff" can be NULL if the information is not desired.

ADC is a pointer to SHORTSIZ16 that is set by an ADC overload. "ADC" can be NULL if the information is not desired.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_read_raw_data

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_create_channel_group

Create a group of HP E1431A channels

Synopsis

SHORTSIZ16 e1431_create_channel_group(SHORTSIZ16 chanCount,
SHORTSIZ16 *chanList)

Description

e1431_create_channel_group creates and initializes a channel group. A channel group allows you to issue commands to several HP E1431A channels at once, simplifying system setup. You can overlap channel groups. The state of an individual HP E1431A channel that is in more than one channel group, is determined by the most recent operation performed on the group to which this channel belongs.

This function dynamically allocates memory to keep track of the groups which have been created. This memory can be freed by *e1431_delete_channel_group* and *e1431_delete_all_chan_groups*.

chanCount is the count of channels passed in the second parameter, **chanList*. This number should be between 1 and the maximum number of channels declared with *e1431_assign_channel_numbers* (8 times the *modCount* parameter).

**chanList* is the pointer to the list of logical channels identifiers of the group to be created. This list must be in ascending order and contain no repeats.

Reset Value

Not applicable.

Return Value

If successful, this function returns a negative integer, *groupID*, which is then used to reference the channel group in most other functions in this library.

If there is an error, it returns positive error number. This is the only function that returns a positive error number.

See Also

e1431_assign_channel_numbers

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_debug_level

Enable/disable register write printout

Synopsis

void e1431_debug_level(SHORTSIZ16 level)

Description

e1431_debug_level currently offers one level of debugging. This level consists of printing out a message each time a register write occurs to the HP E1431A. The message includes the register number, and the new contents being written. This function, in conjunction with Appendix A, "The VXI Registers", allows detailed examination of the sequence of register writes as well as the contents of the HP E1431A registers at the bit level.

level is an integer whose value is either 0 (no debug), or a non null value (register writes printing).

Reset Value

After a reset, *level* is set to 0, printing of register writes is disabled.

Return Value

void.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_delete_channel_group
e1431_delete_all_channel_groups

Delete a group of HP E1431A channels
Delete all groups of HP E1431A channels

Synopsis

SHORTSIZ16 e1431_delete_channel_group(SHORTSIZ16 groupID)

SHORTSIZ16 e1431_delete_all_channel_groups(void)

Description

e1431_delete_channel_group deletes a group previously made using *e1431_create_channel_group*. The state of an individual channel is maintained if it is referenced by another group.

e1431_delete_all_channel_groups deletes all channel groups. *groupID* is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

Reset Value

Not applicable.

Return Value

e1431_delete_all_chan_groups always returns 0. *e1431_delete_channel_group* returns 0 if successful, a negative error number otherwise.

See Also

e1431_assign_channel_numbers

e1431_create_channel_group

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_display_state

Dump HP E1431A module states in easy-to-read format

Synopsis

void e1431_display_state()

Description

e1431_display_state uses printf to dump the current state of each module that was in the module list when *e1431_assign_channels* was called.

Reset Value

Not Applicable.

Return Value

None

See Also

e1431_assign_channels

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_finish_measure

Cleans up the VXI bus after a measurement is no longer needed

Synopsis

SHORTSIZ16 e1431_finish_measure(SHORTSIZ16 ID)

Description

e1431_finish_measure cleans up after a measurement is no longer needed. It turns off any signals that the group may be sending onto the VXI bus. This call is not needed for the simple case of one group being used to acquire data. This call stops a previous group measurement from interfering with a current group measurement if the two groups happen to share TTLTRG lines for module synchronization.

See *e1431_init_measure* for a description of multi-module measurements.

ID is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_init_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_free_all_memory

Free all memory and un-initialize library

Synopsis

SHORTSIZ16 e1431_free_all_memory(void)

Description

e1431_free_all_memory frees all memory allocated for the library. After *e1431_free_all_memory* is called, the library is effectively turned off. All settings are lost. A call to *e1431_init_io_driver* is necessary to restart the library.

e1431_free_all_memory is intended only if you need to free memory allocated with the function, malloc because the HP E1485A does not reclaim memory when restarting downloaded programs.

Reset Value

Not applicable.

Return Value

Returns 0 if successful, a negative error number otherwise.

See Also

e1431_init_io_driver

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_get_error_string

Return pointer to last error string

Synopsis

```
char  *e1431_get_error_string()
```

Description

e1431_get_error_string returns a pointer to a string holding a one line description of the last error condition.

Example

```
SHORTSIZ16 error;
 error=e1431_set_span(groupID,span);
  if(error)
    printf("error: %d %s\n",error,e1431_get_error_string());
```

Reset Value

Not applicable.

Return Value

pointer to char

See Also

e1431_print_errors

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_get_register_address

Get memory mapped address of HP E1431A register

Synopsis

```
SHORTSIZ16  e1431_get_register_address(SHORTSIZ16 chanID,
                                       SHORTSIZ16 regOffset,
                                       volatile SHORTSIZ16 **addr)
```

Description

e1431_get_register_address returns the address of a register on a single HP E1431A, identified by one of its channels, *chanID*. This address can be used to directly access the register in question.

Caution

Bus errors that occur when accessing a register from an address returned by this function are not trapped. Use extreme care when using the address returned.

chanID is the ID of a single channel.

regOffset is the offset of the register relative to the base address of the HP E1431A. Register offsets have defines of the form E1431_*<register_name>*_REG in the file e1431.h. For example, E1431_VXI_ID_REG is at register offset 0. The list of HP E1431A registers, and their detailed description, may be found in Appendix A, "The VXI Registers."

addr is a POINTER TO A POINTER to a SHORTSIZ16.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_read_register

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_get_scale

Calculate scale factor for current board settings

Synopsis

SHORTSIZ16 e1431_get_scale(SHORTSIZ16 ID, FLOATSIZ64 *scale)

Description

e1431_get_scale returns a scaling factor to convert the raw data into volts. The data from e1431_read_raw_data() should be multiplied by this scale factor to get voltage. The scale factor accounts for range, span, and multi-pass filtering effects. A call with group ID will return E1431_PARAMETER_UNEQUAL if all channels do not have the same scale factor.

ID is the ID of a group or single channel.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_read_raw_data

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_init_io_driver
e1431_set_interface_addr

Initialize I/O driver
Set SICL interface address

Synopsis

SHORTSIZ16 e1431_init_io_driver(void)

Description

e1431_init_io_driver must be the first routine called when using the HP E1431A library. It performs whatever initialization the I/O driver (for example, SICL) needs for the environment in which this library is running.

e1431_set_interface_addr is the one exception to the above rule. Call *e1431_set_interface_addr* before *e1431_init_io_driver* to change the SICL interface address from the default of "vxi" to another interface name. See your SICL documentation for more information. SICL is the "Standard Instrument Interface Library" that the HP E1431 library calls to talk to the hardware.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise into the process' memory.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_init_measure

Initialize measurement, move HP E1431As to IDLE state

Synopsis

SHORTSIZ16 e1431_init_measure(SHORTSIZ16 ID)

SHORTSIZ16 e1431_init_measure_to_booted(SHORTSIZ16 ID)

SHORTSIZ16 e1431_init_measure_finish(SHORTSIZ16 ID)

Description

e1431_init_measure places all modules in the group into the idle state. The modules can be in any state before the call so *e1431_init_measure* can be used to abort a current measurement.

This call resets the hardware and reloads all registers from stored state information in RAM on the controlling computer. Most *e1431_set_XXXX* calls change only RAM copy of the state info, not the hardware. The call to *e1431_init_measure* causes your setup information to be sent to the hardware.

After the call to *e1431_init_measure* completes successfully, the HP E1431A will ready for arming and triggering (see *e1431_arm_measure* and *e1431_trigger_measure*). If auto arm or auto trigger are on, the hardware may proceed beyond the idle state without further intervention.

Overload flags are reset by *e1431_init_measure*; see *e1431_check_overloads*.

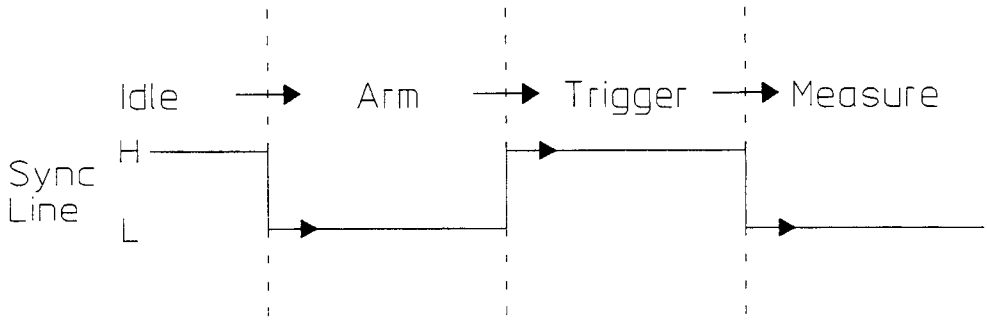
e1431_init_measure_to_booted is for the special case of the library user wanting to initialize the modules half way, (to booted state), performing some communications with other hardware on the VXI bus, and then finishing the measurement with *e1431_init_measure_finish*. Most users will need only the *e1431_init_measure* to perform data collection with the HP E1431A.

ID is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

The measurement itself consists of two phases, the measurement initialization, and the measurement loop. Each of these phases consists of several states, through which the measurement progresses. The transition from one state to the next is tied to a transition in the sync line (one of the TTLTRG lines on the VXI backplane). This SYNC line is "wired-ORed" such that all HP E1431As in a multiple module system must release it for it to go high. Only one HP E1431A is required to pull the SYNC line low. In a single HP E1431A system, the SYNC line is local to the module and is not tied into a TTLTRG line on the VXI backplane.

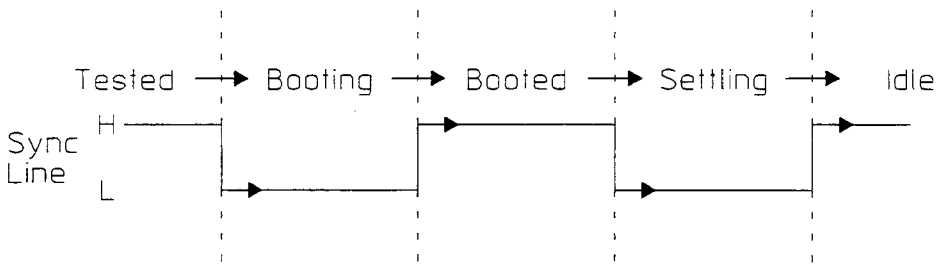
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The measurement initialization states, and the corresponding SYNC line transitions (with 'H' for high, 'L' for low) are:



This complete measurement sequence initialization, from TESTED, through BOOTING, BOOTED, SETTLING and ultimately IDLE, can be performed with a call to the function *e1431_init_measure*.

The progression of measurement loop states and the corresponding SYNC line transistions are:



In the *idle state* the HP E1431A clears the FIFO. The HP E1431A remains in the IDLE state until it sees a high to low transition of the SYNC line. If the HP E1431A is programmed for auto arming, with *e1431_set_auto_arm*, the SYNC line immediately goes low, exiting the IDLE state for the ARM state. The HP E1431A may also be moved to the ARM state by an explicit call to the function, *e1431_arm_measure*.

Upon entering the *arm state* the HP E1431A starts saving new data in its FIFO. It remains in the arm state until the SYNC signal goes high. If an HP E1431A is programmed with a pre-trigger delay, it collects enough data samples to satisfy this pre-trigger delay, and then releases the SYNC line. If no pre-trigger delay has been programmed, the SYNC line goes high immediately. When all HP E1431As in a system have released the SYNC line, allowing it to go high, a transition to the *trigger state* occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Upon entering the *trigger state* an HP E1431A continues collecting data into the FIFO, discarding any data prior to the pre-trigger delay. An HP E1431A remains in the trigger state until it sees a high-to-low transition of the SYNC line. The SYNC line is pulled low by any HP E1431A which encounters a trigger condition and is programmed to pull the SYNC line. If any HP E1431A is programmed for auto triggering, with *e1431_set_auto_trigger*, the SYNC line goes low immediately, exiting the TRIGGER state. The SYNC line may also be pulled low by an explicit call to the function, *e1431_trigger_measure*.

Upon entering the *measure state* an HP E1431A continues to collect data. The HP E1431A also presents the first data from the FIFO to the selected output port, making it available to the controller to read. The HP E1431A holds the SYNC line low as long as it is actively collecting data.

In *block mode*, the HP E1431A stops taking data as soon as a block of data has been collected, including any programmed pre- or post-trigger delays. In continuous mode, the HP E1431A stops taking data when the FIFO overflows. The HP E1431A will remain in the measure state, not taking data, until the FIFO is emptied, then it will move to the idle state.

A channel group that spans more than one module will need to be configured to use the TTL trigger lines on the VXI bus for inter-module communications. This configuration is automatically performed in the *init_measure()* call unless defeated using the *e1431_set_auto_group_meas()* function.

The following discussion outlines what is done automatically by *init_measure()*, or what must be done by the useer if *e1431_set_auto_group_meas()* has been used to defeat auto configuration.

There are 4 pairs of VXIBUS TTL lines that can be used for multiple module synchronization. The pair is selected using the *e1431_set_ttltrg_lines()* (use a group ID to ensure all modules in the group are set to the same pair of line).

All modules need to be set to use the shared sync line rather than the default setting of internal sync. This can be done with the *e1431_set_multi_sync()* function called again using the group ID.

All modules need to be set to use the VXIBUS TTL lines as the clock source. Use *e1431_set_clock_source()*.

The "system module" needs to be set to output the clock. Use *e1431_set_clock_master()* with a channel ID.

All system sync pulses will now come from the "system module" and will drive the measurement state machines on all boards in the group.

After a measurement has completed, the modules are left set up. If a module (call it module'A') is driving the TTL trigger lines and a different group is started which also drives the TTL trigger lines, and that different group does not include module 'A', Module 'A' will conflict and prevent the different group from functioning. In this case a call to *e1431_finish_measure* for group 'A' will turn off module 'A', allowing the new group to function.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note that if the new group includes all modules of the old group, the conflict will not occur since *e1431_init_measure()* will reset all modules as needed. Also note that single module groups do not drive the TTL trigger lines, so single module groups are immune from causing or receiving this conflict.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_arm_measure

e1431_trigger_measure

e1431_finish_measure

e1431_check_overloads

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_init_replay
e1431_replay_data
e1431_replay_data_wanted

Initialize replay
Send data for replay processing
Returns status of DSPs

Synopsis

SHORTSIZ16 e1431_init_replay(SHORTSIZ16 ID)

SHORTSIZ16 e1431_replay_data(SHORTSIZ16 ID,void *buffers[])

SHORTSIZ16 e1431_replay_data_wanted(SHORTSIZ16 ID,
SHORTSIZ16 *want);

Description

The HP E1431A is capable of processing user data through the channel DSPs.

e1431_init_replay() replaces *e1431_init_measure()* when replay mode is desired. Set all measurement parameters before calling *e1431_init_replay()* as you would before *e1431_init_measure()*. *e1431_init_replay()* is effectively replacing the *e1431_init_measure()* call used for normal measurements.

Repeated calls to *e1431_replay_data()* will send the data to the DSPs and the resulting processed data will accumulate in the FIFO exactly as if a real measurement was taking place. *e1431_block_available()* should be used to test whether a block of result data has accumulated in the FIFO, and if not, more calls to *e1431__replay_data()* will produce more data. Data can be read with any read_xxxx_data() calls and will be properly scaled.

e1431_replay_data_wanted() can be used to query the HP E1431A to find out if the HP E1431A will accept more data yet. *e1431_replay_data()* can be called regardless of the valued returned by *e1431_replay_data_wanted()*, but the call to *e1431_replay_data()* will block until the data is accepted.

ID is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

buffers is a list of pointers to blocks of data. These blocks are 1024 points or 16 bit data of 512 points of 32 bit data, depending upon the setting of *e1431_set_replay_data_size()*.

**want* will be set to 1 if data is wanted by the DSPs, 0 otherwise.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_center_freq

e1431_set_replay_data_size

e1431_set_zoom_mode

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_preset

Preset values

Synopsis

SHORTSIZ16 e1431_preset(SHORTSIZ16 ID)

Description

e1431_preset sets a single channel or a group of channels back to initial default values.

ID identifies a group of channels or a single channel to be preset to initial values.

e1431_preset is called by *e1431_create_channel_group* for newly created groups so *e1431_preset* need not be called for new groups.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_create_channel_group

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_print_errors

Enable/disable function error printout

Synopsis

SHORTSIZ16 e1431_print_errors(SHORTSIZ16 enable)

Description

e1431_print_errors enables/disables the printing of error messages when any function returns an error.

If the library is used in a host computer environment, the errors are output to stdout (normally the console screen). If the library is used in the HP E1485A environment, the error messages are output to a terminal connected to the RS-232 port available on this module.

It is normal while developing code to include a call to *e1431_print_errors* enabling error printing early in the code. Once the code has been fully debugged, you may choose to remove this call.

enable is a boolean, with the value being either TRUE or FALSE.

Reset Value

After a reset, *enable* is set to FALSE. Error printing is disabled.

Return Value

Always returns 0.

See Also

e1431_trace_level

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_read_raw_data
e1431_read_float32_data
e1431_read_float64_data

Read raw data from HP E1431A channel
Read scaled float data from HP E1431A channel
Read scaled float data from HP E1431A channel

Synopsis

SHORTSIZ16 e1431_read_raw_data(SHORTSIZ16 ID, void *buffer,
LONGSIZ32 size, LONGSIZ32 *actualCount)

SHORTSIZ16 e1431_read_float32_data(SHORTSIZ16 ID, FLOATSIZ32 *buffer,
LONGSIZ32 size, LONGSIZ32 *actualCount)

SHORTSIZ16 e1431_read_float64_data(SHORTSIZ16 ID, FLOATSIZ64 *buffer,
LONGSIZ32 size, LONGSIZ32 *actualCount)

Description

e1431_read_raw_data returns a block of raw, unscaled data. *buffer* should be either a pointer to SHORTSIZ16 or SHORTSIZ32 depending upon the setting of the data size.

e1431_read_float32_data returns a block of 32 bit floating point data.

e1431_read_float64_data returns a block of 64 bit floating point data.

e1431_read_raw_data, *e1431_read_float32_data* and *e1431_read_float64_data* can be called with either a group ID or a channel ID. If called with a group ID the buffer must be large enough to hold all channels in the group. These two functions will return ERR1431_BUS_ERROR if there is no data ready, so use *e1431_block_available()* to determine data readiness before calling a read_data function. These functions will attempt to read one block size worth of data as set by *e1431_set_blocksize()*.

e1431_read_raw_data is the only block data fetching call that should be used when *e1431_set_append_status* has been used to turn on the status trailer. In that case, the six word trailer will follow the data for each channel, and the data buffer must be large enough to accommodate the added data.

If channel IDs are used, any of these functions has to be called as many times as there are active channels in the group for which the data acquisition has just been performed. Also, the channels must be called in order of the channel list used to create the group. Each channel must be read completely before the next channel data is available. All these restrictions are dealt with by using a group ID instead of channel ID.

These library functions reads are performed via VME. If you need Local Bus reads, you should refer to the library provided with the HP E1485A module in order to perform such data reads.

ID is the ID of a group or single channel.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

buffer is a pointer to the array for returned data.

size is the size, in data points, of *buffer*. If *size* is less than the block size set with *e1431_set_blocksize()*, some channel data will be left on the HP E1431A and will corrupt future reads until cleared by a transition to the IDLE state.

You should always make this size less than or equal to the actual allocated memory for *buffer* or the function may overrun your *buffer*.

actualCnt is a pointer to a long integer. It is set to the actual number of data points transferred into *buffer*. It will always be less than or equal to *size*.

Function	Data size	Bytes per data point
e1431_read_raw_data()	16	2
e1431_read_raw_data()	32	4
e1431_read_float32_data()	32	4
e1431_read_float64_data()	64	8

Data size is set using *e1431_set_data_size*.

Reset Value Not applicable.

Return Value Return 0 if successful, a negative error number otherwise.

See Also e1431_get_scale

e1431_set_append_status

e1431_set_data_size

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_read_register
e1431_write_register

Read register from a single HP E1431A
Write register to a single HP E1431A

Synopsis

```
SHORTSIZ16  e1431_read_register(SHORTSIZ16 chanID,
                                SHORTSIZ16 regOffset,
                                SHORTSIZ16 *data)
SHORTSIZ16  e1431_write_register(SHORTSIZ16 chanID,
                                 SHORTSIZ16 regOffset,
                                 SHORTSIZ16 data)
```

Description

e1431_read_register reads the contents of a register of an HP E1431A module, identified by one of its channels, *chanID*.

e1431_write_register writes data to a particular register of an HP E1431A module.

These functions all trap bus errors that might occur while reading or writing a register, and return the error, ERR1430_BUS_ERROR, if a bus error occurs.

chanID is the ID of a single channel which is used to identify the module.

regOffset is the offset of the register relative to the base address of the HP E1431A. Register offsets have defines of the form E1431_*<register_name>*_REG in the file e1431.h. For instance, E1431_VXI_ID_REG is at register offset 0. The list of HP E1431A registers, and their detailed description, may be found in the hardware appendix.

data is the value of the data, to be read from, or to be written into, a register.

accType is the type of register to access. E1431_READ_REG for a read register. E1431_WRITE_REG for a write register.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_get_register_address

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_reenable_interrupt

Reenable interrupt hardware

Synopsis

SHORTSIZ16 e1431_reenable_interrupt(SHORTSIZ16 ID)

Description

e1431_reenable_interrupt re-writes the interrupt settings to the hardware since the setting are lost as a result of the interrupt. The last thing an interrupt response routine should do is call *e1431_reenable_interrupt*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

Reset Value

None.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_interrupt

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_replay_data
e1431_replay_data_wanted

Send data for replay processing
Returns status of DSPs

Synopsis

SHORTSIZ16 e1431_replay_data(SHORTSIZ16 ID,void *buffers[])

SHORTSIZ16 e1431_replay_data_wanted(SHORTSIZ16 ID,
SHORTSIZ16 *want);

Description

The HP E1431A is capable of processing user data through the channel DSPs.

Repeated calls to *e1431_replay_data()* will send the data to the DSPs and the resulting processed data will accumulate in the FIFO exactly as if a real measurement was taking place. *e1431_block_available()* should be used to test whether a block of result data has accumulated in the FIFO, and if not, more calls to *e1431__replay_data()* will produce more data. Data can be read with any read_xxxx_data() calls and will be properly scaled.

e1431_replay_data_wanted() can be used to query the HP E1431A to find out if the HP E1431A will accept more data yet. *e1431_replay_data()* can be called regardless of the valued returned by *e1431_replay_data_wanted()*, but the call to *e1431_replay_data()* will block until the data is accepted.

e1431_init_replay() replaces the *e1431_init_measure()* call used for real time measurements. For more information about this function, see the description on page 9-26.

ID is the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*.

buffers is a list of pointers to blocks of data. These blocks are 1024 points or 16 bit data of 512 points of 32 bit data, depending upon the setting of *e1431_set_replay_data_size()*.

**want* will be set to 1 if data is wanted by the DSPs, 0 otherwise.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_init_replay

e1431_set_center_freq

e1431_set_replay_data_size

e1431_set_zoom_mode

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_reset

Reset (group of) HP E1431A modules

Synopsis

SHORTSIZ16 e1431_reset(SHORTSIZ16 ID)

Description

e1431_reset resets the (group of) channel(s), *ID*, to power-up settings. You probably will not need to use this call because higher level calls reset when necessary.

Normal library use of creating a group, calling *init_measure()* and then collecting the data, will not need to use *e1431_reset*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

e1431_reset is called by *assign_channel_numbers* so most users should never need to call *e1431_reset*. It is provided here for completeness.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_reset_lbus

Reset or enable Local Bus

Synopsis

SHORTSIZ16 e1431_reset_lbus(SHORTSIZ16 ID, SHORTSIZ16 state)

Description

e1431_reset_lbus controls the reset state of the Local Bus. E1431_RESET_LBUS_ON puts it into reset and E1431_RESET_LBUS_OFF allows the Local Bus to function.

The local bus is put into reset during *e1431_init_measure()* and must be enabled with *e1431_reset_lbus* before data can flow out the local bus.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

state must be either E1431_RESET_LBUS_ON or E1431_RESET_LBUS_OFF.

Reset Value

After a reset the Local Bus is set to E1431_LBUS_MODE_PIPE and E1431_RESET_LBUS_ON.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_lbus_mode

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_reset_measure

Reset current measure, move HP E1431As to TESTED

Synopsis

SHORTSIZ16 e1431_reset_measure(SHORTSIZ16 ID, SHORTSIZ16 opcode)

Description

e1431_reset_measure moves the HP E1431A measurement state machine to the TESTED state.

This function call is not normally needed because *e1431_init_measure* prepares the modules for data gathering.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel. If the measurement involves more than one module, it is mandatory that a *group ID* be used, rather than a *channel ID*. Using a group ID guarantees that all modules in the group "move" synchronously to the TESTED state.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_init_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_selftest

Perform self test of hardware

Synopsis

SHORTSIZ16 e1431_selftest(SHORTSIZ16 mode,
SHORTSIZ16 arg)

Description

e1431_selftest performs a selftest of the hardware. It returns 0 if the test passes and a negative error code if the self test fails.

mode currently must be the logical or of E1431_SELFTEST_ALL_MODULES and either E1431_SELFTEST_FAST or E1431_SELFTEST_LONG.

arg is not currently used.

Example

```
error=e1431_selftest(E1431_SELFTEST_ALL_MODULES|E1431_SELFTEST_FAST,0);

if(error)
 {
printf("Yikes!%s\\n",e1431_get_error_string());
  exit(0);
}
```

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_AC_settling
e1431_get_AC_settling

Select AC settling flag
Get AC settling flag

Synopsis

SHORTSIZ16 e1431_set_AC_settling(SHORTSIZ16 ID,
SHORTSIZ16 state)

SHORTSIZ16 e1431_get_AC_settling(SHORTSIZ16 ID,
SHORTSIZ16 *state)

Description

Some input settings cause the module to wait until transients have died down before returning from *e1431_init_measure()*. This wait is approximately 10 seconds. *e1431_set_AC_settling* allows you to turn off this delay, but the collected data may not be accurate while the transient dies.

The default setting of E1431_AC_SETTLING_ON is intended to be the normal setting for most users of HP E1431A.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel. If even one channel of a group requires the wait, *e1431_init_measure()* will wait.

state is either E1431_E1431_AC_SETTLING_OFF or E1431_AC_SETTLING_ON

Reset Value

Each group defaults to E1431_AC_SETTLING_ON

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_set_coupling

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_active
e1431_get_active

Set a group or channel active
Get group or channel active state

Synopsis

SHORTSIZ16 e1431_set_active(SHORTSIZ16 ID,SHORTSIZ16 state)

SHORTSIZ16 e1431_get_active(SHORTSIZ16 ID,SHORTSIZ16 *state)

Description

e1431_set_active is used to switch between groups before starting data collection with *init_measure*.

e1431_get_active returns the state of a channel or group.

State can be either E1431_CHANNEL_ON or E1431_CHANNEL_OFF.

If *ID* is a channel ID, that one channel is set to state.

If *ID* is a groupID, all channels of other groups sharing modules with group ID are set to E1431_CHANNEL_OFF and groupID channels are set to state.

A call to *e1431_create_channel_group* automatically calls *e1431_set_active* for the new group so in the simple case of one group, you never need to call *e1431_set_active*.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_create_channel_group

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_analog_input

Set all analog input parameters

Synopsis

```
SHORTSIZ16  e1431_set_analog_input(SHORTSIZ16 ID,
                                   SHORTSIZ16 mode,
                                   SHORTSIZ16 source,
                                   SHORTSIZ16 state,
                                   SHORTSIZ16 ground,
                                   SHORTSIZ16 coupling,
                                   FLOATSIZ32 range)
```

Description

e1431_set_analog_input sets all parameters associated with the analog input section of an HP E1431A or group of HP E1431As.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

mode determines the input mode in the front end. E1431_INPUT_MODE_VOLT sets the volt input mode. E1431_INPUT_MODE_ICP sets the icp input mode. This parameter may also be set with *e1431_set_input_mode*.

source selects the input to the ADC. E1431_INPUT_SOURCE_BNC selects the front panel connector. E1431_INPUT_SOURCE_ZERO grounds the ADC input. E1431_INPUT_SOURCE_SUMBUS selects the VXI SUMBUS line. This parameter may also be set with *e1431_set_input_high*.

state determines the state of the analog anti-alias filter in the front end. E1431_ANTI_ALIAS_ANALOG_ON enables it, and E1431_ANTI_ALIAS_ANALOG_OFF bypasses it. This parameter may also be set with *e1431_set_anti_alias_analog*.

ground selects the grounding of the input connector shell. E1431_INPUT_FLOATING connects the shell to the chassis through a 1 MΩ resistor. E1431_INPUT_GROUNDED grounds the connector shell through a 50 Ω resistor. This parameter may also be set with *e1431_set_input_low*.

coupling determines the AC or DC coupling mode of the input. Using E1431_COUPLING_DC will connect the input directly to the amplifier. E1431_COUPLING_AC inserts a series capacitor between the input and the amplifier. This parameter may also be set with *e1431_set_coupling*.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

range is the full scale range in volts. Signal inputs whose absolute value is larger than full scale will generate an ADC overflow error. The possible values for *range* (in volts) are:

0.005 V	0.01 V	0.02 V	0.05 V	0.1 V	0.2 V	0.5 V	1 V	2 V	5 V	10 V

The actual range that is set will be the nearest legal range value that is greater than or equal to the value specified by the *range* parameter. If a value is specified that is greater than the maximum range, the range is set to the maximum of 10.0 volts. This parameter may also be set with *e1431_set_range*.

Reset Value

After a reset, *mode* is set to E1431_ICP_INPUT_MODE_VOLT, *source* is set to E1431_INPUT_SOURCE_BNC, *state* is set to E1431_ANTI_ALIAS_ANALOG_ON, *ground* is set to E1431_INPUT_FLOATING, *coupling* is set to E1431_COUPLING_DC, and *range* is set to the maximum of 10.0 volts.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_anti_alias_analog

e1431_set_coupling

e1431_set_input_low

e1431_set_input_mode

e1431_set_input_high

e1431_set_range

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_anti_alias_analog
e1431_get_anti_alias_analog

Enable/disable analog anti-alias filter
Get current state of analog anti-alias filter

Synopsis

SHORTSIZ16 e1431_set_anti_alias_analog(SHORTSIZ16 ID, SHORTSIZ16 state)

SHORTSIZ16 e1431_get_anti_alias_analog(SHORTSIZ16 ID, SHORTSIZ16 *state)

Description

e1431_set_anti_alias_analog enables or disables the analog anti-alias filter, of a single channel or group of channels *ID*, depending on the value given in *state*. The digital counterpart to the analog anti-alias filter is controlled with another function, *e1431_set_anti_alias_digital*.

e1431_get_anti_alias_analog returns the current state of the analog anti-alias filter, of a single channel or group of channels *ID*, into a memory location pointed to by **state*.

ID is either the ID of a group of channels that was obtained by a call to *e1431_create_channel_group*, or the ID of a single channel.

state determines the state of the analog anti-alias filter in the front end. E1431_ANTI_ALIAS_ANALOG_ON enables it, and E1431_ANTI_ALIAS_ANALOG_OFF bypasses it. This parameter may also be set with *e1431_set_analog_input*.

Reset Value

After a reset, *state* is set to E1431_ANTI_ALIAS_ANALOG_ON.

Return Value

Return 0 if successful, a negative error number otherwise:

- _NO_ID, if *ID* is not a valid channel or group ID.
- _BUS_ERROR, if access to the registers fails.
- _ILLEGAL_ANTI_ALIAS_MODE, if *state* has an illegal value.
- _PARAMETER_UNEQUAL, if *state* is not the same for all channels in the group.

See Also

e1431_create_channel_group

e1431_set_analog_input

e1431_set_anti_alias_digital

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_anti_alias_digital
e1431_get_anti_alias_digital

Enable/disable digital anti-alias filter
Get current state of digital anti-alias filter

Synopsis

SHORTSIZ16 e1431_set_anti_alias_digital(SHORTSIZ16 ID, SHORTSIZ16 state)

SHORTSIZ16 e1431_get_anti_alias_digital(SHORTSIZ16 ID, SHORTSIZ16 *state)

Description

e1431_set_anti_alias_digital enables or disables the digital anti-alias filter, of a single channel or group of channels *ID*, depending on the value given in *state*. The analog counterpart to the digital anti-alias filter is controlled with another function, *e1431_set_anti_alias_analog*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_anti_alias_digital returns the current state of the digital anti-filter, of a single channel or group of channels *ID*, into a memory location pointed to by **state*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

e1431_set_anti_alias_digital can disable the filter only in top span with filter mode set to E1431_ONEPASS. All other settings will turn on the digital anti alias filter regardless of calls to *e1431_set_anti_alias_digital*.

state determines the state of the digital anti-alias filter.
E1431_ANTI_ALIAS_DIGITAL_ON enables it, and
E1431_ANTI_ALIAS_DIGITAL_OFF bypasses it. This parameter may also be set with *e1431_set_decimation_filter*.

Reset Value

After a reset, *state* is set to E1431_ANTI_ALIAS_DIGITAL_ON.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_anti_alias_analog

e1431_set_decimation_filter

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_append_status
e1431_get_append_status

Enable/disable appending status onto data
Get current state of append status switch

Synopsis

SHORTSIZ16 e1431_set_append_status(SHORTSIZ16 ID,
SHORTSIZ16 append)

SHORTSIZ16 e1431_get_append_status(SHORTSIZ16 ID,
SHORTSIZ16 *append)

Description

e1431_set_append_status controls the appending of a trailer of status data to each channel of data.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

An HP E1431A module with append status set will append a six word trailer to each channel block of data. This trailer is intended to be used for local bus data transfers or with *e1431_read_raw_data()*.

e1431_get_append_status returns the current state of append status, of a single channel or group of channels *ID*, into a memory location pointed to by **append*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

append selects whether or not status information is appended to a data block. Specifying E1431_APPEND_STATUS_ON means that an extra block of status information is appended to the end of each data block transferred. E1431_APPEND_STATUS_OFF disables this feature. This parameter may also be set with *e1431_set_data_format*.

This block of status information consists of 6 words. These words are either 16 or 32 bits long (lower 16 bits padded with ones in the later case), depending on the data size set with *e1431_set_data_size*. The words are appended in the following order:

Reserved (words 1 & 2 & 3) - currently contain 0xffff, reserved for future use.

Decimation Phase Counter (word 4) - These two words contain the time (in number of sample clocks) elapsed between the occurrence of the digital trigger, and the output of the digital filter.

