COPPER MOUNTAIN®
TECHNOLOGIES

631 E New York Street | Indianapolis, IN 46202 USA
www.coppermountaintech.com

CmtNA IVI-C Reference

Revision 21.2 13.08.2021

U.S.: +1.317.222.5400
Latin America: +1.9154.706.5920
Singapore: +65.63.23.6546
EMAE: +44 75 03 69 21 13

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

CmtNA IVI-C Driver

Welcome to the **CmtNA** version **21.2.0** IVI driver. CMT's IVI drivers simplify the creation and maintenance of instrument control applications in a variety of development environments. IVI drivers allow to programmatically control instrumentation while providing a greater degree of instrument interchangeability and code reuse. This driver supports compiling application programs for 32 or 64 bit platforms.

Programming with the CmtNA Driver

Basics of programming this driver including example code. How to: Reference the Driver, Create an Instance of the Driver, Initialize the Driver, and Access Items of Various Data Types.

Programming with the IVI-C Driver - Using the IVI-C driver in the following application development environments: Visual C++, NI LabWindow/CVI, NI LabVIEW.

Example Programs installed with the driver can be built and run. They demonstrate basic driver usage in a variety of application development environments.

CmtNA Driver Reference

IVI-C Driver Reference - Documentation for each IVI-C function, attribute, and error code.

IVI Overview

General information on IVI features common to all IVI drivers as well as compliance and installation information.

IVI Compliance Information - IVI-required compliance information for the CmtNA driver.

Installation - Compliance information for installation and update CmtNA driver.

Web Sites

Copper Mountain Technologies

IVI Driver Downloads

IVI Foundation - Usage Guides, Specifications, Shared Components Downloads

MSDN Online

Getting Started

To get started using the CmtNA device and driver software, refer to the getting started guide for a device. The getting started guide explains how to complete the following tasks:

- Install the software and hardware.
- Configuring Hardware in MAX.
- Program the hardware.
- Make the first measurement.
- Troubleshoot.

The getting started guide for a device is printed in the hardware kit, installed with the driver software, and available at Copper Mountain Technologies.

NOTE Complete the getting started guide for a device before using the information in this help file.

Install the software and hardware

NOTE	Install the software before inserting the PXIe-S5090 harware to the PXIe chassis.

1. Verify that the following software components are installed on the PXIe system on which the PXIe-S5090 software depends:

- NI FlexRIO Driver
- NI-VISA
- IVI Shared Components

Install them if necessary.

2. Install the CMT PXIe-S5090 VNA software.

- Download the installation file **Setup_PXIe_S2VNA_vX.X.exe** from the Copper Mountain Technologies web site.
- Run the Setup_PXIe_S2VNA_vX.X.exe installer file onto embedded controller in the PXIe chassis. Follow the instructions of the installation wizard.

