User Manual

Tektronix

VX4101
DMM/Counter

070-9678-01

This document supports firmware version 1.0

Warning
The servicing instructions are for use by qualified personnel only. To avoid personal injury, do not perform any servicing unless you are qualified to do so. Refer to the Safety Summary prior to performing service.

Printed in the U.S.A.

Tektronix, Inc., P.O. Box 1000, Wilsonville, OR 97070–1000

WARRANTY

Tektronix warrants that this product will be free from defects in materials and workmanship for a period of three (3) years from the date of shipment. If any such product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; or c) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THIS PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESSED OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

Status and Events

Appendices

General Safety Summary

Review the following safety precautions to avoid injury and prevent damage to this product or any products connected to it. To avoid potential hazards, use this product only as specified.

Only qualified personnel should perform service procedures.

While using this product, you may need to access other parts of the system. Read the *General Safety Summary* in other system manuals for warnings and cautions related to operating the system.

To Avoid Fire or Personal Injury

Connect and Disconnect Properly. Do not connect or disconnect probes or test leads while they are connected to a voltage source.

Ground the Product. This product is indirectly grounded through the grounding conductor of the mainframe power cord. To avoid electric shock, the grounding conductor must be connected to earth ground. Before making connections to the input or output terminals of the product, ensure that the product is properly grounded.

Observe All Terminal Ratings. To avoid fire or shock hazard, observe all ratings and marking on the product. Consult the product manual for further ratings information before making connections to the product.

Do not apply a potential to any terminal, including the common terminal, that exceeds the maximum rating of that terminal.

Do Not Operate Without Covers. Do not operate this product with covers or panels removed.

Use Proper Fuse. Use only the fuse type and rating specified for this product.

Avoid Exposed Circuitry. Do not touch exposed connections and components when power is present.

Do Not Operate With Suspected Failures. If you suspect there is damage to this product, have it inspected by qualified service personnel.

Do Not Operate in Wet/Damp Conditions.

Do Not Operate in an Explosive Atmosphere.

Keep Product Surfaces Clean and Dry.

Provide Proper Ventilation. Refer to the manual's installation instructions for details on installing the product so it has proper ventilation.

Symbols and Terms

Terms in this Manual. These terms may appear in this manual:

WARNING. *Warning statements identify conditions or practices that could result in injury or loss of life.*

CAUTION. *Caution statements identify conditions or practices that could result in damage to this product or other property.*

Terms on the Product. These terms may appear on the product:

DANGER indicates an injury hazard immediately accessible as you read the marking.

WARNING indicates an injury hazard not immediately accessible as you read the marking.

CAUTION indicates a hazard to property including the product.

Symbols on the Product. The following symbols may appear on the product:

WARNING High Voltage	Protective Ground (Earth) Terminal	CAUTION Refer to Manual	Double Insulated

Service Safety Summary

Only qualified personnel should perform service procedures. Read this *Service Safety Summary* and the *General Safety Summary* before performing any service procedures.

Do Not Service Alone. Do not perform internal service or adjustments of this product unless another person capable of rendering first aid and resuscitation is present.

Disconnect Power. To avoid electric shock, disconnect the main power by means of the power cord or, if provided, the power switch.

Use Care When Servicing With Power On. Dangerous voltages or currents may exist in this product. Disconnect power, remove battery (if applicable), and disconnect test leads before removing protective panels, soldering, or replacing components.

To avoid electric shock, do not touch exposed connections.

Preface

This manual assumes you are familiar with the operation of VXIbus instruments and with the purpose and function of this instrument.

Please read and follow all instructions for installation and configuration. Use the Installation Checklist to ensure proper installation and to record your initial settings.

The *Operating Basics* section gives a summary of VXIbus operation and presents an overview of the operation of this instrument.

The *Syntax and Commands* section provides a summary of all the commands followed by detailed descriptions of each command. *Appendix E: Examples* contains example programs that demonstrate the programmable features of this instrument.

The *Status and Events* section contains an explanation of the Status and Event Reporting System and lists the system messages.

The *Reference Guide* contains a summary of all the SCPI instrument commands.

Conventions

The names of all switches, controls, and indicators appear in this manual exactly as they appear on the instrument.

Specific conventions for programming are given in the sections *Syntax and Commands* and *Appendix E: Examples*.

Getting Started

Product Description

The VX4101 DMM/Counter is a C size single slot VXI instrument for use in a mainframe conforming to the VXIbus Specification. The VX4101 provides three of the most common functions needed in a typical VXI system:

- Full function 5 1/2 digit Digital Multimeter (DMM)
- Full function 2-channel 250 MHz Universal Counter
- Scanner master function for the Tektronix SurePath™ family of VXI relay modules

All instruments are VXI message-based and use command programming syntax compatible with Standard Commands for Programmable Instruments (SCPI) 1995 and IEEE 488.2. Included with the instrument is a complete VXI*plug&play* Win Framework instrument driver.

You can control the VX4101 either by using the VXI*plug&play* instrument driver or by issuing SCPI commands. Complete VXI*plug&play* driver descriptions are described in the help files on the plug&play diskette shipped with the instrument. A listing of the basic functions appear in *About the Instrument Drivers* in this manual.

You can load on-line help files and text files from from the diskette included with this instrument. Descriptions of the most basic commands you can use with the *VXIplug&play* instrument drivers appear in *About the Instrument Drivers.*

Instrument Descriptions

All instruments on the module are located at a single VXI logical address. You can access the instruments sequentially in the same manner as multiple instruments in a typical test system. They can be set up sequentially, then triggered and operated concurrently, with local on-instrument processing and memory for each function.

Extensive triggering capability is provided between each instrument and the VXI backplane.

Digital Multimeter (DMM)

The DMM has full function autoranging with 5 1/2 digit resolution. Measurements can be returned as single measurements, or as an array with up to 4096 measurements via the Fast Data Channel Protocol. Power cycle averaging modes are available for 60 Hz, and 50 Hz. The autozero and autogain capabilities minimizes offset and gain drift errors without removing the customer input connections. The DMM has floating input isolation of 300 VDC or AC $_{RMS}$. It also features programmable aperture time from 1 ms to 2 seconds in 1 ms steps.

Ranges of operation are:

- DC voltage at 30 mV, 300 mV, 3 V, 30 V, and 300 V
- AC True RMS, either AC or DC coupled, at 30 mV_{RMS}, 300 mV_{RMS}, 3 V_{RMS}, 30 V_{RMS} and 300 V_{RMS}
- 4-wire or 2-wire Ω measurement at ranges of 30 Ω, 300 Ω, 3 kΩ, 30 kΩ, 300 kΩ, 3 MΩ, 30 MΩ, and 300 MΩ
- DC Current at 150 mA and 1A

Acquisition triggering is provided from one of several sources, including VXI TTL triggers, counter front panel arm, software triggers, periodic trigger, counter end-of-measurement, and SurePath™ relay switched and settled.

Universal Counter

The Universal Counter in the VX4101 brings unprecedented performance to VXI in terms of both speed and resolution. The features include:

- Two input channels for frequency, period, rise and fall time, positive and negative pulse width, frequency ratio, totalizer, time interval, and AC/DC voltage measurements
- Optional 3 GHz prescaler input on channel 3
- 1 MHz to 250 MHz frequency measurements (500 MHz optional)
- Input signal conditioning: x1, x10, x100 attenuation with 0.5 to 10.0 variable gain and up to ±100 V offset depending on attenuation
- DC or AC coupling, and 50 Ω or 1-MΩ input (with automatic over current protection for the 50 Ω mode)
- 250 ns resolution (1 ps with averaging) and timing measurements from –1 ns single-shots to 99 day averaging
- Measurement gating from one of several sources, including VXI TTL triggers, counter front panel arm, software triggers, periodic trigger, DMM end-of-measurement, and SurePath™ relay switched and settled.
- Channel 2 can be used as a high speed gate for channel 1

SurePath™ Modules

The scanner master control supports the functions of the Tektronix SurePath™ family of VXI relay modules, such as the VX4320, VX4330, VX4350, and VX4380. The features of some of the SurePath™ family are as follows:

VX4320 1.3 GHz RF Multiplexer. This module is a single-wide VXIbus module intended to switch RF and high-frequency digital signals. It is arranged as eight 1×4 coaxial switches with a 50 Ω characteristic impedance. Each 1×4 section has five type SMB male snap-on subminiature coaxial connectors.

VX4330 120-Channel Scanner/Multiplexer. This module provides six 1×10 4-wire multiplexer (mux) sections. Each of these six sections can be independently configured under software control as two 1×10 2-wire mux, a 1×20 2-wire mux, or as a 1×40 1-wire mux. In addition, each section can be programmed to connect it to the section above or below to produce a 1×60 4-wire mux, a 1×120 2-wire mux, or a 1×240 1-wire mux.

VX4350 64-Channel SPST/SPDT Switching Module. This module provides 64 independent single-pole double-throw relays. A 0 Ω resistor is placed in series with the common contact of each relay. This resistor may be replaced by a resistor with a larger value to limit the current that flows through the relay in order to protect the relay contacts. Pads are provided for optional metal oxide varistors (MOVs). These varistors are connected from the common contact to the normally closed contact, and from the common contact to the normally open contact of each relay. Using the optional MOVs protects the relay contacts from over voltage conditions encountered when switching electrical power to inductive loads. The optional MOVs and current limiting resistors are user installed, and are not available as factory options.

VX4351 40–Channel, 10 Amp, SPST Switch Module. This module contains 40 SPST (form A) relays. Each relay may be controlled independently. The contacts of each relay are connected to one of three 30-pin high current connectors which are mounted on the module's front panel. Circuitry is included on the VX4351 to verify the control signals that are applied to each relay coil driver.

VX4380 256-Crosspoint Relay Matrix Module. This module provides four 4×16 2-wire matrix sections. Each section can be configured to connect either the four rows or the sixteen columns to the section above or below it to produce up to a 16×16 2-wire matrix or a 4×64 2-wire matrix.

The VX4101 SurePath™ Master provides serial I/O interface for control of SurePath™ relay modules. It also monitors the power fuses of all SurePath™ Relay modules, and provides a serial input interface to identify each module that it controls.

VX4381 4 × 4, 10 Amp, Dual 1-Wire Switch Matrix. This module provides two 4 × 4 single-wire 10 A matrix sections. These sections can be configured to operate as a single 4 × 4 two-wire matrix.

NOTE*. There is a query for the VX4101 which will return a list of the SurePath™ family relay modules that the VX4101 SurePath™ Master supports.*

Physical Description

Figure 1–1 shows the VX4101 switches and fuses, and Figure 1–2 shows the front panel.

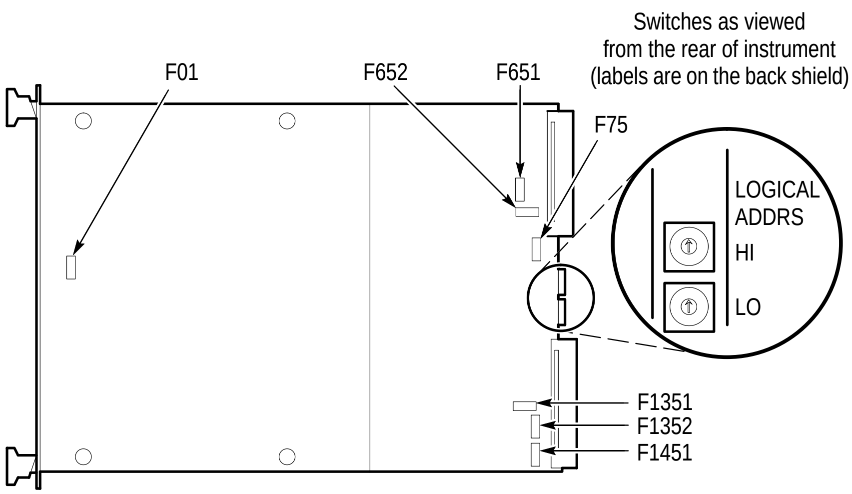


Figure 1–1: VX4101 VXIbus Connectors, Fuses, and Switch Locations

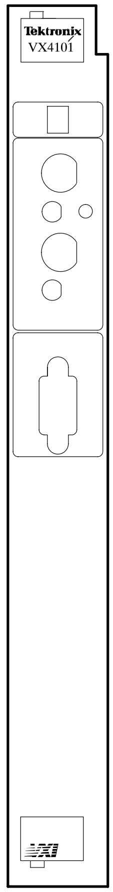


Figure 1–2: VX4101 Front Panel

Controls and Indicators

The following logical address switches must be correctly set to ensure proper operation. Refer to Figure 1–1 for their physical locations.

Logical Address Switches

The VX4101 supports VXI dynamic addressing. It is shipped with the switches set to FF so that the Slot 0 will automatically assign an address to the module. If you wish to set a specific address, please follow the instructions below.

Each functional module in a VXIbus System must be assigned a unique logical address, from 1 to decimal 255 (hexadecimal FF). The base VMEbus address of the VX4101 is set to a value between hexadecimal C0 ($C0_{16}$) and hexadecimal FF (FF_{16}) by two hexadecimal rotary switches. Align the desired switch position with the arrow on the module shield.

The physical address of the instrument is on a 64 byte boundary. If the Logical Address switch representing the most significant digit (LA-HI) of the logical address is set to position X and the switch representing the least significant digit (LA-LO) of the logical address is set to position Y, then the base physical address of the module will be [($40_{16} \times XY_{16}$) + $C000_{16}$]. For example:

			Base Physical Address	
L.A.	**HI**	**LO**	**decimal**	**hexadecimal**
A_{16}	0_{16}	A_{16}	(64 * 10) + 49152 = 49792	(40_{16} * A_{16}) + $C000_{16}$ = $C280_{16}$
15_{16}	1_{16}	5_{16}	(64 * 21) + 49152 = 50496	(40_{16} * 15_{16}) + $C000_{16}$ = $C540_{16}$

L.A. is the Logical Address

IEEE-488 Address

In order to use and program the VX4101 in an IEEE-488.2 environment, you must know the IEEE-488 address of the module. Different manufacturers of IEEE-488 interface devices have different algorithms for equating a logical address with an IEEE-488 address. Consult the operating manual of the IEEE-488 module being used.

LEDs

The VX4101 has four LEDs visible on its front panel. These LEDs are labeled as follows:

- Power LED – this LED is On if all six fuses for the six power buses are intact. Any single fuse being blown results in the LED turning OFF
- Fail LED – this LED is normally OFF. During power-on or reset self-test, the LED will be ON for the duration of the test. If during normal operation a failure in the VXI interface of the VX4101 is detected, the LED will come ON and the SYSFAIL line on the backplane will be true
- Message LED – this LED flickers ON when the VX4101 is being addressed on the VME backplane by its commander
- ERR LED – this LED is normally OFF. However, it may blink on and off to indicate error conditions. The most common reason is a command syntax error has been detected. Other error conditions that will cause the LED to blink are discussed elsewhere in this manual. Sending the “SYSTem:ERRor?” query to the instrument will return the cause of the error. When all errors in the queue have been retrieved, the error LED will return to the OFF state

Front Panel Connectors

Refer to *Appendix B:Input/Output Connections* for more information.

Fuses

The VX4101 has 6 fuses that limit the amount of current that each module can draw from the VXI backplane +5, -5.2, +24, -24, +12, and -12 V power pins. These fuses protect the module in case of an accidental shorting of the power bus or any other situation where excessive current might be drawn.

If any fuse opens, the module will assert SYSFAIL* on the VXIbus.

If the +5 V fuse opens, the VXIbus Resource Manager will be unable to assert SYSFAIL INHIBIT to disable SYSFAIL*.

If any fuse opens, remove the fault before replacing the fuse. Replacement fuse information is given in *Appendix H: Replaceable Parts*. Refer to the Barber pole page, following page C–10, before performing any service to this product.

BIT (Built-in Test)

The VX4101 performs a thorough interface self-test at power-up. This test ensures that the VXIbus interface is fully functional and ready for communication with the controller and that each instrument is initialized and ready for operation.

Built-In Test is provided by extensive self-tests which can be invoked via IEEE 488.2 and SCPI commands and queries.

Self-Test Highlights

The following are highlights of each test performed.

CPU Self-Test. The self-test for the CPU tests the following components:

- CPU RAM
- CPU timers, including periodic trigger
- Internal CPU interrupts
- SW triggers

DMM Self-Test. Circuitry tested on the DMM includes RAM testing, power supply and DMM reference testing, amplifier and A/D converter testing, and control circuitry testing. The self-test for the DMM tests the following components:

- The two 4-Kbyte DMM measurement buffers
- Logic registers

- Measurements of GND are taken at the 30 mV DC range, 3 VDC range, and 300 mV AC/DCV range
- The 2.75 V reference is tested in the 3 V range

Counter Self-Test. The self-test for the Counter tests the following components:

- The two 4-Kbyte Counter measurement buffers.
- Logic registers
- The analog front end pre-amp offset, pre-amp inverter, and pre-amp gain digital to analog converters (DACs).
- A 2.5 MHz signal is routed in through a test source and checked for accuracy.

SurePath™ Self-Test. While no specific self-test exists, operation of a SurePath™ relay card implicitly tests the control logic and data path for both the VX4101 and the SurePath™ card in use.

Accessories

The following table lists the standard accessories included with the VX4101.

Table 1–1: Standard Accessories

Accessory	Part Number
VX4101 User Manual	070-9678-XX
VX4101 Reference	070-9679-XX
DE-9 Connector	VX1784S
VXI*plug&play* 16-bit driver	063-2598-XX
VXI*plug&play* 32-bit driver	063-2822-XX

The optional accessories for the VX4101 include the following:

Table 1–2: Optional Accessories

Accessory	Part Number
BNC to BNC Cable	012-0057-01
SMA to SMA Cable	174-1428-00
SMA to BNC Cable	VX1760

Table 1–2: Optional Accessories (Cont.)

Accessory	Part Number
SMB to BNC Cable	VX1729
Modification Kit for VX1630 cable to use the VX4101 DMM with the VX4330 Scanner	VX1630 Option DM

VXI*plug&play* Software

VXI*plug&play* Soft Front Panels (SFPs) provide interactive control of VXI instruments using software virtual instruments and instrument drivers that are controller independent. SFPs are graphical user interfaces that emulate the physical controls and displays typically found on monolithic instruments. The instrument drivers call a common set of I/O control functions that are independent of instrument types, interface types, operating systems, programming languages, and networking mechanisms.

The installation program installs the VXI*plug&play* drivers as defined by the VXI*plug&play* Alliance for the framework appropriate for your processing environment, as follows:

- WIN
- WINNT
- WIN95

When the installation is complete, double clicking on the icon launches the Soft Front Panel which displays a representation of the traditional controls and indicators for an instrument. By selecting the appropriate controls on the SFP, you can verify that the instrument has been correctly installed and is functional, and perform almost all of the functions of the instrument. The soft front panel software will:

- Control the instrument
- Display data
- Provide command line query and response (Talk/Listen)
- Provide error and event reporting

See *Installation* for instructions on installing and running the SFP and using the C driver for program control. Detailed descriptions of the VXI*plug&play* drivers are given in on-line Help and text files on the disk shipped with the instrument.

Installation

This section describes how to install the VX4101 module in the mainframe, how to install the software, and how to run a functional check. At the end of the section, you will find a checklist to summarize your installation choices.

Tools Required

A slotted screwdriver set is required for proper installation.

Requirements and Cautions

The VX4101 Module is a C-size VXIbus instrument module and therefore may be installed in any C- or D-size VXIbus mainframe slot other than Slot 0. To install the module in a D-size mainframe, consult the operating manual for the mainframe. Refer to *Controls and Indicators* for information on selecting and setting the Logical Address switch of the module. This switch defines the programming address of your module.

CAUTION. *Note that there are two printed ejector handles on the card. To avoid installing the card incorrectly, make sure the ejector marked "VX4101" is at the top. Installing it incorrectly may damage the DIN connectors on the module.*

If the VX4101 is used in a Tektronix mainframe, all VX4101 cooling requirements are met.

NOTE. *If the VX4101 is inserted in a slot with any empty slots to the left of the module, the VME daisy-chain jumpers must be installed on the backplane in order for the VXI Module to operate properly. Check the manual of the mainframe being used for instructions on jumper settings. Jumpers are not necessary for autoconfiguring backplane designs such as those in Tektonix Mainframes. If there are no jumpers, there will be no interrupts and bus masters will not operate properly.*

Module Installation Procedure

Follow these steps to install the VX4101:

CAUTION. *The VX4101 Module is a piece of electronic equipment and therefore has some susceptibility to electrostatic damage (ESD). ESD precautions must be taken whenever the module is handled.*

1. Record the revision levels, serial numbers (located on the label on the top shield of the VX4101), and switch settings on the *Installation Checklist*.
2. Verify that the switches are set to the correct values. Refer to *Controls and Indicators* for more information on setting switches.
3. Make sure that the mainframe power is off.
4. Insert the module into one of the instrument slots of the mainframe (see Figure 1–3).

CAUTION. *Verify that the mainframe is able to provide adequate cooling and power with this module installed. Refer to the mainframe Operating Manual for instructions. If the mainframe cannot cool the unit adequately, the unit may not operate properly and may be damaged.*

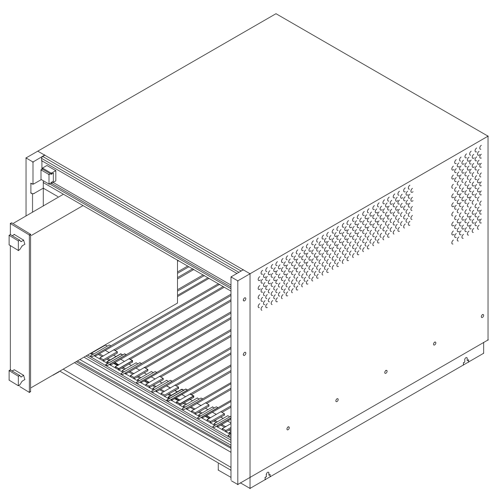


Figure 1–3: Module Installation

Software Installation

Each VXI*plug&play* instrument includes storage media containing a stand-alone, executable soft front panel (SFP) to control the instrument interactively. You can use the soft front panels immediately after installing the software. No particular application development environment (ADE) is necessary. The soft front panels and the supporting software were developed in conformance with the guidelines of the VXI*plug&play* Systems Alliance.

All VXI*plug&play* products are classified within a particular framework. The concept of framework was developed by the VXI*plug&play* Systems Alliance to categorize operating systems, programming languages, and I/O software libraries to bring the most useful products to the most end-users. The framework supported by the driver distributed with this VXI module is printed on the label of the media.

The source code as well as the dynamic link library are distributed so that the end user has the flexibility of using either in the end application.

The instrument driver distributed with the Tektronix VX4101 is VXI*plug&play* compliant. The driver uses VISA calls that are portable across platforms and development environments. Tektronix uses only the ANSI C implementation in instrument driver source code. No platform-specific libraries are included in the driver. The driver source code will compile using MSVC, Borland, Symantec or Watcom compilers without having to use foreign libraries other than the VISA Dynamic Link Library.

The MSVC and Borland import libraries for visa.dll are located in *VXIPNP\<Framework>\LIB\MSC* and *VXIPNP\<Framework>\LIB\BC* respectively. All driver dlls are located at *VXIPNP\<Framework>\BIN*.

Installation Procedure

Use the following procedure to install the VXI*plug&play* software:

1. Insert the media containing the driver files in the appropriate drive.
2. Locate the file *Setup.exe*, as follows:
   - On a 3 1/2 inch floppy disk, the file is on disk 1
   - On a CD ROM, the file will be in the *tkvx4101* directory
3. Launch setup.exe as follows:
   - In Windows or Windows NT/3.x, use the File menu and select Run. Then, browse to find setup.exe or type the drive letter and program name
   - In Windows 95 or or Windows NT/4.x, use Start and select Run. Browse to locate setup.exe and click on OK
4. Follow the directions of the installation program.

Following installation, driver files (see list below) will be found in locations defined by the VXI*plug&play* Alliance. Where required, modifications to your autoexec.bat and system.ini files may be automatically completed.

Driver Files

The instrument driver for the VX4101 is distributed with a number of C source code files, header files, dynamic link libraries and other supporting files. A breakdown of the modules that the files control are as follows:

VX4101. These are the high level "controller" portion of the driver. This part of the driver does the actual communication with the instrument. The tkvx4101.c or tkvx4101.dll have functions which perform VISA function calls. The other modules, making up the total driver, call functions in the tkvx4101 to get system configuration information and to communicate to each specific instrument.

Device-Specific Files. Those files supporting the specific instruments are as follows:

- Files controlling the DMM have the prefix *tkmpdmm*
- Files controlling the counter have the prefix *tkmpctr*
- Files controlling the SurePath™ family of scanners have the prefix *tkmpscan*

File Locations. For the VX4101 driver, the following files will be in the VXIPNP\<Framework>\TKVX4101 directory:

tkvx4101.c
tkvx4101.fp
tkvx4101.mak
tkvx4101.def
tkvx4101.hlp
tkvx4101.exe

Supporting driver files will be located in the following directories:

VXIPNP\<Framework\support\tkmpdmm:

tkmpdmm.c
tkmpdmm.fp
tkmpdmm.mak
tkmpdmm.def
tkmpdmm.hlp
tkmpdmm.exe

VXIPNP\<Framework\support\tkmpctr:

tkmpctr.c
tkmpctr.fp
tkmpctr.mak
tkmpctr.def
tkmpctr.hlp
tkmpctr.exe

VXIPNP\<Framework\support\tkmpscan:

tkmpscan.c
tkmpscan.fp
tkmpscan.mak
tkmpscan.def
tkmpscan.hlp
tkmpscan.exe

Several *.uir (User Interface Resource) files will support the GUI executables (CVI files). The following files are installed in the VXIPNP\<Framework>\INCLUDE directory:

tkvx4101.h
tkmpdmm.h
tkmpctr.h
tkmpscan.h
tkvx4101.bas
tkmpdmm.bas
tkmpctr.bas
tkmpscan.bas

The VXIPNP\<Framework>\BIN directory includes the following files:

tkvx4101.dll
tkmpdmm.dll
tkmpctr.dll
tkmpscan.dll

The VXIPNP\KB directory includes the following file:

tkvx4101.kb
tkmpdmm.kb
tkmpctr.kb
tkmpscan.kb

The VXIPNP\<Framework>\LIB\MSC\ directory includes the following files:

tkvx4101.lib
tkmpdmm.lib

tkmpctr.lib
tkmpscan.lib

The .DLL files are tested in LabWindows/CVI, LabView, HPVEE, Visual Basic, MSVC/C++ and Borland C/C++ before distribution. If you want to modify the driver algorithms, all files are distributed to facilitate rebuilding the .DLL file with the modifications.

***NOTE**. Tektronix recommends that you back up your original source files before modifying the driver files*

***CAUTION.** The Soft Front Panel (SFP) distributed with this driver is built using the distributed .DLL files. Any modification to the .DLL files used by the SFP could make it unusable. Re-installing the driver will write over any modified files with the original files and restore SFP operation.*

Installation Checklist

Installation parameters will vary depending on the mainframe being used. Be sure to consult the mainframe operating manual before installing and operating the module.

Revision Level:___________

Serial No.: ___________

Mainframe Slot Number: ___________

Switch Settings:___________

VXIbus Logical Address Switch: ___________

Cable Installed (if any): ___________

VXI*plug&play* software installed: ___________

Performed by: _________________________ Date: ______________

Functional Check

A VXIbus hard reset occurs when another device, such as the VXIbus Resource Manager, asserts the backplane SYSRESET* line. A VXIbus soft reset occurs when another device, such as the Slot 0 Controller, sets the Reset Bit in the VX4101 Control Register.

Self-Test

The VX4101 DMM/Counter has a two level hierarchy of self-tests. At power-on or on a VXIbus hard or soft reset, an extensive interface test is run. IEEE and 488.2 commands can be used to run more extensive self-tests of instrument specific functions. During the interface test ran at power-on or during a hard or soft reset the following actions take place:

1. The backplane SYSFAIL* line is asserted, indicating that the module is not ready for communication.
2. A test of the VXIbus interface logic is performed.
3. Each instrument is configured to its *RST initial state.
4. On successful completion of the interface test, the backplane SYSFAIL* line is negated and the VX4101 enters the VXIbus PASSED state (ready for normal operation). If the interface test is not successful, the backplane SYSFAIL* line remains asserted and the VX4101 enters the VXIbus FAILED state.

Instrument Self-Tests. Instrument specific self tests can be run at any time during normal operations. To run self-test solely on the active instrument, the TEST:ALL? query should be sent. To test all instruments on the VX4101, the *TST? query should be sent. See the instrument specific *Syntax and Commands* section for more detailed information on each self-test.

During a commanded self-test:

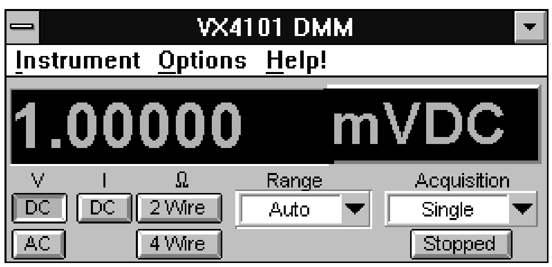
1. The backplane SYSFAIL* line is not asserted.
2. At the end of a successful instrument self-test, the instrument is placed in its *RST (power-on) configuration.
3. At the end of a successful *TST?, the entire VX4101 will be placed in its *RST (power-on) state.

Operational Check

To perform the operational check, do the following:

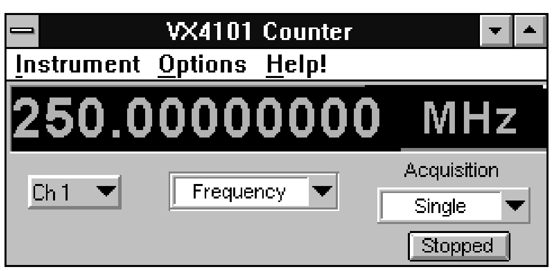
1. Double click on the TKVX4101 icon to start the Soft Front Panel.

2. Launch the DMM by clicking on the DMM icon.



3. Click on *Stopped* and the DMM will take a DC Voltage measurement.

4. Close the DMM and launch the Counter.



5. Using the center pulldown ring control select DC Volts and click on the Stopped button. The Counter will take a DC Voltage measurement.

6. Close the Counter and launch SurePath™.

7. If SurePath™ modules are present immediately to the right of the VX4101, any detected SurePath™ Module will have its button highlighted. Select one of the highlighted buttons. Close a relay. If there are no SurePath™ modules in the system, none of the buttons will be highlighted.

8. Close SurePath™ and exit the Soft Front Panel.

SYSFAIL* Operation

SYSFAIL* will operate under the following circumstances:

If any fuse opens, the module will assert SYSFAIL* on the VXIbus.

If the +5 V fuse opens, the VXIbus Resource Manager will be unable to assert SYSFAIL INHIBIT to disable SYSFAIL*.

***NOTE**. If a +5 V fuse opens, remove the fault before replacing the fuse.*

Replacement fuse information is given in the Specifications *section of the manual.*

About the Instrument Drivers

Tektronix distributes VXI*plug&play* compliant instrument drivers with VXI instruments. These drivers consist of functions required by the VXI*plug&play* alliance specifications as well as instrument specific functions. Tektronix has also developed a suite of auto-connection functions and other miscellaneous information gathering functions to make building applications easier.

System Requirements

The host computer must be capable of controlling VXI instruments and the Virtual Instrument Software Architecture (VISA) must be installed. Whether you are using a VXI embedded controller or an external computer (using GPIB or a VXIbus extender such as MXIbus) you may contact the vendor of your VXI Slot 0 resource device to obtain VISA.

All functions, including the instrument specific functions, are described fully in the Windows help file(s) distributed with the driver. The VX4101 driver also has examples using the driver function calls and accompanying readme text files explaining the operation of the examples. These examples are compilable files which use auto-connect and then perform configuration, measure and read functions to demonstrate driver usage. The following is an overview of all these non-instrument specific functions which are distributed with all Tektronix VXI*plug&play* instrument drivers.

tkvx4101_autoConnectToAll

Syntax

```
ViStatus tkvx4101_autoConnectToAll (ViSession instrumentArray[],
ViInt16 arrayLength, ViInt16 *numberConnected);
```

Description

This command attempts to find an instrument of this type in the system. It will connect to all instances of the instrument found. If no instrument of this type is found, then the autoConnectToAll command fails.

tkvx4101_autoConnectToFirst

Syntax

```
ViStatus tkvx4101_autoConnectToFirst (ViSession *instrumen-
tHandle);
```

Description

This function attempts to find an instrument of this type in the system. It will connect to the first instrument found if there are multiple instruments of this kind. There is no guaranteed order in the search algorithm. If no instrument of this type is found autoConnectToFirst will fail.

tkvx4101_autoConnectToLA

Syntax

```
ViStatus tkvx4101_autoConnectToLA (ViSession *instrumentHandle,
ViInt16 logicalAddress);
```

Description

This function attempts to find and connect to an instrument of this type with the specified logical address. If no instrument of this type is found with the specified logical address, the command will fail.

tkvx4101_autoConnectToSlot

Syntax

```
ViStatus tkvx4101_autoConnectToSlot (ViSession instrumentArray[],
ViInt16 arrayLength, ViInt16 *numberConnected, ViInt16 slot);
```

Description

This function attempts to find and connect to an instrument of this type in the specified slot. If no instrument of this type is found in the specified slot, autoConnectToSlot will fail.

tkvx4101_close

Syntax

```
ViStatus tkvx4101_close (ViSession instrumentHandle);
```

Description

This close function closes the communication channel with the instrument which is identified by the instrument handle. Once closed the instrument handle is no longer valid and must not be used.

tkvx4101_error_message

Syntax

```
ViStatus tkvx4101_error_message (ViSession instrumentHandle,
ViStatus errorCode, ViChar errorMessage[]);
```

Description This function takes an error number which was returned from one of the driver functions and returns a corresponding ASCII error message.

tkvx4101_error_query

Syntax

```
ViStatus tkvx4101_error_query (ViSession instrumentHandle,
ViInt32 *errorCode, ViChar errorMessage[]);
```

Description This function will query the instrument for errors which have occurred and return an error code. The instrument will return to zero if no errors have occurred.

vtkvx4101_getInstrDesc

Syntax

```
ViStatus tkvx4101_getInstrDesc (ViSession instrumentHandle,
ViChar instrumentDescriptor[]);
```

Description This function returns the instrument descriptor string of a connected instrument.

tkvx4101_getLogicalAddress

Syntax

```
ViStatus tkvx4101_getLogicalAddress (ViSession instrumentHandle,
ViInt16 *logicalAddress);
```

Description This function returns the logical address of a connected instrument.

tkvx4101_getManufacturerID

Syntax

```
ViStatus tkvx4101_getManufacturerID (ViSession instrumentHandle,
ViInt16 *manufacturerIdentification);
```

Description

This function returns the manufacturer identification number of a connected instrument. The manufacturer identification number was assigned to the manufacturer of this instrument by the VXI Consortium. This number is unique for each manufacturer.

tkvx4101_getModelCode

Syntax

```
ViStatus tkvx4101_getModelCode (ViSession instrumentHandle,
ViInt16 *modelCode);
```

Description

This function returns the model identification number of a connected instrument. This was assigned by the manufacturer of this instrument. This number is unique within each manufacturers product line.

tkvx4101_getSlotList

Syntax

```
ViStatus tkvx4101_getSlotList (ViInt16 slotArray[], ViInt16
slotArrayLength, ViInt16 *numberFound);
```

Description

This function returns a list of all slots which contain an instrument of this type.

tkvx4101_getSlotNumber

Syntax

```
ViStatus tkvx4101_getSlotNumber (ViSession instrumentHandle,
ViInt16 *slot);
```

Description

This function return the slot number of a connected instrument.

tkvx4101_getSlotandLAList

Syntax

```
ViStatus tkvx4101_getSlotandLAList (ViInt16 slotArray[], ViInt16
logicalAddressArray[], ViInt16 arrayLength, ViInt16 *number-
Found);
```

Description This function returns lists of all slots and corresponding logical addresses which contain an instrument of this type.

tkvx4101_getVisaRev

Syntax

```
ViStatus tkvx4101_getVisaRev (ViSession instrumentHandle, ViChar
revision[]);
```

Description This function retrieves the current working VISA revision.

tkvx4101_init

Syntax

```
ViStatus tkvx4101_init (ViRsrc resourceName, ViBoolean IDQuery,
ViBoolean resetDevice, ViSession *instrumentHandle);
```

Description This function opens a communication channel to the instrument. It returns a handle which is used to identify the communications channel for all other driver functions.

tkvx4101_reset

Syntax

```
ViStatus tkvx4101_reset (ViSession instrumentHandle);
```

Description This function resets the instrument to its default reset state.

tkvx4101_revision_query

Syntax
```
ViStatus tkvx4101_revision_query (ViSession instrumentHandle,
ViChar driverRevision[], ViChar instrumentRevision[]);
```

Description This function returns the revision of the driver and of the instrument firmware.

tkvx4101_self_test

Syntax
```
ViStatus tkvx4101_self_test (ViSession instrumentHandle, ViInt16
*selfTestResult, ViChar selfTestMessage[]);
```

Description This function executes instrument self test.

tkvx4101_sleep

Syntax
```
ViStatus tkvx4101_sleep (ViInt32 secondstoDelay);
```

Description This function allows for a delay in program operation.

Operating Basics

Instrument Functions

The VX4101 DMM/Counter is a VXIbus message-based instrument and communicates using the VXIbus Word Serial Protocol. The VX4101 is programmed by issuing ASCII characters from the system controller via the VXIbus commander and the VXIbus mainframe backplane. Refer to the manual for the VXIbus device that will be the VX4101 Modules commander for details on the operation of that device.

Functional Overview

The VX4101 provides three full function instruments in an economical package using a single VXIbus slot with one logical address.

To learn how the VX4101 controls functions of each instrument, see *Theory of Operations*.

Digital Multimeter (DMM)

The full function autoranging DMM has a 5 1/2-digit resolution. Measurements can be returned as single measurements, or as an array with up to 4096 measurements. Power cycle averaging modes are provided at 400 Hz, 60 Hz, and 50 Hz. The autozero capability minimizes offset drift errors without requiring removal of the customer input connections. Floating input isolation of 300 VDC or AC peak is provided.

Universal Counter

The Counter/Timer/Trigger (CTT) provides the full set of universal counter measurements:

- Two input channels for frequency, period, pulse width, rise/fall Time and totalizer measurements
- Two inputs for measurements of ratio, interval, phase angle, and high-speed gating
- Automatic measurements of AC/DC voltages
- x1, x10, x100 attenuation, variable gain selection for measurements from 10 mV to 200 V
- DC or AC coupling, and 50 Ω or 1 MΩ input (with automatic overcurrent protection for the 50 Ω mode)
- 250 ps resolution (1 ps with averaging) and timing measurements from –1 ns single-shots to 99 day averaging

- VXI Trigger, software command, periodic timer, and front panel arm input for gating
- Frequency input from 1 μHz to 250 MHz (500 MHz optional)
- Optional third channel capable of 200 MHz to 3 GHz frequency, period and ratio measurements

Scanner Master Control

The scanner master control supports the functions of the SurePath™ family of VXI relay modules.

The VXI Interface and VX4101 board provides serial I/O interface for control of scanner slave and SurePath™ relay modules. The daughter board also monitors the power fuses of all LocalBus slave modules, and provides a serial input interface to identify each local bus slave that it controls.

As part of the self test, SurePath™ modules automatically verify the control logic every time a relay operation is performed.

Power-on

The instrument runs its interface test and is ready for communication within five seconds. The VXIbus Resource Manager can add an additional delay. The Power LED will be on. The Failed LED will be off. The default condition of the module after power-on is listed in the *RST command description.

The format and syntax for the command strings are described in the *Command Syntax* section. A complete description of each command in alphabetical order is in the *Command Descriptions* section.

LEDs at Power-On. At power-on following a successful interface test, the front panel LEDs will be in the following states:

Power LED	On
Fail	Off
Message LED	Off
ERR LED	Off

Instrument I/O

CAUTION. *If the user's mainframe has other manufacturers' computer boards operating in the role of VXIbus foreign devices, the assertion of BERR* (as defined by the VXIbus Specification) may cause operating problems on these boards.*

VXIbus Basics

The VX4101 Module is a C-size single slot VXIbus Message-Based Word Serial instrument. It uses the A16, A32, and D16 VME interface available on the backplane P1 connector and does require A32 address space. The module is a D16 interrupter. The module will accept a D8, D16 or D32 interrupt acknowledge cycle.

The module supports the Normal Transfer Mode of the VXIbus using the Write Ready, Read Ready, Data In Ready (DIR), and Data Out Ready (DOR) bits of the module Response register. The read and write operations in Normal Transfer Mode are as follows:

Normal Transfer Mode Read Operation

A Normal Transfer Mode read of the VX4101 Module proceeds as follows:

1. The commander reads the VX4101 Response register and checks if the Write Ready and DOR bits are true. If they are, the commander proceeds to the next step. If not, the commander continues to poll these bits until they become true.
2. The commander writes the Byte Request command (hexadecimal 0DEFF) to the Data Low register of the VX4101.
3. The commander reads the VX4101 Response register and checks if the Read Ready bit is true. If it is, the commander proceeds to the next step. If not, the commander continues to poll this bit until it becomes true.
4. The commander reads the VX4101 Data Low register.

Normal Transfer Mode Write Operation

A Normal Transfer Mode write to the VX4101 Module proceeds as follows:

1. The commander reads the VX4101 Response register and checks if the Write Ready and DIR bits are true. If they are, the commander proceeds to the next step. If not, the commander continues to poll the Write Ready and DIR bits until they are true.

2. The commander writes the Byte Available command which contains the data (hexadecimal 0BCXX or 0BDXX, depending on the End bit) to the Data Low register of the VX4101.

The module has no registers beyond those defined for VXIbus message based devices. All communications with the module are through the Data Low register, the Response register, or the VXIbus interrupt cycle. Any attempt by another module to read or write to any undefined location of the VX4101 address space may cause incorrect operation of the module.

As with all VXIbus devices, the VX4101 Module has registers located within a 64 byte block in the A16 address space. The base address of the VX4101 device registers is determined by the device unique logical address and can be calculated as follows:

$$\text{Base Address} = V_{16} * 40_{16} + C000_{16}$$

where V is the logical address of the device as set by the Logical Address switches.

Configuration Registers

Table 2–1 contains a list of the VX4101 Configuration registers and a complete description of each register. The offset is relative to the module base address.

Table 2–1: Register Definitions

Register	Address (hexadecimal)	Type	Value (Bits 15-0)
ID Register	0000	RO	1001 1111 1111 1101 (hexadecimal 9FFD)
Device Type	0002	RO	0111 0111 1001 1010 (hexadecimal 779A)
Status	0004	R	Defined by state of interface
Control	0004	W	Defined by state of interface
Offset	0006	WO	Assigned by Resource Manager
Protocol	0008	RO	1011 0111 1111 1111 (hexadecimal B7FF)
Response	000A	RO	Defined by state of the interface
Data High	000C		Not used

Table 2–1: Register Definitions (Cont.)

Register	Address (hexadecimal)	Type	Value (Bits 15-0)
Data Low	000E	W	Not fixed; command-dependent
Data Low	000E	R	Not fixed; command-dependent

RO is Read Only
WO is Write Only
R is Read
W is Write

VMEbus Interrupt Level Selection

Each function module in a VXIbus System can generate an interrupt at a specified level on the VMEbus to request service from the interrupt handler located on its commander. The VX4101 supports programmable interrupt selection for setting the interrupt level for the device.

Interrupts are used by the module to return VXIbus Protocol Events to the module commander.

Theory of Operation

The VX4101 is a landmark VXI product incorporating a 5 1/2 digit Digital Multimeter (DMM), a 250 MHz Counter, and a SurePath™ Master in a single C-Size VXIbus slot. The product design maximizes functionality while minimizing cost and component real estate.

The VX4101 has a single logical address with multiple instruments at that address. This simplifies the hardware and firmware in the VX4101, minimizing RAM and bus interface hardware and allowing a single parser to be used for all instruments. This reduces the cost and allows some unique features to be added to the VX4101.

In order to differentiate a command that is used by multiple instruments, such as the MEAS:VOLT:DC? command, the VX4101 incorporates the SCPI Instrument Subsystem. This subsystem permits the MEASure command to be directed to the intended instrument. An example of how the VX4101 uses the MEASure command is described in detail later in this section.

The VX4101 is a SCPI compliant instrument and by extention, IEEE-488.2 compliant. IEEE-488.2 is based on a model that requires all instrument activity to be sequential. Because of its unique architecture, the VX4101 permits the user to operate all instrument functions on the VX4101 simultaneously with non-sequential responses to queries. This permits test times to be decreased and lowers system costs.

This feature requires that certain IEEE-488.2 requirements be disabled. At reset, the VX4101 is fully IEEE-488.2 compliant. The user may turn off certain IEEE-488.2 behaviors, allowing the various instruments to be operated independently. This protocol is called the Asynchronous Protocol and is described in detail later in this section.

The Instrument Subsystem

Command and queries in the VX4101 fall into the following two classes:

- Global
- Instrument

A global command or query has the same behavior regardless of which instrument is active. Examples of global class commands are *RST, ABORt, *IDN?, and INSTrument:SELect.

Instrument class commands have different behavior depending upon which instrument is currently active. An example of an instrument class query is

MEASure:VOLT:DC?. This query is invalid for SurePath™ and valid for the DMM and Counter; the DMM and Counter each use different methods and inputs to make a DC measurement.

The command set for the VX4101 SCPI Instrument Subsystem is shown below. These commands allow the user to query the VX4101 as to the exact configuration of instruments and to select to which instrument is currently active. Complete descriptions for these commands appear in the *Syntax and Commands* section.

***NOTE**. It is important to remember that while only one instrument can be active at a time, multiple instruments can be operating simultaneously. Active simply indicates which instrument is currently receiving all instrument class commands and queries. Another instrument which was previously selected and commanded, may be in the process of making a measurement while the currently active instrument is being set up.*

SCPI Instrument Subsystem

The command set for the VX4101 instrument subsystem is as follows:

```
INSTrument
    :CATalog?
        :FULL?
        :LONG? <instrument number>
    :COUNt?
    :SELect <instrument name>
    :NSELect <instrument number>
```

The Asynchronous Protocol

The Asynchronous Protocol allows a multi-instrument card to exceed the limitations of 488.2 while still supporting customers who require 488.2 compatibility. It requires some additions to standard SCPI. These additions are modeled after the *:SYSTem:ERRor* subsystem as defined by *SCPI 1995***.**

IEEE 488.2

The IEEE 488.2 standard requires that a query be completed prior to the receipt of any new commands or queries (as defined in Chapter 6, *Message Exchange Control Protocol*, of *IEEE 488.2 Standard*). A query is completed after the controller has read the response. On a single instrument card, this does not pose a significant problem, since typically a new operation can't begin until the previous operation has completed. On a card with multiple instruments, such as the VX4101, this restriction can cause serious performance penalties. A query to the DMM to measure 1024 voltages might take a long time to return. The other instruments are unavailable while this occurs. According to the 488.2 Message Exchange Protocol Enforcer (MEPE), commands and queries to the Counter cannot be sent while the DMM query is in progress.

Another feature of 488.2 which does not work well in a multiple instrument environment is the concept of "*addressed with nothing to say*". According to the 488.2 MEPE, if a VXI read occurs and an instrument has nothing in its buffer, the VXIbus card should ignore the read, forcing a bus time-out. During the bus time-out period, the bus is locked-out from any other communication with any module in the mainframe.

When a compound query (multiple queries separated by semicolons and terminated by one Program Message Terminator or PMT) is sent to a 488.2 instrument, the instrument processes each query in a serial fashion and generates a single compound response composed of the in-order individual responses separated by semicolons. The basis for the IEEE 488.2 requirement is the need to unambiguously associate a response with the query that generated the response.

Asynchronous Protocol

The purpose of the Asynchronous Protocol is to allow the user to select either strict 488.2 enforcement or asynchronous querying and commanding of instruments. When the asynchronous protocol is enabled, a response queue in the instrument will queue query responses until they are requested by the user. When you request data from this queue via a VXI read, you receive responses in FIFO order. Since these responses can be returned in a different order than the queries for which they were requested, the VX4101 uses a different method of associating a response with the associated query. This is described in a later section entitled *Response Formats***.**

In IEEE 488.2, one of the tasks of the MEPE is to protect against deadlock. The Asynchronous Protocol prevents deadlock by ensuring that the response queue never blocks the parser. There are two steps in the process. First, one entry in the queue is reserved for the overflow message. Second, if the queue becomes full, any subsequent writes to the queue will result in a loss of the message. If a message has been lost in this fashion, then the next VXIbus read will result in transmission of the buffer overflow error message. Data which is already in the response queue will remain there until it is read.

The 488.2 synchronization commands, *WAI, *OPC, and *OPC? are handled as global synchronization devices by the Asynchronous Protocol. This allows the user to explicitly control the order of commands to multiple instruments on a single device.

The SCPI commands which implement the protocol are as listed below. Descriptions of the commands, the protocol, and examples follow.

***NOTE**. Although it is possible to switch back and forth between strict 488.2 enforcement and the Asynchronous Protocol during instrument operation, Tektronix recommends that the user choose one mode or the other and leave the instrument in that mode until it is reset. Switching back and forth during operation could cause the instrument to enter an undesirable state under some circumstances.*

```
SYSTem
    :LANGuage    "SYNChronous" | "ASYNchronous"
    :RQUeue
        :SNUMber
            [:SET]<seq #>
            :STATe<boolean>
        :QMODe NEXT | ALL
```

Asynchronous Protocol Command Suite Description

This section lists and describes the SCPI command set used in the asynchronous protocol.

Asynchronous Protocol Command Set

`SYSTem:LANGuage [["SYNChronous"] | "ASYNchronous"]`

This command allows you to choose between strict 488.2 protocol enforcement and the Asynchronous Protocol. The power on and default states of this feature will be *SYNChronous***,** which implies strict 488.2 enforcement. An argument of *ASYNchronous* will enable the Asynchronous Protocol. Both query and command are supported.

If the currently selected language is *SYNChronous*, any attempt to use an Asynchronous Protocol command or query will generate the following error message:

-210,“Settings conflict;Execute <:SYST:LANG ASYN> first”

`SYSTem:RQUeue:SNUMber:STATe <boolean>"ON" | "OFF"`

This command turns the <*sequence #*> style of tags on or off. See the section *Response Formats* for detailed information. Both query and command are supported.

`SYSTem:RQUeue:SNUMber[:SET] <seq #>`

This command sets the next <*sequence #*> to be used. See the section *Response Formats* for detailed information. Both query and command are supported.

```
SYSTem:RQUeue:QMODe [[ "NEXT "] | "ALL"]
```

This command allows the user to select the instruments query mode while using the Asynchronous Protocol. Both command and query are supported.

The default is the *NEXT* mode. In this mode, a VXIbus read is interpreted as having been preceded by an implicit *SYSTem:RQUeue[:NEXT]?* query requesting the next entry from the queue. For every VXIbus read supplied, one response will be returned from the response queue.

In the *ALL* mode, a VXIbus read is interpreted as having been preceded by an implicit *SYSTem:RQUeue:ALL?* query requesting all entries currently residing in the response queue. For every VXIbus read supplied, all entries in the response queue will be returned in FIFO order, semi-colon separated, as one message. The response queue will be emptied as described under the *SYSTem:RQUeue:ALL?* query.

Response Formats

When using the Asynchronous Protocol, query responses can be returned out of order with respect to the queries which generated the responses. For this reason, each response must be tagged with a unique identifier which associates the response to a particular query. The general form of a query response in the asynchronous mode is as follows:

```
<tag>,<response>
```

The <*response*> field is identical to the query response returned by the instrument when strict IEEE 488.2 enforcement is enabled. The <*tag*> field has two user selectable formats. In the first, which is the default, the tag field is defined as follows:

```
"<instrument name>:<canonical query string>"
```

The instrument name is the same name which would be used by the command *INSTrument:SELect* <*instrument name*>. For queries which are of a global nature and not associated with a specific instrument, the instrument mnemonic used is VX4101. The canonical query string is based upon the original query which generated this response. The canonical form is the query string expanded to include all default nodes, with each node represented in its short form in upper case ASCII.

For example, if the MEAS:FREQ? query had been issued to the Counter, the response from the queue using this format would be:

```
"Counter:MEAS:FREQ?", 1.00000000000000E+07
```

The Asynchronous Protocol also addresses the 488.2 concept of *addressed with nothing to say*. When the VX4101 is using the Asynchronous Protocol and the response queue is empty, a VXI word serial read will cause the VX4101 to return a default message. This ensures that a query will not tie up the VXI backplane.

For queries taking a significant amount of time to return, it is possible to use a software function, such as a service request generated by activity in the SCPI Status registers, rather than polling for the response. This determines when a response is available.

When there is no data in the response queue, the following response will be generated in this mode:

```
"VX4101:RQU?","EMPTY"
```

When there is an overflow in the response queue, the following response will be generated in this mode:

```
"VX4101:RQU?","OVERFLOW"
```

The second format, which is selected with *SYSTem:SNUMber:STATe ON*, is defined as the following:

```
<instrument number>:<sequence number>
```

The instrument number is the same number which would be used by the command *INSTrument:NSELect <instrument number>*. For queries which are of a global nature and not associated with a specific instrument, the instrument number used is 0. The sequence number is a one-up number assigned to a query as it is received by the card. As a query is received, it is assigned the current number. This number is then incremented such that the next query received will have a sequence number one larger than the last. Note that sequence numbers are global to the card and have no direct correlation with the instrument number.

The sequence number defaults to zero on power-on. To set the sequence number to a desired value, use the following command: *SYSTem:RQUeue:SNUMber[SET]<sequence number>*. After the instrument receives this command, the next received query will be assigned the *<sequence number>*specified in the command. Each subsequent sequence number will increment by 1 each time a query is received.

The query *SYSTem:RQUeue:SNUMber[:SET]?* can be used to determine what the next sequence number will be. Since the response to this query will contain a sequence number itself (assuming numerical tags are enabled), the response to this query will be the sequence number which will be used by the next query received.

For example, if the message is the sixth message in the queue and the Counter identification number is 3, the response for the example above would be:

```
3:6,1.00000000000000E+07
```

The purpose of two different formats is to support different user requirements. Using the longer ASCII format allows a user typing commands into a talker/listener to get easily readable feedback as to the source of the current response. The more terse numerical format is easier to parse in automated test software.

When there is no data in the response queue, the following response will be generated in this mode:

```
0:-1,"EMPTY"
```

When the response queue has overflowed, the following response will be generated in this mode:

```
0:-2,"OVERFLOW"
```

Asychronous Protocol Example

The following examples illustrate the use of the Asynchronous Protocol. These examples also show how instruments are selected as active prior to receiving commands.

Prerequisites. In order to get the responses shown in the examples, the instrument must have previously performed a word serial read.

NOTE*. In the strict 488.2 example, sending the word serial read prematurely could cause a bus timeout. In the Asynchronous Protocol examples, performing a word serial read would simply cause the default empty message to be returned.*

Example 1. Example 1 illustrates the disadvantage of strict 488.2 enforcement. After the measure query has been sent to the VX4101 DMM, no other commands or queries can be sent to the card until the query returns.

Command	Response
*RST	
SYST:ERR?	0,“No error”
*IDN?	Tektronix, VX4101, B000001, Firmware v.1.0.0/SCPI:95.0
INST:CAT?	VX4101,SurePath,DMM,Counter
INST:SEL DMM	
INST:SEL?	DMM
MEAS:ARR:VOLT:DC? 10,MIN,MAX	
<Wait for 35 seconds>	

Command	Response
<READ>	#3130+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01, +1.00000E-01,+1.00000E-01,+1.00000E-01
INST:SEL COUNTER	
INST:SEL?	Counter
MEAS:FREQ?	1.00000000000000E+07

Examples 2, 3 and 4. Examples 2,3, and 4 show how the Asynchronous Protocol allows multiple queries to be pending simultaneously. Example 2 uses the long ASCII tags. Example 3 uses the short numerical tag and the default starting sequence number of 0. Example 4 uses the short numerical tag and resets the sequence number to 100.

***NOTE**. In examples 3 and 4, the sequence numbers are returned "out of order" with regard to the order in which the responses are returned, but "in order" with regard to the order the queries were received.*

Command	Response
*RST	
SYST:LANG ASYN	
SYST:ERR?	"VX4101:SYST:ERR?",0,"No error"
*IDN?	"VX4101:*IDN?",Tektronix, VX4101, B000001, Firmware v.1.0.0/SCPI:95.0
INST:CAT?	"VX4101:INST:CAT?",VX4101,SurePath,DMM,Counter
INST:SEL DMM	
INST:SEL?	"VX4101:INST:SEL?",DMM
MEAS:ARR:VOLT:DC? 10,MIN,MAX	"VX4101:RQU?","EMPTY"
INST:SEL COUNTER	
INST:SEL?	"VX4101:INST:SEL?",Counter
MEAS:FREQ?	"Counter:MEAS:SCAL:FREQ?",1.00000000000000E+07
<READ>	"VX4101:RQU?","EMPTY"
<Wait for 35 seconds>	

Command	Response
<READ>	“DMM:MEAS:ARR:VOLT:DC?”, #3130+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01, +1.00000E-01,+1.00000E-01,+1.00000E-01
<READ>	“VX4101:RQU?”,“EMPTY”

Command	Response
*RST	
SYST:LANG ASYN	
SYST:RQU:SNUM:STAT ON	
SYST:ERR?	0:0,0,“No error”
*IDN?	0:1,Tektronix, VX4101, B0000001, Firmware v.1.0.0/SCPI:95.0
INST:CAT?	0:2,VX4101,SurePath,DMM,Counter
INST:SEL DMM	
INST:SEL?	0:3,DMM
MEAS:ARR:VOLT:DC? 10,MIN,MAX	0:-1,“EMPTY”
INST:SEL COUNTER	
<READ>	0:-1,“EMPTY”
INST:SEL?	0:5,Counter
MEAS:FREQ?	3:6,1.00000000000000E+07
<READ>	0:-1,“EMPTY”
<Wait for 35 seconds>	
<READ>	2:4, #3130+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01, +1.00000E-01,+1.00000E-01,+1.00000E-01
<READ>	0:-1,“EMPTY”

Command	Response
*RST	
SYST:LANG ASYN	
SYST:RQU:SNUM:STAT ON	
SYST:RQU:SNUM 100	

Command	Response
SYST:ERR?	0:100,0,“No error”
*IDN?	0:101,Tektronix, VX4101, B0000001, Firmware 1.0.0/ SCPI:95.0
INST:CAT?	0:102,VX4101,SurePath,DMM,Counter
INST:SEL DMM	
INST:SEL?	0:103,DMM
MEAS:ARR:VOLT:DC? 10,MIN,MAX	0:-1,“EMPTY”
INST:SEL COUNTER	
INST:SEL?	0:105,Counter
MEAS:FREQ?	3:106, 1.00000000000000E+07
<READ>	0:-1,“EMPTY”
<Wait for 35 seconds>	
<READ>	2:104,#3130+1.00000E-01,+1.00000E-01,+1.00000E-01,+1.00 000E-01,+1.00000E-01,+1.00000E-01,+1.00000E-01, +1.00000E-01,+1.00000E-01,+1.00000E-01
<READ>	0:-1,“EMPTY”

Trigger Commands

For DMMs and scanners, it is common to think of the devices as triggering to take a measurement or to switch to the next scan line. For counters, on the other hand, it is more common to think of them as arming to take a measurement. For the purposes of the following discussions, triggering and arming can be considered synonymous. Any differences will be noted.

One difference is that in addition to the STARt arming source, the Counter implements a STOP arming source. Counting is enabled when the start source is asserted and disabled when the stop source is asserted.

For the Surepath™ master, triggers are implemented in firmware, while for the DMM and Counter, the triggers are implemented in hardware. For this reason, the resolution and accuracy of the Surepath™ timers is less than those of the DMM and Counter triggers. Since the Surepath™ trigger times are of the same order as typical settling delays, impact to instrument performance is negligible.

The following is a description of the VX4101 trigger sources:

Table 2–2: VX4101 Trigger Sources

Trigger	Source
HOLD	Instrument trigger source is disconnected from current source (except IMMediate). This trigger is fixed.
IMMediate	Device enter the triggered state immediately after receiving the INITiate command. This trigger is fixed.
BUS	Trigger source is either a word serial trigger command or the 488.2 common command *TRG. This trigger is delayable.
TTLTrigger<0-7>	One of the VXIbus TTL trigger lines. This trigger is delayable.
COMMand<0-4>	One of five software command triggers. TRIG:FIR<N> is used to send these triggers. This trigger is delayable.
TIMer	Trigger source is the programmable periodic timer. This trigger is fixed.
SUREPATH	Trigger is generated whenever a scan line has switched and settled. This trigger is delayable.
DMM	Trigger is generated whenever the DMM has completed a measurement. This trigger is delayable.
COUNTER	Trigger is generated whenever the Counter has completed a measurement. This trigger is delayable.
CTR_EXTARM	This trigger is the digital representation of the analog signal input into the Counter front panel arm signal. This trigger is delayable.

Figure 2-1 shows the trigger architecture of the VX4101. The VX4101 has two types of trigger sources, fixed and delayable. A delayable trigger source supports three trigger modes; pass-through, delay by time, and delay by trigger. Fixed trigger sources are always in pass-through mode. Most VX4101 triggers can be routed to any of the VXIbus TTL trigger lines.

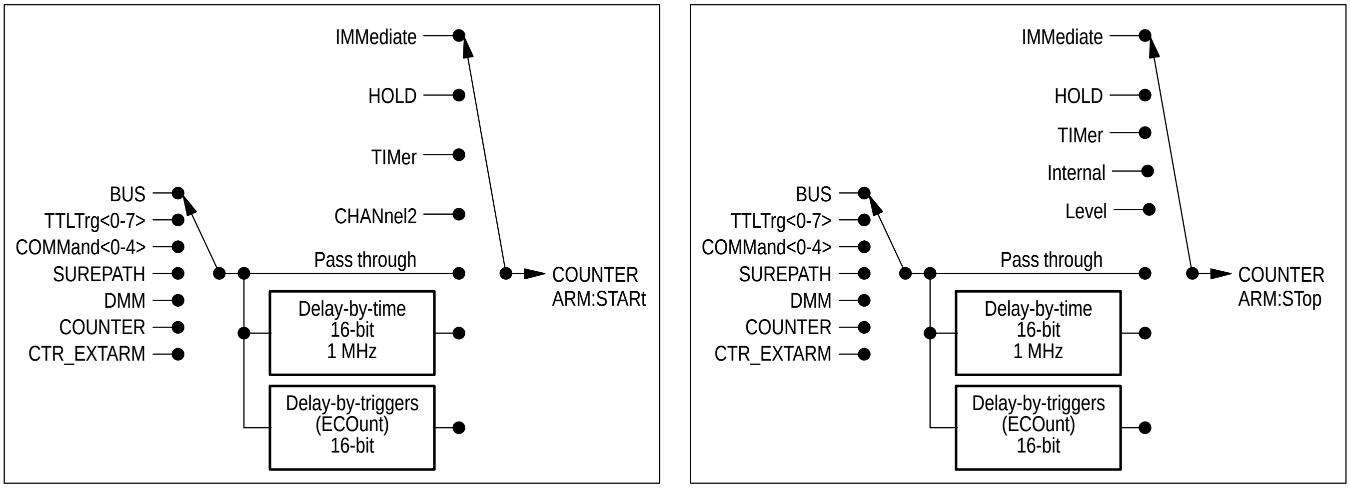
In addition to the above trigger sources, the Counter has the following three additional sources.

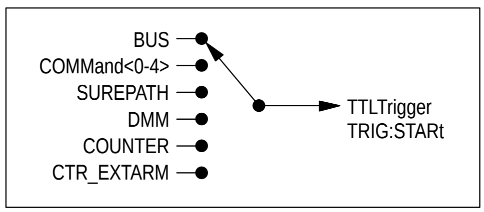
CTR_CHAN2. This is the signal on channel 2 of the Counter. It acts as a high speed gate and can start and stop a measurement on channel 1 of the Counter. This source is fixed and can only be used as an ARM:STARt source on channel 1 of the Counter.