Reserved (word 5) - currently contains 0xffff, reserved for future use.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Service bits (word 6) - This last word indicates if various conditions have occurred during the collection of the block of data.

Bit 0x8000 (data size 16) or 0x80000000 (data size 32) indicates an under-range condition. 1 if not under-range, 0 if under-range.

Bit 0x4000 (data size 16) or 0x40000000 (data size 32) indicates a trigger event. 1 if this channel triggered the measurement.

Bit 0x2000 (data size 16) or 0x20000000 (data size 32) indicates an overload condition. 1 if overloaded.

Reset Value

After a reset, *append* is set to E1431_APPEND_STATUS_OFF.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_auto_arm
e1431_get_auto_arm

Set auto arm state
Get current auto arm state

Synopsis

SHORTSIZ16 e1431_set_auto_arm(SHORTSIZ16 ID,
SHORTSIZ16 armState)

SHORTSIZ16 e1431_get_auto_arm(SHORTSIZ16 ID,
SHORTSIZ16 *armState)

Description

e1431_set_auto_arm sets the arm mode, of a single channel or group of channels *ID*, to the value given in *armState*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_auto_arm returns the current value of the arm mode, of a single channel or group of channels *ID*, into a memory location pointed to by **armState*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

armState determines which arm event will allow the module to advance from the IDLE state into the ARM state. E1431_MANUAL_ARM sets the module to wait for a arm event to occur either from the system (SYNC line), or from the *e1431_arm_measure* command, in order to perform the transition. E1431_AUTO_ARM sets the module to perform the transition as soon as it enters the IDLE state.

Reset Value

After a reset, *armState* is set to E1431_MANUAL_ARM.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_arm_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_auto_group_meas
e1431_get_auto_group_meas

Select auto group set up
Get state of auto group set up

Synopsis

SHORTSIZ16 e1431_set_auto_group_meas(SHORTSIZ16 ID,
SHORTSIZ16 state)

SHORTSIZ16 e1431_get_auto_group_meas(SHORTSIZ16 ID,
SHORTSIZ16 *state)

Description

e1431_set_auto_group_meas is used to control the automatic multi-module set up performed in e1431_init_measure. Since the default is to perform the automatic set up, you probably will never need to call this function. If this function is used to turn off the automatic group setup, each module in a group will need to be set up by your program. For more information about programming module groups, see "Programming Multiple Channels" in Chapter 8.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

state is either E1431_AUTO_GROUP_MEAS_ON or E1431_AUTO_GROUP_MEAS_OFF

Reset Value

Each group defaults to E1431_AUTO_GROUP_MEAS_ON

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_create_channel_group

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_auto_trigger
e1431_get_auto_trigger

Set auto trigger state
Get current auto trigger state

Synopsis

SHORTSIZ16 e1431_set_auto_trigger(SHORTSIZ16 ID,
SHORTSIZ16 trigState)

SHORTSIZ16 e1431_get_auto_trigger(SHORTSIZ16 ID,
SHORTSIZ16 *trigState)

Description

e1431_set_auto_trigger sets the trigger state, of a single channel or group of channels *ID*, to the value given in *trigState*.

e1431_get_auto_trigger returns the current value of the trigger state, of a single channel or group of channels *ID*, into a memory location pointed to by **trigState*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

trigState determines which trigger event will allow the module to advance from the TRIGGER state into the MEASURE state. E1431_MANUAL_TRIGGER sets the module to wait for a trigger event to occur either locally or from the system (SYNC line), or from the *e1431_trigger_measure* command, in order to perform the transition. E1431_AUTO_TRIGGER sets the module to perform the transition as soon as it enters the TRIGGER state.

Reset Value

After a reset, *trigState* is set to E1431_MANUAL_TRIGGER.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_trigger_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_blocksize
e1431_get_blocksize

Set measurement blocksize
Get current measurement blocksize

Synopsis

SHORTSIZ16 e1431_set_blocksize(SHORTSIZ16 ID,
LONGSIZ32 blocksize)

SHORTSIZ16 e1431_get_blocksize(SHORTSIZ16 ID,
LONGSIZ32 *blocksize)

Description

e1431_set_blocksize sets the measurement blocksize, of a single channel or group of channels *ID*, to the value given in *blocksize*. If a measurement is in progress while calling this function, the measurement is aborted with *e1431_stop_measure*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_blocksize returns the current value of the measurement blocksize, of a single channel or group of channels *ID*, into a memory location pointed to by **blocksize*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

blocksize selects the number of sample points in a block. Any number between 1 and 16384 (16k) is a valid value for *blocksize*. This parameter should not take into account the size of the appended status data, as defined in *e1431_set_append_status*. This parameter may also be set with *e1431_set_data_format*.

Reset Value

After a reset, the measurement *blocksize* is set to 1024 (1K).

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_append_status

e1431_set_data_format

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_calibration
e1431_get_calibration

Set calibrations on or off
Get current calibration setting

Synopsis

SHORTSIZ16 e1431_set_calibration(SHORTSIZ16 ID, SHORTSIZ16 state)

SHORTSIZ16 e1431_get_calibration(SHORTSIZ16 ID, SHORTSIZ16 *state)

Description

e1431_set_calibration controls the application of correction data (calibrations) to the input data of a single channel or group of channels *ID*.

The calibrations are factory set and can not be changed.

state must be either E1431_CALIBRATION_ON or E1431_CALIBRATION_OFF.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

Reset Value

After a reset, *state* is set to E1431_CALIBRATION_ON.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_burn_calibration

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_center_freq
e1431_get_center_freq

Set zoom center frequency
Get zoom center frequency

Synopsis

SHORTSIZ16 e1431_set_center_freq(SHORTSIZ16 ID, FLOATSIZ32 freq)

SHORTSIZ16 e1431_get_center_freq(SHORTSIZ16 ID, FLOATSIZ32 *freq)

Description

e1431_set_center_freq sets the center frequency of the "zooming" when reprocessing data. This value is not used for normal data acquisition modes as the HP E1431A is a baseband only input module.

freq is in hertz, ranges between 0 and 65535 inclusive, and has a resolution of 1 Hz.

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_init_replay

e1431_set_zoom_mode

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_clock_master
e1431_get_clock_master

Set sample clock master
Get sample clock master

Synopsis

SHORTSIZ16 e1431_set_clock_master(SHORTSIZ16 ID, SHORTSIZ16 state)

SHORTSIZ16 e1431_get_clock_master(SHORTSIZ16 ID, SHORTSIZ16 *state)

Description

e1431_set_clock_master controls whether the HP E1431A module drives the VXI bus clock line as selected by *e1431_set_ttltrg_lines*. *e1431_set_clock_master* is called internally by the library when a multi-module group is being set up. You will probably not need to use this function. It is provided here for the rare case of setting up VXI bus signals manually.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_clock_master returns the current value of the master clock for a single channel or group of channels *ID*, into a memory location pointed to by * *state*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

state determines the state of the master clock. E1431_MASTER_CLOCK_ON drives the sample clock from this module onto the VXI bus. If two modules are driving the same TTL trigger line, damage could occur. E1431_MASTER_CLOCK_OFF turns off the sample clock drive.

Reset Value

After a reset, *state* is set to E1431_MASTER_CLOCK_OFF.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_ttltrg_lines

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_clock_source
e1431_get_clock_source

Set sample clock source
Get current value of sample clock source

Synopsis

SHORTSIZ16 e1431_set_clock_source(SHORTSIZ16 ID, SHORTSIZ16 source)

SHORTSIZ16 e1431_get_clock_source(SHORTSIZ16 ID,
SHORTSIZ16 *source)

Description

e1431_set_clock_source sets the source of the sample clock, of a single channel or group of channels *ID*, to the value given in *source*. If a measurement is in progress while calling this function, the measurement is aborted with *e1431_stop_measure*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_clock_source returns the current value of the sample clock source, of a single channel or group of channels *ID*, into a memory location pointed to by **source*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

source determines the source for the sample clock.
E1431_CLOCK_INTERNAL_196K selects the internal oversampled clock, of 196.608 kHz decimated by 3. This is an effective clock of 65,536 Hz.
E1431_CLOCK_VXI_DEC_3 selects the VXI bus oversampled clock, decimated by 3, provided by the clock master.

Note

If the clock comes from the VXI bus, it is provided by the rightmost HP E1431A module in the group. This should be set with a call to *e1431_set_active_group*.

Reset Value

After a reset, *source* is set to E1431_CLOCK_INTERNAL_196K.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_coupling
e1431_get_coupling

Set input coupling to AC or DC
Get current state of input coupling

Synopsis

SHORTSIZ16 e1431_set_coupling(SHORTSIZ16 ID, SHORTSIZ16 coupling)

SHORTSIZ16 e1431_get_coupling(SHORTSIZ16 ID, SHORTSIZ16 *coupling)

Description

e1431_set_coupling sets the input coupling, of a single channel or group of channels *ID*, to the value given in *coupling*.

e1431_get_coupling returns the current state of the input coupling, of a single channel or group of channels *ID*, in a memory location pointed to by **coupling*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

coupling determines the AC or DC coupling mode of the input. Using E1431_COUPLING_DC will connect the input directly to the amplifier. E1431_COUPLING_AC inserts a series capacitor between the input connector and the amplifier. This parameter may also be set with *e1431_set_analog_input*.

Reset Value

After a reset, *coupling* is set to E1431_COUPLING_DC.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_analog_input

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_data_format

Set all data format parameters, except data port

Synopsis

```
SHORTSIZ16  e1431_set_data_format(SHORTSIZ16 ID,
                                  LONGSIZ32 blocksize,
                                  SHORTSIZ16 size,
                                  SHORTSIZ16 mode,
                                  SHORTSIZ16 append)
```

Description

e1431_set_data_format sets all of the parameters associated with the data format section of an HP E1431A or group of HP E1431As.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

blocksize selects the number of sample points in a block. Any number between 1 and 16384 (16k) is a valid value for *blocksize*. This parameter should not take into account the size of the appended status data. This parameter may also be set with *e1431_set_blocksize*.

size selects the number of bits of precision for the fixed point, two's complement data outputs from the HP E1431A. The legal values for this parameter are E1431_DATA_SIZE_16, E1431_DATA_SIZE_32, and E1431_DATA_SIZE_32_SERV. This parameter may also be set with *e1431_set_data_size*.

mode selects whether the HP E1431A's data collection operates in block mode or continuous mode. E1431_BLOCK_MODE selects block transfer mode. E1431_CONTINUOUS_MODE means data collection will be continuous. This parameter may also be set with *e1431_set_data_mode*.

append selects whether or not status information is appended to a data block. Specifying E1431_APPEND_STATUS_ON means that an extra block of status information is appended to the end of each data block transferred. E1431_APPEND_STATUS_OFF disables this feature. This parameter may also be set with *e1431_set_append_status*.

Reset Value

After a reset, *blocksize* is set to 1024 (1K), *size* is set to E1431_DATA_SIZE_16, *mode* is set to E1431_BLOCK_MODE, and *append* is set to E1431_APPEND_STATUS_OFF.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_append_status

e1431_set_blocksize

e1431_set_data_mode

e1431_set_data_size

e1431_stop_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_data_mode
e1431_get_data_mode

Set data collection to block or continuous mode
Get current data collection mode

Synopsis

SHORTSIZ16 e1431_set_data_mode(SHORTSIZ16 ID, SHORTSIZ16 mode)

SHORTSIZ16 e1431_get_data_mode(SHORTSIZ16 ID, SHORTSIZ16 *mode)

Description

e1431_set_data_mode sets the data collection mode, of a single channel or group of channels *ID*, to the value given in *mode*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_data_mode returns the current state of the data collection mode, of a single channel or group of channels *ID*, into a memory location pointed to by **mode*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

mode selects whether the HP E1431A's data collection operates in block mode or continuous mode. E1431_BLOCK_MODE selects block transfer mode. The HP E1431A will stop collecting data, and go to the IDLE state, as soon as one block of data has been collected. E1431_CONTINUOUS_MODE selects continuous data collection. The HP E1431A stays in the MEASURE state and collects data until the FIFO overflows. In that case it goes to the IDLE state. In either mode, the HP E1431A may be explicitly asked to stop the measurement, with *e1431_stop_measure*. This parameter may also be set with *e1431_set_data_format*.

Reset Value

After a reset, *mode* is set to E1431_BLOCK_MODE.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_data_format

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_data_port
e1431_get_data_port

Set data port to VME or Local Bus
Get current data port

Synopsis

SHORTSIZ16 e1431_set_data_port(SHORTSIZ16 ID, SHORTSIZ16 port)

SHORTSIZ16 e1431_get_data_port(SHORTSIZ16 ID, SHORTSIZ16 *port)

Description

e1431_set_data_port sets a single channel or group of channels *ID*, to deliver data either on the VME backplane or on the Local Bus (for high speed data transfers), depending on the value of *port*. In the Local Bus mode, a group of HP E1431As must be contiguous in one mainframe and positioned to the left of the host module (for example, the HP E1485A). The module on the extreme left generates data and the others append their data to the data which is pipelined through them, from left to right.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_data_port returns the current value of the data port, of a single channel or group of channels *ID*, into a memory location pointed to by **port*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

port determines the path the HP E1431A uses to make its data available to the host. E1431_SEND_PORT_VME causes data to be accessed by the host via the VMEbus. E1431_SEND_PORT_LBUS causes data to be transmitted via the Local Bus.

Reset Value

After a reset, *port* is set to E1431_SEND_PORT_VME

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_lbus_mode

e1431_reset_lbus

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_data_size
e1431_get_data_size

Set size of samples to 16 or 32 bits
Get current value of data size

Synopsis

SHORTSIZ16 e1431_set_data_size(SHORTSIZ16 ID, SHORTSIZ16 size)

SHORTSIZ16 e1431_get_data_size(SHORTSIZ16 ID, SHORTSIZ16 *size)

Description

e1431_set_data_size sets the sample data size, of a single channel or group of channels *ID*, to the value given in *size*. If a measurement is in progress while calling this function, the measurement is aborted with *e1431_stop_measure*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_data_size returns the current value of the sample data size, of a single channel or group of channels *ID*, into a memory location pointed to by **size*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

size selects the number of bits per sample. The output is fixed point, two's complement data. The legal values for this parameter are E1431_DATA_SIZE_16, E1431_DATA_SIZE_32, and E1431_DATA_SIZE_32_SERV. This parameter may also be set with *e1431_set_data_format*.

Choosing 16-bit precision allows for more data throughput on the bus. Choosing 32 bits allows more dynamic range (there are exactly 21 bits of dynamic range, the lower order bits being padded with 0s). Choosing 32 bits with *service* (for example, E1431_DATA_SIZE_32_SERV) allows you to get additional information with each sample: overload condition in bit 7, under-range condition in bit 6, and trigger event in bit 5.

Reset Value

After a reset, *size* is set to E1431_DATA_SIZE_16.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_data_format

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_decimation_bandwidth
e1431_get_decimation_bandwidth

Set data decimation bandwidth
Get current data decimation bandwidth

Synopsis

SHORTSIZ16 e1431_set_decimation_bandwidth(SHORTSIZ16 ID, SHORTSIZ16 decBw)

SHORTSIZ16 e1431_get_decimation_bandwidth(SHORTSIZ16 ID, SHORTSIZ16 *decBw)

Description

e1431_set_decimation_bandwidth sets the decimation bandwidth, of a single channel or group of channels *ID*, to the value given in *decBw*. The span is derived from the decimation bandwidth, and sample clock (FS) as follows:

span = FS / (2.56 * 2**decBw).

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

Decimation allows data reduction on oversampled data, saving only those points needed to reconstruct the waveform. A decimation of 2 keeps every other data point, a decimation of 4 keeps every fourth data point, etc. The bandwidth of the data must be reduced at the same time to prevent aliasing.

e1431_get_decimation_bandwidth returns the current value of the decimation bandwidth, of a single channel or group of channels *ID*, into a memory location pointed to by **decBw*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

decBw selects the bandwidth of the filter and the amount of decimation applied to the signal. The possible values for this parameter range from 0 to 16. With the internal sample clock of 65,536 Hz, this leads to spans of 25.6 kHz, 12.8 kHz, 6.4 kHz, 3.2 kHz, 1.6 kHz, 800 Hz, 400 Hz, 200 Hz, 100 Hz, 50 Hz, 25 Hz, 12.5 Hz, 6.25 Hz, 3.125 Hz, 1.56 Hz, 0.78 Hz, and 0.39 Hz. This parameter may also be set with *e1431_set_decimation_filter*.

Reset Value

After a reset, *decBw* is set to 0, therefore leading to a full span of FS/2.56

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_span

e1431_set_decimation_filter

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_decimation_filter

Set all decimation filter parameters, except settling time

Synopsis

SHORTSIZ16 e1431_set_decimation_filter(SHORTSIZ16 ID, SHORTSIZ16 output, SHORTSIZ16 state, SHORTSIZ16 decBw)

Description

e1431_set_decimation_filter sets all of the parameters associated with the decimation filter section of an HP E1431A or group of HP E1431As, except the filter settling time.

ID is either the ID of a group of channels that was obtained by a call to *e1431_create_channel_group*, or the ID of a single channel.

output selects the type of output from the decimation filter. Using E1431_ONEPASS for this parameter selects the output of the last filter in the chain. This is the normal operating mode of the filter. Using E1431_MULTIPASS causes an output consisting of the time multiplexed outputs of all cascaded filters equal to or narrower than the programmed bandwidth. Using E1431_MULTIPASS_OVER_SAMPLED give a result similar to E1431_MULTIPASS but the data is over-sampled. This parameter may also be set with *e1431_set_decimation_output*.

state determines the state of the digital anti-alias filter. E1431_ANTI_ALIAS_DIGITAL_ON enables it, and E1431_ANTI_ALIAS_DIGITAL_OFF bypasses it. This parameter may also be set with *e1431_set_anti_alias_digital*.

decBw selects the bandwidth of the filter and the amount of decimation applied to the signal. The values for this parameter range from 0 to 16. With a sample clock of 65,536 Hz, this leads to spans of 25.6 kHz, 12.8 kHz, 6.4 kHz, 3.2 kHz, 1.6 kHz, 800 Hz, 400 Hz, 200 Hz, 100 Hz, 50 Hz, 25 Hz, 12.5 Hz, 6.25 Hz, 3.125 Hz, 1.56 Hz, 0.78 Hz, and 0.39Hz. This parameter may also be set with *e1431_set_decimation_bandwidth*.

Reset Value

After a reset, *state* is set to E1431_ANTI_ALIAS_DIGITAL_ON, *output* is set to E1431_ONEPASS, and *decBw* is set to 0 (for example, maximum span).

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_anti_alias_digital

e1431_set_decimation_bandwidth

e1431_set_decimation_output

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_decimation_output
e1431_get_decimation_output

Set single or multi-pass filter output
Get current state of filter output

Synopsis

SHORTSIZ16 e1431_set_decimation_output(SHORTSIZ16 ID,
SHORTSIZ16 output)

SHORTSIZ16 e1431_get_decimation_output(SHORTSIZ16 ID,
SHORTSIZ16 *output)

Description

e1431_set_decimation_output sets the filter output, of a single channel or group of channels *ID*, to single or multi-pass, based on the value given in *output*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_decimation_output returns the current pass status of the filter output, of a single channel or group of channels *ID*, into a memory location pointed to by **output*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

output selects the type of output from the decimation filter. The decimation filter consists of a cascaded chain of sections, each decimating the data stream by a factor of two and reducing its bandwidth by a factor of two. Using E1431_ONEPASS for this parameter selects the output of the last filter in the chain. This is the normal operating mode of the filter. Using E1431_MULTIPASS causes an output consisting of the time multiplexed outputs of all cascaded filters equal to or narrower than the programmed bandwidth. This mode is useful when gathering data for synchronous analysis, or octave measurements. Using E1431_MULTIPASS_OVER_SAMPLED gives results similar to E1431_MULTIPASS with the data over-sampled. This parameter may also be set with *e1431_set_decimation_filter*.

When in the multipass mode, with a data size of 32 (see *e1431_set_data_size*), each data sample is tagged with a 5 bits *passcount*. The passcount is placed over the least significant data bits (bits 0 to 4), of the 32 bits data sample.

Reset Value

After a reset, *output* is set to E1431_ONEPASS.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_data_size

e1431_set_decimation_filter

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_filter_settling_time
e1431_get_filter_settling_time

Change default filter settling time
Get current settling time of filter

Synopsis

SHORTSIZ16 e1431_set_filter_settling_time(SHORTSIZ16 ID,
SHORTSIZ16 setlTime)

SHORTSIZ16 e1431_get_filter_settling_time(SHORTSIZ16 ID,
SHORTSIZ16 *setlTime)

Description

e1431_set_filter_settling_time sets the filter settling time, of a single channel or group of channels *ID*, to the value given in *setlTime*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_filter_settling_time returns the current value of the filter settling time, of a single channel or group of channels *ID*, into a memory location pointed to by **setlTime*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

setlTime is the time given to the filter to settle its output. This time has an impact on the time spent in the SETTLING state of the measurement loop: The HP E1431A will wait for that time to elapse, before moving to the IDLE state. This parameter is in sample clock units, and is set to any value between 0 samples, and 255 samples.

Reset Value

After a reset, *setlTime* is set to 64 samples.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_front_end
e1431_get_front_end

Control common front end bus selection
Get current front end bus selection

Synopsis

SHORTSIZ16 e1431_set_front_end(SHORTSIZ16 ID, SHORTSIZ16 select)

SHORTSIZ16 e1431_get_front_end(SHORTSIZ16 ID, SHORTSIZ16 *select)

Description

There exist a signal bus that is routed to all 8 inputs of the E1431. *e1431_set_front_end* selects the source of this bus. Choices are the VXI SUMBUS, ground, or an internally generated 1024 Hz 0.1 volt square wave. This common bus should normally be set to ground, and is set there by *e1431_create_channel_group* and *e1431_preset*.

Use *e1431_set_input_high* with the parameter E1431_INPUT_SOURCE_SUMBUS to connect an input to this common bus.

e1431_get_front_end returns the current selection for the common bus.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel*, or the ID of a single channel.

select must be one of three values, E1431_FE_GROUND, E1431_FE_CAL for the square wave, or E1431_FE_VXI to connect to the VXI analog line SUMBUS.

Reset Value

E1431_FE_GROUND

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_input_high

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_input_high
e1431_get_input_high

Set source of input signal to ADC
Get current source of input signal to ADC

Synopsis

SHORTSIZ16 e1431_set_input_high(SHORTSIZ16 ID, SHORTSIZ16 source)

SHORTSIZ16 e1431_get_input_high(SHORTSIZ16 ID, SHORTSIZ16 *source)

Description

e1431_set_input_high sets the input to the ADC, of a single channel or group of channels *ID*, to the value given in *source*.

e1431_get_input_high returns the current input source selection, of a single channel or group of channels *ID*, into a memory location pointed to by **source*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

source selects the input to the ADC.

- E1431_INPUT_SOURCE_BNC selects the front panel connector.
- E1431_INPUT_SOURCE_ZERO grounds the ADC input.
- E1431_INPUT_SOURCE_SUMBUS selects the VXI SUMBUS line.

Reset Value

After a reset, *source* is set to E1431_INPUT_SOURCE_BNC.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_analog_input

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_input_low
e1431_get_input_low

Set input connector shield to float or ground
Get current state of input connector shield

Synopsis

SHORTSIZ16 e1431_set_input_low(SHORTSIZ16 ID,
SHORTSIZ16 ground)

SHORTSIZ16 e1431_get_input_low(SHORTSIZ16 ID,
SHORTSIZ16 *ground)

Description

e1431_set_input_low sets the input connector shield, of a single channel or group of channels *ID*, to the value given in *ground*.

e1431_get_input_low returns the current state of the input connector shield, of a single channel or group of channels *ID*, into a memory location pointed to by **ground*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel*, or the ID of a single channel.

ground selects the grounding of the input connector shell.
E1431_INPUT_FLOATING connects the shell through a 1 MΩ resistor.
E1431_INPUT_GROUNDED grounds the connector shell to the chassis through a 50Ω resistor. This parameter may also be set with *e1431_set_analog_input*.

Reset Value

After a reset, *ground* is set to E1431_INPUT_FLOATING.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_analog_input

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_input_mode
e1431_get_input_mode

Set input mode to volt or ICP
Get current state of input mode

Synopsis

SHORTSIZ16 e1431_set_input_mode(SHORTSIZ16 ID, SHORTSIZ16 mode)

SHORTSIZ16 e1431_get_input_mode(SHORTSIZ16 ID, SHORTSIZ16 *mode)

Description

e1431_set_input_mode sets the input mode, of a single channel or group of channels *ID*, to the value given in *mode*.

e1431_get_input_mode returns the current value of the input mode, of a single channel or group of channels *ID*, into a memory location pointed to by **mode*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

mode determines the input mode in the front end. E1431_INPUT_MODE_VOLT sets the volt input mode. E1431_INPUT_MODE_ICP sets the ICP input mode. This parameter may also be set with *e1431_set_analog_input*.

Reset Value

After a reset, *mode* is set to E1431_INPUT_MODE_VOLT. The input mode is volt.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_analog_input

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_input_offset
e1431_get_input_offset

Set value of the input offset DACs
Get value of the input offset DACs

Synopsis

```
SHORTSIZ16  e1431_set_input_offset(SHORTSIZ16 chanID,
                                   FLOATSIZ32 range,
                                   SHORTSIZ16 coarse,
                                   SHORTSIZ16 fine)
SHORTSIZ16  e1431_get_input_offset(SHORTSIZ16 chanID,
                                   FLOATSIZ32 range,
                                   SHORTSIZ16 *coarse,
                                   SHORTSIZ16 *fine)
```

Description

e1431_set_input_offset sets the coarse and fine offset nulling DACs on the board. Each range of each channel has a DAC setting that is kept by the library and sent to the HP E1431A when *e1431_init_measure* is called. Two values are stored for each range; an AC value, and a DC value. The coupling set by *e1431_set_coupling* controls which offset is dealt with by *e1431_set_input_offset*.

e1431_get_input_offset retrieves the coarse and fine offset correction factors.

Since the function *e1431_auto_zero* sets these offset correction factors automatically, most users will never need *e1431_set_input_offset* or *e1431_get_input_offset*.

chanID is the ID of a single channel.

range is the input range setting which can range from 0.005 Volts to 10.0 Volts.

coarse 8 bit coarse DAC setting from 0 to 255. A setting of 128 is the center of the DAC and gives no correction. Both coarse and fine DACs invert the signal. Settings above 128 pull the signal in the negative direction. Settings below 128 pull the signal in positive direction.

fine 8 bit fine DAC setting from 0 to 255. Center at 128. See coarse DAC for more information.

Reset Value

After a reset, all settings are 128, for no corrections.

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_auto_zero

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_interrupt

Set all interrupt parameters

Synopsis

SHORTSIZ16 e1431_set_interrupt(SHORTSIZ16 ID, SHORTSIZ16 priority, SHORTSIZ16 mask)

Description

e1431_set_interrupt sets all parameters associated with the interrupt capability of an HP E1431A or group of HP E1431As.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

priority specifies which of the seven VME interrupt lines to use. The only legal values are 0 through 7. Specifying 0 turns the interrupt off, while 7 is the highest priority. This parameter may also be set with *e1431_set_interrupt_priority*.

mask specifies the mask of events on which to interrupt. This mask is created by ORing together various condition bits. Refer to *e1431_set_interrupt_mask* for the definition of the conditions which may be part of the mask. This parameter may also be set with *e1431_set_interrupt_mask*.

Reset Value

After a reset, *priority* is set to 0 (none), and *mask* is set to 0 (all causes masked).

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_reenable_interrupt

e1431_set_interrupt_mask

e1431_set_interrupt_priority

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_interrupt_mask
e1431_get_interrupt_mask

Set interrupt mask
Get current interrupt mask

Synopsis

SHORTSIZ16 e1431_set_interrupt_mask(SHORTSIZ16 ID, SHORTSIZ16 mask)

SHORTSIZ16 e1431_get_interrupt_mask(SHORTSIZ16 ID,
SHORTSIZ16 *mask)

Description

e1431_set_interrupt_mask sets the interrupt mask, of a single channel or group of channels *ID*, to the value given in *mask*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_interrupt_mask returns the current value of the interrupt mask, of a single channel or group of channels *ID*, into a memory location pointed to by **mask*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

mask specifies the mask of events on which to interrupt. This parameter may also be set with *e1431_set_interrupt*. This mask is created by ORing together the various conditions defined in the following table.

Interrupt Mask Bit Definitions

Define (in e1431.h)	Description
E1431_IRQ_READ_VALID	At least one data word in FIFO
E1431_IRQ_BLOCK_READY	Block of data ready in FIFO
E1431_IRQ_ARMED	HP E1431A armed and waiting for trigger
E1431_IRQ_MEAS_DONE	HP E1431A has stopped taking new data
E1431_IRQ_OVERLOAD	Input has exceeded range
E1431_IRQ_ERROR	Hardware fault in ADC or filter

Once the mask has been set and an interrupt occurs, read the E1431_STATUS_REG register to determine the cause of the interrupt. The bit position of the interrupt mask and status registers match so the defines can be used to set and check IRQ bits.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The interrupt mask bits are reset as a result of the interrupt and need to be rewritten to the hardware with *e1431_reenable_interrupt()* to allow the next interrupt.

Reset Value

After a reset, *mask* is set to 0 (all causes masked).

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_reenable_interrupt

e1431_set_interrupt

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_interrupt_priority
e1431_get_interrupt_priority

Set interrupt priority
Get current interrupt priority

Synopsis

SHORTSIZ16 e1431_set_interrupt_priority(SHORTSIZ16 ID,
SHORTSIZ16 priority)

SHORTSIZ16 e1431_get_interrupt_priority(SHORTSIZ16 ID,
SHORTSIZ16 *priority)

Description

e1431_set_interrupt_priority sets the interrupt priority, of a single channel or group of channels *ID*, to the value given in *priority*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_interrupt_priority returns the current value of the interrupt priority, of a single channel or group of channels *ID*, into a memory location pointed to by **priority*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

priority specifies which of the seven VME interrupt lines to use. The only legal values are 0 through 7. Specifying 0 turns the interrupt off, while 7 is the highest priority. This parameter may also be set with *e1431_set_interrupt*.

Reset Value

After a reset, *priority* is set to 0 (none).

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_interrupt

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_lbus_mode
e1431_get_lbus_mode

Set mode for Local Bus
Get current mode for Local Bus

Synopsis

SHORTSIZ16 e1431_set_lbus_mode(SHORTSIZ16 ID, SHORTSIZ16 mode)

SHORTSIZ16 e1431_get_lbus_mode(SHORTSIZ16 ID, SHORTSIZ16 *mode)

Description

e1431_set_lbus_mode sets the Local Bus to one of four settings: append, generate, insert, pipe.

e1431_get_lbus_mode returns the current setting.

The local bus is put into reset during *e1431_init_measure()* and must be enabled with *e1431_reset_lbus* before data can flow out the local bus.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

mode must be one of the following:

- E1431_LBUS_MODE_GENERATE
- E1431_LBUS_MODE_PIPE
- E1431_LBUS_MODE_INSERT
- E1431_LBUS_MODE_APPEND

Reset Value

After a reset the Local Bus is set to E1431_LBUS_MODE_PIPE and E1431_RESET_LBUS_ON.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_reset_lbus

e1431_set_data_port

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_multi_sync
e1431_get_multi_sync

Set multi-module system synchronization , Get current multi-module system sync state

Synopsis

SHORTSIZ16 e1431_set_multi_sync(SHORTSIZ16 ID, SHORTSIZ16 sync)

SHORTSIZ16 e1431_get_multi_sync(SHORTSIZ16 ID, SHORTSIZ16 *sync)

Description

e1431_set_multi_sync sets the multi-module system synchronization, of a single channel or group of channels *ID*, to the value given in *sync*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_multi_sync returns the current value of the multi_module system synchronization, of a single channel or group of channels *ID*, into a memory location pointed to by **sync*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

sync is used to select the synchronization. The HP E1431A supports synchronous operation among multiple HP E1431As by using a VXI TTLTRG line to drive all the modules in a system from the same clock. E1431_SYSTEM_SYNC_OFF sets the SYNC to be generated locally. E1431_SYSTEM_SYNC_ON sets the module to use the SYNC line from the VXI backplane, which is selected with *e1431_set_ttltrg_lines*. This mode uses the SYNC line for multi-module synchronization capabilities including: booting of the digital filters, synchronization of the local oscillators, arming, and triggering.

Reset Value

After a reset, *sync* is set to E1431_SYSTEM_SYNC_OFF.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_ttltrg_lines

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_ramp
e1431_get_ramp

Set ramp state
Get current ramp state

Synopsis

SHORTSIZ16 e1431_set_ramp(SHORTSIZ16 ID, SHORTSIZ16 state)

SHORTSIZ16 e1431_get_ramp(SHORTSIZ16 ID, SHORTSIZ16 *state)

Description

e1431_set_ramp sets the ramp state, of a single channel or group of channels *ID*, to the value given in *state*. Turning on the ramp will cause the DSP to manufacture data rather then read from the input ADCs. This is useful for debugging.

e1431_get_ramp returns the current value of the state, of a single channel or group of channels *ID*, into a memory location pointed to by **state*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

state can be either E1431_RAMP_ON or E1431_RAMP_OFF

Reset Value

After a reset, state is set to E1431_RAMP_OFF.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_range
e1431_get_range

Set range of HP E1431A
Get current range of HP E1431A

Synopsis

SHORTSIZ16 e1431_set_range(SHORTSIZ16 ID, FLOATSIZ32 range)

SHORTSIZ16 e1431_get_range(SHORTSIZ16 ID, FLOATSIZ32 *range)

Description

e1431_set_range sets the range, of a single channel or group of channels *ID*, to the value given in *range*.

e1431_get_range returns the current value of the range, of a single channel or group of channels *ID*, into a memory location pointed to by **range*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

range is the full scale range in volts. Signal inputs with an absolute value larger than full scale generate an ADC overflow error. The 11 discrete possible values for *range* (in volts) are: 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, and 10.

The actual setting for the range is the nearest legal value that is greater than or equal to the value specified by the *range* parameter. If a value is specified that is greater than the maximum range, the range is set to the maximum, 10.0 volts. This parameter may also be set with *e1431_set_analog_input*.

Reset Value

After a reset, the input *range* is set to the maximum of 10.0 volts.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_analog_input

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_replay_data_size
e1431_get_replay_data_size

Set size of replay input data
Get current replay data size

Synopsis

SHORTSIZ16 e1431_set_replay_data_size(SHORTSIZ16 ID, SHORTSIZ16 size)

SHORTSIZ16 e1431_get_replay_data_size(SHORTSIZ16 ID,
SHORTSIZ16 *size)

Description

e1431_set_replay_data_size sets the sample data size for replay data input to the value given in *size*.

e1431_get_data_size returns the current value of the replay sample data size, of a single channel or group of channels *ID*, into a memory location pointed to by **size*.