3. Power off the chassis. Install PXIe-S5090 to the PXIe chassis. Power on the chassis.

4. Use Measurement & Automation Explorer (MAX) to check CMT hardware (see Configuring Hardware in MAX).

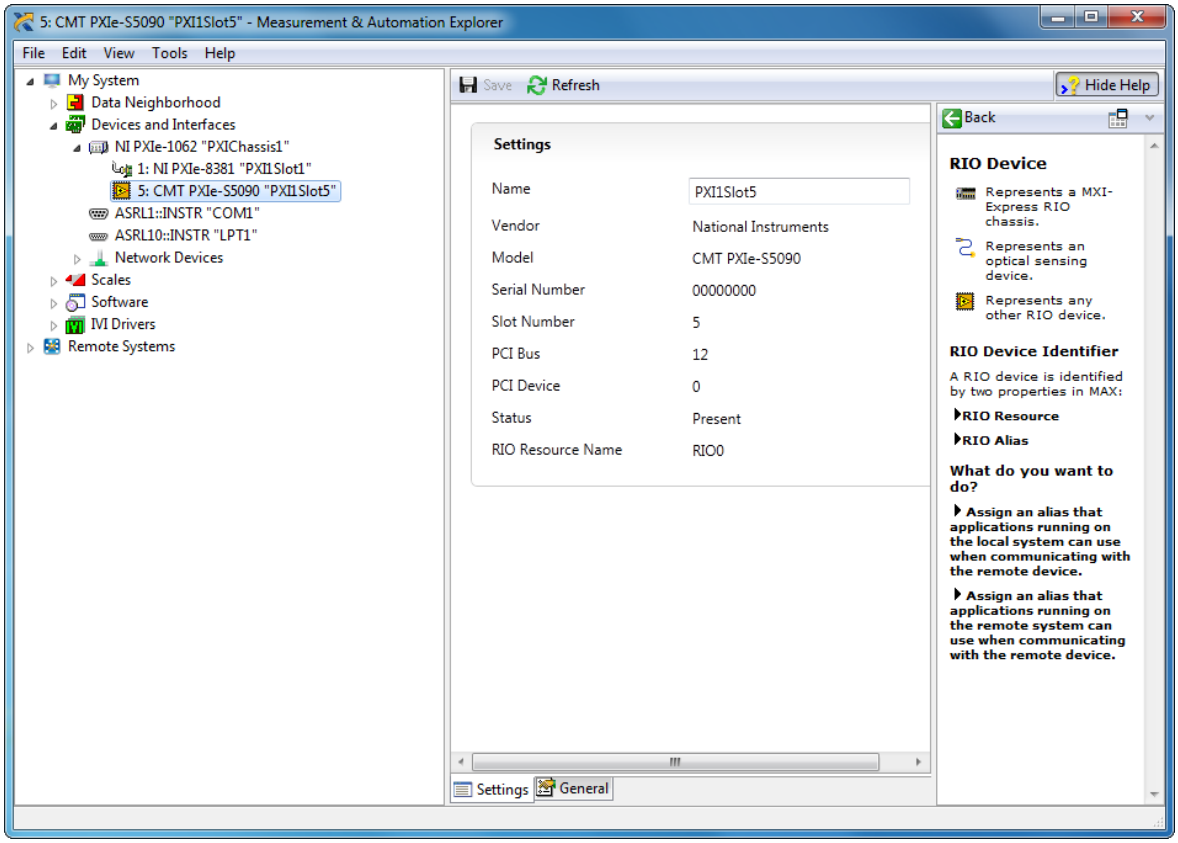
Configuring Hardware in MAX

Use Measurement & Automation Explorer (MAX) to configure CMT hardware. MAX informs other programs about which devices reside in the system and how they are configured. MAX is automatically installed with CmtNA.

1. Launch MAX by navigating to **Start > All Programs > National Instruments > Measurement & Automation** or by clicking the **Measurement & Automation** desktop icon.

2. In the **Configuration** pane, double-click **Devices and Interfaces** to see the list of installed devices. Installed devices appear under the name of their associated chassis.

3. Expand the Chassis tree item. MAX lists all devices installed in the chassis. Default device names may vary.



NOTE If the hardware listed is not seen, press to refresh the list of installed devices. If the device is still not listed, power off the system, ensure the device is correctly installed, and restart.

4. Record the device identifier MAX assigns to the hardware. Use this identifier when programming NI vector network analyzer device.

5. The MAX self-test performs a basic verification of hardware resources. To self-test a module in MAX, select the module in the configuration tree and select Self-Test in the MAX toolbar.

For more details of the renaming CMT vector network analyzer module, see Renaming a Hardware Module in MAX.

Renaming a Hardware Module in MAX

Measurement & Automation Explorer (MAX) allows to rename CMT vector network analyzer module. The MAX name is used in CMtNA to operate the hardware resources. it is not required to change the module name from the default but doing so can make programming easier.

To rename a hardware module in MAX, complete the following steps:

1. Select the VNA module in the chassis tree item and select Rename in the MAX toolbar.
2. Enter the new name for the module.

NOTE The device name must consist only of alphanumeric and underscore characters.

---

3. Press <Enter>.
4. Verify that the new name for the module is displayed.

IVI Compliance Information

This section provides IVI-required compliance information for the CmtNA driver.

Compliance Category

IVI drivers come in a variety of types and configurations. This topic provides IVI-required compliance information on the various categories of IVI compliance for the CmtNA driver.

Compliance Category	Compliance Information
Category Name	IVI-C Custom Specific Instrument Driver.
Class Specification Version	No IVI instrument class supported.

Optional Features

This topic provides IVI-required information regarding optional IVI driver features supported by the CmtNA driver.

Feature	Supported
Interchangeability checking	No
State caching	No
Coercion recording	No

Driver Identification

This topic provides IVI-required identity information for the CmtNA driver.