INTernal. This is a fixed source and is only valid as an ARM:STOP source and specifies that other internal settings on the Counter (either the aperture or delay-by-events Counter) will be used to stop the Counter.

LEVel. This is a fixed source and is only valid as an ARM:STOP source. It is specifies that the ARM:STARt source is level-sensitive instead of edge-sensitive. The Counter will remain armed as long as the start source is asserted.

***NOTE**. The Counter has trigger modes and sources which the other two instruments do not have.*





NOTE: *If Channel 2 is selected as the arm start, then level is automatically selected as the Arm stop.*

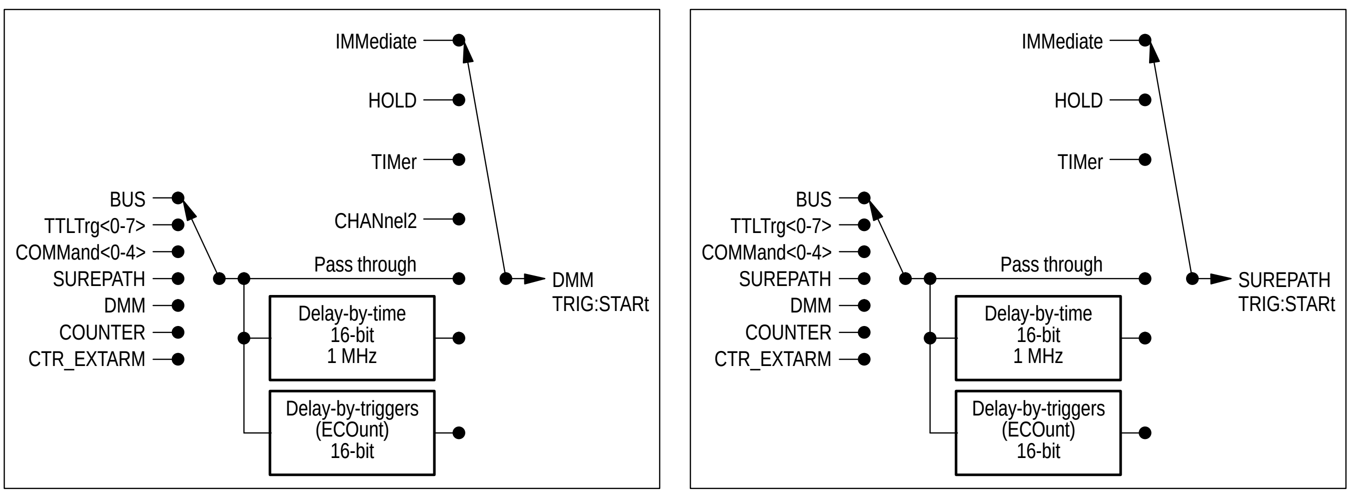


Figure 2–1: VX4101 Trigger Architecture

Syntax and Commands

Command Syntax

Command protocol and syntax for the VX4101 Module are as follows:

- A command string consists of a string of ASCII-encoded characters terminated by a <program message terminator>. The <program message terminator> is optional white space, followed by any one of the following command terminations:

  A line feed <LF> or new line <NL> character (hexadecimal 0A, decimal 10)

  The END bit set

  The END bit with a line feed <LF> or new line <NL>

  The command string is buffered until the terminator is encountered, at which time the entire string is processed.

- In addition to terminating a command, the semi-colon character directs the SCPI command parser to interpret the next command with the assumption that all characters up to and including the last colon in the previous command have just been parsed

- White space characters can be used to make a command string more readable. These characters are ASCII-encoded bytes in the range hexadecimal 00-09 and 0B-20 (decimal 0-9 and 11-32). This range includes the ASCII control characters and the space, but excludes the line feed <NL>. White space characters are ignored when legally encountered in a command string. White space is allowed anywhere in a command string, except for the following:

  Within a program mnemonic (for example ROUTE)

  Around a colon (:) mnemonic separator (for example ROUTE: CLOSE or ROUTE :CLOSE)

  Between a mnemonic and a (?) (for example CLOSE ?)

  Following an asterisk (*) (for example * STB?)

  Within a number (for example 12 34)

  At least one white space character is required between a command/query header and its associated arguments. For example in the command

  ```
  route:configure:join m1,(1:6)
  ```

the command header is the string “route:configure:join”. The arguments associated with this command are the module name “m1” and the section list “(1:6)”. At least one white space character must be sent before the first argument.

In the query

```
route:close? (@m1(1:64))
```

the query header is the string “route:close?”. The argument associated with this query is the channel list “(@m1(1:64))”. At least one white space character must be sent before the channel list argument.

- A SCPI command or query is composed of one or more keywords separated by colons. A keyword can be sent in either short or long form. The short form of a keyword is composed of capital letters in the command descriptions. The long form is composed of all characters in the keyword. The keywords in a command or query can be a combination of long and short forms. Commands and queries are parsed in a case-independent manner
- Multiple data parameters passed by a command are separated by a comma (,)
- A question mark (?) following a command indicates that a response will be returned. All responses from the VX4101 are terminated with a line feed <LF> (hexadecimal 0A) character
- In the command descriptions, the following special characters are used. Except for the colon (:), these characters are not part of the command and should not be sent. If an optional field is omitted, the default for the command is applied

[ ]	Brackets indicate an optional field
|	A bar indicates a logical OR choice
:	A colon is used to separate command fields
< >	Field indicator

Syntax Example The following is a command for the DMM:

```
CONFigure[:SCALar][:VOLTage]:DC [<expected value>
[,<resolution>]]
```

Each of the following commands is a valid form of this command. This is only a partial list of commands and is intended only for illustrative purposes:

```
CONFIGURE:SCALAR:VOLTAGE:DC 5.0,.001
```

```
conf:DC
```

```
CONF:scal:DC 5.0

Conf:Dc MAX,min

confIGURE:scalAR:voltAGE:DC 21

conf:DC 10,maximum
```

***NOTE**. Examples throughout this manual use various forms of the command syntax to further illustrate these concepts.*

SCPI/IEEE 488.2 Command Elements

The definition of elements used in SCPI/IEEE 488.2 commands and command descriptions is as follows:

```
<NR1>
```

ASCII integer representation of a decimal number.

```
<NRf>
```

ASCII integer, fixed point or floating point representation of a decimal number.

```
Error/Event queue
```

When the command parser detects a syntax error or data range error, it places an error message describing the error in the Error/Event queue. Bit 2 of the Status Byte Register is set to indicate that this queue is not empty. Bit 5 of the Standard Event Status Register (the Command Error bit) is set if the parser detects a syntax error. Bit 4 of the Standard Event Status register (the Execution Error bit) is set if the parser detects a numeric argument that is out of range. When a SYSTem:ERRor? query is received, an error message is moved from the Error/Event queue and placed in the Output queue.

IEEE-488.2 Common Commands

*CLS

Command Syntax *CLS

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Clears all event status registers and queues.

Examples See *Status & Events* section for examples

*ESE

Command Syntax *ESE <nrfDTO>

Query Syntax `*ESE?`

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits 0–255

Related Commands N/A

Description Sets or queries the contents of the IEEE 488.2 Standard Event Status Enable Register. The contents of this register are unaffected by a register read.

Examples See *Status & Events* section for examples.

*ESR?

Command Syntax N/A

Query Syntax *ESR?

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Queries the contents of the IEEE 488.2 Standard Event Status Register. The contents of this register are cleared after the read is complete.

Examples See *Status & Events* section for examples

*IDN?

Command Syntax N/A

Query Syntax `*IDN?`

Command Class Global

Query Response Tektronix, VX4101, <serial #>, Firmware v.<version>/SCPI:95.0

***RST Value** N/A

Limits N/A

Related Commands
`SYSTem:VERSion?`
`SYSTem:SNUMber`
`SYSTem:OPTions?`

Description Returns a unique instrument identification string which includes board serial number and firmware revision level.

Examples

Command	Response
*IDN?	Tektronix, VX4101, B000001, Firmware v.1.0.0/SCPI:95.0

*OPC

Command Syntax `*OPC`

Query Syntax `*OPC?`

Command Class Global

Query Response 1

***RST Value** N/A

Limits N/A

Related Commands *WAI

Description This command delays the VX4101 parser from processing any further commands until all commands currently in progress have completed. After all commands have completed, the *OPC command sets the operation complete bit in the IEEE 488.2 Standard Event Status Register. The *OPC? query places an ASCII character 1 in the Output queue. Note that a command is considered complete as soon as its action has been initiated. See the description of *WAI for more on this concept.

Examples

Command	Response
INST:SEL SUREPATH; ROUT:CLOS (@1); *OPC	
ESR?	0
ESR?	1
INST:SEL DMM; MEAS:VOLT:DC?	+2.39340E-1
INST:SEL SUREPATH; ROUT:CLOS (@1); *OPC?	1
INST:SEL DMM; MEAS:VOLT:DC?	+2.39340E-1

*RST

Command Syntax `*RST`

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** The following are reset values for the components in the VX4101 DMM/Counter:

Table 3–1: VX4101 Reset Values

Characteristic	Description
Communication Protocol	Synchronous (IEEE 488.2)
Reference Oscillator Source	Internal 10 MHz
TTL Trigger Sources	HOLD
Periodic Trigger Period	0 (Off)
Query Timeout	Off

Table 3–2: Front Panel Arm Reset Values

Characteristic	Description
Threshold	1.4 V

Table 3–3: Counter Channels 1 & 2 Analog Front-End

Characteristic	Description
Coupling	AC
Impedance	1 MΩ
Attenuator	X1
Offset	0 V
Gain	1
Lowpass Filter	Off
Lowpass Filter Frequency	20 MHz
Comparator Slope	Positive
Comparator Level	0 V
Hysteresis	.06 V

Table 3–4: Counter Measurement Settings

Characteristic	Description
Function	SCALar:FREQuency
Channel	1
Start Trigger Source	IMMediate

Table 3–4: Counter Measurement Settings (Cont.)

Characteristic	Description
Stop Trigger Source	INTernal
Mode	Aperture
Events	1000
Auto Setup	On
Time Interval Delay	1.0e–6 Seconds

Table 3–5: DMM Calibration Settings

Characteristic	Description
Line Frequency	As Last Programmed (60 Hz Default)
Source	Internal
Auto-Zero	Off

Table 3–6: DMM Measurement Settings

Characteristic	Description
Function	SCALar:VOLT:DC
Range	MAX (300 V)
Trigger Source	Immediate
Stop Trigger Source	INTernal
Aperture	200e–3 seconds
Auto-Ranging	On
Input Impedance	10e9 Ω
50 Hz NPLC	10
60 Hz NPLC	12

Table 3–7: SurePath Settings

Characteristic	Description
Trigger Source	Immediate
Scan Rate	16e–3 seconds
Default Module Names	M1,M2,M3,...,Mn
Open/Close Dwell Time	0 seconds

Table 3–7: SurePath Settings (Cont.)

Characteristic	Description
Power Fail Action	Open
VX4320	All sections closed to first relay
VX4330	All sections disjoined, set to 4–wire MUX mode, all relays opened
VX4350	All relays opened
VX4380	All relays opened

Limits N/A

Related Commands INSTrument:RESet

Description This command sets all instruments to known states independent of past use. All instruments are deselected by this command. The Error/Event Queue, Event Status Registers, and Event Enable Registers are not affected by this command.

Examples

Command	Response
INST:SEL DMM	
CONF:FRES	
CONF?	":SCAL:FRES 3e+08,180000"
*RST	
INST:SEL DMM	
CONF?	":SCAL:VOLT:DC 300,0.001"

*SRE

Command Syntax *SRE

Query Syntax *SRE?

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits 0–255

Related Commands N/A

Description Sets or queries the contents of the IEEE 488.2 Service Request Enable Register. The contents of this register are unaffected by a register read.

Examples See *Status & Events* section for examples.

*STB?

Command Syntax N/A

Query Syntax *STB?

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Queries the contents of the IEEE 488.2 Status Byte Register. The contents of this register are cleared after the read is complete.

Examples See *Status & Events* Section for examples.

*TRG

Command Syntax *TRG

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands TRIGger:SOURce

Description This command is equivalent to a Group Execute Trigger command. Upon its receipt, places any instrument which has selected BUS as its trigger source in the Device Trigger Active State as defined by the IEEE 488.2 standard.

Examples

Command	Response
INST:SEL DMM	
CONF:VOLT:DC	
TRIG:SOUR BUS	
INIT	
FETC:COUN?	0
*TRG	
FETC:COUN?	1
FETC?	+2.39340E-1

*TST?

Command Syntax N/A

Query Syntax *TST?

Command Class Global

Query Response PASS, VX4101:Self Test Passed, DMM: Self Test passed, COUNTER: Self Test passed.

***NOTE**. PASS can also be FAIL, and Passed can be Failed, depending on the results of the test.*

***RST Value** N/A

Limits N/A

Related Commands TEST:ALL?

Description This query is used to perform a system wide self-test of the VX4101, Counter, and DMM. At the end of the self-test the overall pass/fail status, as well as the pass/fail status for each instrument is returned as a response to the query. If a test fails, a descriptive error message is placed in the error queue and the error LED begins to blink. At the end of the self-test, the VX4101 is returned to the reset condition.

***NOTE**. SurePath™ performs a self-test during each operation and so is not explicitly tested with the *TST command.*

Examples

Command	Response
*TST?	PASS, VX4101:Self Test Passed, DMM: Self Test passed, COUNTER: Self Test passed
INST:SEL:COUNTER	
CONF:FREQ	

Command	Response
CONF?	"1:SCAL:FREQ"
INST:RES	
INST:SEL?	COUNTER
CONF?	"1:SCAL:VOLT:DC"

*WAI

Command Syntax `*WAI`

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands *OPC
*OPC?

Description This command delays the VX4101 parser from processing any further commands until all commands currently in progress have completed. A command is considered complete as soon as its action has been initiated. The instrument may continue processing, collecting, or measuring after the command has been reported as complete. If, for instance, it is desired to wait for a measurement to complete, it is not sufficient to do a *WAI after sending an INIT command. Rather, one should set up and wait for a service request to be generated by a negative transition of the Measurement-In-Progress bit of the event register. In the following example, the DMM measurement is delayed until the relay is closed.

Examples

Command	Response
INST:SEL SUREPATH; ROUT:CLOS (@1); *WAI; INST:SEL DMM; MEAS:VOLT:DC?	+2.39340E-1

General VX4101 Commands

This section summarizes the SCPI commands that control overall functions of the VX4101, including:

- Querying different firmware versions
- Querying measurement options
- Selecting which instrument to use
- Selecting the IEEE 488.2 or the Asynchronous Mode Protocols

A summary listing of the commands in this section are as follows:

```
ABORt

CALibrate:ROSCillator:CLEar
CALibrate:ROSCillator:MANual

INSTrument:CATalog?
INSTrument:CATalog:FULL?
INSTrument:CATalog:LONG?
INSTrument:COUNt?
INSTrument:NSELect
INSTrument[:SELect]

OUTPut:TTLTrg[<N>]:SOURce:CATalog?

SOURce:ROSCillator[:SOURce]
SOURce:ROSCillator:VALue

SYSTem:ERRor?
SYSTem:LANGuage
SYSTem:OPTions?
SYSTem:RQUeue:QMODe
SYSTem:RQUeue:SNUMber[:SET]
SYSTem:RQUeue:SNUMber:STATe
SYSTem:SNUMber?
SYSTem:TIMeout
SYSTem:VERSion?

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:FIRe<0-4>
TRIGger ([:SEQuence1]|:STARt)[:LAYer]:TIMer
```

ABORt

Command Syntax `ABORt`

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands INSTrument:ABORt
INITiate
INITiate:CONTinuous

Description Places all instruments in the IDLE state, aborting any measurement or other instrument activity in progress. The instrument configuration is unchanged and a subsequent INIT command will cause the instrument to re-start the same type of measurement. Regardless of whether the instrument is in synchronous or asynchronous mode, this command can be sent while a query is in progress and any measurements will be aborted.

Examples

Command	Response
INST:SEL DMM	
CONF:ARR:VOLT:DC	
INIT	
FETC:COUN?	127
ABOR	
FETC:COUN?	153
FETC:COUN?	153

NOTE. *After abort, no more measurements are taken.*

CALibrate:ROSCillator:CLEar

Command Syntax `CALibrate:ROSCillator:CLEar`

Query Syntax N/A

Command Class `Global`

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands SOUR:ROSC[?]
SOUR:ROSC:VAL[?]
CAL:ROSC:MAN
CAL:ROSC

Description This command is used to negate the calibration for the currently selected reference oscillator. The reference oscillator is flagged as uncalibrated in nonvolatile memory and any time the current source is reselected, an error message will be generated noting this fact. An uncalibrated reference oscillator is assumed to be exactly 10 MHz. This command would typically be used when a reference oscillator source has been changed (such as changing a Slot 0 controller) and the instrument has not yet been calibrated with the new source.

Examples

Command	Response
SOUR:ROSC INT;SYST:ERR?	-313,“Calibration memory lost; VX4101; Reference Oscillator Uncalibrated”
SOUR:ROSC:VAL?	10000000.0000
INST:SEL COUNTER	
CAL:ROSC 10.001E6	
SOUR:ROSC:VAL?	9999999.0000
CAL:ROSC:CLE	
SOUR:ROSC:VAL?	10000000.000
SOUR:ROSC:VAL?	10000000.0000

CALibrate:ROSCillator:MANual

Command Syntax `CALibrate:ROSCillator:MANual <desired frequency>`

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** Calibration Value Stored in non-volatile memory

Limits 9 MHz $\leq$ Frequency $\leq$ 11 MHz

Related Commands SOUR:ROSC[?]
SOUR:ROSC:VAL[?]
CAL:ROSC:VAL[?]

Description This command allows the user to manually specify the calibrated frequency to be used for the currently selected reference oscillator. The specified frequency value is stored in non-volatile memory and overwrites the calibrated value stored previously. To temporarily override the calibrated value without storing in non-volatile memory, use the the SOUR:ROSC:VAL command.

Examples

Command	Response
SOUR:ROSC INT	
SYST:ERR	–300 “Device specific error; VX4101; Reference Oscillator Not Calibrated”
CAL”ROSC:MAN 9.999999E6	
SOUR:ROSC INT	
SOUR:ROSC:VAL?	9999999.0000
SYST:ERR	0, “No error”

INSTrument:CATalog?

Command Syntax N/A

Query Syntax `INSTrument:CATalog?`

Command Class Global

Query Response `VX4101, SurePath, DMM, Counter`

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This command lists the names of the individual instruments which comprise the VX4101. These names can be used to select the instrument using the INST:SEL command.

Examples

Command	Response
INST:CAT?	VX4101, SurePath, DMM, Counter

INSTrument:CATalog:FULL?

Command Syntax N/A

Query Syntax `INSTrument:CATalog:FULL?`

Command Class Global

Query Response `VX4101,0,SurePath,1,DMM,2,Counter,3`

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This command lists the names and numbers of the individual instruments which comprise the VX4101. The names can be used to select the instrument using the INST:SEL command. The numbers can be used to select the instrument using the INST:NSEL command.

Examples

Command	Response
INST:CAT:FULL?	VX4101,0,SurePath,1,DMM,2,Counter,3

INSTrument:CATalog:LONG?

Command Syntax N/A

Query Syntax `INSTrument:CATalog:LONG? [<instrument number>]`

Command Class Global

Query Response

```
VX4101,0,tksf4101.exe,"VX4101"
SurePath,1,tksfscan.exe,"Surepath Master"
DMM,2,tksfdmm.exe,"DMM"
Counter,3,tksfctr.exe,"Counter"
```

***RST Value** N/A

Limits 0–3

Related Commands N/A

Description This command is primarily for the use of the VXI*plug&play* soft front panel. It lists information on an individual instrument including the instrument name, number, soft front panel executable, and ASCII description. Specify an instrument number in the argument to return information about that instrument. Specify no argument to return information on all instruments. The information is returned separated by commas.

Examples

Command	Response
INST:CAT:LONG 2	DMM,2,tksfdmm.exe,“DMM”

INSTrument:COUNt?

Command Syntax N/A

Query Syntax `INSTrument:COUNt?`

Command Class Global

Query Response `4`

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This command returns the number of instruments which are listed when a INST:CAT? query is performed.

Examples

Command	Response
INST:CAT?	VX4101,SurePath,DMM,Counter
INST:COUN?	4

INSTrument:NSELect

Command Syntax `INSTrument:NSELect <instrument number>`

Query Syntax `INSTrument:NSELect?`

Command Class Global

Query Response `<current instrument number>`

***RST Value** 0

Limits 0–3

Related Commands INSTrument:SELect[?]

Description This command selects or queries the current instrument based upon the instrument number. The VX4101 is always instrument 0 and selecting this number effectively deselects all instruments.

Examples

Command	Response
*RST	
INST:NSEL?	0
INST:NSEL 2	
INST:NSEL?	2

INSTrument[:SELect]

Command Syntax `INSTrument[:SELect] <instrument name>`

Query Syntax `INSTrument[:SELect]?`

Command Class Global

Query Response `<current instrument name>`

***RST Value** VX4101

Limits VX4101, SUREPATH, DMM, COUNTER

Related Commands INSTrument:SELect[?]

Description This command selects or queries the current instrument based upon the instrument name. Selecting the VX4101 effectively deselects all other instruments.

Examples

Command	Response
*RST	
INST:SEL?	VX4101
INST:SEL COUNTER	
INST:SEL?	COUNTER

OUTPut:TTLTrg[<N>]:SOURce:CATalog?

Command Syntax N/A

Query Syntax `OUTPut:TTLTrg[<N>]:SOURce:CATalog?`

Command Class Global

Query Response `HOLD, BUS, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value** N/A

Limits 0–7

Related Commands OUTPut:TTLT[<N>]:SOURce

Description Lists available trigger sources for use with the OUTP:TTLT:SOUR command.

Examples

Command	Response
OUTP:TTLT:SOUR:CAT?	HOLD, BUS, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM

Command Syntax `OUTPut:TTLTrg[<N>]:SOURce <source>`

Query Syntax `OUTPut:TTLTrg[<N>]:SOURce?`

Command Class Global

Query Response `<current source>`

***RST Value** Hold

Limits HOLD,BUS,COMMAND0,COMMAND1,COMMAND2,COMMAND3, COMMAND4,TIMER,SUREPATH,DMM,COUNTER,CTR_EXTARM 0–7 are valid TTL Trigger Suffixes, default is 1

Related Commands OUTPut:TTLT:SOURce:CATalog?

Description Selects or queries the trigger source for the specified VXIbus TTL trigger.

Examples

Command	Response
OUTP:TTLT:SOUR:CAT?	HOLD, BUS, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM
OUTP:TTLT5:SOUR DMMA	
OUTP:TTLT5:SOUR?	DMMA

SOURce:ROSCillator[:SOURce]

Command Syntax `SOURce:ROSCillator[:SOURce] INTernal | CLOCk10 | USER10`

Query Syntax `SOURce:ROSCillator[:SOURce]?`

Command Class Global

Query Response `INTernal|CLOCk10|USER1|USER2|USER2|USER3|USER4|`
`USER5|USER6|USER7|USER8|USER9|USER10`

***RST Value** INTernal

Limits INTernal|CLOCk10|USER1|USER2|USER3|USER4|
USER5|USER6|USER7|USER8|USER9|USER10

Related Commands SOUR:ROSC:VAL

Description This command selects the source of the instruments 10 MHz reference oscillator. Valid sources are the VX4101 on-board 10 MHz crystal and the VXIbus backplane 10 MHz signal. Because the user may have more than one possible backplane source, the VX4101 allows calibration factors for up to 10 back reference oscillator sources (USER1 to USER10) to be stored in non-volatile memory. CLOCk10 is an alias for USER1. When the source has never been calibrated, then a warning is issued that the source is not calibrated. Sources that are not calibrated are assumed to be ideal (10 MHz).

***NOTE**. The Counter uses the 10 MHz signal to drive most of its own control circuitry. If the backplane 10 MHz signal is selected and no signal is present, any attempt to use the Counter will result in a communications failure with the VX4101. To recover from this state, cycle power on the VX4101.*

***NOTE**. The reference oscillator should only be changed when the instrument is quiescent. Switching the reference oscillator source during measurements will result in indeterminate instrument operation.*

Examples See SOURce:ROSCillator:VALue examples

SOURce:ROSCillator:VALue

Command Syntax `SOURce:ROSCillator:VALue<frequency to use>`

Query Syntax `SOURc:ROSCillator:VALue`

Command Class Global

Query Response `<current reference oscillator frequency being used>`

***RST Value** Calibrated internal source value

Limits DEFault or 9 MHz ≤ value ≤ 11 MHz

Related Commands SOUR:ROSC

Description This command provides a temporary override of the calibrated oscillator frequency with a frequency supplied by the user. This can be used to provide a calibrated value for a USER oscillator without running a calibration cycle or to simulate an ideal oscillator. This value is lost whenever the oscillator source is switched. Choosing DEFault as the value re-initializes the oscillator frequency to the calibrated value, if available. If the calibrated value is not available, 10 MHz is used.

Examples

Command	Response
*RST	
SOUR:ROSC?	INTernal
SOUR:ROSC:VAL?	9999998.0000
SOUR:ROSC CLOC	
SOUR:ROSC?	CLOCk10
SOUR:ROSC:VAL?	10000001.000
SOUR:ROSC USER2	
SOUR:ROSC?	USER2
SYST:ERR?	–300 “Device specific error; VX4101; Reference Oscillator Not Calibrated”
SOUR:ROSC:VAL?	10000000.0000

SYSTem:ERRor?

Command Syntax N/A

Query Syntax `SYSTem:ERRor?`

Command Class Global

Query Response 0,“No error“ or `<error #>`,“error string“

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This query returns in FIFO order any error messages which have been queued. Error messages are of the form *<error #>,error string*. If the queue is empty, this is reported as 0, *No Error*. Depending upon the type of error, the error string may indicate which instrument generated the error.

Examples

Command	Response
XXX	
SYST:ERR?	–113,"Undefined header; Command not found; XXX\0D"
SYST:ERR?	0,"No error"

SYSTem:LANGuage

Command Syntax `SYSTem:LANGuage SYNChronous | ASYNchronous`

Query Syntax `SYSTem:LANGuage?`

Command Class Global

Query Response `SYNC | ASYN`

***RST Value** SYNC

Limits SYNChronous | ASYNchronous

Related Commands SYSTem:RQUeue:SNUMber:STATe[?]
SYSTem:RQUeue:SNUMber[:SET][?]
SYSTem:RQUeue:QMODe[?]

Description This command is used to enable or disable the IEEE 488.2 Message Exchange Protocol Enforcer (MEPE). The MEPE is enabled when the language selected is

SYNChronous. The MEPE is disabled and Query Responses are tagged when the language selected is ASYNchronous. See the *Theory of Operations* in *Operating Basics* for a detailed description of this mechanism.

Examples

Command	Response
*RST	
SYST:LANG?	SYNC
SYST:LANG ASYN	
SYST:LANG?	ASYN

SYSTem:OPTions?

Command Syntax N/A

Query Syntax `SYSTem:OPTions?`

Command Class Global

Query Response `NOOPT` or `<currently available options>`

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This query returns the options for which the VX4101 is currently configured. Two options are available in the VX4101:

- 1C – Counter 500 MHz Frequency Option
- 2C – Counter Channel 3 Prescalar Option

Examples

Command	Response
SYST:OPT?	NOOPT

Command Syntax N/A

Query Syntax `SYSTem:OPTion:DESCription <option code>`

Command Class Global

Query Response `<string describing option code>`

***RST Value** N/A

Limits N/A

Related Commands "1C," "2C," "NOOPt"

Related Commands SYST:OPT?

Description This query provides a textual description of the option codes returned by the SYST:OPT? query. The argument must be delimited by quotes.

Examples

Command	Response
SYST:OPT?	1C
SYST:OPT:DESC? "!C"	Counter 500 MHz option

SYSTem:RQUeue:QMODe

Command Syntax `SYSTem:RQUeue:QMODe NEXT | ALL`

Query Syntax `SYSTem:RQUeue:QMODe?`

Command Class Global

Query Response `NEXT | ALL`

***RST Value** N/A (SYNChronous mode)

Limits NEXT | ALL

Related Commands SYSTem:LANGuage[?]
SYSTem:RQUeue:SNUMber:STATe[?]
SYSTem:RQUeue:SNUMber[:SET][?]

Description This command provides a fast way to retrieve all responses in the Response Queue. When the mode is NEXT, a word serial read will retrieve a single response (assuming one exists). If the mode is ALL, then a word serial response will retrieve all responses, separated by semicolons, currently in the response queue. See the *Theory of Operations* in *Operating Basics* for a detailed description of this mechanism.

Examples

Command	Response
SYST:LANG ASYN	
SYST:RQU:QMOD ALL	
*IDN?	
SYST:ERR?	“VX4101:*IDN?”, Tektronix, VX4101, B00000021, Firmware v.1.0.0/SCPI:95.0; “VX4101:SYST:ERR?”, 0, “No error”

SYSTem:RQUeue:SNUMber[:SET]

Command Syntax `SYSTem:RQUeue:SNUMber[:SET] <next sequence number>`

Query Syntax `SYSTem:RQUeue:SNUMber[:SET]?`

Command Class Global

Query Response `<next sequence number>`

***RST Value** N/A (SYNChronous mode)

Limits 0 to 2,147,483,647 for normal sequence numbers

–1 for empty queue
–2 for queue overflow

Related Commands SYSTem:LANGuage[?]
SYSTem:RQUeue:SNUMber:STATe[?]
SYSTem:RQUeue:QMODe[?]

Description When in ASYNchronous mode, this command is used to specify what the next sequence number should be. By default, the sequence numbers start from zero and increment by one after each query is received. Since the query for the current sequence number has a sequence number in its tag, the query itself is the sequence number which will be used to tag the next Query Response. See the *Theory of Operations* in *Operating Basics* for a detailed description of this mechanism.

Examples

Command	Response
SYST:LANG ASYN	
SYST:RQU:SNUM 100	
SYST:RQU:SNUM?	0:100,101

SYSTem:RQUeue:SNUMber:STATe

Command Syntax `SYSTem:RQUeue:SNUMber:STATe ON|OFF|1|0`

Query Syntax `SYSTem:RQUeue:SNUMber:STATe?`

Command Class Global

Query Response 0 or 1

***RST Value** N/A (SYNChronous mode)

Limits 0, 1, ON, OFF

Related Commands

SYSTem:LANGuage[?]
SYSTem:RQUeue:SNUMber[:SET][?]
SYSTem:RQUeue:QMODe[?]

Description

When in ASYNchronous mode, this command is used to specify whether Query Responses are tagged with ASCII labels or with the numeric instrument number and an incrementing sequence number. See the *Theory of Operations* section for a detailed description of this mechanism.

Examples

Command	Response
SYST:LANG ASYN	
SYST:RQU:SNUM:STAT?	“VX4101:SYST:RQU:SNUM:STAT?”,0
SYST:RQU:SNUM:STAT ON	
SYST:RQU:SNUM:STAT?	0:1,1

SYSTem:SNUMber?

Command Syntax

N/A

Query Syntax

`SYSTem:SNUMber?`

Command Class

Global

Query Response

`<VX4101 serial number>`

***RST Value**

N/A

Limits

N/A

Related Commands

N/A

Description

This query returns the serial number of the VX4101.

Examples

Command	Response
SYST:SNUM?	B0000001

SYSTem:TIMeout

Command Syntax `SYSTem:TIMeout <timeout in seconds>`

Query Syntax `SYSTem:TIMeout?`

Command Class Global

Query Response `<current timeout>`

***RST Value** 0 (infinite timeout)

Limits 0, disable, timeout
1–100,000 seconds

Related Commands N/A

Description Specifies the maximum amount of time a query is allowed to take. If the query has not completed within the specified time, the instrument is aborted and a response, TIMEOUT, is placed in the response queue. This command has no effect if the VX4101 is using the Asynchronous Protocol. For more information on Asynchronous Protocol, see *Theory of Operation* in *Operating Basics*.

Examples

Command	Response
SYST:TIM 10	
INST:SEL DMM	
CONF:ARR:VOLT:DC 512,MAX	
READ?	
<wait 10 seconds>	
	TIMEOUT

SYSTem:VERSion?

Command Syntax N/A

Query Syntax `SYSTem:VERSion?`

Command Class Global

Query Response `<major>.<minor>.<sub-minor>`

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Returns the firmware version of the instrument in <major>.<minor>.<sub-minor> form.

Examples

Command	Response
SYST:VERS?	1.0.0

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:FIRe<0-4>

Command Syntax TRIGger ([:SEQuence1] |:STARt)[:LAYer]:FIRe<0-4>

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits 0-4

Related Commands TRIGger:SOURce:CATalog?
TRIGger:SOURce

Description Generates one of five software triggers.

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM
INST:SEL DMM	
CONF:VOLT:DC	
TRIG:SOUR COMM0	
INIT	
FETC:COUN?	0
TRIG:FIR0	
FETC:COUN?	1
FETC?	-6.03720E+00

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:TIMer

Command Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:TIMer <period of timer in seconds>`

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:TIMer?`

Command Class Global

Query Response `<current timer period in seconds>`

***RST Value** Off

Limits 0. Off

Table 3–8: Trigger Resolution (in μs)

Lower Bound	Upper Bound	Resolution
≥2	<510	2
≥510	<1275	5
≥1275	<2040	8
≥2040	<6375	25
≥6375	<8160	32
≥8160	<12750	50
≥12750	<25500	100

Related Commands

TRIGger:SOURce:CATalog?
TRIGger:SOURce

Description

Sets or queries the value of the period trigger source. Note that this is a global class command and there is only one periodic timer for the VX4101. Any time TRIG:TIM is sent to the card, the period of the timer trigger will be changed for all instruments using it as a source.

Examples

Command	Response
TRIG:TIM 500E–6	
TRIG:TIM?	500E–6
INST:SEL DMM	
TRIG:TIM?	500E–6
TRIG:TIM 45E–3	
INST:SEL COUNTER	
TRIG:TIM?	45E–3
TRIG:TIM .001	
TRIG:TIM?	1E–3
INST:SEL DMM	
TRIG:TIM?	1E–3

SCPI Commands for the Digital Multimeter

This section lists the SCPI commands and queries in alphabetic order for the Digital Multimeter (DMM). The IEEE-488.2 Common Commands are listed in *IEEE-488.2 Common Commands*. A summary listing of the SCPI command set for the DMM is as follows:

```
CALCulate:AVERage?
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCOunt?
CALCulate:LIMit:LOWer[:DATA]
CALCulate:LIMit:REPort[:DATA]?
CALCulate:LIMit:UPPer[:DATA]

CALibrate:LFRequency
CALibrate:VALue
CALibrate:ZERO:AUTO

CONFigure[:SCALar|ARRay][:...]

FETCh?

INITiate[IMMediate|CONTinuous]

INPut:IMPedance

INSTrument:RESet

MEASure[:SCALar|ARRay][:...]?

READ?

SENSe[:...]

STATus:OPERation:CONDition?

TEST:ALL?

VXI[:SERVant]:FDC:SEL
```

Trigger Commands

In addition, this section includes a listing of the commands used for triggering measurements for the DMM. A summary of the trigger commands follow the instrument-specific SCPI commands.

CALCulate:AVERage?

Command Syntax N/A

Query Syntax `CALCulate:AVERage? [< count > [,< offset > [,< step_size >]]]`

Command Class Instrument

Query Response `< average_voltage >`

***RST Value** < DEF > (count), 1 (offset) , 1 (step_size)

Limits 1– 4 K (count), 1 – 4 K (offset), 1 – 4 K (step_size)

Related Commands N/A

Description This query averages a specified number of measurements in the memory buffer. The optional parameters are for averaging selected values in the memory buffer. The optional <count> parameter specifies the number of data points to calculate. If no <count> is specified, then the last number measurements taken will be the default. The optional <offset> parameter determines at which point in memory values will start to be averaged. The optional <step_size> parameter determines the number of measurements to skip before taking a value to be averaged in.

***NOTE**. In order to specify a <step_size>, the user must enter the <count> and the <offset> information.*

Examples

Command	Response
inst:sel dmm	selects the dmm
calc:lim:upper 5.0	sets upper voltage limit
calc:aver?	averages all the data that has been collected
calc:aver? 20,3,2	averages 20 measurement readings. Starts with the 3rd measurement. Averages every other data point

CALCulate:LIMit:ENVelope[:DATA]

Command Syntax `CALCulate:LIMit:ENVelope[:DATA] <threshold1>,<threshold2>`

Query Syntax `CALCulate:LIMit:ENVelope[:DATA]?`

Command Class Instrument

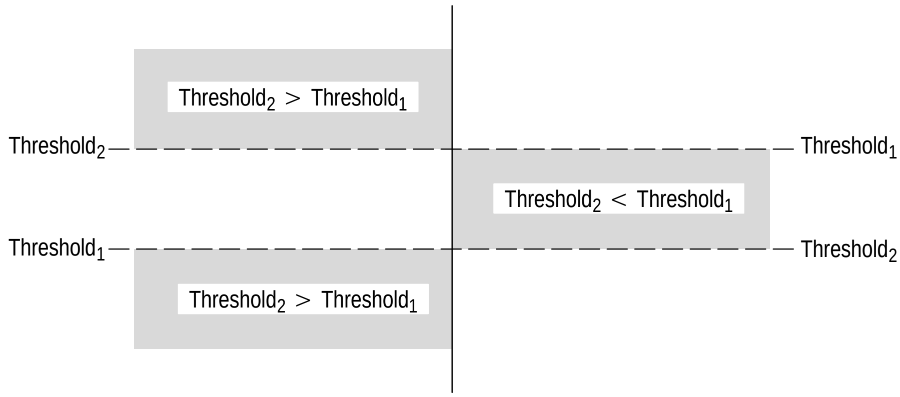
Query Response `<threshold1>,<threshold2>`

***RST Value** N/A

Limits –300 V to +300 V, 0 Ω to +300 MΩ

Related Commands
CALCulate:LIMit:ENVelope:POINt
CALCulate:LIMit:LOWer[:DATA]
CALCulate:LIMit:UPPer[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description This command searches for all the input data values above and below a set of thresholds or in between a set of thresholds in the measurement buffer. The range of the envelope is determined by the *<threshold1>* and the *<threshold2>*. If the *<threshold1>* is greater than the *<threshold2>*, the range of data values searched for is above the <threshold1> value or below the *<threshold2>* value. If the *<threshold1>* is less than *<threshold2>*, the range of data values searched for is between the *<threshold1>* and *<threshold2>*. The query for this command returns the boundary values.



Examples

Command	Response
inst:sel dmm	Selects the DMM
calc:lim:env 5.0,1.0	Sets an envelope voltage. All data points above 5.0 volts and below 1.0 volts are targeted
calc:lim:env?	1.0, 5.0<lf> returns the envelope threshold voltages
calc:lim:env 1.0,5.0	Sets an envelope voltage. All data points between5.0 volts and 1.0 volts are targeted
calc:lim:env?	5.0, 1.0<lf> returns the envelope threshold voltages

CALCulate:LIMit:FCOunt?

Command Syntax N/A

Query Syntax `CALCulate:LIMit:FCOunt?`

Command Class Instrument

Query Response `<numeric_value>`

***RST Value** N/A

Limits N/A

Related Commands CALCulate:LIMit:REPort[:DATA]
CALCulate:LIMit:LOWer:POINt

CALCulate:LIMit:UPPer[:DATA]
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description This query command performs a limit test on the current available data and returns the number of data points that failed a limit test.

***NOTE**. You must call a CALCulate:LIMit:Upper, lower, or envelope command before this query.*

Examples

Command	Response
inst:sel dmm	Selects the DMM
calc:lim:upp 5.0	Sets upper voltage threshold
calc:lim:fco?	4<lf> returns the number of data points outsideof a defined threshold

CALCulate:LIMit:LOWer[:DATA]

Command Syntax `CALCulate:LIMit:LOWer[:DATA] <threshold>`

Query Syntax `CALCulate:LIMit:LOWer[:DATA]?`

Command Class Instrument

Query Response `<threshold>`

***RST Value** N/A

Limits –300 V to +300 V, 0 Ω to 300 MΩ

Related Commands
CALCulate:LIMit:LOWer:POINt
CALCulate:LIMit:UPPer[:DATA]
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description

This command searches for all the input data values below a certain threshold in the measurement buffer. The query for this command returns the boundary value. The example below uses voltage.

***NOTE**. You must call a CALCulate:LIMit:Upper, lower, or envelope command before you can use this query.*

Examples

Command	Response
inst:sel dmm	Selects the DMM
calc:lim:lower 5.0	Sets lower voltage threshold
calc:lim:lower?	5.0<lf> returns the lower threshold voltage

CALCulate:LIMit:REPort[:DATA]?

Command Syntax N/A

Query Syntax `CALCulate:LIMit:REPort[:DATA]?`

Command Class Instrument

Query Response `<memory_index>,<failed_value>[{;<memory_index>,<failed_value>}]`

***RST Value** N/A

Limits N/A

Related Commands CALCulate:LIMit:REPort:POINts

Description

This query command returns the indices of the data values collected in the most recent CALCulate:LIMit:...[:DATA] command. Note:The CALCulate:LIMit:...[:DATA] command must be run before this command. If no data values were found a zero will be returned. (The first point in memory is “1”)

Examples

Command	Response
inst:sel dmm	Selects the DMM
calc:lim:upp 5.0	Sets upper voltage threshold
calc:lim:rep?	2, 6.2; 5,5.01<lf> returns the number of data points outside of a defined threshold

CALCulate:LIMit:UPPer[:DATA]

Command Syntax `CALCulate:LIMit:UPPer[:DATA] <threshold>`

Query Syntax `CALCulate:LIMit:UPPer[:DATA]?`

Command Class Instrument

Query Response `<threshold>`

***RST Value** N/A

Limits N/A

Related Commands
CALCulate:LIMit:UPPer:POINt
CALCulate:LIMit:LOWer[:DATA]
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description This command searches for all the input data values above a certain threshold in the measurement buffer. The query for this command returns the boundary value. The example below uses voltage.

Examples

Command	Response
inst:sel dmm	Selects the DMM
calc:lim:upper 5.0	Sets upper voltage threshold
calc:lim:upper?	5.0<lf> returns the upper threshold voltage

CALibrate:LFRequency

Command Syntax `CALibrate:LFRequency <Line Frequency>`

Query Syntax `CALibrate:LFRequency?`

Command Class Instrument

Query Response The line frequency rejection value.

***RST Value** Last programmed line frequency rejection value.

Limits

<Line Frequency>	50,60,400

Formats

<Line Frequency>	Numeric
Query Response	Numeric

Related Commands

MEASure?
CONFigure
SENSe
READ?

Description

The CALibrate:LFRequency command defines the power line frequency at which the DMM performs noise rejection. You can define the frequency as 50, 60, or 400 Hz. With 400 Hz, the DMM actually uses 50 Hz rejection, since 50 Hz is a subharmonic of 400 Hz.

The line frequency rejection value (60 Hz factory default) is stored in nonvolatile memory for recall after power-on. Issuing the CALibrate:LFRequency command automatically stores the value specified into nonvolatile memory.

***NOTE**. Because much of the setup of the card is dependent on the line frequency rejection value, the DMM function of the module returns to its power-on state when this command is issued.*

The CALibrate:LFRequency? query returns the rejection frequency value stored in nonvolatile memory.

Specifying a value other than 50, 60, or 400 Hz will cause the DMM to default to 60 Hz.

Command	Response
calibrate:lfrequency 60	Define the line rejection frequency to be 60 Hz
CAL:LFR?	60<LF>The programmed line rejection frequency
cal:lfr 400	Define the line rejection frequency to be 400 Hz
CALibrate:LFRequency?	400<LF>

CALibrate:VALue

Command Syntax `CALibrate:VALue <Calibration Input Value>`

Query Syntax NA

Command Class Instrument

Query Response NA

***RST Value** NA

Limits <Calibration Input Value> 0 to 300,000,000 (see description below)

Formats <Calibration Input Value> Numeric

Related Commands
CALibration:Source
CONFigure
MEASure?
READ?
INITiate
INP:IMP

Description

A calibration of the DMM is recommended once a year to maintain published specifications accuracy. For more information, see Calibration for the DMM in *Appendix D: Calibration*.

The CALibrate:VALue command is used to perform the calibration. The set of modes and ranges that must be calibrated are as follows:

DC (10 GΩ input impedance), 0.030, 0.300, 3.00 V ranges
DC (10 MΩ input impedance, all ranges)

AC/DC, all ranges
4-wire Ω, all ranges
2-wire Ω, 30 Ω range at null only
Current, all ranges

The AC mode uses the AC/DC calibration information and is not calibrated separately.

The higher 2-wire Ω ranges depend on the 4-wire Ω gain calibration and the 30 Ω 2-wire null calibration, and are not calibrated separately. If a CALibration:VALue command is issued in the AC mode or higher 2-wire resistance ranges, an "Invalid DMM Calibration Mode" error is generated.

All calibrated modes and ranges except the 2-wire 30 Ω range require both a null and a gain calibration. The 2 wire 30 Ω range requires only a null calibration. The null calibration must be performed prior to the gain calibration. The limits of <Calibration Value Input> depend on the mode and range. For the DC, resistance, and current mode null calibrations, the argument must be exactly 0.

The <Calibration Input Value> is the external calibration voltage, resistance or current applied. Appendix E provides a set of external calibration specification requirements, environmental requirements, and warm-up requirements necessary for a valid calibration.

The DMM must be programmed for the desired mode/range prior to the calibration. Use a CONFigure command with the expected value equal to the range value, followed by an INITiate:[IMMediate] command or a READ? query command.

CAUTION. *Do not send only the CONFigure command followed by a CALIbration:VALue command. You must include an INITiate or READ? command to place the DMM physically in the proper mode and range. Sufficient time must be allowed for both the calibration source to stabilize and the VX4101 to stabilize physically in the desired mode and range prior to performing the calibration.*

For the AC/DC "null" calibration and the gain calibration for all calibrated modes and ranges, the <Calibration Input Value> must be a positive value within the limit listed below. The limits shown are as a percentage of the full scale range value. For example, in the DC 30 V ranges where the gain percentage must be between 65% and 100%, the CALibration:VALue argument must be between 19.5 and 30.0. A value outside of these limits will result in an "Invalid DMM Calibration Input" error.

Table 3–9: Limits of Calibration Input

Mode/Range	Low Limit	High Limit
DC Gain, All Ranges	65%	100%
AC/DC Null, 30 mV Range	30%	40%
AC/DC Null, other Ranges	3.33%	25%
AC/DC Gain, All Ranges	65%	100%
4 wire Resistance, Gain, All Ranges	30%	100%
Current, Gain, All Ranges	65%	100%

The AC/DC “null” calibrations are provided at a nonzero value due to the inherent null noise in any AC measurement system. The “null” calibration limits shown permit a more linear calibration for the region of specified operation for each range (10% to 100% of range value).

In order for the CALibration:VALue command to be used, the DMM must be programmed for CALibration:SOURce EXTernal. This provides a measure of protection against an unintentional calibration or calibration by unqualified personnel. In addition to the required limits above, a calibration reasonableness check is also provided. If the difference between the <Calibration Input Values> specified and the value measured by the VX4101 DMM is outside the required design limits, “DMM Calibration out of Range” error is generated.

The time for the CALibration:VALue command to complete depends on the mode and range and typically takes 1 or 2 seconds for all modes (except AC which takes slightly more). In the DC, resistance and current modes, an auto zero base calibration is also performed with the null calibration. This additional calibration is used as a base reference for the CAL:ZERO:AUTO function.

Examples

Command	Response
Calibrate:source ext	Specifies an external calibration source
CAL:SOUR?	EXT<LF> The programmed calibration source
CONF:FRES 300e3	Configures the DMM for the 4-wire 300 k resistance range
CAL:SOUR?	EXT<LF> The programmed calibration source
READ?	Returns the present resistance value and physically places the DMM in the specified range
Cal:Val 0	Performs a null calibration in the 300 kΩ range. A 0 Ω resistor input is required
CAL:SOUR?	EXT<LF> The programmed calibration source
CONF:FRES 300e3	Configures the DMM for the 4-wire 300 k resistance range

Command	Response
READ?	Returns the present resistance value and physically places the DMM in the specified range
Cal:Val 0	Performs a null calibration in the 300 kΩ range. A 0 Ω resistor input is required
CALIBRATION:VALUE 100.03e3	Performs a gain calibration assuming a 100.03 kΩ resistor has been placed on the input
Cal:sour int	Returns the DMM to internal Calibration source permitting an autozero to be performed

CALibrate:ZERO:AUTO

Command Syntax `CALibrate:ZERO:AUTO <Boolean | ONCE >`

Query Syntax `CALibrate:ZERO:AUTO?`

Command Class Instrument

Query Response

0 (if auto-zero is disabled (OFF))

1 (if auto-zero is enabled (ON))

***RST Value** Off

Limits N/A

Formats <Boolean> ON | 1 | OFF | 0

Related Commands

MEASure?
READ?
INITiate

Description The calibration procedure of the DMM includes commands to calculate and store, in nonvolatile memory, the offset value of every null calibrated function and range on the card. After power-up, the instrument reads these values from the nonvolatile memory, and uses them for null drift compensation.

Because offset drifts with time and temperature, the CALibrate:AUTO:ZERO command rezeroes the card during operation.

If you program the CALibrate:ZERO:AUTO ONCE command, the card immediately performs an autozero operation for the function and range currently in effect. It will then compensate all subsequent measurements for that function by the calculated value.

***NOTE**. You must issue the CALibrate:SOURce INT command before autozero will work.*

If data sampling is in progress, issuing a CALibrate:ZERO:AUTO ONCE command will abort the acquisition.

If you program CALibrate:ZERO:AUTO ON, the card performs an autozero operation each time it receives a MEASure?, READ? or INITiate command *prior to the first measurement*, and whenever a measurement is taken which changes either the current function, or the current range. For example, if autoranging and autozeroing were enabled, issuing a MEASure? command would cause the card to first seek the appropriate range for the input, and once found, perform an autozero. If the range changes during the measurement process, the card performs an autozero operation automatically on the new range. The autozero routine uses the same aperture time as the measurement currently being taken. For example, if the current function were programmed to a 0.5 second aperture, the autozero routine will incur approximately 0.5 seconds of overhead each time it is invoked.

If CALibrate:ZERO:AUTO OFF is programmed, autozeroing is disabled.

Note that the autozero setup is valid for all functions and ranges of the card. That is, the autozeroing cannot be individually programmed for each function. Also the autozero values are not stored in nonvolatile memory. The calibration procedure of the DMM includes commands to calculate and store calibrated base offset values.

The CALibrate:ZERO:AUTO? query returns a zero or a one dependent on whether the autozeroing is disabled.

Examples

Command	Response
calibrate:zero:auto on	Enable the auto zero calibration for every measurement
CAL:ZERO:AUTO?	1<LF> Auto zero enabled
cal:zero:auto off	Disable the auto zero calibration for each measurement
cal:zero:auto?	0<LF> Auto zero disabled

Command	Response
Cal:Zero:Auto 1	1 and 0 can be substituted for ON and OFF respectively
Cal:zero:auto once	Perform an auto zero on the current function and range

CONFigure[:SCALar|ARRay][:...]

Command Syntax

```
CONFigure[:SCALar]

   [:VOLTage]
       :AC                    [<Expected Value>[,<Resolution>]]
       :ACDC                  [<Expected Value>[,<Resolution>]]
       :DC                    [<Expected Value>[,<Resolution>]]
   :CURRent[:DC]              [<Expected Value>[,<Resolution>]]
   :RESistance                [<Expected Value>[,<Resolution>]]
   :FRESistance               [<Expected Value>[,<Resolution>]]

CONFigure:ARRay

   [:VOLTage]
       :AC<Array Size>        [,<Expected Value>[,<Resolution>]]
       :ACDC<Array Size>      [,<Expected Value>[,<Resolution>]]
       :DC<Array Size>        [,<Expected Value>[,<Resolution>]]
   :CURRent[:DC]<Array Size>  [,<Expected Value>[,<Resolution>]]
   :RESistance<Array Size>    [,<Expected Value>[,<Resolution>]]
   :FRESistance<Array Size>   [,<Expected Value>[,<Resolution>]]
```

Query Syntax

```
CONFigure?

CONFigure
    :DC?
    :AC?
    :ACDC?
    :RESistance?
    :FRESistance?
```

Query Response

CONF?	Current DMM input configuration, range, and resolution.
CONF[....]?	Detailed configuration information on the function specified.

***RST Value**

Default Configuration:

```
300 VDC, 10 Mohm Input, 0.001 Resolution, Autoranging ON
```

```
All other configurations at their maximum input ranges,
Autoranging ON
```

(See the SENSe command defaults).

Limits

<Array Size>	1 to 4096
<Expected Value>	See the MEASure? command.
<Resolution>	See the [SENse] [...]:RESolution command

Related Commands

MEASure?
INPut
READ?
FETCh?
CALibrate:LFR
CALibrate:ZERO:AUTO

Description

The CONFigure commands define the input configuration for each of the six functions of the card. They are sublevel commands to the MEASure? commands, as they define the input configuration for a measurement, without taking a measurement. If desired, you can customize the configuration setups via the lower level SENSe commands.

```
CONFigure[:VOLTage]:DC
```

This command sets up DC voltage measurements.

```
CONFigure[:VOLTage]:AC
```

This command sets up AC coupled, AC TRMS voltage measurements.

```
CONFigure[:VOLTage]:ACDC
```

This command sets up DC coupled, AC TRMS voltage measurements.

```
CONFigure:CURRent[:DC]
```

This command sets up DC current measurements.

```
CONFigure:RESistance
```

This command sets up two-wire resistance measurements.

```
CONFigure:FRESistance
```

This command sets up four-wire resistance measurements.

***NOTE**. See the MEASure? command for detailed descriptions of the [<Expected Value>],[<Resolution>], and <Array Size> fields.*

The CONFigure? query returns the current measurement configuration information, with the actual rounded and/or calculated values used by the card.

The CONFigure[.....]? queries return detailed setup information for the requested function.

***NOTE**. If data sampling is in progress, it will be aborted on receipt of a CONFigure command to prevent ambiguous data interpretation. Typically the CONFigure (and SENSE) commands are used to predefine the setup of each of the six measurement functions.*

Examples

Command	Response
CONFigure:SCALar:VOLTage:DC 2	Set to DC Volts, 1 Measurement
CONF?	":SCAL:VOLT:DC 3,1e-05"<LF>
conf:dc 10	Set to DC Volts
conf?	":SCAL:VOLT:DC 30,0.0001"<LF>
conf:volt:ac 5,0.01	Set to AC Volts
configure?	":SCAL:VOLT:AC 30,0.00154919"<LF>
conf:curr:dc max,minimum	Set to DC Amps maximum range and minimum resolution
CONFIGURE?	":SCAL:CURR:DC 1,7.74597e-05"<LF>
conf:res def	Set to two-wire Resistance
conf?	":SCAL:RES 3e+08,180000"<LF>
conf:fresistance	Set to four-wire Resistance
conf?	":SCAL:FRES 3e+08,180000"<LF>
conf:ARRAY:volt:dc 10,50,0.01	Set to DC Volts, 10 Measurements
configure?	":ARR:VOLT:DC 10,300,0.0109545"<LF>
conf:dc?	":ARR:VOLT:DC 10,300,0.0109545 AutoRange OFF AutoZero OFF Aperture=0.00166667 RPSec=600 NPLC=0.1 Rdgs2Avg=2 LFR=60"<LF> *

* The meaning of the returned string is as follows:

Table 3–10: Meaning of Returned String

Returned	Meaning
AutoRange OFF	The state of the autorange enable (ON | OFF | ONCE)
AutoZero OFF	The state of the autozero enable (ON | OFF)
Aperture=0.00166667	The aperture setting
RPSec=600	The number of readings per second corresponding to the aperture
NPLC=0.1	The number of power line cycles corresponding to the aperture
Rdgs2Avg=20	Internal readings count (not user accessable)
LFR=60	The line frequency rejection value

FETCh?

Command Syntax `N/A (query only)`

Query Syntax `FETCh? [<Count>[,<Offset>[,<Step Size>]]]`

`FETCh:COUNt?`

Query Response

`FETCh?` The measurement data.

`FETCh:COUNt?` The number of measurements acquired.

***RST Value** N/A

Limits

<Count> The current <Array Size> in effect.

<Offset> The current <Array Size> in effect.

<Step Size> The current <Array Size> in effect.

Formats Numeric

Related Commands

MEASure?
CONFigure
SENSe
INPUt

INITiate
FETCh?
*STB?

Description

The FETCh? command is the lowest level of the measurement subsystem, and simply retrieves the measurement data acquired through higher level commands. The optional parameters selectively control which of the acquired value(s) the measurement subsystem returns. If the optional parameters are not specified, the command returns the amount of data specified by the <Array Size> field of the higher level commands. For example, if the subsystem acquires thirty measurements, issuing a "FETCh?" returns the thirty values. Issuing a "FETCh? 10" would return the first 10 values acquired. Issuing a "FETCh? 10,21" would return 10 data values, beginning at offset 21 of the array (offset 1 = the first data value). Issuing a "FETCh? 5,1,2" would return five data values, beginning with the first acquired value, and stepping two locations for each value returned (values one, three, five, seven, and nine). If the parameters specify an index outside the number of measurements acquired, an error will be generated. The FETCh commands are nondestructive of the acquired data. (a subsequent FETCh? will return the same data if a new measurement has not been INITiated).

The FETCh:COUNt? command returns the number of measurements acquired since the last INITiate command was received.

The measurement buffer will, however, be emptied if a CONFigure, SENSe, or INPut command is issued. An error is generated if a FETCh? query is requested when no valid data is available.

For multiple data measurements, the data is returned using SCPI block format.

The format is:

```
HEADERdata,data,...,data<LF>
```

where the HEADER field is

```
#xCOUNT
```

x = the size of the count field

COUNT = the number of bytes returned following the header field.

For example, the format of a "fetch? 2" response would be

```
#226+2.98999E+00,+2.99030E+00<LF>
```

The first numeral 2 following the # sign indicates that the COUNT size field is 2. The number 26 following it indicates there are 26 characters in the block following the header.

Examples

Command	Response
FETCh?	#239+1.23456E+00,+2.34567E+00,+3.45678E+00<LF>
FETCh? 1	+1.23456E+00<LF>
FETCh? 2,2	#226+2.34567E+00,+3.45678E+00<LF>
FETC? 2,1,2	#226+1.23456E+00,+3.45678E+00<LF>
Fetch:count?	3<LF>

INITiate[:IMMediate|CONTinuous]

Command Syntax

`INITiate[:IMMediate]`

`INITiate:CONTinuous [<Boolean>]`

Query Syntax

`INITiate:CONTinuous?`

Command Class

Instrument

Query Response

`0 | 1`

***RST Value**

INITiate:CONTinuous OFF

Limits

ON | 1 | OFF | 0

Formats

<Boolean>

Related Commands

MEASure?
CONFigure
SENSe
SENSe
FETCh?
TRIGger
ABORt

Description

The INITiate[:IMMediate] command causes the DMM to arm its trigger for measurement acquisition. If the trigger source is immediate, the card will internally trigger the acquisition. Otherwise, the card waits for the selected trigger source to occur before beginning the acquisition. The FETCh:COUNt?

query reads the number of measurements the card has taken since it was initiated. When the acquisition is complete, you can use the FETCh? to get the measured data.

The INITiate:CONTinuous [ON] command internally triggers the DMM to operate in a free-running mode, where it continuously acquires and updates a single measurement. The FETCh? command can be issued any time after this command to read the current value of the input. Note that the INITiate:CONTinuous command is independent of the trigger setup.

The INITiate:CONTinuous OFF command aborts any measurements currently in progress.

The INITiate:CONTinuous? query returns a zero or the number one indicating whether INITiate:CONTinuous is off or on.

***NOTE**. If data sampling is in progress, issuing an INITiate command aborts the acquisition and starts a new acquisition.*

Examples

Command	Response
INITiate:IMMediate	arm the triggers for an acquisition
init:continuous	put the card into free-running acquisitions

INPut:IMPedance

Command Syntax `INPut:IMPedance <Input Impedance>`

Query Syntax `INPut:IMPedance?`

Command Class Instrument

Query Response The programmed input impedance.

***RST Value** 10e6 Ω (300 VDC range)

Limits 10e6 Ω, 10e9 Ω

Related Commands

MEASure
CONFigure
SENSe

Formats

<Input Impedance> Numeric

Description

The INPut:IMPedance command defines the input impedance for DC voltage measurements on the 30 mV, 300 mV, and 3 V ranges. The impedance may be defined as either 10e6 or 10e9 Ω. Any value other than 10e6 or 10e9 is rounded up to the nearer of these values. The 30 V and 300 V ranges are fixed at a 10 MΩ input impedance. If <Input Impedance> is not specified, 10e6 is assumed. Specifying a value greater than 10e9 or less than zero generates an error.

***NOTE**. If Autoranging is active for DC Voltage measurements, the impedance value is automatically overwritten with 10.0e6 Ω when a DC Voltage measurement is taken.*

The INPut:IMPedance? query returns the currently defined impedance value.

Examples

Command	Response
Input:Impedance 10e9	Define the input impedance as 10e9 Ω
INP:IMP?	1.00E+10<LF>

INSTrument:RESet

Command Syntax `INSTrument :RESet`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This command resets the currently selected instrument without effecting other instruments. The instrument returns to its *RST state. The instrument remains selected.

Examples

Command	Response
INST:SEL DMM	
CONF:RES	
CONF?	":SCAL:RES 3e+08,180000"
INST:RES	
INST:SEL?	DMM
CONF?	":SCAL:VOLT:DC 300,0.001"

MEASure[:SCALar|ARRay][:...]?

Command Syntax N/A (query only)

Query Syntax

```
MEASure[:SCALar]

   [:VOLTage]
       :AC?                   [<Expected Value>[,<Resolution>]]
       :ACDC?                 [<Expected Value>[,<Resolution>]]
       [:DC]?                 [<Expected Value>[,<Resolution>]]
   :CURRent[:DC]              [<Expected Value>[,<Resolution>]]
   :RESistance?               [<Expected Value>[,<Resolution>]]
   :FRESistance?              [<Expected Value>[,<Resolution>]]

MEASure:ARRay

   [:VOLTage]
       :AC?<Array Size>       [,<Expected Value>[,<Resolution>]]
       :ACDC?<Array Size>     [,<Expected Value>[,<Resolution>]]
       [:DC]?                 [,<Expected Value>[,<Resolution>]]
   :CURRent[:DC]?             [,<Expected Value>[,<Resolution>]]
   :RESistance?               [,<Expected Value>[,<Resolution>]]
   :FRESistance?              [,<Expected Value>[,<Resolution>]]
```

Command Class Instrument

Query Response

Measured value(s).

***RST Value**

Default Configuration:
300 V, DC Coupled, 10 MΩ Input, 0.001 Resolution, Autoranging ON,

All other configurations at their maximum input ranges, Autoranging ON (see the SENSe command defaults).

Limits

<Array Size>1 to 4096 (2048 for > 1 second aperture)

<Expected Value>
See Tables 3–11 to 3–14

Formats

<Expected Value>
Numeric, MAXimum, MINimum, DEFault, None*

<Resolution>
Numeric, MAXimum, MINimum, DEFault, None*

<Array Size>
Numeric

* If no value is specified, DEFault is used.

Related Commands

CONFigure?
INPut
SENSe
READ?
INITiate
FETCh?
CALibrate:LFRequency
CALibrate:ZERO:AUTO

Description

The MEASure? commands are the top-level commands in the measurement subsystem. They perform all steps necessary to configure, acquire and return measurement data in a single command. The MEASure? commands perform the equivalent of the CONFigure, READ?, INITiate:IMMediate, and FETCh? commands.

MEASure[:VOLTage][:DC]? performs DC voltage measurements.

MEASure[:VOLTage]:AC? performs AC coupled, AC TRMS voltage measurements.