The replay data size controls the input data width to the DSPs during reprocessing and is independent of the output FIFO size set by *e1431_set_data_size*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

size is either E1431_DATA_SIZE_16 or E1431_DATA_SIZE_32.

Reset Value

After a reset, *size* is set to E1431_DATA_SIZE_16.

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_init_replay

e1431_replay_data

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_span
e1431_get_span

Set measurement span
Get measurement span

Synopsis

SHORTSIZ16 e1431_set_span(SHORTSIZ16 ID, FLOATSIZ32 span)

SHORTSIZ16 e1431_get_span(SHORTSIZ16 ID, FLOATSIZ32 *span)

Description

e1431_set_span sets the decimation bandwidth in terms of measurement bandwidth in Hertz. Calling *e1431_set_span* will result in the same settings as using *e1431_set_decimation_bandwidth*, but *e1431_set_span* can be easier to use.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

Decimation allows data reduction on oversampled data, saving only those points needed to reconstruct the waveform. A decimation of 2 keeps every other data point, a decimation of 4 keeps every fourth data point, etc. The bandwidth of the data must be reduced at the same time to prevent aliasing.

e1431_get_span returns the current span in Hertz. All channels of a module must have the same span.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

span can be one of the following: 25.6 kHz, 12.8 kHz, 6.4 kHz, 3.2 kHz, 1.6 kHz, 800 Hz, 400 Hz, 200 Hz, 100 Hz, 50 Hz, 25 Hz, 12.5 Hz, 6.25 Hz, 3.125 Hz, 1.56 Hz, 0.78 Hz, and 0.39Hz.

This parameter may also be set with *e1431_set_decimation_bandwidth*.

Reset Value

After a reset, *decBw* is set to 0, therefore leading to a full span of FS/2.56 = 25.6kHz

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_decimation_filter

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_time_offset
e1431_get_time_offset

Set input time offset correction value
Get input time offset correction value

Synopsis

```
extern  SHORTSIZ16 e1431_set_time_offset(SHORTSIZ16 chanID,
                                         SHORTSIZ16 decBW,
                                         SHORTSIZ16 value);
extern  SHORTSIZ16 e1431_get_time_offset(SHORTSIZ16 chanID,
                                         SHORTSIZ16 decBW,
                                         SHORTSIZ16 *value);
```

Description

e1431_set_time_offset sets the time offset value used by the DSPs to correct for delay from the input filters. Each channel has two values, one for top span, and one value for all other spans. The values are kept by the library and sent to the HP E1431A when *e1431_init_measure* is called.

e1431_get_input_offset retrieves the time offset value.

Since the function *e1431_auto_zero_and_phase* sets these time correction factors automatically, you will probably never need to use *e1431_set_input_offset* or *e1431_get_input_offset*.

chanID is the ID of a single channel.

decBW is the decimation value used when calling *e1431_set_decimation_bandwidth*.

value is a 16 bit signed number scaled at 0.3 nanoseconds per bit.

Reset Value

After a reset, all settings are 0, for no corrections.

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_auto_zero_and_phase

e1431_auto_zero

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_trigger

Set all trigger parameters, except auto arm and auto trigger

Synopsis

```
SHORTSIZ16  e1431_set_trigger(SHORTSIZ16 ID,
                              SHORTSIZ16 chanState,
                              LONGSIZ32 delay,
                              FLOATSIZ32 lowLevel,
                              FLOATSIZ32 highLevel,
                              SHORTSIZ16 slope,
                              SHORTSIZ16 mode)
```

Description

e1431_set_trigger sets all parameters associated with the trigger section of an HP E1431A or group of HP E1431As, except the auto trigger mode (see *e1431_set_auto_trigger*).

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

chanState determines if the channel or group of channels digitally triggers a measurement. E1431_CHANNEL_ON enables it to trigger a measurement. E1431_CHANNEL_OFF disables it from triggering the measurement. This parameter may also be set with *e1431_set_trigger_channel*.

delay is the elapsed time, in number of samples, between the occurrence of a trigger and the beginning of the data acquisition. A negative delay indicates a pre-trigger condition, where samples prior to the trigger event are included in the first measurement block. Possible values for the trigger delay range between -16383 (maximum pre-trigger delay) and 1M (maximum post-trigger delay). This parameter may also be set with *e1431_set_trigger_delay*.

lowLevel and *highLevel* are the value of the two trigger levels. Their value ranges from -128.67% to +128.67%. These trigger levels may also be set with *e1431_set_trigger_level*. When *mode* is set to E1431_TRIGGER_MODE_LEVEL, the difference between the two levels controls the amount of noise rejection, 10% is a good value to use here.

slope selects the edge or direction of the trigger source on which the trigger occurs. E1431_TRIGGER_SLOPE_POS sets it to a positive crossing of the highLevel, or to an exit of the bounded zone. E1431_TRIGGER_SLOPE_NEG sets it to a negative crossing of the lowLevel, or to an entry into the bounded zone. This parameter may also be set with *e1431_set_trigger_slope*.

mode selects the operating mode of the trigger detection. E1431_TRIGGER_MODE_LEVEL selects the positive or negative crossing of a unique trigger level. E1431_TRIGGER_MODE_BOUND selects the exit from or entry to a zone bounded by the two trigger levels. This parameter may also be set with *e1431_set_trigger_mode*.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Reset Value

After a reset, *chanState* is set to E1431_CHANNEL_ON, *delay* is set to 0, *mainLevel* and *secondLevel* are set to 0%, *slope* is set to E1431_TRIGGER_SLOPE_POS, and *mode* is set to E1431_TRIGGER_MODE_LEVEL.

Return Value

Return 0 if successful, a negative error number otherwise:

See Also

e1431_set_auto_trigger

e1431_set_trigger_channel

e1431_set_trigger_delay

e1431_set_trigger_level

e1431_set_trigger_mode

e1431_set_trigger_slope

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_trigger_channel
e1431_get_trigger_channel

Set a channel as trigger source
Get current trigger source

Synopsis

SHORTSIZ16 e1431_set_trigger_channel(SHORTSIZ16 ID,
SHORTSIZ16 chanState)

SHORTSIZ16 e1431_get_trigger_channel(SHORTSIZ16 ID,
SHORTSIZ16 *chanState)

Description

e1431_set_trigger_channel sets a single channel or group of channels, *ID*, to digitally trigger a measurement, depending on the value given in *chanState*.

e1431_get_trigger_channel returns the current value of the digital trigger source, of a single channel or group of channels *ID*, into a memory location pointed to by **chanState*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

chanState determines if the channel or group of channels can digitally trigger a measurement. E1431_CHANNEL_ON enables it to trigger a measurement. E1431_CHANNEL_OFF disables it from triggering the measurement. This parameter may also be set with *e1431_set_trigger*.

In addition to selecting a channel or group of channels as a source for triggering a measurement, it is possible to set up the following trigger parameters for each of the selected channels: the level may be set with *e1431_set_trigger_level*, the slope may be set with *e1431_set_trigger_slope*, and the mode may be set with *e1431_set_trigger_mode*.

A channel or group of channels which have been set to digitally trigger the measurement will implicitly do it on a local module basis. If it is required that these local trigger conditions translate into system wide (i.e. multi-modules) conditions, it is necessary to configure the module to act upon the system SYNC line (see *e1431_set_multi_sync*).

Reset Value

After a reset, *chanState* is set to E1431_CHANNEL_ON.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_multi_sync

e1431_set_trigger

e1431_set_trigger_level

e1431_set_trigger_mode

e1431_set_trigger_slope

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_trigger_delay
e1431_get_trigger_delay

Set trigger delay
Get current trigger delay

Synopsis

SHORTSIZ16 e1431_set_trigger_delay(SHORTSIZ16 ID, LONGSIZ32 delay)

SHORTSIZ16 e1431_get_trigger_delay(SHORTSIZ16 ID, LONGSIZ32 *delay)

Description

e1431_set_trigger_delay sets the trigger delay, of a single channel or group of channels *ID*, to the value given in *delay*.

e1431_get_trigger_delay returns the current value of the delay, of a single channel or group of channels *ID*, into a memory location pointed to by **delay*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

delay is the elapsed time, in number of samples, between the occurrence of a trigger and the beginning of the data acquisition. A negative delay indicates a pre-trigger condition, where samples prior to the trigger event are included in the first measurement block. Possible values for the trigger delay range between -16383 (maximum pre-trigger delay) and 1M (maximum post-trigger delay). The mqaximum pre-trigger delay is one less than the blocksize. This parameter may also be set with *e1431_set_trigger*.

Reset Value

After a reset, *delay* is set to 0.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_trigger

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_trigger_level
e1431_get_trigger_level

Set trigger level
Get current trigger level

Synopsis

SHORTSIZ16 e1431_set_trigger_level(SHORTSIZ16 ID,
SHORTSIZ16 whichLevel,FLOATSIZ32 level)

SHORTSIZ16 e1431_get_trigger_level(SHORTSIZ16 ID,
SHORTSIZ16 whichLevel,
FLOATSIZ32 *level)

Description

e1431_set_trigger_level sets one of the two trigger levels, of a single channel or group of channels *ID*, to the value given in *level*.

e1431_get_trigger_levels returns one of the two current trigger levels, of a single channel or group of channels *ID*, into a memory location pointed to by **levels*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

whichLevel selects one of the two trigger levels.
E1431_TRIGGER_LEVEL_LOWER sets the low level,
E1431_TRIGGER_LEVEL_UPPER sets the high level.

level is the value of the trigger level, expressed as a percentage of the current input range. Its value can be from -128.67% to +128.67%. After a reset, *lowLevel* and *highLevels* are set to 0%.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_trigger_mode

e1431_set_trigger

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_trigger_mode
e1431_get_trigger_mode

Set trigger mode (either level or bound)
Get current trigger mode

Synopsis

SHORTSIZ16 e1431_set_trigger_mode(SHORTSIZ16 ID, SHORTSIZ16 mode)

SHORTSIZ16 e1431_get_trigger_mode(SHORTSIZ16 ID, SHORTSIZ16 *mode)

Description

e1431_set_trigger_mode sets the trigger mode, of a single channel or group of channels *ID*, to the value given in *mode*.

Note

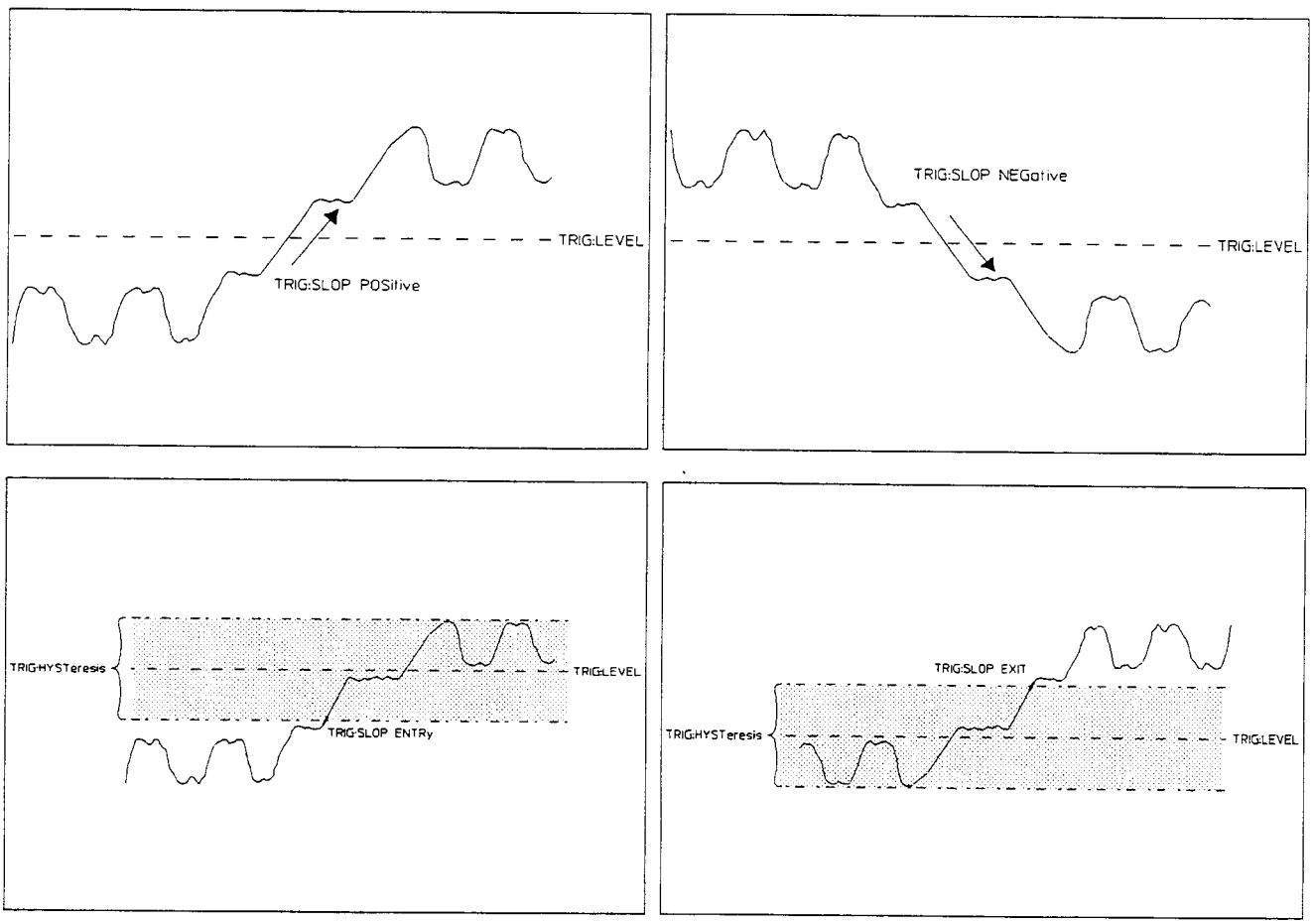
The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_trigger_mode returns the current value of the mode, of a single channel or group of channels *ID*, into a memory location pointed to by **mode*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

mode selects the area covered by the trigger detection. E1431_TRIGGER_MODE_LEVEL selects the positive or negative crossing of a unique trigger level. E1431_TRIGGER_MODE_BOUND selects the exit from or entry into a zone bounded by two trigger levels. See *e1431_set_trigger_slope*, for the direction which is effective. If the trigger slope is positive, the zone is defined as either crossing the level upwards, or exiting the zone. If the trigger slope is negative, the zone is defined as either crossing the level downwards, or entering the zone. This parameter may also be set with *e1431_set_trigger*.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



Reset Value After a reset, *mode* is set to E1431_TRIGGER_MODE_LEVEL.

Return Value Return 0 if successful, a negative error number otherwise.

See Also e1431_set_trigger

e1431_set_trigger_slope

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_trigger_slope
e1431_get_trigger_slope

Set slope of trigger
Get current slope of trigger

Synopsis

SHORTSIZ16 e1431_set_trigger_slope(SHORTSIZ16 ID, SHORTSIZ16 slope)

SHORTSIZ16 e1431_get_trigger_slope(SHORTSIZ16 ID, SHORTSIZ16 *slope)

Description

e1431_set_trigger_slope sets the trigger slope, of a single channel or group of channels *ID*, to the value given in *slope*.

e1431_get_trigger_slope returns the current value of the slope, of a single channel or group of channels *ID*, into a memory location pointed to by **slope*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

slope selects the edge or the direction of the source on which the trigger occurs. E1431_TRIGGER_SLOPE_POS sets it to a positive crossing of the level. E1431_TRIGGER_SLOPE_NEG sets trigger source to a negative crossing of the level. This parameter may also be set with *e1431_set_trigger*. If the trigger mode (see *e1431_set_trigger_mode*) is bound, positive is the same as exiting the zone defined by the two trigger levels, and negative is the same as entering it.

Reset Value

After a reset, *slope* is set to E1431_TRIGGER_SLOPE_POS.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_trigger

e1431_set_trigger_mode

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_try_recover

Control signal trapping

Synopsis

SHORTSIZ16 e1431_set_try_recover(SHORTSIZ16 STATE)

Description

e1431_set_try_recover controls the signal trapping functions of the HP E1431A library. If state is true (non zero), signals are trapped by the library and dealt with, usually by terminating. If state is false (zero) the library ignores the signal, which will allow user software to deal with the signal.

STATE true (non zero) to trap signals, false (zero) to ignore signals.

Reset Value

Default is TRUE, bus signal are trapped

Return Value

Returns 0

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_ttltrg_lines
e1431_get_ttltrg_lines

Select a pair of sync/clock lines
Get current selection of sync/clock lines

Synopsis

SHORTSIZ16 e1431_set_ttltrg_lines(SHORTSIZ16 ID, SHORTSIZ16 ttltrg)

SHORTSIZ16 e1431_get_ttltrg_lines(SHORTSIZ16 ID, SHORTSIZ16 *ttltrg)

Description

e1431_set_ttltrg_lines sets the system VXI ttltrg lines, of a single channel or group of channels *ID*, to the value given in *ttltrg*. If a measurement is in progress while calling this function, the measurement is aborted with *e1431_stop_measure*.

Note

The parameter set by this function is module dependent. Therefore, all channels of the same HP E1431A module are set with the same value, even if a channel ID is used.

e1431_get_ttltrg_lines returns the currently selected VXI ttltrg lines, of a single channel or group of channels *ID*, into a memory location pointed to by **ttltrg*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

ttltrg selects the pair of VXI ttltrg lines used for the system lines. In the following list, the first line is the one used for the SYNC line (that is, controlling measurement loop transitions: booting, synchronizing and settling, arming, triggering and measuring), and the second line is the one used for system clock.

E1431_TTLTRG_01 selects TTLTRG0 for SYNC, and TTLTRG1 for clock.
E1431_TTLTRG_23 selects TTLTRG2 for SYNC, and TTLTRG3 for clock.
E1431_TTLTRG_45 selects TTLTRG4 for SYNC, and TTLTRG5 for clock.
E1431_TTLTRG_67 selects TTLTRG6 for SYNC, and TTLTRG7 for clock.

Reset Value

After a reset, *ttltrg* is set to E1431_TTLTRG_01.

Return Value

Return 0 if successful, a negative error number otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_set_zoom
e1431_get_zoom

Set zoom state for replay only
Get current zoom state

Synopsis

SHORTSIZ16 e1431_set_zoom(SHORTSIZ16 ID, SHORTSIZ16 state)

SHORTSIZ16 e1431_get_zoom(SHORTSIZ16 ID, SHORTSIZ16 *state)

Description

e1431_set_zoom sets the zoom state, of a single channel or group of channels *ID*, to the value given in *state*. Zooming works only when reprocessing data. See *e1431_init_replay()*. If zoom is on, the center frequency set by *e1431_set_center_freq()* is used, otherwise baseband reprocessing takes place.

e1431_get_zoom returns the current value of the state, of a single channel or group of channels *ID*, into a memory location pointed to by **state*.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel.

state can be either E1431_ZOOM_ON or E1431_ZOOM_OFF

Reset Value

After a reset state is set to E1431_ZOOM_OFF.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_init_replay

e1431_replay_data

e1431_set_center_freq

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_trace_level

Enable/disable call tracing printout

Synopsis

void e1431_trace_level(SHORTSIZ16 level)

Description

Debugging is easier if you are able to trace the progression of calls made to and within the HP E1431A library. *e1431_trace_level* causes several different levels of call tracing to be displayed.

- Level 0 is no tracing
- Level 1 displays first level calls, meaning the major E1431_XXX type calls. Some E1431_XXX functions call other E1431_XXX functions. This becomes apparent once you start this type of debugging.
- Level 2 and Level 3 display increasingly smaller functions and would probably not be of interest to most users of the HP E1431A library.

level is an integer whose value is either 0 (no tracing), or a tracing level desired. 1 will probably do for most users.

Reset Value

After a reset, *trace_level* is set to 0, resulting in no call tracing.

Return Value

void.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_trigger_measure

Manually trigger, move HP E1431As from TRIGGER to MEASURE

Synopsis

SHORTSIZ16 e1431_trigger_measure(SHORTSIZ16 ID)

Description

e1431_trigger_measure moves all modules in the group from the TRIGGER state to the MEASURE state.

This function performs a "manual trigger", and does *not* need to be called if one of the modules in the group is set to "autotrigger"; or, any of the channels in the group is set as an active trigger source. See "The HP E1431A's Measurement Process" in Chapter 7, for a detailed description of the measurement states.

This function waits for all modules to be in the TRIGGER state, before proceeding further, and returns an error if this state is not reached after a limited amount of time. After the call to *e1431_trigger_measure* completes successfully, the measurement moves to the MEASURE state, and stays in the MEASURE state until the end of the measurement.

ID is either the ID of a group of channels that was obtained with a call to *e1431_create_channel_group*, or the ID of a single channel. If the measurement involves more than one module, it is mandatory that a *group ID* be used, rather than a *channel ID*. Using a group ID guarantees that all modules in the group "move" synchronously to the MEASURE state, whereas using a channel ID "moves" modules asynchronously.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_arm_measure

e1431_init_measure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

e1431_trigger_time_correction

Get time correction of trigger event

Synopsis

```
SHORTSIZ16  e1431_trigger_time_correction(USHORTSIZ16 decPhase,
                        SHORTSIZ16  groupORdec,
                        FLOATSIZ32  samplefreq,
                        FLOATSIZ32 *delay)
```

Description

e1431_trigger_time_correction calculates a time value in seconds which represents the time between the actual trigger event and the time of the triggered sample.

decPhase is a 16 bit word only available from the appended trailer data when *e1431_set_append_status* is on.

groupORdec is one of two possible pieces of information:

- a negative number is used as a channel group from which the time span is read

  or

- a positive number is taken to be the decimation as set by *e1431_set_decimation_bandwidth*.

samplefreq is the oversample frequency which is the sample frequency of the ADCs, or for convenience, *samplefreq* can be 0 which will cause *e1431_trigger_time_correction* to use the default sample frequency being generated by the HP E1431A hardware.

delay is a pointer which will return the time value in seconds.

Reset Value

Not applicable.

Return Value

Return 0 if successful, a negative error number otherwise.

See Also

e1431_set_append_status

e1431_set_decimation_bandwidth

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Errors

Analog Input Errors

Error Number	Description
1201	Illegal input mode
1202	Illegal anti-alias filter state
1203	Illegal analog input source
1204	Illegal input grounding
1205	Illegal coupling
1206	Illegal ranging down
1207	Illegal ranging up
1208	Range out of limits
1209	Offset out of limits
1210	Illegal active channel

Data / Transferring Data Errors

Error Number	Description
1211	Illegal append status state
1212	Illegal blocksize
1213	Illegal data size
1214	Illegal data mode
1215	Illegal data port
1216	Illegal channel order
1217	Illegal LBUS mode
1218	Illegal LBUS state
1219	FIFO overrun

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Digital Filter Errors

Error Number	Description
1220	Illegal filter output mode
1221	Illegal digital anti-alias filter state
1222	Illegal filter decimation bandwidth
1223	Illegal filter settling time
1224	Illegal zoom state
1225	Illegal center frequency
1226	Illegal ramp state

Timing Errors

Error Number	Description
1230	Illegal clock source
1231	Illegal system sync mode
1232	Illegal trigger/sync line
1233	Illegal sample clock master
1234	Illegal fast arm mode
1235	Illegal group measure state

Trigger Errors

Error Number	Description
1240	Pretrigger delay exceeds size of FIFO
1241	Illegal post trigger delay out of bounds
1242	Illegal trigger level
1243	Illegal trigger level selection
1244	Illegal trigger slope
1245	Illegal trigger mode
1246	Illegal trigger channel
1247	Illegal arm state (manual or auto)
1248	Illegal trigger state (manual or auto)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Group, Module, Channel Errors

Error Number	Description
1250	Channel group ID is not valid
1251	Group or Channel ID is not valid
1252	Illegal module count
1253	No E1431 module at logical address
1254	Invalid module option
1255	Channel ID is not valid
1256	Modules in group not in contiguous slots
1257	Number of modules exceeds limit of 255
1258	Expected group ID
1259	Expected channel ID

IO, Machine, Interrupt Errors

Error Number	Description
1260	Timeout on the status register state
1261	Bus error
1262	VXI register offset is illegal
1263	Illegal interrupt mask
1264	Illegal interrupt priority
1265	Error in system time function
1266	Unable to map A16 address space
1267	Unable to open SICL vxi interface

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Miscellaneous Errors

Error Number	Description
1270	Module parameter not the same across channel group
1271	Unable to malloc memory
1272	Unable to reallocate memory
1273	Signal vector failed
1274	Measure state machine not responding
1275	Autozero timed out waiting for data
1276	E1431 Internal library link list error
1277	Autozero offset DAC failure
1278	Illegal AC settling setting
1279	Illegal mode in function call
1290	Autophase could not zero phase

Calibration and Flash ROM Errors

Error Number	Description
1280	Illegal cal block specified
1281	Posible Hardware error
1282	Rom Vpp Range Error
1283	Rom Command Sequence Error
1284	Rom Byte Program Error
1285	Rom Block Erase Error
1286	Incorrect password
1287	Need either E1431_USER or E1431_FACTORY
1288	Illegal calibration state
1289	Selftest Failure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

10

Programming the HP E1431A with SCPI

SCPI (Standard Commands for Programmable Instruments) is an industry-standard instrument control language. SCPI builds on the IEEE 488.1 and 488.2 standards

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Getting Started

The HP E1431A supports two different SCPI environments:

- Compiled SCPI (C-SCPI)
  HP E1570A/B
  HP E1472A
- Interpreted SCPI in Slot 0 Command Module

Compiled SCPI

Compiled SCPI is intended for software developers with a working knowledge of the C programming language. C-SCPI allows you to achieve the high throughput of register-based cards with the ease of high-level programming. Advantages to using C-SCPI are:

- High throughput of register-based VXI instruments
- Easy to understand SCPI commands
- Use of industry standards (VXIbus and SCPI)
- Supports register-based, message-based, and HP-IB instruments
- Can use standard debugging tools
- C-SCPI preprocessor has a small command set

For more information on using C-SCPI, see the *HP Compiled SCPI User's Guide*. To find out how to install the C-SCPI drivers, see Chapter 1, "Installing the HP E1431A."

Interpreted SCPI in Slot 0 Command Module

Interpreted SCPI is a set of execution routines that run in a Slot 0 Command Module together with a run-time SCPI parser.

For information on how to install the interpreted SCPI drivers, see Chapter 1, "Installing the HP E1431A."

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SCPI Command Structure and Format

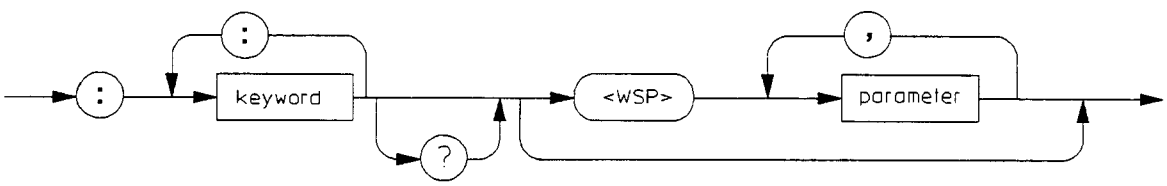
SCPI organizes related functions by grouping them together on a common branch of a command tree. Each branch is assigned a keyword to indicate the nature of the related functions. For example, the functions that control and monitor the status registers are grouped under the STATUS branch of the command tree. The STATUS branch is only one of the major SCPI branches which are called subsystems.

Colons indicate branching points on the command tree. A parameter is separated from the rest of the command by a space.

You can send multiple commands within a single message by separating commands with semicolons. One of the main functions of the command parser is to keep track of a program message's position in the command tree. If a program message contains two commands separated by a semicolon, the command parser assumes that the keywords of the second command come from the same branch of the tree as the final keyword of the preceding command. This allows you to simplify multiple command program messages.

Another way to simplify program messages is to delete implied mnemonics. You can omit some keywords from the command without changing the effect of the command. Implied mnemonics are identified by brackets [ ] in SCPI syntax diagrams.

The illustration below describes the basic syntax of SCPI commands.



NOTE:
WSP = whitespace. ASCII character (Decimal 0-9 or 11-32)

For additional information about SCPI command structure and format, see the *Beginner's Guide to SCPI*, available through your local Hewlett-Packard Sales Office.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Parameter Settings

As the illustration shows, there must be a <WSP>, whitespace or <space>, between the last command keyword and the first parameter in a command. This is one of the few places in SCPI where <space> is required. If you send more than one parameter with a single command, you must separate adjacent parameters using a comma.

Each parameter format has one or more corresponding response-data formats. For example, a setting that you program using a numeric parameter would return either floating point or integer response data when queried. Whether floating point or integer response data is returned, depends on the particular VXI module you are using. However, response data is clearly defined for the module and query. The next chapter, "SCPI Command Reference" specifies the data format for individual commands.

For additional information about SCPI data formats, see the *Beginner's Guide to SCPI*, available through your local Hewlett-Packard Sales Office.

Many commands are channel-dependent, meaning the parameter for each channel can be set independently. However, there are commands that affect all the channels in a single module. These are called *module-dependent commands*. In addition, there are commands that affect all the modules in a multiple-module configuration. These are called *instrument-dependent commands*.

Module-Dependent Commands

Commands that are module-dependent, change a parameter for all the channels of a module. The last channel specified for a module-dependent command, sets the parameter for all channels of the module. In a multiple-module configuration, it is recommended to specify all channels in the channel list of a module-dependent command. Otherwise, a parameter for a module will be set to the last channel specified and may result in an incorrect setting.

The following commands are examples of module-dependent commands:

- DATA:TRAller
- [SENSe]:SWEep:POINts
- TRIGger:DELay
- VINStrument[:CONFigure]:LBUS[:MODE]

For more information about channel lists, see the <Channels> listing under "Syntax Description" in the next chapter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Instrument-Dependent Commands

In SCPI, multiple modules in a card set are equivalent to an "instrument." SCPI commands that set a parameter for all channels of an instrument to the same value are called instrument-dependent commands. No matter how many modules are installed in the instrument, all channels are set with the same value. For more information on addressing multiple HP E1431A modules, see "Addressing the HP E1431A," later in this chapter.

The following commands are examples of instrument-dependent commands:

- DATA:SIZE
- FREQuency:SRATE
- INPut[:STATe]
- [SENSe:]SWEep:MODE

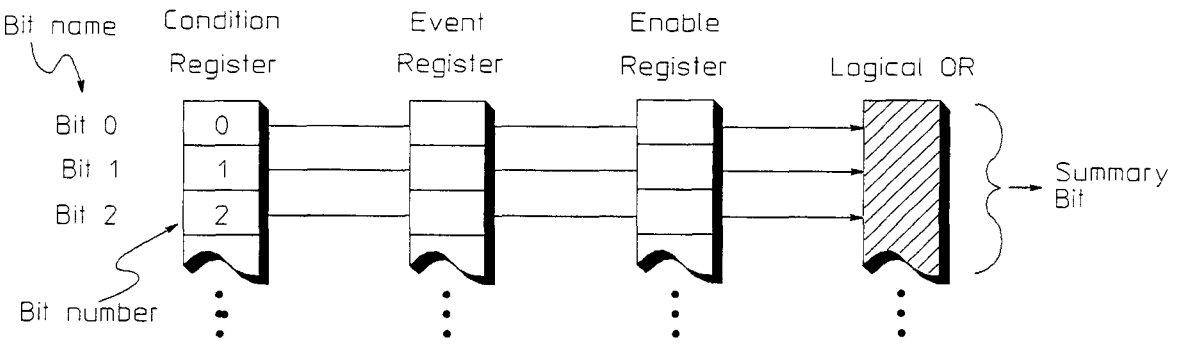
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Using the Status Registers

The HP E1431A's status registers contain information about various module conditions. The following sections describe the registers and tell you how to use them in your programs.

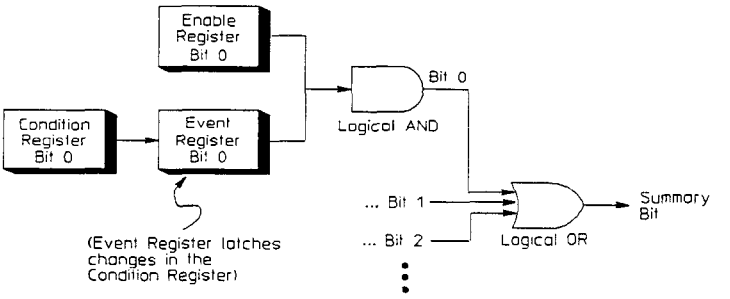
The General Status Register Model

The general status register model, shown below is the building block of the HP E1431A's status system. Most register sets in the module include all of the registers shown in the general model, although commands are not always available for reading or writing a particular register. The model consists of a condition register, an event register, and an enable register.



The flow within a status group starts at the condition register and ends at the register summary bit. (See the illustration below.) You control the flow by altering bits in the enable registers.

The Operation Status and Questionable Status groups are 16 bits wide, while the Status Byte and Standard Event groups are 8 bits wide. In the 16-bit groups, the most significant bit (bit 15) is not used. Bit 15 is always set to 0.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Condition Register

The condition register continuously monitors hardware and firmware status. It represents the current state of the module. It is updated in real time. When the condition monitored by a particular bit becomes true, the bit is set to 1. When the condition becomes false, the bit is reset to 0. Condition registers are read-only.

If there is no command to read a particular condition register, it is simply invisible to you.

Event Register

The event register records condition changes. When a change occurs in the condition register, the corresponding event bit is set to 1. Once set, an event bit is no longer affected by condition changes and subsequent events corresponding to that bit are ignored. The event bit remains set until the event register is cleared— either when the register is read or when the *CLS (clear status) command is sent. Event registers are read-only.

Note Reading the Event Register, clears the Event Register.

Enable Register

The enable register specifies which bits in the event register set a summary bit to 1. The module logically ANDs corresponding bits in the event and enable registers, and ORs all the resulting bits to determine the state of a summary bit. Summary bits are in turn recorded in the Status Byte. (The summary bit is only set to 1 if one or more enabled event bits are set to 1.) Enable registers are read-write.

Enable registers are cleared by *CLS (clear status). Querying enable registers does not affect them. There is always a command to read and write to the enable register of a particular register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

How to Use Registers

There are two methods you can use to access the information in register sets:

- The polling method
- The service request (SRQ) method

Use the polling method when:

- Your language/development environment does not support SRQ interrupts.
- You want to write a simple, single-purpose program and do not want to add the complexity of setting up an SRQ handler.

Use the SRQ method when:

- You need time-critical notification of changes.
- You are monitoring more than one device which supports SRQ.
- You need to have the controller do something else while it is waiting.
- You cannot afford the performance penalty inherent to polling.

The Polling Method

In the polling method, the module has a passive role. It only tells the controller that conditions have changed when the controller asks the right question. In the SRQ method, the module notifies the controller of a condition change without the controller asking. Either method allows you to monitor one or more conditions.