Identification Category	Description
Driver Revision	21.2.0
Driver Vendor	Cmt IVI Drivers
Driver Description	IVI-C Driver for CMT Network Analyzers
Prefix/Component Identifier	CmtNA
Supported Models	PXIe-S5090

File Locations

The installer **Setup_PXIe_S2VNA_vX.X.exe** installs the following files in the specified locations.

Directory disignations used:

<CmtGuiRootDir> is an alias for the following Coppermountaintech file folder location:

- **(32-bit)** Program Files\Coppermountaintech\PXI S2VNA\
- **(64-bit)** Program Files (x86)\Coppermountaintech\PXI S2VNA\

<CmtDataRootDir> is an alias for the following Coppermountaintech file folder location: Program Data\Coppermountaintech\PXI S2VNA\

<IVIROOTDIR32> is an alias for the following Coppermountaintech file folder location:

- **(32-bit)** Program Files\IVI Foundation\IVI\
- **(64-bit)** Program Files (x86)\IVI Foundation\IVI\

<IVIROOTDIR64> is an alias for the following Coppermountaintech file folder location:

- **(64-bit)** Program Files\IVI Foundation\IVI\

Item	Installed Location
S2VNA main GUI application	<CmtGuiRootDir>
S2VNA operating manual S2VNA SCPI programming manual	<CmtGuiRootDir>\Doc
PXI hardware driver	<CmtGuiRootDir>\PxiDriver
USB driver for automatic calibration module	<CmtGuiRootDir>\UsbDriver
Utilities	<CmtGuiRootDir>\Utilities

Item	Installed Location
CmtNA IVI-C driver help	<IVIROOTDIR32>\Drivers\CmtNA\Doc
CmtNAIVI-C driver Header files	<IVIROOTDIR32>\Include
CmtNA IVI-C driver Import Library files (32-bit)	<IVIROOTDIR32>\Lib\msc
CmtNA IVI-C driver Import Library files (64-bit)	<IVIROOTDIR32>\Lib_x64\msc
CmtNA IVI-C driver DLL (32-bit)	<IVIROOTDIR32>\Bin

Additional File Locations for Windows 64-bit

Item	Installed Location
CmtNA IVI-C driver Header files	<IVIROOTDIR64>\Include
CmtNA IVI-C driver Import Library files (64-bit)	<IVIROOTDIR64>\Lib_x64\msc
CmtNA IVI-C driver DLL (64-bit)	<IVIROOTDIR64>\Bin

VNA Runtime File Location

Item	File Location
VNA State Files and Data Files Location	<CmtDataRootDir>\<VNA name>, where <VNA name> is a name of the VNA in the PXI system. For example - PXI1Slot5.

Cmt IVI-C Driver

The CmtNA driver includes a compliant IVI-C driver. The topics in this section detail how to perform specific programming tasks using the IVI-C functionality available with the CmtNA driver.

In This Section

Initializing the IVI-C Driver

Presents the steps required to properly initialize the IVI-C driver.

Accessing Repeated Capabilities

Provides information on the supported repeated capabilities and repeated capability names for the CmtNA driver.

Accessing Instrument-Specific Functionality

Explains how to access the specialized functionality of the CmtNA driver.

Using Simulation

Provides information on using the simulation capabilities of the driver.

Programming with the IVI-C Driver in Various Development Environment

Demonstrates how the CmtNA driver can be used in a various application development environments.

Reference

Presents reference documentation for each IVI-C function and attribute.

Initializing the IVI-C Driver

When using this specific driver directly, the client application references the specific driver (through the use of the "CmtNA" in function call names and by directly references CmtNA.h and CmtNA.lib). For many applications, this is perfectly acceptable.

The examples below describing the **Initialize** function present this technique for initializing an IVI-C driver.

Calling the Initialize Function

The IVI-defined **Initialize** function offers a variety of options and parameters that control fundamental aspects of the driver's behavior. It is critical to understand what options are available and how they impact driver operation.

ResourceName

The resource name parameter can be a logical name present in the IVI Configuration Store or a physical resource descriptor. If the ResourceName is a logical name, then additional options stored in the IVI Configuration Store will be loaded and processed. This allows client code to reuse driver option settings and to keep the driver code as concise as possible, by removing settings that can be stored in the IVI Configuration Store. Any parameters or options passed directly to the **Initialize** function via the OptionString parameter (discussed below) override any settings found in the IVI Configuration Store.