MEASure[:VOLTage]:ACDC? performs DC coupled, AC TRMS voltage measurements.

MEASure:CURRent[:DC]? performs DC current measurements.

MEASure:RESistance? performs two-wire resistance measurements.

MEASure:FRESistance? performs four-wire resistance measurements.

[:SCALar]/:ARRay. The [:SCALar] and :ARRay fields define the number of measurements to be taken. Specifying [:SCALar] defines a single measurement acquisition (<Array Size> = 1). Specifying :ARRay defines multiple measurement acquisitions of count <Array Size>. If <Array Size> is less than 1 or greater than 4096, an error will be generated. Note that specifying an array size of 1 is equivalent to specifying scalar.

<Expected Value>. <Expected Value> is an estimate of the input signal amplitude. If a numeric value is specified, the card will use the range nearest to the value specified to make the measurement (rounded up). <Expected Value> can also be specified as MAXimum, MINimum, or DEFault to use the function's highest range, lowest range, or to default to autoranging respectively. If no value is specified, DEFault (autorange ON) is assumed.

Autoranging uses the currently defined range as the starting value for the autoranging search. If enabled, the card continuously checks and adjusts the input range whenever the signal exceeds the current range, or falls below 9.9% of the range. Once the range has been determined, the card will perform an autozero (if enabled). Specifying any value other than DEFault or none disables autoranging. See the SENSe:RANGe:AUTO command for more detail on autoranging.

Specifying <Expected Value> outside the function's range will generate an error.

<Resolution>. The <Resolution> field defines the resolution of a measurement by implicitly defining the aperture time of the measurement. The aperture is the sampling of the measurement acquisition. The greater the aperture, the more accurate the measurement. If MAXimum is specified as the resolution field, the card will use a 2 second aperture for each measurement. Specifying MINimum uses a 0.8333 ms (1 ms) aperture for 60 (50) Hz line frequency rejection. Specifying DEFault (or none) sets the aperture to 200 ms. See the SENSe[.....]:RESolution commands for the method used to determine the aperture when explicitly defining the <RESolution> field.

For the MEASure:VOLTage[:DC]? commands, <Expected Value> selects the range and input impedances as specified below. |EV| indicates the absolute value of the <expected value>.

Table 3–11: MEASure:VOLTage[DC]? and <Expected Value> ranges

<Expected Value>	Range	Input Impedance
+0.000 ≤ |EV| ≤ +0.030	±30 μV	*
+0.030 < |EV| ≤ +0.300	±300 μV	*
+0.300 < |EV| ≤ +3.000	±3.00 V	*
+3.000 < |EV| ≤ +30.00	±30.0 V	10 MΩ
+30.00 < |EV| ≤ +300.0	±300.0 V	10 MΩ
MAXimum	±300.0 V	10 MΩ
DEFault	Autorange	10 MΩ
MINimum	±30 μV	10 MΩ

***** **The input impedance is dependent on the value specified by the INPut:IMPedance command. It can be either 10 MΩ, or 10 GΩ.**

****** **Specifying DEFault (autorange) as the <expected value> *overwrites* the value specified by the INPut:IMPedance command with 10e6 Ω.**

For the MEASure:VOLTage:AC? and MEASure:VOLTage:ACDC? commands, the TRMS (true root-mean-square) value is returned. The :AC command defines AC coupled voltage measurements (DC rejection). The :ACDC command selects the coupling as AC + DC. EV selects the range as specified below:

Table 3–12: MEASure:VOLTage:AC? and <Expected Value> ranges

<Expected Value>	Range
+0.000 ≤ EV ≤ +0.030	±30 mV
+0.030 < EV ≤ +0.300	±300 mV
+0.300 < EV ≤ +3.000	±3.00 V
+3.000 < EV ≤ +30.00	±30.0 V
+30.00 < EV ≤ +300.0	±300.0 V
MAXimum	±300.0 V
DEFault	Autorange
MINimum	±30 mV

For the MEASure:CURRent? command, <expected value> selects the range as specified in Table 3–13.

Table 3–13: MEASure:CURRent? and <expected value> ranges

<Expected Value>	Range
0.000 ≤ |EV| ≤ 0.150	±150 mA
0.150 < |EV| ≤ 1.000	±1.00 A
MAXimum	±1.00 A
DEFault	Autorange
MINimum	±150 mA

For the MEASure:RESistance? and MEASure:FRESistance? commands, 2-wire or 4-wire resistance measurements are taken respectively. EV selects the range as specified below:

Table 3–14: MEASure:RESistance? and MEASure:FRESistance and <Expected Value> ranges

<Expected Value>	Range
+0 ≤ EV ≤ +30	30 Ω
+30 < EV ≤ +300	300
+300 < EV ≤ +3e3	3e3
+3000 < EV ≤ +30e3	30e3
+30000 < EV ≤ +300e3	300e3
+300e3 < EV ≤ +3e6	3e6
+3e6 < EV ≤ +30e6	30e6
+30e6 < EV ≤ +300e6	300e6
MAXimum	300e6
DEFault	Autorange
MINimum	30 Ω

***NOTE**. Each of the six functions of the DMM maintains a configuration table of its setup, which is independent of the other functions (DC Volts, AC Volts, ACDC Volts, DC Current, 2-wire Resistance, 4-wire Resistance). Issuing a MEASure? command defines the active measurement function for the READ? command, in addition to the array size, range, and resolution (aperture), values of the function's configuration table. The Autorange flag is also set appropriately.*

The MEASure[.....]? commands are executed immediately upon processing. If data sampling is in progress, it will be aborted, and a new acquisition initiated.

For array measurements, the data is returned using SCPI block format.

The format is:

```
HEADERdata,data,...,data<LF>
```

where the HEADER field is

```
#xCOUNT
```

x = the size of the count field

COUNT = the number of bytes returned following the header field.

For example, the format of a "meas:arr? 2" response would be

```
#226+2.98999E+00,+2.99030E+00<LF>
```

The first 2 following the # sign indicates the COUNT size field is 2. The following 26 indicates there are 26 characters in the block following the header.

Examples

Command	Response
Perform a single DC Voltage measurement on the 3 V range. Use the default resolution to set the aperture to 200 ms. Explicitly defining an input range turns off autoranging. Issue the following commands:	
MEASure:SCALar:VOLTage:DC? 2	+1.23456<LF>
CONFigure?	:SCAL:VOLT:DC 3,1e–05

Command	Response
Perform a single AC Voltage measurement on the 300 V (Maximum) range. The resolution specified is rounded per the formula described in the SENSe[.....]:RESolution commands.	
This resolution is equivalent to an aperture of 0.00166667 for 60 Hz rejection	
meas:volt:ac? max, 0.01	+40.9873<LF>
conf?	:SCAL:VOLT:AC 300,0.0109545<LF>
Perform a 2-wire, autorange resistance measurement. The default resolution is used to set the aperture to 200 ms. Issue the following commands:	
meas:resistance?	+12.3456<LF>
CONF?	:SCAL:RES 30,0.0001
Perform three current measurements on the 150 mA range. Specifying the maximum resolution sets the aperture value to 2 seconds. Issue the following commands:	
Meas:arr:curr? 3,0.15,maximum	+0.109876,+0.109678,+0.109444<LF>
conf?	:ARR:CURR:DC 3,0.15,1.58114e-07

READ?

Command Syntax N/A (query only)

Query Syntax `READ?`

Command Class Instrument

Query Response The measurement data for the function currently in effect.

***RST Value** N/A

Limits N/A

Formats N/A

Related Commands

MEASure?
CONFigure
SENSe
INPUt
INITiate
FETCh?

Description

The READ? query is a sub-level command of the measurement subsystem, as its action is dependent on the configuration set up by the higher level MEASure, CONFigure, and SENSe commands. It performs a measurement acquisition by executing an INITiate:IMMediate command, followed by a FETCh? query. The measurement function which is performed is based on the current configuration set up by the higher level commands. For example, if a CONFigure:ARRay:VOLTage:DC? 10,3.0 command was previously issued, a READ? query would initiate, acquire, and return 10 DC V measurements in the 3.0 V range. The READ? query always uses the currently active measurement function. If the CONFigure command above were followed by a SENSe:FUNCtion AC command, a READ? query would execute an AC V measurement.

NOTE*. If data sampling is in progress, issuing a READ? will cause the acquisition to be aborted, and a new acquisition initiated.*

Examples

Command	Response
CONF:VOLT:DC 3.0	define the active configuration as DC Volts
conf?	:SCAL:VOLT:DC 3,1e-05<LF>
READ?	+1.23456<LF> The measured DC voltage
meas:ac? 10	+9.87654<LF> The measured AC voltage
conf?	:SCAL:VOLT:AC 30,0.0001<LF>
read?	+9.67890<LF> The measured AC voltage
sense:function res	define the active configuration as 2W-resistance
conf?	:SCAL:RES 3e+08,1<LF>
read?	+123<LF> The measured resistance

SENSe[:...]

Command Syntax

```
SENSe:FUNCtion <Measurement Function>

SENSe:VOLTage[:DC]
    :AC
    :ACDC
        :RANGe[:UPPer]    <Input Voltage Range>
               :AUTO      <Boolean | ONCE>
        :APERture         <Aperture Value>
        :RPSecond         <Readings/Second>
        :NPLCycles        <Number of Power Line Cycles>
        :RESolution       <Expected Resolution>
        :COUNt            <Array Size>


SENSe:CURRent[:DC]
      :RANGe[:UPPer]      <Input Current Range>
               :AUTO      <Boolean | ONCE>
        :APERture         <Aperture Value>
        :RPSecond         <Readings/Second>
        :NPLCycles        <Number of Power Line Cycles>
        :RESolution       <Expected Resolution>
        :COUNt            < Array Size>

SENSe:RESistance
    :FRESistance
        :RANGe[:UPPer]    <Input Resistance Range>
        :AUTO             <Boolean | ONCE>
        :APERture         <Aperture Value>
        :RPSecond         <Readings/Second>
        :NPLCycles        <Number of Power Line Cycles>
        :RESolution       <Expected Resolution>
        :COUNt            < Array Size>
```

Query Syntax

```
SENSe[.......]?
```

Command Class

Instrument

Query Response

Requested Information

***RST Value**

SENSe:VOLTage[:DC]
300 V, Aperture = 0.2 seconds, NPLC = 12 (60 Hz), 10 (50/400 Hz), Resolution 0.001, Autoranging ON

SENSe:VOLTage:AC
300 V, Aperture = 0.2 seconds, NPLC = 12 (60 Hz), 10 (50/400 Hz), Resolution 0.001, Autoranging ON

SENSe:VOLTage:ACDC
300 V, Aperture = 0.2 seconds, NPLC = 12 (60 Hz), 10 (50/400 Hz), Resolution 0.001, Autoranging ON

SENSe:CURRent[:DC]
1.0 A, Aperture = 0.2 seconds, NPLC = 12 (60 Hz), 10 (50/400 Hz), Resolution 5.0e-6, Autoranging ON

SENSe:RESistance
300.0e6 Ohms, Aperture = 0.2 seconds, NPLC = 12 (60 Hz), 10 (50/400 Hz), Resolution 10,000, Autoranging ON

SENSe:FRESistance
300.0e6 Ohms, Aperture = 0.2 seconds, NPLC = 12 (60 Hz), 10 (50/400 Hz), Resolution 10,000, Autoranging ON

Limits

*<Input Voltage Range> ± 0.03, 0.3, 3.0, 30.0, 300.0 Volts

<Input Current Range> ± 0.150, 1.0 Amps

<Input Resistance Range> 30, 300, 3e3, 30e3, 300e3, 3e6, 30e6, 300e6 Ω

**<Aperture Value>,<Readings/Second>,<Number of Power Line Cycles>, <Resolution>

<Array Size> 1 to 4096 (2048 for aperture >1 second)

* AC/ACDC voltage ranges are positive only.

** See the text below.

Formats

<Measurement Function> DC | AC | ACDC | CURRent | RESistance | FRESistance

<Boolean> ON | 1 | OFF | 0

Related Commands

MEASure?
CONFigure
READ?
INPut
CALibrate:LFRequency

Description

The SENSe commands are low-level commands that customize the setup of the MEASure? and CONFigure commands. For example, if you issued the command MEAS:VOLT? 3.0, and the measured value(s) were overranging, issuing a SENSe:VOLT 30.0 command would switch to the 30 V range without modifying any of the other setups of the DC voltage configuration.

Each of the six functions of the DMM maintains a configuration table of its setup, which is independent of the other functions (DC Volts, AC Volts, ACDC Volts, DC Current, 2-wire Resistance, 4-wire Resistance). The SENSe commands allow individual customization of these parameters for each of the functions.

SENSe:FUNCtion. The SENSe:FUNCtion commands define the measurement function for subsequent READ? commands. For example, if the present configuration mode is a DC voltage measurement, issuing a SENSe:FUNCtion RESistance command will cause the next measurement to be a 2-wire resistance measurement.

SENSe[....]:COUNt. The SENSe[.....]:COUNt commands define the number of measurements to acquire. If <Array Size> is less than 1 or greater than 4096, an error will be generated. Specifying an <Array Size> of 1 is equivalent to specifying scalar.

SENSe[....]:RANGe. The SENSe[.....]:RANGe commands modify the applicable input range setup. Issuing this command disables autoranging. All range values are rounded up to the nearest applicable value. Specifying a value outside the function's range will generate an error.

SENSe[.....]:RANGe:AUTO. The SENSe[.....]:RANGe:AUTO commands control when the card performs auto-ranging.

If you program a SENSe:RANGe:AUTO:ONCE, the card will perform an auto range each time it receives a MEASure?, READ? or INITiate command, *prior to the first measurement only.*

For example, the command MEAS:ARR? 100 would autorange prior to the first measurement, and would then use the range determined for making the 100 measurements. If any measurement exceeded the determined range, a +9.9E+37 (or –9.9E+37) value is stored as the measurement value.

If you program a SENSe:RANGe:AUTO ON command, the card continuously monitors the level of the input signal for every measurement taken. If the signal exceeds the current range, or falls below 9.9% of the range, the card will autorange to seek the appropriate range, and retake the measurement on the new range. If autozeroing is enabled, the card performs an autozero after each new range is determined.

***NOTE**. Note that autoranging is individually programmable for each function of the card.*

For DC V measurements, enabling autoranging will overwrite the INPut:IMPedance value with 10e6 ohms.

```
SENSe[.....]:APERture     <aperture value>
SENSe[.....]:RPSecond     <readings per second>
SENSe[.....]:NPLCycles    <number of Powerline cycles>
SENSe[.....]:RESolution   <Expected Resolution>
```

These commands define four different ways of specifying the aperture time of a measurement, which is the acquisition or sampling time for the measurement. The relationship between the aperture, readings per second, number of power line cycles, and expected resolution are per the formulae below.

All calculated values are internally rounded to values based on apertures supported by the hardware.

The CONFigure[.....]? queries can be used to get the actual (rounded) values calculated by the card for these parameters. If any field corresponds to an aperture less than .0008333 seconds (.001 seconds for 50 Hz line frequency rejection), or an aperture greater than 2.0 seconds, the card will limit the value to that range. Within these limits, the aperture will be rounded to the next allowable aperture value.

***NOTE**. The rounding takes place within reasonable limits. For example, .0166 will be rounded up to .016666....*

The rounded aperture values permitted by the hardware design are as follows

For 60 Hz line frequency rejection:

From .000833... to .2725 seconds in .000833... second steps.

From .2733... to 1.0933... seconds in .00333... second steps.

From 1.10 to 2.0 seconds in .00833... second steps.

For 50 Hz line frequency rejection:

From .001 to .273 seconds in .001 second steps.

From .276 to 1.092 seconds in .004 second steps.

From 1.10 to 2.0 seconds in .01 second steps.

The readings per second and number of power line cycles are calculated based on the allowable aperture as follows:

Readings per Second = 1. / Aperture.

Number of Power Line Cycles = 60. * Aperture, for 60 Hz line frequency rejection, or 50. * Aperture, for 50 Hz line frequency rejection.

If the card is directly programmed for a specified Readings per Second, of Number of Power Line Cycles per second. Using one of the command alternatives above, aperture is solved for using the appropriate formula above, rounded to an allowable value. The Readings per Second and Number of Power Line Cycles will be recalculated based on the actual resulting aperture.

The expected resolution is also related to the measurement aperture. The intent of programming resolution is to provide a means of specifying the approximate accuracy of the measurement. It is not necessarily equal to the resolution of the digits in the readings returned by the card. The VX4101 calculates an expected resolution that roughly approximates the noise level of the measurement. Noise level is inversely proportional to the square root of the aperture.

***NOTE**. For readability reasons, 5.5 digits is returned for an aperture equal one power line cycle or greater, and 4.5 digits is returned for smaller apertures.*

Expected Resolution

The formula for expected resolution uses this relationship with a scaling that provides 5.5 digits resolution (10 uV in the 3V range) for the default aperture of .200 seconds, and 4.5 digits expected resolution for an aperture of 2.0 msec, which approximates the noise level of the card. The formula used to calculate the expected resolution is as follows.

Resolution = (Range/300000.) * (.200/Aperture)^.5

Aperture = .2 * Range^2 / (300000. * Resolution)^2

***NOTE**. The formula to calculate aperture from expected resolution is the inverse.*

For the 1.5A current range and the 300 Mohm resistance range, use a Range of 1.5Amps and 54 Gohms respectively, in the formulas above.

Expected resolution is only an approximation. If you are concerned about optimizing the tradeoff between aperture/ test time and accuracy, use the Accuracy Specifications in the Appendix to determine an aperture. The accuracies and apertures listed in the Appendix are tested at the factory. The accuracy at apertures in between those tested may be assumed to change logarithmically from one specified aperture to the next aperture. For example , if the accuracy for some measurement level calculates to 120 uV for a .2 second aperture, and 140 uV for a .0167 second aperture, the accuracy for a .05 second aperture may be calculated as follows:

140 uV = 120 uV + k*log(.2/.0167), solving for k, k= 18.547 uV.

.05 second aperture accuracy = 120 uV + 18.547 uV * log(.2/.05) = 131.2 uV.

***NOTE**. Before choosing an aperture that is not a power line cycle integer multiple, it should be noted that the accuracy specifications assume a quiet power environment. In many testing environments, the power line noise contributes measurable error, particularly in the most sensitive voltage ranges and in the highest resistance ranges. An aperture that is a power line cycle integer multiple is recommended.*

Finally, if autoranging is enabled, the aperture value will remain unchanged if the card input range changes. However, the expected resolution value will be recalculated based on the new range found.

Example. If the command SENSe:VOLT:DC:RESolution 2.8e-4 were given, and the current DC voltage range were 30; the aperture of the measurement would be:

$$\text{Aperture}= .2 \times 30. \times 30./300000 \times 2.8c-4)^2$$

$$= 0.025510 \text{ seconds}$$

To determine the rounded aperture, divide 0.25510 by the aperture step size of .0008333 and round the result of 30.61 to 30.

The resulting aperture is: 30 × .0008333

The SENSe[.....]? query commands return the requested information.

***NOTE**. If data sampling is in progress, issuing a SENSE command for any function of the configuration currently in effect, or issuing a SENSe:FUNCtion command will cause the acquisition to be aborted.*

Examples

Command	Response
SENSe:VOLTage:DC: RANGe:UPPer 5	Set the DC V range
sens:volt:range?	30<LF> The rounded up range value
sense:voltage:ac:aperture 0.5	Set the AC V aperture value

Command	Response
sens:volt:ac:aper?	0.5<LF> The aperture value
sense:volt:rps?	2<LF> The number of readings per second
SENSE:CURRENT: NPLCYCLES 120	Set to 120 power line cycles
sens:curr:nplc?	120<LF>
sens:curr:aper?	2<LF> Modifying the power line automatically modifies the aperture; the converse is also true
Sense:Resistance:Resolution 0.1	Define the resolution
sens:res:resolution?	56921<LF> The calculated resolution
sense:fres:res?	180000<LF> The current (default) resolution
sense:fres:range 30	change the 4-wire resistance range to 30
sense:fres:res?	0.0001<LF> The calculated resolution
sense:volt:range:auto 1	enable DC V autoranging. 1 is interchangeable with ON
sense:volt:range:auto?	1<LF>

STATus:OPERation:CONDition?

Command Syntax N/A

Query Syntax `STATus:OPERation:CONDition?`

Command Class Instrument

Query Response The operational condition register value.

***RST Value** 0

Limits N/A

Formats Query Response
Numeric

Related Commands MEASure?
READ?

INITiate
ABORt

Description The STATus:OPERation:CONDition query returns the current operational status of the DMM board. The bit definitions of the value are (bit 0 = the least significant bit):

Bit	Definition	Function
0	Calibrating	Set when any CALibration operation is running. Cleared when the CALibration operation is complete
1	Settling	Set when the instrument changes its function or range. Cleared when the all circuitry has settled
2	Ranging	Set when the instrument is autoranging. Cleared when the input range has been found
3	Sweeping	Not used
4	Measuring	Set when the INITiate command is executed. Cleared when the command is complete or aborted
5	Triggering	Set when the instrument is waiting for a trigger signal. Cleared when the trigger is received
6	Arming	Not used
7	Correcting	Set when the instrument is performing an autozero operation. Cleared when the autozero operation is complete
8	Testing (User 1)	Set when the instrument is performing a self-test. Cleared when the self-test is complete
9	Aborting (User 2)	Set when the instrument is in the process of aborting an operation. Cleared when the abort is complete
10	User 3	Not used
11	User 4	Not used
12	User 5	Reserved
13	Instrument Summary	Not used
14	Program Running	Not used
15	Reserved	Always 0

Table 3–15: STATus:OPERation:CONDition Examples

Command	Response	Description
status:operation:condition?	16	Measurement in progress (0010 hex).
stat:oper:cond?	3072	Measurement complete because an ABORt was received (0C00 hex).

TEST:ALL?

Command Syntax N/A

Query Syntax `TEST:ALL?`

Command Class Instrument

Query Response `"DMM Passed Self Test"` or
`"DMM Failed Self Test"`

***RST Value** N/A

Limits N/A

Related Commands *TST

Description This query is used to perform a self test of the DMM. If the test fails, an error message is placed in the error queue and the error LED begins to blink. The following are tested:

- Acquisition memory and control logic
- Programmable aperture circuitry
- Autozero test covering the following:
  - DC amplifier
  - AC amplifier
  - TRMS converter
  - A/D converter
  - Serial to parallel conversion circuitry

Examples

Command	Response
INST:SEL DMM	Selects the DMM
TEST:ALL?	"DMM Passed Self Test"

VXI[:SERVant]:FDC:SEL

These commands allow the setup and use of the instruments Fast Data Channels. The VX4101 architecture implements FDC V2.0 with a maximum of eight FDC channels per VXIbus module. The DMM instrument itself only utilizes one FDC channel.

Command Syntax `VXI[:SERVant]:FDC:SEL <channel number>`

Query Syntax `VXI[:SERVant]:FDC:SEL? <channel number>`

Query Response The query returns the currently selected logical <channel number>. The return value is a single integer, from 1 to the number of logical FDC channels the instrument supports.

***RST Value** Channel 1

Description This command selects the FDC channel to be used by subsequent FDC commands. The channel number parameter is the logical FDC channel number for the instrument. It is not necessarily related to the physical FDC channel.

Command Syntax `VXI[:SERVant]:FDC:OPEN <channel mode> [,<channel number>]`

***RST Value** All open FDC Channels are closed

Description This command opens the selected logical FDC channel. The mode parameter determines the direction of data flow, read only, write only or both read and write. The mode parameter can be one of the following unquoted SCPI labels: RO, WO, RW.

Interpretation of Read and Write is from the servant device's point of view. Data is READ off the VXIbus back plane into the instrument and WRITTEN from the instrument to the VXIbus back plane. If the optional channel number is omitted, the channel referenced by the VXI:FDC:SEL command is used. FDC channels must be opened before they can be accessed for data transfer.

Each opened FDC channel consumes system resources. Bidirectional channels require more resources than Read or Write only channels. Only open FDC channels in the operational mode appropriate for their intended use.

Command Syntax `VXI[:SERVant]:FDC:CLOSe[<channel number>]`

***RST Value** All open FDC Channels are closed

Description This command closes the selected logical FDC channel. If the optional channel number is omitted, the channel referenced by the VXI:FDC:SEL command is used. Closed FDC channels must be opened before they can be accessed for data transfer.

Command Syntax N/A

Query Syntax `VXI[:SERVant]:FDC:CONFiguration?[<channel number>]`

Query Response This query-only command returns the configuration state of the selected FDC channel. If the optional channel number is omitted, the channel referenced by the VXI:FDC:SEL command is used. Return values are the following unquoted ASCII strings:

> CLOSED, OPEN, INITIALIZED, READ ONLY, WRITE ONLY, READ_WRITE

Configuration states of READ ONLY, WRITE ONLY and READ_WRITE imply an OPENED and INITIALIZED state. A channel that is in a CLOSED state must be OPENED before it can be accessed. In practice, a channel will only ever be CLOSED or in one of the three active states: READ ONLY, WRITE ONLY, or both.

***RST Value** `N/A`

Command Syntax `N/A`

Query Syntax `VXI[:SERVant]:FDC?`

Query Response This query only command returns comma separated list of physical FDC channel numbers allocated to the instrument. Physical and logical channel numbers have a one to one correspondance. That is, logical channel 1 is implemented on the first physical channel number returned by the query; logical channel 2 is implemented on the second physical channel number returned in the list, etc.

Physical FDC channel allocation is device dependent. Knowledge of a logical FDC channels physical identity is required by users writing their own low level Commander-side FDC drivers for their host computer.

***RST Value** N/A

Examples

Command	Response
VXI:FDC:SEL <1>	Select logical FDC channel 1 for access
VXI:FDC:Open RO	Open FDC channel 1 for read only operation
VXI:FDC?	2,4 Indicates physical FDC channel 2 is implemented on this device's logical channel 1; physical channel 4 on logical channel 2
VXI:FDC:CONF? 1	READ ONLY Indicates channel 1 is OPENED, INITIALIZED, and READ ONLY
VXI:FDC:CLOSe	Closes FDC channel 1

Trigger Commands for the DMM

A summary of the trigger-related commands for the DMM is as follows:

```
INSTrument:ABORt
```

```
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:DELay
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:ECOunt
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:IMMediate
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:MODE
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:[ALL]?
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce
```

INSTrument:ABORt

Command Syntax `INSTrument:ABORt`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands ABORt
INITiate
INITiate:CONTinuous

Description Places active instrument in the IDLE state, aborting any measurement or other instrument activity in progress. The instrument configuration is unchanged and a subsequent INIT command will cause the instrument to restart the same type of measurement. If the instrument is in asynchronous mode, this command can be sent while a query is in progress and the measurment will be aborted. If however, the instrument is in synchronous mode, this command will be queued while a query is in progress. This is a ramification of the IEEE 488.2 Message Exchange Protocol Enforcer (MEPE).

Examples

Command	Response
INST:SEL DMM	
CONF:ARR:VOLT:DC	512
INIT	
FETC:COUN?	127
INST:ABOR	
FETC:COUN?	153
FETC:COUN?	153

NOTE*. After abort, no more measurements are being taken.*

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:DELay

Command Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:DELay <delay in seconds>`

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:DELay?`

Command Class Instrument

Query Response `<delay in seconds>`

***RST Value** 0 seconds (pass-through)

Limits 0 = pass through, 1 μs – 65.535 ms in 1 μs steps

Related Commands TRIGger:ECOunt

Description Specifies a time delay to occur after receipt of a trigger prior to actually triggering. If the trigger source selected is fixed, this command will have no effect on the instrument triggering. This command always zeros the event count delay, so specifying a delay of zero places the trigger subsystem in pass-through mode. In this mode, the instrument triggers immediately upon receipt of a trigger.

Examples

Command	Response
INST:SEL DMM	
CONF:VOLT:DC	
TRIG:SOUR TTLT0	
TRIG:DEL 1E-3	
INIT	

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:ECOunt

Command Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:ECOunt <triggers to count>`

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:ECOunt?`

Command Class Instrument

Query Response `<triggers to count>`

***RST Value** 0 triggers (pass-through)

Limits 0 = pass through, 1 to 65,535 triggers

Related Commands TRIGger:DELay

Description Specifies the number of triggers to count prior to triggering. Upon receipt of trigger N (where N is the number specified in the comand), the instrument will enter the triggered state. If the trigger source selected is fixed, this command will have no effect on the instrument triggering. This command always zeros the delay by time parameter, so specifying an event count of zero places the trigger subsystem in pass-through mode. In this mode, the instrument triggers immediately upon receipt of a trigger.

Examples

Command	Response
INST:SEL DMM	
CONF:VOLT:DC	
TRIG:SOUR TTLT0	
TRIG:ECO 100	
INIT	

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:IMMediate

Command Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:IMMediate`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Causes a one time entry into the triggered state without receiving the specified trigger. This command is often used to “prime the pump” in cases such as setting up a scan list measurement.

Examples

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM
INST:SEL DMM	
CONF:VOLT:DC	
TRIG:SOUR TTLT0	
INIT	
FETC:COUN?	0
TRIG:IMM	
FETC:COUN?	1
FETC?	–6.03720E+00

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:MODE

Command Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:MODE ONCE|ALL`

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:MODE?`

Command Class Instrument

Query Response `Current state:ONCE|ALL`

***RST Value** ALL

Limits ONCE, ALL

Related Commands N/A

Description When an instrument has been configured for some type of array measurement, this command specifies whether the instrument will perform one or all operations when a trigger is received. If the mode is ALL, then all operations will be completed upon receipt of one trigger condition. If the mode is ONCE, then the instrument will perform one operation and reenter the initiated state. This will continue until the specified number of triggers has been received (and hence, the specified number of operations has been completed).

Examples

Command	Response
INST:SEL DMM	
CONF:ARR:VOLT:DC 3	
TRIG:MODE ONCE	
TRIG:SOUR COMM0	
INIT	
FETC:COUN?	0
TRIG:FIR0	
FETC:COUN?	1
TRIG:FIR0	
TRIG:FIR0	

Command	Response
FETCH:COUN?	3
FETC?	#239–6.04180E+00,–6.04180E+00,–6.04180E+00

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?

Command Syntax N/A

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?`

Command Class Instrument

Query Response `HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value** N/A

Limits N/A

Related Commands TRIGger:SOURce:CATalog:DELayable?
TRIGger:SOURce:CATalog:FIXed?
TRIGger:SOURce
TRIGger:FIRe4
TRIGger:TIMer

Description Lists all trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?

Command Syntax N/A

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?`

Command Class Instrument

Query Response `BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0 COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value** N/A

Limits N/A

Related Commands
TRIGger:SOURce:CATalog
TRIGger:SOURce:CATalog:FIXed?
TRIGger:SOURce
TRIGger:FIRe4 TRIGger:TIMer

Description Lists all delayable trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT:DEL?	BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?

Command Syntax N/A

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?`

Command Class Instrument

Query Response `HOLD,IMMEDIATE,TIMER`

***RST Value** N/A

Limits N/A

Related Commands TRIGger:SOURce:CATalog
TRIGger:SOURce:CATalog:DELayable?
TRIGger:SOURce
TRIGger:FIRe4
TRIGger:TIMer

Description Lists all fixed trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT:FIX?	HOLD,IMMEDIATE,TIMER

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce

Command Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce <source>`

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce?`

Command Class Instrument

Query Response `<current source>`

***RST Value** IMMEDIATE

Limits BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM

Related Commands TRIGger:SOURce:CATalog?
TRIGger:FIRe4
TRIGger:TIMer

Description Selects or queries the trigger source to be used when the instrument is initiated.

Examples

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM
INST:SEL DMM	
CONF:VOLT:DC	
TRIG:SOUR BUS	
INIT	
FETC:COUN?	0
*TRG	
FETC:COUN?	1
FETC?	–6.03720E+00

SCPI Commands for the Counter

This section contains SCPI command summaries for the Universal Counter (Counter). The commands are organized by command subsystem. Some commands within each subsystem are organized by type of command rather than in alphabetical order.

```
CALCulate:AVERage?
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCOunt?
CALCulate:LIMit:LOWer[:DATA]
CALCulate:LIMit:REPort[:DATA]?
CALCulate:LIMit:UPPer[:DATA]
```

```
CALibrate:ARM[:VALue]
CALibrate[<channel>]:DELay
CALibrate:DTI
CALibrate[<channel>]:VALue
CALibrate[<channel>]:HYSTeresis
CALibrate[<channel>]:LFCOmp
CALibrate[<channel>]:LINearity
CALibrate3:BIAS
CALibrate[<channel>]:ZERO
CALibrate[3]:ROSCillator <input frequency>
```

```
CONFigure[1|2][:SCALar|ARRay][:VOLTage][:...]
CONFigure[1|2|3][:SCALar|ARRay]:FREQuency
CONFigure[1|2|3][:SCALar|ARRay]:PERiod
CONFigure[1|2|3][:SCALar|ARRay]:FREQuency:RATio
CONFigure[1|2][:SCALar|ARRay][:RTIMe|FTIMe]
   :RISE:TIME
   :RTIMe
   :FALL:TIME
   :FTIMe
CONFigure[1|2][:SCALar|ARRay]:NWIDth|PWIDth
CONFigure[1|2][:SCALar|ARRay]:TINTerval
CONFigure[1|2][:SCALar|ARRay]:TINTerval:DELay
CONFigure[1|2][:SCALar]:TOTalize
```

```
FETch?
FETCh[:VOLTage][:...]?
   :AC?
   :DC?
   :MINimum?
   :MAXimum?
   :PTPeak?
```

```
FETCh[:...]
   :PERiod?
   :FREQuency?
   :FREQuency:RATio?
   :RISE:TIME?
   :RTIMe?
   :FALL:TIME?
   :FTIMe?
   :NWIDth?
   :PWIDth?
   :TINTerval?
FETCh:TOTalize?
FETCh:COUNt?

INITiate[:IMMediate|CONTinuous]

INPut[1|2]:OFFSet[:ABSolute]
INPut[1|2]:OFFSet:RELative
INPut[1|2]:COUPling
INPut[1|2]:IMPedance
INPut[1|2]:FILTer[:LPASs]:FREQuency
INPut[1|2]:FILTer[:LPASs][:STATe]
INPut[1|2]:ATTenuation
INPut[1|2]:GAIN
INPut[1|2]:COMParator[1|2]:LEVel[:ABSolute]
INPut[1|2]:COMParator[1|2]:LEVel:RELative
INPut[1|2]:COMParator[1|2]:HYSTeresis[:ABSolute]
INPut[1|2]:COMParator[1|2]:HYSTeresis:RELative
INPut[1|2]:COMParator[1|2]:SLOPe
INPut[1|2]:SETup
INPut:SETup:AUTO

INSTrument:RESet

MEASure[1|2][:SCALar|ARRay][:VOLTage][:...]?
MEASure[1|2|3][:SCALar|ARRay]:FREQuency?
MEASure[1|2|3][:SCALar|ARRay]:PERiod?
MEASure[1|2|3][:SCALar|ARRay]:FREQuency:RATio?
MEASure[1|2][:SCALar|ARRay]:RTIMe|FTIMe?
   :RISE:TIME?
   :RTIMe?
   :FALL:TIME?
   :FTIMe?
MEASure[1|2][:SCALar|ARRay]:NWIDth|PWIDth?
MEASure[1|2][:SCALar|ARRay]:TINTerval?
MEASure[1|2][:SCALar|ARRay]:TINTerval:DELay?
MEASure[1|2][:SCALar]:TOTalize?
```

```
READ?

SENSe[:...]:COUNT
SENSe[:...]:APERture?
SENSe[:...]:EVENts
SENSe[:...]:MODe
SENSe[1,2,3]:FUNCtion
SENSe:TINTerval:DELay[:STATe]
SENSe:TINTerval:DELay:TIME

STATus:OPERation:CONDition?
```

Trigger Commands

In addition, this section includes a listing of the commands used for triggering measurements for the Counter. A summary of the trigger commands follow the instrument-specific SCPI commands.

CALCulate:AVERage?

Command Syntax

N/A

Query Syntax

`CALCulate:AVERage? [< count > [,< offset > [,< step_size >]]]`

Command Class

Instrument

Query Response

`<average_data>`

***RST Value**

< DEF > (count), 1 (offset) , 1 (step_size)

Limits

1 – 4 K (count), 1 – 4 K (offset), 1 – 4 K (step_size)

Related Commands

N/A

Description

This query averages a specified number of measurements in the memory buffer. The optional parameters are for averaging selected values in the memory buffer. The optional <count> parameter specifies the number of data points to calculate. If no <count> is specified, then the last number measurements taken will be the default. The optional <offset> parameter determines at which point in memory values will start to be averaged. The optional <step_size> parameter determines the number of measurements to skip before taking a value to be averaged.

***NOTE**. In order to specify a <step_size>, the user must enter the <count> and the <offset> information.*

Examples

Command	Response
inst:sel ctr	selects the Counter
calc:lim:upper 5.0	sets upper data value limit
calc:aver?	averages all the data that has been collected
calc:aver? 20,3,2	averages 20 measurement readings. Starts with the 3rd measurement. Averages every other data point

CALCulate:LIMit:ENVelope[:DATA]

Command Syntax

`CALCulate:LIMit:ENVelope[:DATA] <threshold1>,<threshold2>`

Query Syntax

`CALCulate:LIMit:ENVelope[:DATA]?`

Command Class

Instrument

Query Response

`<threshold1>,<threshold2>`

***RST Value**

N/A

Limits

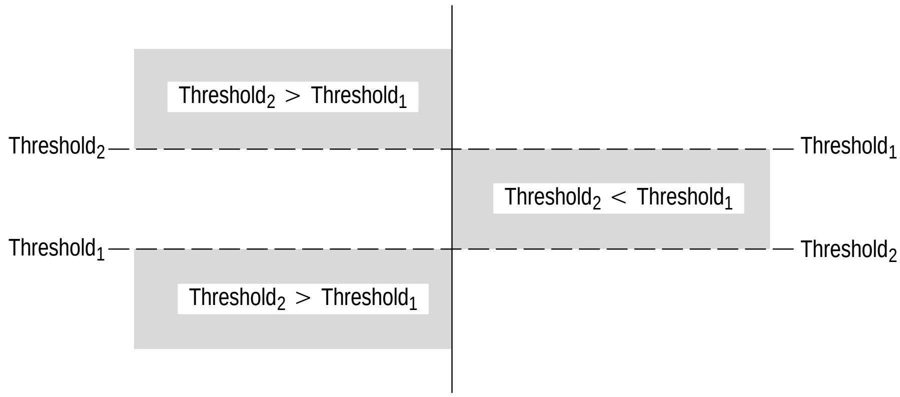
N/A

Related Commands

CALCulate:LIMit:ENVelope:POINt
CALCulate:LIMit:LOWer[:DATA]
CALCulate:LIMit:UPPer[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description

This command searches for all the input data values above and below a set of voltages or in between a set of voltages. The range of the envelope is determined by the *<threshold1>* and the *<threshold2>*. If the *<threshold1>* is greater than the *<threshold2>*, the range of data values searched for is above the <threshold1>value or below the *<threshold2>* value. If the *<threshold1>* is less than *<threshold2>*, the range of data values searched for is between the *<threshold1>* and *<threshold2>*. The query for this command returns the boundary values.



Examples

Command	Response
inst:sel ctr	selects the Counter
calc:lim:env 5000,1000	Sets an envelope value. All data points above 05.0 kHz and below 1.0 kHz are targeted.
calc:lim:env?	1.0, 5.0<lf> returns the envelope threshold value
calc:lim:env 1000,5000	Sets an envelope value. All data points between 5.0 kHz and 1.0 kHz are targeted.
calc:lim:env?	5000, 1000<lf> returns the envelope threshold value

CALCulate:LIMit:FCOunt?

Command Syntax N/A

Query Syntax `CALCulate:LIMit:FCOunt?`

Command Class Instrument

Query Response `<numeric_value>`

***RST Value** N/A

Limits N/A

Related Commands CALCulate:LIMit:REPort[:DATA]
CALCulate:LIMit:LOWer:POINt

CALCulate:LIMit:UPPer[:DATA]
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description

This query command performs a limit test on the current available data and returns the number of data points that failed a limit test.

***NOTE**. You must send a CALC:limit:upper, lower, or envelope command before this query.*

Examples

Command	Response
inst:sel ctr	selects the Counter
calc:lim:upp 5000	sets upper threshold value
calc:lim:fco?	2<lf> returns the number of data points outside of a defined threshold.

CALCulate:LIMit:LOWer[:DATA]

Command Syntax `CALCulate:LIMit:LOWer[:DATA] <threshold>`

Query Syntax `CALCulate:LIMit:LOWer[:DATA]?`

Command Class Instrument

Query Response `<threshold>`

***RST Value** N/A

Limits –300 V to +300 V

Related Commands

CALCulate:LIMit:LOWer:POINt
CALCulate:LIMit:UPPer[:DATA]
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description This command searches for all the input data values below a certain threshold. The query for this command returns the boundary value.

***NOTE**. You must send a CALC:limit:upper, lower, or envelope command must be sent before this query.*

Examples

Command	Response
inst:sel ctr	selects the Counter
calc:lim:lower 5.0	sets lower threshold value
calc:lim:lower?	5.0<lf> returns the lower threshold value

CALCulate:LIMit:REPort[:DATA]?

Command Syntax N/A

Query Syntax `CALCulate:LIMit:REPort[:DATA]?`

Command Class Instrument

Query Response `<memory_index>,<failed_value>[{;<memory_index>,<failed_value>}]`

***RST Value** N/A

Limits N/A

Related Commands CALCulate:LIMit:REPort:POINts

Description This query command returns the indices of the data values collected in the most recent CALCulate:LIMit:...[:DATA] command.

***NOTE**. The CALCulate:LIMit:...[:DATA] command must be run before this command. If no data values were found a zero will be returned. (The first point in memory is "1")*

***NOTE**. You must send a CALC:limit:upper, lower, or envelope command before this query.*

Examples

Command	Response
inst:sel ctr	selects the Counter
calc:lim:upp 5000	sets upper threshold value
calc:lim:rep?	2, 6002.0; 5,5001.0<lf> returns the number of data points and data values outside of a defined threshold

CALCulate:LIMit:UPPer[:DATA]

Command Syntax `CALCulate:LIMit:UPPer[:DATA] <threshold>`

Query Syntax `CALCulate:LIMit:UPPer[:DATA]?`

Command Class Instrument

Query Response `<threshold>`

***RST Value** N/A

Limits N/A

Related Commands
CALCulate:LIMit:UPPer:POINt
CALCulate:LIMit:LOWer[:DATA]
CALCulate:LIMit:ENVelope[:DATA]
CALCulate:LIMit:FCount
CALCulate:LIMit:REPort[DATA]?

Description This command searches for all the input data values above a certain threshold. The query for this command returns the boundary value.

***NOTE**. A CALC:limit:upper, lower, or envelope command must be sent before this query.*

Examples

Command	Response
inst:sel ctr	selects the ctr
calc:lim:upper 5.0	sets upper threshold value
calc:lim:upper?	5.0<lf> returns the upper threshold value

CALibrate:ARM[:VALue]

Command Syntax `CALibrate:ARM[:VALue] <arm input voltage>`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits <arm input voltage>
0.0 or 20.0

Formats <arm input voltage>
Numeric

Related Commands N/A

Description The CALibrate:ARM:VALue <arm input voltage> command performs a gain and zero calibration on the ARM input. This command requires an input calibration source set to 0.0 ±0.001 V for calibrating the offset, and 20.0 V ±0.1% for calibrating the gain.

Examples

Command	Response
CALIBRATE:ARM:VALUE 0	N/A, Perform the ARM input offset correction
cal:arm 20	N/A, Perform the ARM input gain correction

CALibrate[<channel>]:DELay

Command Syntax

```
CALibrate[<channel>]:DELay
CALibrate:DELay <cross channel specification>
```

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits

<cross channel specification>	12 or 21
<channel>	1 or 2

Formats

<cross channel specification>	Numeric
<channel>	Numeric

Related Commands N/A

Description

The CALibrate[<channel>]:DELay command performs calibration of the internal delay cabling for the channel specified. This command requires an input frequency source set to a 10 MHz square wave at ±1.0 V ±0.1 V.

The CALibrate:DELay <cross channel specification> commands perform cross-channel delay calculations between the two input channels. This command requires the 10 MHz square wave to be connected to both channel inputs of the card using a ‘T’ connector and equal length cables.

<cross channel specification> can be either 12 or 21 for calculating channel 1 to 2’s delay, and channel 2 to 1’s delay respectively.

Examples

Command	Response
CALIBRATE1:DELAY	N/A, Perform the channel 1 cabling delay calibration
cal2:del	N/A, Perform the channel 2 cabling delay calibration
CAL:DELAY 12	N/A, Perform the channel 1 to 2 cross channel delay calibration
cal:delay 21	N/A, Perform the channel 2 to 1 cross channel delay calibration

CALibrate:DTI

Command Syntax `CALibrate:DTI`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits N/A

Formats N/A

Related Commands N/A

Description The CALibrate:DTI command performs a statistical analysis of the internal hardware for determining the setting of the digital time interpolation bit used by the Counter hardware. This command requires an input frequency source set to a 10 MHz square wave at ±0.5 V ±0.1 V.

Examples

Command	Response
CALIBRATE:DTI	N/A, Perform the DTI calculations.

CALibrate[<channel>]:VALue

Command Syntax `CALibrate[<channel>]:VALue <input voltage>`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits

<input voltage>
per the table values specified below.

Formats

<channel>
1 or 2

<input voltage>
Numeric

Related Commands

N/A

Description

The CALibrate[<channel>]:VALue <input voltage> command calculates the gain correction factors for the Counter attenuation and gain settings. This command requires an input calibration source set to the <input voltage> values below (±0.1%).

Table 3–16: Input Calibration Source Settings

<input voltage>	<input voltage>
+50.0	–50.0
+5.0	–5.0
+0.5	–0.5
–1.0	

[<channel>] specifies the channel being calibrated, and can be specified as either 1 or 2. If [<channel>] is not specified, channel 1 is assumed.

Examples

Command	Response
CALIBRATE1:VALUE 5	N/A, Perform the gain correction for channel 1 for a 5 V input
CAL2:VAL –5.0	N/A, Perform the gain correction for channel 2 for a –5.0 V input

CALibrate[<channel>]:HYSTeresis

Command Syntax `CALibrate[<channel>]:HYSTeresis`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits N/A

Formats <channel>
1 or 2

Related Commands N/A

Description The CALibrate[<channel>]:HYSTeresis command calculates the hysteresis linearization factors. This command uses an internal reference, and so does not require any external input.

[<channel>] specifies the channel being calibrated, and can be specified as either 1 or 2. If [<channel>] is not specified, channel 1 is assumed.

Examples

Command	Response
CALIBRATE1:HYSTERESIS	N/A, Perform the hysteresis linearization for channel 1
CAL2:HYST	N/A, Perform the hysteresis linearization for channel 2

CALibrate[<channel>]:LFCOmp

Command Syntax `CALibrate[<channel>]:LFCOmp`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits N/A

Formats <channel> 1 or 2

Related Commands N/A

Description The CALibrate[<channel>]:LFCOmp command performs a low frequency compensation on the inputs. This command requires an input frequency source set to a square wave of ±2.5 V ±0.1 V at 1 kHz.

[<channel>] specifies the channel being calibrated, and can be specified as either 1 or 2. If [<channel>] is not specified, channel 1 is assumed.

Examples

Command	Response
CALIBRATE1:LFCOMP	N/A, Perform the low frequency compensation for channel 1
CAL2:LFCO	N/A, Perform the low frequency compensation for channel 2

CALibrate[<channel>]:LINearity

Command Syntax `CALibrate[<channel>]:LINearity`

Query Syntax N/A

Query Syntax N/A

***RST Value** N/A

Limits N/A

Formats <channel> 1 or 2

Related Commands N/A

Description The CALibrate[<channel>]:LINearity command calculates linearity corrections for the preamp section of the Counter hardware. This command requires an input calibration source set to 0.5 V ±0.1%.

[<channel>] specifies the channel being calibrated, and can be specified as either 1 or 2. If [<channel>] is not specified, channel 1 is assumed.

Examples

Command	Response
CALIBRATE1:LINEARITY	N/A, Perform the linearization correction for channel 1
CAL2:LIN	N/A, Perform the linearization correction for channel 2

CALibrate3:BIAS

Command Syntax `CALibrate3:BIAS`

Query Syntax N/A

Query Syntax N/A

***RST Value** N/A

Limits N/A

Formats N/A

Related Commands N/A

Description The CALibrate3:BIAS performs a sensitivity adjustment for the channel 3 input of the card.

Examples

Command	Response
CALIBRATE3:SENSITIV	N/A, Perform the sensitivity adjustment

CALibrate[<channel>]:ZERO

Command Syntax `CALibrate[<channel>]:ZERO`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits N/A

Formats <channel>
1 or 2

Related Commands N/A

Description The CALibrate[<channel>]:ZERO command performs offset corrections for each of the gain and attenuation settings of the Counter. This command requires an input calibration source set to 0.0 V ±0.001 V.

[<channel>] specifies the channel being calibrated, and can be specified as either 1 or 2. If [<channel>] is not specified, channel 1 is assumed.

Examples

Command	Response
CALIBRATE1:ZERO	N/A, Perform the offset calibration for channel 1
CAL2:ZERO	N/A, Perform the offset calibration for channel 2

CALibrate [3] :ROSCillator <input frequency>

Command Syntax `CALibrate:ROSCillator <input frequency>`

Query Syntax N/A

Query Response N/A

Command Class `Instrument (Counter)`

***RST Value** N/A

Limits CAL (defaults to channel 1),
CAL1 (channel 1),
CAL2 (channel 2),
or CAL3 (channel 3)
9 MHz ≤ <input frequency> ≤ 11 MHz

Related Commands SOUR:ROSC[?]
SOUR:ROSC:VAL[?]
CAL:ROSC:MAN
CAL:ROSC:CLE

Description This command is used to calibrate the currently selected reference oscillator using an externally supplied signal with a known frequency and the VX4101 counter. The suffix on the Calibrate keyword specifies to which channel the external signal is connected. The argument specifies the frequency of this signal. After the calibration is complete, the calibrated reference oscillator frequency is stored in nonvolatile memory and will be used for all subsequent measurements.

Examples

Command	Response
SOUR:ROSC INT; SYST:ERR?	-313,"Calibration memory lost; VX4101; Reference Oscillator Uncalibrated"
SOUR:ROSC:VAL?	10000000.0000
INST:SEL COUNTER	
CAL:ROSC	10.001E6
SOUR:ROSC:VAL?	9999999.0000

CONFigure

The CONFigure commands tell the Counter what type of measurement to make and the input channel(s) to use. The measurement won't be made until a INITiate or READ? command is given.

The input coupling and impedance are not changed by these commands. The user must select the coupling and impedance that makes sense for the input signal(s) and the measurement desired. The input attentuation, offset, gain, and comparator hysteresis are not changed by these commands. However, the comparator slopes and thresholds are changed to defaults by these commands. If INPut:SETup:AUTO (autotrigger) is set to ON or ONCE, the input attenuation, offset, gain, and comparator hysteresis may change when the measurement is started. The coupling and impedance are set with the INPut:COUPling and INPut:IMPedance commands.

After a CONFigure command, INPut and SENSe commands may be used to modify the CONFigure setup before the measurement is started by an INITiate or READ? command. These modifications may be undone at INITiate by some types of measurements.

The Counter may be programmed to make and store up to 1000 measurements. The SCALar commands program the Counter for one measurement. The ARRAy commands program the Counter for 1 to 1000 measurements as specified by the <array size> parameter.

CONFigure[1|2][:SCALar|ARRay][:VOLTage][:...]

Command Syntax

```
CONFigure[1|2][:SCALar]
   [:VOLTage]
      :DC [<expected value>[,<resolution>]]
      :MINimum [<expected value>[,<resolution>]]
      :MAXimum [<expected value>[,<resolution>]]
      :PTPeak [<expected value>[,<resolution>]]
CONFigure[1|2]:ARRay
   [:VOLTage]
      :AC <array size>[,<expected value>[,<resolution>]]
      :DC <array size>[,<expected value>[,<resolution>]]
      :MINimum <array size>[,<expected value>[,<resolution>]]
      :MAXimum <array size>[,<expected value>[,<resolution>]]
      :PTPeak <array size>[,<expected value>[,<resolution>]]
```

Query Syntax `CONFigure?`

Query Response N/A

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	ignored
<resolution>	ignored

Related Commands N/A

Description These commands configure the Counter to measure the voltage on the CONFigure suffix input channel. The <expected value> and <resolution> parameters are accepted but ignored.

The maximum and minimum voltage of the input signal are determined assuming either a DC signal or a repetitive signal with a frequency of at least 1000 Hz. The user must select the proper INPut:COUPling and INPut:IMPedance separately. The voltage measurements are calculated as follows:

AC = (maximum – minimum) / 2.828
DC = (maximum + minimum) / 2
MAXimum = maximum
MINimum = minimum
PTPeak = maximum – minimum

These commands will modify the following:

input setup – no effect
function – set to AC, DC, MIN, MAX or PTP
autosetup mode – no effect
aperture/events mode – no effect

The actions of these commands are modified by the following:

<expected value> – no effect
<resolution> – no effect
autosetup mode – no effect
aperture/events mode – no effect

CONFigure[1|2|3][:SCALar|ARRay]:FREQuency

Command Syntax

```
CONFigure[1|2|3][:SCALar]:FREQuency [<expected value>[,<resolu-
tion>]]
```

```
CONFigure[1|2|3]:ARRay:FREQuency <arraysize>[,<expected
value>[,<resolution>]]
```

Query Syntax

N/A

Query Response

N/A

***RST Value**

Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	none
<resolution>	none

Related Commands

N/A

Description

This command configures the Counter to measure the frequency of the signal on the CONFigure suffix input channel. The units of <expected value> and <resolution> are both Hz.

This command will modify the following:

function – set to FREQuency
aperture – described below
aperture/events mode – set to APERture
if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:FREQuency:APERture command.

CONFigure[1|2|3][:SCALar|ARRay]:PERiod

Command Syntax

`CONFigure[1|2|3][:SCALar]:PERiod [<expected value>[,<resolution>]]`

`CONFigure[1|2|3]:ARRay:PERiod <array size>[,<expected value>[,<resolution>]]`

Query Syntax

`CONFigure?`

Query Response

N/A

***RST Value**

Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	none
<Resolution>	none

Related Commands

N/A

Description

This command configures the Counter to measure the period of the signal on the CONFigure suffix input channel. The units of <expected value> and <resolution> are seconds.

This command will modify the following:

function – set to PERiod
aperture – described below
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level, and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log <expected> - \log <resolution>)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:FREQuency:APERture command.

CONFigure[1|2|3][:SCALar|ARRay]:FREQuency:RATio

Command Syntax

```
CONFigure[1|2|3][:SCALar]:FREQuency:RATio <second channel>,[<expected value>[,<resolution>]]
```

```
CONFigure[1|2|3]:ARRay:FREQuency:RATio <array size>,<second channel>[,<expected value>[,<resolution>]]
```

Query Syntax

```
CONFigure?
```

Query Response

N/A

***RST Value**

Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	ignored
<Resolution>	ignored

Related Commands

N/A

Description

This command configures the Counter to measure the ratio of the frequencies of the signals on the CONFigure suffix input channel and <second channel>. <expected value> and <resolution> have no units. The parameters <expected value> and <resolution> are accepted but are not used.

This command will modify the following:

function – set to FREQ:RAT
aperture – set to default
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level, slope can be modified

The SENSe suffix selects the input channel for the numerator, the <second channel> selects the input channel for the denominator. Ratios of a channel to itself are always one and the Counter will generate an error if programmed to do it. <second channel> can be 1, 2 or 3. When channel 3 is not being used and the Counter mode is EVENts, the <second channel> will be used as the input to the EVENts counter. In the case where input channel 3 is used and the counter mode is EVENts, the other specified input channel will be used as the input to the EVENts counter.

CONFigure[1|2][:SCALar|ARRay][:RTIMe|FTIMe]

Command Syntax

```
CONFigure[1|2][:SCALar]

   :RISE:TIME [<low reference>[,<high reference>[,<expected
   value>[,<resolution>]]]]

   :RTIMe [<low reference>[,<high reference>[,<expected
   value>[,<resolution>]]]]

   :FALL:TIME [<low reference>[,<high reference>[,<expected
   value>[,<resolution>]]]]

   :FTIMe [<low reference>[,<high
   reference>[,<expectedvalue>[,<resolution>]]]]

CONFigure[1|2]:[ARRay]

   :RISE:TIME <array size>[,<low reference>[,<high
   reference>[,<expected value>[,<resolution>]]]]

   :RTIMe <array size>[,<low reference>[,<high
   reference>[,<expected value>[,<resolution>]]]]

   :FALL:TIME <array size>[,<low reference>[,<high
   reference>[,<expected value>[,<resolution>]]]]

   :FTIMe <array size>[,<low reference>[,<high
   reference>[,<expected value>[,<resolution>]]]]
```

Query Syntax `CONFigure?`

Query Response N/A

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<low reference>	10 to 90
<high reference>	10 to 90
<Expected Value>	none
<Resolution>	none

Related Commands

N/A

Description

This command configures the Counter to measure the rise or fall time of the signal on the CONFigure suffix input channel. The units of the <low reference> and <high reference> are percentage. The units of <expected value> and <resolution> are seconds. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected rise/fall time. This measurement uses both comparators of the input channel.

If a reference value is specified and INPut:SETup:AUTO is set to OFF, INPut:SETup:AUTO will be set to ONCE. If a reference value is not specified, the input channel comparator 1 level will be set to –.25 V and input channel comparator 2 level will be set to +.25 V for a rise time measurement. For fall time, the reverse is set.

This command will modify the following:

function – set to RTIMe or FTIMe
aperture – described below
aperture/events mode – set to APERture

The input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:RTIMe:APERture command.

CONFigure[1|2][:SCALar|ARRay]:NWIDth|PWIDth

Command Syntax

```
CONFigure[1|2][:SCALar]

   :NWIDth [<reference>[,<expected value>[,<resolution>]]]

   :PWIDth [<reference>[,<expected value>[,<resolution>]]]

CONFigure[1|2]:[ARRay]

   :NWIDth <array size>[<reference>[,<expected value>
   [,<resolution>]]]

   :PWIDth <array size>[<reference>[,<expected value>
   [,<resolution>]]]
```

Query Syntax `CONFigure?`

Query Response N/A

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<reference>	10 to 90
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description

This command configures the Counter to measure the positive or negative pulse width time of the signal on the CONFigure suffix input channel. The units of the <reference> are percentage. The units of <expected value> and <resolution> are seconds. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected pulse width. This measurement uses both comparators of the input channel. The reference value is used to set the comparator threshold levels to a percentage of the peak-to-peak signal.

If a reference value is specified and INPut:SETup:AUTO is set to OFF, INPut:SETup:AUTO will be set to ONCE. If a reference value is not specified, the input channel comparators 1 and 2 levels will be set to 0 V.

This command will modify the following:

function – set to PWID or NWID
aperture – described below
aperture/events mode – set to APERture

The input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:PWIDth:APERture command.

CONFigure[1|2][:SCALar|ARRay]:TINTerval

Command Syntax

```
CONFigure[1|2][:SCALar]:TINTerval [<expected value>[,<resolu-
tion>]]
```

```
CONFigure[1|2]:ARRay:TINTerval <array size>[,<expected value>
[,<resolution>]]
```

Query Syntax

```
CONFigure?
```

Query Response N/A

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	none
<Resolution>	none

Description This command sets the Counter to make a time interval measurement. This measurement is made between input channels 1 and 2. The CONFigure suffix selects the input channel for the beginning of the interval. The end of the interval will be from the remaining channel. The measurement is made from the first detected rising edge on the first channel to the first following rising edge on the second channel. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected time interval.

This command will modify the following:

function – set to TINT
aperture – described below
aperture/events mode – set to APERture

If autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log <expected> - \log <resolution>)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:TINTerval:APERture command.

CONFigure[1|2][:SCALar|ARRay]:TINTerval:DELay

NOTE. *Time Interval With Delay uses all available hardware timers. When the counter is in this mode, the aperture is controlled by software and has a minimum period of approximately 10 ms.*

Command Syntax

```
CONFigure[1|2][:SCALar]:TINTerval:DELay <delay>[,<expected value>
[,<resolution>]]
```

```
CONFigure[1|2]:ARRay:TINTerval:DELay <array size>,<delay>
[,<expected value>[,<resolution>]]
```

Query Syntax

```
CONFigure?
```

Query Response N/A

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<delay>	1e–9 to 9e+6
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description This command sets the Counter to make a time interval measurement with a delay. This measurement is made between input channels 1 and 2. The CONFigure suffix selects the input channel for the beginning of the interval, the end of the interval will be from the remaining channel. The measurement is made from the first detected rising edge on the first channel to the first following rising edge after the delay time on the second channel. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected time interval.

This command will modify the following:

function – set to TINT
aperture – described below
aperture/events mode – set to APERture

If autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log <expected> - \log <resolution>)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:TINTerval:APERture command.

CONFigure[1|2][:SCALar]:TOTalize

Command Syntax `CONFigure[1|2][:SCALar]:TOTalize`

Query Syntax `CONFigure?`

Query Response N/A

***RST Value** Default configuration

Limits N/A

Related Commands N/A

Description This command configures the Counter to count events on both input channels 1 and 2. This measurement is unique. It allows you to fetch the totals while it is still counting. Each fetch will return updated count values. This command has an aperture of up to 99 days. Use the ABORt command to end this measurement. Since this measurement allows reading the Counter hardware while it is counting, there is some risk that the value returned will be in error.

FETCh?

The FETCh? commands, except FETCh:COUNt?, cause the instrument to put the results of a measurement, or array of measurements, in the output buffer to be read back by the user. If a measurement or array of measurements haven't completed, the VX4101 will delay the read until the measurement(s) have completed.

The results of some measurements can be returned as a different measurement, for instance frequency and period. If a frequency measurement was made, it can be also be read back as a period by the command FETC:PER?

If the measurement was an array measurement specified with commands such as MEAS:ARR:func or CONF:ARR:func or SENS:func:COUN, the optional parameters <count>, <start> and <step> apply.

The default value of <count> is the number of measurements requested for the array. The default value of <start> is 1. The default value of <step> is 1. The number of values returned is <count>. The first of the values returned is value <start>. Then every <step> values after start are returned.

For instance, the command FETC? 4,3,2 will return four values starting with value three and stepping 2 values. The values returned would be 3, 5, 7, and 9.

If the range of requested values is beyond the number of measurements, an error will be generated and no values returned.

The format of the returned values varies with the measurement function and whether it was an array measurement or not.

Command Syntax N/A

Query Syntax `FETCh? [<count>[,<start>[,<step>]]]`

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Sets up to return the results of the previous

measurement. See the FETCh?:func command descriptions below for the format of the output data.

The default value of <count> is the number of measurements requested for the array. The default value of <start> is 1. The default value of <step> is 1. The number of values returned is <count>. The first of the values returned is value <start>. Then every <step> values after start are returned.

FETCh[:VOLTage][:...]?

Command Syntax N/A

Query Syntax

```
FETCh[:VOLTage]
    :AC? [<count>[,<start>[,<step>]]]
    :DC? [<count>[,<start>[,<step>]]]
    :MINimum? [<count>[,<start>[,<step>]]]
    :MAXimum? [<count>[,<start>[,<step>]]]
    :PTPeak? [<count>[,<start>[,<step>]]]
```

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Sets up to return the results of the previous voltage

measurement. If a single voltage measurement has been made, any of the other voltage function results may be fetched. If an array of voltage measurements have been made, only the results of the requested voltage measurement function may be fetched.

The default value of <count> is the number of measurements requested for the array. The default value of <start> is 1. The default value of <step> is 1. The number of values returned is <count>. The first of the values returned is value <start>. Then every <step> values after start are returned.