When you monitor a condition with the polling method, you must

1. Determine which register contains the bit that monitors the condition.
2. Send the unique SCPI query that reads that register.
3. Examine the bit to see if the condition has changed.

The polling method works well if you do not need to know about changes the moment they occur. The SRQ method is more effective if you must know immediately when a condition changes. To detect a change in a condition using the polling method, your program would need to continuously read the registers at very short intervals. This makes the program less efficient. In this case it is better to use the SRQ method.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The SRQ Method

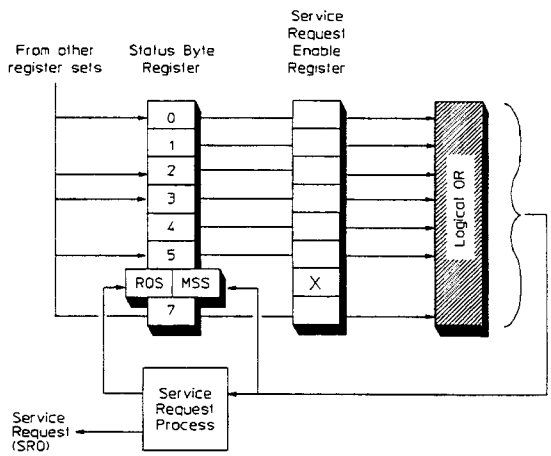
When you monitor a condition with the SRQ method, you must

1. Determine which bit monitors the condition.
2. Determine how that bit reports to the request service (RQS) bit of the Status Byte.
3. Send SCPI commands to enable the bit that monitors the condition and to enable the summary bits that report the condition to the RQS bit.
4. Enable the controller to respond to service requests.

When the condition changes, the module sets its RQS bit and the module's SRQ line. The controller is informed of the change as soon as it occurs. The time the controller would otherwise have used to monitor the condition can now be used to perform other tasks. Your program determines how the controller responds to the SRQ.

Generating a Service Request

To use the SRQ method, you must understand how service requests are generated. As shown below, other register sets in the module report to the Status Byte. Many of them report directly, but some may report indirectly.



Bit 6 of the Status Byte serves two functions; the request service function (RQS) and the master summary status function (MSS). The RQS bit changes whenever something changes that it is configured to report. The RQS bit is cleared when it is read with a serial poll. The MSS bit is set in the same way as the RQS bit. However, the MSS bit is cleared only when the condition that set it is cleared. The MSS bit is read with *STB?.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

When a register set causes its summary bit in the Status Byte to change from 0 to 1, the module can initiate the service request (SRQ) process. However, the process is only initiated if both of the following conditions are true:

- The corresponding bit of the Service Request enable register is also set to 1.
- The module does not have a service request pending.
- (A service request is considered to be pending between the time the module's SRQ process is initiated and the time the controller reads the Status Byte register with a serial poll.)

The SRQ process sets the SRQ line true. It also sets the Status Byte's request service (RQS) bit to 1. Both actions are necessary to inform the controller the module requires service. Setting the SRQ line only informs the controller that some device on the bus requires service. Setting the RQS bit allows the controller to determine which device requires service. That is, it tells the controller that this particular device requires service.

If your program enables the controller to detect and respond to service requests, it should instruct the controller to perform a serial poll when the SRQ line is set true. Each device on the bus returns the contents of its Status Byte register in response to this poll. The device whose RQS bit is set to 1 is the device that requested service.

Note

When you read the module's Status Byte with a serial poll, the RQS bit is reset to 0. Other bits in the register are not affected.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

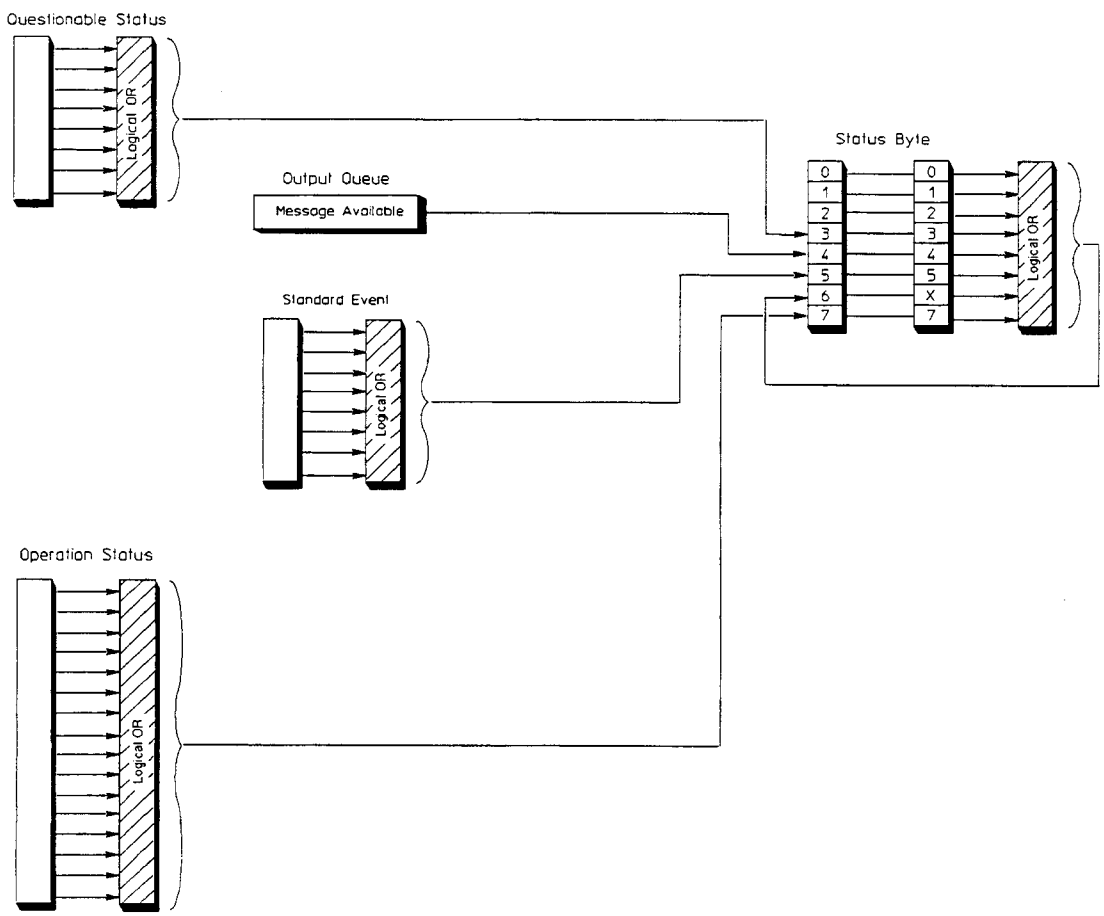
The HP E1431A Registers Sets

The HP E1431A uses four register sets to keep track of the module's status:

- Status Byte
- Questionable Status
- Standard Event
- Operational Status

Their reporting structure is summarized in the illustration below. They are described in greater detail in the following sections.

Register bits not explicitly presented in the following sections are not used in the HP E1431A. A query to one of these bits returns a value of 0.

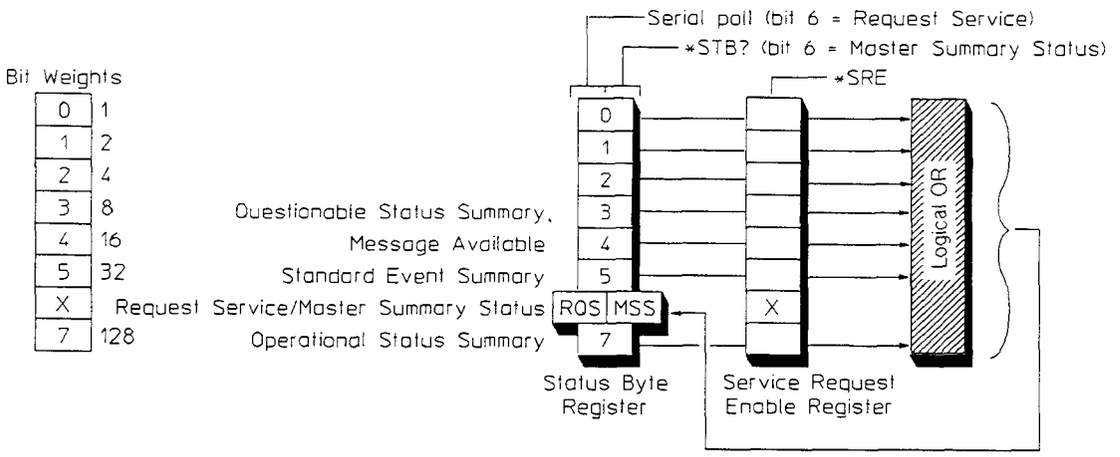


HP E1431A Register Sets

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Status Byte

The Status Byte summarizes the states of the other register sets and monitors the HP E1431A's output queue. It is also responsible for generating service requests (see "Generating Service Requests" earlier in this chapter).



The Status Byte is unique because it does not exactly conform to the general status model presented earlier. It contains only two registers: the Status Byte register and the Service Request enable register. The Status Byte registers behaves like a condition register for all bits except bit 6. The Service Request enable behaves like a standard enable registers except that bit 6 is always set to 0.

Bits in the Status Byte register are set to 1 under the following conditions:

- Questionable Status Summary (bit 3) is set to 1 when one or more enabled bits in the Questionable Status event register are set to 1.
- Message Available (bit 4) is set to 1 when the output queue contains a response message.
- Standard Event Summary (bit 5) is set to 1 when one or more enabled bits in the Standard Event event register are set to 1.
- Master Summary Status (bit 6, when read by *STB?) is set to 1 when one or more enabled bits in the Status Byte register are set to 1.
- Request Service (bit 6, when read by serial poll) is set to 1 by the service request process (see "Generating a Service Request" earlier in this chapter).
- Operation Status Summary (bit 7) is set to 1 when one or more enabled bits in the Operation Status event register are set to 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The illustration also shows the commands you use to read and write the Status Byte registers. The following statements are example commands using the Status Byte and Status Byte enable register.

***SRE 16** Generate an SRQ interrupt when messages are available in the output queue.

***SRE?** Find out what events are enabled to generated SRQ interrupts.

***STB?** Read the Status Byte event register.

See "Setting and Querying Registers" later in this chapter for more information about these commands.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

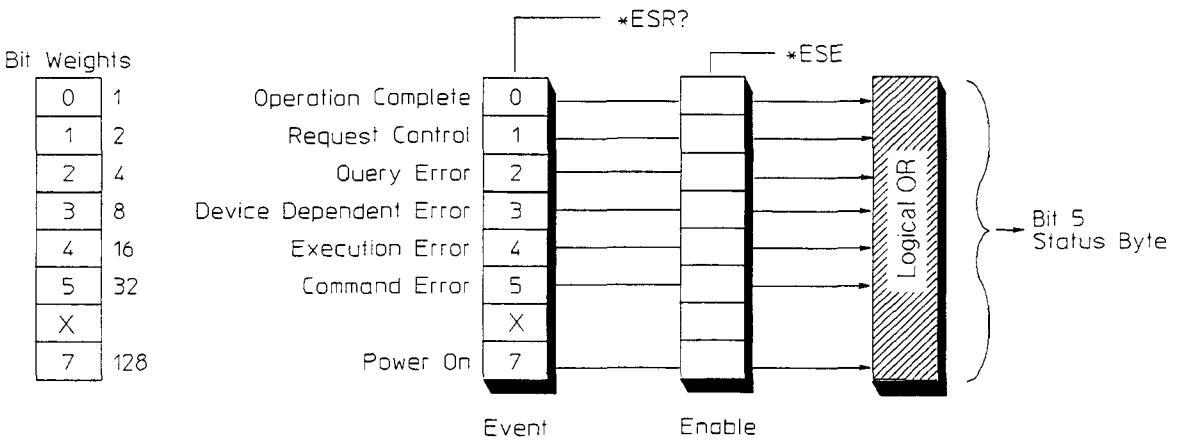
Questionable Status Register Set

The Questionable Status register is not implemented, but is included for compatibility with SCPI.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Standard Event Register Set

The Standard Event register set monitors module errors as shown below. It is one of the simplest and most frequently used. The unique aspect of this group is that you program it using common commands, while you program all other register sets through the STATUS subsystem.



The Standard Event Register set does not conform to the general status register model described at the beginning of this chapter. It contains only two registers: the Standard Event event register and the Standard Event enable register.

Bits in the Standard Event Status event register are set to 1 under the following conditions:

- Operation Complete (bit 0) is set to one when the following two events occur (in the order listed):
  - You send the *OPC command to the module.
  - The module completes all pending overlapped commands.
- Query Error (bit 2) is set to 1 when the module detects a query error.
- Device Dependent Error (bit 3) is set to 1 when the command parser or execution routines detect a device-dependent error.
- Execution Error (bit 4) is set to 1 when the command parser or execution routines detect an execution error.
- Command Error (bit 5) is set to 1 when the command parser detects a command or syntax error.
- Power On (bit 7) is set to 1 when you turn on the module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The illustration also shows the commands you use to read and write the Standard Event register sets. Example commands using Standard Event registers:

***ESE 20** Generate a summary bit whenever there is an execution or command error

***ESE?** Query the state of the standard Event enable register?

***ESR?** Query the state of the Standard Event event register.

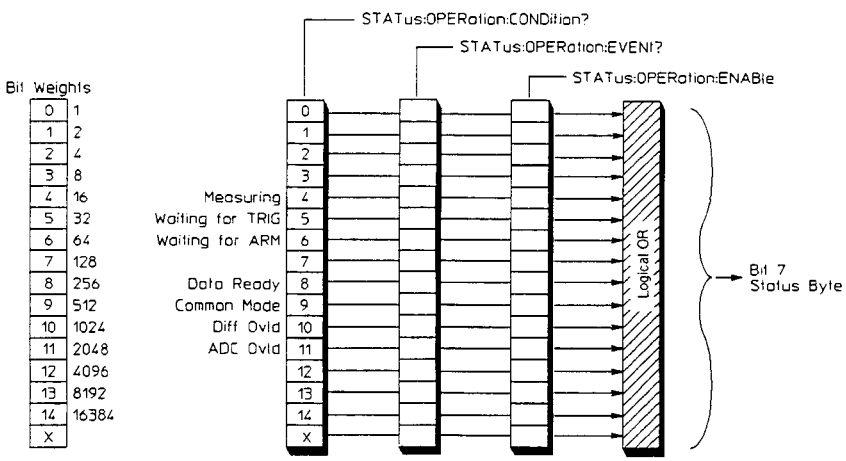
See "Setting and Querying Registers" later in this chapter for more information about using these commands.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Operation Status Register Set

The Operation Status register set monitors conditions in the module's measurement process.

This register set includes a condition register, an event register, and an enable register. It is accessed through the STATUS subsystem. See "Setting and Querying Registers" later in this chapter for more information about using these commands.



Bits in the Operation Status condition register are set to 1 under the following conditions:

- Measuring (bit 4) is set to 1 while the module is collecting data for a measurement.
- Waiting for TRIG (bit 5) is set to 1 when the module is ready to accept a trigger signal from one of the trigger sources. (If a trigger signal is sent before this bit is set, the signal is ignored.)
- Waiting for ARM (bit 6) is set to 1 when the module is ready to be armed. If you send the ARM:IMM command before this bit is set, the command is ignored.
- Data Ready (bit 8) is set when a complete block of data from all active channels, is available and ready on the VME data port or the Local Bus.
- Common Mode (bit 9) is set to 1 when the voltage at either side of the differential inputs gets too high or too low.
- Diff Ovld (bit 10) is set to 1 when the difference between the high and low sides of the differential inputs exceeds the current range.
- ADC Ovld (bit 11) is set to 1 when the ADC "clips," that is, exceeds the maximum level for the ADC.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Setting and Querying Registers

The previous register set illustrations include the commands you use to read from and to write to the registers. Most commands have a *set form* and a *query form*.

Use the set form of the command to write to a register. The set form is shown in the illustrations. The set form of a command takes an extended numeric parameter.

Use the query form of the command to read a register. Add a "?" to the set form to create the query form of the command. Commands ending with a "?" in the illustrations are query-only commands. These commands cannot set the bits in the register, they can only query or read the register.

The register set illustrations also include the bit weights you use to specify each bit in the register. For example, to get the Waiting for Trigger condition register (bit 5 in Operation Status register set) to generate a service request, send the following commands:

STATUS:PRESET Sets the Enable register bits in the Operational Status and the Questionable Status register sets to 0.

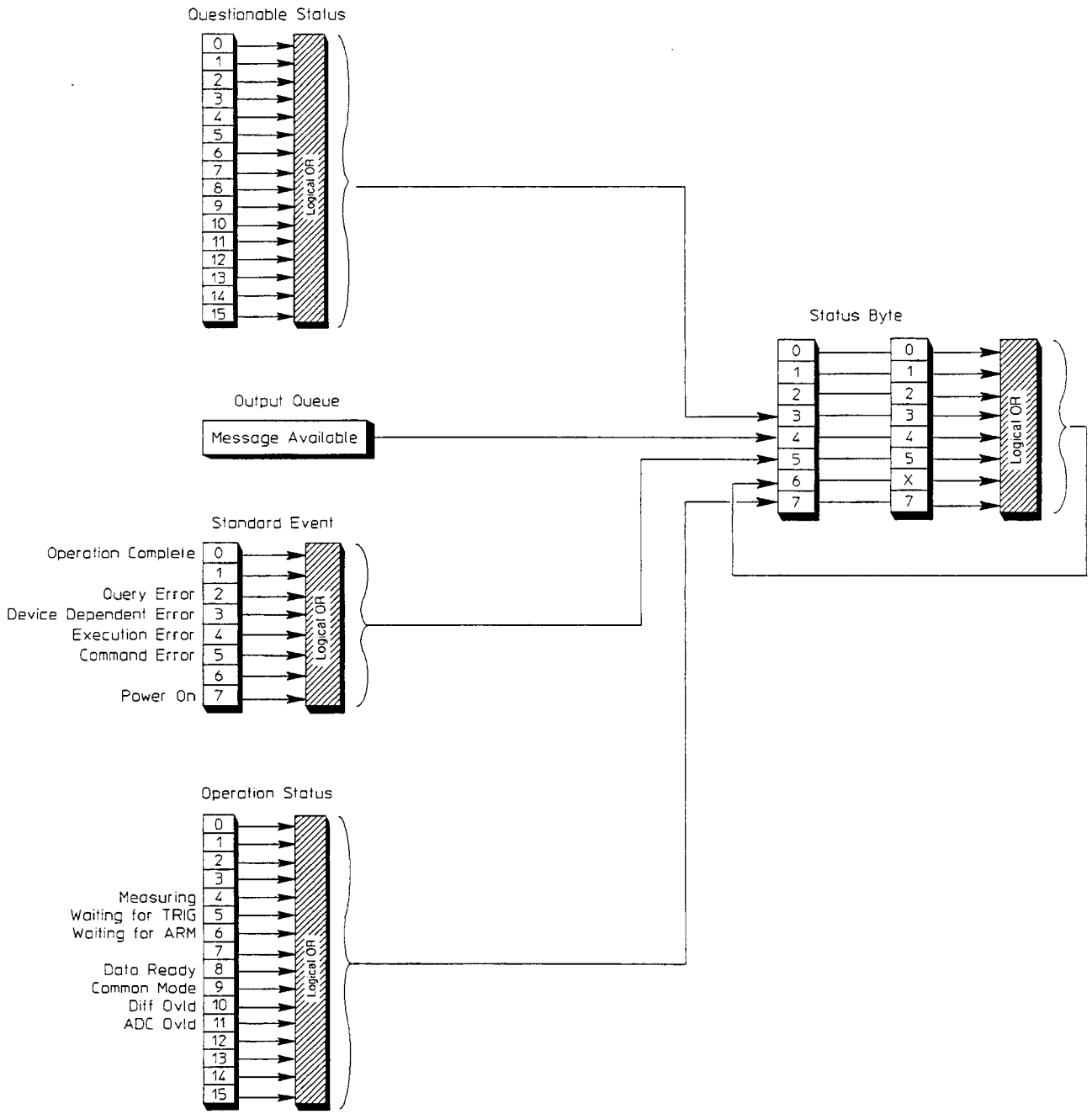
STATUS:OPERATION:ENABLE 32 Sets the Waiting for Trigger Enable register (bit 5) to 1.

***SRE 128** Sets bit 7 of the Service Request Enable register to 1.

See the next chapter for more information about these commands.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

HP E1431A Register Set Summary



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Addressing the HP E1431A

The HP E1431A address in a C-SCPI environment consist of 3 parts; an interface select code, the primary address and the secondary address.

The interface select code specifies the interface. Seven (7) is a typical number for the HP-IB interface.

The primary address, typically 09, indicates which HP-IB port in the system controller is used to communicate with the Slot 0 Control Module, for example the HP E1406A/B.

The secondary address indicates the device-specific address. In this case, it represents the VXI logical address.

The VXI logical address ranges in value from 1 to 255. The logical address to the HP-IB cannot be used directly, but must be encoded into the HP-IB secondary address. In addition, the logical address of the HP E1431A must be a multiple of 8, not including 0.

If the logical address is 8, the HP-IB secondary address is encoded to 1. If the logical address is 40, the secondary address is encoded to 5. In these examples, the HP-IB address is 70901 and 70905.

Software running in a computer writing to a Slot 0 Control Module needs to use all three addresses: the select code, the primary HP-IB address, and the secondary address.

Multiple HP E1431A modules are called card sets. When addressing multiple cards in a set, the address of the first card must be a multiple of 8. Additional cards must be logically contiguous to the first card. The logical address of the first card is used for all cards in the set, which SCPI considers an "instrument."

Addressing a Card versus a Channel

SCPI commands are addressed to an "instrument;" *not* to a channel on an individual card. You can not address a SCPI command to an individual channel.

However, you can specify a parameter setting for a specific channel using a channel list. For more information about using channel lists, see "Syntax Descriptions" in the next chapter.

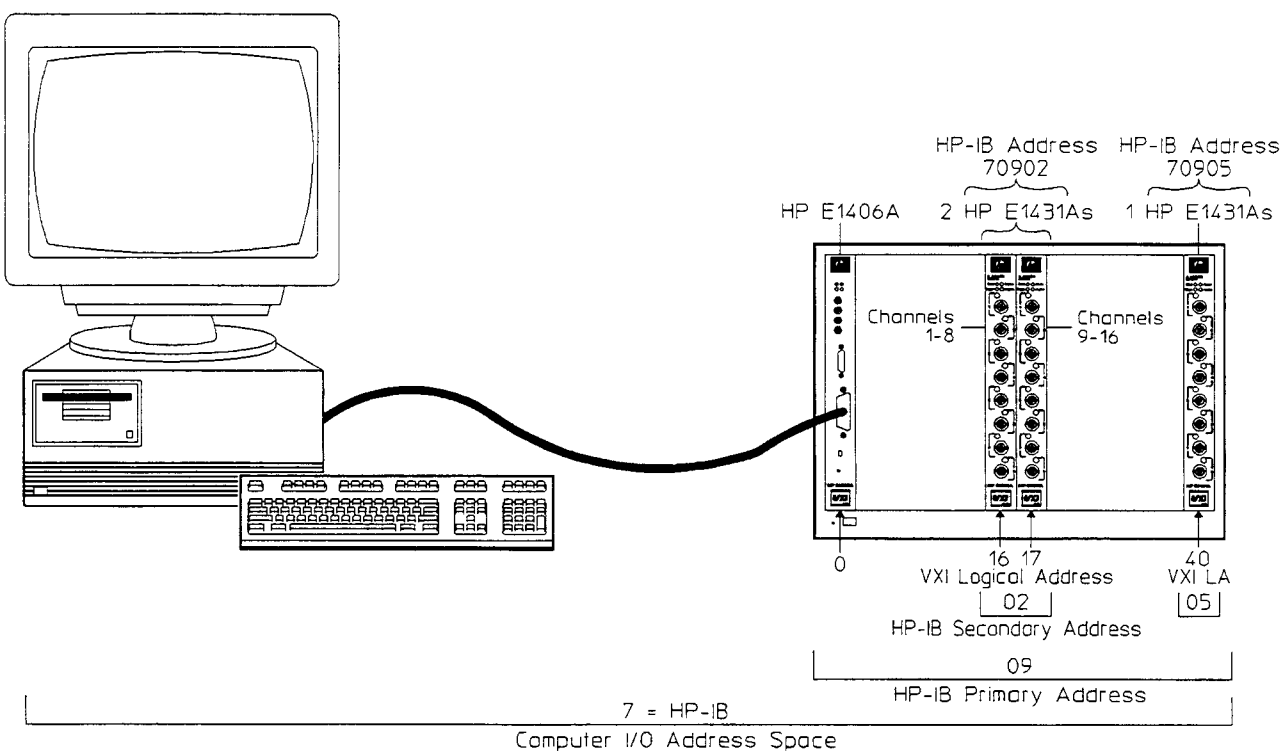
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Example

HP-IB Address of the HP E1431A Card Sets (instrument):
<select code><primary HP-IB address><secondary HP-IB address>

Interface Select Code: 7
HP E1406 Command Module at Primary HP-IB Address: 09
HP E1431A Card 1 at VXI Logical Address: 16
HP E1431A Card1 Secondary HP-IB Address: 02
HP E1431A Card 2 at VXI Logical Address: 17
HP E1431A Card 2 Secondary HP-IB Address: 02

HP-IB Address: 70902



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C-SCPI Example Program

This section contains an example program written in C-SCPI. The program creates a group containing two channels and makes a block measurement with the group.

```
/*  CSCPI example program
 *
 * This program takes one block of data from channel 1 and one from
 * channel 4.  It also autoranges and reads back the resulting range.
 *
 * One call to cscpi_exe with a constructed string is shown as well as
 * SCPI Preprocessor commands.
 *
 * See the HP Compiled SCPI Command Reference for further information.
 *
 */

#include <stdio.h>
#include <stdlib.h>
#include <cscpi.h>                  /* C-SCPI header file */

#define BLOCK_SIZE  10
#define NUM_CHANS  2

/* forward reference */
static void error_check();

/* declare the instrument */
INST_DECL(vm,"E1431",REGISTER);

void main()
{
float32 data[NUM_CHANS][BLOCK_SIZE];/* storage for result data */
int32 channum[NUM_CHANS];           /* channel number list */
int32 i,b;                          /* loop indexes */
double ranges[BLOCKxes              */
double ranges[BLOCK_SIZE];          /* ranges read back from E1431 */
char sendstring[100];               /* string sent using cscpi_exe */
char resultstring[100];             /* string received
*
 /* start up CSCPI */
                    INST_STARTUP();
 /* assume E1431 at VXI local address 16 */
                    INST_OPEN(vm,"vxi,16");
 /* check if opened */
                    if ( vm==0)
                    {
                     printf("open failed on vm\n");
                     exit(1);
                    }
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
/* start from a known point (reset) */
                INST_SEND(vm,"*RST\n");
                error_check();
/* turn on desired channels */
                INST_SEND(vm,"inp off,(@1:8)\n");  /* all off */
                error_check();
                INST_SEND(vm,"inp on,(@1,4)\n"); /* 1 and 4 on */
                error_check();
/* zero out the DC offset of the inputs */
                INST_SEND(vm,"inp:offs:auto once");
                error_check();
/* auto range to get good readings */
                INST_SEND(vm,"volt:rang:auto once,0.1,(@1,4)\n");
                error_check();
/* set block size to 10 for this example */
                INST_SEND(vm,"swe:poin 10\n");
                error_check();
/* set format */
                INST_SEND(vm,"form real,32\n");
                error_check();
/* set manual arm and trigger for demonstration , default is IMM */
                INST_SEND(vm,"arm:sour man\n");
                error_check();
                INST_SEND(vm,"trig:sour man\n");
                error_check();
/* initialize E1431 for data readings */
                INST_SEND(vm,"init\n");
                error_check();
/* arm the module */
                INST_SEND(vm,"arm\n");
                error_check();
/* trigger the module */
                INST_SEND(vm,"*trg\n");
                error_check();
/* read a block for each channel */
                for(i=0;i<NUM_CHANS;i++
                {
                /* get channel number */
                 INST_QUERY(vm,"data:bnum?\n","%d",&(channum[i]));
                 error_check();
                /* query for current range setting */
                /* build command using channel number */
                 sprintf(sendstring,"volt:rang? (@%d)\n",channum[i]);
                /* send command to E1431 */
                 cscpi_exe(vm,sendstring,strlen(sendstring),
                                resultstring,sizeof(resultstring));
                 error_check();
                /* convert ascii string to number */
                 ranges[i]=atof(resultstring);
                /* read voltage data for one channel */
                 INST_QUERY(vm,"data?\n","%,BLOCK_SIZEf",&(data[i][0]));
                 error_check();
                }
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

```
  /* print the collected data */
  /* print header */
 printf("point ");
                    for(i=0;i<NUM_CHANS;i++
 {
                    printf("chan %1d rng=%6.3f  _ ",channum[i],ranges[i]);
 }
 printf("\n");

  /* print data */
 for(b=0;b<BLOCK_SIZE;b++)
 {
                    printf("%3d - ",b);
 for(i=0;i<NUM_CHANS;i++
 {
                     printf("%8.4f    [b]);
 }
 printf("\n");
 }

}

/*
 *  error checking and reporting
 */
void error_check()
{
int16               errnum;
char                errstring[100];

                    INST_QUERY(vm,"syst:err?\n","%d,%s",&errnum,errstring);
                    if(errnum)
                    {
                     printf("ERROR %d: %s\n",errnum,errstring);
                    }
}
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SCPI Quick Reference

Common Commands

Command	Description
*CLS	Clears the Status Byte
*ESE 0 ~ 255	Sets or queries bits in the Standard Event Enable register
*ESR?	Reads and clears the Standard Event register
*IDN?	Returns module's identification string
*OPC	Enables the status bit too be set for compeltion of overlapped commands
*OPC?	Queries completion of overlapped commands
*RST	Executes a device reset
*SRE 0 ~ 255	Sets or queries bits in the Service Request Enable register
*STB?	Reads Status Byte register
*TRG	Triggers measurement
*TST?	Performs self test; returns 0 = good 1 = bad
*WAI	Wait-to-continue command

Digital Filter Control

Command	Description
CALCulate:FILTer[:GATE]:TIME value	Digital AA filter settling: 0 to 255 points or time (PNT or S)
CALibration:STATe ON | OFF,(@cl)	Corrections: On/Off
DATA:TRAIler ON | OFF	Status trailer: On/Off
INPut[:STATe] ON | OFF,(@cl)	Channel: On/Off
[SENSe:]FREQuency:SPAN value,(@cl)	Span: 390 mHz to 25.6 kHz
[SENSe:]FREQuency:SRATE value,(@cl)	Sample rate: 0.5 Hz to 32.768 kHz
[SENSe:]FILTer[:LPASs][:STATe] ON | OFF,(@cl)	Digital AA filter: On/Off

@cl = channel list

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Inputs

Command	Description
INPut:BIAS[:STATE] ON | OFF,(@cl)	ICP: On/Off
INPut:COUPling AC | DC | GROund,(@cl)	INPUT coupling: AC/DC/GND
INPut:FILTer[:LPASs][:STATe] ON | OFF,(@cl)	Analog AA filter: On/Off
INPut:LOW FLOat | GROund,@cl)	BNC shell grounding: Floating/Gnd
INPut:PHAse:AUTO	Automatically performs autozero and phase calibration
[SENSe:]VOLTage[:DC]:RANGe[:UPPer] n,(@cl)	Set Range: 0.005 to 10.0

Measurement / Trigger

Command	Description
ARM[:IMMediate]	Arms the trigger if ARM:SOURce is manual
ARM:SOURce IMMediate | MANual	AutoArm On/Off
DATA:SIZE n	FIFO width: 16 | 32
INITiate[:IMMediate]	Prepares instrument to acquire data
[SENSe]:SWEep:MODE BLOCk | CONTinuous	Measeurment mode: Continuous/Single
[SENSe]:SWEep:POINts n	Blocksize: 1-16384
TRIGger:DELay value,(@cl)	Trigger delay time or points (S or PNT)
TRIGger:HYSTeresis n,(@cl)	Trigger hysteresis percent
TRIGger:IMMediate	Triggers the instrument
TRIGger:LEVel n,(@cl)	Trigger level volts
TRIGger:MODE LEVel | BOUND,(@cl)	Trigger type: level/bounded
TRIGger:SLOPe POS | NEG | ENTRy | EXIT,(@cl)	Trigger slope: positive/negative/entry/exit
TRIGger:SOURce MANual | IMMediate	AutoTrigger: On/Off
TRIGger[:STATe] ON | OFF,(@cl)	Trigger: On/Off

@cl = channel list

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Status Reporting

Command	Description
STATus:OPERation[:EVENT]?	Reads and clears the Operation Status event register
STATus:OPERation:CONDition?	Reads the Operation Status event register
STATus:OPERation:ENABle	Sets and queries bits in the Operation Status event register
STATus:OPERation[:EVENT]?	Reads and clears the Operation Status event register
STATus:PRESet	Sets bits in the enable registers to their default status
STATus:QUEStionable:CONDition?	Reads and clears the Questionable Status condition register
STATus:QUEStionable:ENABle	Sets and queries bits in the Questionable Status event register
STATus:QUEStionable[:EVENT]?	Reads and clears the Questionable Status event register
SYSTem:ERRor?	Returns one error message from the error queue

System Control

Command	Description
DATA:BNUMber?	What channel will be read next?
DATA[:DATA]?	Reads Data from HP 1431A
DATA:SCALe:FACTor? (@cl)	Get Scaling
DATA:SCALe[:STATe]	Enables/disables raw data scaling into volts
FORMat[:DATA] ASCii | REAL | PACKed [,32 | 64]	Format
INPut:OFFSet:AUTO ONCE	Autozero
[SENSe:]VOLTage:[DC:]RANGe:AUTO ONCE,t,(@cl)	Autorange for t time
SYSTem:VERSion?	Returns SCPI version to which module complies
VXI:CONFigure:CNUMber?	Return maximum channel number

@cl = channel list

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VXI Bus Signals

Command	Description
OUTPut:TTLTrg T01 | T23 | T45 | T67	TTL trigger: 01/23/45/67
VINStrument[:CONFigure]:LBUS[:MODE] APPend | GENerate | INSert | PIPeline,(@cl)	LBUS Mode: append/generate/insert/pipeline
VINS[:CONF]:PORT LBUS | VME,(@cl)	FIFO output: VME/LBUS
VINS:LBUS:RES (@cl)	Reset Lbus

Diagnostics

Command	Description
DIAGnostic:ACONfiguration ON | OFF	Sets up multiple modules to operate together
DIAG:ACSettling ON | OFF,@cl	Determines if the module waits for input transients to decay
DIAGNOSTIC:BSERial	Changes seerial number of module
DIAGnostic:INPut:OFFSet:COARse n,@cl	Sets input offset adjust of DAC(coarse)
DIAGnostic:INPut:OFFSet:FINE n, @cl	Sets input offset adjust of DAC (fine)
DIAGnostic:LBUS:RESet ON | OFF @cl	Controls the state of local bus interface circuit
DIAGnostic:REGister @cl	Read or write to individual registers
DIAGnostic:SCLock:INPut INT | EXT @cl	Chooses the source of the sample clock
DIAGnostic:SCLock:OUTPut ON | OFF @cl	Sets the master sample clock
DIAGnostic:SYNC INT | EXT @cl	Selects the shared SYNC line

@cl = channel list

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

11

SCPI Command Reference

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The Command Reference chapter describes all of the HP E1431A's SCPI commands. Each command has the following:

1. A brief description of the command. This one- or two-line description appears just below the heading.
2. A syntax description. This consists of two fields. One field specifies whether the command has only a command form, only a query form, or both. The other field shows you the syntax expected by the command parser. A detailed description for the elements appearing in the syntax description follows this section. For additional information about message syntax see the *Beginner's Guide to SCPI*, available through your local Hewlett-Packard Sales Office.
3. Example statements. This field appears at the end of the syntax description. It contains a C-SCPI command and an HP BASIC output statement that use the command.
4. A return format description. This field is only used if the command has a query form. It tells you how data is returned in response to the query.
5. An attribute summary. This field identifies overlapped commands requiring synchronization; defines the command's preset state and specifies compliance with SCPI. A "confirmed" command complies with SCPI 1994. An "approved" command complies with SCPI 1995. An "instrument-specific" command does not conform to the SCPI standard.
6. A detailed description. This field contains additional information about the command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Finding the Right Command

- If you can not find a command you have seen in a program, remember that commands can omit implied mnemonics.