Interface	Physical Resource Descriptor Syntax
PXI	PXI[bus]SLOT[#slot] Example: PXI1SLOT3

IdQuery

If this is enabled, the driver will query the instrument model and compare it with a list of instrument models that are supported by the driver. If the model is not supported, **Initialize** will fail with the IVI_ERROR_ID_QUERY_FAILED error code.

Reset

If this is enabled, the driver will perform a reset of the instrument. If the reset fails, **Initialize** will fail with the IVI_ERROR_RESET_FAILED error code.

OptionString

The OptionString allows the user to pass optional settings to the driver. These settings override any settings that are specified in the IVI Configuration Store. If the IVI Configuration Store is not used, (a resource descriptor is passed as the ResourceName instead of a logical name) then any setting that is not specified has a default value as specified by IVI.

Option Name	Description	Default
QueryInstrStatus	Specifies whether the IVI specific driver queries the instrument status at the end of each user operation. Querying the instrument status is very useful for debugging. After validating the program, the user can set this attribute to False to disable status checking and maximize performance. The user specifies this value for the entire IVI driver session.	false
Simulate	Specifies whether or not the IVI specific driver simulates instrument driver I/O operations. If simulation is enabled, the specific driver functions do not perform instrument I/O. For output parameters that represent instrument data, the specific driver functions return simulated values.	false
Cache	Not supported. Specifies whether or not to cache the value of attributes. When caching is enabled, the IVI specific driver keeps track of the current instrument settings so that it can avoid sending redundant commands to the instrument.	false
InterchangeCheck	Not supported. Specifies whether the IVI specific driver performs interchangeability checking. If the Interchange Check attribute is enabled, the specific driver maintains a record of each interchangeability warning that it encounters. The user calls the Get Next	false

Option Name	Description	Default
	Interchange Warning function to extract and delete the oldest interchangeability warning from the list.	
RangeCheck	Not supported. Specifies whether the IVI specific driver validates attribute values and function parameters. If enabled, the specific driver validates the parameter values that users pass to driver functions. Validating attribute values and function parameters is useful for debugging. After validating the program, the user can set this attribute to False to disable range checking and maximize performance.	false
RecordCoercions	Not supported. Specifies whether the IVI specific driver keeps a list of the value coercions it makes for ViInt32 and ViReal64 attributes. If the Record Value Coercions attribute is enabled, the specific driver maintains a record of each coercion. The user calls the Get Next Coercion Record function to extract and delete the oldest coercion record from the list.	false
DriverSetup	Specifies additional settings supported by the driver, but not defined by IVI.	""

DriverSetup

This is used to specify settings that are supported by the driver but not defined by IVI. If the Options String parameter contains an assignment for the Driver Setup attribute, the Initialize function assumes that everything following 'DriverSetup=' is a part of the assignment. The following settings are supported by the CmtNA driver.

Option Name	Description	Default
Visible	If true, UI is visible.	false

Option Name	Description	Default
WaitReady	If true, initialization function waits for the device to be ready (up to 15 seconds).	false
DirectVisaAddress	If true, its VISA Address is used instead of the ResourceName. For example: VISA Address: "TCPIP0::127.0.0.1::hislip0::INSTR", key with the option line: "DirectVisaAddress=true". DirectVisaAddress allows the user to control Copper Mountain Technologies Vector Network Analyzers (VNA) (S2VNA, PXI S2VNA) that support the HiSLIP protocol and a compatible automation command set. However, there are limitations: • Does not run server applications (S2VNA, PXI-VNA). • Not combined with settings that change the state at startup: "Visible = ", "Simulate=".	false

Example 1 (simulation)

The following code initializes the driver in simulation mode.

C++

```
#include "CmtNA.h"

void main()
{
    ViStatus status;
    ViSession session;

    // Initialize the driver with no range checking and no state caching and reset
    the instrument
```

C++

```
    status = CmtNA_InitWithOptions("PXI1SLOT3", VI_TRUE, VI_TRUE,
    "Simulate=true", &session);


    // ... call any driver functions


    status = CmtNA_close(session);
}
```

Accessing Repeated Capabilities

This page provides information on the supported repeated capabilities and repeated capability names for the CmtNA driver.

Repeated Capabilities

The following table provides detailed information on each repeated capability supported by the CmtNA driver.