If a single measurement is made, the format of the returned voltage is:

n.nnnn<lf>

If an array measurement is made, the format of the returned values is:

#abbrn.nnnnesxx,rn.nnnnesxx, ... ,rn.nnnnesxx<lf>

where

a = number of b digits
b = number of characters in the returned data not including #abb
r = – or space
n = value
s = + or –
xx = exponent

for example:

meas:arr:DC? 10

#3119 2.5712e+01, 1.5392e+02,–1.0252e+02, 2.5642e+02,–9.7442e+01, 3.0777e+01,–2.2567e+02, 2.5647e+02,–9.7447e+01, 3.0777e+01

FETCh[:...]?

Command Syntax N/A

Query Syntax

```
FETCh
    :PERiod? [<count>[,<start>[,<step>]]]
    :FREQuency? [<count>[,<start>[,<step>]]]
    :FREQuency:RATio? [<count>[,<start>[,<step>]]]
    :RISE:TIME? [<count>[,<start>[,<step>]]]
    :RTIMe? [<count>[,<start>[,<step>]]]
    :FALL:TIME? [<count>[,<start>[,<step>]]]
    :FTIMe? [<count>[,<start>[,<step>]]]
    :NWIDth? [<count>[,<start>[,<step>]]]
    :PWIDth? [<count>[,<start>[,<step>]]]
    :TINTerval? [<count>[,<start>[,<step>]]]
```

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description If a single measurement is made, the format of the returned value is:

n.nnnnnnnnnnnnnnnesxx<lf>

If an array measurement is made, the format of the returned values is:

#abbrn.nnnnnnnnnnnnnnnesxx,rn.nnnnnnnnnnnnnnnesxx, ... ,
rn.nnnnnnnnnnnnnnnesxx<lf>

where

a = number of b digits
b = number of characters in the returned data not including #abb
r = space or –
n = value
s = + or –
xx = exponent

The default value of <count> is the number of measurements requested for the array. The default value of <start> is 1. The default value of <step> is 1. The number of values returned is <count>. The first of the values returned is value <start>. Then every <step> values after start are returned.

Examples

Command	Response
meas:arr:freq? 5	#3114 1.000028872529214e+07, 1.000028882397919e+07, 1.000028897200977e+07, 1.000028892266624e+07, 1.000028879930743e+07

FETCh:TOTalize?

Command Syntax N/A

Query Syntax `FETCh:TOTalize?`

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description The totalize measurement returns two integers of up to 15 digits. The first integer is the total counts from input channel 1 and the second integer is the total counts from input channel 2.

Examples

Command	Response
meas:tot?	“9179, 0”
fetc?	“62710055, 0”
fetc?	“72161385, 0”
fetc?	“81704262, 0”
fetc?	“100501585, 0”
fetc?	“112160067, 0”
abort	

FETCh:COUNt?

Command Syntax N/A

Query Syntax `FETCh:COUNt?`

Query Response Returns the number of measurements completed. This command may be used to monitor the progress of a measurement or an array of measurements.

***RST Value** N/A

Limits N/A

Related Commands N/A

Description N/A

***CAUTION.** Do not continuously issue FETCh?COUNt or STAT:OPER:COND? queries to check if the measurement is complete. Provide a minimum .100 ms delay between each query to permit the multitasking system to process the measurement efficiently or use *SRQ to avoid processing delays caused by polling.*

INITiate[:IMMediate|CONTinuous]

Command Syntax `INITiate[:IMMediate]`

`INITiate:CONTinuous`

Query Syntax `INITiate:CONTinuous?`

Query Response Returns a 0 if the Counter is not taking continuous measurements, a 1 if it is.

***RST Value** N/A

Limits N/A

Related Commands `READ, MEASure, ARM`

Description These commands initiate the measurement the Counter is currently selected for. If the CONTinuous form is used, measurements are continuously made until a *RST or ABORt command is used. While the Counter is continuously making measurements, the FETCh? command may be used to return the results of the most recently completed measurement.

INPut

This subsystem controls the characteristics of the instrument's input channels.

INPut[1|2]:OFFSet[:ABSolute]

Command Syntax `INPut[1|2]:OFFSet[:ABSolute]<offset>|MINimum|MAXimum|DEFault`

Query Syntax `INPut[1|2]:OFFSet[:ABSolute]? [MINimum|MAXimum|DEFault]`

Query Response Without one of the optional arguments, this command moves to the output buffer the current setting in volts of the offset. If one of the optional arguments is used, the MINimum, MAXimum or DEFault value for offset is moved to the output buffer instead.

If the offset was set with the INPut:OFFSet:RELative command, the value returned will be the voltage the offset will be set to at INITiate time.

***RST Value** 0

Limits <offset> –1.0 to +1.0

Related Commands INPut:OFFSet:RELative

Description This command sets the offset voltage for the channel specified by the INPut suffix. The units are volts. The offset voltage is subtracted from the signal after the input attenuator and before the input gain.

DEFault = 0
MINimum = –1.0
MAXimum = 1.0

This command will abort any command in progress.

Examples

The channel 1 input signal is a 0.5 V_{p-p} sine wave with a +.25 VDC component and the user wishes to remove the DC component with the INPut:OFFset command.

*RST
INSTrument:SELect Counter
INPut:SETup:AUTo off
SENSe:FUNCtion “FREQuency”
INPut:COUPling DC
INPut:OFFSet .25

The channel 2 input signal is a 3 V_{p-p} sine wave with a –4 VDC component and the user wishes to remove the DC component with the INPut:OFFset command.

*RST
INSTrument:SELect Counter
INPut:SETup:AUTo off
SENSe2:FUNCtion “FREQuency”
INPut2:COUPling DC
INPut2:ATTenuator 10
INPut2:OFFSet –.4

***NOTE**. The input signal –4 VDC component has been divided by the attenuator setting of 10 requiring an offset setting of –0.4 instead of –4)*

INPut[1|2]:OFFSet:RELative

Command Syntax

`INPut[1|2]:OFFSet:RELative <offset>`

Query Syntax

`INPut[1|2]:OFFSet:RELative?`

Query Response

This query moves to the output buffer the input channel offset voltage corrected for the attenuation.

***RST Value**

N/A

Limits <offset> –100 to +100

Related Commands INPut[1|2]:OFFSet[:ABSolute]

Description This command sets the offset voltage for the channel specified by the INPut suffix. This value will be used at INITiate to set the offset to a voltage that is relative to the input.

Examples The channel 1 input signal is a 0.5 V_{p-p} sine wave with a +0.25 VDC component. The user wishes to remove the DC component with the INPut:OFFset:RELative command.

```
*RST
INSTrument:SELect Counter
INPut:SETup:AUTo off
SENSe:FUNCtion “FREQuency”
INPut:COUPling DC
INPut:OFFSet:RELative .25
```

The channel 2 input signal is a 3 V_{p-p} sine wave with a –4 VDC component. The user wishes to remove the DC component with the INPut:OFFset:RELative command. Note the use of the attenuator command to reduce the signal level to acceptable levels inside the input block.

```
*RST
INSTrument:SELect Counter
INPut:SETup:AUTo off
CONFigure:FREQuency
INPut2:COUPling DC
INPut2:ATTenuator 10
INPut2:OFFSet:RELative –4
```

***NOTE**. The input signal –4 VDC component will be divided by the attenuator setting of 10. However, the INPut:OFFSet:RELative command takes this into account. The actual offset hardware is set to –0.4 V.*

INPut[1|2]:COUPling

Command Syntax `INPut[1|2]:COUPling AC|DC|DEFault`

Query Syntax `INPut[1|2]:COUPling? [DEFault]`

Query Response

Without the optional parameter, this command moves to the output-buffer the current setting of the input block signal *coupling* for the specified *channel*. If the optional argument DEFault is included, the default coupling is moved to the output-buffer. The possible values returned are “AC” and “DC”.

***RST Value**

AC

Limits

N/A

Related Commands

N/A

Description

This command sets the input block signal coupling for the specified channel.

DEFault = AC

INPut[1|2]:IMPedance

Command Syntax

`INPut[1|2]:IMPedance <numeric_value>|DEFault`

Query Syntax

`INPut[1|2]:IMPedance? [DEFault]`

Query Response

Without the optional parameter, this command moves to the output-buffer the current setting of the input terminating impedance for the specified channel. If the optional argument DEFault is included, the default impedance is moved to the output-buffer. The possible values returned are 50 Ω or 1E6 Ω.

***RST Value**

1E6

Limits

N/A

Related Commands

N/A

Description

This command sets the input terminating impedance for the specified channel. If <numeric_value> is less than 60, the impedance is set to 50 Ω. If <numeric_value> is greater than 60, the impedance is set to 1E6 Ω (1 MΩ).

DEFault = 1E6

INPut[1|2]:FILTer[:LPASs]:FREQuency

Command Syntax `INPut[1|2]:FILTer[:LPASs]:FREQuency <numeric_value>|MINimum|MAXimum|DEFault`

Query Syntax `INPut[1|2]:FILTer[:LPASs]:FREQuency? [MINimum|MAXimum|DEFault]`

Query Response Without one of the optional arguments, this command moves to the output buffer the current setting in Hz of the input lowpass filter. If one of the optional arguments is included, the MINimum, MAXimum or DEFault value for filter is moved to the output-buffer instead.

***RST Value** 20e6

Limits N/A

Related Commands INPut[1|2]:FILTer[:LPASs][:STATe]

Description This command sets the input block signal lowpass filter for the selected channel. If <numeric_value> is less than 30E6, the filter is set to 20E6 Hz. If <numeric_value> is greater than 30E6, the filter is set to 100E6 Hz.

***NOTE**. This setting has no effect on the signal unless the INPut<channel>:FILTer[:LPASs][:STATe] ON command is given. The units for <numeric_value> are in Hertz.*

DEFault = 20E6
MINimum = 20E6
MAXimum = 100E6

INPut[1|2]:FILTer[:LPASs][:STATe]

Command Syntax `INPut[1|2]:FILTer[:LPASs][:STATe] <Boolean>|DEFault`

Query Syntax `INPut[1|2]:FILTer[:LPASs][:STATe]? [DEFault]`

Query Response

Without the optional parameter, this command moves to the output buffer the current setting of the input lowpass filter state for the specified channel. If the optional argument DEFault is included, the default filter state is moved to the output_buffer. The possible values returned are 1 for ON and 0 for OFF.

***RST Value**

OFF

Limits

N/A

Related Commands

N/A

Description

This command sets the input block signal lowpass filter state for the selected channel to ON or OFF.

DEFault = OFF

INPut[1|2]:ATTenuation

Command Syntax

`INPut[1|2]:ATTenuation <numeric_value>|DEFault|MINimum|MAXimum`

Query Syntax

`INPut[1|2]:ATTenuation? [DEFault|MINimum|MAXimum]`

Query Response

Without one of the optional parameters, this command moves to the output-buffer the current setting of the input block signal attenuator for the specified channel. If one of the optional parameters is used, the default, minimum or maximum value for attenuation is moved to the output-buffer. The possible values returned are 1, 10 or 100.

***RST Value**

1

Limits

N/A

Related Commands

N/A

Description

This command sets the input block signal attenuator for the specified channel. If <numeric_value> is less than 2, the attenuator is set to 1. If <numeric_value> is between 2 and 20, the attenuator is set to 10. If <numeric_value> is greater than 20, the attenuator is set to 100.

DEFault = 1
MINimum = 1
MAXimum = 100

INPut[1|2]:GAIN

Command Syntax `INPut[1|2]:GAIN <numeric value>|DEFault|MINimum|MAXimum`

Query Syntax `INPut[1|2]:GAIN? [DEFault|MINimum|MAXimum]`

Query Response Without one of the optional parameters, this command moves to the output buffer the current setting of the input gain for the specified channel. If the one optional parameters is used, the DEFault, MINimum or MAXimum value for gain is moved to the output buffer.

***RST Value** 1

Limits <numeric value> 0.4 to 10.0

Related Commands N/A

Description This command sets the input block signal gain for the specified channel.

DEFault = 1.0
MINimum = 0.4
MAXimum = 10.0

***NOTE**. When using the X10 gain, the 20 MHz filter should be turned on. This combination provides the best sensitivity for measurements below 20 MHz. Also, when using the X5 gain, the 100 MHz filter should be turned on. This combination provides the best sensitivity for measurements up to 100 MHz. The X1 and X2.5 gains can be used up the full 500 MHz bandwidth of the product.*

Table 3–17 shows the settings for optimum sensitivity.

Table 3–17: Optimum Sensitivity Settings

Measurement	Gain	Filter
<20 MHz	X10	20 MHz
<100 MHz	X5	100 MHz
<500 MHz	X2.5	None

INPut[1|2]:COMParator[1|2]:LEVel[:ABSolute]

Command Syntax

```
INPut[1|2]:COMParator[1|2]:LEVel[:ABSolute]<numeric value>|DEFault|MINimum|MAXimum
```

Query Syntax

```
INPut[1|2]:COMParator[1|2]:LEVel[:ABSolute]?[DEFault|MINimum|MAXimum]
```

Query Response

Without one of the optional arguments, this command moves to the output buffer the current setting in volts of the threshold level of the currently selected channel and comparator. If one of the optional arguments is included, the MINimum, MAXimum or DEFault value for level is moved to the output buffer instead. The value returned is calculated as follows:

VL = S*G*(VLR/A + VO)

***RST Value**

0

Limits

<numeric value> –0.5 to .5

Related Commands

INPut:COMParator:LEVel:RELative

Description

This command sets the threshold level for the input channel and comparator selected. The units are volts.

DEFault = 0
MINimum = –0.5
MAXimum = 0.5

To calculate what a particular input voltage will be at the comparator, the following equation may be used:

VL = S*G*(VO + VI/A)

where

VL = voltage at the comparator
S = 1 for positive slope, –1 for negative slope
G = gain setting
VO = offset voltage
VI = input voltage
A = attenuator setting

Examples The channel 1 input signal varies between +0.5 and –0.5 V and the user wishes to cause the comparator to trigger at 0.25 V.

```
*RST
INSTrument:SELect Counter
INPut:SETup:AUTo off
SENSe:FUNCtion “FREQuency”
INPut:COUPling DC
INPut:COMParator:LEVel .25
```

The channel 2 input signal goes between –3 V and +3 V and the user wishes to cause the comparator to trigger at –1 V relative to the input signal.

```
*RST
INSTrument:SELect Counter
INPut:SETup:AUTo off
SENSe2:FUNCtion “FREQuency”
INPut2:COUPling DC
INPut2:ATTenuator 10
INPut2:OFFSet –.1
```

(note that the input signal has been divided by the attenuator setting of 10 requiring a level setting of –0.1 instead of –1)

INPut[1|2]:COMParator[1|2]:LEVel:RELative

Command Syntax `INPut[1|2]:COMParator[1|2]:LEVel:RELative <numeric_value>`

Query Syntax `INPut[1|2]:COMParator[1|2]:LEVel:RELative?`

Query Response The threshold level corrected for attenuation, slope, gain and offset of the channel and comparator selected is moved to the output buffer. If the level was not previously set by the relative command, the value returned is calculated by the formula:

VLR = A((S*VL)/G – VO)

where

VLR = comparator trigger level relative to the input
A = current attenuator setting
S = slope, 1 for positive, –1 for negative
VL = comparator level previously set
G = current gain setting
VO = current offset setting (converted from a relative offset if necessary)

***RST Value** N/A

Limits N/A

Related Commands INPut[1|2]:COMParator[1|2]:LEVel[:ABSolute]

Description This command sets the comparator threshold level voltage of the channel and comparator selected. This value will be used at INITiate time to set the comparator level to a voltage that is the same as if the voltage was fed to the input. This compensates for attenuation, offset and gain settings. The unit for <numeric value> is volts. The comparator level hardware will be set when the measurement is INITiated. The following formula will be used:

VL = S*G*(VLR/A + VO)

where

VL = comparator level voltage
S = slope, 1 for positive, –1 for negative
G = current gain setting
VLR = relative comparator level previously set
A = current attenuator setting
VO = current offset setting (converted from a relative offset if necessary)

INPut[1|2]:COMParator[1|2]:HYSTeresis[:ABSolute]

Command Syntax `INPut[1|2]:COMParator[1|2]:HYSTeresis[:ABSolute] <numeric_value>|DEFault|MINimum|MAXimum`

Query Syntax `INPut[1|2]:COMParator[1|2]:HYSTeresis[:ABSolute]?[DEFault|MINimum|MAXimum]`

Query Response

Without one of the optional arguments, this command moves to the output-buffer the current setting in volts of the hysteresis of the currently selected channel and comparator. If one of the optional arguments is included, the MINimum, MAXimum or DEFault value for hysteresis is moved to the output-buffer instead.

If the hysteresis was previously set with the relative hysteresis command, the query will calculate the hysteresis voltage to put in the output buffer by the formula:

VH = G(VHR/A)

where

VH = comparator hysteresis that will be output for query
G = current gain setting
VHR = relative comparator hysteresis voltage previously set
A = current attenuator setting

***RST Value**

.06

Limits

N/A

Related Commands

INPut[1|2]:COMParator[1|2]:HYSTeresis:RELative <numeric_value>

Description

This command sets the hysteresis of the channel and comparator selected. The units are in volts. If a <numeric_value> is specified, it is rounded to the nearest of the values 0.01, 0.02, 0.03, 0.04, 0.05 and 06.

DEFault = .06
MINimum = .01
MAXimum = .06

INPut[1|2]:COMParator[1|2]:HYSTeresis:RELative

Command Syntax

`INPut[1|2]:COMParator[1|2]:HYSTeresis:RELative <numeric_value>`

Query Syntax

`INPut[1|2]:COMParator[1|2]:HYSTeresis:RELative?`

Query Response

This query places in the output buffer the selected input channel and comparator hysteresis setting in volts. If the hysteresis was not set by the relative command, the value returned is calculated by the formula:

VHR = A(VH/G)

where

VHR = comparator hysteresis relative to the input
A = current attenuator setting
VH = comparator hysteresis previously set
G = current gain setting

***RST Value** N/A

Limits N/A

Related Commands INPut[1|2]:COMParator[1|2]:HYSTeresis[:ABSolute]

Description This command sets the hysteresis voltage for the selected input channel and comparator. This value will be used at INITiate time to set the comparator hsyteresis to a voltage that is the same as if the hysteresis voltage was implemented at the input. This compensates for attenuation and gain settings. The unit for <numeric value> is volts. The comparator hysteresis hardware will be set when the measurment is INITiated. The following formula will be used:

VH = G(VHR/A)

where

VH = comparator hysteresis set at INITiate time
G = current gain setting
VHR = relative comparator hsyteresis set here
A = current attenuator setting

INPut[1|2]:COMParator[1|2]:SLOPe

Command Syntax `INPut[1|2]:COMParator[1|2]:SLOPe POSitive|NEGative|DEFault`

Query Syntax `INPut[1|2]:COMParator[1|2]:SLOPe? [DEFault]`

Query Response Without one of the optional parameters, this command moves to the output buffer the current setting of the slope of the selected input channel and comparator. If the optional parameter is used the default value for slope is moved to the output buffer.

***RST Value** POSitive

Limits N/A

Related Commands N/A

Description This command sets the slope for the selected input channel and comparator. The slope may be either positive or negative.

***NOTE**. This command inverts the input signal at the comparator and the comparator threshold level will need to reflect this.*

INPut[1|2]:SETup

Command Syntax `INPut[1|2]:SETup <expected PTP>[,<expected offset>]`

Query Syntax N/A

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This command allows a user to set up input channel 1 or 2 by specifying an expected peak-to-peak input voltage and, optionally, an expected input offset voltage. The Counter will set the input channel attenuation, offset and gain to settings that would center the expected signal in 80% of the comparator range. Both comparators’ slope is set to POSitive, level to 0 V and hysteresis to MAXimum.

INPut:SETup:AUTO

Command Syntax `INPut:SETup:AUTO ON|OFF|ONCE`

Query Syntax `INPut:SETup:AUTO? [DEFault]`

Query Response This command moves to the output buffer the current setting of autosetup. The returned value will be 0 for OFF, 1 for ON and ONCE for ONCE.

***RST Value** OFF

Limits N/A

Related Commands N/A

Description This command controls the autosetup of input channels 1 and 2. If it is set ON, each measurement will be proceeded by an automatic setup of the input channel(s) involved in the measurement. If it is set OFF, the current settings are used for the measurement. For ARRay measurements, the ONCE setting will cause an autosetup to occur only for the first measurement. For SCALar measurements, ONCE will cause an autosetup with each measurement.

An autosetup adjusts the input channel’s attenuation, offset and gain so that the input signal’s peak-to-peak voltage into the comparator is centered on about 80% of the range of the comparator.

INPut commands which manually set the input channel hardware other than COUPling, IMPedance and FILTering will set autosetup to OFF.

The time a measurement is taken is influenced by the ARM subsystem.

INSTrument:RESet

Command Syntax `INSTrument:RESet`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This command resets the currently selected instrument without affecting other instruments. The instrument returns to its *RST state. The instrument remains selected.

Examples

Command	Response
INST:SEL Counter	
CONF:FREQ	
CONF?	"CONF1:SCAL:FREQ"
INST:RES	
INST:SEL?	Counter
CONF?	"CONF1:SCAL:VOLT:DC"

MEASure

The MEASure queries tell the Counter what type of measurement to make and the input channel(s) to use. It also INITiates a measurement and moves to the output buffer the results of the measurement when it completes.

The input coupling and impedance are not changed by these commands. The user must select the coupling and impedance that makes sense for the input signal(s) and the measurement desired. The coupling and impedance are set with the INPut:COUPling and INPut:IMPedance commands. The input, attenuation, offset, gain, and comparator hysteresis are not changed by these commands. However, the comparator slopes and thresholds are changed to defaults by these commands. If INPut:SETup:AUTO (autotrigger) is set to ON or ONCE, the input attenuation, offset, gain, and comparator hysteresis may change when the measurement is started.

The Counter may be programmed to make and store up to 1000 measurements. The SCALar commands program the Counter for one measurement. The ARRAy commands program the Counter for 1 to 1000 measurements as specified by the <array size> parameter.

MEASure[1|2][:SCALar|ARRay][:VOLTage][:...]?

Command Syntax N/A

Query Syntax

```
MEASure[1|2][:SCALar]
    [:VOLTage]
        :AC? [<expected value>[,<resolution>]]
        :DC? [<expected value>[,<resolution>]]
        :MINimum? [<expected value>[,<resolution>]]
        :MAXimum? [<expected value>[,<resolution>]]
        :PTPeak? [<expected value>[,<resolution>]]
MEASure[1|2]:ARRay
    [:VOLTage]
        :AC? <array size>[,<expected value>[,<resolution>]]
        :DC? <array size>[,<expected value>[,<resolution>]]
        :MINimum? <array size>[,<expected value>[,<resolution>]]
        :MAXimum? <array size>[,<expected value>[,<resolution>]]
        :PTPeak? <array size>[,<expected value>[,<resolution>]]
```

Query Response See the FETCh? commands.

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	ignored
<resolution>	ignored

Related Commands N/A

Description

These commands configure the Counter to measure the voltage on the MEASure suffix input channel. The measurement is initiated and the result placed in the output buffer. The <expected value> and <resolution> parameters are accepted but ignored.

The maximum and minimum voltage of the input signal are determined assuming either a DC signal or a repetitive signal with a frequency of at least 1000 Hz. The user must select the proper INPut:COUPling and INPut:IMPedance separately. The voltage measurements are calculated as follows:

AC = (maximum – minimum) / 2.828
DC = (maximum + minimum) / 2
MAXimum = maximum
MINimum = minimum
PTPeak = maximum – minimum

These commands can or will modify the following:

function – set to AC, DC, MIN, MAX or PTP

autosetup mode – no effect
aperture/events mode – no effect

MEASure[1|2|3][:SCALar|ARRay]:FREQuency?

Command Syntax N/A

Query Syntax

```
MEASure[1|2|3][:SCALar]:FREQuency? [<expected value>[,<resolu-
tion>]]
```

```
MEASure[1|2|3]:ARRay:FREQuency? <array size>[,<expected value>
[,<resolution>]]
```

Query Response See the FETCh? commands

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	none
<resolution>	none

Related Commands N/A

Description This command configures the Counter to measure the frequency of the signal on the MEASure suffix input channel. The measurement is initiated and the result placed in the output buffer. The units of <expected value> and <resolution> are both Hertz.

This command will modify the following:

function – set to FREQuency
aperture – described below
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log <expected> - \log <resolution>)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:FREQuency:APERture command.

MEASure[1|2|3][:SCALar|ARRay]:PERiod?

Command Syntax N/A

Query Syntax

```
MEASure[1|2|3][:SCALar]:PERiod? [<expected value>[,<resolution>]]
```

```
MEASure[1|2|3]:ARRay:PERiod? <array size>[,<expected value>
[,<resolution>]]
```

Query Response See the FETCh? commands.

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description This command configures the Counter to measure the period of the signal on the MEASure suffix input channel. The measurement is initiated and the result placed in the output buffer. The units of <expected value> and <resolution> are seconds.

This command will modify the following:

function – set to PERiod
aperture – described below
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:FREQuency:APERture command.

MEASure[1|2|3][:SCALar|ARRay]:FREQuency:RATio?

Command Syntax N/A

Query Syntax

```
MEASure[1|2|3][:SCALar]:FREQuency:RATio? <second channel>,[<expected value>[,<resolution>]]
```

```
MEASure[1|2|3]:ARRay:FREQuency:RATio? <array size>,<second channel>[,<expected value>[,<resolution>]]
```

Query Response See the FETCh? commands.

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	ignored
<Resolution>	ignored

Related Commands N/A

Description

This command configures the Counter to measure the ratio of the frequencies of the signals on the MEASure suffix input channel and <second channel>. The measurement is initiated and the result placed in the output buffer. Ratio, <expected value>, and <resolution> have no units. The parameters <expected value> and <resolution> are accepted but are not used.

This command will modify the following:

function – set to FREQ:RAT
aperture – set to default
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

The SENSe suffix selects the input channel for the numerator. The <second channel> selects the input channel for the denominator. Ratios of a channel to itself are always one and the Counter will generate an error if programmed to do it. The <second channel> can be 1, 2 or 3. When channel 3 is not being used and the Counter mode is EVENts, the SENSe suffix channel will be used as the input to the EVENts Counter. In the case where input channel 3 is used and the Counter mode is EVENts, the other specified input channel will be used as the input to the EVENts Counter.

MEASure[1|2][:SCALar|ARRay]:RTIMe|FTIMe?

Command Syntax

N/A

Query Syntax

```
MEASure[1|2][:SCALar]

    :RISE:TIME? [<low reference>[,<high reference>[,<expected
    value>[,<resolution>]]]]

    :RTIMe? [<low reference>[,<high reference>
    [,<expectedvalue>[,<resolution>]]]]

    :FALL:TIME? [<low reference>[,<high reference>[,<expected
    value>[,<resolution>]]]]

    :FTIMe? [<low reference>[,<high reference>[,<expected value>
    [,<resolution>]]]]

MEASure[1|2]:ARRay

    :RISE:TIME? <array size>[,<low reference>[,<high reference>
    [,<expected value>[,<resolution>]]]]
```

```
:RTIMe? <array size>[,<low reference>[,<high reference>
[,<expected value>[,<resolution>]]]]

:FALL:TIME? <array size>[,<low reference>[,<high reference>
[,<expected value>[,<resolution>]]]]

:FTIMe? <array size>[,<low reference>[,<high reference>
[,<expected value>[,<resolution>]]]]
```

Query Response See the FETCh? commands

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<low reference>	10 to 90
<high reference>	10 to 90
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description This command configures the Counter to measure the rise or fall time of the signal on the MEASure suffix input channel. The measurement is initiated and the result placed in the output buffer. The units of the <low reference> and <high reference> are percentage. The units of <expected value> and <resolution> are seconds. If the <expected value> argument is used, the expected period of the input signal should be used, instead of the expected rise/fall time. This measurement uses both comparators of the input channel.

If a reference value is specified and INPut:SETup:AUTO is set to OFF, INPut:SETup:AUTO will be set to ONCE. If a reference value is not specified, the input channel comparator 1 level will be set to –.25 V and input channel comparator 2 level will be set to +.25 V for a rise time measurement. For fall time, the reverse is set.

This command will modify the following:

function – set to RTIMe or FTIMe
aperture – described below
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:RTIMe:APERture command.

MEASure[1|2][:SCALar|ARRay]:NWIDth|PWIDth?

Command Syntax N/A

Query Syntax

```
MEASure[1|2][:SCALar]

    :NWIDth? [<reference>[,<expected value>[,<resolution>]]]
    :PWIDth? [<reference>[,<expected value>[,<resolution>]]]

MEASure[1|2]:ARRay

    :NWIDth? <array size>[<reference>[,<expected value>
    [,<resolution>]]]

    :PWIDth? <array size>[<reference>[,<expected value>
    [,<resolution>]]]
```

Query Response See the FETCh? commands.

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<reference>	10 to 90
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description

This command configures the Counter to measure the positive or negative pulse width time of the signal on the MEASure suffix input channel. The measurement is initiated and the result placed in the output buffer. The units of the <reference> are percentage. The units of <expected value> and <resolution> are seconds. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected pulse width. This measurement uses both comparators of the input channel. The reference value is used to set the comparator threshold levels to a percentage of the peak-to-peak signal.

If a reference value is specified and INPut:SETup:AUTO is set to OFF, INPut:SETup:AUTO will be set to ONCE. If a reference value is not specified, the input channel comparators 1 and 2 levels will be set to 0 V.

This command will modify the following:

function – set to PWID or NWID
aperture – described below
aperture/events mode – set to APERture

The input channel attenuation, offset, gain, level and slope can be modified
If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:PWIDth:APERture command.

MEASure[1|2][:SCALar|ARRay]:TINTerval?

Command Syntax

N/A

Query Syntax

```
MEASure[1|2][:SCALar]:TINTerval? [<expected value>[,<resolu-
tion>]]
```

```
MEASure[1|2]:ARRay:TINTerval? <array size>[,<expected value>
[,<resolution>]]
```

Query Response

See the FETCh? commands.

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description This command sets the Counter to make a time interval measurement and fetch the result. This measurement is made between input channels 1 and 2. The MEASure suffix selects the input channel for the beginning of the interval, the end of the interval will be from the remaining channel. The measurement is made from the first detected rising edge on the first channel to the first following rising edge on the second channel. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected time interval.

This command will modify the following:

function – set to TINT
aperture – described below
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:TINTerval:APERture command.

MEASure[1|2][:SCALar|ARRay]:TINTerval:DELay?

***NOTE**. Time Interval With Delay uses all available hardware timers. When the counter is in this mode, the aperture is controlled by software and has a minimum period of approximately 10 ms.*

Command Syntax N/A

Query Syntax

```
MEASure[1|2][:SCALar]:TINTerval:DELay? <delay>[,<expected value>
[,<resolution>]]
```

```
MEASure[1|2]:ARRay:TINTerval:DELay? <array size>,<delay>
[,<expected value>[,<resolution>]]
```

Query Response See the FETCh? commands.

***RST Value** Default configuration

Limits

<Array Size>	1 to 1000
<delay>	1e–9 to 9e–6
<Expected Value>	none
<Resolution>	none

Related Commands N/A

Description

This command sets the Counter to make a time interval measurement with a delay and fetch the result. This measurement is made between input channels 1 and 2. The MEASure suffix selects the input channel for the beginning of the interval, the end of the interval will be from the remaining channel. The measurement is made from the first detected rising edge on the first channel to the first following rising edge after the delay time on the second channel. If the <expected value> argument is used, the expected period of the input signal should be used, not the expected time interval.

This command will modify the following:

function – set to TINT
aperture – described below
aperture/events mode – set to APERture

if autosetup mode is ON or ONCE, the input channel attenuation, offset, gain, level and slope can be modified

If neither of the optional arguments are used:

aperture is set to the default value

If only the <expected value> argument is used:

aperture is set to the default value

If both <expected value> and <resolution> are used the aperture is calculated as follows:

$$aperture = 10^{(-9 + \log < expected > - \log < resolution >)}$$

with a minimum of 1e–8 and a maximum of 5 seconds. A longer aperture may be set with the SENSe:TINTerval:APERture command.

MEASure[1|2][:SCALar]:TOTalize?

Command Syntax N/A

Query Syntax `MEASure[1|2][:SCALar]:TOTalize?`

Query Response See the FETCh? commands.

***RST Value** Default configuration

Limits N/A

Related Commands N/A

Description This command initiates the Counter to count events on both input channels 1 and 2. This measurement is unique. It allows you to fetch the totals while it is still counting. Each fetch will return updated count values. This command has a default aperture of 99 days. Use the ABORt command to end this measurement. Since this measurement allows reading the Counter hardware while it is counting, there is some risk that the value returned will be in error.

READ?

Command Syntax N/A

Query Syntax `READ?`

Query Response The read query causes an INITiate:IMMediate action and a FETCh? query. See the INITiate and FETCh command descriptions.

***RST Value** N/A

Limits N/A

Related Commands INITiate, FETCh?

Description N/A

SENSe[:...]:COUNT

The SENSe commands allow the user to select the input channel, type of measurement to be made and the manner in which it is made. It does not cause a measurement to be made.

Only the SENSe:FUNCtion command has a SENSe suffix. This suffix will select an input channel to be used for the FUNCtion. If a SENSe suffix is used on the other SENSe commands, a “No suffix allowed error” will be set.

Command Syntax

```
SENSe:VOLTage

    :AC:COUNT <array size>|DEFault|MINimum|MAXimum
    :DC:COUNT <array size>|DEFault|MINimum|MAXimum
    :MINimum:COUNT <array size>|DEFault|MINimum|MAXimum
    :MAXimum:COUNT <array size>|DEFault|MINimum|MAXimum
    :PTPeak:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:FREQuency:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:PERiod:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:FREQuency:RATio:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:RISE:TIME:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:RTIMe:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:FALL:TIME:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:FTIMe:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:PWIDth:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:NWIDth:COUNT <array size>|DEFault|MINimum|MAXimum

SENSe:TINTerval:COUNT <array size>|DEFault|MINimum|MAXimum
```

Query Syntax

```
SENSe:VOLTage

    :AC:COUNt? [DEFault|MINimum|MAXimum]
    :DC:COUNt? [DEFault|MINimum|MAXimum]
    :MINimum:COUNt? [DEFault|MINimum|MAXimum]
    :MAXimum:COUNt? [DEFault|MINimum|MAXimum]
    :PTPeak:COUNt? [DEFault|MINimum|MAXimum]

SENSe:FREQuency:COUNt? [DEFault|MINimum|MAXimum]

SENSe:PERiod:COUNt? [DEFault|MINimum|MAXimum]

SENSe:FREQuency:RATio:COUNt? [DEFault|MINimum|MAXimum]
```

```
SENSe:RISE:TIME:COUNt? [DEFault|MINimum|MAXimum]

SENSe:RTIMe:COUNt? [DEFault|MINimum|MAXimum]

SENSe:FALL:TIME:COUNt? [DEFault|MINimum|MAXimum]

SENSe:FTIMe:COUNt? [DEFault|MINimum|MAXimum]

SENSe:PWIDth:COUNt? [DEFault|MINimum|MAXimum]

SENSe:NWIDth:COUNt? [DEFault|MINimum|MAXimum]

SENSe:TINTerval:COUNt? [DEFault|MINimum|MAXimum]
```

Query Response

These queries all do the same thing. These queries move to the output buffer the currently set <array size> count that was set by the most recent SENSe:...:COUNt, CONFigure or MEASure command. If one of the optional arguments is used, the DEFault, MINimum or MAXimum value of count is moved to the output buffer instead.

***RST Value**

1

Limits

<array size> 1 to 1000

Related Commands

N/A

Description

These commands all do the same thing. These commands set the Counter to do <array size> number of measurements. This count will remain in effect no matter what SENSe:FUNCtion is selected.

The default value of count is 1.

MINimum = 1
MAXimum = 1000

CONFigure and MEASure commands also set this count.

SENSe[:...]:APERture

***NOTE**. Time Interval With Delay uses all available hardware timers. When the counter is in this mode, the aperture is controlled by software and has a minimum period of approximately 10 ms.*

Command Syntax

```
SENSe:FREQuency:APERture <time>|DEFault|MINimum|MAXimum
SENSe:PERiod:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FREQuency:RATio:APERture <time>|DEFault|MINimum|MAXimum
SENSe:RISE:TIME:APERture <time>|DEFault|MINimum|MAXimum
SENSe:RTIMe:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FALL:TIME:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FTIMe:APERture <time>|DEFault|MINimum|MAXimum
SENSe:PWIDth:APERture <time>|DEFault|MINimum|MAXimum
SENSe:NWIDth:APERture <time>|DEFault|MINimum|MAXimum
SENSe:TINTerval:APERture <time>|DEFault|MINimum|MAXimum
```

Query Syntax

```
SENSe:FREQuency:APERture? [DEFault|MINimum|MAXimum]
SENSe:PERiod:APERture? [DEFault|MINimum|MAXimum]
SENSe:FREQuency:RATio:APERture? [DEFault|MINimum|MAXimum]
SENSe:RISE:TIME:APERture? [DEFault|MINimum|MAXimum]
SENSe:RTIMe:APERture? [DEFault|MINimum|MAXimum]
SENSe:FALL:TIME:APERture? [DEFault|MINimum|MAXimum]
SENSe:FTIMe:APERture? [DEFault|MINimum|MAXimum]
SENSe:PWIDth:APERture? [DEFault|MINimum|MAXimum]
SENSe:NWIDth:APERture? [DEFault|MINimum|MAXimum]
SENSe:TINTerval:APERture? [DEFault|MINimum|MAXimum]
```

Query Response

These queries all do the same thing. These queries move to the output buffer the currently set aperture time that was set by the most recent SENSe:...:APERture, CONFigure or MEASure command. If one of the optional arguments is used, the DEFault, MINimum or MAXimum value of aperture is moved to the output buffer instead.

***RST Value** le–1

Limits <time> 1e–8 to 9e6

Related Commands

SENSe:FREQuency:MODe APERture|EVENts|DEFault
SENSe:PERiod:MODe APERture|EVENts|DEFault
SENSe:FREQuency:RATio:MODe APERture|EVENts|DEFault
SENSe:RISE:TIME:MODe APERture|EVENts|DEFault
SENSe:RTIMe:MODe APERture|EVENts|DEFault
SENSe:FALL:TIME:MODe APERture|EVENts|DEFault
SENSe:FTIMe:MODe APERture|EVENts|DEFault
SENSe:PWIDth:MODe APERture|EVENts|DEFault
SENSe:NWIDth:MODe APERture|EVENts|DEFault
SENSe:TINTerval:MODe APERture|EVENts|DEFault

Description

These commands all do the same thing. These commands set the Counter measurement aperture <time>. Whether the aperture time is used or not depends on the mode set with one of the SENSe:...:MODE commands. The default mode is to use an aperture. The units are in seconds and range from 1e–8 to 9e6 in 1e–9 size steps. This count will remain in effect no matter what SENSe:FUNCtion is selected.

CONFigure and MEASure commands also set the aperture.

SENSe[:...]:EVENts

***NOTE**. Event mode is invalid for Time Interval with Delay measurements.*

Command Syntax

```
SENSe:FREQuency:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:PERiod:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:FREQuency:RATio:EVENts <number of events>|DEFault|MINimum
|MAXimum

SENSe:RISE:TIME:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:RTIMe:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:FALL:TIME:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:FTIMe:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:PWIDth:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:NWIDth:EVENts <number of events>|DEFault|MINimum|MAXimum

SENSe:TINTerval:EVENts <number of events>|DEFault|MINimum|MAXimum
```

Query Syntax

```
SENSe:FREQuency:EVENts? [DEFault|MINimum|MAXimum]

SENSe:PERiod:EVENts? [DEFault|MINimum|MAXimum]

SENSe:FREQuency:RATio:EVENts? [DEFault|MINimum|MAXimum]

SENSe:RISE:TIME:EVENts? [DEFault|MINimum|MAXimum]

SENSe:RTIMe:EVENts? [DEFault|MINimum|MAXimum]

SENSe:FALL:TIME:EVENts? [DEFault|MINimum|MAXimum]

SENSe:FTIMe:EVENts? [DEFault|MINimum|MAXimum]

SENSe:PWIDth:EVENts? [DEFault|MINimum|MAXimum]

SENSe:NWIDth:EVENts? [DEFault|MINimum|MAXimum]

SENSe:TINTerval:EVENts? [DEFault|MINimum|MAXimum]
```

Query Response

These queries all do the same thing. These queries move to the output buffer the number of events that was set by the most recent SENSe:...:EVENts command.

If one of the optional arguments is used, the DEFault, MINimum or MAXimum value of aperture is moved to the output buffer instead.

***RST Value** 1e3

Limits <number of events> 1 to 9e15

Related Commands

SENSe:FREQuency:MODe APERture|EVENts|DEFault
SENSe:PERiod:MODe APERture|EVENts|DEFault
SENSe:FREQuency:RATio:MODe APERture|EVENts|DEFault
SENSe:RISE:TIME:MODe APERture|EVENts|DEFault
SENSe:RTIMe:MODe APERture|EVENts|DEFault
SENSe:FALL:TIME:MODe APERture|EVENts|DEFault
SENSe:FTIMe:MODe APERture|EVENts|DEFault
SENSe:PWIDth:MODe APERture|EVENts|DEFault
SENSe:NWIDth:MODe APERture|EVENts|DEFault
SENSe:TINTerval:MODe APERture|EVENts|DEFault

Description

These commands all do the same thing. These commands set the Counter <number of events>. If the SENSe:<function>:MODE is set to EVENTs, the Counter will make a measurement for a <number of events> number of cycles of the input signal This <number of events> will remain in effect no matter what SENSe:FUNCtion is selected. The range of events is 1 to 9e15 step 1, however the measurement must complete in 9e6 seconds (99 days) to avoid errors.

The default value is 1e3

SENSe[:...]:MODe

***NOTE**. Event mode is invalid for Time Interval with Delay measurements.*

Command Syntax

```
SENSe:FREQuency:MODe APERture|EVENts|DEFault
SENSe:PERiod:MODe APERture|EVENts|DEFault
SENSe:FREQuency:RATio:MODe APERture|EVENts|DEFault
SENSe:RISE:TIME:MODe APERture|EVENts|DEFault
SENSe:RTIMe:MODe APERture|EVENts|DEFault
SENSe:FALL:TIME:MODe APERture|EVENts|DEFault
SENSe:FTIMe:MODe APERture|EVENts|DEFault
SENSe:PWIDth:MODe APERture|EVENts|DEFault
SENSe:NWIDth:MODe APERture|EVENts|DEFault
SENSe:TINTerval:MODe APERture|EVENts|DEFault
```

Query Syntax

```
SENSe:FREQuency:MODe? [DEFault]
SENSe:PERiod:MODe? [DEFault]
SENSe:FREQuency:RATio:MODe? [DEFault]
SENSe:RISE:TIME:MODe? [DEFault]
SENSe:RTIMe:MODe? [DEFault]
SENSe:RISE:TIME:MODe? [DEFault]
SENSe:FTIMe:MODe? [DEFault]
SENSe:PWIDth:MODe? [DEFault]
SENSe:NWIDth:MODe? [DEFault]
SENSe:TINTerval:MODe? [DEFault]
```

Query Response

These queries all do the same thing. These queries move to the output buffer the currently set mode that was set by the most recent SENSe:...:MODE, CONFigure or MEASure command. If the optional DEFault argument is used, then the default value of mode will be moved to the output buffer instead.

***RST Value** APERture

Limits N/A

Related Commands

SENSe:FREQuency:APERture <time>|DEFault|MINimum|MAXimum
SENSe:PERiod:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FREQuency:RATio:APERture <time>|DEFault|MINimum|MAXimum
SENSe:RISE:TIME:APERture <time>|DEFault|MINimum|MAXimum
SENSe:RTIMe:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FALL:TIME:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FTIMe:APERture <time>|DEFault|MINimum|MAXimum
SENSe:PWIDth:APERture <time>|DEFault|MINimum|MAXimum
SENSe:NWIDth:APERture <time>|DEFault|MINimum|MAXimum
SENSe:TINTerval:APERture <time>|DEFault|MINimum|MAXimum
SENSe:FREQuency:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:PERiod:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:FREQuency:RATio:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:RISE:TIME:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:RTIMe:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:FALL:TIME:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:FTIMe:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:PWIDth:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:NWIDth:EVENts <number of events>|DEFault|MINimum|MAXimum
SENSe:TINTerval:EVENts <number of events>|DEFault|MINimum|MAXimum

Description

These commands all do the same thing. These commands set the Counter to make a measurement for a length of time (APERture) or for a number of cycles of the input signal (EVENTs). This MODE will remain in effect no matter what SENSe:FUNCtion is selected. However some functions don’t have one or both of the APERture and EVENts modes. Those functions ignor the mode setting.

The default mode is APERture.

CONFigure and MEASure commands set the mode to APERture.

SENSe[1|2|3]:FUNCtion

Command Syntax

```
SENSe:FUNCtion "TOTalize"

SENSe[1,2]:FUNCtion "VOLTage:AC"

    "VOLTage:DC"
    "VOLTage:MINimum"
    "VOLTage:MAXimum"
    "VOLTage:PTPeak"
    "RISE:TIME"
    "RTIMe"
    "FALL:TIME"
    "FTIMe"
    "PWIDth"
    "NWIDth"
    "TINTerval"

SENSe[1,2,3]:FUNCtion "FREQuency"

    "PERiod"
    "FREQ:RATio <second channel>"
```

Query Syntax

```
SENSe[1|2|3]:FUNCtion?
```

Query Response

This query moves to the output buffer the currently selected function. Functions are selected by a SENSe:FUNCtion, CONFigure or MEASure command. The possible strings returned are:

```
"VOLT:AC"
"VOLT:DC"
"VOLT:MIN"
"VOLT:MAX"
"VOLT:PTP"
"FREQ"
"PER"
"FREQ:RAT <secondchannel>"
"RISE:TIME"
"RTIM"
"FALL:TIME"
"FTIM"
"PWID"
"NWID"
"TINT"
"TOT"
```

If the SENSe suffix is for a different input channel than previously configured, the Counter sets an error “Channel not configured”.

***RST Value** “VOLT:DC“

Limits N/A

Related Commands CONFigure, MEASure

Description The SENSe:FUNCtion command selects a function and input channel without changing most of the set up of the Counter. The input coupling and impedance are not changed by these commands. The user must select the coupling and impedance are not changed by these commands. The user must select the coupling and impedance that makes sense for the input signal. The input attenuation, offset, gain, and comparator hysteresis are not changd by these commands. However, the comparator slopes and thresholds are changed to defaults by these commands. If INPut:SETup:AUTO (autotrigger) is set to ON or ONCE, the input attenuation, offset, gain, and comparator hysteresis may change when the measurement is started. See the CONFigure or MEASure command descriptions for detail of these commands.

SENSe:TINTerval:DELay[:STATe]

Command Syntax `SENSe:TINTerval:DELay[:STATe] ON|OFF|DEFault`

Query Syntax `SENSe:TINTerval:DELay[:STATe]? [DEFault]`

Query Response Returns 0 for off, 1 for on

***RST Value** OFF

Limits N/A

Related Commands SENSe:TINTerval:DELay:TIME

Description This command controls adding a delay between the when the first channel senses an edge to when the second channel is enabled to sense an edge.

The default value is off.

SENSe:TINTerval:DELay:TIME

Command Syntax `SENSe:TINTerval:DELay:TIME <numeric value>|MINimum|MAXimum|DEFault`

Query Syntax `SENSe:TINTerval:DELay:TIME? [DEFault]|[MINimum]|[MAXimum]`

Query Response Returns current value of the time interval delay or the default, minimum or maximum value.

***RST Value** le–6

Limits <numeric value> 1e–9 to 9e+6

Related Commands SENSe:TINTerval:DELay[:STATe]

Description This command sets the delay time for the time interval with delay function. The default value is 1e–6.

STATus:OPERation:CONDition?

Command Syntax N/A

Query Syntax `STATus:OPERation:CONDition?`

Command Class Instrument

Query Response The operational condition register value.

***RST Value** 0

Limits N/A

Formats Query Response
Numeric

Related Commands

MEASure?
READ?
INITiate
ABORt

Description

The STATus:OPERation:CONDition query returns the current operational status of the Counter. The bit definitions of the value are (bit 0 = the least significant bit):

Bit	Definition	Function
0	Calibrating	Set when any CALibration operation is running. Cleared when the CALibration operation is complete
1	Settling	Set when the instrument changes its function or range. Cleared when the all circuitry has settled
2	Ranging	Set when the instrument is auto-ranging. Cleared when the input range has been found
3	Sweeping	Not used
4	Measuring	Set when an INITiate command is executed. Cleared when the command is complete or aborted
5	Triggering	Not used
6	Arming	Set when the instrument is waiting for an arm signal. Cleared when the arm is received
7	Correcting	Set when the instrument is performing an auto-zero operation. Cleared when the auto-zero operation is complete
8	Testing (User 1)	Set when the instrument is performing a self-test. Cleared when the self-test is complete
9	Aborting (User 2)	Set when the instrument is in the process of aborting an operation. Cleared when the abort is complete
10	User 3	Not used
11	User 4	Not used
12	User 5	Reserved
13	Instrument Summary	Not used
14	Program Running	Not used
15	Reserved	Always 0

Table 3–18: STATus:OPERation:CONDition Examples

Command	Response	Description
status:operation:condition?	16	Measurement in progress (0010 hex)
stat:oper:cond?	3072	Measurement complete because an ABORt was received (0C00 hex)

Trigger Commands for the Counter

A summary of the trigger-related commands for the Counter is as follows:

```
ARM ([:SEQuence1] |:STARt) [:LAYer]:DELay
ARM ([:SEQuence1] |:STARt) [:LAYer]:ECOunt
ARM ([:SEQuence1] |:STARt) [:LAYer]:IMMediate
ARM ([:SEQuence1] |:STARt) [:LAYer]:LEVel
ARM ([:SEQuence1] |:STARt) [:LAYer]:MODE
ARM ([:SEQuence1] |:STARt) [:LAYer]:SLOPe
ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce:CATalog[:ALL]?
ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce:CATalog:DELayable?
ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce:CATalog:FIXed?
ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce
ARM ([:SEQuence2] |:STOP) [:LAYer]:DELay
ARM ([:SEQuence2] |:STOP) [:LAYer]:ECOunt
ARM ([:SEQuence2] |:STOP) [:LAYer]:IMMediate
ARM ([:SEQuence2] |:STOP) [:LAYer]:SOURce:CATalog[:ALL]?
ARM ([:SEQuence2] |:STOP) [:LAYer]:SOURce:CATalog:DELayable?
ARM ([:SEQuence2] |:STOP) [:LAYer]:SOURce:CATalog:FIXed?
ARM ([:SEQuence2] |:STOP) [:LAYer]:SOURce
```

```
INSTrument:ABORt
```

```
TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?
```

ARM ([:SEQuence1] |:STARt) [:LAYer]:DELay

Command Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:DELay <delay in seconds>`

Query Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:DELay?`

Command Class Instrument

Query Response `<delay in seconds>`

***RST Value** 0 seconds (pass-through)

Limits 0 = pass through, 1 µs – 65.535 ms in 1 µs steps

Related Commands ARM:ECOunt

Description Specifies a time delay to occur after receipt of an arm signal prior to actually arming the Counter. If the arming source selected is fixed, this command will have no effect on the Counter arming. This command always zeros the event count delay, so specifying a delay of zero places the arm subsystem in pass-through mode. In this mode, the instrument arms immediately upon receipt of a trigger.

Examples

Command	Response
INST:SEL COUNTER	
CONF:VOLT:DC	
ARM:SOUR TTLT0	
ARM:DEL 1E–3	
INIT	

ARM ([:SEQuence1] |:STARt) [:LAYer]:ECOunt

Command Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:ECOunt <triggers to count>`

Query Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:ECOunt?`

Command Class Instrument (COUNTER)

Query Response `<triggers to count>`

***RST Value** 0 triggers (pass-through)

Limits 0 = pass through, 1 to 65,535 triggers

Related Commands ARM:DELay

Description Specifies the number of arm signals to count prior to arming the Counter. Upon receipt of arming signal N (where N is the number specified in the comand), the Counter will enter the armed state. If the arm source selected is fixed, this command will have no effect on the Counter arming. This command always zeros the delay by time parameter, so specifying an event count of zero places the arm subsystem in pass-through mode. In this mode, the instrument arms immediately upon receipt of a trigger.

Examples

Command	Response
INST:SEL COUNTER	
CONF:VOLT:DC	
ARM:SOUR TTLT0	
ARM:ECO 100	
INIT	

ARM ([:SEQuence1] |:STARt) [:LAYer]:IMMediate

Command Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:IMMediate`

Query Syntax N/A

Command Class Instrument (COUNTER)

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Cause a one time entry by the Counter into the armed state without receiving the specified start arm signal. This command is often used to “prime the pump” in cases such as setting up a scan list measurement.

Examples

Command	Response
INST:SEL COUNTER	
ARM:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2
CONF:FREQ	
ARM:SOUR TTLT0	
INIT	
FETC:COUN?	0
ARM:IMM	
FETC:COUN?	1
FETC?	1.234567890E6

ARM ([:SEQuence1] |:STARt) [:LAYer]:LEVel

Command Syntax `ARM ([:SEQuence1] |:STARt)[:LAYer]:LEVel`

Query Syntax `ARM ([:SEQuence1] |:STARt)[:LAYer]:LEVel?`

Command Class Instrument (COUNTER)

Query Response TTL | ECL | ZERO | MAXimum | MINimum | AUTO | <desired voltage>

***RST Value** 1.4 Volts

Limits
TTL = 1.6 V
ECL = 1.3 V
ZERO = 0.0 V
MAXimum = 20.0 V
MINimum = –20.0 V
AUTO = 1.4 V
–20 to +20 V

Related Commands ARM:SLOPe

Description

Specifies the comparator level of Counter External Arm Signal. When the External Arm Signal crosses this level with the appropriate slope, the CTR_EX-TARM trigger source will be activated.

Examples

Command	Response
INST: SEL COUNTER	
ARM: SLOP NEG	
ARM: LEV TTL	
INST: SEL DMM	
TRIG: SOUR CTR_EXTARM	
CONF: VOLT: DC	
INIT	
<wait until falling edge of external arm crosses to 1.6 V>	
FETC?	+1.00000E-01

ARM ([:SEQuence1] |:STARt) [:LAYer]:MODE

Command Syntax

`ARM ([:SEQuence1] |:STARt) [:LAYer]:MODE ONCE|ALL`

Query Syntax

`ARM ([:SEQuence1] |:STARt) [:LAYer]:MODE?`

Command Class

Instrument

Query Response

`Current state:ONCE|ALL`

***RST Value**

ALL

Limits

ONCE, ALL

Related Commands

N/A

Description

When the Counter has been configured for some type of array measurement, this command specifies whether the Counter will perform one or all operations when an arm is received. If the mode is ALL, then all operations will be completed upon receipt of one arm signal. If the mode is ONCE, then the instrument will perform one operation and reenter the initiated state. This will continue until the specified number of arm signals have been received (and hence, the specified number of operations have been completed).

Examples

Command	Response
INST:SEL COUNTER	
CONF:ARR:FREQ 3	
ARM:MODE ONCE	
ARM:SOUR COMM0	
INIT	
FETC:COUN?	0
TRIG:FIR0	
FETC:COUN?	1
TRIG:FIR0	
TRIG:FIR0	
FETCH:COUN?	3
FETC?	#2411.234567891E6, 1.234567890E6, 1.234567892E6

ARM ([:SEQuence1] |:STARt) [:LAYer]:SLOPe

Command Syntax `ARM (:SEQuence1] |:STARt)[:LAYer]:SLOPe`

Query Syntax `ARM (:SEQence1] |:STARt)[:LAYer]:SLOPe?`

Command Class Instrument (COUNTER)

Query Response POSitive | NEGative

***RST Value** POSitive

Limits POSitive | NEGative

Related Commands ARM:LEVel

Description Specifies the slope of the Counter External Arm Signal. When the slope is POSitive, the CTR_EXTARM trigger source will be activated by a rising edge passing through the specified level (see ARM:LEVel). When the slope is NEGative, the CTR_EXTARM trigger source will be activated by a falling edge passing through the specified level (see ARM:LEVel).

Examples See ARM:LEVel example

ARM ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?

Command Syntax N/A

Query Syntax `ARM ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?`

Command Class Instrument (COUNTER)

Query Response `HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7`

***RST Value** N/A

Limits N/A

Related Commands ARM:SOURce:CATalog:DELayable?
ARM:SOURce:CATalog:FIXed?
ARM:SOURce

Description Lists all trigger sources available for use with the ARM:SOUR command. This command specifies the start arming source for the Counter.

Examples

Command	Response
ARM:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2

ARM ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?

Command Syntax N/A

Query Syntax `ARM ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?`

Command Class Instrument (COUNTER)

Query Response `BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value** N/A

Limits N/A

Related Commands ARM:SOURce:CATalog?
ARM:SOURce:CATalog:FIXed?
ARM:SOURce

Description Lists all delayable trigger sources available for use with the ARM:SOUR command. This command specifies the start arming source for the Counter.

Examples

Command	Response
ARM:SOUR:CAT:DEL?	BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM

ARM ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?

Command Syntax N/A

Query Syntax `ARM ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?`

Command Class Instrument (COUNTER)

Query Response `HOLD,IMMEDIATE,TIMER,CTR_CHAN2`

***RST Value** N/A

Limits N/A

Related Commands ARM:SOURce:CATalog?
ARM:SOURce:CATalog:DELayable?
ARM:SOURce

Description Lists all fixed trigger sources available for use with the ARM:SOUR command. This command specifies the start arming source for the Counter.

Examples

Command	Response
ARM:SOUR:CAT:FIX?	HOLD,IMMEDIATE,TIMER,CTR_CHAN2

ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce

Command Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce <source>`

Query Syntax `ARM ([:SEQuence1] |:STARt) [:LAYer]:SOURce?`

Command Class Instrument (COUNTER)

Query Response `<current source>`

***RST Value** IMMEDIATE

Limits BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2

Related Commands ARM:SOURce:CATalog?

Description

Selects or queries the start arming source to be used when the Counter is initiated.

Examples

Command	Response
INST:SEL COUNTER	
ARM:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2
CONF:FREQ	
ARM:SOUR BUS	
INIT	
FETC:COUN?	0
*TRG	
FETC:COUN?	1
FETC?	1.234567890E6

ARM (:SEQuence2 |:STOP) [:LAYer]:DELay

Command Syntax

`ARM (:SEQuence2 |:STOP) [:LAYer]:DELay <delay in seconds>`

Query Syntax

`ARM (:SEQuence2 |:STOP) [:LAYer]:DELay?`

Command Class

Instrument (COUNTER)

Query Response

`<delay in seconds>`

***RST Value**

0 seconds (pass-through)

Limits

0 = pass through, 1 μs – 65.535 ms in 1 μs steps

Related Commands

ARM:STOP:ECOunt

Description

Specifies a time delay to occur after receipt of a stop arm signal prior to actually disarming the Counter. If the stop arming source selected is fixed, this command will have no effect on the Counter arming. This command always zeros the event count delay, so specifying a delay of zero places the arm subsystem in pass-through mode. In this mode, the instrument disarms immediately upon receipt of an arming signal.

Examples

Command	Response
INST:SEL COUNTER	
CONF:VOLT:DC	
ARM:STOP:SOUR TTLT0	
ARM:STOP:DEL 1E–3	
INIT	

ARM (:SEQuence2 |:STOP) [:LAYer]:ECOunt

Command Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:ECOunt <triggers to count>`

Query Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:ECOunt?`

Command Class Instrument (COUNTER)

Query Response `<triggers to count>`

***RST Value** 0 triggers (pass-through)

Limits 0 = pass through, 1 to 65,535 triggers

Related Commands ARM:STOP:DELay

Description Specifies the number of stop arm signals the instrument will count prior to disarming the Counter. Upon receipt of arming signal N (where N is the number specified in the comand), the Counter will exit the armed state. If the arm source selected is fixed, this command will have no effect on disarming the Counter. This command always zeros the delay by time parameter, so specifying an event count of zero places the arm subsystem in pass-through mode. In this mode, the instrument arms immediately upon receipt of a trigger.

Examples

Command	Response
INST:SEL COUNTER	
CONF:VOLT:DC	
ARM:STOP:SOUR TTLT0	
ARM:STOP:ECO 100	
INIT	

ARM (:SEQuence2 |:STOP) [:LAYer]:IMMediate

Command Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:IMMediate`

Query Syntax N/A

Command Class Instrument (COUNTER)

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Cause a one-time exit by the Counter from the armed state without receiving the specified stop arm signal.

Examples

Command	Response
INST:SEL COUNTER	
ARM:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2
CONF:FREQ	

Command	Response
ARM:STOP:SOUR TTLT0	
INIT	
FETC:COUN?	0
ARM:STOP:IMM	
FETC:COUN?	0

ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce:CATalog[:ALL]?

Command Syntax N/A

Query Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce:CATalog[:ALL]?`

Command Class Instrument

Query Response `HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, INTERNAL, LEVEL`

***RST Value** N/A

Limits N/A

Related Commands ARM:STOP:SOURce:CATalog:DELayable?
ARM:STOP:SOURce:CATalog:FIXed?
ARM:STOP:SOURce

Description Lists all trigger sources available for use with the ARM:STOP:SOUR command. This command specifies the stop arming source for the Counter.

Examples

Command	Response
ARM:STOP:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, CTR_CHAN2, INTERNAL, LEVEL

ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce:CATalog:DELayable?

Command Syntax N/A

Query Syntax `ARM (:SEQuence2 |:STOP)[:LAYer]:SOURce:CATalog:DELayable?`

Command Class Instrument (COUNTER)

Query Response `BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value** N/A

Limits N/A

Related Commands
ARM:STOP:SOURce:CATalog?
ARM:STOP:SOURce:CATalog:FIXed?
ARM:STOP:SOURce

Description Lists all delayable trigger sources available for use with the ARM:STOP:SOUR command. This command specifies the stop arming source for the Counter.

Examples

Command	Response
ARM:STOP:SOUR:CAT:DEL?	BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM

ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce:CATalog:FIXed?

Command Syntax N/A

Query Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce:CATalog:FIXed?`

Command Class Instrument (COUNTER)

Query Response `HOLD,IMMEDIATE,TIMER,INTERNAL,LEVEL`

***RST Value** N/A

Limits N/A

Related Commands ARM:STOP:SOURce:CATalog?
ARM:STOP:SOURce:CATalog:DELayable?
ARM:STOP:SOURce

Description Lists all fixed trigger sources available for use with the ARM:STOP:SOUR command. This command specifies the stop arming source for the Counter.

Examples

Command	Response
ARM:STOP:SOUR:CAT:FIX?	HOLD,IMMEDIATE,TIMER,INTERNAL,LEVEL

ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce

Command Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce <source>`

Query Syntax `ARM (:SEQuence2 |:STOP) [:LAYer]:SOURce?`

Command Class Instrument (COUNTER)

Query Response `<current source>`

***RST Value** IMMEDIATE

Limits BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM, INTERNAL, LEVEL

Related Commands ARM:STOP:SOURce:CATalog?

Description Selects or queries the stop arming source to be used when the Counter is initiated.

Examples

Command	Response
INST:SEL COUNTER	
ARM:STOP:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM, INTERNAL, LEVEL
CONF:ARR:FREQ 512	
ARM:SOUR BUS	
ARM:STOP:SOUR BUS	
INIT	
FETC:COUN?	0
*TRG	
FETC:COUN?	23
FETC:COUN?	47
*TRG	
FETC:COUN?	53

INSTrument:ABORt

Command Syntax `INSTrument:ABORt`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands ABORt
INITiate
INITiate:CONTinuous

Description Places active instrument in the IDLE state, aborting any measurement or other instrument activity in progress. The instrument configuration is unchanged and a subsequent INIT command will cause the instrument to re-start the same type of measurement. If the instrument is in asynchronous mode, this command can be sent while a query is in progress and the measurment will be aborted.

If the instrument is in synchronous mode, this command will be queued while a query is in progress. This is a ramification of the IEEE 488.2 Message Exchange Protocol Enforcer (MEPE).

Examples

Command	Response
INST:SEL DMM	
CONF:ARR:VOLT:DC	512
INIT	
FETC:COUN?	127
INST:ABOR	
FETC:COUN?	153
FETC:COUN?	153

NOTE*. After abort, no more measurements are taken.*

TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?