  For example, the command SENSe:FREQuency:SPAN 1024 HZ contains the implied mnemonic SENSe. SENSe can be omitted to create the equivalent command FREQuency:SPAN 1024 HZ. (See "Implied Mnemonics" in the *Beginner's Guide to SCPI*.) You will not find an entry for FREQuency:SPAN—or any other command that omits an implied mnemonic—in the Command Reference. You will find the FREQuency:SPAN command with the SENSe commands.
- If you do not find a command where you expect it, try scanning the quick reference tables that begin on page 10-25 for the equivalent command that contains the implied mnemonic.

  Each command has a brief description. After you locate the equivalent command, you can find a more detailed description in the command reference.
- If you are looking for a command that accesses a particular function, use the index.

  For example, if you want to find the command that changes the sample rate, look for "sample rate" in the index. It sends you to the page that describes the SENSe:FREQuency:SRATe command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Command Syntax

This section describes the syntax elements used in the SCPI command reference. It also describes the general syntax rules for both kinds of command and query messages.

Note

For a more detailed discussion of message syntax, including example program listings, refer to the *Beginner's Guide to SCPI* available through your local Hewlett-Packard Sales Office.

Special Syntactic Elements

Several syntactic elements have special meanings:

- colon (:) — When a command or query contains a series of keywords, the keywords are separated by colons. A colon immediately following a keyword tells the command parser that the program message is proceeding to the next level of the command tree. A colon immediately following a semicolon tells the command parser that the program message is returning to the base of the command tree.
- semicolon (;) — When a program message contains more than one command or query, a semicolon is used to separate them from each other. For example, if you want to autorange inputs and then start a measurement using one program message, the message would be:

```
SENSE:VOLT:RANGE:AUTO ONCE,0.1,(@1,4);:INITIATE:IMMEDIATE
```

- comma (,) — A comma separates the data sent with a command or returned with a response. For example, the SENSE:VOLT:DC:RANGE:AUTO command requires two values to select the best range for the current input signal: one for the number of times to autorange, once; and one for the length of time in seconds. A message to autorange the inputs for 1.5 seconds would be:

```
SENSE:VOLT:DC:RANGE:AUTO ONCE,1.5
```

- <WSP> — One white space is required to separate a program message (the command or query) from its parameters. For example, the command "SENSE:VOLT:DC:RANGE:AUTO ONCE,1.5" contains a space between the program header (SENSE:VOLT:DC:RANGE:AUTO) and its program data (ONCE,1.5). White space characters are not allowed within a program header.

For more information, see the *Beginner's Guide to SCPI* available through your local Hewlett-Packard Sales Office.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Conventions

Syntax and return format descriptions use the following conventions:

- < > Angle brackets enclose the names of items that need further definition. The definition will be included in accompanying text. In addition, detailed descriptions of these elements appear at the end of this section.
- ::= "is defined as" When two items are separated by this symbol, the second item replaces the first in any statement that contains the first item. For example, A::=B indicates that B replace A in any statement that contains A.
- | "or" When items in a list are separated by this symbol, one and only one of the items can be chosen from the list. For example, A|B indicates that A or B can be chosen, but not both.
- ... An ellipsis (trailing dots) is used to indicate that the preceding element may be repeated one or more times.
- [ ] Square brackets indicate that the enclosed items are optional.
- { } Braces are used to group items into a single syntactic element. They are most often used to enclose lists and to enclose elements that are followed by an ellipsis.

Although the command interpreter is not case sensitive, the case of letters in the command keyword is significant in the Command Reference. Keywords that are longer than four characters can have a short form or a long form. SCPI accepts either form. Upper-case letters show the short form of a command keyword. For more information, see the *Beginner's Guide to SCPI*.

SCPI is sensitive to white space characters. White space characters are not allowed within command keywords. They are only allowed when they are used to separate a command and a parameter.

A message terminator is required at the end of a program message or a response message. Use <NL>, <^END>, or <NL> <^<END> as the *program message terminator*. The word <NL> is an ASCII new line (line feed) character. The word <^END> means that End or Identify (EOI) is asserted on the HP-IB interface at the same time the preceding data byte is sent. Most programming languages send these terminators automatically. For example, if you use the HP BASIC OUTPUT statement, <NL> is automatically sent after your last data byte. If you are using a PC, you can usually configure your system to send whatever terminator you specify.

For more information about terminating messages, see the *Beginner's Guide to SCPI*.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Syntax Descriptions

Syntax descriptions in the SCPI command reference chapter use the following elements:

<Channels> This item designates channel lists. Channel lists are used to specify the input channels on a single
HP E1431A module, or the input channels of multiple HP E1431A card sets. The syntax for <Channels> is:

<Channels> ::= (@<Channel_Range>,<Module_Channel>)
<Channel_Range> ::= <First_Channel_Specifier>:<Second_Channel_Specifier>
Channel_Specifier ::= Integer
limits: dependent upon system configuration
both instances of <Channel_Specifier> must contain
the same number of dimensions
<Module_Channel> ::= <Module_Channel>(<Channel_Range>,
<Channel_Range>)

Example sense:voltage:dc:range:upper 0.5V (@1,3,4:6) sets the input range for channels 1, 3, 4, 5, and 6 to 0.5 volts.

In query-only commands, channel lists can specify only one channel. For example, to determine if channel 3 is activated to acquire data send "INP? (@3)".

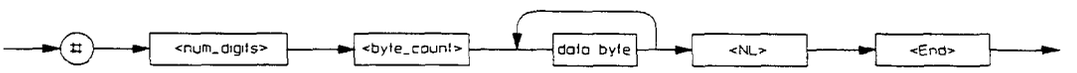
<CHAR> This item designates a string of ASCII characters. There are no delimiters. Usually, the string is from an explicit set of responses. Maximum length is 12 characters.

<DEF_BLOCK> This item designates definite-length-block data which takes the following form:

<DEF_BLOCK> ::= #<byte><length_bytes><data_byte>[<data_byte>] . . .
<byte> ::= number of length bytes to follow (ASCII encoded)
<length_bytes> ::= number of data bytes to follow (ASCII encoded)
<data_byte> ::= unsigned 8-bit data

If the data is ASCII encoded (FORMat:DATA ASCii command):

<DEF_BLOCK> ::= floating-point-numeric[,floating-point-numeric] . . . <NL>



See the *Beginner's Guide to SCPI* for more information about block data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<STRING> This item specifies any 8-bit characters delimited by single quotes or double quotes. The beginning and ending delimiter must be the same. If the delimiter character is in the string, it must be entered twice. (For example, to get "EXAMPLE", enter ""EXAMPLE"".

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*CLS command

Clears the Status Byte by emptying the error queue and clearing all event registers.

Command Syntax: *CLS

Example Statements:
```
INST_SEND(vm,"*cls")
OUTPUT 70902;"*CLS"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This command clears the Status Byte register. It does so by emptying the error queue and clearing (setting to 0) all bits in the event registers of the Questionable Status, Standard Event, and Operationa Status register sets:

In addition, *CLS cancels any preceding *OPC command or query. This ensures that bit 0 of the Standard Event register will not be set to 1 and that a response will not be placed in the module's output queue when pending overlapped commands are completed.

*CLS does not change the current state of enable registers or transition filters.

Note To guarantee that the Status Byte's Message Available and Master Summary Status bits are cleared, send *CLS immediately following a Program Message Terminator.

See "The HP E1431A's Register Sets" in the previous chapter for more information on the Status Byte register.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*ESE command/query

Sets bits in the Standard Event enable register.

Command Syntax: *ESE <Event Status Register>

<Event Status Register> ::= number
limits: 0:255

Example Statements: INST_SEND(vm,"*ese 128")
OUTPUT 70902;"*ESE 60"

Query Syntax: *ESE?

Return Format: Integer

Attribute Summary: Preset State: +0
SCPI Compliance: confirmed

Description: This command allows you to set bits in the Standard Event enable register. Assign a decimal weight to each bit you want set (to 1) according to the following formula:

$$2^{(bit_number)}$$

with acceptable values for bit_number being 0 through 7. Add the weights and then send the sum with this command.

When an enable register bit is set to 1, the corresponding bit of the Standard Event event register is enabled. All enabled bits are logically ORed to create the Standard Event summary, which reports to bit 5 of the Status Byte. Bit 5 is only set to 1 if both of the following are true:

- One or more bits in the Standard Event event register are set to 1.
- At least one set bit is enabled by a corresponding bit in the Standard Event enable register.

The query returns the current state of the Standard Event enable register. The state is returned as a sum of the decimal weights of all set bits.

For more information on the Standard Event register set, see "HP E1431A Register Sets" in the previous chapter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*ESR? query

Reads and clears the Standard Event event register.

Query Syntax: *ESR?

Example Statements:
```
INST_SEND(vm,"*esr?")
OUTPUT 70902;"*ESR?"
```

Return Format: Integer

Attribute Summary: Preset State: +0
SCPI Compliance: confirmed

Description: This query returns the current state of the Standard Event event register. The state is returned as a sum of the decimal weights of all set bits. The decimal weight for each bit is assigned according to the following formula:

$$2^{(bit_number)}$$

with acceptable values for bit_number being 0 through 7.

The query clears the register after it reads the register.

A bit in this register is set to 1 when the condition it monitors becomes true. A set bit remains set, regardless of further changes in the condition it monitors, until one of the following occurs:

- You read the register with this query.
- You clear all event registers with the *CLS command.

For more information on the Standard Event register set, see "HP E1431A Register Sets" in the previous chapter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*IDN? query

Returns a string that identifies the module.

Query Syntax: *IDN?

Example Statements:
```
INST_SEND(vm,"*idn?")
OUTPUT 70902;"*idn?"
```

Return Format: HEWLETT-PACKARD,E1431A,<serial_number>,<software_revision>

Attribute Summary: Preset State: instrument dependent
SCPI Compliance: confirmed

Description: The response to this query uniquely identifies your module and indicates the version of the module's software.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*OPC command/query

Sets or queries completion of all pending overlapped commands.

Command Syntax: *OPC

Example Statements:
```
INST_SEND(vm,"*opc")
OUTPUT 70902;"*OPC"
```

Query Syntax: *OPC?

Return Format: Integer

Attribute Summary:
Preset State: not applicable
SCPI Compliance: confirmed

Description: There are no overlapped commands at this time. *OPC is included in the command set to be compatible with SCPI.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*RST command

Executes a device reset.

Command Syntax: *RST

Example Statements:
```
INST_SEND(vm,"*rst")
OUTPUT 70902;"*RST?"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This command returns the module to a reset state. In addition, *RST cancels any pending *OPC command or query.

The reset state is the same as the preset state. The preset state of each command is listed in the Attribute Summary.

The following are *not* affected by this command:

- The error queue.
- The state of all enable registers.
- Calibration constants.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*SRE command/query

Sets bits in the Service Request enable register.

Command Syntax: *SRE <Status Enable Register>

<Status Enable Register> ::= number
limits:0:255

Example Statements:
```
INST_SEND(vm,"*sre 40")
OUTPUT 70902;"Sre 17"
```

Query Syntax: *SRE?

Return Format: Integer

Attribute Summary: Preset State: +0
SCPI Compliance: confirmed

Description: This command allows you to set bits in the Service Request enable register. Assign a decimal weight to each bit you want set (to 1) according to the following formula:

$$2^{(bit_number)}$$

with acceptable values for bit_number being 0 through 7. Add the weights and then send the sum with this command.

Note The module ignores the setting you specify for bit 6 of the Service Request enable register. This is because the corresponding bit of the Status Byte register is always enabled.

The module requests service from the active controller when one of the following occurs:

- A bit in the Status Byte register changes from 0 to 1 while the corresponding bit of the Service Request enable register is set to 1.
- A bit in the Service Request enable register changes from 0 to 1 while the corresponding bit of the Status Byte register is set to 1.

The query returns the current state of the Service Request enable register. The state is returned as a sum of the decimal weights of all set bits.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*STB? query

Reads the Status Byte register.

Query Syntax: *STB?

Example Statements:
```
INST_SEND(vm,"*stb?")
OUTPUT 70902;"*STB?"
```

Return Format: Integer

Attribute Summary: Preset State: variable
SCPI Compliance: confirmed

Description: This command allows you to set bits in the Status Byte register. The state is returned as a sum of the decimal weights of all set bits. The decimal weight for each bit is assigned according to the following formula:

$$2^{(bit_number)}$$

with acceptable values for bit_number being 0 through 7.

The register is not cleared by this query. To clear the Status Byte register, you must send the *CLS command.

Bits in the Status Byte register are defined as follows:

- Bit 0 is reserved.
- Bit 1 is reserved.
- Bit 2 is reserved.
- Bit 3 summarizes all enabled bits of the Questionable Status register.
- Bit 4 is the Message Available (MAV) bit. It is set whenever there is something in the module's output queue.
- Bit 5 summarizes all enabled bits of the Standard Event Status register.
- Bit 6, when read with this query (*STB?), acts as the Master Summary Status (MSS) bit. It summarizes all enabled bits of the Status Byte register. (Bit 6 acts as the Request Service (RQS) bit when it is read by a serial poll.
- Bit 7 summarizes all enabled bits of the Operation Status register.

For more information on the Status Byte register, see "HP E1431A Register Sets" in the previous chapter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*TRG command

Triggers the module when TRIG:SOUR is MANual.

Command Syntax: *TRG

Example Statements:
```
INST_SEND(vm,"*trg")
OUTPUT 70902;"*TRG"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This command triggers the module when the following two conditions are met:

- Manual triggering is designated as the trigger source. (See the TRIG:SOUR MAN command.)
- The module is waiting to trigger. (Bit 5 of the Operation Status register must be set). It is ignored at all other times.

The *TRG command has the same effect as TRIG:IMM.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*TST? query

Tests the module's hardware and returns the results.

Query Syntax: *TST?

Example Statements:
```
INST_SEND(vm,"*tst?")
OUTPUT 70902;"*TST?"
```

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: The module's self-test performs a full calibration and then compares the calibration results to specified limits. If the results are within specified limits, the module returns 0. If the results exceed the specified limits, the module returns 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

*WAI command

Holds off processing of subsequent commands until all preceding commands have been processed.

Command Syntax: *WAI

Example Statements:
```
INST_SEND(vm,"*wai")
OUTPUT 70902;"*WAI"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: There are no overlapped commands at this time. *WAI is included in the command set to be compatible with SCPI.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

ARM[:IMMediate] command

Arms the trigger if ARM:SOUR is MAN.

Command Syntax: ARM[:IMMediate]

Example Statements:
```
INST_SEND(vm,"arm")
OUTPUT 70902;"ARM:IMM"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: Two conditions must be met before this command arms the trigger.

- Manual arming must be selected (ARM:SOUR is MAN).
- This command immediately follows the INIT:IMM command.

Sending ARM:IMM *before* these conditions are met, results in an error, "-1260, Timeout on the status register." This means a time-out has occured while waiting for the measurement to reach the idle state.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

ARM:SOURce command/query

Specifies the type of arming for the analyzer's trigger.

Command Syntax: ARM:SOURce <Arm Source>

<Arm Source> ::= IMMediate|MANual

Example Statements:
```
INST_SEND(vm,"ARM:SOUR IMM")
OUTPUT 70902;"arm:source manual"
```

Query Syntax: ARM:SOURce?

Return Format: IMM|MAN

Attribute Summary: Preset State: IMM
SCPI Compliance: confirmed

Description: To select automatic arming send IMM. The module waits for the hardware to settle and then waits for a trigger signal (specified by TRIG:SOUR) before starting the measurement. When the measurement is completed, the trigger is automatically re-armed.

To select manual arming send MAN. The module waits for the hardware to settle then waits for the ARM[:IMM] command, and then waits for a trigger signal before starting the measurement. The ARM[:IMM] command must be sent to re-arm the trigger after the measurement is completed.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CALCulate:FILTer[:GATE]:TIME command/query

Specifies the settling time for the anti-alias filters.

Command Syntax: CALCulate:FILTer[:GATE]:TIME <Settling Time>

<Settling Time> ::= <Samples>|<Time>

<Samples> ::= integer
limits: 0:255

<Time> ::= a real number[<units>]
limits: dependent upon sampling rate
<units> ::= S

Example Statements:
```
INST_SEND(vm,"CALC:FILT:TIME 120")
OUTPUT 70902;"calculate:filter:gate:time 15"
```

Query Syntax: CALCulate:FILTer[:GATE]:TIME?

Return Format: Integer

Attribute Summary: Preset State: 64
SCPI Compliance: confirmed

Description: This command specifies the time given to the digital anti-alias filters to settle their output. The specified time has an impact on the time spent in the SETTLING state of the measurement loop. The module waits for the specified time to elapse, before moving to the IDLE state.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CALibration:STATe command/query

Enables the use of the calibration constants.

Command Syntax: CALibration:STATe <Calibration Enable>, <Channels>

<Calibration Enable> ::= OFF|ON

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"CAL:STAT ON")
OUTPUT 70902;"cal:state 0"
```

Query Syntax: CALibration:STATe?

Return Format: Integer

Attribute Summary: Preset State: +1 (ON)
SCPI Compliance: confirmed

Description: This command enables (or disables) the use of the calibration constants.

The calibration constants are enabled when you send a *RST.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATA:BNUMber? query

Specifies which channel of data will be read next.

Query Syntax: DATA:BNUMber?

Example Statements:
```
INST_SEND(vm,"DATA:BNUM?")
OUTPUT 70902;"data:bnumber?"
```

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: Use this query to determine which channel is providing data when reading data with the DATA[:DATA]? query.

This is an instrument-dependent query. The value returned from the query applies to the entire card set, regardless of the number of HP E1431A modules in the specified instrument.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATA[:DATA]? query

Reads data from the HP E1431A.

Query Syntax: DATA[:DATA]?

Example Statements:
```
INST_SEND(vm,"DATA?")
OUTPUT 70902;"data:data?"
```

Return Format: <DEF_BLOCK>

When data is ASCII-encoded (FORMat ASCii), <DEF_BLOCK> takes the following form:

<DEF_BLOCK> ::= [<data point>,<data point> . . .]
<data point> ::= a floating point number

When data is binary encoded (FORMat[:DATA] PACK or FORMat[:DATA] REAL) <DEF_BLOCK> takes the following form:

<DEF_BLOCK> ::= #<byte><length_bytes>[<data point>, <data point> . . .]
<byte> ::= one ASCII-encoded byte specifying the number of length bytes to follow
<length_bytes> ::= ASCII-encoded bytes specifying the number of data bytes to follow

The following definitions apply to both ASCII- and REAL data.

<data point> ::= a floating point number
limits: -9.9e+37 : 9.9e+37

The following definitions apply to PACKed data.

<data point> ::= a floating point number
limits: 16- or 32-bit signed binary

Attribute Summary:
Preset State: not applicable
SCPI Compliance: instrument-specific

Description: Use this command to read data from the module. The data is the next available block of data and is sent in the ascending order of the active channels. For example, if Channels 1, 3, and 5 are active; data from Channel 1 is sent first, then the data from Channel 3, and the data from Channel 5 is sent last. Use the DATA:BNUMber? query to determine which channel will be next.

You can load an array directly if the data format is real (FORMat[:DATA] REAL) and you are using C-SCPI. See the *HP Compiled SPCI User's Guide* for additional information.

Data can be read in either raw or scaled format. See the DATA:SCALE commands for information about specifying the scaling.

This is an instrument-dependent query. The value returned from the query applies to the entire card set, regardless of the number of HP E1431A modules in the specified instrument.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

If you want to read 16-bit unscaled ADC data, specify:

- DATA:SCALe OFF
- DATA:SIZE 16
- FORMat[:DATA] PACK,16

If you want to read 32-bit unscaled ADC data, specify:

- DATA:SCALe OFF
- DATA:SIZE 32
- FORMat[:DATA] PACK,32

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATA:SCALe:FACTor? query

Queries the scaling factor for scaled data from the module.

Query Syntax: DATA:SCALe:FACTor? <Channels>

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"DATA:SCALE:FACTOR? (@2)")
OUTPUT 70902;"data:scal:fact?"
```

Return Format: Depends on data format

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: Scaled data is IEEE 32-bit or 64-bit floating-point data from the ADC that has been scaled into volts.

The channel list can specify only one channel. See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not sent, the query defaults to Channel 1.

See the DATA:SCALE[:STATe] command for information about enabling scaled data output.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATA:SCALe[:STATe] command/query

Enables/disables the scaling of raw data into volts.

Command Syntax: DATA:SCALe[:STATe] <Scaling>

<Scaling> ::= OFF|ON

Example Statements:
```
INST_SEND(vm,"DATA:SCALE OFF")
OUTPUT 70902;"data:scal:stat on"
```

Query Syntax: DATA:SCALe[:STATe]?

Return Format: Integer

Attribute Summary: Preset State: on
SCPI Compliance: instrument-specific

Description: Data can be read from the module in either raw or scaled format.

If DATA:SCALE OFF is sent, output data is the unscaled fixed-point data from the ADC. The data is in signed twos-complement integer format and represents a fraction of the current range setting. The number returned from the DATA:SCALe:FACTor? query can be used to determine the range setting in volts.

If DATA:SCALE ON is sent, output data is in IEEE 32-bit or 64-bit floating point format and scaled in volts.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATA:SIZE command/query

Specifies the number of precision bits for the module's output of data.

Command Syntax: DATA:SIZE <FIFO Size>

<FIFO Size> ::= number
limits: 16:32

Example Statements:
```
INST_SEND(vm,"DATA:SIZE 32")
OUTPUT 70902;"data:size 16"
```

Query Syntax: DATA:SIZE?

Return Format: Integer

Attribute Summary: Preset State: 16
SCPI Compliance: instrument-specific

Description: This command sets the number of precision bits for the fixed point, two's complement data outputs from the ADC. For greatest accuracy use 32-bit mode. If the size of the stored data is important, use the 16-bit mode.

This command does *not* affect the data returned by the DATA[:DATA]? query for real and ascii formats..

Data collection is an instrument-dependent value. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATA:TRAiler command/query

Enables/disables status information appended to the data block.

Command Syntax: DATA:TRAiler <Data Trailer>

<Data Trailer> ::= OFF|ON

Example Statements:
```
INST_SEND(vm,"DATA:TRA 0")
OUTPUT 70902;"data:trailer on"
```

Query Syntax: DATA:TRAiler?

Return Format: Integer

Attribute Summary: Preset State: +0 (OFF)
SCPI Compliance: instrument-specific

Description: Status information consisting of six (6) words can be appended to a block of data collected in the HP E1431A. The status information indicates an ADC overload, an under-range condition, a trigger event, and the decimation phase counter.

The 6 words are appended in the following order:

Word 1	Reserved
Word 2	Reserved
Word 3	Decimation Phase Counter most significant word
Word 4	Decimation Phase Counter lease significant word
Word 5	Reserved
Word 6	Service Bits: bit 7 - Overload Condition bit 6 - Under-range Condition bit 5 - Trigger Event

If the VMEbus is used to transfer the data then the status byte is located in the lower byte of the 16-bit data. When the data is output via the local bus, the status byte is marked as the last byte of a transfer block.

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:ACONfiguration command/query

Allows the software to automatically set up multiple modules to operate together.

Command Syntax: DIAGnostic:ACONfiguration <State>

<State> ::= OFF|ON

Example Statements:
```
INST_SEND(vm,"diag:acon on")
OUTPUT 70902;"diagnostic:aconfiguration "
```

Query Syntax: DIAGnostic:ACONfiguration?

Return Format: Integer

Attribute Summary: Preset State: +1 (ON)
SCPI Compliance: instrument-specific

Description: If DIAGnostic:ACONfiguration is ON, the software automatically manages the inter-module TTLTRG lines.

If DIAGnostic:ACONfiguration is set to OFF, you will need to configure the TTLTRG lines yourself.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAG:ACSettling command/query

Determines if the module waits for input transients to decay.

Command Syntax: DIAGnostic:ACCSettling <State> <Channels>

<State> ::= OFF|ON

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diag:acs on (@4:6)")
OUTPUT 70902;"diagnostic:acsettling 0"
```

Query Syntax: DIAGnostic:ACSettling? (@4)

Return Format: Integer

Attribute Summary: Preset State: +1 (ON)
SCPI Compliance: instrument-specific

Description: If DIAGnostic:ACSettling is ON, the module wait for all input transients to decay before making an ac coupled measurement. For more information about AC-coupling, see the INPut:COUPLing command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGNOSTIC:BSERial command

Changes the serial number of an HP E1431A module.

Command Syntax: DIAGnostic:BSERial "<Serial_Number>"

Example Statements:
```
INST_SEND(vm,"diagnostic:bserial "0123A45213"")
OUTPUT 70902;"diag:bser "9876B00112"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: To read the current serial number of the module use the *IDN? query.

Only use this command with a single module. If the command is sent to an "instrument," the serial number will only be encoded on the first module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:INPut:OFFSet:COARse command/query

Sets the input offset adjusment DAC.

Command Syntax: DIAGnostic:INPut:OFFSet:COARse <Coarse_Offset>, <Channels>

<Coarse_Offset> :: = number
limits: 0:255

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diag:inp:offs:coar 0, (@3)")
OUTPUT 70902;"diagnostic:input:offset:coarse 200"
```

Query Syntax: DIAGnostic:INPut:OFFSet:COARse?

Return Format: Integer

Attribute Summary: Preset State: +128
SCPI Compliance: instrument-specific

Description: Use this command to manually correct the input channels DC offset. The offset value is an 8-bit value placed in the DAC.

The input offset values are automatically adjusted by the INPut:OFFSet:AUTO command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:INPut:OFFSet:FINE command/query

Sets the input offset adjusment DAC.

Command Syntax: DIAGnostic:INPut:OFFSet:FINE <Coarse_Offset>, <Channels>

<Fine_Offset> :: = number
limits: 0:255

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diag:inp:offs:fine 50, (@2:4)")
OUTPUT 70902;"diagnostic:input:offset:fine 100"
```

Query Syntax: DIAGnostic:INPut:OFFSet:FINE? (@4)

Return Format: Integer

Attribute Summary:
Preset State: +128
SCPI Compliance: instrument-specific

Description: Use this command to manually correct the input channels DC offset. The offset value is an 8-bit value placed in the DAC.

The input offset values are automatically adjusted by the INPut:OFFSet:AUTO command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:LBUS:RESet command/query

Controls the state of the Local Bus interface circuit.

Command Syntax: DIAGnostic:LBUS:RESet <State>, <Channels>

<State> ::= ON|OFF

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"DIAG:LBUS:RES 0,(@3)")
OUTPUT 70902;"diagnostic:lbus:reset on"
```

Query Syntax: Diagnostic:LBUS:RESet? (@3)

Return Format: Integer

Attribute Summary: Preset State: +0
SCPI Compliance: instrument-specific

Description: This command allows you to control the local bus reset process. Adjacent modules on the local bus must be reset carefully.

Set all modules into reset, then take each module out of reset, left-to-right. (The direction of the reset is critical.) The SCPI driver places all HP E1431A's in a local bus reset at power on or at *RST. The INITiate:IMMediate command takes each module out of reset left-to-right.

This command can be used to keep the module in a local bus reset as opposed to the VINS:LBUS:RESet command which toggles the module in and out of the reset state.

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:REGister command/query

Read or write to individual registers on the HP E1431A.

Command syntax: DIAGnostic:REGister <Register Number>, <Value>, <Channels>

<Register Number> ::= address of register

<Value> ::= 16-bit value to write to the register

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diagnostic:register 62, 0, (@1,9)")
OUTPUT 70902;"DIAGnostic:REG 4 7"
```

Query syntax: DIAGnostic:REGister? <Register Number>, <Channels>

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: This command allows you to do low-level reads and writes to the registers on the HP E1431A. See the Appendix for additional information about the VXI registers.

There is one set of registers for each HP E1431A module. Use the channel numbers to identify the module you want to address in a multiple-module configuration. For example, send DIAG:REG 4,7 (@1,9) to write the value 7 to register 4 on the first module (identified by Channel 1) and the second module (identified by Channel 9).

Use the query form to read a register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:SCLock:INPut command/query

Chooses the source of the sample clock.

Command Syntax: DIAGnostic:SCLock:INPut <Clock>, <Channels>

<Clock> ::= INTernal|EXTernal

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diag:scl:inp int, (@4:6)")
OUTPUT 70902;"diagnostic:sclock:input ext"
```

Query Syntax: DIAGnostic:SCLock:INPut? (@4)

Return Format: INT|EXT

Attribute Summary: Preset State: INT
SCPI Compliance: instrument-specific

Description: INTernal chooses the 196 kHz internal clock.

EXTernal chooses one of the TTLtrig lines which should be driven by another HP E1431A using the DIAG:SCLock:OUTPut command.

This setting is automatically handled when DIAG:ACON is ON (the default).

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:SCLock:OUTPut command/query

Sets the master sample clock.

Command Syntax: DIAGnostic:SCLock:OUTPut <State>, <Channels>

<State> ::= OFF|ON

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diag:scl:outp on, (@1,9)")
OUTPUT 70902;"diagnostic:sclock:output 0"
```

Query Syntax: DIAGnostic:SCLock:OUTPut? (@4)

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: This command specifies the HP E1431A's internal sample clock as the master. This setting is automatically handled when DIAG:ACON is ON (the default).

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DIAGnostic:SYNC command/query

Selects the shared SYNC line.

Command Syntax: DIAGnostic:SYNC <State>, <Channels>

<State> ::= INTernal|EXTernal

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"diag:sync int, (@4:6)")
OUTPUT 70902;"diagnostic:sync external"
```

Query Syntax: DIAGnostic:SYNC? (@4)

Return Format: INT|EXT

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: Use this command to select the SYNC line for multiple modules.

Send DIAGnostic:SYNC EXTernal to specify the SYNC line on the VXI backplane.

Send DIAGnostic:SYNC INTernal to specify the internal SYNC line.

This setting is automatically handled when DIAG:ACON is ON (the default).

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FORMat[:DATA] command/query

Specifies the data type to be used during output of a data block.

Command Syntax: FORMat[:DATA] <Data Format>

<Data Format> ::= ASCii|PACKed,16|PACKed,32|REAL,32|REAL,64

Example Statements: INST_SEND(vm,"FORMAT:DATA ASCii")
OUTPUT 70902;"FORM PACK"

Query Syntax: FORMat[:DATA]?

Return Format: ASC,7|PACK,16|PACK,32|REAL,32|REAL,64

Attribute Summary: Preset State: ASC,7
SCPI Compliance: confirmed

Description: The command determines the output format for data.

FORM:DATA ASCii selects extended numeric data for output from the module. Data is output as a comma-separated list of ASCii numbers with 7 digits of resolution.

FORM:DATA PACKed selects a binary format for the output from the module. Data encoding is signed binary. Data can be 16-bit (FORM PACK,16) or 32-bit (FORM PACK,32). Data is sent as a definite-length block.

FORM:DATA REAL selects the binary floating-point format (the binary floating-point format defined in the IEEE 754-1985 standard). It can be 32-bit (FORM REAL,32) or 64-bit (FORM REAL,64). Data is sent as a definite-length block. If you are using C-SCPI, you can load an array directly. See the *HP Compiled SCPI User's Guide*.

See the DATA:SCALE commands and the DATA[:DATA]? query for additional information about selecting the appropriate data format when when reading data.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INITiate[:IMMediate] command

Prepares the instrument (cardset) to acquire data.

Command Syntax: INITiate[:IMMediate]

Example Statements:
```
INST_SEND(vm,"init")
OUTPUT 70902;"INITIATE:IMM"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This command forces the measurement loop to the IDLE state. HP E1431A hardware parameter values are set by this command.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set.

For more information about the measurement loop see "The HP E1431A's Measurement Process" in Chapter 7. For more information about programming VXI cardsets, see "Addressing the HP E1431A" in Chapter 10.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut:BIAS[:STATe] command/query

Enables/disables the ICP supply on the corresponding input channel.

Command Syntax: INPut:BIAS[:STATe] <ICP Supply>, <Channels>

<ICP Supply> ::= OFF|ON

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"inp:bias off,(@2:6)")
OUTPUT 70902;"INPUT:BIAS 1,(@3)"
```

Query Syntax: INPut:BIAS[:STATe]? (@2)

Return Format: Integer

Attribute Summary: Preset State: +0 (OFF)
SCPI Compliance: confirmed

Description: This command connects (or disconnects) the internal ≈ 4 mA current source to the input connector. The nominal voltage output is 26 to 32 Vdc (open circuit).

To get rid of the DC bias that is generated when INP:BIAS is ON, turn on AC coupling. See the INPut:COUPling command.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not used, the command defaults to Channel 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut:COUPling command/query

Selects AC, DC, or ground coupling for the specified channel or group of channels.