Repeated Capability	IVI-C Style	Physical Names		
Channel	Parameter	Channel1 Channel2 Channel3 Channel4 Channel5 Channel6 Channel7 Channel8	Channel9 Channel10 Channel11 Channel12 Channel13 Channel14 Channel15 Channel16	
Measurem ent	Parameter	Measurement1 Measurement2 Measurement3 Measurement4 Measurement5 Measurement6 Measurement7	Measurement9 Measurement10 Measurement11 Measurement12 Measurement13 Measurement14 Measurement15	

Repeated Capability	IVI-C Style	Physical Names		
		Measurement8	Measurement16	
Marker	Parameter	Marker1 Marker2 Marker3 Marker4 Marker5 Marker6 Marker7 Marker8	Marker9 Marker10 Marker11 Marker12 Marker13 Marker14 Marker15 Marker16	
Standard	Parameter	Standard1 Standard2 Standard3 Standard4 Standard5 Standard6 Standard7 Standard8 Standard9 Standard10	Standard11 Standard12 Standard13 Standard14 Standard15 Standard16 Standard17 Standard18 Standard19 Standard20	Standard21 Standard22 Standard23 Standard24 Standard25 Standard26 Standard27 Standard28 Standard29 Standard30

Repeated Capability	IVI-C Style	Physical Names		
Port	Parameter	Port1 Port2 Port3 Port4 Port5 Port6 Port7 Port8		

Accessing Instrument-Specific Functionality

The instrument-specific capabilities are those methods and properties that are unique to the instrument. In practice, the instrument-specific portion of the driver exposes the full functionality of the instrument -- including functionality that may also be available in the class-compliant capabilities. Users that do not care about interchangeability or that need to access functionality that is not defined by IVI will use the instrument-specific interfaces in application programs.

When to use the instrument specific interfaces

- When the required functionality is not exposed in the class-compliant interfaces.
- When interchangeability is not a requirement.
- When performance is critical. Sometimes a driver may perform better using of the instrument specific interfaces because they match the instrument functionality more closely than the class-compliant interfaces.

Example

The following example demonstrates accessing instrument-specific function on the driver. Note that instrument-specific function calls are always prefixed with the instrument-specific prefix ("CmtNA"). In contrast, calls to IVI-defined class-compliant functions are prefixed with the IVI class name.

C++

```
#include "CmtNA.h"


void main()
{
    ViStatus status;
    ViSession session;


    status = CmtNA_InitWithOptions("PXI1SLOT3", VI_TRUE, VI_TRUE,
    "Simulate=true", &session);


    status = CmtNA_close(session);
}
```

Using Simulation

An IVI driver can operate without the presence of an actual instrument by using simulation mode. When simulation is turned on, the driver performs no instrument communication, but attempts to generate results which allow client applications to reasonably execute. The simulation can be turned on by using either of the following two techniques:

- Passing the "Simulate=true" option string parameter to the **InitWithOptions** function.
- Setting the **CMTNA_ATTR_SIMULATE** attribute to VI_TRUE.

The IVI-C driver contains a **CMTNA_ATTR_SIMULATE** attribute. Note that the simulation can be turned on, if a driver with it is initialized off.

Using The Option String

The fourth and last parameter to the InitWithOptions function is the option string. See the help on InitWithOptions for more details. To turn simulation on with the option string, include the following in the option string:

```
Simulate=true
```

For example, the InitWithOptions call might look like this:

```
CmtNA_InitWithOptions("PXI1SLOT2", VI_TRUE, VI_TRUE, "Simulate=true",
&session);
```

Limitations of Simulation

Simulation is designed to allow the driver to be used without an instrument or any I/O software or hardware. However, simulation is not intended to fully emulate the instrument. There are some important differences to be aware of:

- Measurement data queries will return a small array of values or interpolation.
- External trigger and output power control not working.

Programming with the IVI-C Driver in Various Development Environments

IVI-C drivers are implemented using standard Windows DLL technology. Consequently, IVI-C drivers can be used in a wide variety of development environments.

The topics in this section provide detailed instructions on how to access and use IVI-C drivers in a variety of popular development environments. Each topic includes a complete example of IVI-C driver usage.

Using NI LabVIEW

Provides instructions on how to use the IVI-C driver in National Instruments LabVIEW.

Using Visual C++

Creating an Application with Microsoft Visual Studio 2010 - Describes the process for including an IVI-C driver in a Visual C++ project.

Using .NET

Explains how to use the driver in .NET languages, such as VB.NET and C#.

IVI-C Reference

This section contains information about each function and attribute exposed from the IVI-C driver. Functions and attributes are presented in a hierarchical function tree as well as in a flat list by name.