Command Syntax N/A

Query Syntax `TRIGger ([:SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?`

Command Class Instrument

Query Response `HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value** N/A

Limits N/A

Related Commands
TRIGger:SOURce:CATalog:DELayable?
TRIGger:SOURce:CATalog:FIXed?
TRIGger:SOURce
TRIGger:FIRe4
TRIGger:TIMer

Description Lists all trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM

SCPI Commands for the SurePath™ Modules

This section lists the SCPI commands and queries for the SurePath™ modules in alphabetic order. A summary listing of the SCPI command set for the SurePath™ Modules is as follows:

```
INITiate:CONTinuous
INITiate[:IMMediate]

INSTrument:ABORt
INSTrument:RESet

[ROUTe:]CLOSe
[ROUTe:]CLOSe:DWELl
[ROUTe:]CLOSe:MODE

[ROUTe:]CONFigure
[ROUTe:]CONFigure:DISJoin
[ROUTe:]CONFigure:JOIN

[ROUTe:]ID?

[ROUTe:]MODule[:DEFine]
[ROUTe:]MODule:CATalog?
[ROUTe:]MODule:CATalog:SUPPorted?
[ROUTe:]MODule:DELete[:NAME]
[ROUTe:]MODule:DELete:ALL

[ROUTe:]OPEN
[ROUTe:]OPEN:ALL
[ROUTe:]OPEN:DWELl

[ROUTe:]PFAil

[ROUTe:]SCAN
[ROUTe:]SCAN:RATE

STATus:OPERation:CONDition?
```

Trigger Commands

In addition, this section includes a listing of the commands used for triggering measurements for the SurePath™ Modules. A summary of the trigger commands follow the instrument-specific SCPI commands.

***NOTE**. The Examples used in the command summaries are for illustrative purposes only and apply only to the VX4380 model of the SurePath™ Modules.*

INITiate: CONTinuous

Command Syntax `INITiate: CONTinuous [0|OFF|1|ON]`

Query Syntax `INITiate: CONTinuous?`

Command Class Instrument

Query Response 0 = continuous initiate not enabled
1 = continuous initiate enabled

***RST Value** N/A

Limits 0, 1, ON, or OFF

Related Commands ABORt
INSTrument:ABORt
INITiate

Description Instrument initiates its current trigger sequence. After the instrument has completed the current trigger sequence, it enters the initiated state a second time. It will continue this cycle until an abort, reset, or INIT:CONT OFF is received. See instrument documentation for details on instrument state after an INITiate.

Examples See TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

INITiate [:IMMediate]

Command Syntax `INITiate [:IMMediate]`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands
ABORt
INSTrument:ABORt
INITiate:CONTinuous

Description Instrument initiates its current trigger sequence. After the instrument has completed the current trigger sequence, it enters the idle state. See instrument documentation for details on instrument state after an INITiate.

Examples See TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

INSTrument:ABORt

Command Syntax `INSTrument:ABORt`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands
ABORt
INITiate
INITiate:CONTinuous

Description Places active instrument in the IDLE state, aborting any measurement or other instrument activity in progress. The instrument configuration is unchanged and a subsequent INIT command will cause the instrument to re-start the same type of measurement. If the instrument is in asynchronous mode, this command can be sent while a query is in progress and the measurement will be aborted.

If the instrument is in synchronous mode, this command will be queued while a query is in progress.

Examples

See the TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

NOTE*. After abort, no more measurements are taken.*

INSTrument:RESet

Command Syntax

`INSTrument:RESet`

SurePath™ Module

Valid Command for all SurePath™ modules

Query Syntax

N/A

Command Class

Instrument

Query Response

N/A

***RST Value**

N/A

Limits

N/A

Related Commands

N/A

Description

This command resets the currently selected instrument without affecting other instruments. The instrument returns to its *RST state. The instrument remains selected.

Examples

Command	Response
INST:SEL DMM	
CONF:RES	
CONF?	":SCAL:RES 3e+08,180000"

Command	Response
INST:RES	
INST:SEL?	DMM
CONF?	“:SCAL:VOLT:DC 300,0.001”

[ROUTe:]

The definition of elements used in SCPI/IEEE 488.2 commands and command descriptions for the [ROUTe:] commands are as follows:

<NR1> ASCII integer representation of a decimal number.

<NRf> ASCII integer, fixed point or floating point representation of a decimal number.

<module_name> A user-defined ASCII string to be associated with the local bus address of a relay module. <module_name> strings must start with a letter and may consist of alphanumeric characters, underscores, and digits. The maximum length of a <module_name> is 12 characters.

<channel_spec> One or more <NR1> ASCII strings separated by “!” characters that specify a relay on a relay module. The format of a <channel_spec> field for each of the SurePath relay modules is:

- VX4320 RF Multiplexer: <NR1> ! <NR1>

  The range of the first <NRf> field is 1 to 4. This field specifies a relay within one of the sections of the VX4320. The range of the second <NRf> field is 1 to 8. This field specifies a section of the VX4320. A one-dimensional <channel_spec> may also be used to specify a channel on a VX4320 Module. The one-dimensional <channel_spec> is given by the formula:

  $$((\text{section} - 1) \times 4) + \text{relay}$$

  where variables “section” and “relay” are section and relay numbers specified in a two-dimensional <channel_spec>.

- VX4330 Scanner/Multiplexer: <NR1> ! <NR1>

  The first <NR1> field specifies a relay within the specified section. The range of this <NR1> field depends on the current configuration of the section of the VX4330 specified in the second <NR1> field. The range of the second <NR1> field is 1 to 6. This field specifies a section of the VX4330.

1 – 10	4-wire
1 – 20	4-wire independent
1 – 20	2-wire
1 – 40	1-wire

- VX4350 General Purpose Switching module: <NR1>

  The range of this field is 1 to 64. It specifies one of 64 relays on the VX4350.

- VX4351 40-Channel, 10 Amp, SPST Switch Module: <NRI>

  The range of this field is 1 to 40, specifying one of the 40 relays on the module. If the module is placed in two-wire mode, then the range of the field becomes 1 to 20, specifying one of the twenty relay pairs available on the module.

- VX4380 Matrix: <NR1> ! <NR1> ! <NR1>

  The range of the first <NR1> field is 1 to 4. It specifies the row of a relay in one of the sections of the VX4380. The range of the second <NR1> field is 1 to 16. It specifies the column of a relay in one of the sections of the VX4380. The range of the third <NR1> field is 1 to 4. It specifies a section of the VX4380. A one dimensional <channel_spec> may also be used to specify a channel on a VX4380 Module. The one dimensional <channel_spec> is given by the formula:

  $$((\text{section} - 1) \times 64) + ((\text{row} - 1) \times 16) + \text{column}$$

  where variables “section” and “row” and “column” are section, row, and column numbers specified in a three-dimensional <channel_spec>.

- VX4381 4 × 4, 10 Amp, Dual 1-Wire Switch Matrix: <NR1> ! <NR1> ! <NR1> or <NR1> ! <NR1>

  By default, the VX4381 is configured as two 4 × 4 matrices. In this case, the first <NR1> field has a range of 1 to 4 and specifies the row of a relay in one of the two sections. The second <NR1> also has a range of 1 to 4, but specifies the column of a relay in one of the two sections. The third <NR1> field of a three-dimensional <channel_spec> can have the value of 1 or of 2, referring to the first or second matrix.

  If the module is placed in two-wire mode, the second matrix is operated parallel with the first matrix. Therefore, a two-dimensional <channel spec> is required. The first <NR1> field in this case has a range of 1 to 4, and specifies relays on the numbered row of both matrices. The second <NR1> field also has a range of 1 to 4, and specifies relays on the numbered column of both matrices.

In either mode, a one-dimensional <channel_spec> may also be used to specify a channel on a VX4381 module. In one-wire mode, this is given by the formula:

((section – 1) * 16) + ((row – 1) * 4) + column

In two-wire mode, the one-dimensional <channel_spec> is given by the formula:

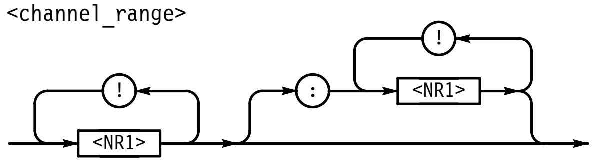
((row–1) * 4) + column

<channel_list> A list of channel numbers on one or more relay modules.

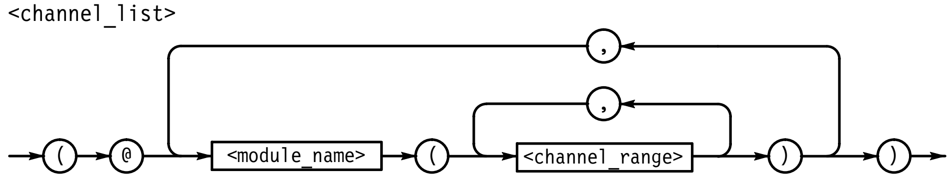
The limits on the channel numbers in a <channel_list> depend on the model number of the relay module(s) specified in the <channel_list>.

The syntax of a <channel_list> is shown in the following diagram:

<channel_range>



<channel_list>



A module_name is an ASCII string that has been associated with a relay module in a ROUTe:MODule:DEFine command. A channel_range may consist of a single <channel_spec> or a range of <channel_spec>s. A range of <channel_spec>s is indicated by two <channel_spec>s separated by a colon (:) character. A <channel_spec> may have one, two or three dimensions depending on the architecture of the relay module it applies to.

The following are examples of valid <channel_list>s for Tektronix VX4320, VX4330, VX4350, and VX4380 relay modules. In these examples it is assumed that the Option 01 is installed on a VX4320 Module. A VX4330, VX4350, and VX4380 are installed in consecutive slots to the right of the VX4320. The default module names for the VX4320, VX4330, VX4350, and VX4380 in this example are m1, m2, m3, and m4 respectively. These module names may be overridden by specifying new module names with the [ROUTe:]MODule[DEFine] command. It is also assumed in these examples that all sections on the VX4330 Module have been configured as 10-to-1 4-wire scanners.

<channel_list>	Channels Specified
`(@m1(1!2))`	Channel number 1 of section 2 on the VX4320 Module
`(@m1(4!1,3!8))`	Channel number 4 in section 1 and relay number 3 in section 8 of the VX4320 Module
`(@m1(4!1:4!8))`	Channel number 4 in sections 1 through 8 on the VX4320 Module
`(@m1(4!1,4!2,4!3,4!4, 4!5,4!6,4!7,4!8))`	Channel number 4 in sections 1 through 8 on the VX4320 Module
`(@m2(1!6))`	Channel 1 in section 6 of the VX4330 Module
`(@m2(1!1:10!6))`	All channels on the VX4330 Module in the following order: 1!1, 1!2, 1!3, 1!4, 1!5, 1!6, 2!1, 2!2, 2!3, 2!4, 2!5, 2!6, ... , 10!1, 10!2, 10!3, 10!4, 10!5, 10!6
`(@m2(1!3:10!3))`	All 10 channels in section 3 of the VX4330
`(@m3(1:64))`	All 64 channels on the VX4350
`(@m3(1,2,3,10,11, 20:13))`	Channels 1, 2, 3, 10, 11, and 20 through 13 on the VX4350
`(@m4(1!13!3))`	The channel that connects row 1 to column 13 in section 3 of the VX4380
`(@m4(65))`	The channel that connects row 1 to column 1 in section 2 of the VX4380
`(@m4(1!1!2))`	Same as the previous example
`(@m4(1:16))`	The channels that connect columns 1 through 16 to row 1 in section 1 of the VX4380
`(@m4(1!1!1:1!16!1))`	Same as the previous example
`(@m4(1!1!1:2!3!4))`	Channels 1!1!1, 1!1!2, 1!1!3, 1!1!4, 1!2!1, 1!2!2, 1!2!3, 1!2!4, 1!3!1, 1!3!2, 1!3!3, 1!3!4, 2!1!1, 2!1!2, 2!1!3, 2!1!4, 2!2!1, 2!2!2, 2!2!3, 2!2!4, 2!3!1, 2!3!2, 2!3!3, 2!3!4 on the VX4380 Module

As the <channel_list> syntax diagram shows, channels on more than one relay module may be specified in a <channel_list>. The next example specifies channels on three different relay modules:

<channel_list>	Channels Specified
`(@m1(1!1), m2(4!6), m4(3!13!2))`	Channel 1 of section 1 on the VX4320, Channel 4 of section 6 of the VX4330, and the channel on the VX4380 that connects row 3 to column 13 in section 2.

As mentioned above, the module names used in a <channel_list> may be specified with a [ROUTe:]MODule[:DEFine] command. The command

```
route:module:Define rfmux, 1
```

changes the module name assigned to the VX4320 to “rfmux”. The following <channel_list> can then be used to specify channels on the VX4320.

```
(@rfmux(3!1,2!2))
```

specifies channel 3 in section 1 and channel 2 in section 2 of the VX4320.

The order in which channels are specified is important in the [ROUTe:] CLOSE? <channel_list> and [ROUTe:]OPEN? <channel_list> queries. The states of the channels are returned in the same order that the channels are specified in the <channel_list>.

The order in which channels are specified is also important in the [ROUTe:] SCAN <channel_list> command. This determines the order in which the relays will be closed each time a trigger event is detected.

The order in which channels are specified in a <channel_list> is important in the [ROUTe:]CLOSe <channel_list> command when channels in the same section of a VX4320 or a VX4330 are specified. A VX4320 can close only one channel in a section. If a [ROUTe:]CLOSE <channel_list> command specifies more than one relay in a section of a VX4320, the last channel in the <channel_list> will be closed.

For example, the command

```
close (@m2(1!1,2!1))
```

will close channel 2 of section 1 of the VX4320.

A VX4330 can close only one channel in a group of joined sections that have been specified in a [ROUTe:]CLOSe:MODe SCAN,<module_name>,<section_list> command. If more than one channel in such a group of sections is specified in a [ROUTe:]CLOSe <channel_list> command, the last channel specified will be closed. For example, the commands

```
route:configure:join m2,(1:6)

route:close:mode scan,m2,(1:6)
```

join the commons of all six sections of the VX4330 Module and set the mode of the [ROUTe]:CLOSE <channel_list> to scan mode for all six sections of the VX4330.

The command

```
route:close (@m2(1!1,1!6))
```

will then result in channel 1 of section 6 being closed and all other channels on the module being opened.

<section_list> One or more <nr1> fields separated by comma (,) or colon (:) characters and enclosed in left and right parentheses. A <section_list> is used to specify the sections of a relay module to be acted upon by a [ROUTe:]CONFigure or [ROUTe:]CLOSe:MODE command. The following are examples of valid <section_list>s.

For commands directed to a VX4330 which has six scanner sections:

<section_list>	Sections Specified
`(1:6)`	Sections 1 through 6
`(1,2,3)`	Sections 1, 2 and 3
`(1:3,5:6)`	Sections 1 through 3 and 5 and 6
`(1:3,5,6)`	Same as previous example
`(3)`	Section 3

[ROUTe:]CLOSe

Command Syntax `[ROUTe:]CLOSe <channel_list>`

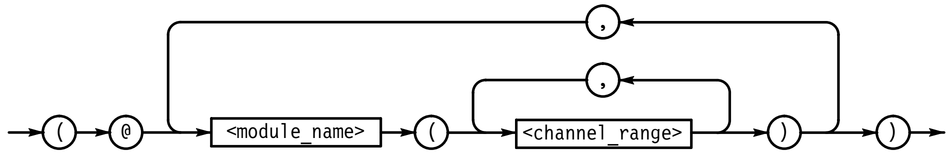
SurePath™ Module Valid command for VX4330 modules only.

Query Syntax `[ROUTe:]CLOSe?<channel_list>`

Query Response Indicates which relays are closed.

***RST Value** All relays on all modules are set to the open position.

Limits The syntax of a <channel_list> for the VX4330 is described by the following diagram:



A module_name is an ASCII string that has been associated with a relay module in a [ROUTe:]MODule[:DEFine] command. A channel_range may consist of a single <channel_spec> or a range of them. A <channel_spec>s range consists of two <chann el_specs> separated by a colon (:) character.

Related Commands

[ROUTe:]OPEN <channel_list>
[ROUTe:]OPEN:DWELl
[ROUTe:]CLOSe:DWELl
[ROUTe:]CLOSe:MODE
[ROUTe:]CONFigure:JOIN
[ROUTe:]MODule:DEFine
OUTPut:TTLTrg<n>[:STATe]

Description

The action taken when this command is received depends on the close mode that has been assigned to the sections specified in the <channel_list>. The close mode of a section is set to either “mux” or “scan” by the [ROUTe:]CLOSe:MODE command. At power-on or after an *RST command or SYSTem:PREset command, the close mode of all VX4330 sections is set to “mux”. If the close mode of a section has been set to “mux”, then a [ROUTe:]CLOSE command can be used to close one or more relays in that section.

Examples

The following sequence of commands illustrates the operation of sections that have been assigned a close mode of mux. The VX4330 in these examples has been assigned a module name of m1.

Command	Response
output:ttltrg7:State on	Enable VXI TTL trigger 7
route:close:Dwell m1,.5	Assign a close dwell of 0.5 seconds to the scanner
route:close:mode mux,m1,(1:3)	Set the close mode of sections 1 through 3 of the scanner with module name “m1” to “mux”. Open all channels in sections 1 through 3
route:close (@m1(1!1:10!1,1!2:5!2,5!3:10!3))	Close channels 1 through 10 in section 1, channels 1 through 5 in section 2 and channels 5 through 10 in section 3. After these relays are closed, wait 0.5 seconds then pulse VXI TTL trigger 7 and pulse the front panel encode signals corresponding to sections 1, 2, and 3

If the close mode of a section has been set to scan, then a [ROUTe:]CLOSE command will open all relays in that section before the channel specified in the <channel_list> is closed. Only one channel at a time is closed in a section that has been assigned a close mode of scan. Also, if two or more sections that have been assigned a close mode of scan are joined with the [ROUT:]CONFigure:JOIN command, only one channel in that group of sections will be closed at a time. When a channel in one of these sections is specified in a [ROUTe:] CLOSe command, all channels in all of the joined sections will be opened before the specified channel is closed.

The following sequence of commands illustrates the interaction between the [ROUTe:]CLOSe, [ROUTe:]CLOSe:MODE and [ROUTe:]CONFigure:JOIN commands. The SurePath™ Module in these commands has been assigned module name m1.

Command	Response
output:ttltrg7:State on	Enable VXI TTL trigger 7
close:Dwell m1, .5	Assign a close dwell time of 0.5 seconds to the scanner
open:dwell m1, .2	Assign an open dwell time of 0.2 seconds to the scanner
route:close:mode scan,m1,(1:6)	Set the close mode of all sections of the scanner to scan. Open all channels in all six sections of this module
route:configure:join m1,(1:6)	Join the commons of all six sections of the scanner
route:close (@m1(1!1))	Open all channels in sections 1 through 6; wait 0.2 seconds; close channel 1 of section 1; wait 0.5 seconds; then pulse VXI TTL trigger 7 and pulse the front panel encode signals corresponding to section 1
route:close (@m1(10!4))	Open all channels in sections 1 through 6; wait 0.2 seconds; close channel 10 of section 4; wait 0.5 seconds then pulse VXI TTL trigger 7 and pulse the front panel encode signals corresponding to section 4

Enabled VXI TTL triggers are pulsed low for 200 ns after the close dwell time has expired after a channel is closed.

The VX4330 has two front panel encode signals for each section. If the configuration of a section is set to 40-to-1 1-wire, 20-to-1 2-wire, or 10-to-1 4-wire, both encode signals corresponding to that section are pulsed low for 4 ms after a relay is closed in that section, 5 ms after the enabled VXI TTL triggers are pulsed.

If the section is set to the 10-to-1 4-wire independent configuration, the encode signal corresponding to the lower 2-wire common is pulsed when an odd numbered channel is closed in that section. The encode signal corresponding to the upper 2-wire common is pulsed when an even numbered channel is closed in that section.

[ROUTe:]CLOSe:DWELl

Command Syntax `[ROUTe:]CLOSe:DWELl <module_name>,<nrf>`

SurePath™ Module Valid command for all SurePath™ modules.

Query Syntax N/A

***RST Value** The close dwell time of all modules is set to 0 seconds.

Limits The value of the time specified in this command must be between 0 and 6.5535 seconds.

Related Commands [ROUTe:]CLOSe
[ROUTe:]SCAN
OUTPut:TTLTrg<n>[:STATE]

Description Set the time to wait after closing a relay before proceeding and pulsing any enabled VXI TTL trigger signals.

Examples

Command	Response
output:ttltrg1:state on	Enable VXI TTL trigger 1
route:close:dwell m1,.25	Set the close dwell time for the VX4330 to 0.25 seconds
route:open:dwell m1,0.5	Set the open dwell time for the VX4330 to 0.5 seconds
route:configure fwire,m1,(3)	Set the configuration of section 3 to 10-to-1 4-wire. This command causes all channels in section 3 to be opened
route:close (@m1(1!3:10!3))	Close relays 1 through 10 in section 3 of the VX4330, wait 0.25 seconds, then pulse VXI TTL trigger 1
route:configure twire,m1,(1:6)	Set the configuration of all sections to 20-to-1 2-wire. This command causes all channels in all sections to be opened
route:scan (@m1(1!6:20!6))	Define a scan list consisting of relays 1 through 20 in section 6 of the VX4330
trigger:Sequence: source:ttltrg2	Define VXI TTL trigger 2 as the trigger source for the defined scan list
trigger:sequence: delay 1	Set the trigger delay time to 1 second
initiate:immediate	Initiate the scan sequence

After this, each time the VXI TTL trigger 2 is pulsed low, the following sequence of events occurs:

1. One second delay. This is the delay specified in the trigger:Sequence:delay command.
2. The current relay in the scan list is opened.
3. 0.5 second delay. This is the delay specified in the route:open:dwell command.
4. Close the next relay in the scan list.
5. 0.25 second delay. This is the delay specified in the `route:close:dwell` command.
6. Pulse VXI TTL trigger 1 low for 200 ns.
7. Wait 5 ms then pulse the front panel encode signals corresponding to the section of the close relay. The encode signals are pulsed low for 4 ms.

[ROUTe:]CLOSe:MODE

Command Syntax `[ROUTe:]CLOSe:MODE <mode>,<module_name>,<section_list>`

SurePath™ Module Valid command for the VX4330 only

Query Syntax N/A

***RST Value** All sections of the SurePath™ VX4330 Module are set to operate in the mux mode. In this mode one or more channels in a section may be closed at the same time.

Limits This command applies only to the SurePath™ VX4330 Module. The section numbers in the <section_list> argument must be between 1 and 6.

Related Commands [ROUTe:]CONFigure:JOIN
[ROUTe:]CONFigure:DISJoin
[ROUTe:]CONFigure

Description This command is supported only for SurePath™ VX4330 Modules. It sets the mode of operation of sections on a VX4330 to scan or mux. When this command is received, all relays in the sections specified in the <section_list> argument are opened. In the mux mode, one or more relays in a section may be closed at the same time. When a relay is closed in a section that has been set to scan mode, all relays in that section are opened before the relay is closed. If several sections are set to operate in scan mode and the commons of these sections are joined (see the [ROUTe:]CONFigure:JOIN command) then when a relay is closed in any of the sections, all relays in all of the sections are opened first.

Examples

Three additional VX4330 Modules are installed in consecutive slots to the right of this module.

Command	Response
route:close:mode mux m1,(1:6)	Set all six sections of the first VX4330 to mux mode
route:conf twire,m1,(1:6)	Configure all six sections of the first VX4330 to 20-to-1 2-wire scanner/mux
route:close (@m1(1!1:10!1))	Close the first 10 relays in section one of the first VX4330
route:close:mode scan,m3,(1:3)	Set the close mode of sections 1 through 3 of the third VX4330 to "scan"
route:conf:join m3,(1:3)	Join the commons of sections 1 through 3 of the third VX4330
route:conf owire,m3,(1:3)	Set the configuration of sections 1 through 3 of the third VX4330 to 40 to 1 one wire
NOTE: *The previous three commands configure sections 1 through 3 of the third VX4330 as a 120-to-1 1-wire VX4330. The 1-wire common of any of these sections can be used as the 120-to-1 1-wire common.*	
close (@m3(1!1))	Open all channels in sections 1 through 3 of the third VX4330, then close channel 1 of section 1.
close? (@m3(1!1:40!3))	1 0
close (@m3(40!3))	Open all channels in sections 1 through 3 of the third VX4330, then close channel 40 of section 3.
close? (@m3(1!1:40!3))	0 1

[ROUTe:]CONFigure

Command Syntax `[ROUTe:]CONFigure <configuration>,<module_name>,<section_list>`

SurePath™ Module Valid command only for VX4330 SurePath™ module

Query Syntax N/A

***RST Value** N/A

Limits This command is valid only for SurePath™ VX4330 Modules. The section numbers specified in the <section_list> portion of this command must be between 1 and 6.

Related Commands [ROUTe:]CLOSe
[ROUTe:]CLOSe:MODE
[ROUTe:]OPEN

Description Each section of a SurePath™ VX4330 Module may be configured as a 40-to-1 1-wire scanner, or as a 20-to-1 2-wire scanner, or as a 10-to-1 4-wire scanner. Also, each section may be configured as a 10-to-1 4-wire scanner with independent control of the upper and lower halves of the 4-wire common. The <configuration> portion of this command indicates which of these configurations is selected according to the following table:

<configuration>	Configuration
OWIRE	40-to-1 1-wire scanner/mux
TWIRE	20-to-1 2-wire scanner/mux
FWIRE	10-to-1 4-wire scanner/mux
FWIRI	10-to-1 4-wire scanner/mux with independent control of the upper and lower halves of the 4-wire common

The <module_name> argument of the command indicates to which module the command is directed. The specified <module_name> is associated with a relay module with a [ROUTe:]MODule:DEFine command.

The <section_list> argument indicates which sections of the selected module are to be configured. All relays in the section(s) specified in this list are opened when the [ROUTe:]CONFigure command is received.

Examples Three additional VX4330 Modules are installed in consecutive slots to the right of the VX4101.

Command	Response
route:configure fwire,m1,(1:6)	Configure all six sections of the first VX4330 as 10-to-1 4-wire scanners
route:close:mode scan,m1,(1:3)	Set the mode of operation of sections 1, 2, and 3 of the first VX4330 to scan. In this mode, only one channel in a section is closed at a time
route:close:mode mux,m1,(4:6)	Set the mode of operation of sections 4, 5 and 6 of the first VX4330 to mux. In this mode one or more channels in a section may be closed at the same time
rout:conf owire,m3,(1)	Configure section 1 of the third VX4330 as a 40-to-1 1-wire scanner
rout:conf twire,m2,(2,3)	Configure sections 2 and 3 of the second VX4330 as 20-to-1 2-wire scanners
route:configure fwiri, m3,(1:6)	Configure all six sections of the third VX4330 as 10-to-1 4-wire scanners with independent control of upper and lower halves of the 4-wire common
route:close (@m3(1!1))	Connect the lower half of channel 1, section 1, to the lower half of the 4-wire common. In this example, section 1 is assumed to be configured as a 10-to-1 4-wire scanner with independent control of the upper and lower halves of the 4-wire common
route:close (@m3(20!1))	Connect the upper half of the tenth channel of section 1 to the upper half of the 4-wire common. In this example, section 1 is assumed to be configured as a 10-to-1 4-wire scanner with independent control of the upper and lower halves of the 4-wire common

[ROUTe:]CONFigure:DISJoin

Command Syntax `[ROUTe:]CONFigure:DISJoin <module_name>`

SurePath™ Module Valid command only for the VX4330 SurePath™ module.

Query Syntax N/A

***RST Value** All sections on the VX4330 module are disjoined.

Limits N/A

Related Commands

[ROUTe:]CONFigure:JOIN
[ROUTe:]CONFigure
[ROUTe:]CLOSe:MODE

Description

Disconnect the commons of all sections of a module.

Examples

Two additional VX4330 modules are installed in consecutive slots to the right of the VX4101.

Command	Response
route:configure: disjoin m1	Disconnect the commons of the first scanner

[ROUTe:]CONFigure:JOIN

Command Syntax

`[ROUTe:]CONFigure:JOIN <module_name>,<section_list>`

SurePath™ Module

Valid command only for the VX4330 SurePath™ module

Query Syntax

N/A

***RST Value**

All sections on all modules are disjoined.

Limits

The section numbers specified in the <section_list> argument must be between 1 and 6 for VX4330 SurePath™ Modules.

Related Commands

[ROUTe:]CONFigure:DISJoin
[ROUTe:]CONFigure
[ROUTe:]MODE

Description

This command connects the commons of adjacent sections on scanner modules. On VX4330 Modules, if a section that is configured as a 4-wire scanner is joined to a section that is configured as a 1-wire or 2-wire scanner, only the lower half of the 4-wire scan common is connected. If a 4-wire section is joined to another 4-wire section, both halves of the 4-wire scan common are joined. Also for the VX4330, if two or more sections that are set to scan mode are joined, when a [ROUTe:]CLOSe command closes a relay in one of these sections, all relays in the sections are opened before the specified relay is closed.

Examples Two additional VX4330 Modules are installed in consecutive slots to the right of the VX4101.

Command	Response
route:configure:join m1,(1:3)	Connect the commons of sections 1, 2, and 3 on the first VX4330
route:close:mode scan,m2,(1:4)	Set the mode of operation of sections 1 through 4 on the second VX4330 to scan mode. In this mode, only one relay is a section is closed at a time
route:conf twire,m2,(1:4)	Configure sections 1 through 4 on the second VX4330 as 20-to-1 2-wire scanners
route:conf:join m2,(1:4)	Connect the commons of sections 1 through 4 on the second VX4330. Since these sections have been set to operate in the scan mode, and have been configured as 20-to-1 2-wire scanners, these sections now comprise a single 80-to-1 2-wire scanner
route:close (@m2(10!1))	Close channel 10 of section 1 of the second VX4330
route:close (@m2(2!4))	Close channel 2 of section 4 of the second VX4330. Since the first four sections of this module are joined and are set to operate in the scan mode, all relays in sections 1 through 4 of this module are opened before this relay is closed

[ROUTe:]ID?

Command Syntax `[ROUTe:]ID?`

SurePath™ Module Valid command for all SurePath™ modules

***RST Value** N/A

Limits N/A

Related Commands [ROUTe:]MODule:CATalog?

Description This query returns a list of the model numbers of the modules controlled by the SurePath™ Master. The first model number returned is that of the module that the Scanner Master is installed on. Subsequent model numbers are those of modules in consecutive slots to the right of the first module.

Examples

A VX4330, VX4380, and VX4320 are installed in consecutive slots to the right of the slot containing the VX4350. The default module names for the VX4350, VX4380, VX4330, and VX4320 in this configuration are m1, m2, m3, and m4 respectively. These module names may be altered with the [ROUTe:]MODule:DEFine command.

Command	Response
route:id?	VX4350, VX4380, VX4330, VX4320
route:module:Catalog?	“M1”, “M2”, “M3”, “M4”
route:close (@m1(1))	Close relay number 1 on the VX4350
route:open:all m2	Open all relays on the VX4380
route:close (@m3(1!6))	Close relay number 1 in section 6 of the VX4330
route:close (@m4(3!1:3!8))	Close relay number 3 in all eight sections of the VX4320

[ROUTe:]MODule[:DEFine]

Command Syntax

`[ROUTe:]MODule[:DEFine] <module_name>,<nrf>`

SurePath™ Module

Valid command for all SurePath™ modules

Query Syntax

`[ROUTe:]MODule[:DEFine]? <module_name>`

Query Response

See examples.

***RST Value**

Default module names are assigned as follows:

The module that is immediately to the right of the SurePath™ Master.

Consecutive slots to the right of module “M1” are assigned module names “M2”, “M3”, ..., “M11”.

Limits

A module name may consist of up to 12 characters. The name must start with a letter and may consist of alphanumeric characters, underscores, and digits.

Related Commands

[ROUTe:MODule[:DELete]
[ROUTe:MODule:DELete:ALL
[ROUTe:]MODule:CATalog?

Description This command assigns a module name to a relay switching module. This name is used to identify the module in channel lists in [ROUTe:]OPEN, [ROUTe:]CLOSe, and [ROUTe:]SCAN commands.

Examples A VX4380 and VX4330 are installed in consecutive slots to the right of the slot containing the VX4350.

Command	Response
route:module:define gp_switch,1	Assign module name "gp_switch" to the VX4350
route:close (@gp_switch(1:64))	Close all 64 relays on the VX4350
route:module:define matrix,2	Assign module name "matrix" to the VX4380
route:close (@matrix(4!16!3))	Close the relay at row 4, column 16 in section 3 of the VX4380
route:module:define? gp_switch	1
module:define scanner,3	Assign module name "scanner" to the VX4330
open:all scanner	Open all channels on the VX4330
route:conf owire, scanner,(1:6)	Set the configuration of all sections of the VX4330 to 40 to 1 one wire
close (@scanner(30!2))	Close channel 30 of section 2 on the VX4330
module? scanner	3

[ROUTe:]MODule:CATalog?

Command Syntax `[ROUTe:]MODule:CATalog?`

SurePath™ Module Valid command for all SurePath™ modules

***RST Value** N/A

Limits N/A

Related Commands [ROUTe:]MODule:DEFine[:NAME]
[ROUTe:]MODule:DELete[:NAME]
[ROUTe:]MODule:DELete[:ALL]

Description This command returns a list of defined module names.

Examples A VX4380 and VX4330 are installed in consecutive slots to the right of the slot containing the VX4350.

Command	Response
route:module:catalog?	"M1", "M2", "M3"
route:module:define matrix_1,2	Assign module name "matrix_1" to the VX4380
route:module:catalog?	"M1", "MATRIX_1", "M3"
route:module:define? matrix_1	2
route:module:delete matrix_1	Delete module name "matrix_1"
route:module:catalog?	"M1", "M3"

[ROUTe:]MODule:CATalog:SUPPorted?

Command Syntax N/A

Query Syntax `ROUTe:MODule:CATalog:SUPPorted`

Query Response `VX4320, VX4330, VX4350, VX4380`

***RST Value** N/A

Limits N/A

Related Commands N/A

Description This query returns a list of SurePath™ modules that are supported in this version of firmware.

Examples

Command	Action	Returns
inst:sel SurePath	Selects the SurePath™ master	
rout:mod:cat:supp?	Returns Surepath module supported.	VX4320, VX4330, VX4350, VX4380

[ROUTe:]MODule:DELete[:NAME]

Command Syntax `[ROUTe:]MODule:DELete[:NAME] <module_name>`

SurePath™ Module Valid command for all SurePath™ modules.

Query Syntax N/A

***RST Value** N/A

Limits N/A

Related Commands [ROUTe:]MODule:[DEFine][:NAME]
[ROUTe:]MODule:CATalog?
[ROUTe:]MODule:DELete:ALL

Description This command deletes a module name definition. After this command is executed, the specified module name is no longer associated with a relay module.

Examples

Command	Response
route:module:catalog?	“M1”, “M2”, “M3”
route:module:delete m1	Delete module name “M1”
route:module:catalog?	“M2”, “M3”

[ROUTe:]MODule:DELete:ALL

Command Syntax `[ROUTe:]MODule:DELete:ALL`

SurePath™ Module Valid command for all SurePath™ modules.

Query Syntax N/A

***RST Value** N/A

Limits N/A

Related Commands [ROUTe:]MODule[:DEFine]
[ROUTe:]MODule:CATalog?
[ROUTe:]MODule:DELete[:NAME]

Description This command will delete all module name definitions.

Examples

Command	Response
route:module:catalog?	“M1”, “M2”, “M3”
route:module: delete:all	Delete all module names
route:module:catalog?	“ ”

[ROUTe:]OPEN

Command Syntax `[ROUTe:]OPEN <channel_list>`

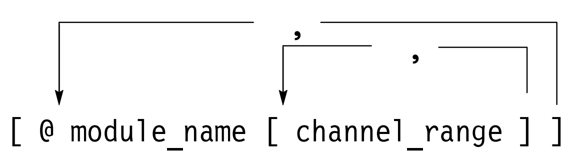
SurePath™ Module Valid command for VX4330, VX4350, and VX4380 SurePath™ modules

Query Syntax `[ROUTe:]OPEN? <channel_list>`

Query Response See examples.

***RST Value** All relays on all modules are set to the open position.

Limits The syntax of a <channel_list> is described by the following diagram:



A module_name is an ASCII string that has been associated with a relay module in a [ROUTe:]MODule[:DEFine] command. A channel_range may consist of a single <channel_spec> or a range of them. A <channel_spec>s range is separated by a colon (:) character.

Command	Response
13!2	Two wire channel 13 in section 2
1!2:20!2	Two wire channels 1 through 20 in section 2
8!3:10!3	Four wire channels 8 through 10 of section 3
19!4	Four wire independent channel 19 in section 4

Related Commands

[ROUTe:]CLOSe <channel_list>
[ROUTe:]MODule[:DEFine]
[ROUTe:]OPEN:DWELl

Description

This command closes the relays specified in the <channel_list> portion of this command.

After these relays are opened, a delay specified in a previously issued [ROUTe:]OPEN:DWELl command is generated. This command is used to assign an open delay to each module controlled by an SurePath™ Master daughter board. If more than one module is specified in the <channel_list> argument of the [ROUTe:]OPEN command, the longest dwell time assigned to any of the specified modules is used.

Examples

Three additional VX4330 Modules are installed in consecutive slots to the right of the first VX4101. The default module names for these three modules are m1, m2, and m3. These module names may be altered with the [ROUTe:]MODule[:DEFine] command.

Command	Response
route:open:dwell m1,.1	Assign an open dwell time of 0.1 seconds to the first VX4330
route:open:dwell m2,.2	Assign an open dwell time of 0.2 seconds to the second VX4330
route:open:dwell m3,.5	Assign an open dwell time of 0.5 seconds to the third VX4330
configure owire,m1,(1:6)	Set the configuration of all six sections of the first VX4330 to 40-to-1 1-wire
configure twire,m2,(1:6)	Set the configuration of all six sections of the second VX4330 to 20-to-1 2-wire
configure fwire,m3,(1:6)	Set the configuration of all six sections of the third VX4330 to 10-to-1 4-wire

Command	Response
route:open (@m1(1:40))	Open channels all channels in section 1 of the first VX4330 then wait 0.1 seconds
route:open (@m2(1!2:10!2))	Open channels 1 through 10 in section 2 of the second VX4330 then wait 0.2 seconds
route:open (@m3(1!1:10!6))	Open all channels in all sections of the third VX4330 then wait 0.5 seconds. This command is equivalent to the command: route:open:all m3
open (@m1(1!1), m2(1!1),m3(1!1))	Open channel 1 in section 1 of all three VX4330 Modules then wait 0.5 seconds

[ROUTe:]OPEN:ALL

Command Syntax `[ROUTe:]OPEN:ALL [module_name]`

SurePath™ Module Valid command for VX4330, VX4350, and VX4380 SurePath™ modules

Query Syntax N/A

***RST Value** All relays on all modules are set to the open position.

Limits N/A

Related Commands [ROUTe:]CLOSe
[ROUTe:]MODule[:DEFine]

Description If a module name is not specified in this command, open all relays on all modules controlled by the SurePath™ Master. If a module name is specified, open all relays on the specified module only. In either case, do not change the state of the configuration relays on the modules.

Examples

Command	Response
ROUTE:OPEN:ALL	Open all relays on all Modules controlled by the SurePath™ Master. Do not change the state of the configuration relays on VX4330 Modules
rout:open:all	Same as the first example
route:open:all gp	Open all relays on the module that has been assigned module name “gp”. See the [ROUTe:]MODule:DEFine command

[ROUTe:]OPEN:DWELl

Command Syntax `[ROUTe:]OPEN:DWELl <module_name>,<nrf>`

SurePath™ Module Valid command for VX4330, VX4350, and VX4380 SurePath™ modules

Query Syntax N/A

***RST Value** The open dwell time of all modules is set to 0 seconds.

Limits The value of the time specified in this command must be between 0 and 6.5535 seconds.

Related Commands [ROUTe:]OPEN
[ROUTe:]SCAN

Description This command sets the time to wait after opening a relay before proceeding.

Examples

Command	Response
output:ttltrg1:state on	Enable VXI TTL trigger 1.
route:close:dwell m1,.25	Set the close dwell time for the VX4330 to 0.25 seconds.
route:open:dwell m1,0.5	Set the open dwell time for the VX4330 to 0.5 seconds.
route:configure twire,m1,(1:6)	Set the configuration of all sections to 20 to 1 2-wire. This command causes all channels in all sections to be opened.
route:scan (@m1(1!6:20!6))	Define a scan list consisting of relays 1 through 20 in section 6 of the VX4330.
trigger:Sequence: source ttltrg2	Define VXI TTL trigger 2 as the trigger source for the defined scan list.
trigger:sequence:delay 1	Set the trigger delay time to 1 second
initiate:immediate	Initiate the scan sequence

After this sequence, each time the VXI TTL trigger 2 is pulsed low, the following sequence of events occur:

1. One second delay. This is the delay specified in the trigger:Sequence:delay command.
2. The current relay in the scan list is opened.

3. 0.5 second delay. This is the delay specified in the route:open:dwell command.
4. Close the next relay in the scan list.
5. 0.25 second delay. This is the delay specified in the route:close:dwell command.
6. Pulse VXI TTL trigger 1 low for 3 ns.
7. Wait 5 ms then pulse the front panel encode signals corresponding to the section of the close relay. The encode signals are pulsed low for 4 ms.

[ROUTe:]PFAil

Command Syntax `[ROUTe:]PFAil <action_at_powerfail>`

SurePath™ Module Valid command for VX4330, VX4350, and VX4380 SurePath™ modules

Query Syntax N/A

***RST Value** All relays on all modules are opened when power is removed from the VXI chassis.

Limits N/A

Related Commands N/A

Description This command specifies the state that all latching relays on all modules controlled by the SurePath™ Master are to be placed in when power is removed from the VXI chassis containing the modules. <action_at_powerfail> must be OPEN or SAME. If OPEN is specified, all latching relays are opened at power fail. If SAME is specified, all latching relays are left in their current state at powerfail.

***NOTE**. VXI chassis +5 V power is maintained for 4 ms after ACFAIL is asserted, in compliance with VXI Specifications. This allows for orderly system shutdown and implementation of the PFAil OPEN option.*

Examples

Command	Response
route:pfail same	Leave all latching relays in their current state at powerfail
route:pfail open	Open all latching relays at powerfail

[ROUTe:]SCAN

Command Syntax

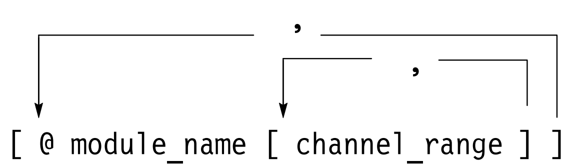
```
[ROUTe:]SCAN <channel_list>
```

SurePath™ Module Valid command for VX4330, VX4350 and VX4380 modules

Query Syntax N/A

***RST Value** No scan list is defined.

Limits The syntax of a <channel_list> is described by the following diagram:



A module_name is an ASCII string that has been associated with a relay module in a [ROUTe:]MODule[:DEFine] command. A channel_range may consist of a single <channel_spec> or a range of them.

Related Commands

TRIGger[:SEQuence]:SOURce
TRIGger[:SEQuence]:COUNt
TRIGger[:SEQuence]:DELay
TRIGger[:SEQuence]:[IMMediate]
INITiate:CONTinuous
INITiate[:IMMediate]
[ROUTe:]CLOSe:DWELl
[ROUTe:]OPEN:DWELl

Description This command defines a list of scan relay closures to sequence. When the ROUTe:SCAN <channel_list> command is received, all relays in this list are opened. In addition to defining a scan list, a trigger source must be specified using the TRIGger[:SEQuence]:SOURce command. Trigger events are not recognized until triggers are armed by a INITiate[:IMMediate] or INITiate:CONTinuous command.

When the first trigger event is detected, the first relay in the scan list is closed. When the second trigger event is detected, the first relay is opened and the second relay is closed. When the n^{th} trigger event is detected, the $(n–1)^{th}$ relay is opened and the n^{th} relay is closed. The act of opening the $(n–1)^{th}$ relay and closing the nth relay is called sequencing the scan list. At any given time after the first trigger event is detected, only one relay in the scan list is closed.

The TRIGger[:SEQuence]:COUNt command may be used to specify the number of times to sequence through the entire scan list. The TRIGger [:SEQuence]:DELay, [ROUTe:]CLOSe:DWELl, and [ROUTe:]OPEN:DWELl commands may optionally be used to specify the time to wait after a trigger event is detected, a relay is closed or a relay is opened. A TRIGger[:SEQuence] :IMMediate command causes the scan list to be sequenced without the delay specified by a previously issued TRIGger[:SEQuence]:DELay command.

Examples

A VX4380, VX4330, and VX4350 are installed in consecutive slots to the right of the slot containing the VX4101.

Command	Response
route:module:define gp,1	Assign module name “gp” to the VX4350
route:module:define matrix,2	Assign module name “matrix” to the VX4380
route:module:define scanner,3	Assign module name “scanner” to the VX4380
route:configure twire,scanner,(1:6)	Set the configuration of all sections of the VX4330 to 20-to-1 2-wire. This command causes all channels in all sections to be opened
route:scan (@gp(1:64), matrix(1!1!1, 2!10!3), scanner(1!1:20!1))	Define a scan list consisting of relays 1 through 64 on the VX4350, relays at row 1, column 1 of section 1 and row 2, column 10 of section 3 of the VX4380 and relays 1 through 20 of section 1 of the VX4330
trigger:sequence: source immediate	Define a trigger source of “immediate”. This means to sequence through the scan list without waiting for a trigger event
trigger:sequence:count 5	Sequence through the entire scan list five times
route:close:dwell gp,.5	Wait 0.5 seconds after closing a relay on the VX4350
initiate:immediate	Begin sequencing through the scan list
*OPC	Set the Operation Complete bit of the Standard Event Status register after sequencing through the scan list five times

Command	Response
*wai; init:cont	Wait until the scan list has been sequenced through five times, then begin sequencing through the list repeatedly until an ABORt command is received
abort	Quit sequencing through the scan list and place the trigger subsystem in the idle state

[ROUTe:]SCAN:RATE

Command Syntax

`[ROUTe:]SCAN:RATE <scan_rate>,<module_name>`

SurePath™ Module

Valid command for the VX4330 only

***RST Value**

The VX4330 scan rate is set to normal. The minimum time to sequence to the next channel in a scan list is approximately 16 ms. The SurePath™ master on the VX4101 runs on a multitasking system. The accuracy of the scan time depends on the load placed on the VX4101. Each time a channel is opened or closed, front panel encode signals are generated, and the control signals applied to all relays on the module are verified.

Limits

This command applies only to VX4330 Modules.

Related Commands

ABORt
INITiate[:IMMediate]
INITiate:CONTinuous
OUTPut:TTLTrg<n>[:STATe]
[ROUTe:]CLOSe:DWELl
[ROUTe:]OPEN:DWELl
[ROUTe:]SCAN
TRIGger:[SEQuence]:COUNt
TRIGger[:SEQuence]:DELay
TRIGger[:SEQuence][:IMMediate]
TRIGger[:SEQuence]:SOURce

Description

<scan_rate> is either NORMal or FAST.
<module_name> is the module name assigned to the VX4330.

This command controls the maximum rate at which VX4330 channels in a scan list can be sequenced. If a <scan_rate> of NORMal is specified in the [ROUTe:]SCAN:RATE command, it takes approximately 16 ms plus the sum of the close dwell and open dwell times assigned to the VX4330 and the trigger delay time to:

open the VX4330 channel that is currently closed,

close the next VX4330 channel, and

pulse the front panel encode signal(s) corresponding to the closed channel.

The control signals applied to all relays on the VX4330 Module are verified after the current channel is opened and after the next channel in the scan list is closed.

If a <scan_rate> of FAST is specified in the [ROUTe:]SCAN:RATE command, it takes approximately 8 milliseconds plus the sum of the close dwell and open dwell times assigned to the VX4330 and the trigger delay time to:

open the channel that is currently closed, and

to close the next channel in the scan list.

The VX4330 front panel encode signals are disabled and the relay control signals are not verified after a channel is opened or closed.

Examples

Command	Response
scan (@m1(1!1:10!6))	Define a scan list consisting of channels 1 through 10 in each section of the VX4330 Module
trigger:count 1	Set the number of times to sequence through the scan list (after an INITiate[:IMMediate] command is received) to 1
trigger:source bus	Enable a VXI TRIGGER command as a trigger source
output:ttltrg1:State on	Enable VXI TTL trigger 1 to be pulsed each time a channel is closed
close:Dwell m1,0	Set the close dwell time associated with the VX4330 to 0
open:dwell m1,0	Set the open dwell time associated with the VX4330 to 0
trig:del 0	Set the trigger delay time to 0
scan:rate norm,m1	Set the scan rate of the VX4330 to normal
init	Initiate the scan list. Sequence through the entire scan list one time. Each time a VXI TRIGGER command is sent to the SurePath™ Master VXI Interface, it takes 16 ms to open the currently closed channel, close the next channel, pulse the corresponding VX4330 front panel encode signal and pulse VXI TTL trigger 1. The control signals applied to each relay on the scanner are verified each time a channel is opened or closed.
abort	Abort the scan list
scan:rate fast,m1	Set the scan rate of the VX4330 to FAST
trigger:sour immediate	Set the trigger source to immediate

Command	Response
initiate:continuous	Initiate the scan list. Sequence through the entire scan list repeatedly until an ABORt command is received. It takes 8 ms to open the currently closed channel, close the next channel and pulse VXI TTL trigger 1. The VX4330 front panel encode signals are set to a high logic level and the relay control signals applied to the scanner relays are not verified each time a channel is opened or closed
abort	Abort the scan list

STATus:OPERation:CONDition?

Command Syntax `STATus:OPERation:CONDition?`

SurePath™ Module Valid command for all SurePath™ modules

***RST Value** 0

Limits N/A

Related Commands N/A

Description This query returns the contents of the SCPI Status Operation Condition register. For the SurePath™ Master, the value of this register is always equal to 0. The bit definitions are:

Bit	Definition	Function
0	Calibrating	Not used
1	Settling	Set when the instrument changes its function or range. Cleared when the all circuitry has settled
2	Ranging	Not used
3	Sweeping	Set when a scan list operation begins. Cleared when the list has been completely traversed or an abort occurs
4	Measuring	Not used
5	Triggering	Set when the instrument is waiting for an arm signal. Cleared when the arm is received
6	Arming	Not used
7	Correcting	Not used

Bit	Definition	Function
8	Testing (User 1)	Not used
9	Aborting (User 2)	Set when the instrument is in the process of aborting an operation. Cleared when the abort is complete
10	User 3	Not used
11	User 4	Not used
12	User 5	Reserved
13	Instrument Summary	Not used
14	Program Running	Not used
15	Reserved	Always 0

Examples

Command	Response
status:operation: condition?	00000

Trigger Commands for the SurePath™ Modules

The trigger-related commands for the SurePath™ Modules are as follows:

```
TRIGger[:SEQuence]:COUNt
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:IMMediate
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:DELay
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:ECOunt
TRIGger ([: SEQuence1] |:STARt)[:LAYer]:MODE
```

TRIGger[:SEQuence]:COUNt

Command Syntax `TRIGger[:SEQuence]:COUNt`

SurePath™ Module Valid command for VX4330, VX4350 and VX4380 SurePath™ modules

Query Syntax N/A

***RST Value** 1

Limits The count specified in this command must be between 1 and 65535.

Related Commands ABORt, [ROUTe:]SCAN,
TRIGger[:SEQuence]:SOURce,
TRIGger[:SEQuence][:IMMediate],
TRIGger[:SEQuence]:DELay

Description This command specifies the number of times to sequence through an entire scan sequence when an INITiate[:IMMediate] command is received.

Examples

Two VX4330 Modules are installed in the slots next to the VX4101. The default module names “m1” and “m2” have been assigned.

Command	Response
route:conf twire,m1,(1:6)	Set the configuration of all six sections of the first VX4330 to 20-to-1 2-wire
route:conf fwire,m2,(1:3)	Set the configuration of sections 1 through 3 of the second VX4330 to 10-to-1 4-wire
configure owire,m2,(4:6)	Set the configuration of sections 4 through 6 of the second VX4330 to 40-to-1 1-wire
scan (@m1(1!5:20!5),m2(1!3:10!3,1!6:40!6))	Define a scan list consisting of 2-wire channels 1 through 20 in section 5 of the first VX4330, 4-wire channels 1 through 10 in section 3 of the second VX4330 and 1-wire channels 1 through 40 in section 6 of the second VX4330
trigger:sequence:count 2	Define the number of times to sequence through the scan list before setting the state of the trigger subsystem back to the idle state
trigger:sequence:source immediate	Sequence through the scan list without waiting for a trigger event after an INITiate[:Immediate] command is received
initiate:immediate	Initiate the scan sequence. Sequence through the entire scan list two times

TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?

Command Syntax

N/A

Query Syntax

`TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog[:ALL]?`

Command Class

Instrument

Query Response

`HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM`

***RST Value**

N/A

Limits

N/A

Related Commands

TRIGger:SOURce:CATalog:DELayable?
TRIGger:SOURce:CATalog:FIXed?
TRIGger:SOURce
TRIGger:FIRe4
TRIGger:TIMer

Description

Lists all trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM

TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?

Command Syntax

N/A

Query Syntax

`TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:DELayable?`

Command Class

Instrument

Query Response

```
BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5,
TTLTRG6, TTLTRG7, COMMAND0 COMMAND1, COMMAND2, COMMAND3,
COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM
```

***RST Value**

N/A

Limits

N/A

Related Commands

TRIGger:SOURce:CATalog
TRIGger:SOURce:CATalog:FIXed?
TRIGger:SOURce
TRIGger:FIRe4 TRIGger:TIMer

Description

Lists all delayable trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT:DEL?	BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM

TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?

Command Syntax N/A

Query Syntax `TRIGger ([: SEQuence1] |:STARt)[:LAYer]:SOURce:CATalog:FIXed?`

Command Class Instrument

Query Response `HOLD,IMMEDIATE,TIMER`

***RST Value** N/A

Limits N/A

Related Commands
TRIGger:SOURce:CATalog
TRIGger:SOURce:CATalog:DELayable?
TRIGger:SOURce
TRIGger:FIRe4
TRIGger:TIMer

Description Lists all fixed trigger sources available for use with the TRIG:SOUR command.

Examples

Command	Response
TRIG:SOUR:CAT:FIX?	HOLD,IMMEDIATE,TIMER

TRIGger ([: SEQuence1] |:STARt) [:LAYer]:SOURce

Command Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:SOURce <source>`

Query Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:SOURce?`

Command Class Instrument

Query Response `<current source>`

***RST Value** IMMEDIATE

Limits BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, SUREPATH, DMM, COUNTER, CTR_EXTARM

Related Commands TRIGger:SOURce:CATalog?
TRIGger:FIRe4
TRIGger:TIMer

Description Selects or queries the trigger source to be used when the instrument is initiated.

Examples See TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

TRIGger ([: SEQuence1] |:STARt) [:LAYer]:IMMediate

Command Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:IMMediate`

Query Syntax N/A

Command Class Instrument

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description

Causes a one time entry into the triggered state without receiving the specified trigger. This command is often used to prepare for cases such as setting up a scan list measurement. The example below shows an example of an automated integrated trigger scan of four resistance values in autorange.

Examples

Command	Response
TRIG:SOUR:CAT?	HOLD, IMMEDIATE, BUS, TTLTRG0, TTLTRG1, TTLTRG2, TTLTRG3, TTLTRG4, TTLTRG5, TTLTRG6, TTLTRG7, COMMAND0, COMMAND1, COMMAND2, COMMAND3, COMMAND4, TIMER, SUREPATH, DMM, COUNTER, CTR_EXTARM
INST:SEL DMM	Select DMM
CONF:ARR:RES 4	Configure for 4 resistance readings
TRIG:MODE:ONCE	Require a separate trigger for each
TRIG:SOUR SUREPATH	Select DMM Trigger source as SurePath™
INIT	Intiate and wait for trigger
INST:SEL SUREPATH	Select SurePath™ instrument
ROUT:SCAN (@m1 (1!1!1,1!2!1,1!3!1,1!4!1))	Program a 4 channel scan list for the VX4380
ROUT:CLOS:DWEL m1, .3	0.3 second delay from channel close to trigger
TRIG:SOUR DMM	Select SurePath™ trigger source as DMM
INIT	Initiate and wait for trigger
INIT:IMM	Self trigger first channel to get started
FETC:COUN?	0 (no measurements available initially).Wait for measure completion. May take a few seconds.
FETC:COUN?	4
FETC?	#252+1.15123E+02, +1.15456E+03, +1.15789E+04, +1.16000E+05

TRIGger ([: SEQuence1] |:STARt) [:LAYer]:DELay

Command Syntax

`TRIGger ([: SEQuence1] |:STARt) [:LAYer]:DELay <delay in seconds>`

Query Syntax

`TRIGger ([: SEQuence1] |:STARt) [:LAYer]:DELay?`

Command Class

Instrument

Query Response `<delay in seconds>`

***RST Value** 0 seconds (pass-through)

Limits 0 = pass through, 1 μs – 65.536 ms in 1 μs steps

Related Commands TRIGger:ECOunt

Description Specifies a time delay to occur after receipt of a trigger command before actually triggering. If the selected trigger source is fixed, this command will have no effect on the instrument triggering. This command always zeros the event count delay, so specifying a delay of zero places the trigger subsystem in pass-through mode. In this mode, the instrument triggers immediately upon receipt of a trigger command.

Examples See TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

TRIGger ([: SEQuence1] |:STARt) [:LAYer]:ECOunt

Command Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:ECOunt <triggers to count>`

Query Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:ECOunt?`

Command Class Instrument

Query Response `<triggers to count>`

***RST Value** 0 triggers (pass-through)

Limits 0 = pass through, 1 to 65,536 triggers

Related Commands TRIGger:DELay

Description Specifies the number of triggers to count prior to triggering. Upon receipt of trigger N (where N is the number specified in the command), the instrument will enter the triggered state. If the trigger source selected is fixed, this command will have no effect on the instrument triggering. This command always zeros the delay by time parameter, so specifying an event count of zero places the trigger subsystem in pass-through mode. In this mode, the instrument triggers immediately upon receipt of a trigger.

Examples See the TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

TRIGger ([: SEQuence1] |:STARt) [:LAYer]:MODE

Command Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:MODE ONCE|ALL`

Query Syntax `TRIGger ([: SEQuence1] |:STARt) [:LAYer]:MODE?`

Command Class Instrument

Query Response `Current state: ONCE|ALL`

***RST Value** ALL

Limits ONCE, ALL

Related Commands N/A

Description When an instrument has been configured for some type of array measurement, this command specifies whether the instrument will perform one or all operations when a trigger is received. If set to ALL mode, then all operations will be completed upon receipt of one trigger condition. If set to ONCE mode, then the instrument will perform one operation and enter the initiated state a second time. This will continue until the specified number of triggers has been received and the specified number of operations has been completed.

Examples See TRIGger([: SEQuence1] |:STARt) [:LAYer]:IMMediate command for an example.

Status and Events

Status and Event Reporting System

The VX4101 Status system uses a hierarchical set of registers to provide status information on all three instruments, as well as on the VX4101 itself. The structure of each instrument register is composed of a set of three registers and two transition registers as defined in the SCPI standard. The Standard Event Status Register and Status Byte Register are defined by the IEEE 488.2 standard. For the purposes of this discussion, if a register bit is referred to as being set, its value is a positive logic one (1). A bit which is cleared has a value of positive logic zero (0).

In order to maintain conformance to the SCPI standard, the VX4101 has a SCPI Questionable status register. This register is typically used to convey information about the quality of an instrument operation. Because of the binary nature of these indications, it is often difficult to tell what is suspect about a given operation. To solve this problem, the VX4101 places warning messages into the error/event queue instead of using these registers. These registers are identical in structure to the (OSR) Operational Status Registers (simply substitute the keyword QUEStionable for OPERational), but are always in the cleared state.

The following pages contain an overview of the VX4101 status registers and a description of the associated command set.