Command Syntax: INPut:COUPling <Input Coupling>, <Channels>

<Input Coupling> ::= ACIDCIGROund

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"inp:coup ground,(@4)")
OUTPUT 70902;"INPUT:COUPLING AC"
```

Query Syntax: INPut:COUPling? (@4)

Return Format: ACIDCIGRO

Attribute Summary: Preset State: DC
SCPI Compliance: confirmed

Description: See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not used, the command defaults to Channel 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut:FILTer[:LPASs][:STATe] command/query

Enables/disables the analog anti-alias filter for the specified input channel.

Command Syntax: INPut:FILTer[:LPASs]<STATe>, <Channels>

<Channels> ::= CHANNEL_LIST

<STATe> ::= OFF/ON

Example Statements:
```
INST_SEND(vm,"inp:filt:lpass on,(@3)")
OUTPUT 70902;"INPUT:FILT 0(@6)"
```

Query Syntax: INPut:FILTer[:LPASs][:STATe]? (@3)

Return Format: Integer

Attribute Summary: Preset State: +1 (ON)
SCPI Compliance: confirmed

Description: When INPut:FILTer is OFF, the corresponding input channel bypasses the analog anti-alias low pass filter.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not used, the command defaults to Channel 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut:LOW command/query

Sets the specified channel's input shield to float or to ground.

Command Syntax: INPut:LOW <Input Ground>, <Channels>

<Input Ground> ::= GROund|FLOat

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"inp:low float,(@3)")
OUTPUT 70902;"INPUT:LOW GROUND"
```

Query Syntax: INPut:LOW? (@3)

Return Format: GRO|FLO

Attribute Summary: Preset State: FLO
SCPI Compliance: confirmed

Description: To connect the module's input shield to ground through 50 Ω nominally, send INP:LOW GRO.

To float the module's input shield through 1 MΩ, send INP:LOW FLO. The input connector ground is not completely isolated from the chassis ground.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not used, the command defaults to Channel 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut:OFFSet:AUTO command

Automatically corrects the input channels DC offset to zero.

Command Syntax: INPut:OFFSet:AUTO <Auto zero>

<Auto zero> ::= ONCE

Example Statements:
```
INST_SEND(vm,"inp:offs:auto once")
OUTPUT 70902;"INPUT:OFFSET:AUTO ONCE"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: This command sets the input offset DAC of a single channel or group of channels to a value that zeros out the offset of the input amplifier of the ADC.

The command saves the current state of the module, grounds the input of the ADC, and calculates the DC offset voltage needed to zero the ADC. It calculates this autozero value for each range setting. Whenever a range change occurs in the future, the offset value for that range is placed in the offset DAC to zero the input offset.

Warning These offset values are lost when power is turned off.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut:PHASe:AUTO command

Automatically performs an autozero and phase calibration.

Command Syntax: INPut:PHASe:AUTO ONCE,TIME|FREQuency

Example Statements:
```
INST_SEND(vm,"inp:phas:auto once,time")
OUTPUT 70902;"INPUT:PHASE:AUTO ONCE,FREQ"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: This command sets the input offset DAC of a single channel or group of channels to a value that zeros out the offset of the input amplifier of the ADC, then does a phase calibration.

Send INP:PHAS:AUTO ONCE,TIMe to optimize a measurement in the time domain.

Send INP:PHAS:AUTO ONCE,FREQuency to optimize measurements in the frequency domain. If in doubt as to how best to optimize your measurement, send INP:PHAS:AUTO ONCE,FREQ.

See INPut:OFFSet:AUTO for more information about the autozero operation.

Warning These offset values are lost when power is turned off.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

INPut[:STATe] command/query

Activates a channel or a group of channels.

Command Syntax: INPut[:STATe] <Channel State>, <Channels>

<Channel State> ::= OFF|ON

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"INPUT:STATE ON,(@1,3)")
OUTPUT 70902;"inp on,(@4,5,6)"
```

Query Syntax: INPut[:STATe]? (@1)

Return Format: Integer

Attribute Summary:
Preset State: +1 (ON)
SCPI Compliance: confirmed

Description: Use this command to control which channels acquire data before starting data collection with the INITialize:IMMediate command.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If a channel list is not specified, the command defaults to Channel 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

OUTPut:TTLTrig[T01|T02|...T67] command/query

Specifies the driving of the VXI backplane TTL trigger bus.

Command Syntax: "OUTPut:TTLTrig[T01|T23|T45|T67] OFF|0|ON|1

Example Statements: INST_SEND(vm,"output:ttltrigT23 ON")
OUTPUT 70902;"OUTP:TTLT01 OFF"

Query Syntax: OUTPut:TTLTrig[T01|T02|...T[1|2|...|67]]?

Return Format: Real

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: You must specify two TTL trigger lines on the VXI backplane. Allowable values are: 01, 23, 45, and 67.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]FILTer[:LPASs][:STATe] command/query

Enables/disables the digital anti-alias filter.

Command Syntax: [SENSe:]FILTer[:LPASs]<STATe>,<Channels>

<Channels> ::= CHANNEL_LIST

<STATe> ::= OFF/ON

Example Statements:
```
INST_SEND(vm,"sens:filt:lpass:state on,(@3)")
OUTPUT 70902;"FILTER 0"
```

Query Syntax: [SENSe:]FILTer[:LPASs][:STATe]? (@3)

Return Format: Integer

Attribute Summary:
Preset State: +1 (ON)
SCPI Compliance: confirmed

Description: This command controls the state of the digital low-pass filter. This filter should always be ON to prevent aliased data. Use caution when turning this filter OFF to collect unprotected data.

For additional information about aliasing and the sampling theorem, see *Spectrum & Network Measurements* available through your Hewlett-Packard Sales Office.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]FREQuency:SPAN command/query

Specifies the span or bandwidth of the decimation filter.

Command Syntax: [SENSe:]FREQuency:SPAN <Frequency Span>, <Channels>

<Frequency Span> ::= number
limits: 0.39:25600

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,":FREQ:SPAN 800, (@2)")
OUTPUT 70902;"sens:freq:span MIN"
```

Query Syntax: [SENSe:]FREQuency:SPAN? (@2)

Return Format: Floating Point Number

Attribute Summary: Preset State: 25.6 kHz
SCPI Compliance: confirmed

Description: The command sets the span of the output decimation filter. The decimation filter is a low-pass filter. The size of the span determines the sample clock rate.

The sample clock rate is determined by the following formula:

$$\text{span} = Fs / (2.56 \times 2) \quad \text{output sample rate} = Fs / 2^N$$

where Fs is the input sample frequency

The span is proportional to the sampling rate. To increase the sampling rate, you must increase the span. As the span is reduced, the output sample rate is reduced through decimation.

The allowable values of span are descending powers of two from the maximum span. If a requested span is between two allowable values, the higher allowable span is set. Allowable frequency span values are as follows:

25.6 kHz	1.6 kHz	100 Hz	6.25 Hz
12.8 kHz	800 Hz	50 Hz	3.125 Hz
6.4 kHz	400 Hz	25 Hz	1.56 Hz
3.2 kHz	200 Hz	12.5 Hz	0.78 Hz
			0.39 Hz

This is a module-dependent command. All channels of the module are set with the same span value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]FREQuency:SRATe command/query

Specifies the sample clock rate.

Command Syntax: [SENSe:]FREQuency:SRATe <Sample Rate>, <Channels>

<Sample Rate> ::= number
limits: 1:65536

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,":FREQ:SRATE 1.024kHz, (@1)")
OUTPUT 70902;"sens:freq:srat MIN, (@2,6)"
```

Query Syntax: [SENSe:]FREQuency:SRATe? (@6)

Return Format: Floating Point Number

Attribute Summary: Preset State: 65536
SCPI Compliance: instrument-specific

Description: The command can be used instead of FREQ:SPAN to set clock sample rate. The clock sample rate is determined by the following formula:

$$fs = (span * 2.56) / 2$$

where 1 Hz = 1 sample per second

The span is proportional to the sampling rate. To increase the sampling rate, you must increase the span. As the span is reduced, the output sample rate is reduced through decimation.

The allowable values for the sample clock are descending powers of two from the maximum rate. If a requested rate is between two allowable values, the higher allowable rate is set. Allowable sample clock rates are as follows:

65536 Hz	4096 Hz	256 Hz	16 Hz
32768 Hz	2048 Hz	128 Hz	8 Hz
16384 Hz	1024 Hz	64 Hz	4 Hz
8192 Hz	512 Hz	32 Hz	2 Hz
			1 Hz

This is a module-dependent command. All channels of the module are set with the same sample rate. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]SWEep:MODE command/query

Specifies how the HP E1431A collects data; either continuous or block mode.

Command Syntax: [SENSe:]SWEep:MODE <Sweep Mode>

<Sweep Mode> ::= BLOCk|CONTinuous

Example Statements:
```
INST_SEND(vm,"swe:mode block")
OUTPUT 70902;"SENSE:SWEEP:MODE CONTINUOUS"
```

Query Syntax: [SENSe:]SWEep:MODE?

Return Format: BLOC|CONT

Attribute Summary: Preset State: BLOC
SCPI Compliance: confirmed

Description: To select block transfer mode, send SWE:MODE BLOC. The module will stop collecting data as soon as one block of data has been collected. At this time the MEAS_DONE bit in the status register is set. To specify the size of the block, use the SWEep:POINts command.

To select continuous transfer mode, send SWE:MODE CONT. The module stays in the measure state and collects data until the FIFO overflows. Once the FIFO overflows, the MEAS_DONE and the BLOCK_READY bits in the status register are set. When these bits are set, the data can be read out. After the data is read out, its space is freed up in the FIFO. If data is read out as fast as new data is entered into the FIFO, it is possible to run the module in continuous mode.

To specify the width of the FIFO, use the DATA:SIZE command.

To stop a measurement, use the INITiate:IMMediate command.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]SWEep:POINts command/query

Specifies the number of samples in a data block.

Command Syntax: [SENSe:]SWEep:POINts <Block Size>

<Block Size> ::= number
limits: 1:16384

Example Statements: INST_SEND(vm,"SWEEP:POINTS 512")
OUTPUT 70902;"sens:swe:poin 8192"

Query Syntax: [SENSe:]SWEep:POINts?

Return Format: Floating Point Number

Attribute Summary: Preset State: 32
SCPI Compliance: instrument-specific

Description: This command sets the number of sample points in a data block that are collected before the MEAS_DONE or BLOCK_READY bits in the status register are set. See [SENSe:]SWEep:MODE to see how these status bits are set.

Specifying a number less than the minimum or larger than the maximum results in the error, *Illegal blocksize*.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instruemnt, are set with the same blocksize.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]VOLTage[:DC]:RANGe:AUTO command

Automatically selects the best range on the specified channel for the current input signal.

Command Syntax: [SENSe:]VOLTage[:DC]:RANGe:AUTO <Autorange>, <Time>, <Channels>

<Autorange> ::= ONCE

<Time> ::= number
limits: 0:9.9e37

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"sens:volt:dc:range:auto once,1.0")
OUTPUT 70902;"VOLTAGE:RANGE:AUTO ONCE,1.5,(@5)"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This command automatically adjusts the input range once.

The module searches for the lowest, non-overloading input range during the specified time period (t). If the module detects an overload condition, the module steps the input range up through successive range values until the input is no longer in an overload condition.

If conditions change and the input range is no longer approriate, you will need to send the VOLT:RANG:AUTO ONCE command again.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not used, the command defaults to Channel 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[SENSe:]VOLTage[:DC]:RANGe[:UPPer] command

Specifies the input range for a single channel or group of channels.

Command Syntax: [SENSe:]VOLTage[:DC]:RANGe[:UPPer] <Input Range>, <Channels>

<Input Range> ::= number
limits: 0.005:10.0

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,":volt:rang .5V, (@6)")
OUTPUT 70902;"SENS:VOLT:DC:RANG:UPP 2V"
```

Attribute Summary: Preset State: 10.0 V
SCPI Compliance: confirmed

Description: Valid input ranges are 0.005V, 0.01V, 0.02V, 0.05V, 0.1V, 0.2V, 0.5V, 1V, 2V, 5V, and 10V. If you send a value that is not allowed, it is rounded up to the next higher value. If you send a value that is greater than the maximum range, the range is set to 10.0V, the maximum.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. If the channel list is not used, the command defaults to Channel 1.

See the [SENSe:]VOLTage:[DC:]RANGe:AUTO command for information about the autorange feature.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:OPERation:CONDition? query

Reads the Operation Status condition register.

Query Syntax: STATus:OPERation:CONDition?

Example Statements:
```
INST_SEND(vm,"STAT:OPERATION:COND?")
OUTPUT 70902;"STAT:OPER:CONDITION?"
```

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This query returns the sum of the decimal weights of all bits currently set to 1 in the Operation Status condition register. (The decimal weight of a bit is 2^n, where n is the bit number.)

See "Operation Status Register Set" in the previous chapter for a definition of bits in the register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:OPERation:ENABle command/query

Sets and queries bits in the Operation Status enable register.

Command Syntax: STATus:OPERation:ENABle <Operation Enable Register>

<Operation Enable Register> ::= number
limits: 0:32767

Example Statements:
```
INST_SEND(vm,"Status:Oper:Enab 96")
OUTPUT 70902;"status:operation:enable 2"
```

Query Syntax: STATus:OPERation:ENABle?

Return Format: Integer

Attribute Summary: Preset State: not affected by Preset
SCPI Compliance: confirmed

Description: To set a single bit in the Operation Status enable register to 1, send the bit's decimal weight with this command. To set more than one bit to 1, send the sum of the decimal weights of all the bits. (The decimal weight of a bit is 2^n, where n is the bit number.)

All bits are initialized to 0 when the module is turned on or when the STAT:PRES command is sent. However, the current setting of bits is *not* modified when you send the *RST command.

See "Operation Status Register Set" in the previous chapter for a definition of bits in the register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:OPERation[:EVENt]? query

Reads and clears the Operation Status event register.

Query Syntax: STATus:OPERation[:EVENt]?

Example Statements:
```
INST_SEND(vm,"STAT:OPERATION:EVENT?")
OUTPUT 70902;"STAT:OPER?"
```

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This query returns the sum of the decimal weights of all bits currently set to 1 in the Operation Status event register. (The decimal weight of a bit is 2^n, where n is the bit number.)

Note The Operation Status event register is automatically cleared after it is read by this query.

See "Operation Status Register Set" in the previous chapter for a definition of bits in the register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:PRESet command

Sets bits in most enable registers to their default state.

Command Syntax: STATus:PRESet

Example Statements:
```
INST_SEND(vm,"STAT:PRES")
OUTPUT 70902;"status:preset"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: STATUS:PRESet has the effect of bringing all events to the Questionable Status and Operation Status register sets *without* creating an SRQ or reflecting events in a serial poll.

It sets all enable register bits in the Questionable Status and Operation Status registers to 0.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:QUEStionable:CONDition? query

Reads and clears the Questionable Status condition register.

Query Syntax: STATus:QUEStionable:CONDition?

Example Statements:
```
INST_SEND(vm,"STAT:QUESTIONABLE:COND?")
OUTPUT 70902;"stat:ques:condition?"
```

Return Format: Integer

Attribute Summary:
Preset State: not applicable
SCPI Compliance: confirmed

Description: This query returns the sum of the decimal weights of all bits currently set to 1 in the Questionable Status condition register. (The decimal weight of a bit is 2^n, where n is the bit number.)

See "Questionable Status Register Set" in the previous chapter for a definition of bits in the register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:QUEStionable:ENABle command/query

Sets and queries bits in the Questionable Status enable register.

Command Syntax: STATus:QUEStionable:ENABle <Questionable Enable Register>

<Questionable Enable Register> ::= number
limits: 0:32767

Example Statements:
```
INST_SEND(vm,"stat:ques:enab 1")
OUTPUT 70902;"stat:questionable:enable 513"
```

Query Syntax: STATus:QUEStionable:ENABle?

Return Format: Integer

Attribute Summary: Preset State: not affected by Preset
SCPI Compliance: confirmed

Description: To set a single bit in the Questionable Status enable register to 1, send the bit's decimal weight with this command. To set more than one bit to 1, send the sum of the decimal weights of all the bits. (The decimal weight of a bit is 2^n, where n is the bit number.)

All bits are initialized to 0 when the module is turned on or when the STAT:PRES command is sent. However, the current setting of bits is *not* modified when you send the *RST command.

See "Questionable Status Register Set" in the previous chapter for a definition of bits in the register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

STATus:QUEStionable[:EVENt]? query

Reads and clears the Questionable Status event register.

Query Syntax: STATus:QUEStionable[:EVENt]?

Example Statements:
```
INST_SEND(vm,"STAT:QUESTIONABLE:EVENT?")
OUTPUT 70902;"stat:ques?"
```

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This query returns the sum of the decimal weights of all bits currently set to 1 in the Questionable Status event register. (The decimal weight of a bit is 2^n, where n is the bit number.)

Note The Questionable Status event register is automatically cleared after it is read by this query.

See "Questionable Status Register Set" in the previous chapter for a definition of bits in the register set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SYSTem:ERRor? query

Returns one error message from the module's error queue.

Query Syntax: SYSTem:ERRor?

Example Statements:
```
INST_SEND(vm,"syst:err?")
OUTPUT 70902;"SYSTEM:ERROR?"
```

Return Format: Integer
STRING

Attribute Summary: Preset State: not affected by Preset
SCPI Compliance: confirmed

Description: The error queue temporarily stores error messages. When you send the SYST:ERR query, one message is moved from the error queue to the output queue so your controller can read the message. The error queue delivers messages to the output queue in the order received.

Note The error queue is cleared when you turn on the module and when you send the *CLS command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SYSTem:VERSion? query

Returns the SCPI version to which the module complies.

Query Syntax: SYSTem:VERSion?

Example Statements:
```
INST_SEND(vm,"syst:vers?")
OUTPUT 70902;"SYSTEM:VERSION?"
```

Return Format: Fixed Point Number

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: The format of the return is YYYY.V The Ys represent the SCPI year-version and the V represents the revision number for that year.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:DELay command/query

Specifies a pre-trigger or a post-trigger delay for the specified channel.

Command Syntax: TRIGger:DELay <Channel Trigger Delay>, <Channels>

<Channel Trigger Delay> ::= <Samples>|<Time>

<Samples> ::= integer <units>
limits: -16383:1048575 (See below for pre-trigger limits.)
<units> ::= PNT

<Time> ::= a real number[<units>]
limits: dependent upon sampling rate
<units> ::= S

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"TRIG:DEL -0.2 s, (@3)")
OUTPUT 70902;"trigger:delay 128 PNT"
```

Query Syntax: TRIGger:DELay? (@3)

Return Format: Floating Point Number

Attribute Summary: Preset State: +0
SCPI Compliance: instrument-specific

Description: This command specifies the delay in one of two ways; as a specified amount of time or as the number of samples between two points: the point at which the module is triggered and the point at which the specified channel starts collecting data. The query always returns the delay in terms of time, seconds.

Pre-trigger values are entered as a negative (–) quantity; the amount of time or the number of samples *before* the trigger event to be saved as data. If a requested pre-trigger delay is larger than the FIFO, an error results. The maximum pre-trigger value is one less than the number of samples in the data block. The absolute maximum pre-trigger value is -16383 samples.

Post-trigger values are entered as a positive (+) quantity; the amount of time or the number of samples to ignore *after* the trigger event before data collection occurs. The maximum post-trigger delay is 1048575 samples.

Use the FREQ:SPAN or FREQ:SRAT commands to set the sample rate for pre-trigger and post-trigger values specified in terms of output samples. To set the number of points in the blocksize, use the [SENSe:]SWEep:POINts command.

This is a module-dependent command. If you are programming multiple modules, specify *all* channels in the channel list. Otherwise, the trigger delay set for the last specified channel will be applied to *all* channels in that module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:HYSTeresis command/query

Specifies the trigger-level region which causes the module to trigger.

Command Syntax: TRIGger:HYSTeresis <Range>, <Channels>

<Range> ::= number <units>
limits: -128.67:128.67
<units> ::= V

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"TRIGGER:HYST 10")
OUTPUT 70902;"trig:hyst 65 V,(@3)"
```

Query Syntax: TRIGger:HYSTeresis? (@3)

Return Format: Floating Point Number

Attribute Summary: Preset State: +0 V
SCPI Compliance: instrument-specific

Description: This command sets the size of the trigger level region. The region is specified in Volts around the value specified with TRIG:LEVEL. You specify the trigger level as a percentage of the trigger channel's current input region.

The upper and lower levels are defined as follows:

Upper Level = TRIG:LEV + TRIG:HYST/2

Lower Level = TRIG:LEV - TRIG:HYST/2

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. The command defaults to Channel 1 if the channel list is not used.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:IMMediate command

Triggers the HP E1431A.

Command Syntax: TRIGger:IMMediate

Example Statements:
```
INST_SEND(vm,"TRIGGER:IMMEDIATE")
OUTPUT 70902;"trig:imm"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: confirmed

Description: This command manually tiggers the HP E1431A, moving from the ARM state to the TRIGGER state. TRIG:STATe must be OFF.

This command has the same effect as the *TRG command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:LEVel command/query

Specifies the level of the input signal which causes the module to trigger.

Command Syntax: TRIGger:LEVel <LEVEL>, <Channels>

<LEVEL> ::= number <units>
limits: -128.67:128.67

<units>::= V

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"TRIGGER:LEV 0.5 V,(@3:5)")
OUTPUT 70902;"trig:lev -1 V"
```

Query Syntax: TRIGger:LEVel? (@5)

Return Format: Floating Point Number

Attribute Summary: Preset State: +0 V
SCPI Compliance: confirmed

Description: You specify the trigger level as a percentage of the trigger channel's current input range. TRIG:STATe must be ON.

Set the size of the trigger region with the TRIG:LEVEL:HYST command. The upper and lower levels are defined as follows:

Upper Level = TRIG:LEV + TRIG:HYST/2

Lower Level = TRIG:LEV - TRIG:HYST/2

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. The command defaults to Channel 1 if the channel list is not used.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:MODE command/query

Deterimines how TRIGger:SLOPe and TRIGger:LEVel parameters will be used.

Command Syntax: TRIGger:MODE <Trigger Mode>, <Channels>

<Trigger Mode> ::= LEVel|BOUNd

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,TRIGGER:MODE LEVEL,(@4,5)")
OUTPUT 70902;"trig:mode boun,(@2)"
```

Query Syntax: TRIGger:MODE? (@4)

Return Format: LEV|BOUN

Attribute Summary: Preset State: LEVel
SCPI Compliance: instrument-specific

Description: This command is used in conjunction with the TRIG:LEVEL and TRIG:SLOPE commands.

To trigger a measurement when the input signal reaches a specific level on either a positive or negative transition; use TRIG:MODE LEVel.

To trigger a measurement when the input signal reaches a certain voltage region; use TRIG:MODE BOUND. See the TRIG:SLOPe ENTRy|EXIT command for additional information about triggering within a specific region of values.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:SLOPe command/query

Specifies the slope of the signal which triggers the module.

Command Syntax: TRIGger:SLOPe <Trigger Slope>, <Channels>

<Trigger Slope> ::= POSitive|NEGative|ENTRy|EXIT

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"TRIGGER:SLOPE ENTRY,(@4)")
OUTPUT 70902;"trig:slop exit"
```

Query Syntax: TRIGger:SLOPe? (@4)

Return Format: POS|NEG|ENTR|EXIT

Attribute Summary: Preset State: POS
SCPI Compliance: confirmed

Description: Use this command when TRIG:STATE is ON.

Use the POSitive and NEGative parameters with the TRIG:MODE LEVel command. The channel triggers when the signal crosses one of the limit levels. The limits are defined as follows:

Upper Level = TRIG:LEV + TRIG:HYST/2

Lower Level = TRIG:LEV - TRIG:HYST/2

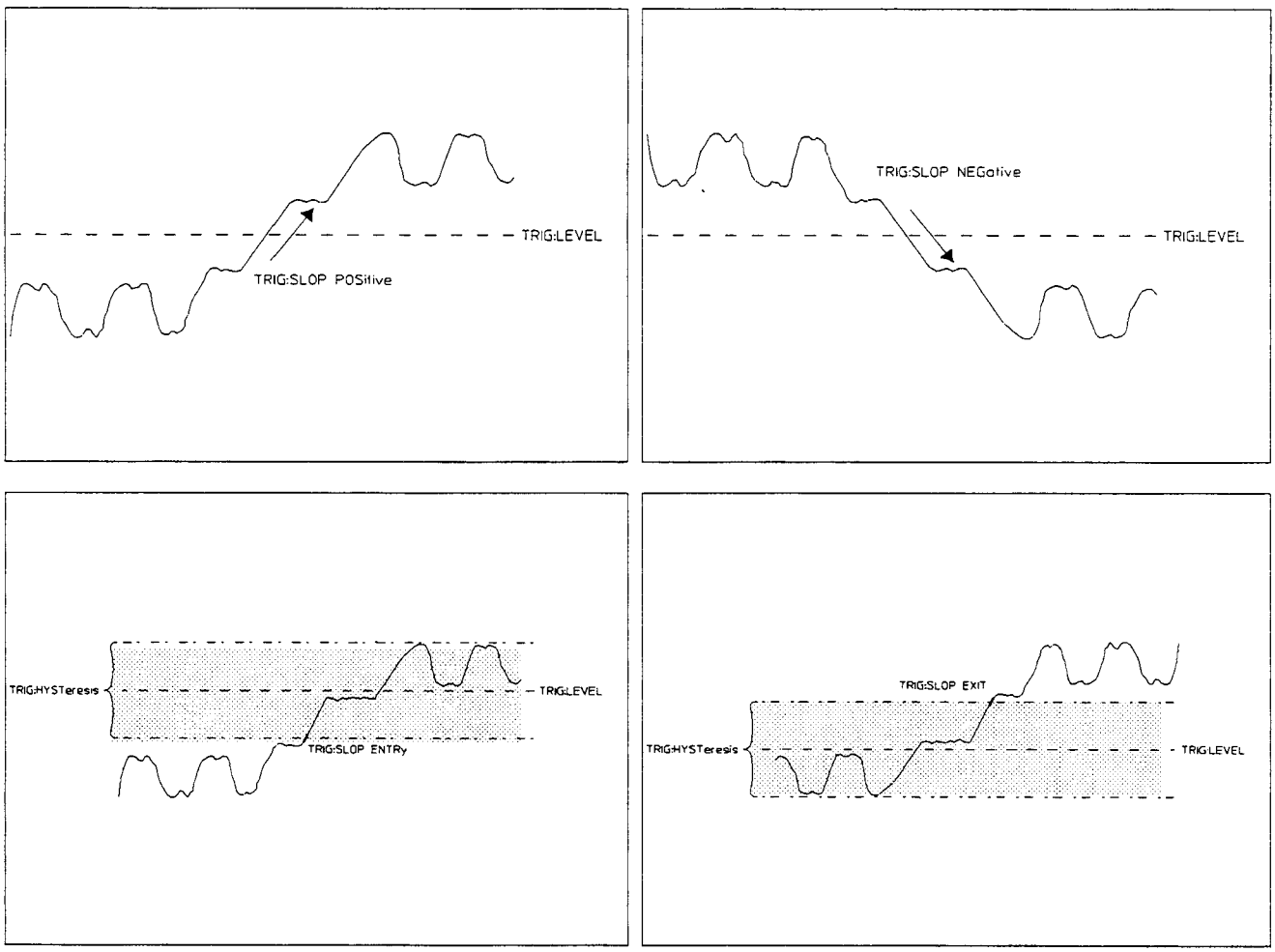
Use the ENTRy and EXIT parameters with the TRIG:MODE BOUND command. The channel triggers when the signal enters or leaves the region defined by the limit levels. The limits are defined above. The type of transition (positive or negative) has no effect if TRIG:SLOP ENTR or TRIG:SLOP EXIT is used.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The query returns the currently specified slope.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. The command defaults to Channel 1 if the channel list is not used.

See the TRIG:SOURce command for information about automatic triggering.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:SOURce command/query

Selects manual or free run triggering.

Command Syntax: TRIGger:SOURce <Trigger Source>

<Trigger Source> ::= MANuallIMMediate

Example Statements: INST_SEND(vm,"TRIGGER:SOURCE MANUAL")
OUTPUT 70902;"trig:sour imm"

Query Syntax: TRIGger:SOURce?

Return Format: MAN|IMM

Attribute Summary: Preset State: IMM
SCPI Compliance: confirmed

Description: To select free run triggering send IMM. The module automatically triggers as soon as it is armed. TRIG:STATe must be OFF.

Note

The module must be waiting to trigger when it receives a trigger signal or trigger command, otherwise the signal or command is ignored.

Bit 5 of the Operation Status condition register is set to 1 when the module is waiting to trigger.

To select manual triggering send MAN. The module triggers as soon as the command is sent. TRIG:STATe must be OFF.

See "Syntax Descriptions" at the beginning of this chapter for more information about channel lists. The command defaults to Channel 1 if the channel list is not used.

This is an instrument-dependent command. All channels, regardless of the number of HP E1431A modules in the designated instrument, are set with the same value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TRIGger:STATe command/query

Enables/disables a channel's ability to trigger a measurement.

Command Syntax: TRIGger:STATe <Trigger State>, <Channels>

<Trigger State> ::= OFF|ON

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,TRIGGER:STATE ON,(@4,5)")
OUTPUT 70902;"trig:stat on, (@2)"
```

Query Syntax: TRIGger:STATe? (@5)

Return Format: Integer

Attribute Summary: Preset State: +0 (OFF)
SCPI Compliance: confirmed

Description: This command enables an individual channel to trigger the measurement. Use this command in conjunction with the TRIGGER:LEVEL and TRIGGER:HYSTERESIS commands.

If TRIG:STATe is OFF, use *TRG or TRIG:IMM to trigger a measurement. In addition, you could set TRIG:SOURce to IMM for free run triggering.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VINStrument[:CONFigure]:LBUS[:MODE] command/query

Selects the mode in which the local bus operates.

Command Syntax: VINStrument[:CONFigure]:LBUS[:MODE] <Local Bus Mode>,<Channels>

<Local Bus Mode> ::= GENerate|APPend|INSert|PIPeline|RESet

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"VINS:LBUS GEN")
OUTPUT 70902;"vinstrument:configure:lbus reset"
```

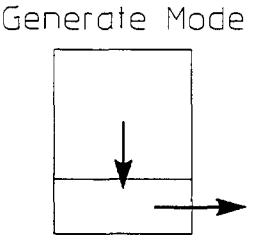
Query Syntax: VINStrument[:CONFigure]:LBUS[:MODE]?

Return Format: GEN|APP|INS|PIP|RES

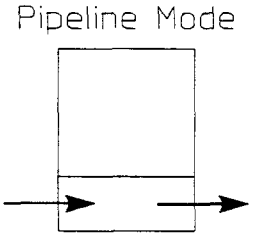
Attribute Summary: Preset State: PIPeline
SCPI Compliance: instrument-specific

Description: This command is used to set up single- or multiple module systems to transfer data via the local bus.

To output data generated in the module to the *right* on the local bus, send VINS:LBUS GENerate.

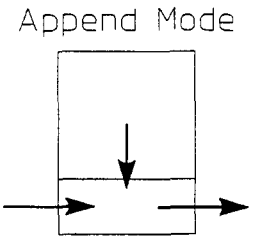


To pass data from the local bus on the *left* of the module to the local bus on the *right* of the module, send VINS:LBUS PIPeline. The module does not output any of its own data via the local bus.

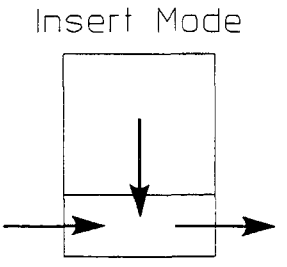


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To "pipeline" data from the right to the left of the module, and then *append* the module's own data to the left, send VINS:LBUS APPend.



To output data generated in the module and then switch to pipeline mode, send VINS:LBUS INSert.



To reset the currently selected mode, send VINS:LBUS RESet.

The local bus must be selected as the data port with the VINS:PORT command (VINS:PORT LBUS). Otherwise, the local bus will remain in the PIPeline mode until selected as the data port.

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VINStrument[:CONFigure]:PORT command/query

Selects which port on the module delivers the data.

Command Syntax: VINStrument[:CONFigure]:PORT <Port>, <Channels>

<Port> ::= LBUS|VME

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"VINS:PORT LBUS")
OUTPUT 70902;"vinstrument:configure:port vme"
```

Query Syntax: VINStrument[:CONFigure]:PORT?

Return Format: LBUS|VME

Attribute Summary:
Preset State: VME
SCPI Compliance: instrument-specific

Description: This command sets the data output port for the module to either the VME data output register or the local bus.

This is a module-dependent command. All channels of the module are set with the same value. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise, you may overlook one of the module settings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VXI:CONFigure:CNUMber? query

Returns the maximum number of available channels.

Query Syntax: VXI:CONFigure:CNUMber?

Example Statements:
```
INST_SEND(vm,"VXI:CONF:CNUM?")
OUTPUT 70902;"vxi:configure:cnumber?"
```

Return Format: Integer

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: Use this query to determine the number of channels available in your system. This is particularly useful if you are using multiple HP E1431A modules. For more information about addressing card sets, see "Addressing the HP E1431A" in Chapter 10.

This is an instrument-dependent query. The value returned from the query applies to the entire card set, regardless of the number of HP E1431A modules in the specified instrument.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VINStrument:LBUS:RESet command

Resets the local bus.

Command Syntax: VINStrument:LBUS:RESet <Channels>

<Channels> ::= CHANNEL_LIST

Example Statements:
```
INST_SEND(vm,"VINS:LBUS:RES ON,(@1:16)")
OUTPUT 70902;"vinstrument:lbus:reset 0,(@1)"
```

Attribute Summary: Preset State: not applicable
SCPI Compliance: instrument-specific

Description: This is a module-dependent command. All channels of the module are reset with the same command. If you are programming multiple modules, specify at least one channel from each module in the channel list. Otherwise you may overlook one of the modules.