Functions and Attributes

Displays the IVI-C functions and attributes together in a flat list.

Errors and Warnings

Presents all driver errors and warnings ordered by error code.

Functions and Attributes

Function

Name	Description
CmtNA_init	Opens the I/O session to the instrument. Driver methods and properties that access the instrument are only accessible after Initialize is called. Initialize optionally performs a Reset and queries the instrument to validate the instrument model.
CmtNA_close	Closes the I/O session to the instrument. Driver methods and properties that access the instrument are not accessible after Close is called.
CmtNA_InitWithOptions	Resets the device to a known initialization state with defaults as specified for the device in the optionString parameter in the CmtNA_InitWithOptions function.
CmtNA_self_test	Performs an instrument self test, waits for the instrument to complete the test, and queries the instrument for the results. If the instrument passes the test, TestResult is zero and TestMessage is "Self test passed".
CmtNA_reset	Resets the device to a known initialization state.
CmtNA_revision_query	Queries the revision of the device.

Name	Description
CmtNA_ResetWithDefaults	Does the equivalent of Reset and then disables class extension capability groups, sets attributes to initial values defined by class specs, and configures the driver to option string settings used when Initialize was last executed.
CmtNA_Disable	Places the instrument in a quiescent state where it has minimal or no impact on the system to which it is connected.
CmtNA_LockSession	Obtains a multithread lock on the driver after waiting until all other execution threads have released their locks on the instrument session.
CmtNA_UnlockSession	Releases a previously obtained multithread lock.
CmtNA_error_query	Queries the instrument and returns instrument specific error information. This function can be used when QueryInstrumentStatus is True to retrieve error details when the driver detects an instrument error.
CmtNA_error_message	Translates the error return value from an IVI driver function to a user-readable string. The user should pass a buffer with at least 256 bytes for the ErrorMessage parameter.
CmtNA_GetError	This function retrieves and then clears the IVI error information for the session or the current execution thread. If the user specifies a valid IVI session for the Vi parameter, Get Error retrieves and then clears the

Name	Description
	error information for the session. If the user passes VI_NULL for the Vi parameter, Get Error retrieves and then clears the error information for the current execution thread. If the Vi parameter is an invalid session, the function does nothing and returns an error. Normally, the error information describes the first error that occurred since the user last called the CmtNA_GetError or CmtNA_ClearError function.
CmtNA_ClearError	This function clears the error code and error description for the current execution thread and for the IVI session. If the user specifies a valid IVI session for the Vi parameter, this function clears the error information for the session. If the user passes VI_NULL for the Vi parameter, this function clears the error information for the current execution thread. If the Vi parameter is an invalid session, the function does nothing and returns an error.
CmtNA_GetAttributeViInt32	Queries the value of a ViInt32 attribute. This low-level function can be used to get the values of inherent IVI attributes, and instrument-specific attributes.
CmtNA_GetAttributeViReal64	Queries the value of a ViReal64 attribute. This low-level function can be used to get the values of inherent IVI attributes, and instrument-specific attributes.
CmtNA_GetAttributeViString	Queries the value of a ViString attribute. This low-level function can be used to get the values of inherent IVI attributes, and instrument-specific attributes.

Name	Description
CmtNA_GetAttributeViBoolean	Queries the value of a ViBoolean attribute. This low-level function can be used to get the values of inherent IVI attributes, and instrument-specific attributes.
CmtNA_GetAttributeViSession	Queries the value of a ViSession attribute. This low-level function can be used to get the values of inherent IVI attributes, and instrument-specific attributes.
CmtNA_SetAttributeViInt32	Sets the value of a ViInt32 attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.
CmtNA_SetAttributeViReal64	Sets the value of a ViReal64 attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.
CmtNA_SetAttributeViString	Sets the value of a ViString attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.
CmtNA_SetAttributeViBoolean	Sets the value of a ViBoolean attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.
CmtNA_SetAttributeViSession	Sets the value of a ViSession attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.