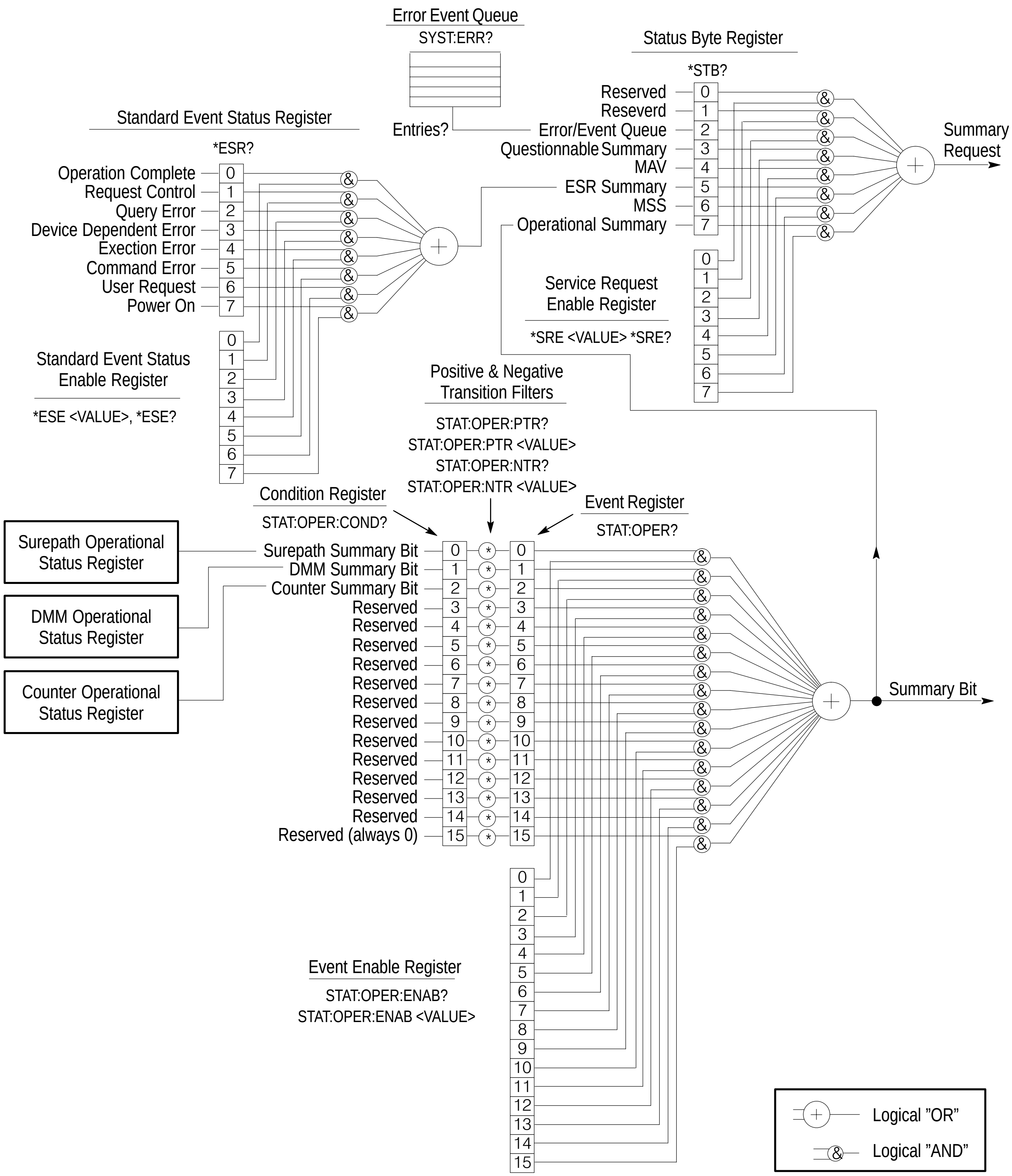


Figure 4–1: VX4101 Standard Registers

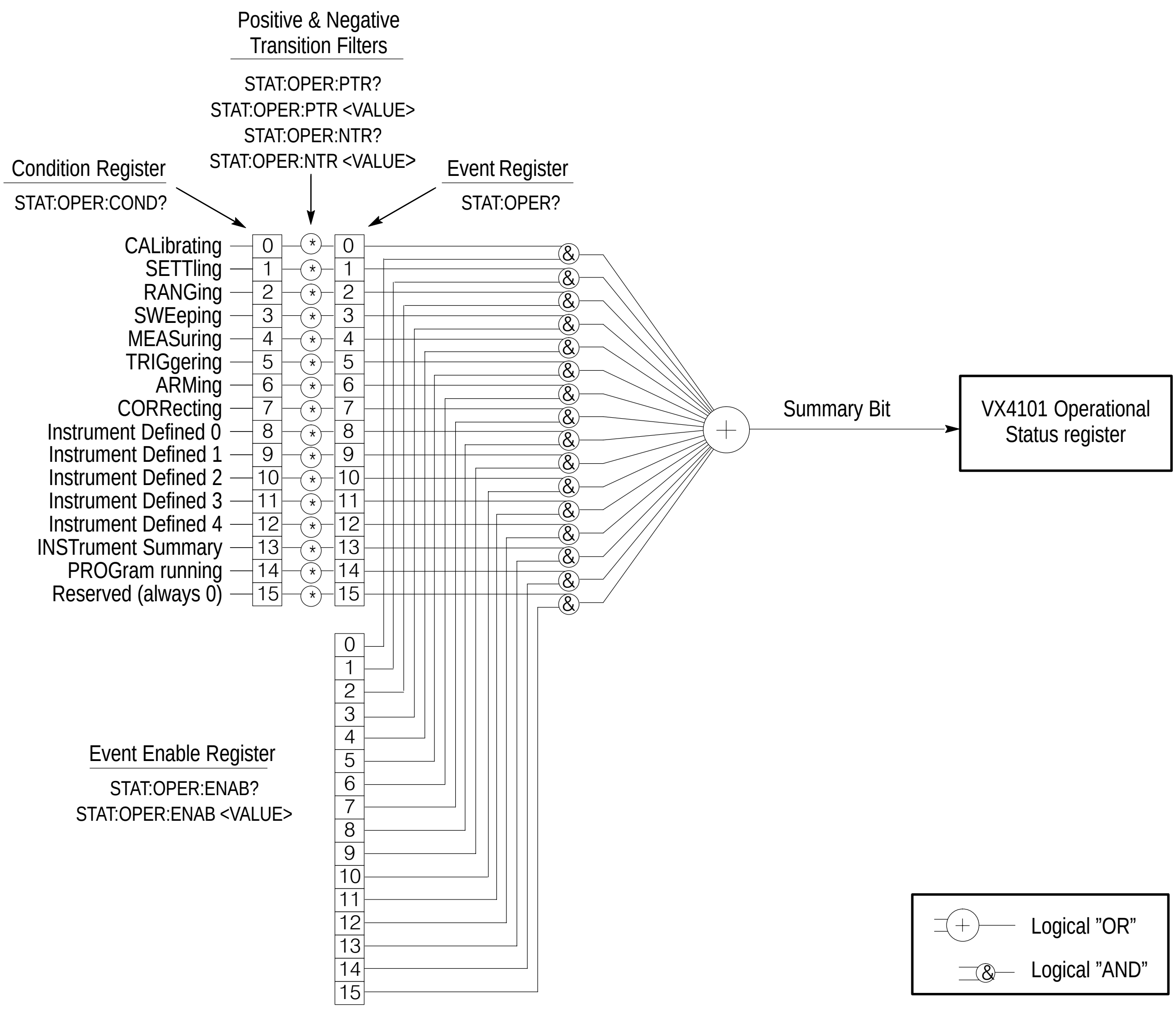


Figure 4–2: Instrument Operational Status Registers

Table 4–1: Instrument Operational Status Register

Bit #	Name	Description
0	CALibrating	Instrument performing calibration
1	SETTling	Waiting for controls to settle
2	RANGing	Instrument is changing range
3	SWEeping	Sweep is in progress
4	MEASuring	A triggered measurement is in progress
5	TRIGgering	Instrument is waiting for trigger condition

Table 4–1: Instrument Operational Status Register (Cont.)

Bit #	Name	Description
6	ARMing	Instrument is waiting for arm condition
7	CORRecting	Instrument is performing a correction
8	Instrument Defined 0	Self-Test in progress
9	Instrument Defined 1	Abort in progress
10	Instrument Defined 2	Reserved by VX4101
11	Instrument Defined 3	Reserved by VX4101
12	Instrument Defined 4	Reserved by VX4101
13	INSTrument summary	Not used by VX4101
14	PROGram running	User-defined program is running
15	Reserved	Always zero

The Instrument Operational Status Register structure appears in the above table. An instrument uses this register during normal operation to record its current state. As mentioned above, this register is composed of three subregisters and two transition filters.

The Condition Register is updated by the instrument firmware in real-time. Bits are set or cleared to identify what action the instrument is currently performing.

***NOTE**. The STATus:OPERational:CONDitional? query is described in the individual instrument sections.*

Note that some bits may not be used by some instruments and that there are five bits which are instrument defined. See the individual instrument documentation to determine which bits are used.

Following the condition register are two sets of transition filters. The Positive Transition Filter is used to specify which bits of the Condition Register are only of interest when they go from the Cleared to the Set state. The Negative Transition Filter is also used to specify which bits of the Condition Register are only of interest when they go from the Set to the Cleared state. For information on any transition of a bit, both filters would be set.

Any transition which successfully passes through the transition filters sets the corresponding bit in the event register. It is important to remember that for a bit with the Negative Transition Filter set, a event in the Condition Register would cause the same bit in the Event Register to be set, not cleared. Reading this register causes it to be cleared.

Table 4–2: VX4101 Operational Status Register

Bit #	Name	Description
1	Surepath Summary	Summary bit from Surepath OSR
2	DMM Summary	Summary bit from DMM OSR
3	Counter Summary	Summary bit from Counter OSR
4	Reserved	Reserved by VX4101
5	Reserved	Reserved by VX4101
6	Reserved	Reserved by VX4101
7	Reserved	Reserved by VX4101
8	Reserved	Reserved by VX4101
9	Reserved	Reserved by VX4101
10	Reserved	Reserved by VX4101
11	Reserved	Reserved by VX4101
12	Reserved	Reserved by VX4101
13	Reserved	Reserved by VX4101
14	Reserved	Reserved by VX4101
15	Reserved	Always zero

The third register is the Event Enable Register. Bits are set or cleared in this register to indicate which bits of the Event Register should be propagated to the summary bit. The summary bit is the logical OR of each bit in the Event Register logically ANDed with the corresponding bit in the Event Enable Register. The summary bit becomes a single bit of another register in the next level of the register hierarchy.

In the case of the VX4101, the next level in the hierarchy is the VX4101 Operational Status Register. This register has the same structure as the Instrument Operational Status Register with the following exception:

> The summary bit of the VX4101 Operational Status register propagates to a bit in the IEEE 488.2 Status Byte Register.

The IEEE 488.2 Standard Event Status Register provides general status on the VX4101 and all instruments and is at the same hierarchical level as the VX4101 Operational Status Register. Its structure is simpler than the operational status register, in that the Condition Register and transition filters do not exist. This register is cleared when read (by a *ESR? query) and when a *CLS command is received. The register is eight bits wide as shown in the table below.

Table 4–3: Status Byte Register

Bit #	Name	Description
0	Operation Complete	Set in response to *OPC?
1	Request Control	Not Used by VX4101
2	Query Error	Error occurred during query
3	Device Specific Error	Any error besides query, command or execution
4	Execution Error	Error in command or query parameters
5	Command Error	Command or query syntax error
6	User Request	Not used by VX4101
7	Power On	Indicates initial power on condition

Table 4–4: IEEE 488.2 Standard Event Status Register

Bit #	Name	Description
0	Reserved	Not Used by VX4101
1	Reserved	Not Used by VX4101
2	Error/Event Queue	Set when Error/Event Queue has one or more entries
3	Questionable Summary	Always zero
4	MAV	Set when a message is available for a VXIbus read
5	ESR Summary	Summary bit from IEEE 488.2 Standard Event Status Register
6	MSS	Master Summary Status
7	Operational Summary	Summary bit from VX4101 Operational Status Register

The summary bit from the IEEE 488.2 Standard Event Status Register propagates to the IEEE 488.2 Status Byte Register. The IEEE 488.2 Standard Event Status Enable Register is used to specify which events compromise the summary bit.

The IEEE 488.2 Status Byte Register is the top of the status hierarchy. Like the IEEE 488.2 Event Status Register, this register does not have a Condition Register or transition filters.

The IEEE 488.2 Service Request Enable Register is used to specify which bits of the IEEE 488.2 Status Byte Register propagate to the summary bit. This summary bit is a special case in that when it is set, the VX4101 generates a VXIbus Service Request.

*CLS

Command Syntax `*CLS`

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Clears all event status registers and queues.

*ESE

Command Syntax `*ESE`

Query Syntax `*ESE?`

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits 0–255

Related Commands N/A

Description Sets or queries the contents of the IEEE 488.2 Standard Event Status Enable Register. The contents of this register are unaffected by a register read.

*ESR

Command Syntax N/A

Query Syntax `*ESR?`

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Queries the contents of the IEEE 488.2 Standard Event Status Register. The contents of this register are cleared after the read is complete.

*SRE

Command Syntax `*SRE`

Query Syntax `*SRE?`

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits 0–255

Related Commands N/A

Description Sets or queries the contents of the IEEE 488.2 Service Request Enable Register. The contents of this register are unaffected by a register read.

STATus

Command Syntax `STATus : PRESet`

Query Syntax N/A

Command Class Global

Query Response N/A

***RST Value** N/A

Limits N/A

Related Commands *CLS

Description This command clears the enable registers of all Operational Status Registers, sets all Positive Transition Filters, and clears all Negative Transition Filters. This command has no effect on the IEEE488.2 Standard Event Status Register, IEEE 488.2 Standard Event Status Enable Register, IEEE 488.2 Status Byte Register, or IEEE 488.2 Service Request Enable Register.

STATus : QUEue [:NEXT]?

Command Syntax N/A

Query Syntax `STATus : QUEue [:NEXT]?`

Command Class Global

Query Response `<error #>,<error message>` or `0,"No error"` if no error.

***RST Value** N/A

Limits N/A

Related Commands SYST:ERR?

Description Returns next item from error/event queue in FIFO order.

STATus : QUEue : ENABle <numeric list>

Command Syntax `STATus : QUEue : ENABle <numeric list>`

Query Syntax `STATus : QUEue : ENABle?`

Command Class Global

Query Response Currently enabled range as a numeric list, NR1 format

***RST Value** (–499:–100)

Limits TBD

Related Commands N/A

Description Allows user to specify by error number which errors and events should be placed in the error/event queue.

STATus : OPERation [:EVENt]?

Command Syntax N/A

Query Syntax `STATus : OPERation [:EVENt]?`

Command Class Instrument

Query Response Register contents, NR1 format, 0–32767

***RST Value** N/A

Related Commands N/A

Description Returns contents of Operational Event Register for currently selected instrument. Register contents are cleared after read completed.

STATus : OPERation : ENABle

Command Syntax `STATus : OPERation : ENABle`

Query Syntax `STATus : OPERation : ENABle?`

Command Class Instrument

Query Response Register contents, NR1 format, 0–32767

***RST Value** N/A

Limits 0-32767, all enabled

Related Commands N/A

Description Set or query the Operational Enable Register for the currently selected instrument. Setting a bit in this register allows the corresponding bit in the Operational Event Register to propagate to the summary bit.

STATus : OPERation : PTRansition

Command Syntax `STATus : OPERation : PTRansition`

Query Syntax `STATus : OPERation : PTRansition?`

Command Class Instrument

Query Response Filter contents, NR1 format, 0–32767

***RST Value** 32,767

Limits 0–32767

Related Commands N/A

Description Set or query the Operational Positive Transition Filter for the currently selected instrument. Setting a bit in this filter latches zero to one transitions of the corresponding bit in the Operational Condition Register into in the Operational Event Register.

STATus : OPERation : NTRansition

Command Syntax `STATus : OPERation : NTRansition`

Query Syntax `STATus : OPERation : NTRansition?`

Command Class Instrument

Query Response Filter contents, NR1 format, 0–32767

***RST Value** 0

Limits 0–32767

Related Commands N/A

Description Set or query the Operational Negative Transition Filter for the currently selected instrument. Setting a bit in this filter latches one to zero transitions of the corresponding bit in the Operational Condition Register into the Operational Event Register.

STATus : QUEStionable [:EVENt]?

Command Syntax N/A

Query Syntax `STATus : QUEStionable [:EVENt]?`

Command Class Instrument

Query Response 0

***RST Value** N/A

Limits 0–32767

Related Commands N/A

Description Returns contents of Questionable Event Register for currently selected instrument. Register contents are cleared after read completed.

STATus : QUEStionable : CONDition?

Command Syntax N/A

Query Syntax `STATus : QUEStionable : CONDition?`

Command Class Instrument

Query Response 0

***RST Value** 32767 (all enabled)

Limits N/A

Related Commands N/A

Description Returns contents of Questionable Condition Register for currently selected instrument. Register contents are unaffected by this query.

STATus : QUEStionable : ENABle

Command Syntax `STATus : QUEStionable : ENABle`

Query Syntax `STATus : QUEStionable : ENABle?`

Command Class Instrument

Query Response Register contents, NR1 format, 0–32767

***RST Value** 32767 (all enabled)

Limits 0–32767

Related Commands N/A

Description Set or query the Questionable Enable Register for the currently selected instrument. Setting a bit in this register allows the corresponding bit in the Questionable Event Register to propagate to the summary bit.

STATus : QUEStionable : PTRansition

Command Syntax `STATus : QUEStionable : PTRansition`

Query Syntax `STATus : QUEStionable : PTRansition?`

Command Class Instrument

Query Response Filter contents, NR1 format, 0–32767

***RST Value** 32767

Limits 0–32767

Related Commands N/A

Description Set or query the Questionable Positive Transition Filter for the currently selected instrument. Setting a bit in this filter latches zero to one transitions of the corresponding bit in the Questionable Condition Register into in the Questionable Event Register.

STATus : QUEStionable : NTRansition

Command Syntax `STATus : QUEStionable : NTRansition`

Query Syntax `STATus : QUEStionable : NTRansition?`

Command Class Instrument

Query Response Filter contents, NR1 format, 0–32767

***RST Value** 0

Limits 0–32767

Related Commands N/A

Description Set or query the Questionable Negative Transition Filter for the currently selected instrument. Setting a bit in this filter latches one to zero transitions of the corresponding bit in the Questionable Condition Register into the Questionable Event Register.

*STB?

Command Syntax N/A

Query Syntax `*STB?`

Command Class Global

Query Response Register contents, NR1 format, 0–255

***RST Value** N/A

Limits N/A

Related Commands N/A

Description Queries the contents of the IEEE 488.2 Status Byte Register. The contents of this register are cleared after the read is complete.

Status Subsystem Example

The following is an example of using the Status Subsystem to receive a service request at the end of a DMM array measurement. The DMM sets bit 4 of the Operational Condition Register while a measurement is in progress and clears the bit when the measurement has completed (or aborted). The service request will therefore depend upon a negative transition of this bit.

When using the Status subsystem commands for the VX1401 the VX4101 must be selected using “INST:NSEL 0” commands. It is unnecessary when using all other global commands to actually select the VX4101.

NOTE. *Successful interrupt generation is dependent upon following the sequence outlined in this example.*

Table 4–5: Status Subsystem and Service Requests

Command/Query	Response	Comments
*RST		Reset
*CLS		Clear all event status registers and queues
INST:SEL DMM		Select DMM
STAT:OPER:PTR 0		Don't report any positive transitions
STAT:OPER:NTR 16		Report negative transitions of bit 4
STAT:OPER:ENAB 16		Propagate bit 4 into summary
STAT:OPER?	0	Clear event register by reading
CONF:ARR:VOLT:DC 150		Take approximately 30 seconds of measurements
INST:NSEL 0		Select VX4101
STAT:OPER:PTR 2		Report positive transitions of bit 1
STAT:OPER:NTR 0		Don't report any negative transitions
STAT:OPER:ENAB 2		propagate bit 1 into summary
STAT:OPER?	0	Clear event register by reading
*SRE 128		Generate SRQ on Bit 8
*STB?	16	Clear status register by reading (except MAV)
INST:SEL DMM		Select DMM
INIT		Initiate measurement
<WAIT FOR SRQ>		Approximately 30 seconds
FETC:COUN?	150	All measurements taken
INST:NSEL 0		Select VX4101
*STB?	208	Operational Summary bit, MSS bit, MAV bit
STAT:OPER?	2	Bit 1 (DMM Summary)
INST:SEL DMM		Select DMM
STAT:OPER?	16	Bit 4 (Measuring)
FETC?	<DATA IN IEEE 488.2 BLOCK FORMAT>	

Appendices

Appendix A: Specifications

VX4101 General Characteristics

Table A–1: VXI Instrument Characteristics

Characteristics	Description
VXI General Characteristics	The instrument provides a VXI interface that complies with Revision 1.4. The VXI interface is defined by the VXI Consortium, Inc.
Interface Type	Message Based (1.4)
Other Protocols	Word Serial (WSP), FDC 2.0
Firmware Revision	1.0
Hardware Revision	671-3328-00/671–3532-00 671-3533-00/671-3537-00
Manufacturer ID	9FFD
Device ID	779A
Power Dissipated	42W
Cooling Required	For 15° C Rise, 0.08 mm H^2O@2.3 L/s
Operating Ambient Temperature	0–50° C
DMM Input Ratings	300 V CAT II 1A DC
Counter Input Ratings	300 V CAT II
TTL Outputs	VXI TTLTRG* Lines TTLTRG0* through TTLTRG7* under program control

Table A–2: Power Supply Voltage and Current

DC Volts	Current
+24 V	325 mA
+12 V	130 mA
+5 V	3.5 A
–2 V	60 mA
–5.2 V	1.2 A
–12 V	72 mA
–24 V	175 mA

Table A–3: Environmental/Reliability Characteristics

Characteristics	Description
Temperature	
Operating	Meets or exceeds MIL-T-28800E for Type III, 0 to 50° C external ambient, when operated in a mainframe providing Class 3 equipment
Non-operating	–40° C to + 71° C
Relative Humidity	
Operating	Up to 95% at up to 30° C, and up to 45%, at up to 50° C
Nonoperating	Up to 95%, at up to 50° C
Altitude (1) Operating	6,000 ft. altitude
Altitude (2)	Meets or exceeds MIL-T-28800E for Type III, (operating to 10,000 ft., nonoperating to 15,000 ft.)

Table A–4: VX4101-Specific Characteristics

Characteristics	Description
VXI Compliance	VXI message-based, fully compliant with revision 1.4 of the VXI specification
VXI Device Classification	Message based device
VME Interrupter Level	Dynamically configured
VXI Logical Address	Switch selectable to a value between 0 and 254
Contents of device/manufacturer dependent VXI registers.	ID Register: 9FFD hexadecimal. Device Type: 779A hexadecimal.
VXI TTL Trigger Outputs	One or more of the VXI TTLTRG* signals may be driven. All TTLTRG* outputs may be disabled
VXI TTL Trigger Inputs	One of the VXI TTLTRG* signals may be selected to be polled or to act as an interrupt source to the modules microprocessor
Module Size	“C” size, one slot wide
VXI Protocol	Word Serial (WSP) with Fast Data Channel for DMM data
Command Set	SCPI, I.EEE-488.2

Table A–5: VX4101-Specific Physical Characteristics

Characteristics	Description
Weight	
Dimensions	

Table A–5: VX4101-Specific Physical Characteristics (Cont.)

Characteristics	Description
Mounting location	Installs in an instrument module slot (1-12) of a “C” size VXIbus mainframe
Front panel signal connectors	BNC for Counter channels 1 and 2 SMA for Counter channel 3 SMB for Counter external arm DB-9 (female) for DMM

Over Voltage Indication

Benchtop DMMs typically display some maximum value and an overvoltage indication when an over voltage or over range condition is measured. These features are intended to indicate a possible unsafe condition.

For modules (plug in or VXI), a display and an easily noticeable over voltage indication light are not feasible and over voltage becomes the responsibility of the user. The VX4101 DMM returns a value of 9.99E + 37 or –9.9E + 37 to detect this situation.

CAUTION. *The user should provide some visible indication of a possible unsafe condition if a value greater that 9.9E + 37 or less than –9.9E + 37 is returned. Due to rounding errors inherent in floating point storage, it is recommended that the user not test for explicit equality with the over range values.*

Digital Multimeter (DMM) Specifications

Table A–6: Aperture Specifications

Characteristics	Description
Aperture (50 Hz)	1 ms to 2 sec in 1 ms to 10 ms steps (total of 560 apertures)
Aperture (60 Hz)	833 μs to 2 sec in 833 μs to 8.33 ms steps (total of 680 apertures)

Table A–7: Digits vs. Aperture

Digits	Aperture	Readings/Second
5.5	≥16.67 ms (20.0 ms)	≤60 (50)
4.5	<16.67 ms	>60 (50)

Table A–8: Memory Capacity

Characteristics	Description
On-Board Memory	4096 Measurements

Table A–9: DC Voltage

Range	Maximum Reading	Resolution
30 mV	± 30.0000 mV	100 nV
300 mV	± 300.000 mV	1 µV
3 V	± 3.00000 V	10 µV
30 V	± 30.0000 V	100 µV
300 V	± 300.000 V	1 mV

Table A–10: Accuracy Specifications for 2-Second Aperture

Range	24 Hour	90 Day	1 Year	Temp Coefficient
30 mV	0.006% + 8 µV (9 µV)	0.012% + 8 µV (9 µV)	0.018% + 8 µV (9 µV)	0.0015% + 1.5 µV
300 mV	0.005% + 14 µV (15 µV)	0.010% + 14 µV (15 µV)	0.015% + 14 (15 µV)	0.0015% + 2.5 µV
3 V	0.004% + 70 µV (80 µV)	0.008% + 70 µV (80 µV)	0.012% + 70 µV (80 µV)	0.0015% + 5 µV
30 V	0.0055% + 700 µV (800 µV)	0.011% + 700 µV (800 µV)	0.016% + 700 µV (800 µV)	0.0015% + 50 µV
300 V	0.007% + 8 mV (9 mV)	0.014% + 8 mV (9 mV)	0.020% + 8 mV (9 mV)	0.0015% + 500 µV

***NOTE**. Values shown above in parenthesis are for a 200 ms aperature.*

Table A–11: Accuracy Specification for ≤ 1 Millisecond Aperture

Range	24 Hour	90 Day	1 Year	Temp Coefficient
30 mV	0.007% + 15 µV (12 µV)	0.012% + 15 µV (12 µV)	0.018% + 15 µV (12 µV)	0.0015% + 1.5 µV
300 mV	0.005% + 30 µV (20 µV)	0.010% + 30 µV (20 µV)	0.015% + 30 µV (20 µV)	0.0015% + 2.5 µV
3 V	0.004% + 150 µV (100µV)	0.008% + 150 µV (100µV)	0.012% + 150 µV (100µV)	0.0015% + 5 µV
30 V	0.0055% + 1.5 mV (1.0 mV)	0.011% + 1.5 mV (1.0 mV)	0.016% + 1.5 mV (1.0 mV)	0.0015% + 50 µV
300 V	0.008% + 15 mV (10 mV)	0.014% + 15 mV (10 mV)	0.020% + 15 mV (10 mV)	0.0015% + 500 µV

***NOTE**. Values shown above in parenthesis are for a 16.67 ms or 20 ms aperture.*

***NOTE**. Percents listed above are percents of reading. The instrument setup is as follows:*

Autozero on following twenty minute warm-up period

Tcal is the calibration temperature (18° to 28° C). Specifications are for Tcal ±3°C. Multiply the total temperature coefficient by the difference between the actual operating temperature and Tcal ±3°C.

Add 0.001% of the range setting to the first reading following a range change

For autozero Off, do the following:

For daily and long term drift, add 24 μV per day for the 30 mV and 300 mV ranges, and 0.0027% of range for the other ranges.

For temperature drift, add .4 μV per °C for the 30 mV range, .001% of range per °C for the other ranges.

Table A–12: DC Input Resistance

Range	Resistance
30 mV, 300 mV, 3 V	10 MΩ or $>$10 GΩ, software-programmable
30 V, 300 V Ranges	10 MΩ ± 5%

Table A–13: DC Input Protection

Range	Protection
V + V–	350 V on all ranges
V + to Chassis	
V– to Chassis	

Table A–14: DC CMRR

Aperture	CMRR (DC)
All	110 dB

Table A–15: DC CMRR (0 to 400 Hz)

Aperture	CMRR 0-60 Hz	CMRR 0–400 Hz
2 seconds	113 dB	113 dB
200 ms	93 dB	93 dB
20 ms	73 dB	73 dB
16.7 ms	71 dB	71 dB
1 ms	61 dB (26 dB, 30, 300 V range)	45 dB (10 dB, 30, 300V range)

Table A–16: DC Normal Mode Rejection (50/60 Hz)

Aperture	NMR
2 seconds	90 dB
200 ms	80 dB
20 ms (50 Hz), 16.67 ms (60 Hz)	60 dB

Table A–17: DC ECMR (50/60/400 Hz)

Frequency	ECMR
50 Hz	122 dB
60 Hz	120 dB
400 Hz	102 dB

Table A–18: TRMS AC Voltage (DC Coupled and AC Coupled)

Range	Maximum Reading	Resolution
30 mV	30.0000 mV	100 nV
300 mV	300.000 mV	1 μV
3 V	3.00000	10 μV
30 V	30.0000 V	100 μV
300 V	300.000	1 mV

Table A–19: TRMS Accuracy Specifications –24-Hour

Range	Frequency	DC Coupled	AC Coupled	Temp Coefficient / °C
30 mV	20–45 Hz	0.75% + 180 µV	not specified	.03% + 15.0 µV
	45–100 Hz	0.40% + 180 µV	not specified	.03% + 15.0 µV
	100–200 Hz	0.30% + 180 µV	not specified	.03% + 15.0 µV
	0.2–10 kHz	0.30% + 180 µV	0.40% + 350 µV	.03% + 15.0 µV
	10–20 kHz	0.80% + 180 µV	0.90% + 350 µV	.03% + 15.0 µV
	20–50 kHz	2.00% + 350 µV	not specified	.03% + 15.0 µV
300 mV	20–45 Hz	0.75% + 300 µV	not specified	.03% + 15.0 µV
	45–100 Hz	0.40% + 300 µV	0.50% + 600 µV	.03% + 15.0 µV
	0.1–10 kHz	0.30% + 300 µV	0.40% + 600 µV	.03% + 15.0 µV
	10–20 kHz	0.80% + 300 µV	0.90% + 600 µV	.03% + 15.0 µV
	20–50 kHz	2.00% + 500 µV	2.00% + 750 µV	.03% + 15.0 µV
3 V	20–45 Hz	0.75% + 9 mV	not specified	.03% + 1.5 mV
	45–100 Hz	0.40% + 9 mV	0.50% + 12 mV	.03% + 1.5 mV
	0.1–10 kHz	0.30% + 9 mV	0.40% + 12 mV	.03% + 1.5 mV
	10–20 kHz	0.75% + 9 mV	0.90% + 12 mV	.03% + 1.5 mV
	20–50 kHz	2.50% + 9 mV	2.50% + 12 mV	.03% + 1.5 mV
30 V	20–45 Hz	0.75% + 30 mV	not specified	.03% + 2.5 mV
	45–100 Hz	0.40% + 30 mV	0.50% + 40 mV	.03% + 2.5 mV
	0.1–10 kHz	0.30% + 30 mV	0.40% + 40 mV	.03% + 2.5 mV
	10–20 kHz	1.20% + 30 mV	1.30% + 40 mV	.03% + 2.5 mV
	20–50 kHz	2.90% + 30 mV	2.90% + 40 mV	.03% + 2.5 mV
300 V	20–45 Hz	0.85% + 3 00 mV	not specified	.03% + 25 mV
	45–100 Hz	0.60% + 300 mV	0.70% + 400 mV	.03% + 25 mV
	0.1–10 kHz	0.50% + 300 mV	0.60% + 400 mV	.03% + 25 mV
	10–20 kHz	1.50% + 300 mV	1.75% + 400 mV	.03% + 25 mV
	20–50 kHz	2.50% + 300 mV	2.50% + 400 mV	.03% + 25 mV

***NOTE**. Specifications are for > 10% of range and are valid for all apertures (15% of range for 30 mV range).*

Table A–20: TRMS Accuracy–90 Day and 1 Year

Characteristic	Description
90-Day Accuracy	Add 0.08% of reading to 24-Hour Specifications above
1 Year Accuracy	Add 0.15% of reading to 24-Hour Specifications above

Table A–21: TRMS Crest Factor

Characteristic	Description
100% Full Scale	1.9:1
25% Full Scale	7:1

Table A–22: TRMS Input Impedance

Characteristics	Description
30 mV and 300 mV Ranges	>10 GΩ, <120 pF
3 V, 30 V, and 300 V Ranges	1.01 MΩ ± 5%, <120 pF

Table A–23: TRMS Input Protection – V+ to V–, V+ to Chassis, and V– to Chassis

Characteristics	Description
DC & AC/RMS	350 V on all ranges
AC Peak	450 V on all ranges

Table A–24: TRMS CMRR (0 to 400 Hz)

Characteristics	Range	Description
0 to 60 Hz	30 mV	>64 dB
0 to 60 Hz	300 mV	>61 dB
0 to 60 Hz	3, 30 V	>48 dB
0 to 60 Hz	300 V	>53 dB
0 to 400 Hz	30 mV	>48 dB
0 to 400 Hz	300 mV	>45 dB
0 to 400 Hz	3, 30 V	>31 dB
0 to 400 Hz	300 V	>36 dB

Table A–25: Resistance (2-Wire and 4-Wire)

Range	Maximum Reading	Resolution
30 Ω	30.0000 Ω	100 μΩ
300 Ω	300.000 Ω	1 mΩ

Table A–25: Resistance (2-Wire and 4-Wire) (Cont.)

Range	Maximum Reading	Resolution
3 kΩ	3.00000 kΩ	10 mΩ
30 kΩ	30.0000 kΩ	100 mΩ
300 kΩ	300.000 kΩ	1 Ω
3 MΩ	3.00000 MΩ	10 Ω
30 MΩ	30.0000 MΩ	100 Ω
300 MΩ	300.000 MΩ	1 kΩ (nom)

Table A–26: Resistance Accuracy Specifications for 2-Second Aperture

Range	Source Current	24 Hour	90 Day	1 Year	Temp Coefficient / °C
30 Ω	1 mA	0.017% + 8 mΩ (9 mΩ)	0.023% + 8 mΩ (9 mΩ)	0.037% + 8 mΩ (9 mΩ)	0.003% + 1.5 mΩ
300 Ω	1 mA	0.010% + 14 mΩ (15 mΩ)	0.015% + 14 mΩ (15 mΩ)	0.020% + 14 mΩ (15 mΩ)	0.003% + 2.5 mΩ
3 kΩ	1 mA	0.010% + 70 mΩ (80 mΩ)	0.015% + 70 mΩ (80 mΩ)	0.020% + 70 mΩ (80 mΩ)	0.003% + 5 mΩ
30 kΩ	100 µA	0.010% + 700 mΩ (800 mΩ)	0.015% + 700 mΩ (800 mΩ)	0.020% + 700 mΩ (800 mΩ)	0.003% + 50 mΩ
300 kΩ	10 µA	0.010% + 7 Ω (8 Ω)	0.015% + 7 Ω (8 Ω)	0.020% + 7 Ω (8 Ω)	0.003% + 500 mΩ
3 MΩ	1 µA	0.040% + 70 Ω (80 Ω)	0.060% + 70 Ω (80 Ω)	0.080% + 70 Ω (80 Ω)	0.008% + 5 Ω
30 MΩ	100 nA	0.300% + 700 Ω (800 Ω)	0.400% + 700 Ω (800 Ω)	0.500% + 700 Ω (800 Ω)	0.050% + 50 Ω
300 MΩ	100 nA	2.000% + 7 kΩ (8 kΩ)	2.200% + 7 kΩ (8k Ω)	2.500% + 7 kΩ (8 kΩ)	0.250% + 500 Ω

***NOTE**. Values shown in parenthesis above are for a 200 ms aperture.*

Table A–27: Resistance Accuracy Specifications for 1 Millisecond Aperture

Range	Source Current	24 Hour	90 Day	1 Year	Temp Coefficient / °C
30 Ω	1 mA	0.037% + 15 mΩ (12 mΩ)	0.037% + 15 m Ω (12 mΩ)	0.037% + 15 mΩ (12 mΩ)	0.003% + 1.5 mΩ
300 Ω	1 mA	0.015% + 30 mΩ (20 mΩ)	0.020% + 30 mΩ (20 mΩ)	0.025% + 30 mΩ (20 mΩ)	0.003% + 2.5 mΩ
3 kΩ	1 mA	0.015% + 150 mΩ (100 mΩ)	0.020% + 150 mΩ (100 mΩ)	0.025% + 150 mΩ (100 mΩ)	0.003% + 5.0 mΩ
30 kΩ	100 µA	0.015% + 1.5 Ω (1.0 Ω)	0.020% + 1.5 Ω (1.0 Ω)	0.025% + 1.5 Ω (1.0 Ω)	0.003% + 50 mΩ
300 kΩ	10 µA	0.015% + 15 Ω (10 Ω)	0.020% + 15 Ω (10 Ω)	0. 025% + 15 Ω (10 Ω)	0.003% + 500 mΩ
3 MΩ	1 µA	0.040% + 150 Ω (100 Ω)	0.060% + 150 Ω (100 Ω)	0.080% + 150 Ω (100 Ω)	0.008% + 5 Ω
30 MΩ	100 nA	0.300% + 1.5 kΩ (1.0 kΩ)	0.400% + 1.5 kΩ (1.0 kΩ)	0.500% + 1.5 kΩ (1.0 kΩ)	0.050% + 50 Ω
300 MΩ	100 nA	2.000% + 15 kΩ (10 kΩ)	2.200% + 15 kΩ (10 kΩ)	2.500% + 15 kΩ (10 kΩ)	0.250% + 500 Ω

***NOTE**. Values shown in parenthesis above are for a 16.67 ms or 20 ms aperture.*

***NOTE**. Percents listed above are percents of reading. Instrument set up is as follows:*

Autozero On, after 20 minute warm–up.

Tcal is the calibration temperature (18 to 28° C). Specifications are for Tcal ±3° C. Multiply the total temperature coefficient by the difference between the actual operating temperature and Tcal ±3° C.

Add 0.002% of the range setting to the first reading following a range change.

For Autozero Off, do the following:

For daily and long term drift, add 500 μΩ/Day for the 30 Ω range, 1 mΩ/ Day for the 300 Ω range, and 0.0001% of range for the other ranges.

For temperature drift, increase the second part of the temperature coefficient by a factor of 4 (increase 1.5 mΩ to 6 mΩ, 2.5 mΩ to 10 mΩ, etc.)

Table A–28: Resistance Input Protection-All Ranges

Terminals	DC and AC/RMS	AC Peak
R+ to R–	250 V	250 V
R+ to chassis	350 V	450 V
R– to chassis	350 V	450 V

***NOTE**. For Common Mode Rejection, see DCV Specifications.*

Table A–29: DC Current

Range	Maximum Reading	Resolution
150 mA	±150.000 mA	0.5 μA
1 A	±1.00000 A	5 μA

Table A–30: DC Current Sense Resistance

Characteristic	Description
Sense Resistance	0.206 Ω

Table A–31: DC Current Accuracy Specifications for 2-Second Aperture

Range	Burden	24 Hour	90 Day	1 Year	Temp Coefficient / °C
150 mA	±0.1 V	0.10% + 40 μA (50 μA)	0.12% + 40 μA (50 μA)	0.15% + 40 μA (50 μA)	0.05% + 8 μA
1A	±0.4 V	0.15% + 70 μA (800 μA)	0.15% + 70 μA (800 μA)	0.18% + 70 μA (800 μA)	0.05% + 50 μA

***NOTE**. Values shown in parenthesis above are for 16.67 or 20 ms aperture.*

Table A–32: DC Current Accuracy Specifications for 1 Millisecond Aperture

Range	Burden	24 Hour	90 Day	1 Year	Temp Coefficient / °C
150 mA	±0.1 V	0.10% + 60 μA	0.12% + 70 μA	0.15% + 70 μA	0.05% + 8 μA
1 A	±0.3 V	0.15% + 100 μA	0.15% + 100 μA	0.18% + 100 μA	0.05% + 50 μA

***NOTE**. Percents listed above are percent of reading. Instrument set up is as follows:*

Autozero On, after 20 minute warm–up.

Tcal is the calibration temperature (18 to 28° C). Specifications are for Tcal ±3° C. Multiply the total temperature coefficient by the difference between the actual operating temperature and Tcal ±3° C.

Add 0.02% of the range setting to the first reading following a range change.

Burden voltage is the maximum voltage drop caused by the current measurement for the maximum reading in the range.

For Autozero Off, do the following:

For daily and long term drift, add 110 μA/Day for the 150 mA range and 120 μA/Day for the 1A range.

For temperature drift, add 2 μA per °C for 150 mA range, and 15 μA per °C for the 1A range.

Table A–33: Input Protection

Characteristic	Description
Input Protection	2 A or ±0.6 V

***NOTE**. For Normal mode rejection (50/60 Hz), see DCV Specifications*

Table A–34: AC CMRR

Aperture	CMRR 0-60 Hz	CMRR 0–400 Hz
2 seconds	115 dB	115 dB
200 ms	95 dB	95 dB
20 ms	75 dB	75 dB
16.7 ms	73 dB	73 dB
1 ms	61 dB	45 dB

Table A–35: AC Normal Mode Rejection

Aperture	NMR
2 seconds	90 dB
200 ms	80 dB
20 ms (50 Hz), 16.67 ms (60 Hz)	60 dB

Table A–36: AC ECMR (All Apertures)

Frequency	AC ECMR
50 Hz	122 dB
60 Hz	120 dB
400 Hz	102 dB

Table A–37: TRMS AC Voltage (DC Coupled and AC Coupled)

Range	Maximum Reading	Resolution
30 mV	30.0000 mV	100 nV
300 mV	300.000 mV	1 µV
3 V	3.00000	10 µV
30 V	30.0000 V	100 µV
300 V	300.000	1 mV

Table A–38: Accuracy Specifications — 24-Hour Range

Range	Frequency	DC Coupled	AC Coupled	Temp Coefficient
30 mV	20–45 Hz	0.75% + 180 µV	not specified	.01% + 10.0 µV
	45–100 Hz	0.40% + 180 µV	not specified	.01% + 10.0 µV
	100–200 Hz	0.30% + 180 µV	not specified	01% + 10.0 µV
	0.2–10 kHz	0.30% + 180 µV	0.40% + 350 µV	01% + 10.0 µV
	10–20 kHz	0.80% + 180 µV	0.90% + 350 µV	01% + 10.0 µV
	20–50 kHz	2.00% + 350 µV	not specified	01% + 10.0 µV
300 mV	20–45 Hz	0.75% + 300 µV	not specified	.01% + 6.0 µV
	45–100 Hz	0.40% + 300 µV	0.50% + 600 µV	.01% + 6.0 µV
	0.1–10 kHz	0.30% + 300 µV	0.40% + 600 µV	.01% + 6.0 µV
	10–20 kHz	0.80% + 300 µV	0.90% + 600 µV	.01% + 6.0 µV
	20–50 kHz	2.00% + 500 µV	2.00% + 750 µV	.01% + 6.0 µV
3 V	20–45 Hz	0.75% + 9 mV	not specified	.01% + 1.5 mV
	45–100 Hz	0.40% + 9 mV	0.50% + 12 mV	.01% + 1.5 mV
	0.1–10 kHz	0.30% + 9 mV	0.40% + 12 mV	.01% + 1.5 mV
	10–20 kHz	0.75% + 9 mV	0.90% + 12 mV	.01% + 1.5 mV
	20–50 kHz	2.50% + 9 mV	2.50% + 12 mV	01% + 1.5 mV
30 V	20–45 Hz	0.75% + 30 mV	not specified	.01% + 1.5 mV
	45–100 Hz	0.40% + 30 mV	0.50% + 40 mV	.01% + 1.5 mV
	0.1–10 kHz	0.30% + 30 mV	0.40% + 40 mV	01% + 1.5 mV
	10–20 kHz	1.20% + 30 mV	1.30% + 40 mV	.01% + 1.5 mV
	20–50 kHz	2.90% + 30 mV	2.90% + 40 mV	01% + 1.5 mV
300 V	20–45 Hz	0.85% + 3 00 mV	not specified	.01% + 15 mV
	45–100 Hz	0.60% + 300 mV	0.70% + 400 mV	.01% + 15 mV
	0.1–10 kHz	0.50% + 300 mV	0.60% + 400 mV	.01% + 15 mV
	10–20 kHz	1.50% + 300 mV	1.75% + 400 mV	.01% + 15 mV
	20–50 kHz	2.50% + 300 mV	2.50% + 400 mV	.01% + 15 mV

***NOTE**. Specifications are for $>$10% of range and are valid for all apertures (15% of range for 30 mV range).*

Table A–39: Accuracy

Characteristic	Description
90-Day Accuracy	Add 0.08% of reading to 24-Hour Specifications above
1 Year Accuracy	Add 0.15% of reading to 24-Hour Specifications above

Table A–40: Crest Factor

Characteristic	Description
100% Full Scale	1.9:1
25% Full Scale	7:1

Table A–41: Input Impedance

Characteristics	Description
30 mV and 300 mV Ranges	>10 GΩ, <120 pF
3 V, 30 V, and 300 V Ranges	1.01 MΩ + 5%, <120 pF

Table A–42: Input Protection – V+ to V–, V+ to Chassis, and V– to Chassis

Characteristics	Description
DC & AC/RMS	350 V on all ranges
AC Peak	450 V on all ranges

Table A–43: Common Mode Rejection (0 to 60 Hz and 0 to 400 Hz)

Characteristics	Range	Description
0 to 60 Hz	30, 300 mV	>68 dB
0 to 60 Hz	3, 30 V	>48 dB
0 to 60 Hz	300 V	>53 dB
0 to 400 Hz	30, 300 mV	>56 dB
0 to 400 Hz	3, 30 V	>36 dB
0 to 400 Hz	300 V	>41 dB

Table A–44: Resistance (2-Wire and 4-Wire)

Range	Maximum Reading	Resolution
30 Ω	30.0000 Ω	100 μΩ
300 Ω	300.000 Ω	1 mΩ
3 kΩ	3.00000 kΩ	10 mΩ
30 kΩ	30.0000 kΩ	100 mΩ

Table A–44: Resistance (2-Wire and 4-Wire) (Cont.)

Range	Maximum Reading	Resolution
300 kΩ	300.000 kΩ	1 Ω
3 MΩ	3.00000 MΩ	10 Ω
30 MΩ	30.0000 MΩ	100 Ω
300 MΩ	300.000 MΩ	1 kΩ (nom)

Table A–45: Accuracy Specifications for 2-Second Aperture

Range	Source Current	24 Hour	90 Day	1 Year	Temp Coefficient
30 Ω	1 mA	0.017% + 8 mΩ (9 mΩ)	0.023% + 8 mΩ (9 mΩ)	0.037% + 8 mΩ (9 mΩ)	0.002% + 6.0 mΩ
300 Ω	1 mA	0.010% + 14 mΩ (15 mΩ)	0.015% + 14 mΩ (15 mΩ)	0.020% + 14 mΩ (15 mΩ)	0.002% + 10 mΩ
3 kΩ	1 mA	0.010% + 70 mΩ (80 mΩ)	0.015% + 70 mΩ (80 mΩ)	0.020% + 70 mΩ (80 mΩ)	0.002% + 20 mΩ
30 kΩ	100 µA	0.010% + 700 mΩ (800 mΩ)	0.015% + 700 mΩ (800 mΩ)	0.020% + 700 mΩ (800 mΩ)	0.003% + 200 mΩ
300 kΩ	10 µA	0.010% + 7 Ω (8 Ω)	0.015% + 7 Ω (8 Ω)	0.020% + 7 Ω (8 Ω)	0.003% + 2 Ω
3 MΩ	1 µA	0.040% + 70 Ω (80 Ω)	0.060% + 70 Ω (80 Ω)	0.080% + 70 Ω (80 Ω)	0.008% + 20 Ω
30 MΩ	100 nA	0.300% + 700 Ω (800 Ω)	0.400% + 700 Ω (800 Ω)	0.500% + 700 Ω (800 Ω)	0.050% + 200 Ω
300 MΩ	100 nA	2.000% + 700 Ω (800 Ω)	2.200% + 700 Ω (800 Ω)	2.500% + 700 Ω (800 Ω)	0.250% + 200 Ω

***NOTE**. Values shown in parenthesis above are for a 200 ms aperture.*

Table A–46: Accuracy Specifications for 1 Millisecond Aperture

Range	Source Current	24 Hour	90 Day	1 Year	Temp Coefficient
30 Ω	1 mA	0.035% + 15 mΩ (12 mΩ)	0.035% + 15 m Ω (12 mΩ)	0.035% + 15 mΩ (12 mΩ)	0.002% + 1.5 mΩ
300 Ω	1 mA	0.015% + 30 mΩ (20 mΩ)	0.020% + 30 mΩ (20 mΩ)	0.025% + 30 mΩ (20 mΩ)	0.002% + 2.5 mΩ
3 kΩ	1 mA	0.015% + 150 mΩ (100 mΩ)	0.020% + 150 mΩ (100 mΩ)	0.025% + 150 mΩ (100 mΩ)	0.002% + 5.0 mΩ

Table A–46: Accuracy Specifications for 1 Millisecond Aperture (Cont.)

Range	Source Current	24 Hour	90 Day	1 Year	Temp Coefficient
30 kΩ	100 μA	0.015% + 1.5 Ω (1.0 Ω)	0.020% + 1.5 Ω (1.0 Ω)	0.025% + 1.5 Ω (1.0 Ω)	0.002% + 50 mΩ
300 kΩ	10 μA	0.015% + 15 Ω (10 Ω)	0.020% + 15 Ω (10 Ω)	0. 025% + 15 Ω (10 Ω)	0.002% + 500 mΩ
3 MΩ	1 μA	0.040% + 150 Ω (100 Ω)	0.060% + 150 Ω (100 Ω)	0.080% + 150 Ω (100 Ω)	0.008% + 5 Ω
30 MΩ	100 nA	0.300% + 1.5 kΩ (1.0 kΩ)	0.400% + 1.5 kΩ (1.0 kΩ)	0.500% + 1.5 kΩ (1.0 kΩ)	0.050% + 50 Ω
300 MΩ	100 nA	2.000% + 1.5 kΩ (1.0 kΩ)	2.200% + 1.5 kΩ (1.0 kΩ)	2.500% + 1.5 kΩ (1.0 kΩ)	0.250% + 50 Ω

***NOTE**. Values shown in parenthesis above are for a 16.67 ms or 20 ms aperture.*

***NOTE**. Percents listed above are percent of reading. Instrument set up: Autozero On, after 20 minute warm–up. Tcal is the calibration temperature (18 to 28° C). Specifications are for Tcal ±3° C. Add 0.002% of the range setting to the first reading following a range change. For Autozero Off, add 500 μΩ/Day for the 30 Ω range, 1 mΩ/ Day for the 300 Ω range, and 0.0001% of range for the other ranges. Multiply the total temperature coefficient by the difference between the actual operating temperature and Tcal ±3° C.*

Table A–47: Input Protection-All Ranges

Terminals	DC and AC/RMS	AC Peak
R+ to R–	250 V	250 V
R+ to chassis	350 V	450 V
R– to chassis	350 V	450 V

***NOTE**. For Common Mode Rejection, see DCV Specifications.*

Table A–48: DC Current

Range	Maximum Reading	Resolution
150 mA	±150.000 mA	0.5 μA
1 A	±1.00000 A	5 μA

Table A–49: Sense Resistance

Characteristic	Description
Sense Resistance	0.206 Ω

Table A–50: Accuracy Specifications for 2-Second Aperture

Range	Burden	24 Hour	90 Day	1 Year	Temp Coefficient
150 mA	±0.1 V	0.10% + 40 µA (50 µA)	0.12% + 40 µA (50 µA)	0.15% + 40 µA (50 µA)	0.05% + 8 µA
1A	±0.4 V	0.15% + 70 µA (800 µA)	0.15% + 70 µA (800 µA)	0.18% + 70 µA (800 µA)	0.05% + 50 µA

***NOTE**. Values shown in parenthesis above are for 16.67 or 20 ms aperture.*

Table A–51: Accuracy Specifications for 1 Millisecond Aperture

Range	Burden	24 Hour	90 Day	1 Year	Temp Coefficient
150 mA	±0.1 V	0.10% + 60 µA	0.12% + 70 µA	0.15% + 70 µA	0.05% + 8 µA
1 A	±0.3 V	0.15% + 100 µA	0.15% + 100 µA	0.18% + 100 µA	0.05% + 50 µA

***NOTE**. Percents listed above are percent of reading. Add 50 mΩ for 2-wire measurement. Instrument set up: Autozero On, after 20 minute warm–up. Tcal is the calibration temperature (18 to 28° C). Specifications are for Tcal ±3° C. Add 0.02% of the range setting to the first reading following a range change. Burden voltage is the maximum voltage drop caused by the current measurement for the maximum reading in the range. For Autozero Off, add 110 µA/Day for the 150 mA range and 120 µA/Day for the 1A range. Multiply the total temperature coefficient by the difference between the actual operating temperature and Tcal ±3° C.*

Table A–52: Input Protection

Characteristic	Description
Input Protection	2 A or ±0.6 V

***NOTE**. For Noise reject-Normal mode, see DCV Specifications*

Universal Counter Specifications

This section contains specification tables for the counter. The specifications contain a variety of equations and mathematical operators. A table at the end of the counter specifications contains definitions for the operators.

Table A–53: General Specifications

Characteristics	Description
Number of Channels	One input channel for frequency, period, pulse width, time interval, time interval after delay, rise time, fall time, an ratio. A second input channel can be used independently for frequency, period, pulse width, rise time, and fall time; or in conjunction with the first input for time interval, time interval after delay, and phase angle. The second input can also be used for very-high-speed gating. A third input channel (OPTIONAL) uses a divide-by-4 prescaler for extremely high speed signals. It measures frequency or period, or measures ratio in conjunction with either of the first two inputs.
Number of Digits	Up to 17 digits will be returned. Engineering notation is used for positioning the decimal point except for totalizing event measurements.
Gating	Front panel Arm signal, VXI command gating, VXI backplane triggers, and fixed interval timing. Gating can be: level, pulse on from one source and off from a second source, or on from the edge of one pulse then off from the corresponding edge of a second pulse.

Table A–54: Channel 1 and 2 Frequency

Characteristics	Description
Range	0.1 µHz to 250 MHz (500MHz Optional)
Least Significant Digit	$10exp\{Int(log((P_{VCO}/4) * F^2/(N \pm F(P_{VCO}/4)))) + 1\}$.

Table A–54: Channel 1 and 2 Frequency (Cont.)

Characteristics	Description
Resolution	UnGated Resolution: $\pm$ LSD $\pm$ 1.414(TJE) F^2/N. Gated Resolution: $\pm$ ((P_{VCO}) F^2/n) $\pm$ (1.414(TJE) * F^2/n).
Accuracy	$\pm$ Resolution $\pm$ F(TBE).

Table A–55: Channel 1 and 2 Period

Characteristics	Description
Range	4 ns to 9 million seconds (2ns min. with 500 MHz Option).
Least Significant Digit	250 ps or 250 ps/N rounded downward to the nearest decade.
Resolution	UnGated Resolution: $\pm$ LSD $\pm$ 1.414(TJE)/N $\pm$ 2ps Gated Resolution: $\pm$ (P_{VCO}/n) $\pm$ (1.414(TJE)/n). $\pm$ 2ps
Accuracy	$\pm$ Resolution $\pm$ TBE_{VCO}. *NOTE:* Detection of signals crossing prescribed thresholds is subject to timing uncertainty (trigger jitter error) due to system noise and threshold inaccuracies. This error affects all measurements but is more pronounced for single-shot measurements and for signals with slow rise and fall time. The timing trigger jitter error (TJE) equals: $\frac{[(RMSnoiseofinputsignal)^2 + (RMSnoiseofinputChannel)^2]^{1/2}}{InputSlewRateattheTriggerPoint}$

Table A–56: Channel 3 Frequency

Characteristics	Description
Range	200 MHz to 3000 MHz
Least Significant Digit	10exp{Int(log((P_{VCO}/4) * 4 * F^2/(N $\pm$ 4 * F(P_{VCO}/4)))) + 1}.
Resolution	Ungated Resolution: $\pm$ LSD $\pm$ 1.414(TJE) F^2/N. Gated Resolution: $\pm$ ((P_{VCO}) * F^2/n) $\pm$ (1.414(TJE) * F^2/n).
Accuracy	$\pm$ Resolution $\pm$ F(TBE)

Table A–57: Channel 3 Period

Characteristics	Description
Range	333 ps to 5 ns
Least Significant Digit	250 ps or 250 ps/N rounded downward to the nearest decade.
Resolution	Ungated Resolution: $\pm$ LSD $\pm$ 1.414(TJE)/N $\pm$ 2 ps Gated Resolution: $\pm$ (P_{VCO}/n) $\pm$ (1.414(TJE)/n) $\pm$ 2 ps
Accuracy	$\pm$ Resolution $\pm$ TBE_{VCO}

Table A–58: Time Interval

Characteristics	Description
Frequency Range	Up to 250 MHz (500 MHz with option 1C) Up to 150 MHz for Time Interval with Delay
Range	–1 ns to 9 million seconds
Least Significant Digit	250 ps for N = 1, single-shot (1 ns/√N) for N > 1 rounded downward to the nearest decade.
Resolution	$\pm$ LSD $\pm$ 1.414 ((TJE_1 + TJE_2)/√N) $\pm$ 2ps
Accuracy	$\pm$TBE (Time Interval) $\pm$ Resolution + ($TLE_1/Slew_S$ – $TLE_2/Slew_E$)$\pm$200 ps

Table A–59: Frequency Ratio

Characteristics	Description
Ch 1/Ch 2 and Ch 2/Ch 1	10^{-14} to 10^{14}
Ch 1/Ch 3 and Ch 2/Ch 3 with Option 2C	10^{-15} to 1
Ch 3/Ch 1 and Ch 3/Ch 2 with Option 2C	1 to 10^{15}

Table A–60: Channels 1 and 2 Totalizer

Characteristics	Description
Range	1 to 2^{52} (4.5 $\times$ 10^{15})
Least Significant Digit	One count.

Table A–61: Channels 1 and 2 Rise/Fall Time

Characteristics	Description
Frequency Range	Up to 250 MHz (0.5 V_{p-p} minimum signal for specified accuracy).
Range	250 ps single-shot 1 ns to 9 million seconds
Least Significant Digit	250ps for N = 1, single-shot (1 ns/√N) for N > 1 rounded downward to the nearest decade.
Resolution	LSD ± 1.414 ((TJE_1 + TJE_2)/√N) ± 2ps RMS
Accuracy	±TBE (Time Interval) ± Resolution + (TLE_2/$Slew_S$ – TLE_1/$Slew_E$)±200 ps

Table A–62: Channels 1 and 2 Positive/Negative Pulse Width

Characteristics	Description
Frequency Range	Up to 250 MHz (500 MHz with Option 1C)
Range	1 ns to 9 million seconds
Least Significant Digit	250 ps (single-shot)
Resolution	± LSD ± 1.414 ((TJE_1 + TJE_2)/√N) ± 2ps (for 1 ≤ N ≤9 million)
Accuracy	± Resolution ± Width (TBE) ± TLE_1 ± TLE_2 ± 1 ns.

Table A–63: Ratio 1/2,2/1,3/1,3/2,1/3,2/3

Characteristics	Description
Range	10^{-12} to 10^{12}
Least Significant Digit	(F_n/ F_d) * Gate time (divide F by 4 for channel 3 prescaler)
Resolution	Same as LSD
Accuracy	Same as LSD

Table A–64: Voltages

Characteristics	Description
Offset Range	±100 V DC
Maximum Input Signal	± 100 V (DC + peak AC) + offset

Table A–65: Channels 1 and 2 Input Characteristics

Characteristics	Description
Hysteresis 1,2 (@ 1MHz)	Programmable in 10 mV_{p-p} steps from: Minimum (10 mV_{p-p} on-to-off) to Maximum (60 mV_{p-p} on-to-off) × Attenuation/Gain
Coupling 1,2	AC,DC
Trigger Slope 1,2	Independent selection of slope polarity
Attenuator 1,2	×1, ×10, ×100 Nominal
Impedance 1,2	1 MΩ shunted by 15 pF or 50 Ω
Damage Level 1,2	Input voltage at 1 MΩ ×1: 300 V (DC or peak AC) derate to 5 V (DC + peak AC) at 20 dB/decade above 10 kHz ×10: 300 V (DC or peak AC) derate to 5 V (DC + peak AC) at 20 dB/decade above 10 MHz ×100: 300 V (DC or peak AC) derate to 5 V (DC + peak AC) at 20 dB/decade above 10 MHz Input Voltage at 50 Ω = 5 V (0.5 Ω) maximum (Overload protection auto-switch to 1M Ω over 840 mW (6.5 VDC) in less than 5 seconds.)
Common Input Limitations	None
Sensitivity 1,2	–20 dBm to 200 MHz, then 6 db per octave to –12 dBm at 500 MHz 3dB Low Frequency limit with AC coupling: 1 e6 Ω input impedance = 10 Hz 50 Ω input impedance = 200 kHz
Input Channel Noise 1,2	± 2mV_{rms} typ < 500 μV_{rms}

Table A–66: Channel 3 Input Characteristics

Characteristic	Description
Coupling	AC
Trigger Slope 3	Positive slope (threshold at mean)
Impedance 3	50 Ω
Damage Level 3	5 VPTP (AC)

Table A–66: Channel 3 Input Characteristics (Cont.)

Characteristic	Description
Common Input Limitations	None
Sensitivity 3	200 MHz to 1000 MHz–(–30 dBm) 1000 MHz to 3000 MHz–(–25 dBm)

Table A–67: Arm Characteristics

Characteristic	Description
Arm Sources	VXI TTL Trigger lines, front panel arm, Channel 2, DMM, SurePath, Software, and an internal timer
Arm Start/Stop	Except for channel 2, the arm can be a level or a pulse to start and a second pulse to stop. The start and stop pulses may be different sources. Channel 2 is high speed arm that is a level
Front Panel Arm Polarity	Positive or Negative
Threshold	Programmable
Measurement Time-out	Programmable time-out to abort measurements in case no input signal is present

Table A–68: Channel 1 and 2 Trigger Level

Characteristic	Description
Trigger Level Range	± 4 mV to ± 225 V
Trigger Level Accuracy	For DC or 1 kHz 100 MHz: Resolution: 0.2 mV * (Gain/Attenuation) Error = ± 7.5% of setting ± 25mV *Attenuation

Table A–69: Channels 1 and 2 Autotrigger

Characteristic	Description
Auto Trigger Range 1,2	Same as user selected for continuous waveforms > = 1 kHz

Table A–70: Front Panel Connectors

Characteristic	Description
Connector Types	BNC (Ch1, Ch2), SMA (Ch3), SMB (Arm)

Table A–71: TimeBase Characteristics

Characteristic	Description
Time Base Characteristics	The time base can be drawn from four different sources: the on-board 10 MHz source, the optional on-board high accuracy 10 MHz source, the VXI Slot 0 10 MHz source, and the Slot 0 External 10 MHz source. The user can select, by programmed command, the source and what degree of accuracy is required. The Slot 0 External 10 MHz source can be synchronized to other systems.
Default Time Base	On-board 25 ppm VXI Slot 0 required accuracy 100 ppm VXI Slot 0 External As required
External Time Base	User provided
Frequency Range	VXI cards are available with tightly controlled specifications, including a Rubidium Vapor Oscillator with a claimed accuracy of $< 5 \times 10^{-11}$ per month stability.
Interface	10 MHz connected into 50 Ω Slot 0 connector. The Slot 0 card must be programmed to source the VXI backplane 10 MHz from the front-panel input.

Table A–72: Gate Input Trigger

Characteristic	Description
Gate Inputs	There are four sources of gate inputs: VXI TTL backplane triggers, on-board processor generated triggers (either by VXI Command or timer algorithm), front panel Gate/Arm input, or Channel 2 input as gate. All gate inputs will start and stop counting synchronously with the measured input on its next transition (Trigger) for all measurements except totalizing. *NOTE:* Tektronix terminology uses the measured signals input (Channel 1 or Channel 2) as Triggers, processor and backplane on/off signals as Gates, and the front-panel control signal as the Arm signal. When the Channel 2 input signal controls the starting and stopping of the Channel 1 signal, the Channel 2 signal will be referred to as a Gate.
Front Panel Gate Input Interface	SMB connector
Polarity	Programmable
Gate Type	Programmable: level, edge, pulse-on/pulse-off
Edge Triggering Amplitude	±500 mV minimum, ±20 V Maximum
Pulse Width Requirements	40 ns minimum pulse width
Gate Start/End Uncertainty	$<$ 20 ns

Table A–73: VXIBus TTLTRG Gate Input

Characteristic	Description
TTLTRG Timing	Synchronous with VXI backplane 10 MHz Asynchronous
TTLTRG Type	Programmable: level, edge, pulse-on/pulse-off

Table A–73: VXIBus TTLTRG Gate Input (Cont.)

Characteristic	Description
Channel-B Counter Input Gate Polarity	Programmable
Gate Type	Programmable: level, edge
Edge Triggering Amplitude/Pulse Width Requirements	±5 mV minimum, ±200 V Maximum 2.86 ns minimum

Table A–74: Software Gate/Trigger

Characteristic	Description
Software Gate Type	Level
Software Gate Duration Time Range	Command based: User defined Timer based: 10 ns to 9 million seconds, 1 ns resolution

Table A–75: Gate Duration Modes

Characteristic	Description
Gate Duration Time Range	2.86 ns minimum (Channel 2 Gate) 1000 seconds maximum
Gate Duration Indefinite	Software command or hardware based
Gate Output	At the conclusion of the measurement, defined time, or defined number or events.

Table A–76: Measurement Throughput

Characteristic	Description
Throughput Rate	75 measurements/second for array measurements.

Table A–77: Counter Specifications Terms

Term	Description	
Frequency Terms	F	Frequency of the signal being measured (F = 1/P).
	F_L	Frequency of the leading (first) signal being measured.
	F_T	Frequency of the trailing (second) signal being measured.
	F_1	Frequency of the Channel 1 signal being measured.
	F_2	Frequency of the Channel 2 signal being measured.
	F_n	Frequency of the Numerator (Ratio Measurement).
	F_d	Frequency of the Denominator (Ratio Measurement).
	F_{VCO}	Frequency of the Universal Counter Voltage Controlled Oscillator time base. (F_{VCO} has been chosen so that it is not evenly divisible by 10 kHz, thereby reducing the likelihood of aliasing.) It is calculated as follows: ($F_{VCO} = 10e6 \times 304.0/3.0$)
Numeric Terms	N	Number of events in a measurement interval. For Auto-Averaging, N is the greater of one or the number of events in a 100 ms interval.
	N_1	Number of events in a leading (first) measurement interval.
	N_2	Number of events in a trailing (second) measurement interval.
	N_1	Number of events in a Channel 1 measurement interval.
	N_2	Number of events in a Channel 2 measurement interval.
	n	For gated measurements, n is the number of events within the gated interval.

Table A–77: Counter Specifications Terms (Cont.)