When programming multiple modules, it is important to turn off the bus reset in sequential order. Send VINS:LBUS:RES to the left-most module first. Continue sending the command in the direction of the right-most module. This ensures the local bus is cleared of all data and the modules have been reset in the appropriate sequence.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Errors

SCPI Command Errors

Error Number	Description
-100	Command error. This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error as defined in *IEEE 488.2*, 11.5.1.1.4 has occurred.
-101	Invalid character. A syntactic element contains a character which is invalid for that type; for example, a header containing an ampersand, SETUP&. This error might be used in place of errors -114, -121, -141, and perhaps others.
-102	Syntax error. An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.
-103	Invalid separator. The parser was expecting a separator and encountered an illegal character; for example, the semicolon was omitted after a program message unit, *EMC 1 :CH1:VOLTS 5.
-104	Data type error. The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.
-105	GET not allowed. A Group Execute Trigger was received within a program message (see *IEEE 488.2*, 7.7).
-108	Parameter not allowed. More parameters were received than expected for the header; for example, the *EMC common command only accepts one parameter, so receiving *EMC 0,,1 is not allowed.
-109	Missing parameter. Fewer parameters were received than required for the header; for example, the *EMC common command requires one parameter, so receiving *EMC is not allowed.
-110	Command header error. An error was detected in the header. This error message is used when the device cannot detect the more specific errors described for errors -111 through -119.
-111	Header separator error. A character which is not a legal header separator was encountered while parsing the header; for example, no white space followed the header, thus *GMC"MACRO" is an error.
-112	Program mnemonic too long. The header contains more that twelve characters (see *IEEE 488.2*, 7.6.1.4.1).]
-113	Undefined header. The header is syntactically correct, but it is undefined for this specific device; for example, *XYZ is not defined for any device.
-114	Header suffix out of range. The value of a numeric suffix attached to a program mnemonic, see Syntax and Style section 6.2.5.2, makes the header invalid.
-120	Numeric data error. This error, as well as errors -121 through -129, are generated when parsing a data element which appears to be numeric, including the nondecimal numeric types. This particular error message is used if the device cannot detect a more specific error.
-121	Invalid character in number. An invalid character for the data type being parsed was encountered; for example, an alpha in a decimal numeric or a "9" in octal data.
-123	Exponent too large. The magnitude of the exponent was larger than 32000 (see *IEEE 488.2*, 7.7.2.4.1).
-124	Too many digits. The mantissa of a decimal numeric data element contained more than 255 digits excluding leading zeros (see *IEEE 488.2*, 7.7.2.4.1).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Error Number	Description
-128	Numeric data not allowed. A legal numeric data element was received, but the device does not accept one in this position for the header.
-130	Suffix error. This error, as well as errors -131 through -139, are generated when parsing a suffix. This particular error message is used if the device cannot detect a more specific error.
-131	Invalid suffix. The suffix does not follow the syntax described in *IEEE 488.2*, 7.7.3.2, or the suffix is inappropriate for this device.
-134	Suffix too long. The suffix contained more than 12 characters (see *IEEE 488.2*, 7.7.3.4).
-138	Suffix not allowed. A suffix was encountered after a numeric element which does not allow suffixes.
-140	Character data error. This error, as well as errors -141 through -149, are generated when parsing a character data element. This particular error message is used if the device cannot detect a more specific error.
-141	Invalid character data. Either the character data element contains an invalid character or the particular element received is not valid for the header.
-144	Character data too long. The character data element contains more than twelve characters (see *IEEE 488.2*, 7.7.1.4).
-148	Character data not allowed. A legal character data element was encountered where prohibited by the device.
-150	String data error. This error, as well as errors -151 through -159, are generated when parsing a string data element. This particular error message is used if the device cannot detect a more specific error.
-151	Invalid string data. A string data element was expected, but was invalid for some reason (see *IEEE 488.2*, 7.7.5.2); for example, an END message was received before the terminal quote character.
-158	String data not allowed. A string data element was encountered but was not allowed by the device at this point in parsing.
-160	Block data error. This error, as well as errors -161 through -169, are generated when parsing a block data element. This particular error message is used if the device cannot detect a more specific error.
-161	Invalid block data. A block data element was expected, but was invalid for some reason (see *IEEE 488.2*, 7.7.6.2); for example, an END message was received before the length was satisfied.
-168	Block data not allowed. A legal block data element was encountered but was not allowed by the device at this point in parsing.
-170	Expression error. This error, as well as errors -171 through -179, are generated when parsing an expression data element. This particular error message is used if the device cannot detect a more specific error.
-171	Invalid expression. The expression data element was invalid (see *IEEE 488.2*, 7.7.7.2); for example, unmatched parentheses or an illegal character.
-178	Expression data not allowed. A legal expression data was encountered but was not allowed by the device at this point in parsing.
-180	Macro error. This error, as well as errors -181 through -189, are generated when defining a macro or executing a macro. This particular error message is used if the device cannot detect a more specific error.
-181	Invalid outside macro definition. Indicates that a macro parameter placeholder ($<$number) was encountered outside of a macro definition.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Error Number	Description
-183	Invalid inside macro definition. Indicates that the program message unit sequence, sent with a *DDT or *DMC command, is syntactically invalid (see *IEEE 488.2*, 10.7.6.3).
-184	Macro parameter error. Indicates that a command inside the macro definition had the wrong number or type of parameters.

SCPI Execution Errors

Error Number	Description
-200	Execution error. This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that an Execution Error as defined in *IEEE 488.2*, 11.5.1.1.5 has occurred.
-201	Invalid while in local. Indicates that a command is not executable while the device is in local due to a hard local control (see *IEEE 488.2*, 5.6.1.5); for example, a device with a rotary switch receives a message which would change the switches state, but the device is in local so the message can not be executed.
-202	Settings lost due to rtl. Indicates that a setting associated with a hard local control (see *IEEE 488.2*, 5.6.1.5) was lost when the device changed to LOCS from REMS or to LWLS from RWLS.
-210	Trigger error.
-211	Trigger ignored. Indicates that a GET, *TRG, or triggering signal was received and recognized by the device but was ignored because of device timing considerations; for example, the device was not ready to respond. Note: a DTO device always ignores GET and treats *TRG as a Command Error.
-212	Arm ignored. Indicates that an arming signal was received and recognized by the device but was ignored.
-213	Init ignored. Indicates that a request for a measurement initiation was ignored as another measurement was already in progress.
-214	Trigger deadlock. Indicates that the trigger source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.
-215	Arm deadlock. Indicates that the arm source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.
-220	Parameter error. Indicates that a program data element related error occurred. This error message is used when the device cannot detect the more specific errors described for errors -221 through -229.
-221	Settings conflict. Indicates that a legal program data element was parsed but could not be executed due to the current device state (see *IEEE 488.2*, 6.4.5.3 and 11.5.1.1.5.).
-222	Data out of range. Indicates that a legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the device (see *IEEE 488.2*, 11.5.1.1.5.).
-223	Too much data. Indicates that a legal program data element of block, expression, or string type was received that contained more data than the device could handle due to memory or related device-specific requirements.
-224	Illegal parameter value. Used where exact value, from a list of possibles, was expected.
-225	Out of memory. The device has insufficient memory to perform the requested operation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Error Number	Description
-226	Lists not same length. Attempted to use LIST structure having individual LIST's of unequal lengths.
-230	Data corrupt or stale. Possibly invalid data; new reading started but not completed since last access.
-231	Data questionable. Indicates that measurement accuracy is suspect.
-240	Hardware error. Indicates that a legal program command or query could not be executed because of a hardware problem in the device. Definition of what constitutes a hardware problem is completely device-specific. This error message is used when the device cannot detect the more specific errors described for errors -241 through -249.
-241	Hardware missing. Indicates that a legal program command or query could not be executed because of missing device hardware; for example, an option was not installed. Definition of what constitutes missing hardware is completely device-specific.
-250	Mass storage error. Indicates that a mass storage error occurred. This error message is used when the device cannot detect the more specific errors described for errors -251 through -259.
-251	Missing mass storage. Indicates that a legal program command or query could not be executed because of missing mass storage; for example, an option that was not installed. Definition of what constitutes missing mass storage is device-specific.
-252	Missing media. Indicates that a legal program command or query could not be executed because of a missing media; for example, no disk. The definition of what constitutes missing media is device-specific.
-253	Corrupt media. Indicates that a legal program command or query could not be executed because of corrupt media; for example, bad disk or wrong format. The definition of what constitutes corrupt media is device-specific.
-254	Media full. Indicates that a legal program command or query could not be executed because the media was full; for example, there is no room on the disk. The definition of what constitutes a full media is device-specific.
-255	Directory full. Indicates that a legal program command or query could not be executed because the media directory was full. The definition of what constitutes a full media directory is device-specific.
-256	File name not found. Indicates that a legal program command or query could not be executed because the file name on the device media was not found; for example, an attempt was made to read or copy a nonexistent file. The definition of what constitutes a file not being found is device-specific.
-257	File name error. Indicates that a legal program command or query could not be executed because the file name on the device media was in error; for example, an attempt was made to copy to a duplicate file name. The definition of what constitutes a file name error is device-specific.
-258	Media protected. Indicates that a legal program command or query could not be executed because the media was protected; for example, the write-protect tab on a disk was present. The definition of what constitutes protected media is device-specific.
-260	Expression error. Indicates that a expression program data element related error occurred. This error message is used when the device cannot detect the more specific errors described for errors -261 through -269.
-261	Math error in expression. Indicates that a syntactically legal expression program data element could not be executed due to a math error; for example, a divide-by-zero was attempted. The definition of math error is device-specific.
-270	Macro error. Indicates that a macro-related execution error occurred. This error message is used when the device cannot detect the more specific errors described for errors -271 through -279.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Error Number	Description
-271	Macro syntax error. Indicates that a syntactically legal macro program data sequence, according to *IEEE 488.2*, 10.7.2, could not be executed due to a syntax error within the macro definition (see *IEEE 488.2*, 10.7.6.3.).
-272	Macro execution error. Indicates that a syntactically legal macro program data sequence could not be executed due to some error in the macro definition (see *IEEE 488.2*, 10.7.6.3.).
-273	Illegal macro label. Indicates that the macro label defined in the *DMC command was a legal string syntax, but could not be accepted by the device (see *IEEE 488.2*, 10.7.3 and 10.7.6.2); for example, the label was too long, the same as a common command header, or contained invalid header syntax.
-274	Macro parameter error. Indicates that the macro definition improperly used a macro parameter placeholder (see *IEEE 488.2*, 10.7.3).
-275	Macro definition too long. Indicates that a syntactically legal macro program data sequence could not be executed because the string or block contents were too long for the device to handle (see *IEEE 488.2*, 10.7.6.1).
-276	Macro recursion error. Indicates that a syntactically legal macro program data sequence could not be executed because the device found it to be recursive (see *IEEE 488.2*, 10.7.6.6).
-277	Macro redefinition not allowed. Indicates that a syntactically legal macro label in the *DMC command could not be executed because the macro label was already defined (see *IEEE 488.2*, 10.7.6.4).
-278	Macro header not found. Indicates that a syntactically legal macro label in the *GMC? query could not be executed because the header was not previously defined.
-280	Program error. Indicates that a downloaded program-related execution error occurred. This error message is used when the device cannot detect the more specific errors described for errors -281 through -289. A downloaded program is used to add algorithmic capability to a device. The syntax used in the program and the mechanism for downloading a program is device-specific.
-281	Cannot create program. Indicates that an attempt to create a program was unsuccessful. A reason for the failure might include not enough memory.
-282	Illegal program name. The name used to reference a program was invalid; for example, redefining an existing program, deleting a nonexistent program, or in general, referencing a nonexistent program.
-283	Illegal variable name. An attempt was made to reference a nonexistent variable in a program.
-284	Program currently running. Certain operations dealing with programs may be illegal while the program is running; for example, deleting a running program might not be possible.
-285	Program syntax error. Indicates that a syntax error appears in a downloaded program. The syntax used when parsing the downloaded program is device-specific.
-286	Program runtime error.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SCPI Device-Specific Errors

Error Number	Description
-300	Device-specific error. This is the generic device-dependent error for devices that cannot detect more specific errors. This code indicates only that a Device-Dependent Error as defined in *IEEE 488.2* has occurred.
-310	System error. Indicates that some error has occured.
-311	Memory error. Indicates that an error was detected in the device's memory. The scope of this error is device-dependent.
-312	PUD memory lost. Indicates that the protected user data saved by the *PUD command has been lost.
-313	Calibration memory lost. Indicates that nonvolatile calibration data used by the *CAL? command has been lost.
-314	Save/recall memory lost. Indicates that the nonvolatile data saved by the *SAV? command has been lost.
-315	Configuration memory lost. Indicates that nonvolatile configuration data saved by the device has been lost. The meaning of this error is device-specific.
-330	Self-test failed.
-350	Queue overflow. A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded.

SCPI Query Error

Error Number	Description
-400	Query error. This is the generic query error for devices that cannot detect more specific errors. This code indicates only that a Query Error as defined in *IEEE 488.2*, 11.5.1.1.7 and 6.3 has occurred.
-410	Query INTERRUPTED. Indicates that a condition causing an INTERRUPTED Query error occurred (see *IEEE 488.2*, 6.3.2.3); for example, a query followed by DAB or GET before a response was completely sent.
-420	Query UNTERMINATED. Indicates that a condition causing an UNTERMINATED Query error occurred (see *IEEE 488.2*, 6.3.2.2); for example, the device was addressed to talk and an incomplete program message was received.
-430	Query DEADLOCKED. Indicates that a condition causing an DEADLOCKED Query error occurred (see *IEEE 488.2*, 6.3.1.7); for example, both input buffer and output buffer are full and the device cannot continue.
-440	Query UNTERMINATED after indefinite response. Indicates that a query was received in the same program message after an query requesting an indefinite response was executed (see *IEEE 488.2*, 6.5.7.5).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

HP E1431A-Specific Errors

Error Number	Description
1201	Illegal input mode
1202	Illegal anti-alias filter state
1203	Illegal analog input source
1204	Illegal input grounding
1205	Illegal coupling
1206	Illegal ranging down
1207	Illegal ranging up
1208	Range out of limits
1209	Offset out of limits
1210	Illegal active channel
1211	Illegal append status state
1212	Illegal blocksize
1213	Illegal data size
1214	Illegal data mode
1215	Illegal data port
1216	Illegal channel order
1217	Illegal LBUS mode
1218	Illegal LBUS state
1220	Illegal filter output mode
1221	Illegal digital anti-alias filter state
1222	Illegal filter decimation bandwidth
1223	Illegal filter settling time
1224	Illegal zoom state
1225	Illegal center frequency
1226	Illegal ramp state
1230	Illegal clock source
1231	Illegal system sync mode
1232	Illegal trigger/sync line

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Error Number	Description
1233	Illegal sample clock master
1234	Illegal fast arm mode
1235	Illegal group measure state
1240	Pretrigger delay exceeds size of FIFO
1241	Illegal post trigger delay out of bounds
1242	Illegal trigger level
1243	Illegal trigger level selection
1244	Illegal trigger slope
1245	Illegal trigger mode
1246	Illegal trigger channel
1247	Illegal arm state (manual or auto)
1248	Illegal trigger state (manual or auto)
1250	Channel group ID is not valid
1251	Group or Channel ID is not valid
1252	Illegal module count
1253	No E1431 module at logical address
1254	Invalid module option
1255	Channel ID is not valid
1256	Modules in group not in contiguous slots
1257	Number of modules exceeds limit of 255
1258	Expected group ID
1259	Expected channel ID
1260	Timeout on the status register state
1261	Bus error
1262	VXI register offset is illegal
1263	Illegal interrupt mask
1264	Illegal interrupt priority
1265	Error in system time function
1266	Unable to map A16 address space

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Error Number	Description
1267	Unable to open SICL vxi interface
1270	Module parameter not the same across channel group
1271	Unable to malloc memory
1272	Unable to reallocate memory
1273	Signal vector failed
1274	Measure state machine not responding
1275	Auto zero timed out waiting for data
1276	E1431 Internal library link list error
1277	Auto zero offset DAC failure
1278	Illegal AC settling setting
1279	Illegal mode in function call
1280	Illegal cal block specified
1281	Posible Hardware error
1282	Rom Vpp Range Error
1283	Rom Command Sequence Error
1284	Rom Byte Program Error
1285	Rom Block Erase Error
1286	Incorrect password
1287	Need either E1431_USER or E1431_FACTORY
1288	Illegal calibration state
1289	Selftest Failure
1290	Autophase could not zero phase

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A

The VXI Registers

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VXI Register Description

Register's Overview

HP E1431A complies with the register based VXIbus specification, revision 1.4. It is an A16, D16 device (16 bits address, 16 bits data) which can be set-up and controlled through an interface made of up to 32 read and 32 write registers.

The registers are located in the VXI A16 address space, at the address 49152 + (Logical Address * 64). The module's Logical Address may take a value between 0 and 255, as set by the Logical Address configuration switches. HP E1431A does not support the optional dynamic configuration as defined by the VXI specification.

The HP E1431A complies with the HP's Virtual Instrument / Local Bus system specification, for register based devices. This specification defines a set of registers on top of those for the extended devices.

The HP's virtual instrument register are classified into core registers (must be implemented), basic registers (must "appropriately" be implemented by devices with data transfer capabilities), and common registers (optional).

The description of the registers, use the following conventions:

VXI	Register defined in VXIbus specification
HP core	HP's virtual instrument register - required
HP basic	HP's virtual instrument register - data transfer capabilities (slave for HP E1431A)
HP common	HP's virtual instrument register - optional
8	Channel dependent register the contents of which depends on the value (0..7) in the <Channel Select> field of the Hardware SELection register
SU	Register access possible in the HP E1431A "Super User' mode only (see SU bit in Control register)
Reserved	Register is unused and reserved for future use. It's content is not relevant. If marked with a "*"

The current description covers those registers necessary for the HP E1431A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The HP E1431A's Registers

	Read	Write
3E	Trigger DELay 1 (LSW)	
3C	Trigger DELay 0 (MSW)	
3A	RESERVED	
38	RESERVED	
36	RESERVED	Phase Adjust (8)
34	Filter SETup 1 register	
32	Filter SETup 0 register	
30	Block SIZe register	
2E	Data FORmat register	
2C	Filter SENd register	Filter RECeive register (SU)
2A	Trigger LEVel 1 (8)	
28	Trigger LEVel 0 (8)	
26	Trigger Sample Source register	
24	Decimation Phase	RESERVED
22	Send DATa 1 (HP basic) (LSW)	RESERVED (HP basic) *
20	Send DATa 0 (HP basic) (MSW)	RESERVED (HP basic) *
1E	SUBClass register (VXI)	Filter ADDress 1 register (LSW)
1C	DESCription register (HP core)	Filter ADDress 0 register (MSW)
1A	Common CAPabilities register (HP core)	RESERVED *
18	Port CONtrol register (HP core)	
16	RESERVED (HP common) *	
14	IRQ config 0 register (HP common)	
12	RESERVED (HP common) *	
10	RESERVED (HP common) *	
0E	RESERVED (HP common) *	
0C	RESERVED *	
0A	RESERVED *	Analog SETup register (8)
08	Overload DETailed register (8)	D.C. OFfset register (8)
06	Hardware INFormation register	Hardware SELection register
04	STATus register (VXI)	CONTrol register (VXI)
02	Device TYPe register (VXI)	RESERVED (VXI) *
00	ID register (VXI)	RESERVED (Logical Address) (VXI) *

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Registers Grouping

For the purpose of their detailed description, registers have been grouped into subsections as follows:

Configuration: ID register (VXI), Device TYPe register (VXI), Logical Address (VXI)

Virtual Instrument Configuration: SUBClass register(VXI), DESCription register (HP core), Common CAPabilities register (HP core), Port CONtrol register HP core)

Operation Control: STATus register (VXI), CONTrol register (VXI), IRQ config 0 register (HP common)

Analog Front-End: Analog SETup register (8), D.C. OFfset register (8), Overload SUMmary register, Overload DETailed register (8)

Data Transfer: Block SIZe register, Data FORmat register, Send DATa 1 register (HP basic), Send DATa 0 register (HP basic)

Trigger: Trigger Sample Source register, Trigger LEVel 0 and 1, Trigger DELay 0 and 1

DigitalFilter: Filter SETup 0 and 1 registers,

System: Hardware SELection register, Hardware INFormation register, Filter ADDress 0 and 1, Filter SENd, and Filter RECeive registers

Registers Access

The DSPs read the registers during the "Booting" state only. Registers can be read or written to anytime, but only change the DSP state at "Booting".

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Configuration registers

ID Registers (VXI)

A read of this register provides information about HP E1431A's configuration. Its definition complies with the VXI specification.

ID Register (VXI) 0x00		
15 - 14	13 - 12	11 - 0
Device Class	Address Space	Manufacturer ID

Device Class: This field indicates the classification of the VXIbus device. Its value is (01) for an extended device, since HP E1431A supports the SUBClass register, as described in both the VXIbus and HP Virtual Instrument specifications.

Address Space: This field indicates the addressing mode of the device's operational registers. Its value is (11) for an A16 Only device.

Manufacturer ID: This number uniquely identifies the manufacturer of the device. Its value is FFFh (all ones) to indicate Hewlett-Packard.

Logical Address (VXI) *

This write register is defined by the optional Dynamic Configuration protocol. The HP E1431A does not support dynamic configuration (its logical Address is set by hardware switches), and therefore does not support this register. A write to this register will generate a bus error.

Device TYPe register (VXI)

This register contains a device dependent type identifier. Its definition complies with the VXI specification.

Device TYPE register (VXI) 0x02	
15 - 12	11 - 0
Required Memory	Model Code

Required Memory: This field is significant for A16/A24 and A16/A32 devices. For A16 Only devices (i.e. HP E1431A), it returns the upper part of the manufacturer Model Code (0000).

Model Code: This field contains a unique card identifier which is defined by the manufacturer. Its value is 0x01C8 for HP E1431A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Virtual Instrument Configuration registers

SUBClass Register (VXI)

The read of this register gives the subclass of the device. Its definition complies with the VXI specification.

SUBClass register (VXI) 0x1E		
15	14 - 12	11 - 0
0	Manufacturer Subclass	Manufacturer ID

Manufacturer Subclass: (111) - indicates that the device is a Virtual Instrument.

Manufacturer ID: 0xFFF - indicates that the device is an HP product

DESCription Register (VXI)

This register indicates whether or not HP E1431A supports some general capabilities.

DESCription register (HP core) 0x1C										
15 - 14	13 - 11	10 - 8	7	6	5	4	3	2	1	0
Rev	Sub Type	Mod Type	0	A16	VME Slave Send	VME Slave Rec	VME Mast Send	VME Mast Rec	LBUS Send	LBUS Rec

Rev: (11) - indicates the current revision of the Virtual Instrument specification

Sub Type: (111) - indicates an unspecified sub-classification within the module type

Mod Type: (011) - indicates a signal acquisition / conditioning front end

A16: (1) - indicates that Virtual Instruments' basic registers are in A16 space

VME Slave Send: (0) - indicates that the device does not implement the VMEbus Slave protocol for sending data to another virtual instrument. Note that HP E1431A implements the Send DATa 1 and Send DATa 0 registers, but not the Send Status register. Therefore it can be a Slave sender.

VME Slave Rec: (0) - indicates that the device does not implement the VMEbus Slave protocol for receiving data from another virtual instrument.

VME Mast Send: (0) - indicates that the device cannot be a Master sender to another virtual instrument.

VME Mast Rec: (0) - indicates that the device cannot be a Master receiver from another virtual instrument.

LBUS Send: (1) - indicates that the device can send Local Bus Data.

LBUS Rec: (0) - indicates that the device cannot receive Local Bus Data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Common CAPabilities Register (HP core)

This register indicates whether or not the HP E1431A implements certain optional registers.

Common CAPabilities register (HP core) 0x1A							
15 - 7	6	5	4	3	2	1	0
000000000	Send Count	Rec Count	IRQ1	IRQ0	Send Frame	Rec Frame	Slave Address

Send Count: (0) - indicates that Send Count register is not implemented.

Rec Count: (0) - Indicates that the Receive Count register is not implemented.

IRQ1: (0) - indicates that the IRQ config 1 register is not implemented.

IRQ0: (1) - indicates that the IRQ config 0 register is supported.

Send Frame: (0) - indicates that the Send Frame register is not implemented.

Rec Frame: (0) - indicates that the Receive Frame register is not implemented.

Slave Address: (0) - indicates that the Slave Address register is not implemented.

Port CONtrol Register (HP core)

This register is used to control the data transfer capabilities of the HP E1431A.

Port CONtrol register (HP core) 0x18					
15 - 10	9 - 8	7 - 5	4 - 2	1	0
Reserved	LBus Mode	Send Port	Reserved	LBus Pipe	LBus Enable

LBus Mode: This field is interpreted by the module when the LBus Enable bit makes a transition from (0) to (1), LBus Pipe is set to (0), and Send Port is LBus (010). When those conditions apply, the following LBus Modes are defined:

- (00) module is data generator, i.e. no pipeline allowed
- (01) module is in append mode, i.e. pipeline until end of frame, then append
- (10) module is in insert mode, i.e. start as a data generator, then switch to pipeline
- (11) locks module in pipeline mode

Send Port: These bits specify which data port is to be used for sending data to another virtual instrument. Of the eight options defined under the HP Virtual Instrument specification, only two are recognized by HP E1431A:

- (010) selects the Local bus
- (011) selects the VMEbus slave Send DATa register

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

LBus Pipe: Writing a (0) to this bit puts the Local Bus into the mode described in the field LBus Mode. Writing a (1) to this bit puts the module in the pipe mode. At reset, HP E1431A will be in this mode.

LBus Enable: Writing a (1) to this bit enables the Local Bus as the data output port, rather than the VXI Send DATa register. Writing a (0) to this bit disables the Local Bus interface.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Operation Control registers

STATus register (VXI)

A read of this condition register provides information about HP E1431A's status as defined in the following table. Its definition complies with the VXI specification for bits 15, 14, 3 and 2. All other bits are device dependent.

STATus register (VXI) 0x04													
15	14	13	12	11	10	9	8	7	6 - 4	3	2	1	0
0	MODID*	ERROR	OVLD	Meas. Done	Armed	Block Ready	Read Valid	Wait	State	Ready	Passed	Sysfail Inhibit	Reset

Bit 15: (VXI A24/A32 devices) HP E1431A is an A16 device, therefore this bit always returns 0.

MODID*: A (1) in this field indicates that the device is not selected via the P2 MODID line. A (0) indicates that the device is selected by a high state on the P2 MODID line.

ERROR: A (1) in this field indicates that an error has occurred. The error cause may be found in the Hardware INFormation register. This bit is cleared when the Hardware Information register is read.

OVLD (OVerLoaD): This bit indicates that an overload is currently occurring or has occured on one or more of the 8 channels. More detailed information can be read out of the Overload register. This bit is cleared when the Hardware Information register is read.

Meas(ure) Done: This bit is set whenever the module is in the Idle state, or in the Measure state and data collection has stopped due to a FIFO overflow or to an end of block. When this bit is set, the module releases the Sync line. Once all modules release the line and it goes high, all module go to the Idle state.

Armed: This bit is set whenever the module is in the Trigger state, or is in the Arm state and has satisfied it's pre-trigger requirements. When this bit is set, the module releases the Sync line. Once all modules release the line and it goes high, all modules go to the Trigger state.

Block Ready: This bit will be set whenever there is sufficient data in the FIFO to represent a transfer block.

Read Valid: This bit is set whenever there is a valid data word available to be read via the Data register.

Wait: This bit is set (1) after an access has been made to a write register with a latency time. It is reset (0) as soon as the information in the register has been read by the addressee (front-end, digital filter). An attempt to access any of these latent registers while Wait is set will generate a bus error.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

State: These three bits indicate the current state of the measurement loop as follows: (000) Tested, (001) Booting, (010) Booted, (011) Sync/Settling, (100) Idle, (101) Arm, (110) Trigger, (111) Measure (a.k.a. Acquire).

Ready: This bit is set after completion of a power-on initialization sequence. A (1) in this bit, in combination with a (1) in the Passed field, indicates that the device is ready to accept its full set of operational commands.

Passed: A (0) in this field indicates that the device is either executing or has failed it's self test. A (1) indicates that the self test has successfully completed.

Sysfail Inhibit, Reset: These two fields allow you to re-read the corresponding bit values written in the CONTrol register. See their description in the CONTrol register.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CONtrol register (VXI)

A write to this register causes specific actions to be executed by HP E1431A. Its definition complies with the VXI specification for bits 15, 1 and 0. All other bits are device dependent.

CONTrol register (VXI) 0x04												
15	14 - 12	11	10	9	8	7	6 - 5	4	3	2	1	0
A24/A32	Reserved	Reserved	Reserved	Super User	Reserved	Pull Sync	OPCODE	Reserved	Reload Replay	Control Enable*	Sysfail Inhibit	Reset

A24/A32 Enable: A (1) in this field enables access to the device's A24 or A32 registers. Since HP E1431A is an A16 Only device, the value of this bit is not interpreted.

Super User: A (0) in this bit prevents access to the Filter RECeive register in user mode (see VME address modifier). A (1) changes mode to super user, therefore allowing access to this register. An attempt to access the Filter RECeive register when this bit is (0) will result in a bus error.

Pull Sync: Writing a (1) to this bit will cause measurement loop state transitions, and will eventually (if System Sync bit is set - see trigger setup) cause the module to pull the Sync line low. Pull Sync must be used with the 'NOP' OPCODE set (i.e. 00).

OPCODE: This field controls the measurement loop. See the measurement loop section for a complete description of the states.

- (00) NOP. No impact on the measurement loop state (see Pull Sync).
- (01) STOP. stops the measurement loop (except when booting). Must be cleared with a NOP after use.
- (10) REBOOT. boot AD2105 when Sync line goes active. Must be cleared with a NOP after use. Note - in multi-module system, all modules should be given desired OPCODE first then NOPs next.

Reload Replay: Writing a 1 to this bit while the E1431 is in replay mode (see filter setup1) causes the E1431 to consume on block of replay data.

Control Enable*: A (1) in this bit will cause the write to bits 3-13 to be ignored. In other words, this bit must be (0) for bits 3-13 to be affected.

Sysfail Inhibit: A (1) in this bit disables the device from driving the SYSFAIL* line.

Reset: A (1) in this bit forces the device into a Reset state (same as power-on).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Using Replay mode

Use the following procedure for Replay mode.

1. Set up as for a normal measurment. The input setting will be ignored.
2. Set to Replay mode.
3. Walk measurement state machine to trigger.
4. Load first 1K blocks of data into memory using filter address registers.
5. Pullsync to process first block.
6. When the active bit for all DSPs shows done, reload the next 1K data blocks and set the replay bit in the control register to 1.
7. Block available bit behaves as in real-time mode.

IRQ config 0 register (HP common)

This read and write register configures HP E1431A's interrupt source. It provides for selection of the VMEbus IRQ level used, and a device specific mask.

IRQ config 0 register (HP common) 0x14								
15 - 14	13	12	11	10	9	8	7 - 3	2 - 0,
Reserved	ERROR	Overload	Measure Done	Armed	Block Ready	Read Valid	Reserved	Priority

Mask (bits 13 - 8): When writing, these bits provide a mask for the corresponding bits of the STATus register. If any one of the status bits is set while its corresponding mask bit is set, then an interrupt will be generated. This is done by asserting the appropriate interrupt line on the VXI backplane. A read only echoes the Mask bits.

Priority: This field (write only) selects which of the seven priority interrupt lines (1-7) will be asserted by the interrupter. A (0) in this field turn off the interrupt. The interrupter implements Release On AcKnowledge (ROAK).

After ERROR IRQ read the Hardware Information register to clear latched condition (FIFO full or Filter Error).

After Overload IRQ, read the Hardware Information register to clear latched overload condition.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Analog Front-End registers

Analog SETup register (8)

A read is channel dependent. The setup of each analog front end is controlled by writing to this register, in conjunction with the "Channel Select" field of the Hardware SELection register.

Analog SETup register (8) 0x0A								
15	14 - 13	12	11	10 - 8	7	6	5	4 - 0
Active*	Reserved	Corr*	ICP	Input Source	Anti Alias*	Grounding	AC Coupling	Range

Active*: Writing a (0) to this bit activates that channel, and makes its data part of the FIFO Output. Writing a (1) to this bit makes that channel inactive, and no data is output for it.

Corr*: Writing a (0) to this bit enables amplitude corrections for the filter losses by the DSP filters. Writing a (1) to this bit will result in raw uncorrected data. The module must progress through booting for this bit to be recognized.

ICP: Writing a (0) to this bit causes the Voltage mode to be selected. Writing a (1) to this bit causes the current (ICP) mode to be selected.

Input Source: This field selects the input to the ADC as follows:

- (000) Ground
- (001) Input BNC on front panel
- (010) A bus common to all ADCs whose source is selected using the Hardware SELection register
- (011), (1xx) not used. Reserved for extensions.

Anti Alias*: Writing a (0) to this bit causes a low-pass filter to be applied to the input signal to eliminate aliasing effects introduced by sampling. Writing a (1) to this bit causes the filter to be by-passed. This bit only controls the analog part of the anti-alias filter (see the Filter SETup registers for controlling the anti alias digital counterpart).

Grounding: Writing a (0) to this bit causes the (differential) input to be floating. Writing a (1) to this bit causes the input to be grounded (through approximately 50 ohms).

AC Coupling: Writing a (0) to this bit causes the input to be DC coupled to the 1 Mohms input. Writing a (1) to this bit causes a capacitor to be put in series with the input connector. The resulting low frequency cutoff (-3 dB) is approximately 1 Hz.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Range: Writing to this field allows to select the attenuators/gain combination providing one of the 17 desired ranges:

Input Range (Volts)	Range bits	Tested[1]
10 V	0x05	yes
5 V	0x04	yes
3.16 V	0x03	no
2 V	0x02	yes
1 V	0x01	yes
500 mV	0x00	yes
316 mV	0x0b	no
200 mV	0x0a	yes
100 mV	0x09	yes
50 mV	0x08	yes
31.6 mV	0x13	no
20 mV	0x12	yes
10 mV	0x11	yes
5 mV	0x10	yes
3.16 mV	0x1b	no
2 mV	0x1a	no
1 mV	0x1f	no

[1] = Tested at factory. If no, not guaranteed to meet specification.

All other Ranges are reserved and should not be used.

D.C. OFfset register (8)

This register is channel dependent. The DC Offset of each analog front end (also called "zeroing") is controlled by writing to this register, together with the "Channel Select" field of the Hardware SELection register. This offset is intended for calibrating out the DC Offset of the input amplifier and ADC.

D.C. OFfset register (8) 0x08		
15 - 9	8	7 - 0
Not Used	Fine Mode	DC Offset Value

Fine Mode: Writing a (0) to this bit causes the offset value to be treated as a main value (also known as "coarse"). Writing a (1) to this bit causes the offset value to be treated as a fine value, for fine tuning.

Offset Value: A write to this 8 bits field defines the channel specific offset (either "coarse" or "fine", depending on the Fine Mode bit), for the current range. The scale of this DC Offset is dependent upon the selected range, since it is applied after the attenuators/gain stage.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Overload DETailed register (8)

This register is channel dependent. A read to this register provides detailed information on the overload conditions that occurred on one of the channels during the current/last measurement. This register "latches" overload conditions (for example, a positive transition / event) and is reset upon reading.

Overload DETailed register (8) 0x08			
15 - 3	2	1	0
Not Used	Common	Differential	ADC

Common: This bit indicates that a common mode overload occurred.

Differential: This bit indicates that a differential mode overload occurred.

ADC: This bit indicates that the A/D converter detected an overload.

Data Transfer registers

Block SIZe register

This read/write register defines the size of the data block (same for all channels.)

Block SIZe register 0x30	
15 - 14	13 - 0
Not Used	Block Size

Block Size: This field defines the blocksize expressed in number of samples minus one. Possible values for this parameter are 0, 1 - 16383, corresponding to a block size of 1 through 16384 (16k). Refer to the Data FORmat register for the format/size of the data. The Append Status data (6 words) is not counted for in the Block Size figure.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Data FORmat register

The data format out of the FIFO to either the Local Bus or the VME Bus is controlled by writing to this read/write register.