Name	Description
CmtNA_ReadString	Reads the entire contents of the buffer until the termination character /n and return the data as string.
CmtNA_WriteString	Writes a string to the I/O Stream and flush the buffer.
CmtNA_Queryf	Performs a formatted write and read through a single call to an operation.
CmtNA_Printf	Converts, formats, and sends the parameters (designated by ...) to the device as specified by the format string. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.
CmtNA_Scanf	Reads, converts, and formats data using the format specifier. Stores the formatted data in the parameters (designated by ...). This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.
CmtNA_VQueryf	Performs a formatted write and read through a single call to an operation. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.
CmtNA_VPrintf	Converts, formats, and sends the parameters (designated by params) to the device or interface as specified by the format string. This

Name	Description
	function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.
CmtNA_VScanf	Reads, converts, and formats data using the format specifier. Stores the formatted data in the parameters designated by params. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.
CmtNA_ConfigureSweep	Configures the number of points for the sweep measurement and the sweep mode.
CmtNA_ConfigureFrequencyCenterSpan	Sets the sweep range for the channel using center and span value of the frequencies.
CmtNA_ConfigureFrequencyStartStop	Sets the sweep range for the channel using start and stop value of the frequencies.
CmtNA_ConfigurePowerStartStop	Sets the sweep range for the channel using start and stop value of the output power.
CmtNA_ConfigurePowerCenterSpan	Sets the sweep range for the channel using center and span value of the output power.
CmtNA_SetSegmentData	Sets the array of the segment sweep table.

Name	Description
CmtNA_GetSegmentData	Gets the array of the segment sweep table.
CmtNA_SaveSegmentTable	Saves the segment table in a file.
CmtNA_RecallSegmentTable	Recalls the segment table file. The file must be saved by the CmtNA_SaveSegmentTable function.
CmtNA_HoldAllChannels	Turns OFF the continuous trigger initiation mode for all channels.
CmtNA_ContinuousAllChannels	Turns ON the continuous trigger initiation mode for all channels.
CmtNA_TriggerInit	Puts the channel to the Trigger Waiting state for the one trigger event.
CmtNA_WaitForTriggerState	Waits the specified state of the analyzer has been reached. The function can be used to wait for the end of the sweep. If the continuous initiation mode is turned ON - the parameter of the function must be WAIT, otherwise HOLD.
CmtNA_WaitForSweepComplete	Minimizes the analyzer main window removing it from desktop.
CmtNA_WaitForOperationComplete	Method returns when all pending operations are complete or maxTimeMilliseconds exceeded.

Name	Description
CmtNA_TriggerSingle	Generates a trigger signal and initiates a sweep. As opposed to the function CmtNA_TriggerImmediate this function is pending till the end of the sweep.
CmtNA_TriggerSingleAndWait	Generates a trigger signal and initiates a sweep. As opposed to the function CmtNA_TriggerImmediate this function is pending till the end of the sweep.
CmtNA_TriggerImmediate	Generates a trigger signal and initiates a sweep. The function is completed immediately after the generation of the trigger signal (does not wait the end of a sweep).
CmtNA_TriggerRestart	Aborts the sweep. The channels in the Single trigger initiation mode transit to the Hold state. The channels in the Continuous trigger initiation mode transit to the trigger waiting state, if the trigger source is set to Internal, the channel immediately starts a new sweep.
CmtNA_MeasurementSetParameter	Sets the measurement parameter of the trace.
CmtNA_MeasurementGetParameter	Gets the measurement parameter of the trace.
CmtNA_MeasurementAutoScale	Executes the auto scale function for the trace.

Name	Description
CmtNA_MeasurementAutoRefValue	Executes the auto reference function for the trace. The function automatically sets the RLEVel value.
CmtNA_ClearAverage	Clears and restarts the averaging process, when the averaging function is turned on.
CmtNA_ChannelFrequencyData	Gets the frequency array of the measurement points.
CmtNA_MeasurementFetchX	Gets the X-axis values array. The X-axis values array is the frequency, power or time values array depending on the trace setup. The array contains real values. The array size is N, where N is the number of measurement points.
CmtNA_MeasurementFetchFormatted	Returns measurement data in the current format as set by the Measurement. Format property. Smith and Polar formats are not supported.
CmtNA_MeasurementFetchComplex	Returns real and imaginary values of measurement data.
CmtNA_MeasurementFetchMemoryFormatted	Returns measurement data in the current format as set by the Measurement.Format property. Smith and Polar formats are not supported.
CmtNA_MeasurementFetchMemoryComplex	Returns real and imaginary values of Memory data.