Term	Description
Timing Terms	DJ Delay Jitter P The period of the signal being measured (P = 1/F). P_{VCO} (1/ F_{VCO}). Approximately 0.987 ns. $Slew_E$ Slew at the ending edge of a measurement. For sine waves, the slew rate at the 50% level is as follows: $2 \times \pi \times F \times VPP/2$ For sine waves at 10% (or 90%) the slew rate is as follows: $\cos(\arcsin(-1.0 + (10\% \text{ of } 2.0))) \times 2 \times \pi \times F \times VPP/2$ $Slew_S$ Slew at the starting edge of a measurement. TBE Time Base Error of the 10 MHz root time base provided by the Slot 0 board, a frequency plugged into the Slot 0 board, or the optional on-board oscillator. TBE_{VCO} Time Base Error of the root time base applied to the frequency of the Universal Counter Voltage Controlled Oscillator time base. TJE_1 Trigger Jitter Error of the Channel 1 event. TJE_2 Trigger Jitter Error of the Channel 2 event. *NOTE:* Detection of signals crossing prescribed thresholds is subject to timing uncertainty (trigger jitter error) due to system noise and threshold inaccuracies. This error affects all measurements but is more pronounced for single-shot measurements and for signals with slow rise and fall time. The timing trigger jitter error (TJE) equals: $\frac{[(RMSnoiseofinputsignal)^2 + (RMSnoiseofinputChannel)^2]^{1/2}}{InputSlewRateattheTriggerPoint}$ TLE_1 Trigger Level Error of the leading (first) event. TLE_2 Trigger Level Error of the trailing (second) event. TJE Trigger Jitter Error which is the input signal RMS noise voltage. TJE_1 Trigger Jitter Error of the leading (first) event. TJE_2 Trigger Jitter Error of the trailing (second) event.
Mathematical Terms	exp{} Exponentiates to the power given by the expression within the bracket. Int() Converts the expression within the parenthesis to the nearest integer of lesser or equal value. For example: Int(1.7) = 1 and Int(-1.7) = -2 (Use floor (x) in the C programming language.) log() Computes the base 10 (common) logarithm of the expression within the parenthesis.

SurePath™ Specifications

The VX4101 controls the following currently available SurePath™ switching modules as slaves:

- VX4320 1.3 GHz RF Multiplexer
- VX4330 120 Channel Scanner/Multiplexer
- VX4350 64 Channel SPDT
- VX4380 256 Cross Point Relay Matrix Module

Maximum number of slave modules: 11

***NOTE**. As new SurePath switching modules become available the VX4101 will be updated to control these products*

Certfications and Compliances

Table A–78: Certifications and compliances

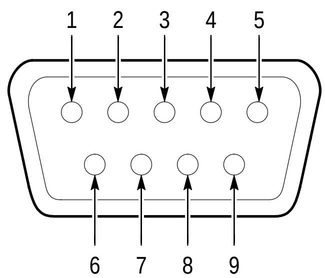
EC Declaration of Conformity – EMC	Meets intent of Directive 89/336/EEC for Electromagnetic Compatibility. Compliance was demonstrated to the following specifications as listed in the Official Journal of the European Communities: EN 55011 — Class A Radiated and Conducted Emissions EN 50081-1 Emissions: EN 60555-2 — AC Power Line Harmonic Emissions EN 50082-1 Immunity: IEC 801-2 — Electrostatic Discharge Immunity IEC 801-3 — RF Electromagnetic Field Immunity IEC 801-4 — Electrical Fast Transient/Burst Immunity IEC 801-5 — Power Line Surge Immunity To ensure compliance with EMC requirements this module must be installed in a mainframe which has backplane shields installed which comply with Rule B.7.45 of the VXIbus Specification. Only high quality shielded cables having a reliable, continuous outer shield (braid & foil) which has low impedance connections to shielded connector housings at both ends should be connected to this product.
EC Declaration of Conformity – Low Voltage	Compliance was demonstrated to the following specification as listed in the Official Journal of the European Communities: Low Voltage Directive 73/23/EEC EN 61010-1:1993 — Safety requirements for electrical equipment for measurement, control, and laboratory use

Table A–78: Certifications and compliances (Cont.)

Approvals	UL3111-1 – Standard for electrical measuring and test equipment CAN/CSA C22.2 No. 1010.1 – Safety requirements for electrical equipment for measurement, control and laboratory use
Safety Certification of Plug-in or VXI Modules	For modules (plug-in or VXI) that are safety certified by Underwriters Laboratories, UL Listing applies only when the module is installed in a UL Listed product. For modules (plug-in or VXI) that have cUL or CSA approval, the approval applies only when the module is installed in a cUL or CSA approved product.
Installation Category Descriptions	Terminals on this product may have different installation category designations. The installation categories are: CAT III Distribution-level mains (usually permanently connected). Equipment at this level is typically in a fixed industrial location CAT II Local-level mains (wall sockets). Equipment at this level includes appliances, portable tools, and similar products. Equipment is usually cord-connected CAT I Secondary (signal level) or battery operated circuits of electronic equipment

Appendix B: Input/Output Connections

Table B–1: Digital Multimeter (DMM) Input/Output Connections



Pin Number	Signal
6	Voltage/2-wire Ω +, 4-wire Ω +
1	Voltage/2-wire Ω –, 4-wire Ω –
9	4-wire Ω +
5	4-wire Ω –
7	Current +
3	Current –
2	Not connected
4	Not connected
8	Not connected

Appendix C: Examples

This appendix contains various examples for using the VX4101. They include the following:

- Using the DMM Subsystem
- Using the VXI*plug&play* driver function calls

Using the DMM Subsystem

The six functions that the DMM measures are:

- DC Volts
- AC Volts
- AC + DC Volts
- DC Current
- 2-Wire Resistance
- 4-Wire Resistance

Each of these functions can be independently configured to optimize their usage. The configurable parameters for each function are:

Range (The input range)

Aperture (The measurement acquisition time)

Count (The number of measurements to take)

AutoRange (Selective enabling/disabling of the autorange function)

Command Parameters

Before illustrating the operation of the measurement subsystem, we must define the parameters used in these commands. For a DC voltage measurement, the command syntax is:

```
MEASure
    [:SCALar][:VOLTage][:DC]? [<Expected Value>[,<Resolution>]]
    :ARRay[:VOLTage][:DC]? <Array Size>[,<Expected
    Value>[,<Resolution>]]
```

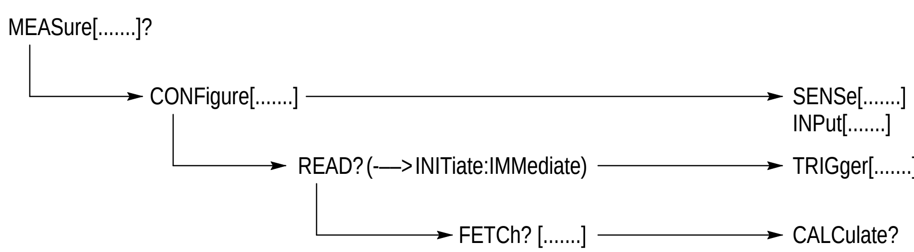
SCALar and ARRay Fields. These fields define the number of measurements to take. Specifying [:SCALar] performs a single measurement acquisition. Specifying :ARRay performs multiple measurements of count <Array Size>.

[<Expected Value>] Field. This field is an estimate of the input signal amplitude. If a numeric value is specified, the card will use the range nearest to the value specified to make the measurement (rounded up). [<Expected Value>] can also be specified as MAXimum, MINimum, or DEFault to use the card's highest range, lowest range, or to default to autoranging respectively. If no value is specified, DEFault (autorange) is assumed.

Resolution Field. This field defines the resolution of a measurement by implicitly defining the aperture time of the measurement. The aperture is the time the DMM takes to acquire a measurement. The greater the aperture, the more accurate the measurement. If MAXimum is specified as the resolution field, the card will use a 2 second aperture for each measurement. Specifying MINimum uses a 0.8333 ms (1 ms) aperture for 60 (50) Hz line frequency rejection. Specifying DEFault (or none) sets the aperture to 200 ms.

Making Measurements

The measurement subsystem is a hierarchical series of commands which give varying degrees of control over configuring for and acquiring measurements. Executing any command within the hierarchy causes that command, and all commands below it to be executed. The top of the hierarchy provides the simplest means of acquiring a measurement. Traversing down the hierarchy provides increasing degrees of control over the acquisition via the related commands listed to their right. The hierarchy is:



MEASure? The MEASure? commands are the top of the hierarchy. They perform all steps necessary to configure, acquire, and return measurement data in a single command. As an example, executing a MEASure? command performs the following equivalent set of commands:

```
MEASure?-->CONFigure-->READ?(--> INITiate:IMMediate)--> FETCh?
```

READ? Similarly, executing a READ? query executes the following equivalent set of commands:

```
READ?(-->INITiate:IMMediate)-->FETCh?
```

DMM Operation. Using the above definitions, the following top-level command would perform five DC voltage measurements, on the the 3 V range (nearest range above 2.5 V), using a 2 second aperture.

```
MEAS:ARR:VOLT:DC? 5,2.5,MAX
```

NOTE*. The resolution field implicitly defines the aperture time. A detailed description of the correlation between the aperture and the specified resolution is given in the SENSe command description.*

For example, the above command might return (in SCPI block format):

```
#265+2.50123E+00,+2.49801E+00,+2.49799,+2.50000E+00,+2.49916E
+00<LF>
```

At this point, the DC Voltage measurement configuration is set for 5 measurements on the 3 V range, with an aperture of 2 seconds. To perform another set of 5 measurements, the READ? command could be used, since the current setup has been defined via the MEASure command. If the next measurements needed to be taken on the 30 V range, the following command could be given to modify the input range without affecting any other parameters:

```
SENSe:VOLT:DC:RANGE 30
```

command could be given to modify the input range without affecting any of the other parameters. At this point, a READ? command would acquire 5 measurements on the 30 V range.

Next, suppose a single AC voltage measurement of unknown amplitude was to be performed.

The command:

```
meas:volt:ac?
```

would perform and return a single AC voltage measurement. Since [<Expected Value>] and [<Resolution>] is not specified, the card would autorange (DEFault) to find the input signal amplitude, using a 200 ms aperture. For example, if the input value were 12 V, the returned value would be:

```
+1.20000E+01<LF>
```

At this point, the AC Voltage measurement configuration is set to a count of 1 on the 30 V range with an aperture of 200 ms.

CONFigure Command. The next level of the hierarchy below the MEASure commands are the CONFigure commands. The CONFigure commands are used to modify the appropriate measurement configurations without taking any measurements. For example, the command

```
configure:voltage:ac 3
```

would cause the next READ? command to perform a 3 Volt AC measurement.

Related to the CONFigure commands are the SENSe and INPut commands. Like the CONFigure commands, they are used to modify the appropriate measurement configurations without taking any measurements. However, they provide a much greater degree of control over the acquisition parameters, as they can modify individual parameters in the configuration table of a function. For example the SENSe commands can modify parameters such as the range, aperture, and count, and autoranging setups of a function.

READ? Query. The next level down in the hierarchy is the READ? query. It is at this level that data is actually acquired and stored by the card. The READ? command uses the setups defined in the commands above to set the acquisition parameters and input relays to their appropriate state.

Implicit in the READ? command is an INITiate:IMMediate execution, which causes the card to acquire the data, and store it in memory. When the data has been acquired, a FETCh? command is executed to return the results. Note that inherent in the READ? query is the possibility of timeouts waiting for the card to complete its acquisition. For example, if the timeout value is set to 1 second and the acquisition process takes 2 seconds, a timeout will occur.

Alternative to the READ? commands are the INITiate, *STB?, and FETCh? commands. These commands allow greater control over when and how data is acquired by defining explicit conditions related to the data acquisition. Using these commands allow starting an acquisition, querying the status to determine when the measurement is complete, and then fetching the results.

FETCh? Finally, at the bottom of the hierarchy is the FETCh? command, which returns the values acquired by the card. The values in the buffer can also be acted upon by the CALCulate commands.

Examples

The following examples illustrate the operation of the hierarchy, and the interrelationships of the commands.

Command	Response
MEASure:VOLTage:DC?3,.2	Set the input function to DC volts, 1 measurement, the 3 V range, and the aperture to 200 ms. Take the measurement and return the results
CONFIGURE:VOLTAGE:AC	Set the input function to AC volts, DEFault (autorange), and 200 ms aperture
READ?	+12.3456E + 00<LF> Initiate the card, and return the AC value read
configure?	:SCAL:VOLT:AC30, 0.0001
sense:volt:ac:range 2	Set the AC voltage range to the closet range above 2 V (3 V range)
conf?	:VOLT:AC 3,0.00001 Note the resolution automatically changed to reflect six significant digits
conf:volt:dc	Set the input function to DC volts, DEFault (autorange), and six digit resolution
conf?	SCAL:VOLT:AC 300.0, 0.001 Autorange starting at 300 Volts setup by the MEASure command above, six significant digits
read?	+1.23456E + 02<LF>
sense:function:volt:ac	Change the input function back to AC
conf?	:VOLT:AC 3,0.00001
INITiate:IMMediate	Initiate an acquisition with the current AC function configuration parameters <WAIT FOR MEASUREMENT COMPLETE STATUS>
fetch?	+1.23456E + 02<LF> AC value

AutoRanging. The MEASure, CONFigure, and SENSe:RANGe:AUTO commands controls when the card performs autoranging. For the SENSe command, the syntax is:

```
SENSe:RANGe:AUTO < ON | OFF | ONCE >
```

If ONCE is programmed, the card will perform an auto-range each time a MEASure?, READ? or INITiate command is received, prior to the first measurement only. For example, the command MEAS:ARR? 100 would autorange prior to the first measurement, and would then use the range determined for making the 100 measurements. If any measurement exceeded the determined range, a +9.99E+09 (or -9.99E+09) value is stored as the measurement value.

If ON is programmed, the card continuously monitors the level of the input signal for every measurement taken. If the signal exceeds the current range, or falls below 10% of the range, the card will autorange to seek the appropriate range, and retake the measurement on the new range. If the signal is right at the 10% point, the card slides the 10% value +0.0/–0.5 % to prevent pinging between ranges.

If OFF is programmed, no autoranging is performed.

Note that autoranging is individually programmable for each function of the card.

Autozeroing. The calibration procedure of the DMM includes commands to calculate and store, in nonvolatile memory, the offset value of every function and range on the card. After power-on, these values are read from the nonvolatile memory, and used to automatically compensate for the offset of every measurement taken by the card.

Because offsets drift with time and temperature, the CALibrate:AUTO:ZERO command can be used to rezero the card while it's in operation. The syntax of the command is

```
CALibrate:AUTO:ZERO < ON | OFF | ONCE >
```

Depending upon what you program for CALibrate:AUTO:ZERO, the card performs the following operations:

Command	Response
ONCE	The card immediately performs an autozero for the function and range currently in effect. All subsequent measurements for that function and range are then compensated by the calculated value. Note that if data sampling is in progress, issuing this command will cause the acquisition to be aborted.
ON	The card performs an autozero each time a MEASure?, READ? or INITiate command is received, *prior to the first measurement,* and whenever a measurement is taken which changes either the current function, or the current range. For example, if autoranging and autozeroing were enabled, issuing a MEAS? command would cause the card to first seek the appropriate range for the input, and once found, would perform an autozero. If during the measurement process the range changed, an autozero would be performed automatically on the new range.
OFF	Disables autozeroing.

***NOTE**. Note that the autozero setup is valid for all functions and ranges of the card. That is, the autozeroing cannot be individually programmed for each function. Also the autozero values are NOT stored in nonvolatile memory.*

Using VXI*plug&play* Driver Function Calls

The main objective of the VXI*plug&play* Systems Alliance is to increase ease of use for end users of VXI technology through open, multi-vendor VXI systems. To achieve *plug&play* capability for multi-vendor system software tools, the Alliance requires that a unified I/O driver software architecture, accommodating all points of view in the market, be used as a foundation. The VXI*plug&play* Systems Alliance jointly defined and implemented the Virtual Instrument Software Architecture (VISA) I/O driver architecture that provides this unified foundation.

The key beneficiaries of this objective are the end-users of VXI technology. End-users can be confident that products and services from vendors conforming to VXI*plug&play* frameworks can be integrated more easily than products that lack a shared system-level framework for application.

Using VISA Function Calls

The following code was written in LabWindows/CVI using VISA calls for communication to the instrument. This is an example of the basic message based instrument communication. For more in-depth discussion see the documentation distributed with your VISA software installation diskettes.

Before You Begin. Run Start Up Resource Manager (or SURM) or Resource Manager (RESMAN) before running the program unless your communication bus is GPIB. With GPIB, resource management of VXI instruments is automatically handled during power-on.

```
#include <string.h>

#include <stdio.h>

#include <visa.h>

int main(void)

{

   ViSession defaultRM,instrHndl;

/*Communication channels*/

   ViUInt32 retCnt;
```

/*The actual character count of sent or received msgs*/

```
ViStatus retError;
```

/*Used to check for errors*/

```
ViChar statusMessage[256];
```

/*Buffer to hold the error messages*/

```
ViChar cmd[256];
```

/*Buffer for commands that are sent to the instrument*/

```
ViChar rcv[256];
```

/*Buffer for holding responses from the instrument*/

Initialize the system. Obtain a session to the Default Resource Manager resource.

```
retError = viOpenDefaultRM (&defaultRM);
```

/*Negative return values are errors and a readable string describing the error is obtained by calling the viStatusDesc function. This error checking example is not all encompassing. Status values greater than VI_SUCCESS (0) have meanings as well. Success codes and warnings are positive.*/

```
if(retError<VI_SUCCESS){viStatusDesc (defaultRM,
retError,statusMessage);
   printf("%s\n",statusMessage); return -1;}
```

/*Open communication with a VXI device at VXI logical address of 3. If this was for a GPIB communication system the “VXI0::3::INSTR” would be changed to “GPIB0::3::INSTR”. Notice that the session to the Default Resource Manager must be used here as the first parameter. The new session to the device is instrHndl and will be used to communicate with the instrument.*/

```
retError = viOpen (defaultRM, "VXI0::3::INSTR", VI_NULL,
VI_NULL,&instrHndl);
```

/*If retError is less than VI_SUCCESS here the status message must still be obtained by using the default RM session. There would not be a valid instrHndl session.*/

```
if(retError<VI_SUCCESS){viStatusDesc (defaultRM,
retError,statusMessage);
   printf("%s\n",statusMessage); return -1;}
```

/*Set the timeout attribute for the new session for communication to 5 seconds. Other attributes include fastdata channel setup and read/write termination characters/methods./*

```
retError = viSetAttribute (instrHndl, VI_ATTR_TMO_VALUE, 5000);

if(retError<VI_SUCCESS){viStatusDesc (defaultRM, retError,

   statusMessage);

   printf("%s\n",statusMessage);}
```

/*The viWrite function sends the “*IDN?” command to the instrument and the viRead function does an input request to obtain the response to the query.*/

```
sprintf(cmd,"*IDN?");

retError = viWrite (instrHndl, cmd, strlen(cmd) , &retCnt);

if(retError<VI_SUCCESS){viStatusDesc (instrHndl, retError,

   statusMessage);

   printf("%s\n",statusMessage);}
```

/*Set the timeout attribute for the new session for communication up to five seconds. Other attributes include fast data channel setup and read/write termination characters and methods.*/

```
retError = viRead (instrHndl, rcv, 256, &retCnt);

if(retError<VI_SUCCESS){viStatusDesc (instrHndl, retError,

   statusMessage);

   printf("%s\n",statusMessage);}
```

/*In this example the instrument response is printed to the standard output window.*/

```
rcv[strlen(rcv)]='\0';

printf("%s\n",rcv);
```

/*Close down the system. Be sure to close both the defaultRM and instrHndl sessions.*/

```
retError = viClose (instrHndl);

if(retError<VI_SUCCESS){viStatusDesc (instrHndl, retError,

   statusMessage);

   printf("%s\n",statusMessage);}

retError = viClose (defaultRM);

if(retError<VI_SUCCESS){viStatusDesc (defaultRM, retError,

   statusMessage);

   printf("%s\n",statusMessage);}

}
```

WARNING

The following servicing instructions are for use only by qualified personnel. To avoid injury, do not perform any servicing other than that stated in the operating instructions unless you are qualified to do so. Refer to all Safety Summaries before performing any service.

Appendix D: Calibration

This section contains calibration procedures for the following modules in the VX4101:

- Digital Multimeter (DMM)
- Counter

Calibration for the DMM

The DMM calibration procedure consists of the following:

- Null and gain calibration in all DC ranges
- All AC TRMS (DC Coupled) ranges
- All 4-wire resistance ranges and the two current ranges.

An additional 2-wire Ω null calibration in the 30 Ω range is required and may be used to compensate for 2-wire Ω field wiring.

Nonzero values specified below may be ±10% of the value specified, with a maximum value equal the value of the range. Enter the exact value as displayed on the calibrator (or as measured by an accurate system DMM if a DMM is used as a transfer standard. The tolerances shown following the applied values below are the accuracy to which the `cal:val` command argument must be known. A calibration standard meeting those accuracies must be used (or an accurate system DMM used as a transfer standard).

***NOTE**. See the CALibration:VALue command description in the SCPI Command section of this user manual for a description of the calibration commands and additional information about the calibration of the VX4101 DMM.*

Between applying an external voltage and sending the cal:val command in the procedures below, sufficient time should be allowed for both the calibration source and the VX4101 input circuit to settle to the tolerance shown. A good way to assure this in an automated program is to take continuous measurements with the VX4101 DMM after the voltage, resistance or current is applied, make sure that the reading is stable within a value equal to 25% of the accuracy of the measurement, and that the VX4101 measurement is reasonable.

Before You Begin

Connect calibrator voltage outputs to VX4101 voltage inputs.

DC Mode Calibration Procedure

Performing this procedure will calibrate the DMM through its entire range of DC measurements. This procedure is separated into phases to permit calibration of all ranges, but should be run in its entirety. Perform the following steps to calibrate the DMM in DC mode:

DC Range .03 VDC with 10 MΩ Impedance

This part of the procedure calibrates the DMM for 0.03 VDC with 10 MΩ impedance.

1. Program the VX4101 as follows:

```
conf:dc .03
```

2. Program the VX4101 as follows:

```
inp:imp 10e6; init
```

3. Apply 0.0 VDC ±100 nVDC

4. Program the VX4101 as follows:

```
cal:val 0
```

5. Apply 0.029 VDC±1.9 μVDC

6. Program the VX4101 as follows:

```
cal:val .029
```

DC Range .3 VDC with 10 MΩ Impedance

This part of the procedure calibrates the DMM for 0.3 VDC with 10 MΩ impedance.

7. Program the VX4101 as follows:

```
conf:dc .3;init
```

8. Apply 0.0 VDC ±1 μVDC

9. Program the VX4101 as follows:

```
cal:val 0
```

10. Apply 0.290 VDC 5.3 μVDC

11. Program the VX4101 as follows:

`cal:val .29` (or any other desired value)

DC Range 3 VDC with 10 MΩ Impedance

This part of the procedure calibrates the DMM for 03 VDC with 10 MΩ impedance.

12. Program the VX4101 as follows:

`conf:dc 3;init`

13. Apply 0.0 VDC ±4 μVDC

14. Program the VX4101 as follows:

`cal:val 0`

15. Apply 2.90 VDC ±35 μVDC

16. Program the VX4101 as follows:

`cal:val 2.9` (or any other desired value)

DC Range 30 VDC with 10 MΩ Impedance

This part of the procedure calibrates the DMM for 030 VDC with 10 MΩ impedance.

17. Program the VX4101 as follows:

`conf:dc 30;init`

18. Apply 0.0 VDC ±40 μVDC

19. Program the VX4101 as follows:

`cal:val 0`

20. Apply 29.0 VDC ±450 μVDC

21. Program the VX4101 as follows:

`cal:val 29` (or any other desired value)

DC Range 300 VDC with 10 MΩ Impedance

This part of the procedure calibrates the DMM for 300 VDC with 10 MΩ impedance.

22. Program the VX4101 as follows:

```
conf:dc;300;init
```

23. Apply 0.0 VDC ±400 μVDC

24. Program the VX4101 as follows:

```
cal:val 0 init
```

25. Apply 290 VDC ±4.5 mVDC

26. Program the VX4101 as follows:

`cal:val 290` (or any other desired value)

DC Range .03 VDC with 10 MΩ Impedance

This part of the procedure calibrates the DMM for .003 VDC with 10 MΩ impedance.

27. Program the VX4101 as follows:

```
conf:dc .03
```

28. Program the VX4101 as follows:

```
inp:imp 10e9;init
```

29. Apply 0.0 VDC ±100 nVDC

30. Program the VX4101 as follows:

```
cal:val 0
```

31. Apply 0.029 VDC ±1.9 μVDC

32. Program the VX4101 as follows:

`cal:val .029` (or any other desired value)

DC Range 0.3 VDC with 10 GΩ Impedance

This part of the procedure calibrates the DMM for 0.3 VDC with 10 GΩ impedance.

33. Program the VX4101 as follows:

```
conf:dc .3;init
```

34. Apply 0.0 VDC ±1 μVDC

35. Program the VX4101 as follows:

```
cal:val 0 init
```

36. Apply 0.290 VDC ±5.3 μVDC

37. Program the VX4101 as follows:

`cal:val .29` (or any other desired value)

DC Range 3VDC with 10 GΩ Impedance

This part of the procedure calibrates the DMM for 3 VDC with 10 GΩ impedance.

38. Program the VX4101 as follows:

```
conf:dc 3;init
```

39. Apply 0.0 VDC ±4 μVDC

40. Program the VX4101 as follows:

```
cal:val 0
```

41. Apply 2.90 VDC ±35 μVDC

42. Program the VX4101 as follows:

`cal:val 2.9` (or any other desired value)

TRMS AC (DC Coupled) Mode Calibrations

This procedure will calibrate the DMM throughout its range of TRMS AC (DC Coupled) measurements. This procedure is separated into phases to permit calibration of all ranges, but should be run in its entirety. The procedure is as follows:

AC/DC Coupled Range .03 VDC or RMS

This part of the procedure calibrates the DMM for AC/DC Coupled Range 0.03 VDC or RMS.

1. Program the VX4101 as follows:

   `conf:acdc .03;init`

2. Apply 0.009 VRMS ±1.9 μVRMS at 1 KHz
3. Program the VX4101 as follows:

   `cal:val .003` (or any other desired value)

4. Apply .029 VRMS ±1.9 μVRMS at 1 KHz
5. Program the VX4101 as follows:

   `cal:val .029` (or any other desired value)

AC/DC Coupled Range 0.3 VDC or RMS

This part of the procedure calibrates the DMM for AC/DC Coupled Range 0.3 VDC or RMS.

6. Program the VX4101 as follows:

   `conf:acdc .3 init`

7. Apply .03 VRMS ±1.9 μVRMS at 1 KHz
8. Program the VX4101 as follows:

   `cal:val .03` (or any other desired value)

9. Apply .290 VRMS ±5.3 μRMS at 1 KHz
10. Program the VX4101 as follows:

    `cal:val .29` (or any other desired value)

AC/DC Coupled Range 3 VDC or RMS

This part of the procedure calibrates the DMM for AC/DC Coupled Range 3 VDC or RMS.

11. Program the VX4101 as follows:

`conf:acdc 3 init`

12. Apply .300 VRMS ±5.3 μVRMS at 1 KHz

13. Program the VX4101 as follows:

`cal:val .3` (or any other desired value)

14. Apply 2.90 VRMS ±35 μVRMS at 1 KHz

15. Program the VX4101 as follows:

`cal:val 2.9` (or any other desired value)

AC/DC Coupled Range 30 VDC or RMS

This part of the procedure calibrates the DMM for AC/DC Coupled Range 30 VDC or RMS.

16. Program the VX4101 as follows:

`conf:acdc 30 init`

17. Apply 3.0 VRMS ±35 μVRMS at 1 kHz

18. Program the VX4101 as follows:

`cal:val 3` (or any other desired value)

19. Apply 29.0 VRMS ±450 μVRMS at 1 kHz

20. Program the VX4101 as follows:

`cal:val 29` (or any other desired value)

AC/DC Coupled Range 300 VDC or RMS

This part of the procedure calibrates the DMM for AC/DC Coupled Range 300 VDC or RMS.

21. Program the VX4101 as follows:

`conf:acdc 300 init`

22. Apply 30.0 VRMS ±450 μVRMS at 1 kHz

23. Program the VX4101 as follows:

`cal:val 30 init` (or any other desired value)

24. Apply 290 VRMS ±4.5 mVRMS at 1 kHz

25. Program the VX4101 as follows:

`cal:val 29` (or any other desired value)

***NOTE**. TRMS AC (AC Coupled) ranges use the above calibration information. A separate TRMS AC (AC Coupled) calibration is not required.*

Resistance Mode Calibration Procedure

This procedure will calibrate the DMM through its entire range of resistance measurement settings. The procedure is separated into several phases, but should be run in its entirety. The procedure is as follows:

Resistance 30 Ω 4-Wire

This part of the procedure calibrates the DMM for 30 Ω Resistance, 4-wire.

1. Program the VX4101 as follows:

```
conf:fres 30;init
```

2. Apply 0.0 Ω ±1.0 mΩ

3. Program the VX4101 as follows:

```
cal:val 0
```

4. Apply 10.0 Ω ±1.3 mΩ

5. Program the VX4101 as follows:

`cal:val 10` (or any other desired value)

Resistance 30 Ω 2-Wire

This part of the procedure calibrates the DMM for 30 Ω Resistance, 2-wire.

6. Program the VX4101 as follows:

```
conf:res 30;init
```

7. Apply 0.0 Ω ±1.0 mΩ

8. Program the VX4101 as follows:

```
cal:val 0
```

9. Apply 10.0 Ω ±1.3 mΩ

10. Program the VX4101 as follows:

`cal:val 10` (or any other desired value)

Resistance 300 Ω

This part of the procedure calibrates the DMM for 300 Ω resistance.

11. Program the VX4101 as follows:

`conf:fres 300;init`

12. Apply 0.0 Ω ±1.0 mΩ

13. Program the VX4101 as follows:

`cal:val 0`

14. Apply 100.0 Ω ±3 mΩ

15. Program the VX4101 as follows:

`cal:val 10` (or any other desired value)

Resistance 3 kΩ

This part of the procedure calibrates the DMM for 300 kΩ resistance.

16. Program the VX4101 as follows:

`conf:fres 3000;init`

17. Apply 0.0 Ω ±4.0 mΩ

18. Program the VX4101 as follows:

`cal:val 0`

19. Apply 1000.0 Ω ±16 mΩ

20. Program the VX4101 as follows:

`cal:val 100` (or any other desired value)

Resistance 30 kΩ

This part of the procedure calibrates the DMM for 30 kΩ resistance.

21. Program the VX4101 as follows:

`conf:fres 30000;init`

22. Apply 0.0 Ω±40 mΩ

23. Program the VX4101 as follows:

`cal:val 0`

24. Apply 10 kΩ ±.16 Ω

25. Program the VX4101 as follows:

`cal:val 10e3` (or any other desired value)

Resistance 300 kΩ

This part of the procedure calibrates the DMM for 300 kΩ resistance.

26. Program the VX4101 as follows:

`conf:fres 300e3;init`

27. Apply 0.0 Ω ±.4 Ω

28. Program the VX4101 as follows:

`cal:val 0`

29. Apply 100 kΩ ±1.8 Ω

30. Program the VX4101 as follows:

`cal:val 100e3` (or any other desired value)

Resistance 3 MΩ

This part of the procedure calibrates the DMM for 3 MΩ resistance.

31. Program the VX4101 as follows:

`conf:fres 3e6;init`

32. Apply 0.0 Ω ±8 Ω

33. Program the VX4101 as follows:

`cal:val 0`

34. Apply 1.0 MΩ ± 48 Ω

35. Program the VX4101 as follows:

`cal:val 1e6` (or any other desired value)

Resistance 30 MΩ

This part of the procedure calibrates the DMM for 30 MΩ resistance.

36. Program the VX4101 as follows:

`conf:fres 30e6;init`

37. Apply 0.0 Ω ±160 Ω

38. Program the VX4101 as follows:

```
cal:val 0
```

39. Apply 10 MΩ ±2000 Ω

40. Program the VX4101 as follows:

`cal:val 10e6` (or any other desired value)

Resistance 300 MΩ

This part of the procedure calibrates the DMM for 300 MΩ resistance.

41. Program the VX4101 as follows:

```
conf:fres 300e6;init
```

42. Apply 0.0 Ω ±1600 Ω

43. Program the VX4101 as follows:

```
cal:val 0
```

44. Apply 100 MΩ ±200 kΩ

45. Program the VX4101 as follows:

`cal:val 100e6` (or any other desired value)

***NOTE**. 2-wire resistance Ω ranges use the above calibration information. A separate 2-wire Ω calibration (other than at 30 Ω) is not required.*

Current Mode Calibration Procedure

This procedure calibrates the DMM in current mode throughout its range of current measurements. This procedure is separated into phases to permit calibration at all ranges, but is intended to be run in its entirety. The procedure is as follows:

Current .15A

This part of the procedure calibrates for 0.15A current.

1. Program the VX4101 as follows:

   `conf:curr .15;init`

2. Apply 0.0 mA ±8 μA
3. Program the VX4101 as follows:

   `cal:val 0`

4. Apply 140 mA ±35 μA
5. Program the VX4101 as follows:

   `cal:val .14` (or any other desired value)

Current 1A

This part of the procedure calibrates for 1A current.

6. Program the VX4101 as follows:

   `conf:curr 1;init`

7. Apply 0.0 mA ±14 μA
8. Program the VX4101 as follows:

   `cal:val 0`

9. Apply 900 mA ±0.28 mA

***NOTE**. This measurement takes extra time to settle*

---

10. Program the VX4101 as follows:

    `cal:val 1` (or any other desired value)

Calibration for the Counter

This procedure shows you how to calibrate the Counter.

Prerequisites

It is assumed the module has completed its power-on self test. For information on specific commands or syntax, please review the *Syntax and Commands* section.

Equipment Required

You need the following equipment to calibrate the Counter:

- Precision DC calibrator
- Function generator (1 kHz square wave at ±2.5 V)
  (10 MHz square wave at ±1.0 V)
- High frequency signal generator (Required for factory calibration only)
  (1000 MHz Sine @–21 dBm, 10 MHz Sine @ 0 dBm)
- High accuracy clock standard, such as a rubidium clock source (10 MHz @ 0 dBm)

What You Should Know About

Before you begin the calibration procedures for the Counter, you should understand the following:

Channel Specifications. For the calibration procedures referenced below, [<channel>] is either 1 or 2, depending on the channel being calibrated. For example, the command CALibrate[<channel>]:ZERO could be specified as one of the following:

cal1:zero	to zero channel 1, or
cal2:zero	to zero channel 2

If [<channel>] is not specified (e.g. cal:zero), channel 1 is the default.

For channel 1, the input signal should be connected to the CH 1 input of the Counter.

For channel 2, the input signal should be connected to the CH 2 input of the Counter.

NOTE. *For the ARM calibrations, the input should be connected to the SMB connector.*

For the channel 3 (factory) calibrations, the input should be connected to the SMA Connector, if this option is included.

Determining Status. To determine the status of a calibration command, do the following:

1. Issue the following query:

```
STATus:OPERation:CONDition?
```

***NOTE**. Bit 0 in the response data indicates the following:*

Bit 0 = 0	*Indicates that the calibration command is complete*
Bit 0 = 1	*Indicates that the calibration is in progress*

2. You can issue the following query to determine if there was a failure in the calibration:

```
SYSTem:ERRor?
```

***NOTE**. When polling the cards for calibration complete, it is recommended that the polling period be greater than 1 ms to minimize the overhead incurred for processing the query while the calibration is in progress. An alternative method to eliminate the polling overhead is to program the card to generate an SRQ (service request) interrupt when the calibration bit (described above) changes from a 1 to a 0*

About the Adjustment Procedures. The individual channel adjustments detailed in Step Two below should be performed in the sequence in which they appear for the channel 1 and channel 2 inputs of the Counter. Once these steps have been performed, the command function and cross channel calibrations steps in Step Three should be executed.

Step One: Initialize the instrument

To initialize the instrument and prepare it for adjustment, do the following:

1. Send the following command to select the counter function of the VX4101:

```
INSTrument:SELect COUNTER
```

2. Connect the calibrator to the channel being calibrated.

Step Two: Individual Channel Adjustments

Adjust each channel as follows:

Offset Adjustment. To adjust offset, do the following:

1. Set the calibrator to 0.0 ±0.001 V.
2. For the channel being calibrated, send the command

   ```
   CALibrate[<channel>]:ZERO
   ```

   This command will take approximately 17 seconds to execute.

Preamp Linearization. To adjust preamp linearization:

1. Set the calibrator to 0.5 V ±0.1%.
2. For the channel being calibrated, send the command

   ```
   CALibrate[<channel>]:LINearity
   ```

This command takes approximately 2 seconds to execute.

Gain Correction. To adjust gain correction, perform the following steps for each of the voltages specified below:

1. For each of the following voltages, set the calibrator to the voltage value ±0.1%:

<voltage> Value
50
5
0.5
–50
–5
–0.5

2. For the channel being calibrated, send the following command:

   ```
   CALibrate[<channel>]:VALue <voltage>
   ```

   where <voltage> is the value in the table above (e.g. cal1:val 50.0).

This command takes approximately 2 seconds to execute for each voltage value.

3. Set the Offset Gain Calibration as follows:

   Set the calibrator to –1.0 V ± 0.1%

4. For the channel being calibrated, send the command

```
CALibrate[<channel>]:VALue -1.0
```

   This command takes approximately 2 seconds to execute.

Hysteresis Calibration. Since this function uses an internal reference, it requires no external inputs. The calibrator should either be disconnected or set to 0 V for this step.

To calibrate hysteresis, do the following:

1. For the channel being calibrated, send the following command:

```
CALibrate[<channel>]:HYSTeresis
```

This command takes approximately 75 seconds to execute.

Low Frequency Compensation Adjustment. To adjust low frequency compensation:

1. Connect the function generator to the channel being calibrated, and set it to output a 1 kHz square wave at ±2.5 ±0.1 V (1 MΩ load impedance).

2. For the channel being calibrated, send the command

```
CALibrate[<channel>]:LFCOmp
```

   This command takes approximately 6 seconds to execute.

Step Three: Common Function and Cross-Channel Calibration Adjustments

After performing all of the previous procedures steps for both channels, perform the following required additional steps as required:

ARM Input Zero and Gain Correction. To adjust ARM Input zero and gain Correction:

1. Set the calibrator to 0.00 ±0.001 V, and connect it to the ARM input of the card.

2. Send the command

```
CALibrate:ARM:VALue 0.0
```

   This command takes approximately 0.5 second to execute.

3. Set the calibrator to 20.0 V ±0.1%.

4. Send the command

   `CALibrate:ARM:VALue 20.0`

   This command takes approximately 0.5 seconds to execute.

Digital Time Interpolation. To calibrate the Digital Time Interpolation, do as follows:

1. Connect the function generator for a square wave at ±0.5 V ±0.1 V @10 MHz to the channel 1 input (50 Ω load impedance).

2. Send the command:

   `CALibrate:DTI`

   This command takes approximately 16 seconds to execute.

Cross Channel Delays. To calibrate the cross channel delays, do as follows:

1. Using a 50 Ω RF splitter and equal length cables, connect the function generator to both the Channel 1 and Channel 2 inputs. Set up the function generator for a 10 MHz square wave at ±2.0 V ±0.1 V (±0.5 V at each 50 Ω input).

2. For determining the channel 1 cable delay, send the command:

   `CALibrate1:DELay`

   This command takes approximately 3 seconds to execute.

3. For determining the channel 2 cable delay, send the command:

   `CALibrate2:DELay`

   This command takes approximately 3 seconds to execute.

4. For determining the channel 1 to 2 cross channel delay, send the command:

   `CALibrate:DELay 12`

   This command takes approximately 3 seconds to execute.

5. For determining the channel 2 to 1 cross channel delay, send the command:

   `CALibrate:DELay 21`

   This command takes approximately 3 seconds to execute.

***NOTE**. If an invalid input signal is present, these commands will timeout and generate an error after approximately 5 seconds.*

Factory Calibration

The following additional steps are performed during initial factory calibration, when the channel 3 option is available. This step is not part of the normal calibration sequence, and is included here for completeness only.

Channel 3 Pre-Scaler BIAS Adjustment.

1. Set up the high frequency source for a sine wave at 1000 MHz ±100 kHz at –21 dBm, and connect it to the prescaler (channel 3) input.

2. Send the command

```
CALibrate3:BIAS
```

This command takes approximately 2 seconds to execute.

System Clock Calibration

There are two clock sources available which the Counter can use as its reference oscillator. These are the internal 10 MHz clock located on the VX4101 CPU card, and the VXI backplane 10 MHz clock.

Either or both of these sources could be used in the actual operation of the card. It is recommended that both sources be calibrated.

Selecting a noncalibrated source during operation will generate an error. For best results, the VXI clock should be calibrated in the actual system in which it is used.

Calibrate the clock sources as follows:

1. Connect the High Accuracy Clock Standard into the Channel 1 input, and set it to 10 MHz @ 0 dBm

2. Select the source of the clock to be calibrated, as follows:

For calibrating the internal clock, send the command

```
SOURce:ROSCillator INTernal
```

For calibrating the VXI backplane clock, send the command

```
SOURce:ROSCillator CLOCk10
```

3. Issue the following command:

```
CALibrate:ROSCillator
```

This command takes approximately 11 seconds to execute.

***NOTE**. If an invalid input signal is present, this command will time out and generate an error after approximately 12 seconds.*

Appendix E: Counter Performance Verification Procedure

This procedure verifies the performance of the DMM portion of the VX4101. It is not necessary to complete the entire procedure if you are only interested in a specific performance area.

The performance verification is divided into test sets, each testing a specific performance area. The test sets are as follows:

- Period
- Sensitivity
- Rise Time
- Frequency Ratio
- AC Voltage Accuracy
- DC Voltage Accuracy
- AC High-Voltage Accuracy
- DC High-Voltage Accuracy
- Arm Input Function

Required Skills

The following skills are required to perform this procedure:

- Thorough knowledge of test instrument operation and proper measurement techniques
- Knowledge of VXIbus system components and command language programming
- Ability and facility to construct interconnections and fixtures as needed
- Experience in safety precautions when handling high voltage

General Information and Conventions

The VX4101 Counter performance verification requires a calibration standard with basic system accuracies of 0.1% ±1 mVDC, 1 % (0.1 dB) ± 0.1 mV_{RMS}. Many of the tests require less accuracy. Each test record shows the required test signal accuracy for each test.

Equipment Required

Table E–1: Required Equipment

Description	Minimum Requirements	Examples
High Frequency Source	0.1% +1 mVDC 1 % (0.1 dB) + 0.1 mV_{RMS}	HP ESG-4000A Signal Generator
Precision AC/DC Source		Wavetek Datron Calibrator 4800
Low Frequency Source		HP3325 Synthesizer/Function Generator

Considerations for Automated Procedures

If the test steps are automated, make sure the calibration output has had adequate time to settle prior to taking the measurement.

Prerequisites

- Allow 20 minutes of warm-up before testing
- The VXI backplane CLOCK10 signal should be synchronized to all AC signal sources to eliminate Time Base Error considerations from the accuracy specifications. The procedure for synchronizing the CLOCK10 signal is described below. All tests must be run in the order shown because the preceding step may satisfy setup conditions for the next step.

Synchronizing the CLOCK10 Signal

This procedure will synchronize the CLOCK10 signal to all AC sources. You should wait eight seconds between each step. The procedure is as follows:

1. Apply a 10 MHz @1V_{PP} ± 5% signal to the channel one input of the Counter. Wait eight seconds before performing any other operations.
2. Send the following command to the VX4101:

   ```
   sour:rosc cloc
   ```

   Wait eight seconds before performing the next step
3. Send the following command to the VX4101:

   ```
   cal:rosc
   ```

   Wait eight seconds before performing any other operations

About the Test Records

The test records show the maximum and minimum readings for the following test sets:

- Period
- Sensitivity
- Rise Time
- Frequency ratio
- AC voltage accuracy
- DC voltage accuracy
- AC high-voltage frequency accuracy
- DC high-voltage frequency accuracy
- Arm input

Using the Test Record

Each type of test in the test record contains the following:

- A description of the type of performance test, for example, Period Single-Shot Test
- Column showing the channel being tested
- A column to record the reading you receive from each performance test type
- Columns with upper and lower values
- A column to note whether the instrument passed or failed the test

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ fail
Period Single-Shot	One	100.25 ns		99.75 ns	
	Two	100.25 ns		99.75 ns	

Figure E–1: Sample Test Result Record

Running the Tests

Perform the following tasks to run each test:

1. Take the measurement and record the reading in the Reading column.
2. Compare the reading to the figures in the Upper and Lower Value columns.
3. Enter the pass or fail status in the Pass/Fail column.

Before You Begin

Before you begin, do the following:

- Photocopy the Test Records to record performance verification results for your module
- Run the self-test procedure described below

Self-Test Procedure

The self-test is performed by sending a command and query to the VX4101 module. The self-test executes in just under ten seconds.

To initiate the self-test, do the following:

1. Send the following commands and query:

   ```
   *rst
   inst:sel counter
   syst:tim 50
   sour:rosc cloc
   test:all?
   ```

2. The response should indicate that the self-test has passed. If the self-test fails, send the following query for more detailed information about the failure:

   ```
   syst:err?
   ```

Test Records

Table E–2: Period Verification Measurement Results

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
Period Single-Shot	One	100.342 ns		99.658 ns	
	Two	100.342 ns		99.658 ns	
500 MHz	One	2.254 ns		1.746 ns	
	Two	2.254 ns		1.746 ns	
0.1 Second Averaged	One	2.002 ns		1.998 ns	
	Two	2.002 ns		1.998 ns	
Equivalent Input Channel Noise	One	100.090 ms		99.910 ms	
	Two	100.090 ms		99.910 Hz	

Table E–3: Sensitivity Test Measurement Results

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
15 MHz w/ 20 MHz Filter	One	15,000,000.2008 Hz		14,999,999.7992 Hz	
	Two	15,000,000.2008 Hz		14,999,999.7992 Hz	
15 MHz w/100 MHz Filter	One	15,000,000.2424 Hz		4,999,999.7576 Hz	
	Two	15,000,000.2424 Hz		4,999,999.7576 Hz	
75 MHz w/100 MHz Filter	One	75,000,001.101 Hz		74,999,998.899 Hz	
	Two	75,000,001.101 Hz		74,999,998.899 Hz	
75 MHz Sensitivity without Filter	One	75,000,001.142 Hz		74,999,998.858 Hz	
	Two	75,000,001.142 Hz		74,999,998.858 Hz	
250 MHz	One	250,000,001.142 Hz		249,999,998.858 Hz	
	Two	250,000,001.142 Hz		249,999,998.858 Hz	
500 MHz	One	500,000,010.142 Hz		499,999,989.858 Hz	
	Two	500,000,010.142 Hz		499,999,989.858 Hz	
Channel 3 Sensitivity	Three	3,000,000,010.253 Hz		2,999,999,989.747 Hz	

Table E–4: Rise Time Measurement Results

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
Rise Time 1 kHz	One	354 µs		238 µs	
	Two	354 µs		238 µs	
Rise Time 200 MHz	One	1.968 ns		0.984 ns	
	Two	1.968 ns		0.984 ns	

Table E–5: Frequency Ratio Measurement Results

Test Description	Upper Value	Reading	Lower Value	Pass/ Fail
Channel One: Channel Two	1.000000005		0.999999995	
Channel Two: Channel Three	1.000000005		0.999999995	

Table E–6: AC Voltage Accuracy Measurement Results

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
0.9 V_{P-P} @100 kHz	One	0.9925 V		0.8075 V	
	Two	0.9925 V		0.8075 V	
0.36 V_{P-P} @100 kHz	One	0.412 V		0.308 V	
	Two	0.412 V		0.308 V	
0 V_{P-P}	One	25 mV		0 mV	
	Two	25 mV		0 mV	

Table E–7: DC Voltage Accuracy Measurement Results

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
0 VDC	One	25 mV		–25 mV	
	Two	25 mV		–25 mV	

Table E–8: AC High Voltage Frequency Accuracy Measurement Results

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
9.0 V @ 100 kHz	One	9.925 V		8.075 V	
	Two	9.925 V		8.075 V	
90 V_{P-P} @100 kHz	One	99.25 V		80.75 V	
	Two	9.25 V		8.075 V	
3.6 V_{P-P}	One	4.12 V		3.08 V	
	Two	4.12 V		3.08 V	
1.8 V_{P-P}	One	1.96 V		1.64 V	
	Two	1.96 V		1.64 V	

Table E–9: DC High-Voltage Accuracy

Test Description	Channel	Upper Value	Reading	Lower Value	Pass/ Fail
200 VDC	One	217.5 V		182.5 V	
	Two	217.5 V		182.5 V	
–200 VDC	One	–182.5 V		217.5 V	
	Two	–182.5 V		217.5 V	
18 VDC	One	19.6 V		16.4 V	
	Two	19.6 V		16.4 V	
–18 VDC	One	–16.4 V		–19.6 V	
	Two	–16.4 V		–19.6 V	
1.8 VDC	One	1.96 V		1.64 V	
	Two	1.96 V		1.64 V	
–1.8 VDC	One	–1.64 V		–1.96 V	
	Two	–1.64 V		–1.96 V	

Table E–10: Arm Input Function

Test Description	Function Response	Measured Response	Pass/ Fail
0 VDC positive slope	0		
3 VDC positive slope	1		

Table E–10: Arm Input Function (Cont.)

Test Description	Function Response	Measured Response	Pass/ Fail
3 VDC negative slope	0		
0 VDC negative slope	1		

Test Procedures

The test procedures are described below. The procedures are presented in the following sets:

- Period
- Sensitivity
- Rise Time
- Frequency Ratio
- AC Voltage Accuracy
- DC Voltage Accuracy
- AC High-Voltage Accuracy
- DC High-Voltage Accuracy
- Arm Input Function

Period

Perform the procedures below to test Period measurements.

Before You Begin Connect the signal generator to a 50 Ω RF splitter. Connect one of the splitter outputs to channel one and the other to channel two.

1. Period single-shot test:

a. Apply a 4 dBm ± 1 dB, 10 MHz sine wave signal through the splitter to channel one and channel two (10 dBm from the generator).

b. Send the following VX4101 command(s)/query:

```
inp:set:auto once
inp1:imp 50
inp2:imp 50
inp1:coup dc
inp2:coup dc
sens1:func "per"
sens:per:even 1
sens:per:mode even
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "per"
read?
```

   e. Record the reading for channel two.

2. Period 500 MHz test:

   a. Apply 500 MHz @ 4 dBm ± 1 dB to channels one and two.

   b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "per"
inp1:set 1,0
read?
```

   c. Record the reading for channel one.

   d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "per"
inp2:set 1,0
read?
```

   e. Record the reading for channel two.

3. Period 0.1 second averaged test:

   a. Send the following VX4101 command(s)/query to test channel one:

```
sens:per:mode aper
sens1:func "per"
read?
```

   b. Record the reading for channel one.

   c. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "per"
read?
```

   d. Record the reading for channel two.

4. Period equivalent input channel noise < 2 mV test:

   a. Apply 10 Hz @ 4 dBm ± 1 dB to channels one and two.

   b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "per"
inp:set:auto once
inp1:filt on
inp2:filt on
read?
```

   c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "per"
read?
```

e. Record the reading for channel two.

Sensitivity

Follow the procedures below to test sensitivity.

1. 15 MHz sensitivity test with 20 MHz filter:

a. Apply 15 MHz @ –20 dBm ± 1 dB to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "freq"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "freq"
read?
```

e. Record the reading for channel two.

2. 15 MHz sensitivity test with 100 MHz filter:

a. Apply 15 MHz @ –20 dBm ± 1 dB to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "freq"
inp1:filt:freq 100e6
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "freq"
inp2:filt:freq 100e6
read?
```

e. Record the reading for channel two.

3. 75 MHz sensitivity test with 100 MHz filter:

a. Apply 75 MHz @ –20 dBm ± 1 dB to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "freq"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "freq"
read?
```

e. Record the reading for channel two.

4. 75 MHz sensitivity test without filter:

a. Apply 75 MHz @ –20 dBm ±1 dB to channels 1 and 2.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "freq"
inp1:filt off
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "freq"
inp2:filt off
read?
```

e. Record the reading for channel two.

5. 250 MHz sensitivity test:

a. Apply 250 MHz @ –18 dBm ± 1 dB to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "freq"
inp1:set 0.08,0
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query:

```
sens2:func "freq"
inp2:set 0.08,0
read?
```

e. Record the reading for channel two.

6. 500 MHz sensitivity test:

a. Apply 500 MHz @ –12 dBm ± 1 dB to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "freq"
inp1:set 0.159,0
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "freq"
inp2:set 0.159,0
read?
```

e. Record the reading for channel two.

7. Channel three sensitivity test:

a. Apply 3000 MHz @ –25 dBm ± 1 dB to channel three.

b. Send the following VX4101 command(s)/query to test channel three:

```
sens3:func "freq"
read?
```

Rise Time

Follow the procedures below to test rise time.

1. Channel one rise time test:

a. Apply 1 kHz @ 1 V_{P-P} ± 5% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
inp1:filt on
inp2:filt on
inp1:filt:freq 20e6
inp2:filt:freq 20e6
sens1:func "rtim"
inp:set:auto once
sens:rtim:aper 1
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "rtim"
read?
```

e. Record the reading for channel two.

2. 200 MHz rise time test:

a. Apply 200 MHz @ 1 V_{P-P} ± 5% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
inp1:filt off
inp2:filt off
sens1:func "rtim"
inp1:set 1,0
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "rtim"
inp2:set 1,0
read?
```

e. Record the reading for channel two.

Frequency Ratio

Follow the procedures below to test frequency ratio.

1. Channel one: channel two ratio test

Send the following VX4101 command(s)/query:

```
sens1:func "freq:rat 2"
read?
```

a. Record the reading for the channel one:channel two ratio.

2. Channel two: channel three ratio test:

Apply 3000 MHz @ 4 dBm ±1 dB to Channel 3
Send the following VX4101 command(s)/query:

```
sens2:func "freq rat 3"
read?
```

a. Record the reading for the channel two:channel three ratio.

AC Low Voltage Accuracy

Follow the procedures below to test AC voltage accuracy

1. 0.9 V_{PP} @ 100 kHz AC voltage accuracy test:

a. Apply 0.9 $V_{P\text{-}P}$ ± 1% @ 100 kHz to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
inp1:filt on
inp2:filt on
sens1:func "volt:ptp"
sens:volt:PTP:coun 10
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

e. Record the reading for channel two.

2. 0.36 $V_{P\text{-}P}$ @ 100 kHz AC voltage accuracy test:

a. Apply 0.36 $V_{P\text{-}P}$ ± 1% @ 100 kHz to channels one and two.

b. Send the following VX4101 command(s)/query:

```
sens1:func "volt:ptp"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

e. Record the reading for channel two.

DC Voltage Accuracy

Follow the procedures below to test DC voltage accuracy.

1. 0 VDC AC voltage accuracy test:

a. Apply 0 VDC ± 1 mV channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:ptp"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

e. Record the reading for channel two.

2. 0 VDC voltage accuracy test, 1 MΩ impedance test:

a. Apply 0 VDC ± 1 mV channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
inp1:imp 1e6
inp2:imp 1e6
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

AC High Voltage

Follow the procedures below to test AC high voltage frequency accuracy. Each step should be done in the order in which it occurs in the procedure.

CAUTION. *This step requires application of a dangerous voltage level. You could receive a dangerous electrical shock. Do not touch cabling connections while applying high voltage.*

Before You Begin. Replace the RF 50 Ω splitter with a tee.

1. 9 V_{P-P} @ 100 kHz test:

- **a.** Apply 9 V_{P-P} ± 1% @ 100 kHz to channels one and two.
- **b.** Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:ptp"
read?
```

- **c.** Record the reading for channel one.
- **d.** Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

- **e.** Record the reading for channel two.

2. 90 V_{P-P} @ 100 kHz test:

- **a.** Apply 90 V_{P-P} ± 1% @ 100 kHz to channels one and two.
- **b.** Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:ptp"
read?
```

- **c.** Record the reading for channel one.
- **d.** Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

- **e.** Record the reading for channel two.

3. 3.6 V_{P-P} @100 kHz test:

- **a.** Apply 3.6 V_{P-P} ± 1% @ 100 kHz to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:ptp"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

e. Record the reading for channel two.

4. 1.8 V_{P-P} @ 100 kHz test:

a. Apply 1.8 V_{P-P} ± 1% @ 100 kHz to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:ptp"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:ptp"
read?
```

DC High-Voltage Accuracy

Follow the procedures below to test DC high-voltage accuracy:

1. 200 VDC test:

a. Apply 200 VDC ± 1% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

2. –200 VDC test:

a. Apply –200 VDC ± 1% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

3. 18 VDC test:

a. Apply 18 VDC ± 1% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

4. –18 VDC test:

a. Apply –18 VDC ± 1% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

5. 1.8 VDC test:

a. Apply 1.8 VDC ± 1% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

6. –1.8 VDC test:

a. Apply –1.8 VDC ± 1% to channels one and two.

b. Send the following VX4101 command(s)/query to test channel one:

```
sens1:func "volt:dc"
read?
```

c. Record the reading for channel one.

d. Send the following VX4101 command(s)/query to test channel two:

```
sens2:func "volt:dc"
read?
```

e. Record the reading for channel two.

Arm Input Function

Follow the procedures below to test the arm input function:

1. Apply 0 VDC ± 0.1 V to the arm input

Send the following VX4101 commands/query

```
conf:volt:dc
arm:sour ctr_extarm
arm:slope pos
init
fetch:count?
```

2. Apply 3 VDC ± 0.1 V to the arm input

Send the following VX4101 commands/query

```
fetch:count?
```

3. Send the following VX4101 commands/query

```
arm:slope neg
init
fetch:count?
```

4. Apply 0 VDC ± 0.1 V to the arm input

Send the following VX4101 commands/query

```
fetch:count?
```

Appendix F: DMM Performance Verification Procedure

This procedure verifies the performance of the DMM portion of the VX4101. It is not necessary to complete the entire procedure if you are only interested in a specific performance area. The procedure is grouped according to specific sets of tests. The test sets are as follows:

- DC voltage
- AC voltage
- Resistance (4-wire and 2-wire)
- Current
- Common mode rejection

Required Skills

The following skills and knowledge are required to perform this procedure:

- Thorough knowledge of test instrument operation and proper measurement techniques
- Knowledge of VXIbus system components and command language programming
- Ability and facility to construct interconnections and fixtures as needed to perform the procedure
- Experience in safety precautions when handling high voltage

General Information and Conventions

Each step includes the required calibration standard and the required VX4101 commands and queries. The input specified is DC, AC, Ω, or I, and the values shown are in V, V_{RMS}, Ω, or A. Each VX4101 command/query will return five values.

Equipment Required

The same equipment is required for all DMM tests. The VX4101 DMM performance verification requires a calibration standard with basic system accuracies of:

Table F–1: Required Equipment

Description	Minimum Requirements	Examples
DC volt source	0.001% + 2.5 μV DC	Datron 4800 or Fluke 5700
AC volt source	0.075% + 45 μV_{RMS}	
Resistance	0.025% + 0.004 Ω	
Current	0.025% + 15 μA DC	

***NOTE**. Many ranges, such as low range DC and AC, and high range resistance, require less accuracy. Each test step shows exact accuracy requirements for that test.*

Considerations for Automated Procedures

Should this procedure be automated, allow time between changing the calibration standard input and sending the command/query to the VX4101. This allows the calibration standard output to stabilize.

Using Calibration Standards

If you use a Fluke 5700 series calibration standard, the 5700 will not release control back to the system controller until its output has settled. This simplifies the delay task.

However, certain modes and ranges of the VX4101 require additional settling time for the cabling or VX4101 input circuitry. For the resistance ranges or 300 kΩ and greater, additional time is required for the low level source currents (1 nA and 100 pA) to charge the cabling capacitance. For all AC ranges (AC and DC coupled) the capability of measuring AC frequencies down to 25 Hz translates into long settling times.

If you use a Datron 8200 series calibration standard, the system controller is permitted to send commands to other instruments before the output is stabilized. This means that for all modes and ranges, you need to program a delay to allow stabilization to occur. These delays typically need to be determined experimentally. However, once they are determined, the Datron calibrator will typically reduce test time by at least 30%. The two calibrators require a tradeoff between ease of program development and test times.

Semi-Automated Procedures

Another approach to reducing test time is to semi-automate the test by automatically sending all command and queries to the VX4101 and comparing the maximum and minimum results of the last four of five readings with the required maximum and minimum values.

Record them electronically. The calibration standard is operated manually using prompts from the semi-automated program. This usually solves the delay requirement above. The time between manually setting the calibrator and responding to the program prompt provides adequate delay for stabilization, and also provides a procedure that may be used with any calibration standard.

Expected times to perform the full DMM Performance Verification procedure are:

Table F–2: Test Times

Test Type	Time Required
Fully manual	6 to 10 hours
Semi-automated	45 to 75 minutes
Fully-automated	15 to 25 minutes

Prerequisites

- The VX4101 module covers must be in place and the module installed in an approved VXIbus mainframe as described in *Operating Basics*
- The VX4101 has passed the power-on self test
- The VX4101 has been operating for a warm-up period of at least 20 minutes in an ambient environment as specified in *Getting Started*

About the Test Records

The Test Records are divided into sets of 24-hour, 90-day, and one-year records for the following test types:

- DC voltage measurement
- AC voltage measurement
- Resistance (4-wire and 2-wire) measurement

- Current measurement
- Common mode rejection

Using the Test Record

Each type of test within each set contains the following:

- A description of the type of performance test, for example, a DC 300 mV Range Null Check
- Columns with maximum and minimum acceptable readings
- Columns to record the readings you receive from each performance test type
- An additional column to note whether the instrument passed or failed the test

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 3 V Range Null Check with Minimum Aperture	0.00015			–0.00015	

Figure F–1: Sample Test Result Record

Running the Tests. When you run the tests, each VX4101 command/query instruction will return five values. Record the results in the result record as follows:

- Record the maximum of the last four values in the Maximum Measured column
- Record the minimum of the last four values in the Minimum Measured column
- Record pass/fail status in the Pass/Fail column

If both the maximum and minimum recorded values meet the applicable maximum/minimum requirement, the test passes. If the calibration standard does not supply the exact value specified, for example, 10.0001 K instead of 10 K, subtract .0001K from the recorded values as compensation.

Before You Begin

Before you begin, do the following:

- Photocopy the appropriate set of Test Records to record the performance verification results for your module
- Run the self-test procedure described below

Self-Test Procedure

The self-test is performed by sending a command and query to the the VX4101 module. The self-test takes just under ten seconds to execute. The system controller used to send commands and receive query responses should have a timeout setting of a minimum of ten seconds for the self-test as well as all DMM performance verification procedures.