Data FORmat register 0x2E				
15 - 8	7	6 - 4	3	2 - 0
Not Used	Append Status	Channel Count	Block Mode*	Data Format

Append Status: Writing a (1) to this bit means that an extra block of status data is appended to the end of each data block transferred (therefore making the size of the transfer = blocksize + 6 words). Writing a (0) means that no status is appended to the end of the data block transferred. At reset, this bit is (0) (that is, status not appended).

This "tail" information consists of 6 words (16 or 32 bits, depending on data format), in the following order. If data size is 32, the low order 16 bits meaningless.

Decimation Phase counter (WORD 4). This word contains the time elapsed (in number of sample clock ticks) between the occurrence of the digital trigger, and the output of the digital filter. The value is the same for all channels, since they are always set with the same span. The top two bits are the state of the divider that is generating the sample clock from the oversample clock (64K from 192K) at the time of the trigger.

Service bits (half-range and trigger, overload) in positions 15, 14, 13 (if 16 bits word) or in 31, 30, 29 (if 32 bits word). The bits are active high, all remaining bits are set to "1". Trigger indicates that particular channel caused trigger.

Channel Count: Writing a number between 0 and 7 to this field allows you to select the number of active channels (1 to 8), specifically those for which the FIFO controller outputs data to the bus.

Block Mode*: Writing a (0) in this bit selects the block mode data transfer. This means that the module will stop collecting data as soon as one block of data has been collected after a trigger. Writing a (1) in this bit selects continuous mode. This means that data collection will continue, generating multiple blocks of data, until the FIFO overflows or until the module is explicitly programmed to stop.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Data Format: Writing to this field selects the format of the sample. In the following, d is data, o is overload, h is half-range, t is trigger, and p is passcount. The most significant bit (31/15) is on the left. Possible formats are:

code	format
(000) 16 BITS	dddddddd | dddddddd
(010) 32 BITS	dddddddd | dddddddd | ddddd000 | 00000000
(010) 32 BITS	dddddddd | dddddddd | 00000000 | 000ppppp
(011) 32 BITS	dddddddd | dddddddd | ddddd000 | hto00000
(011) 32 BITS	dddddddd | dddddddd | 00000000 | htoppppp

Formats with the pass count field "ppppp" are used when the selected output mode is multi-pass (see Filter SETup).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Send DATa 1 register (HP basic) and Send DATa 0 register

This register contains the data words to be transferred to a VMEbus master. 16 bit word use only SEND DATA 0, 32 bit data uses both.

Send DATa 1 register (HP basic) 0x22
15 - 0
Data word (LSW)

Send DATa 0 register (HP basic) 0x20
15 - 0
Data word (MSW)

These registers provides access to the output of the data FIFO, provided that the device has been programmed to use the VME backplane for data output. The availability of data in these registers is indicated by the Read Valid bit in the STATus register. After each read of the Send DATa X registers, it is loaded with the next available word from the FIFO. Attempt to read this register when empty will result in holding the DTACK, with a BUS ERROR reported when VME time-out occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Trigger Sample Source register

Trigger Sample Source register

This read/write register controls if and when this module will initiate a sync, an arm, and/or a trigger event. A trigger can only be generated when the module is in the Arm measurement state and the Sync line is high, indicating that all modules are ready for a trigger.

Trigger Sample Source Register 0x26							
15 - 10	9	8	7 - 6	5	4 - 2	1	0
Reserved	Reserved	System Sync	TTLTRG Lines	FS Master	FS Source	Auto Arm	Auto Trigger

System Sync: When this bit is high (1), all modules on the same set TTLTRGn lines share the same system sync, any can generate sync, all use sync. When this bit is low (0) the sync line is internal to the module.

TTLTRG Lines: This field indicates which pair of VXIBus TTLTRG lines are used for system synchronization/arming/triggering (a.k.a. Sync line) (even lines), and system sample clock (odd lines) as follows:

- (00) for TTLTRG0, TTLTRG1
- (01) for TTLTRG2, TTLTRG3
- (10) for TTLTRG4, TTLTRG5
- (11) for TTLTRG6, TTLTRG7

FS Master: When this bit is set (1), the module is responsible for generating the system Sample Clock on the VXIbus (see TTLTRG Lines), from its local source.

FS Source: These three bits indicates the source for the sampling frequency. Possible sources are:

- (000) the internal sample clock 1 (196.608 kHz) decimated by 3
- (010) the Sample Clock line from the VXIbus (TTLTrg<n>) decimated by 3

Auto Arm: A (1) in this bit sets the module to perform an auto-arm, when appropriate.

Auto Trigger: A (1) in this bit sets the module to perform an auto-trigger, when appropriate.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Trigger LEVel 0 and 1 register (8)

These read and write registers encode the trigger threshold level for the selected trigger source. The eight registers are channel dependent, and selected with the Channel Select field of the Hardware SELection register.

Trigger LEV 0 and Center Freq register 0x28		
15	14	13 - 0
Mode	Slope	Level

Mode: This bit defines how the trigger levels apply. When (0), the trigger detection works when crossing a level (i.e. level detection trigger). When set to (1), the trigger detection works in "bounded triggering".

Slope LEVEL Triggering: When slope set to (0), a positive crossing of the lower, then upper level will cause a sync. When slope is (1), a negative crossing of the upper then lower level will cause a sync.

BOUNDED Triggering: When slope set to (0), an exit of the bounded range will cause a sync, When slope is (1), an entry into the bounded range will cause a sync.

Level: This value defines the lower threshold. It is the top 14 bits of the 16 bit DAC data.

Center Frequency: This register set the center freq of the ZOOM when the E1431 is in replay mode. The value is in unsigned Hz so 16 bits allow a range of 0 to 65535 Hz center frequency.

Trigger LEV 1 register 0x2A		
15	14	13 - 0
Active*	Reserved	Upper Level

Active*: This field indicates whether the channel has the capability (0) or not (1) to digitally trigger the module. In replay mode, a read of this bit reports the state of the replaying DSPs.
1= Done replaying 0=Busy replaying. You should reload the input replay data only when the DSPs are indicating done.

Level: This value defines upper threshold. It is the top 14 bits of the 16 bit DAC data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Trigger DELay 0 and 1 registers

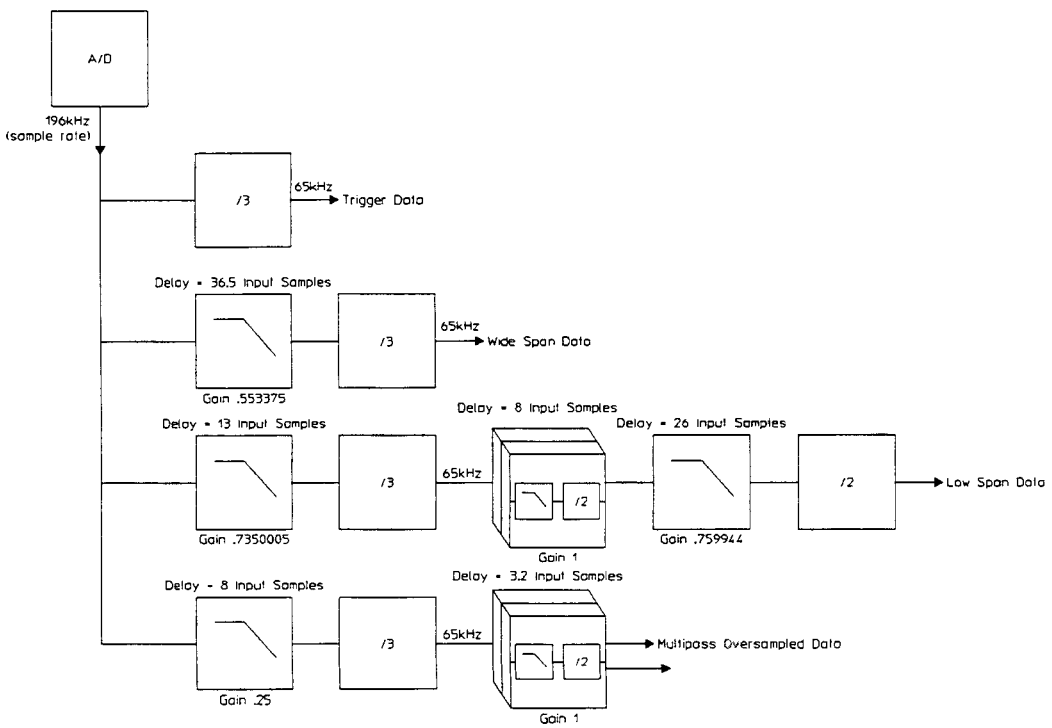
These read and write registers determine the pre- or post-trigger delays, expressed in "number of samples ticks". These registers can be considered to be a sign-magnitude representation of the delay, For example, post-trigger with delay 0 and pre-trigger with delay 0 put the same data into the FIFO.

Trigger DELay 1 register 0x3E			Trigger DELay 0 Register 0x3C
15	14 - 4	3 - 0	15 - 0
Pre/Post Delay	Reserved	Delay (upper bits)	Delay (lower bits)

Pre/Post Delay: either (0) post-trigger delay, or (1) pre-trigger Delay.

Delay: This value is 20 bits long, and ranges from 0 to the following limits: Max. value is 16383, or (00000011111111111111) for the pre-trigger delay (i.e. block size -1). Max. value is 1M, or all ones (11..11) for the post-trigger delay (i.e. 16 sec. at 65k sampling frequency).

Note: Maximum pre-trigger delay is current block size - 1.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Digital Filter registers

Filter SETup 0 register

This read/write register is used to set up the digital filters' mode of operation. This setup is the same for all channels.

Filter SETup 0 register 0x32			
15 8	7 - 4	3 - 1	0
Settling time	Reserved	Reserved	Anti-Alias*

Settling Time: This field contains the settling time of the digital filter, expressed in number of samples. Its value ranges from 0 (no settling) to 255. At reset, its value is internally set to 64 samples.

Anti-Alias*: Writing a (0) to this bit causes a low-pass filter to be applied to the digitized signal to eliminate aliasing effects introduced by sampling. Writing a (1) to this bit causes the filter to be by-passed. This bit only controls the digital part of the anti-alias filter (see the Analog SETup register for controlling the anti alias analog counterpart).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Filter SETup 1 register

This read/write register is used to set up the digital filters' span, output mode and zoom on/off. This setup is the same for all channels.

Filter SETup 1 register 0x34							
15	14	13	12 - 8	7	6	5	4 - 0
Replay	32 Bit	Reserved	Reserved	Ramp	Output Mode		Decimation Bandwidth

Replay: Writing a 1 selects the replay mode for the HP E1431, so the FIFO data comes from the replayed data. Writing a 0 selects normal data acquisition from the input mode.

32 Bit: Writing a 1 selects the 32 bit FIFO mode for replay only. 0 selects 16 bit FIFO. This bit is not used in normal input mode.

Output Mode: Writing a (00) selects single pass output. Writing a (01) selects the multi-pass output. Writting a (10) selects the ZOOM during replay. (11) is reserved.

Decimation Bandwidth: This field allows you to select the decimation bandwidth. Possible values for this field are 0 through 16. The equation for the span is given in the last line of the following table.

Decimation Bandwidth	Single Pass	Single Pass		Multiple Pass Over Sampled	
	Bandwidth	Data Rate	Trig Delay	Bandwidth	Data Rate
0	25.6 kHz	65536	12.2	NA	NA
1	12.8 kHz	32768	15.2	6.4 kHz	65536
2	6.4 kHz	16384	16.1	3.2 kHz	32768
3	3.2 kHz	8192	16.5	1.6 kHz	16384
4	1.6 kHz	4096	16.8	800 Hz	8192
5	800 Hz	2048	16.9	400 Hz	4096
6	400 Hz	1024	16.9	200 Hz	2048
7	200 Hz	512	17	100 Hz	1024
8	100 Hz	256	17	50 Hz	512
9	50 Hz	128	17	25 Hz	256
10	25 Hz	64	17	12.5 Hz	128
11	12.5 Hz	32	17	6.25 Hz	64
12	6.25 Hz	16	17	3.125 Hz	32
13	3.125 Hz	8	17	1.5625 Hz	16
14	1.5625 Hz	4	17	781.25 mHz	8
15	781.25 mHz	2	17	390.625 mHz	4
16	390.625 mHz	1	17	195.3125Hz	2
N	$25.6K/2^N$	$2^{(16-N)}$	See Trigger	$12.8k/2^N$	$2^{(15-N)}$

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Phase Adjustment register

These set of eight registers control the phase adjustment filters in the digital signal path within the DSPs. The adjustment is to correct the error introduced by the analog anti-alias filters. The scaling is .31 nanoseconds per bit. Range is +32K to -32K.

Phase Adjust register 0x36
15 - 0
Phase adjustment .31 nS/bit

Digital Filter registers

Hardware SELection register

This write register allows you to request hardware information and to set various test modes for the module.

Hardware SELection register 0x06					
15 - 11	10 - 8	7	6 - 5	4 - 3	2 - 0
Reserved	Hardware Mode	Reserved	Front-End Select	Reserved	Channel Select

Hardware Mode: This field allows to select different results to be read out of the Hardware INFormation register:

- (000), selects information about the hardware. In that mode, the lower bits of the Hardware SELection register act as the channel selector.
- (010), reserved for Xilinx 3 diagnostics
- (011), reserved for Xilinx 4 diagnostics
- (100), reserved for Xilinx 5 diagnostics
- (101), selects the Analog Front End diagnostic.
- (110), selects the 2105 and memory information (revision ...)
- (111), selects the FIFO diagnostic.

Front-End Select: This field selects a source for a common front end bus. The ADC can select this common bus with the Analog SETup register.

- (00), Ground
- (01), the VXI analog line SUMBUS
- (10), Test signal, 1024 Hz (use AC coupling and 100 mVolt range)
- (11), Ground

Channel Select: This field selects the channel for which channel dependent register writes or reads are performed (when hardware mode bits are set to (000)). A value of 0 through 7 allows you to select channels 1 through 8, respectively.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Hardware INFormation register

This read register provides hardware information as well as results on various tests that could be performed off measurement on the module, namely on its Analog Front End, and on its FIFO operation.

The type of information returned depends on the value set in the hardware mode field of the Hardware SELection register. When selecting hardware information (hardware mode (000)):

Hardware INFormation register (alternative 1) 0x06												
15 - 12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	Error	FIFO Full	OVLD	Filter Error	Reserved	Reserved	Reserved	Local Sync	Reserved	Reserved	Reserved	SU

ERROR: A (1) in this bit indicates that an error has occured. This bit is cleared by a read of Hardware INFormation register

FIFO Full: This error bit indicates that the FIFO has overflowed.

Filter Error: This error bit indicates that at least one of the digital filters is not operating properly. It is significant only when the ERROR bit in the status register is set. Note that the ERROR bit in the STATUS register and the mirror image in this register is a latched "OR" of the FIFO Full and Filter ERROR bits.

The Filter Error bit may be set after the board is reset. Be sure to read this register after a board reset, power on, or flash reprogramming

Local Sync: This status bit signals that the module is either locally booted, locally synchronized, locally armed or locally triggered.

SU: Echoes the Super User (SU) bit of the control register.

A read of the Hardware Information register clears the IRQ conditions: FIFO Full, Filter ERROR, Overload. These IRQ conditions are latched and cleared only with Hardware Information register reads.

When selecting tests (hardware mode other than (000)):

Hardware INFormation register (alternative 2)
15 - 0
Data Value

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Filter ADDress 0 and 1 register

This register allows to select an address in the digital filters' memory space. Subsequent read and/or writes are performed through the Filter SENd and/or Filter RECeive registers.

Filter ADDress 1 register 0x1E		Filter ADDressd 0 register 0x1C
15 - 2	1 - 0	15 - 0
Reserved	Address (MSW)	Filter's Memory Address (LSW)

Address (MSW): The most significant word of the digital filter's memory address.

Filter's Memory Address (LSW): Any valid address in the memory space (ROM or RAM) of the AD2105s.

Filter RECeive register (SU)

This write register contains the value to be written in the memory space of the digital filters. The write must be performed in super user mode (see SU in CONTrol register) to avoid bus errors.

Filter RECeive register 0x2C
15 - 0
Data Value (to be written)

When writing in ROM, the data value uses only the eight lower bits. Writing in the AD2105 ROM is done to load/change the calibration coefficients, or to load/change the code. The tool for writing in ROM is "hwzap" (see Programmer's Toolkit), and follows the specifications of the Flash ROM device (Intel 28F001) used on the HP E1431A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Filter SENd register

This read register contains the value read out of the memory space of the digital filters.

Filter SENd register 0x2C
15 - 0
Data Value (read)

When reading ROM, the data value uses only the eight lower bits, and the higher bits are all set to 1's.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Register' Contents upon reset

After a reset, the registers (and corresponding hardware) of the module are set to the following values:

Register	Reset Value (hexadecimal value)
00, r ID	0x7FFF
02, r Device TYPe	0x01C8
04, r Status	0x400C
04, w Control	0
06, w Hardware Selection	0
06, r Hardware Information	0
08, r Overload Detailed	0
08, w DC Offset	0
0A, w Analog Setup	0
14, r/w IRQ0	0
18, r/w Port Control	0x0062
1A, r Common Capabilities	0x0008
1C, r Description	0xFB42
1C, w Filter Address 0	0
1E, r Subclass	0x7FFF
1E, w Filter Address 1	0
22, r Send Data	0
22, w Receive Data	0
26, r/w Trigger Sample Source	0
28, r/w Trigger Level 0	0
2A, r/w Trigger Level 1	0x8000
2C, r Filter Send	0
2E, r/w Data Format	0x0080
30, r/w BlockSize	0
32, r/w Filter Setup 0	0x4000
34 , r/w Filter Setup 1	0
3C , r/w Trigger Delay 0	0
3E , r/w Trigger Delay 1	0

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Glossary

block mode The HP E1431A stops taking data as soon as a block of data has been collected. For more information, see "The HP E1431A's Measurement Process" in Chapter 7.

block size The number of sample points in a block of data.

call tracing Technique used to debug a C-Library program. The function call, *e1431_trace_level*, prints the parameters of the function each time it is called.

card sets Multiple HP E1431A modules can be grouped into a card set. SCPI considers a card set an "instrument." See "Addressing the HP E1431A" in Chapter 10 for more information.

channel lists Used in SCPI commands to specify parameters for instruments with multiple input channels. For more information see "Syntax Descriptions" in Chapter 11.

continuous mode The HP E1431A collects data continuously. It does not stop taking data unless the FIFO overflows. For more information, see "The HP E1431A's Measurement Process" in Chapter 7.

decimation filter A digital filter that simultaneously decreases the bandwidth of the signal and decreases the sample rate. The digital filter provides alias protection and increases frequency resolution. For more information, see *Spectrum & Network Measurements* available through your Hewlett-Packard Sales Office.

FIFO A First In, First Out buffer and controller which provides 16K samples for each of the eight channels.

group ID Any number of groups of channels may be declared and uniquely identified by a groupID through a call to *e1431_create_channel_group*. A channel may overlap groups.

header Files needed at compile time to interface with program code and C Library functions.

instrument-dependent commands In SCPI, multiple VXI cards in a set are equivalent to an "instrument." SCPI commands that are instrument-dependent set a parameter for all channels of an instrument, even if the channel was not specified.

Local Bus A high-speed port that Hewlett-Packard has defined as a standard byte-wide ECL protocol which can transfer measurement data from left to right at up to 2.62 Msamples per second on the VXI backplane.

logical address The VXI logical address identifies where each HP E1431A is located in the memory map of the VXI system. In the C-SCPI environment, the logical address must be a multiple of 8. (See "Addressing the HP E1431A" in Chapter 10 for more information.) The C Library environment does not have this same restriction.

module-dependent commands SCPI commands that are module-dependent change a parameter for all channels of the module; even when only one channel has been specified in the channel list.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

sampling rate The rate at which the measurement data is sampled. For the HP E1431A, the sampling rate is 2.56 times the frequency span. For more information about sampling theorem (and FFT properties) see *Spectrum & Network Measurements* available through your Hewlett-Packard Sales Office.

settled The digital filter waits N samples before outputting any data.

SICL HP Standard Instrument Control Library, a modular instrument communications library that works with a variety of computers.

SPIL The Signal Processor Interface Library for the HP E1485A/B.

system module The module with the lowest VXI logical address. It needs to be set to output the synchronization pulse for a multiple module group. All system sync pulses come from the system module.

target A channel, a group, or a module that is acted upon by a C library function.

VMEbus An industry standard bus on the VXI backplane for module control, setup and measurement data transfers. For measurement data transfers, the Local Bus offers higher transfer rates.

zooming Allows you to select a frequency span around a specified center frequency so you can focus on a specific frequency band.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Index

!

*cls command 11-8
*ese command 11-9
*esr? command 11-10
*idn? command 11-11
*opc command 11-12
*rst command 11-13
*sre command 11-14
*stb command 11-15
*trg command 11-16
*tst? command 3-3
*wai command 11-18

A

AC settling, setting 9-38, 11-31
accelerometers, power to 7-3
ADC, input signal 9-64, 11-55 - 11-56
addressing, in SCPI 10-20 - 10-24
adjustments 4-2
anti-alias filters
  enabling 9-42 - 9-43, 11-44, 11-50
  settling time 9-62, 11-21
arm
  auto 7-9, 9-46, 11-20
  manual 7-9, 9-2, 11-19
  specifying 11-20
  state, description 7-8, 9-23
  Waiting for ARM bit 10-17
assistance 1-i
auto-configuration 11-30
autorange 9-4, 11-55
autotrigger 9-48, 11-73
autozero 7-6, 9-5, 11-46

B

backdating 6-2
backplane connections 7-4 - 7-5
block diagram 5-3
block mode 7-8, 9-24, 9-56, 11-53
boot state 7-7
booting state 7-7
bus
  data transfer 7-4
  DTB 7-4
  local 7-4
  priority interrupt 7-4
  utility 7-4
  VME 7-10

C

C interface libraries
  compiling program 8-3
  debugging 8-4
  example program 8-3 - 8-4, 8-13 - 8-19
  grouping channels 8-5
  installing 1-7
  linking program 8-3
  logical address 9-3
  multiple channels 8-6
  multiple mainframes 8-7
  setting parameters 8-5
  support reference 9-1
C-SCPI
  description 10-2
  installing driver 1-8
  See also SCPI
calibration
  cycle 2-2
  enabling 9-50, 11-22
  reading 9-9
channel
  activating 9-39
  addressing 9-3, 9-15
  grouping 8-5, 9-12, 9-14, 9-47
  programming 8-6
  specifying 11-23
circuit description 5-2
clock
  See sample clock
collecting data 7-8 - 7-9, 9-24, 9-56, 11-53

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

command reference (C library)
See C interface libraries
command reference, SCPI
conventions 11-5
description 11-2
finding a command 11-3
symbols 11-4
syntax descriptions 11-6
command structure, SCPI 10-3
common commands, SCPI 11-8 - 11-18
condition register
described 10-7
operation status 10-17, 11-57
questionable status 11-61
status byte 10-12
conflicting conditions 8-6
continuous mode 7-9, 9-56, 11-53
cooling requirements 2-6
coupling, input 9-54, 11-43
cover part numbers 4-5
cross-channel time delay match 7-6
cross-mainframe phase accuracy 7-6

D

data
16-bit, unscaled 9-29, 9-58, 11-25
32-bit, unscaled 9-29, 11-25
available 9-7 - 9-8, 11-57
blocksize 9-49, 11-54
checking overloads 9-11, 11-29
collecting 9-56, 11-53
format 7-11, 9-55, 11-40
output 9-55, 11-40
reading 9-29, 11-23 - 11-24, 11-40
replay 7-15, 9-26, 9-33, 9-76
sample size 9-58, 11-28
scaling 9-20, 11-26 - 11-27
setting format 9-55, 11-40
specifying port 9-57, 11-77
status information 9-11, 11-29
transfer bus 7-4
transferring 7-10, 11-28
data collection
block mode 7-8, 9-24, 9-56, 11-53
continuous mode 7-9, 9-56, 11-53
port 9-57, 11-77
setting mode 9-56, 11-53
data sheet 2-2
DC offset, correcting 9-5, 11-46
debugging 8-4, 9-13, 9-28, 9-74, 9-87, 9-90
decimation filter
output 9-61
setting parameters 9-60
setting span 9-59, 11-51
settling time 9-62
default logical address 1-3
device reset 9-22 - 9-25, 11-13
digital AAF 9-60, 11-50
driver
installing C-SCPI 1-8
installing HP VEE 1-6
installing SCPI 1-9
DSPs
status 9-26
time offset 9-78, 11-66
DTB arbitration bus 7-4

E

eerom, programming 9-10
enable register
described 10-7
operation status 11-58
preset 11-60
questionable status 11-62
Status Byte 10-13
^ END 11-5
End or Identify (EOI) 11-5
errors
printing 9-28
reading 9-18, 11-64
event register
described 10-7
operation status 11-59
questionable status 11-63
standard event 10-15
example program
C libraries 8-13 - 8-19
HP E1485A/B environment 8-4
HP-UX environment 8-3
SCPI 10-22
exchange module 4-4

F

flash eerom 9-10
formatting data 9-29, 9-55, 11-40
frequency
bandwidth 9-77, 11-51
center in zoom 9-51
sampling rate 9-77, 11-52
span 9-77, 11-51
front panel

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

description 7-2
part numbers 4-6
removing 4-7

G

glossary 1-1
ground 7-4

H

handle part numbers 4-6
HP VEE, installing driver 1-6
HP VXI Installation Consultant 1-3
HP-IB
addressing commands 10-20 - 10-24
troubleshooting using 3-4

I

I/O initialization 9-21
ICP input mode
description 7-3
enabling 9-66, 11-42
identifying the module 11-11
idle state 7-7 - 7-8, 9-23
incoming inspection 1-2
initializing
I/O 9-21
measurements 7-7, 9-22 - 9-25, 11-41
input channel
activating 9-39, 11-48
autozero 11-46
coupling 9-54, 11-43
DC offset 9-5, 11-46
enabling AAF 9-42 - 9-43, 11-44
enabling ICP 9-66, 11-42
floating 9-65, 11-45
grounding 9-65, 11-45
mode 7-3, 9-66, 11-42
number available 11-78
offset 9-67, 11-46
phase calibration 9-5, 11-47
programming multiple 8-6, 10-20 - 10-24, 11-6
setting parameters 9-40, 10-20
shielding 11-45
input offset 11-33 - 11-34
input range
auto 9-4, 11-55
specifying 9-75, 11-56
installing
C interface libraries 1-7
C-SCPI driver 1-8
HP VEE driver 1-6
module 1-3
SCPI driver 1-9
interrupts
mask 9-69
priority 9-71
reenabling 9-32
setting 9-68

L

line feed character (NL) 11-5
local bus
description 7-4
logic level 1-3
reset 9-35, 11-35, 11-79
setting mode 9-72, 11-75
specifying 9-57, 11-77
logical address
description 10-20 - 10-24
in C interface libraries 9-3
setting 1-3

M

manufacturers' code 4-3
master summary bit (MSS) 10-9, 10-12
MAV bit 10-12
measure state 7-8, 9-24
measurement
blocksize 9-49, 11-54
channel grouping 9-47, 10-20 - 10-24, 11-6
initializing 7-7, 9-16, 9-22 - 9-25, 11-41
interrupting 7-9
Measuring bit 10-17
multiple mainframes 8-7
process 7-6 - 7-15
reset 9-36, 11-41
sample size 11-28
setting channels 9-39
setting span 9-77, 11-51
memory, managing 9-17, 11-13
Message Available bit 10-12
message, termination 11-5
module
installing 1-3
replacing 4-3
shipping 1-11
MXI interface 8-3

N

new line character (NL) 11-5

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

O

operating environment 2-6
operation status register set
  condition register 11-57
  description 10-17
  enable register 11-58
  event register 11-59
output 9-29, 11-40

P

part numbers 4-2
performance tests 2-2
phase calibration 7-6, 9-5, 11-47
phone assistance 1-i
polling method 10-8
post-trigger delay 9-82, 11-66
power requirements 2-6
power supplies 7-4
pre-trigger delay 9-82, 11-66
precision bits, setting 9-58, 11-28
preset 9-27, 11-60
printing errors 9-28
priority interrupt bus 7-4
program message terminators 11-5

Q

query
  form 11-2
  of register sets 10-18
questionable status register set 10-14
  condition register 11-61
  enable register 11-62
  event register 11-63

R

range, auto 9-4, 11-55
re-processing data
  See replay data
reading
  data 9-7 - 9-8, 9-29, 11-23 - 11-24, 11-40
  error messages 11-64
reading error messages 9-18
reenabling interrupts 9-32
register
  address 9-19, 11-36
  See SCPI register set
  See also VXI registers
repair strategy 4-2
replaceable parts 4-2
replay data
  description 7-15
  initializing 9-26
  specifying 9-33
request service bit (RQS) 10-9, 10-12
reset
  device 9-34, 11-13
  local bus 9-35, 11-79
  measurement 9-36, 11-41
RS-232, troubleshooting using 3-2

S

safety
  See inside front cover
  standards 2-6
sample clock
  external 7-12, 11-37
  in multiple mainframes 8-7
  internal 7-12, 11-37
  master 9-52, 11-38
  setting rate 9-59, 11-52
  source 9-53
sample programs
  See example program
sample size 9-58
samples, specifying 9-49, 11-54
sampling rate, setting 9-60, 11-52
scaling data 9-20, 11-26 - 11-27
SCPI
  addressing 10-20 - 10-24
  environments 10-2
  example program 10-22
  format 10-3
  installing driver 1-9
  instrument-dependent commands 10-5
  module-dependent commands 10-4
  Slot 0 10-2
  structure 10-3
  syntax 10-3, 11-4
  version 11-65
SCPI register set
  how to use 10-8
  master summary (MSS) 10-9
  operation status 10-17
  polling method 10-8
  questionable status 10-14
  request service (RQS) 10-9
  SRQ method 10-9
  standard event 10-15
  status byte 10-12
self test 9-37, 11-17

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

serial number, programming 4-3, 11-32
serial poll 10-9 - 10-10
service assistance 1-i
service request
  described 10-8
  enable register 10-10, 11-14
  generating 10-9
  initiating 10-10
  initiating SRQ 10-10
  monitoring conditions 10-9
setting channels 9-3, 11-48
setting format 9-29, 9-55, 11-40
setting parameters
  in SCPI 10-4
  for a channel 10-20
  with C libraries 8-5
settled state 7-7
settling time, setting 9-62, 11-21
shipping module 1-11
signal bus 9-63
software support reference
  See C interface libraries
  See also command reference, SCPI
space character (WSP) 11-4
special syntactic elements 11-4
specifications 2-2
SRQ
  described 10-9
  initiating 10-10
standard event register set 10-15
status byte 10-10, 10-12
  reading 11-15
  service request 11-14
status information
  checking overloads 9-11
  enabling 9-44, 11-29
SYNC line
  description 7-6
  selecting 11-39
  setting 11-49
  transitions 7-7
synchronization 8-6, 8-11, 9-73
syntax
  conventions 11-5
  message terminators 11-5
syntax descriptions 11-6
  CHANNELS 11-6
  CHAR 11-6
  DEF_BLOCK 11-6
  STRING 11-7

T

telephone assistance 1-i
tested state 7-7
transferring data
  description 7-10
  format 9-55, 11-40
  precision bits 9-58, 11-28
  via the local bus 7-10
  VME 7-10
transporting module 1-11
trigger
  auto 7-9, 9-48, 11-73
  enabling 11-74
  free run 9-48, 11-73
  lines 7-5
  manual 7-9, 9-91, 11-16, 11-68
  setting delay 9-79 - 9-80, 9-82, 11-66
  setting level 9-79 - 9-80, 9-83, 11-67, 11-69
  setting mode 9-79 - 9-80, 9-84, 11-70
  setting slope 9-79 - 9-80, 9-86, 11-71
  setting type 9-84, 11-73
  source 9-81
  specifying lines 9-88, 11-49
  state, description 7-8, 9-24
  time correction 9-92 - 9-96, 11-66
  Waiting for TRIG bit 10-17
troubleshooting
  using an HP-IB interface 3-4
  using an RS-232 interface 3-2
tst? command 11-17
TTLTRG lines
  configuring 11-30
  description 7-5
  setting 9-88, 11-49

U

utility bus 7-4

V

VMEbus
  description 7-10
  specifying 9-57, 11-77
voltage mode
  description 7-3
  specifying 9-66, 11-42
VXI
  backplane connections 7-4 - 7-5
  programming multiple mainframes 8-7
VXI registers
  address 9-19

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

description A-2 - A-28
reading 9-31, 11-36
using 7-6
writing 9-31, 11-36

W

WSP 11-4

Z

zooming
center frequency 9-51
description 7-15
setting 9-89

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Declaration of Conformity

According to ISO/IEC Guide 22 and EN 45014

Manufacturer's name:	Hewlett-Packard Company
Manufacturer's address:	Lake Stevens Instrument Division 8600 Soper Hill Road Everett, Washington 98205-1298

declares, that the product

Product Name:	25.6kHz Eight Channel VXI Input Module
Model Number:	HP E1431A

conforms to the following specifications:

Safety:	IEC 1010-1:1990+A1/EN61010:1993
EMC:	CISPR 11: 1990/EN55011 (1991), Group1, Class A IEC 801-2: 1991/EN50082-1 (1992): 4 kV CD, 8 kV AD IEC 801-3: 1984/EN50082-1 (1992): 3 V/m (1) IEC 801-4: 1988/EN50082-1 (1992): 1 kV

Supplementary Information:

The product herewith complies with the requirements of the Low Voltage Directive 73/23/EEC and the EMC Directive 89/336/EEC.

Everett, Washington - Feb 24, 1994

Manfred Bartz, Quality Manger

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Need Assistance?

If you need assistance, contact your nearest Hewlett-Packard Service Office listed in the HP Catalog, or contact your nearest regional office listed at the back of this book. If you are contacting Hewlett-Packard about a problem with your HP E1431A 25.6kHz Eight Channel VXI Input Module, please provide the following information:

- ❑ Model number: HP E1431A
- ❑ Software version:
- ❑ Serial number:
- ❑ Options:
- ❑ Date the problem was first encountered:
- ❑ Circumstances in which the problem was encountered:
- ❑ Can you reproduce the problem?
- ❑ What effect does this problem have on you?

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

About this edition

October 1994: Third Edition. In this edition, the "Using the C Interface Libraries" chapter was changed by adding information to the *e1431_init_measure* command. The "SCPI Command Reference" chapter was changed with corrections to several commands.

August 1994: Second Edition. In this edition, the "Specification" chapter was changed for enhancements to the specifications. The "SPCI Command Reference" chapter was changed for a correction to the VINStrument:LBUS:RESet command.

June 1994: First Edition.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

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