To initiate the self test, send the following command and query:

```
inst:sel dmm
test:all?
```

The response should indicate that the self-test has passed. If the self-test fails, send the following query for more detailed information about the failure:

```
syst:err?
```

24-Hour Test Records

This set of test tecords contains the minimum and maximum specifications for a 24-hour test. The test sets are as follows:

- DC voltage measurement
- AC voltage measurement
- Resistance measurement
- Current measurement
- Common mode rejection

Table F–3: 24-Hour DC Voltage Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 3 V Range Null Check with Minimum Aperture	–0.00015			0.00015	
DC 30 V Range Null Check with Minimum Aperture	–0.0015			0.0015	
DC 30 mV Range Null Check	–9E–006			9E–006	
DC 30 mV Range Null Check with 1 Power Line Cycle Aperture	–1.2E–005			1.2E–005	
DC 300 mV Range Null Check	–1.5E–005			1.5E–005	
DC 300 mV Range Null Check with 1 Power Line Cycle Aperture	–2E–005			2E–005	
DC 3 V Range Null Check	–0.0001			0.0001	
DC 30 V Range Null Check	–0.0008			0.0008	
DC 300 V Range Null Check	–0.009			0.009	
DC 30 mV 30% Range Check	0.00899046			0.00900954	
DC 30 mV 90% Range Check	0.0269894			0.0270106	
DC 30 mV Range Negative Voltage Check	–0.0270106			–0.0269894	
DC 300 mV 30% Range Check	0.0899805			0.0900195	
DC 300 mV 90% Range Check	0.269972			0.270029	

Table F–3: 24-Hour DC Voltage Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 300 mV Negative Voltage Check	–0.270034			–0.269967	
DC 3 V 30% Range Check	0.899884			0.900116	
DC 3 V 90% Range Check	2.69981			2.70019	
DC 3 V Negative Voltage Check	–2.70021			–2.69979	
DC 30 V 90% Range Check	26.9977			27.0023	
DC 30 V Negative Voltage Check	–27.0023			–26.9977	
DC 300 V 90% Range Check	269.972			270.028	
DC 300 mV High Impedance 90% Range Check	0.269972			0.270029	

Table F–4: 24-Hour AC Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 30 mV DC-Coupled 15% Range Check	0.0043065			0.0046935	
AC 30 mV DC-Coupled 90% Range Check	0.026739			0.027261	
AC 30 mV DC-Coupled Negative Voltage Check	0.026739			0.027261	
AC 300 mV DC-Coupled 10% Range Check	0.02961			0.03039	
AC 300 mV DC-Coupled 90% Range Check	0.26889			0.27111	
AC 3 V DC–Coupled 10% Range Minimum Aperture Check	0.2901			0.3099	
AC 3 V DC-Coupled 50% Range Check	1.4865			1.5135	
AC 30 V DC-Coupled 10% Range Check	2.961			3.039	
AC 30 V DC-Coupled 90% Range Check	26.889			27.111	
AC 300 V DC-Coupled 10% Range Check	29.55			30.45	

Table F–4: 24-Hour AC Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 30 mV DC-Coupled 10% Range Low Frequency Check	0.00428625			0.00471375	
AC 3 V DC-Coupled 10% Range Low-Frequency Check	0.28875			0.31125	
AC 300 V DC-Coupled 10% Range Low Frequency Check	29.445			30.555	
AC 300 mV DC-Coupled 90% Range 100 Hz Check	0.26889			0.27111	
AC 30V DC-Coupled 90% Range 100 Hz Check	26.889			27.111	
AC 3 V DC-Coupled 10% Range 10 KHz Check	0.2901			0.3099	
AC 300 V DC-Coupled 10% Range 10 KHz Check	29.55			30.45	
AC 30 mV DC-Coupled 15% Range High Frequency Check	0.00406			0.00494	
AC 300 mV DC-Coupled 90% Range High Frequency Check	0.2641			0.2759	
AC 30 V DC-Coupled 90% Range High Frequency Check	26.187			27.813	
AC 300 mV AC-Coupled 10% Range Low Frequency Check	0.02925			0.03075	
AC 3 V AC-Coupled 90% Range Low Frequency Check	2.6745			2.7255	
AC 30 V AC-Coupled 10% Range Low Frequency Check	2.945			3.055	
AC 30 mV AC-Coupled 15% Range Low Frequency Check	0.004132			0.004868	
AC 300 V AC-Coupled 90% Range Low Frequency Check	267.98			272.02	
AC 300 V AC-Coupled 90% Range High Frequency Check	267.98			272.02	
AC 30 mV AC-Coupled 15% Range High Frequency Check	0.0041095			0.0048905	

Table F–4: 24-Hour AC Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 300 mV AC-Coupled 10% Range High Frequency Check	0.02865			0.03135	
AC 3 V AC-Coupled 90% Range High Frequency Check	2.6205			2.7795	
AC 30 V AC-Coupled 10% Range High Frequency Check	2.873			3.127	

Table F–5: 24-Hour Resistance Measurement Verification Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
30 Ω 4-Wire Null Check	–0.009			0.009	
30 Ω 4-Wire 33% Range Check	9.9893			10.0107	
300 Ω 4-Wire Null Check	–0.015			0.015	
300 Ω 4-Wire 33% Range Check	99.975			100.025	
3 KΩ 4-Wire Null Check	–0.08			0.08	
3 KΩ 4-Wire 33% Range Check	999.82			1000.18	
30 KΩ 4-Wire 33% Range Check	9997			10003	
300 KΩ 4-Wire 33% Range Check	99982			100018	
3 MΩ 4-Wire 33% Range Check	999500			1.0005E+006	
30 MΩ 4-Wire Null Check	–800			800	
30 MΩ 4-Wire 33% Range Check	9.9692E+006			1.00308E+007	
300 MΩ 4-Wire 33% Range Check	9.79992E+007			1.02001E+008	
3 KΩ 2-Wire .3% Range, 1 Power Line Cycle	9.8985			10.1015	

Table F–5: 24-Hour Resistance Measurement Verification Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
300 KΩ 2-Wire .3% Range Check	999.75			1000.25	
300 KΩ 2-Wire 33% Range Check	99975			100025	

Table F–6: 24-Hour Current Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
Current 150 mA Range Null Check	–5E–005			5E–005	
Current 150 mA 10% Range Check	0.014935			0.015065	
Current 150 mA 93% Range Check	0.13981			0.14019	
Current 150 mA Range Negative Input Check	–0.14019			–0.13981	
Current 1 A Range Null Check	–0.0008			0.0008	
Current 1 A 90% Range Check	0.89785			0.90215	

Table F–7: 24-Hour DMM Common Mode Rejection Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC Mode-DC CMR	N/A	N/A		± 890 μV	
DC Mode-AC CMR	N/A	N/A		± 566 μV	
AC Mode-AC CMR	N/A	N/A		± 35.56 μV	

90-Day Verification Test Records

This set of test records contains the minimum and maximum specifications for a 90-day test. The test sets are as follows:

- DC voltage measurement
- AC voltage measurement
- Resistance measurement
- Current measurement
- Common mode rejection

Table F–8: 90-Day DC Voltage Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 3 V Range Null Check with Minimum Aperture	–0.00015			0.00015	
DC 30 V Range Null Check with Minimum Aperture	–0.0015			0.0015	
DC 30 mV Range Null Check	–9E–006			9E–006	
DC 30 mV Range Null Check with 1 Power Line Cycle Aperture	–1.2E–005			1.2E–005	
DC 300 mV Range Null Check	–1.5E–005			1.5E–005	
DC 300 mV Range Null Check with 1 Power Line Cycle Aperture	–2E–005			2E–005	
DC 3 V Range Null Check	–0.0001			0.0001	
DC 30 V Range Null Check	–0.0008			0.0008	
DC 300 V Range Null Check	–0.009			0.009	
DC 30 mV 30% Range Check	0.00898992			0.00901008	
DC 30 mV 90% Range Check	0.0269878			0.0270122	
DC 30 mV Range Negative Voltage Check	–0.0270122			–0.0269878	
DC 300 mV 30% Range Check	0.089976			0.090024	

Table F–8: 90-Day DC Voltage Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 300 mV 90% Range Check	0.269958			0.270042	
DC 300 mV Negative Voltage Check	–0.270047			–0.269953	
DC 3 V 30% Range Check	0.899848			0.900152	
DC 3 V 90% Range Check	2.6997			2.7003	
DC 3 V Negative Voltage Check	–2.70032			–2.69968	
DC 30 V 90% Range Check	26.9962			27.0038	
DC 30 V Negative Voltage Check	–27.0038			–26.9962	
DC 300 V 90% Range Check	269.953			270.047	
DC 300 mV High Impedance 90% Range Check	0.269958			0.270042	

Table F–9: 90-Day AC Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 30 mV DC-Coupled 15% Range Check	0.0043029			0.0046971	
AC 30 mV DC-Coupled 90% Range Check	0.0267174			0.0272826	
AC 30 mV DC-Coupled Negative Voltage Check	–0.0272826			–0.0267174	
AC 300 mV DC-Coupled 10% Range Check	0.029586			0.030414	
AC 300 mV DC-Coupled 90% Range Check	0.268674			0.271326	
AC 3 V DC-Coupled 10% Range Minimum Aperture Check	0.28986			0.31014	
AC 3 V DC-Coupled 50% Range Check	1.4853			1.5147	
AC 30 V DC-Coupled 10% Range Check	2.9586			3.0414	

Table F–9: 90-Day AC Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 30 V DC-Coupled 90% Range Check	26.8674			27.1326	
AC 300 V DC-Coupled 10% Range Check	29.526			30.474	
AC 30 mV DC-Coupled 10% Range Low Frequency Check	0.00428265			0.00471735	
AC 3 V DC-Coupled 10% Range Low-Frequency Check	0.28851			0.31149	
AC 300 V DC-Coupled 10% Range Low Frequency Check	29.421			30.579	
AC 300 mV DC-Coupled 90% Range 100 Hz Check	0.268674			0.271326	
AC 30V DC-Coupled 90% Range 100 Hz Check	26.8674			27.1326	
AC 3 V DC-Coupled 10% Range 10 KHz Check	0.28986			0.31014	
AC 300 V DC-Coupled 10% Range 10 KHz Check	29.526			30.474	
AC 30 mV DC-Coupled 15% Range High Frequen-cy Check	0.0040564			0.0049436	
AC 300 mV DC-Coupled 90% Range High Frequen-cy Check	0.263884			0.276116	
AC 30 V DC-Coupled 90% Range High Frequency Check	26.1654			27.8346	
AC 300 mV AC-Coupled 10% Range Low Frequency Check	0.029226			0.030774	
AC 3 V AC-Coupled 90% Range Low Frequency Check	2.67234			2.72766	
AC 30 V AC-Coupled 10% Range Low Frequency Check	2.9426			3.0574	
AC 30 mV AC-Coupled 15% Range Low Frequency Check	0.0041284			0.0048716	

Table F–9: 90-Day AC Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 300 V AC-Coupled 90% Range Low Frequency Check	267.764			272.236	
AC 300 V AC-Coupled 90% Range High Frequency Check	267.764			272.236	
AC 30 mV AC-Coupled 15% Range High Frequency Check	0.0041059			0.0048941	
AC 300 mV AC-Coupled 10% Range High Frequency Check	0.028626			0.031374	
AC 3 V AC-Coupled 90% Range High Frequency Check	2.61834			2.78166	
AC 30 V AC-Coupled 10% Range High Frequency Check	2.8706			3.1294	

Table F–10: 90-Day Resistance Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
30 Ω 4-Wire Null Check	–0.009			0.009	
30 Ω 4-Wire 33% Range Check	9.9887			10.0113	
300 Ω 4-Wire Null Check	–0.015			0.015	
300 Ω 4-Wire 33% Range Check	99.97			100.03	
3 KΩ 4-Wire Null Check	–0.08			0.08	
3 KΩ 4-Wire 33% Range Check	999.77			1000.23	
30 KΩ 4-Wire 33% Range Check	9996.5			10003.5	
300 KΩ 4-Wire 33% Range Check	99977			100023	
3 MΩ 4-Wire 33% Range Check	999300			1.0007E+006	
30 MΩ 4-Wire Null Check	–800			800	

Table F–10: 90-Day Resistance Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
30 MΩ 4-Wire 33% Range Check	9.9592E+006			1.00408E+007	
300 MΩ 4-Wire 33% Range Check	9.77992E+007			1.02201E+008	
3 KΩ 2-Wire 0.3% Range, 1 Power Line Cycle	9.898			10.102	
300 KΩ 2-Wire 0.3% Range Check	999.7			1000.3	
300 KΩ 2-Wire 33% Range Check	99970			100030	

Table F–11: 90-Day Current Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
Current 150 mA Range Null Check	–5E–005			5E–005	
Current 150 mA10% Range Check	0.014932			0.015068	
Current 150 mA 93% Range Check	0.139782			0.140218	
Current 150 mA Range Negative Input Check	–0.140218			–0.139782	
Current 1 A Range Null Check	–0.0008			0.0008	
Current 1A 90% Range Check	–0.0008			0.0008	

Table F–12: 90-Day DMM Common Mode Rejection Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC Mode-DC CMR	N/A	N/A		± 890 μV	
DC Mode-AC CMR	N/A	N/A		± 566 μV	
AC Mode-AC CMR	N/A	N/A		± 35.56 μV	

One Year Verification Test Record

This set of test records contains the minimum and maximum specifications for a one-year test. The test sets are as follows:

- DC voltage measurement
- AC voltage measurement
- Resistance measurement
- Current measurement
- Common mode rejection

Table F–13: One-Year DC Voltage Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 3 V Range Null Check with Minimum Aperture	–0.00015			0.00015	
DC 30 V Range Null Check with Minimum Aperture	–0.0015			0.0015	
DC 30 mV Range Null Check	–9E–006			9E–006	
DC 30 mV Range Null Check with 1 Power Line Cycle Aperture	-1.2E–005			1.2E–005	
DC 300 mV Range Null Check	–1.5E–005			1.5E–005	
DC 300 mV Range Null Check with 1 Power Line Cycle Aperture	–2E–005			2E–005	
DC 3 V Range Null Check	–0.0001			0.0001	
DC 30 V Range Null Check	–0.0008			0.0008	
DC 300 V Range Null Check	–0.009			0.009	
DC 30 mV 30% Range Check	0.00898938			0.00901062	
DC 30 mV 90% Range Check	0.0269861			0.0270139	
DC 30 mV Range Negative Voltage Check	–0.0270139			-0.0269861	
DC 300 mV 30% Range Check	0.0899715			0.0900285	

Table F–13: One-Year DC Voltage Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC 300 mV 90% Range Check	0.269945			0.270056	
DC 300 mV Negative Voltage Check	–0.270061			–0.26994	
DC 3 V 30% Range Check	0.899812			0.900188	
DC 3 V 90% Range Check	2.6996			2.7004	
DC 3 V Negative Voltage Check	–2.70042			–2.69958	
DC 30 V 90% Range Check	26.9949			27.0051	
DC 30 V Negative Voltage Check	–27.0051			–26.9949	
DC 300 V 90% Range Check	269.937			270.063	
DC 300 mV High Impedance 90% Range Check	0.269945			0.270056	

Table F–14: One-Year AC Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 30 mV DC-Coupled 15% Range Check	0.00429975			0.00470025	
AC 30 mV DC-Coupled 90% Range Check	0.0266985			0.0273015	
AC 30 mV DC-Coupled Negative Voltage Check	–0.0273015			–0.0266985	
AC 300 mV DC-Coupled 10% Range Check	0.029565			0.030435	
AC 300 mV DC-Coupled 90% Range Check	0.268485			0.271515	
AC 3 V DC-Coupled 10% Range Minimum Aperture Check	0.28965			0.31035	
AC 3 V DC-Coupled 50% Range Check	1.48425			1.51575	
AC 30 V DC-Coupled 10% Range Check	2.9565			3.0435	

Table F–14: One-Year AC Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 30 V DC-Coupled 90% Range Check	26.8485			27.1515	
AC 300 V DC-Coupled 10% Range Check	29.505			30.495	
AC 30 mV DC-Coupled 15% Range Low Frequency Check	0.0042795			0.0047205	
AC 3 V DC-Coupled 10% Range Low Frequency Check	0.2883			0.3117	
AC 300 V DC-Coupled 10% Range Low Frequency Check	29.4			30.6	
AC 300 mV DC-Coupled 90% Range 100 Hz Check	0.268485			0.271515	
AC 30V DC-Coupled 90% Range 100 Hz Check	26.8485			27.1515	
AC 3 V DC-Coupled 10% Range 10 KHz Check	0.28965			0.31035	
AC 300 V DC-Coupled 10% Range 10 KHz Check	29.505			30.495	
AC 30 mV DC-Coupled 15% Range High Frequen-cy Check	0.00403975			0.00496025	
AC 300 mV DC-Coupled 90% Range High Frequen-cy Check	0.262885			0.277115	
AC 30 V DC-Coupled 90% Range High Frequency Check	26.1465			27.8535	
AC 300 mV AC-Coupled 10% Range Low Frequency Check	0.029205			0.030795	
AC 3 V AC-Coupled 90% Range Low Frequency Check	2.67045			2.72955	
AC 30 V AC-Coupled 10% Range Low Frequency Check	2.9405			3.0595	
AC 30 mV AC-Coupled 15% Range Low Frequency Check	0.00412525			0.00487475	

Table F–14: One-Year AC Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
AC 300 V AC-Coupled 90% Range Low Frequency Check	267.575			272.425	
AC 300 V AC-Coupled 90% Range High Frequency Check	267.575			272.425	
AC 30 mV AC-Coupled 15% Range High Frequency Check	0.00410275			0.00489725	
AC 300 mV AC-Coupled 10% Range High Frequency Check	0.028515			0.031485	
AC 3V AC-Coupled 90% Range High Frequency Check	2.61645			2.78355	
AC 30 V AC-Coupled 10% Range High Frequency Check	2.8685			3.1315	

Table F–15: One-Year Resistance Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
30 Ω 4-Wire Null Check	–0.009			0.009	
30 Ω 4-Wire 33% Range Check	9.9873			10.0127	
300 Ω 4-Wire Null Check	–0.015			0.015	
300 Ω 4-Wire 33% Range Check	99.965			100.035	
3 KΩ 4-Wire Null Check	–0.08			0.08	
3 KΩ 4-Wire 33% Range Check	999.72			1000.28	
30 KΩ 4-Wire 33% Range Check	9996			10004	
300 KΩ 4-Wire 33% Range Check	99972			100028	
3 MΩ 4-Wire 33% Range Check	999100			1.0009E+006	
30 MΩ 4-Wire Null Check	–800			800	

Table F–15: One-Year Resistance Measurement Results (Cont.)

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
30 MΩ 4-Wire 33% Range Check	9.9492E+006			1.00508E+007	
300 MΩ 4-Wire 33% Range Check	9.74992E+007			1.02501E+008	
3 KΩ 2-Wire 3% Range, 1 Power Line Cycle	9.8975			10.1025	
300 KΩ 2-Wire 0.3% Range Check	999.65			1000.35	
300 KΩ 2-Wire 33% Range Check	99965			100035	

Table F–16: One-Year Current Measurement Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
Current 150 mA Range Null Check	–5E–005			5E–005	
Current 150 mA 10% Range Check	0.0149275			0.0150725	
Current 150 mA 93% Range Check	0.13974			0.14026	
Current 150 mA Range Negative Input Check	–0.14026			–0.13974	
Current 1 A Range Null Check	–0.0008			0.0008	
Current 1A 90% Range Check	0.89758			0.90242	

Table F–17: One-Year DMM Common Mode Rejection Results

Test Description	Minimum	Minimum Measured	Maximum Measured	Maximum	Pass/ Fail
DC Mode-DC CMR	N/A	N/A		± 890 µV	
DC Mode-AC CMR	N/A	N/A		± 566 µV	
AC Mode-AC CMR	N/A	N/A		± 35.56 µV	

Test Procedures

The following sets of tests apply to either 24-hour, 90-day, or one-year performance testing. The test sets are:

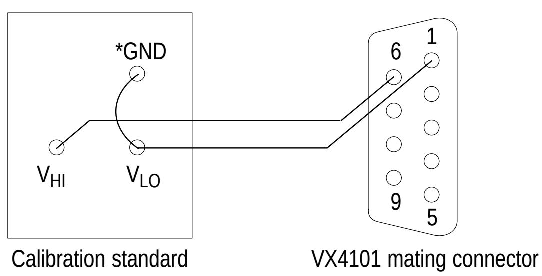
- DC voltage measurement
- AC voltage measurement
- Resistance measurement
- Current measurement
- Common mode rejection

Each set of tests contain specific tests, for example a DC 300 mV Range Null Check. Record the results of each test in the Maximum Measured and Minimum Measured columns in the appropriate Test Results table.

DC Voltage Measurement

Perform the following set of tests of DC voltage measurements for the DMM. Refer to the figure below for required cable connections.

What You Should Know About. Each test in the set depends on the proper operation of the previous test. Therefore, you must follow the order of tests as presented within each set.



***NOTE**. The V_{LO} to GND connection eliminates power line common mode from contributing to measurement error for minimum aperture measurement.*

---

1. DC 3 V range null check with minimum aperture:

   a. Apply 0 VDC
   Send the following VX4101 command(s)/query:

```
cal:zero:auto on
conf:arr:dc 5,3,min
inp:imp 10e6
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

2. DC 30 V range null check with minimum aperture:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,30,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

3. DC 30 mV range null check:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,.03
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

4. DC 30 mV range null check with 1 power line cycle aperture:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
sens:volt:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

5. DC 300 mV range null check:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,.3
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

6. DC 300 mV range null check with 1 power line cycle aperture:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
sens:volt:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

7. DC 3 V range null check:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,3
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

8. DC 30 V range null check:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,30
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

9. DC 300 V range null check:

a. Apply 0 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,300
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

10. DC 30 mV 30% range check:

a. Apply 0.009 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,.03
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

11. DC 30 mV 90% range check:

a. Apply 0.027 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

12. DC 30 mV range negative voltage check:

a. Apply –0.027 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

13. DC 300 mV 30% range check:

a. Apply 0.09 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,.3
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

14. DC 300 mV 90% range check:

a. Apply 0.27 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

15. DC 300 mV negative voltage check:

a. Apply –0.27 VDC
Send the following VX4101 command(s)/query:

```
sens:volt:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

16. DC 3 V 30% range check:

a. Apply 0.9 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,3
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

17. DC 3 V 90% range check:

a. Apply 2.7 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

18. DC 3 V Negative voltage check:

a. Apply –2.7 VDC
Send the following VX4101 command(s)/query:

```
sens:volt:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

19. DC 30 V 90% range check:

a. Apply 27 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,30
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

20. DC 30 V negative voltage check:

a. Apply –27 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

21. DC 300 V 90% range check:

a. Send the the following command before applying high voltage:

```
conf:dc 300
init
```

WARNING. *Step 21 requires application of a dangerous voltage level. You could receive a dangerous electric shock. Do not touch cabling connections while applying high voltage.*

b. Apply 270 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:dc? 5,300
```

c. Record the maximum and minimum results of last four of five readings in the appropriate test record.

22. DC 300 mV high impedance 90% range check:

a. Apply 0.27 VDC
Send the following VX4101 command(s)/query:

```
conf:arr:dc 5,.3
inp:imp 10e9
read?
```

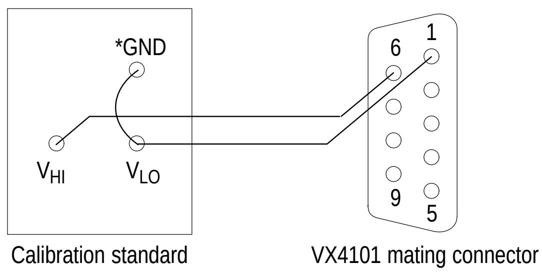
b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

AC Voltage Measurement

Perform the following set of tests of DC voltage measurements for the DMM. Refer to the figure below for required cable connections.

What You Should Know About. Each test in the set depend on the proper operation of the previous test. Therefore, you must follow the order of tests as presented within each test set.

***NOTE**. The connection is the same for DC and AC voltage verification. If you perform this test immediately following the DC Voltage Verification Procedures, it is not necessary to change the cabling.*



***NOTE**. The V_{LO} to GND connection, shown above, eliminates power line common mode from contributing to measurement error for minimum aperture measurement.*

1. AC 30 mV DC-coupled 15% range check:

   a. Apply 0.0045 VDC
   Send the following VX4101 command(s)/query:

   ```
   meas:arr:acdc? 5,.03,min
   ```

   b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

2. AC 30 mV DC-coupled 90% range check:

   a. Apply 0.027 VDC
   Send the following VX4101 command(s)/query:

   ```
   read?
   ```

   b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

3. AC 30 mV DC-coupled negative voltage check:

a. Apply –0.027 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

4. AC 300 mV DC-coupled 10% range check:

a. Apply 0.03 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,.3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

5. AC 300 mV DC-coupled 90% range check:

a. Apply 0.27 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

6. AC 3 V DC-coupled 10% range minimum aperture check:

a. Apply 0.3 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

7. AC 3 V DC-coupled 50% range check:

a. Apply 1.5 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

8. AC 30 V DC-coupled 10% range check:

a. Apply 3 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,30,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

9. AC 30 V DC-coupled 90% range check:

a. Apply 27 VDC
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

10. AC 300 V DC-coupled 10% range check:

a. Apply 30 VDC
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,300,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

11. AC 30 mV DC-coupled 10% range low frequency check:

a. Apply 25 Hz 0.0045 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,.03,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

12. AC 3 V DC-coupled 10% range low frequency check:

a. Apply 25 Hz 0.3 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

13. AC 300 V DC-coupled 10% range low frequency check:

a. Apply 25 Hz 30 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,300,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

14. AC 300 mV DC-coupled 90% range 100 Hz check:

a. Apply 100 Hz 0.27 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,.3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

15. AC 30V DC-coupled 90% range 100 Hz check:

a. Apply 100 Hz 27 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,30,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

16. AC 3 V DC-coupled 10% range 10 KHz check:

a. Apply 10 KHz 0.3 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

17. AC 300 V DC-coupled 10% range 10 KHz check:

a. Apply 10 KHz 30 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,300,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

18. AC 30 mV DC-Coupled 15% high frequency check:

a. Apply 50 kHz 0.0045 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,.03,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

19. AC 300 mV DC-coupled 90% range high frequency check:

a. Apply 50 kHz 0.27 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,.3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

20. AC 30 V DC-coupled 90% range high frequency check:

a. Apply 50 KHz 27 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:acdc? 5,30,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

21. AC 300 mV AC-coupled 10% range low frequency check:

a. Apply 45 Hz 0.03 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,.3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

22. AC 3 V AC-coupled 90% range low frequency check:

a. Apply 45 Hz 2.7 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

23. AC 30 V AC-coupled 10% range low frequency check:

a. Apply 45 Hz 3 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,30,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

24. AC 30 mV AC-coupled 15% range low frequency check:

a. Apply 200 Hz 0.0045 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,.03,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

25. AC 300 V AC-coupled 90% range low frequency check:

a. Send the the following commands before applying high voltage:

```
conf:ac 300
init
```

WARNING. *The following steps require application of a dangerous voltage level. You could receive a dangerous electric shock. Do not touch cabling connections while applying high voltage.*

b. Apply 100 Hz 270 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,300,min
```

c. Record the maximum and minimum results of last four of five readings in the appropriate test record.

26. AC 300 V AC-coupled 90% range high frequency check:

a. Apply 10 kHz 270 V_{RMS}
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

27. AC 30 mV AC-coupled 15% range high frequency check:

a. Apply 20 kHz 0.0045 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,.03,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

28. AC 300 mV AC-coupled 10% range high frequency check:

a. Apply 50 kHz 0.03 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,.3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

29. AC 3 V AC-coupled 90% range high frequency check:

a. Apply 50 kHz 2.7 V_{RMS}
Send the following VX4101 command(s)/query:

```
meas:arr:ac? 5,3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

30. AC 30 V AC-coupled 10% range high frequency check:

a. Apply 50 kHz 3 V_{RMS}
Send the following VX4101 command(s)/query:

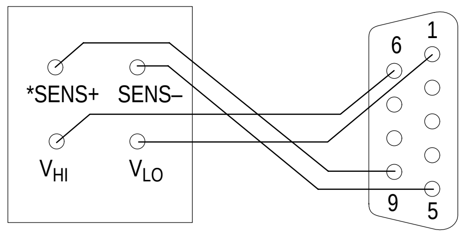
```
meas:arr:ac? 5,30,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

Resistance (4-Wire and 2-Wire) Measurement

Perform the following set of tests to measure resistance. Refer to the figure for required cable connections.

What You Should Know About. Each test in the set depends on the proper operation of the previous test. Therefore, you must follow the order of tests as presented within each test set.



***NOTE**. For 2-wire measurements, pin 9 and 5 of the VX4101 are not connected internally. You can leave the SENS+ and SENS– cables connected or disconnected, as desired, for 2-wire measurements.*

---

1. 30 Ω 4-wire null check:

a. Apply 0 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,30
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

2. 30 Ω 4-wire 33% range check:

a. Apply 10 Ω
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

3. 300 Ω 4-wire null check:

a. Apply 0 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,300
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

4. 300 Ω 4-wire 33% range check:

a. Apply 100 Ω
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

5. 3 KΩ 4-wire null check:

a. Apply 0 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,3e3
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

6. 3 KΩ 4-wire 33% range check:

a. Apply 1000 Ω
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

7. 30 KΩ 4-wire 33% range check:

a. Apply 10000 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,30e3,min
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

8. 300 KΩ 4-wire 33% range check:

a. Apply 100000 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,300e3
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

9. 3 MΩ 4-wire 33% range check:

a. Apply 1E+006 Ω
Send the following VX4101 command(s)/query:

```
conf:arr:fres 5,3e6
sens:fres:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

10. 30 MΩ 4-wire null check:

a. Apply 0 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,30e6
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

11. 30 MΩ 4-Wire 33% range check:

a. Apply 1E+007 Ω
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

12. 300 MΩ 4-wire 33% range check:

a. Apply 1E+008 Ω
Send the following VX4101 command(s)/query:

```
meas:arr:fres? 5,300e6
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

13. 3 KΩ 2-wire 0.3% range, one powerline cycle:

***NOTE**. This step assumes the same cabling and internal calibration standard resistance as when the Two-wire Ω null was calibrated. If the original cabling and equipment configuration is unknown, you may skip this test step.*

---

a. Apply 10 Ω
Send the following VX4101 command(s)/query:

```
conf:arr:res 5,3000
sens:res:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

14. 300 KΩ 2-Wire 0.3% range check:

a. Apply 1000 Ω
Send the following VX4101 command(s)/query:

```
conf:arr:res 5,300e3
sens:res:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

15. 300 KΩ 2-Wire 33% range check:

a. Apply 100000 Ω
Send the following VX4101 command(s)/query:

```
read?
```

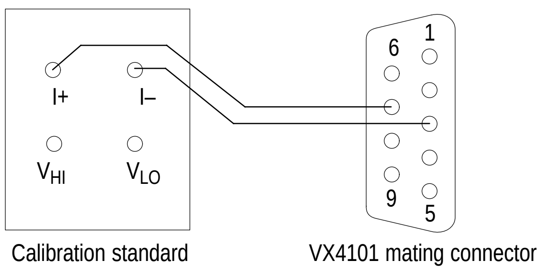
b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

Current Measurement

Perform the following steps to measure current for the DMM. Refer to the figure below for required cable connections.

What You Should Know About. Each test in the set depends on the proper operation of the previous test. Therefore, you must follow the order of tests as presented within each test set.

About the Cable Connections. See the following figure for required cable connections. On both the Datron and Fluke calibration standard the V_{HI} and V_{LO} cable connections are disconnected internally. When you select current mode, you can leave theV_{HI} and V_{LO} cable connections to pin 1 and 6 of the VX4101 unconnected, if desired.



***CAUTION.** Do not apply voltage greater than ± 0.4 V to the current inputs. Application of a voltage greater than ± 0.4 V to the current inputs of the VX4101 module will blow a fuse, requiring substantial disassembly of the module to replace it.*

16. Current 150 mA range null check:

a. Apply 0 A
Send the following VX4101 command(s)/query:

```
conf:arr:curr 5,.15
sens:curr:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

17. Current 150 mA 10% range check:

a. Apply 0.015 A
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

18. Current 150 mA 93% range check:

a. Apply 0.14 A
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

19. Current 150 mA range negative input check:

a. Apply –0.14 A
Send the following VX4101 command(s)/query:

```
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

20. Current 1A range null check:

a. Apply 0 A
Send the following VX4101 command(s)/query:

```
conf:arr:curr 5,1
sens:curr:nplc 1
read?
```

b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

21. Current 1A 90% range check:

a. Apply 0.9 A
Send the following VX4101 command(s)/query:

```
read?
```

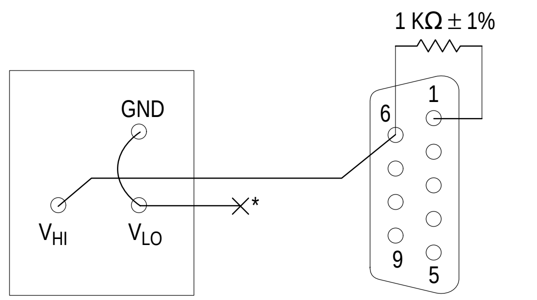
b. Record the maximum and minimum results of last four of five readings in the appropriate test record.

Common Mode Rejection

Perform the following steps to test common mode verification. Refer to the figure below for cable connections.

What You Should Know About. Each test in the set depends on the proper operation of the previous test. Therefore, you must follow the order of tests as presented within each test set.

CAUTION. *Make sure that the Calibration Standard V_{LO} terminal is disconnected from pin 1 (VIN–) of the VX4101 during the common mode test. Failure to do so will destroy the 1 KΩ resistor (or current limit the calibration standard).*



WARNING. *All steps in testing common mode rejection requires application of dangerous voltage levels. To prevent electric shocks, do not touch cabling connections while applying high voltage.*

WARNING. *Internal shielding voltages are at an unsafe level during this test. To prevent electric shocks, do not perform these test steps with the module covers removed.*

1. DC common mode rejection check:

For this step and the next two steps attach a 1 kΩ ± 1% resistor between the V+ pin and the V– pin

Perform the following steps for the DC mode DC common mode rejection test:

a. Apply 0 VDC between the V+ pin and the VX4101 enclosure. The DE9 connector case or the grille on the edge of the module are both suitable connections.

b. Send the following VX4101 command and query:

```
conf:arr:dc? 5,.3
read?
```

c. Record the average of the 5 readings.

d. Increase the voltage between the V+ pin and the VX4101 enclosure from 0 to 199 VDC.

e. Send the following query:

```
read?
```

The difference between any of the five readings and the average recorded above should be within ± 890 μV.

f. Record the maximum absolute change of the five readings in the test record.

2. DC mode AC effective common mode rejection test:

a. With the 1 KΩ resistor still attached as in previous test, apply 20 V_{RMS} 60 Hz between the V+ pin and the VX4101 enclosure.

b. Send the following VX4101 query:

```
read?
```

c. Record the average of the 5 readings.

d. Increase the voltage between the V+ pin and the VX4101 enclosure from 20 to 199 V_{RMS} 60 Hz.

e. Send the following query:

```
read?
```

The difference between any of the five readings and the average recorded above should be within ± 566 μV.

f. Record the maximum absolute change of the five readings in the test record.

3. AC mode AC common mode rejection test:

a. With 1 KΩ resistor still attached as in previous test, apply 30 V_{RMS} 60 Hz between the V+ pin and the VX4101 enclosure.

b. Send the following VX4101 command and query:

```
conf:arr:acdc 5,.3
read?
```

c. Record the average of the last 4 readings.

d. Increase the voltage between the V+ pin and the VX4101 enclosure from 30 to 70 V_{RMS} 60 Hz.

e. Send the following query:

```
read?
```

The difference between any of the five readings and the average recorded above should be within ± 35.65 mV.

f. Record the maximum absolute change of the five readings in the test record.

g. Reduce voltage level below 10 V_{RMS}. This is to ensure that the voltage does not remain at unsafe levels.

Appendix G: User Service

This appendix contains service-related information for the VX4101 that covers the following topics:

- Performance Verification
- Preventive maintenance
- Troubleshooting
- User-replaceable parts

Performance Verification

You may use the Performance Check procedure listed in the *Getting Started* section of that manual to verify that the module is operating correctly. The instrument has been fully tested and calibrated before leaving the factory.

If the self test indicates a failure, contact your Tektronix field office or representative for assistance.

Preventive Maintenance

You should perform inspection and cleaning as preventive maintenance. Preventive maintenance, when done regularly, may prevent VX4101 malfunction and enhance reliability. Inspect and clean the VX4101 as often as conditions require by following these steps:

1. Turn off power and remove the VX4101 from the VXIbus mainframe.
2. Remove loose dust on the outside of the instrument with a lint-free cloth.
3. Remove any remaining dirt with a lint-free cloth dampened with water or a 75% isopropyl alcohol solution. Do not use abrasive cleaners.

Troubleshooting

If you suspect a malfunction, first double check connections to and from the VX4101. If the trouble persists, perform a self test.

If the self test indicates a failure, contact your Tektronix field office or representative for assistance.

Appendix H: Replaceable Parts

This section contains a list of the replaceable modules for the VX4101. Use this list to identify and order replacement parts.

Parts Ordering Information

Replacement parts are available through your local Tektronix field office or representative.

Changes to Tektronix products are sometimes made to accommodate improved components as they become available and to give you the benefit of the latest improvements. Therefore, when ordering parts, it is important to include the following information in your order.

- Part number
- Instrument type or model number
- Instrument serial number
- Instrument modification number, if applicable

If you order a part that has been replaced with a different or improved part, your local Tektronix field office or representative will contact you concerning any change in part number.

Change information, if any, is located at the rear of this manual.

Module Servicing

Modules can be serviced by selecting one of the following three options. Contact your local Tektronix service center or representative for repair assistance.

Module Exchange. In some cases you may exchange your module for a remanufactured module. These modules cost significantly less than new modules and meet the same factory specifications. For more information about the module exchange program, call 1-800-TEK-WIDE, extension 6630.

Module Repair and Return. You may ship your module to us for repair, after which we will return it to you.

New Modules. You may purchase replacement modules in the same way as other replacement parts.

Using the Replaceable Parts List

This section contains a list of the mechanical and/or electrical components that are replaceable for the VX4101. Use this list to identify and order replacement parts. The following table describes each column in the parts list.

Parts List Column Descriptions

Column	Column Name	Description
1	Figure & Index Number	Items in this section are referenced by figure and index numbers to the exploded view illustrations that follow.
2	Tektronix Part Number	Use this part number when ordering replacement parts from Tektronix.
3 and 4	Serial Number	Column three indicates the serial number at which the part was first effective. Column four indicates the serial number at which the part was discontinued. No entries indicates the part is good for all serial numbers.
5	Qty	This indicates the quantity of parts used.
6	Name & Description	An item name is separated from the description by a colon (:). Because of space limitations, an item name may sometimes appear as incomplete. Use the U.S. Federal Catalog handbook H6-1 for further item name identification.
7	Mfr. Code	This indicates the code of the actual manufacturer of the part.
8	Mfr. Part Number	This indicates the actual manufacturer's or vendor's part number.

Abbreviations

Abbreviations conform to American National Standard ANSI Y1.1–1972.

Mfr. Code to Manufacturer Cross Index

The table titled Manufacturers Cross Index shows codes, names, and addresses of manufacturers or vendors of components listed in the parts list.

Manufacturers Cross Index

Mfr. Code	Manufacturer	Address	City, State, Zip Code
00779	AMP INC.	CUSTOMER SERVICE DEPT PO BOX 3608	HARRISBURG, PA 17105–3608
06383	PANDUIT CORP	17303 RIDGELAND AVE	TINLEY PARK, IL 60477–3048
0KB01	STAUFFER SUPPLY CO	810 SE SHERMAN	PORTLAND, OR 97214–4657
22526	BERG ELECTRONICS INC	857 OLD TRAIL ROAD	ETTERS, PA 17319
30817	INSTRUMENT SPECIALTIES CO INC	EXIT 53, RT 80 BOX A	DELAWARE WATER GAP, PA 18327
62559	SCHROFF INC	170 COMMERCE DRIVE	WARWICK, RI 02886–2430
75915	LITTELFUSE INC	800 E NORTHWEST HWY	DES PLAINES, IL 60016–3049
80009	TEKTRONIX INC	14150 SW KARL BRAUN DR PO BOX 500	BEAVERTON, OR 97077–0001
93907	CAMCAR DIV OF TEXTRON INC	ATTN: ALICIA SANFORD 516 18TH AVE	ROCKFORD, IL 611045181
TK1943	NEILSEN MANUFACTURING INC	3501 PORTLAND RD NE	SALEM, OR 97303
TK2647	INSTRUMENT SPECIALTIES CO INC.	C/O TEMCO NW 1336 SE 51ST STREET	HILLSBORO, OR 97123

Replaceable Parts List

Fig. & Index Number	Tektronix Part Number	Serial No. Effective	Serial No. Discont'd	Qty	Name & Description	Mfr. Code	Mfr. Part Number
	070–9678–00			1	MANUAL,TECH:USERS MANUAL		
	070–9679–00			1	MANUAL,TECH:REFERENCE GUIDE		
1	334–9090–00			1	MARKER,IDENT:LABEL MARKED VX4101	80009	334–9090–00
2	367–0411–00			1	HANDLE,EJECTOR:TOP,SINGLE WIDE	62559	20817–328
3	950–4827–00			2	92502–25008:SCREW PHIL M 2.5 X 8 CSK	0KB01	950–4827–00
4	441–2085–00			1	CHASSIS:CHASSIS,VXI APPLICATION	80009	441–2085–00
5	348–1434–00			4	GASKET,EMI:2.912 L	30817	97–613–17–029
6	174–2675–00			1	CA ASSY,SP,ELEC:100 OHM,1.9NS TWISTED PAIR,SHEILDED (REFERENCE DESIGNATOR, J22/J23)	00779	ORDER BY DESCRIPTION
	174–2675–00			1	CA ASSY,SP,ELEC:100 OHM,1.9NS TWISTED PAIR,SHEILDED (REFERENCE DESIGNATOR, J21/J24)	00779	ORDER BY DESCRIPTION
7	671–3328–00			1	CKT BD ASSY:MULTI–PACK MODULE	80009	671–3328–00
8	159–5014–00			1	FUSE,SMD:2.0A,125V,FAST BLOW,0.1 X 0.1 X (REFERENCE DESIGNATOR, F1141)	75915	R451 002
	159–5014–00			1	FUSE,SMD:2.0A,125V,FAST BLOW,0.1 X 0.1 X (REFERENCE DESIGNATOR, F1352)	75915	R451 002
	159–5014–00			1	FUSE,SMD:2.0A,125V,FAST BLOW,0.1 X 0.1 X (REFERENCE DESIGNATOR, F1451)	75915	R451 002
	159–5014–00			1	FUSE,SMD:2.0A,125V,FAST BLOW,0.1 X 0.1 X (REFERENCE DESIGNATOR, F651)	75915	R451 002
	159–5014–00			1	FUSE,SMD:2.0A,125V,FAST BLOW,0.1 X 0.1 X (REFERENCE DESIGNATOR, F652)	75915	R451 002
9	159–5015–00			1	FUSE,SMD:10.0A,125V,FAST BLOW (REFERENCE DESIGNATOR, F75)	75915	R451 010
10	N/A			1	For Service use only	N/A	N/A
11	671–3532–00			1	CKT BD ASSY:CPU UPPER BOARD	80009	671–3532–00
12	211–0894–00			5	SCREW,MACHINE:4–40 X 0.625,T–10,TORX DR	0KB01	211–0894–00
13	211–0897–00			6	SCREW,MACHINE:4–40 X 0.188,T–10,TORX	0KB01	211–0897–00
14	200–4231–00			1	COVER:VXI APPLICATION	TK1943	200–4231–00
15	159–5010–00			1	FUSE,SMD:7A,125V,FAST BLOW,0.1 X 0.1X 0.24,UL REG,CSA CERT, (REFERENCE DESIGNATOR, F1351)	75915	451007
16	211–0373–00			12	SCREW,MACHINE:4–40 X 0.250,T–10 TORX DR	93907	ORDER BY DESCRIPTION
17	337–4070–00			1	SHIELD:SHIELD,DMM BOARD	80009	337–4070–00
18	671–3533–00			1	CKT BD ASSY:DMM PLUS	80009	671–3533–00
19	950–3794–00			2	92505–25005:WASHER WAVY 2.7MM	80009	950–3794–00
20	950–4448–00			2	92501–25010:SCREW M2.5X10 CHEESEHEAD	80009	950–4448–00
21	159–5014–00			1	FUSE,SMD:2.0A,125V,FAST BLOW,0.1 X 0.1 X (REFERENCE DESIGNATOR, F01)	75915	R451 002
22	174–3653–00			1	CABLE ASSY:COAX,RFP,50 OHM,6.0L,SMA,MALE,STR,REAR,PNL,MT,131–0888–00 X PELTOLA	TK2469	501–1429–05

Replaceable Parts List (Cont.)

Fig. & Index Number	Tektronix Part Number	Serial No. Effective	Serial No. Discont'd	Qty	Name & Description	Mfr. Code	Mfr. Part Number
23	174–3549–00			1	CABLE ASSY:COAX,RFP,50 OHM,6.0L,SMA,MALE, STR X PELTOLA (REFERENCE DESIGNATOR, J32–1)	80009	174–3549–00
24	211–0914–00			2	JACKSCREW:JACKSCREW,4–40 X 0.394	80009	211–0914–00
25	160–9691–01			1	IC, MEMORY:CMOS,EPROM (REFERENCE DESIGNATOR, U1121)	80009	160–9691–01
26	131–3199–00			1	CONN,SHUNT:SHUNT,FEMALE,JUMPER (REFERENCE DESIGNATOR, J39, PINS 1,2)	22526	68786–202
27	131–0955–00			2	CONN,RF JACK:BNC,50 OHM,FEMALE	00779	87–3334–017
28	367–0410–00			1	HANDLE,EJECTOR:BOTTOM,SINGLE WIDE	62559	20817–327
29	386–6900–00			1	PANEL VXI:PANEL,COUNTER BOARD	80009	386–6900–00
30	211–0391–00			5	SCR,ASSEM,WSHR:2–56 X 0.437,T–8 TORX DR	93907	ORDER BY DESCRIPTION
31	334–7519–00			1	MARKER,IDENT:MKD VXIBUS HANDLE,EJECTOR,LEXAN	0KB05	334–7519–00
32	348–1365–01			1	SHLD GSKT,ELEC:SYMETRICAL SLOTTED FINGER	TK2647	348–1365–01
33	214–4709–00			1	KEY:KEY TTL RIGHT	80009	214–4709–00
34	131–0890–01			1	CONN,HARDWARE:DSUB,JACK SCREW,4–40 X 0.312 ,W/O WASHERS & NUT	00779	205818–2
35	671–3537–00			1	CKT BD ASSY:COUNTER BOARD	80009	671–3537–00
36	343–0549–00			3	STRAP,TIEDOWN,E:0.098 W X 4.0 L	06383	PLT1M

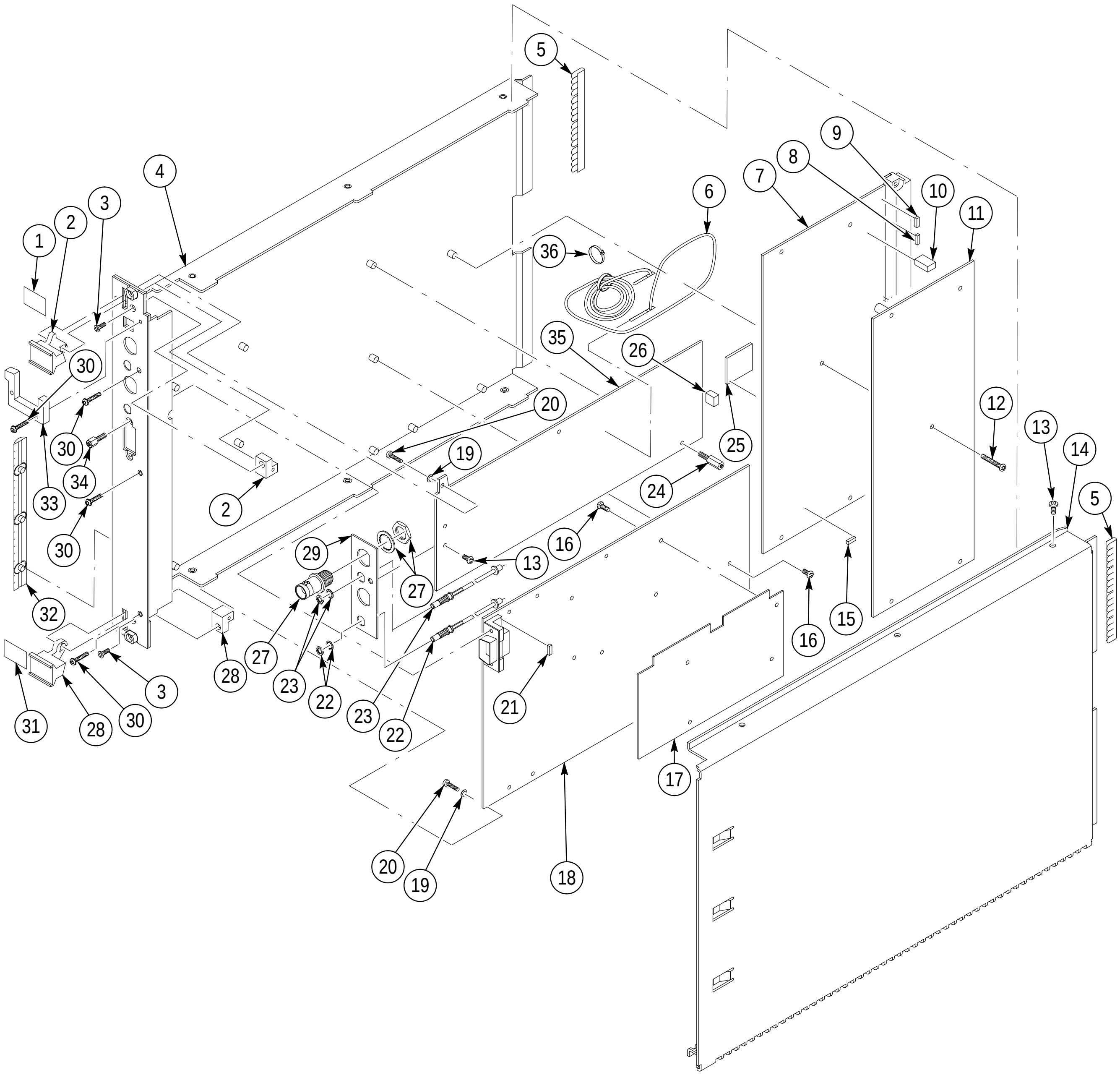


Figure H–1: VX4101 Exploded View

Dressing the Delay Line Cable

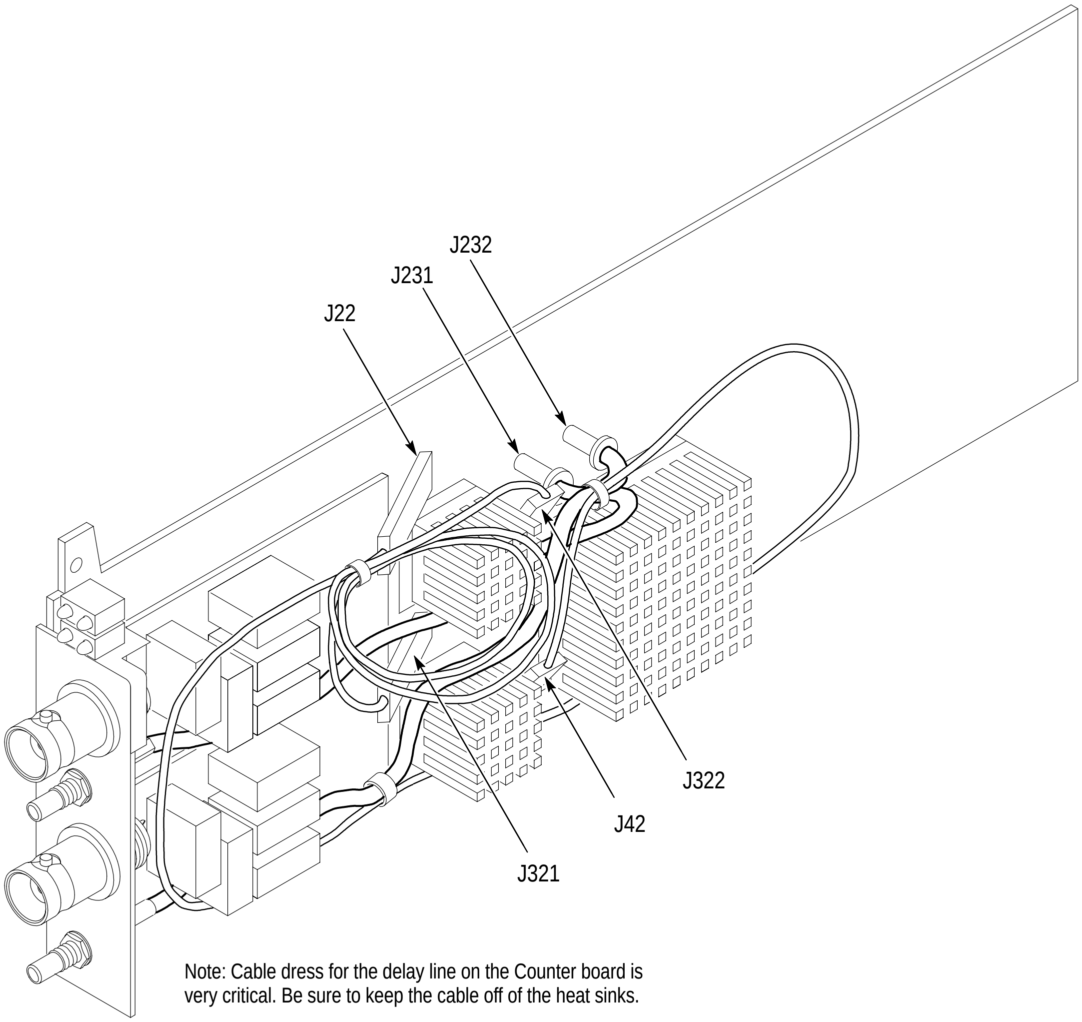


Figure H–2: VX4101 Delay Line Cable Dress

Glossary and Index

Glossary

The terms in this glossary are defined as used in the VXIbus System. Although some of these terms may have different meanings in other systems, it is important to use these definitions in VXIbus applications. Terms which apply only to a particular instrument module are noted. Not all terms appear in every manual.

Accessed Indicator
An amber LED indicator that lights when the module identity is selected by the Resource Manager module, and flashes during any I/O operation for the module.

ACFAIL*
A VMEbus backplane line that is asserted under these conditions: 1) by the mainframe Power Supply when a power failure has occurred (either ac line source or power supply malfunction), or 2) by the front panel ON/STANDBY switch when switched to STANDBY.

A-Size Card
A VXIbus instrument module that is 100.0 × 160 mm × 20.32 mm (3.9 × 6.3 in × 0.8 in), the same size as a VMEbus single-height short module.

Asynchronous Communication
Communications that occur outside the normal "command-response" cycle. Such communications have higher priority than synchronous communication.

Backplane
The printed circuit board that is mounted in a VXIbus mainframe to provide the interface between VXIbus modules and between those modules and the external system.

B-Size Card
A VXIbus instrument module that is 233.4 × 160 mm × 20.32 mm (9.2 × 6.3 in × 0.8 in), the same size as a VMEbus double-height short module.

Bus Arbitration
In the VMEbus interface, a system for resolving contention for service among VMEbus Master devices on the VMEbus.

Bus Timer
A functional module that measures the duration of each data transfer on the Data Transfer Bus (DTB) and terminates the DTB cycle if the duration is excessive. Without the termination capability of this module, a Bus Master attempt to transfer data to or from a non-existent Slave location could result in an infinitely long wait for the Slave response.

Client

In shared memory protocol (SMP), that half of an SMP channel that does not control the shared memory buffers.

CLK10

A 10 MHz, ±100 ppm, individually buffered (to each module slot), differential ECL system clock that is sourced from Slot 0 and distributed to Slots 1–12 on P2. It is distributed to each module slot as a single source, single destination signal with a matched delay of under 8 ns.

CLK100

A 100 MHz, ±100 ppm, individually buffered (to each module slot), differential ECL system clock that is sourced from Slot 0 and distributed to Slots 1–12 on P3. It is distributed to each module slot in synchronous with CLK10 as a single source, single destination signal with a maximum system timing skew of 2 ns, and a maximum total delay of 8 ns.

Commander

In the VXIbus interface, a device that controls another device (a servant). A commander may be a servant of another commander.

Command

A directive to a device. There are three types of commands:

In Word Serial Protocol, a 16-bit imperative to a servant from its commander.

In Shared Memory Protocol, a 16-bit imperative from a client to a server, or vice versa.

In a Message, an ASCII-coded, multi-byte directive to any receiving device.

Communication Registers

In word serial protocol, a set of device registers that are accessible to the commander of the device. Such registers are used for inter-device communications, and are required on all VXIbus message-based devices.

Configuration Registers

A set of registers that allow the system to identify a (module) device type, model, manufacturer, address space, and memory requirements. In order to support automatic system and memory configuration, the VXIbus standard specifies that all VXIbus devices have a set of such registers, all accessible from P1 on the VMEbus.

C-Size Card

A VXIbus instrument module that is 340.0 mm × 233.4 mm × 30.48 mm (13.4 in. × 9.2 in × 1.2 in).

Custom Device
A special-purpose VXIbus device that has configuration registers so as to be identified by the system and to allow for definition of future device types to support further levels of compatibility.

Data Transfer Bus
One of four buses on the VMEbus backplane. The Data Transfer Bus allows Bus Masters to direct the transfer of binary data between Masters and Slaves.

DC SUPPLIES Indicator
A red LED indicator that illuminates when a DC power fault is detected on the backplane.

Device Specific Protocol
A protocol for communication with a device that is not defined in the VXIbus specification.

D-Size Card
A VXIbus instrument module that is 340.0 × 366.7 mm × 30.48 mm (13.4 × 14.4 in × 1.2 in).

DTB
See Data Transfer Bus.

DTB Arbiter
A functional module that accepts bus requests from Requester modules and grants control of the DTB to one Requester at a time.

DUT
Device Under Test.

ECLTRG
Six single-ended ECL trigger lines (two on P2 and four on P3) that function as inter-module timing resources, and that are bussed across the VXIbus subsystem backplane. Any module, including the Slot 0 module, may drive and receive information from these lines. These lines have an impedance of 50 Ω; the asserted state is logical High.

Embedded Address
An address in a communications protocol in which the destination of the message is included in the message.

ESTST
Extended STart/STop protocol; used to synchronize VXIbus modules.

Extended Self Test
Any self test or diagnostic power-on routine that executes after the initial kernel self test program.

External System Controller
The host computer or other external controller that exerts overall control over VXIbus operations.

FDC
See *Fast Data Channel*

Fast Data Channel
A communication protocol that uses a block of memory that is accessible to both client and server. The memory block operates as a message buffer for either data or command communication.

FAILED Indicator
A red LED indicator that lights when a device on the VXIbus has detected an internal fault. This might result in the assertion of the SYSFAIL* line.

IACK Daisy Chain Driver
The circuit that drives the VMEbus Interrupt Acknowledge daisy chain line that runs continuously through all installed modules or through jumpers across the backplane.

ID-ROM
An NVRAM storage area that provides for non-volatile storage of diagnostic data.

Instrument Module
A plug-in printed circuit board, with associated components and shields, that may be installed in a VXIbus mainframe. An instrument module may contain more than one device. Also, one device may require more than one instrument module.

Interface Device
A VXIbus device that provides one or more interfaces to external equipment.

Interrupt Handler
A functional module that detects interrupt requests generated by Interrupters and responds to those requests by requesting status and identity information.

Interrupter
A device capable of asserting VMEbus interrupts and performing the interrupt acknowledge sequence.

IRQ
The Interrupt ReQuest signal, which is the VMEbus interrupt line that is asserted by an Interrupter to signify to the controller that a device on the bus requires service by the controller.

Local Bus
A daisy-chained bus that connects adjacent VXIbus slots.

Local Controller
The instrument module that performs system control and external interface functions for the instrument modules in a VXIbus mainframe or several mainframes. See Resource Manager.

Local Processor
The processor on an instrument module.

Logical Address
The smallest functional unit recognized by a VXIbus system. It is often used to identify a particular module.

Mainframe
Card Cage. For example, the Tektronix VX1410 Mainframe, an operable housing that includes 13 C-size VXIbus instrument module slots.

Memory Device
A storage element (such as bubble memory, RAM, and ROM) that has configuration registers and memory attributes (such as type and access time).

Message
A series of data bytes that are treated as a single communication, with a well defined terminator and message body.

Message Based Device
A VXIbus device that supports VXI configuration and communication registers. Such devices support the word serial protocol, and possibly other message-based protocols.

MODID Lines
Module/system identity lines.

Physical Address
The address assigned to a backplane slot during an access.

Power Monitor
A device that monitors backplane power and reports fault conditions.

P1
The top-most backplane connector for a given module slot in a vertical mainframe such as the Tektronix VX1410. The left-most backplane connector for a given slot in a horizontal mainframe.

P2
The bottom backplane connector for a given module slot in a vertical C-size mainframe such as the VX1410; or the middle backplane connector for a given module slot in a vertical D-size mainframe such as the VX1500.

Query
A form of command that allows for inquiry to obtain status or data.

READY Indicator
A green LED indicator that lights when the power-on diagnostic routines have been completed successfully. An internal failure or failure of +5 V power will extinguish this indicator.

Register Based Device
A VXIbus device that supports VXI register maps, but not high level VXIbus communication protocols; includes devices that are register-based servant elements.

Requester
A functional module that resides on the same module as a Master or Interrupt Handler and requests use of the DTB whenever its Master or Interrupt Handler requires it.

Resource Manager
A VXIbus device that provides configuration management services such as address map configuration, determining system hierarchy, allocating shared system resources, performing system self test diagnostics, and initializing system commanders.

Self Calibration
A routine that verifies the basic calibration of the instrument module circuits, and adjusts this calibration to compensate for short- and long-term variables.

Self Test
A set of routines that determine if the instrument module circuits will perform according to a given set of standards. A self test routine is performed upon power-on.

Servant
A VXIbus message-based device that is controlled by a commander.

Server
A shared memory device that controls the shared memory buffers used in a given Shared Memory Protocol channel.

Shared Memory Protocol
A communications protocol that uses a block of memory that is accessible to both client and server. The memory block operates as a message buffer for communications.

Slot 0 Controller
See Slot 0 Module. Also see Resource Manager.

Slot 0 Module
A VXIbus device that provides the minimum VXIbus slot 0 services to slots 1 through 12 (CLK10 and the module identity lines), but that may provide other services such as CLK100, SYNC100, STARBUS, and trigger control.

SMP
See Shared Memory Protocol.

STARX
Two (2) bi-directional, 50 Ω, differential ECL lines that provide for inter-module asynchronous communication. These pairs of timed and matched delay lines connect slot 0 and each of slots 1 through 12 in a mainframe. The delay between slots is less than 5 ns, and the lines are well matched for timing skew.

STARY
Two (2) bi-directional, 50 Ω, differential ECL lines that provide for inter-module asynchronous communication. These pairs of timed and matched delay lines connect slot 0 and each of slots 1 through 12 in a mainframe. The delay between slots is less than 5 ns, and the lines are well matched for timing skew.

STST
STart/STop protocol; used to synchronize modules.

SYNC100
A Slot 0 signal that is used to synchronize multiple devices with respect to a given rising edge of CLK100. These signals are individually buffered and matched to less than 2 ns of skew.

Synchronous Communications
A communications system that follows the “command-response” cycle model. In this model, a device issues a command to another device; the second device executes the command; then returns a response. Synchronous commands are executed in the order received.

SYSFAIL*
A signal line on the VMEbus that is used to indicate a failure by a device. The device that fails asserts this line.

System Clock Driver
A functional module that provides a 16 MHz timing signal on the Utility Bus.

System Hierarchy
The tree structure of the commander/servant relationships of all devices in the system at a given time. In the VXIbus structure, each servant has a commander. A commander may also have a commander.

Test Monitor
An executive routine that is responsible for executing the self tests, storing any errors in the ID-ROM, and reporting such errors to the Resource Manager.

Test Program
A program, executed on the system controller, that controls the execution of tests within the test system.

Test System
A collection of hardware and software modules that operate in concert to test a target DUT.

TTLTRG
Open collector TTL lines used for inter-module timing and communication.

VXIbus Subsystem
One mainframe with modules installed. The installed modules include one module that performs slot 0 functions and a given complement of instrument modules. The subsystem may also include a Resource Manager.

Word Serial Protocol
A VXIbus word oriented, bi-directional, serial protocol for communications between message-based devices (that is, devices that include communication registers in addition to configuration registers).

Word Serial Communications
Inter-device communications using the Word Serial Protocol.

WSP
See Word Serial Protocol.

10-MHz Clock
A 10 MHz, ±100 ppm timing reference. Also see CLK10.

100-MHz Clock
A 100 MHz, ±100 ppm clock synchronized with CLK10. Also see CLK100.

488-To-VXIbus Interface
A message based device that provides for communication between the IEEE-488 bus and VXIbus instrument modules.

Index

A

C

D

E

F

I

L

M

N

O

P

R

S

T

V