Name	Description
CmtNA_FunctionExecute	Executes the analysis specified by the CMTNA_ATTR_FUNCTION_TYPE.
CmtNA_GetCalculatedFunctionData	Gets the data array, which is the CmtNA_FunctionExecute function analysis result.
CmtNA_GetMaxchannelCount	Gets the maximum number of the channels.
CmtNA_GetMaxTraceCount	Gets the maximum number of the traces in the channel.
CmtNA_MeasurementDataToMemory	Saves measurement trace data in memory.
CmtNA_TraceHoldRestart	This command resets the trace hold function.
CmtNA_DisplaySetDefaults	Restores the display settings to the default values.
CmtNA_InitUserCal	Initializes user calibration.
CmtNA_InitNPortUserCal	Initializes user calibration.
CmtNA_PerformCalibrationAction	Performs one calibration action on the specified ports or the type of calibration action.

Name	Description
CmtNA_ApplyCalibration	Applies calibration to channels. This method must be executed after acquiring all the Standards during calibration session.
CmtNA_ApplySimplefiedCalibration	Calculates the calibration coefficients for the simplified 3 or 4 port calibration from the calibration standards measurements when the 3 or 4 port calibration is selected as the calibration type.
CmtNA_CancelCalibration	Clears the measurement data of the calibration standards.
CmtNA_ResetCalibration	Clears the calibration coefficient table.
CmtNA_GetCalibrationInfo	Gets the information string of the calibration acting between the source and receiver ports.
CmtNA_PerformUnkThru2PortCal	Completes the full 2-port calibration between the specified ports provided that each port was calibrated using full 1-port calibration
CmtNA_CalStandardInsert	Inserts the calibration standard into the selected calibration kit.
CmtNA_CalStandardRemove	Deletes the calibration standard into the selected calibration kit.
CmtNA_SetCalStandardS1PData	Sets the data array of the data-based calibration standard. The first element of the array is 1 determines the number of ports of the calibration standard.

Name	Description
CmtNA_GetCalStandardS1PData	Gets the data array of the data-based calibration standard. The first element of the array is 1 determines the number of ports of the calibration standard.
CmtNA_SetCalStandardS2PData	Sets the data array of the data-based calibration standard. The first element of the array is 2 determines the number of ports of the calibration standard.
CmtNA_GetCalStandardS2PData	Sets the data array of the data-based calibration standard. The first element of the array is 2 determines the number of ports of the calibration standard.
CmtNA_SetCalKitOrderOpen	Sets the number of the calibration standard of the open type, used for the measurement of the specified port.
CmtNA_SetCalKitOrderShort	Sets the number of the calibration standard of the short type, used for the measurement of the specified port.
CmtNA_SetCalKitOrderLoad	Sets the number of the calibration standard of the load type, used for the measurement of the specified port.
CmtNA_SetCalKitOrderThru	Sets the number of the calibration standard of the thru type, used for the measurement between the source and receiver ports.

Name	Description
CmtNA_SetCalKitOrderTRLLine	Sets the number of the calibration standard of the TRL line type, used for the measurement between the source and receiver ports.
CmtNA_SetCalKitOrderTRLThru	Sets the number of the calibration standard of the TRL thru type, used for the measurement between the source and receiver ports.
CmtNA_SetCalKitOrderTRLReflect	Sets the number of the calibration standard of the TRL Reflect type, used for the measurement of the specified port.
CmtNA_GetCalKitOrderOpen	Gets the number of the calibration standard of the open type, used for the measurement of the specified port.
CmtNA_GetCalKitOrderShort	Gets the number of the calibration standard of the short type, used for the measurement of the specified port.
CmtNA_GetCalKitOrderLoad	Gets the number of the calibration standard of the load type, used for the measurement of the specified port.
CmtNA_GetCalKitOrderThru	Gets the number of the calibration standard of the thru type, used for the measurement between the source and receiver ports.
CmtNA_GetCalKitOrderTRLLine	Gets the number of the calibration standard of the TRL line type, used for the measurement between the source and receiver ports.

Name	Description
CmtNA_GetCalKitOrderTRLThru	Gets the number of the calibration standard of the TRL thru type, used for the measurement between the source and receiver ports.
CmtNA_GetCalKitOrderTRLReflect	Gets the number of the calibration standard of the TRL Reflect type, used for the measurement of the specified port.
CmtNA_SetSubclassStdOrder	Sets the subclass used to specify classes of calibration standards by CmtNA_SetCalKitOrderXXX functions.
CmtNA_GetSubclassStdOrder	Gets the subclass used to specify classes of calibration standards by CmtNA_GetCalKitOrderXXX functions.
CmtNA_SetStdSubclassNumber	Sets the subclass number of calibration standard used for measurement by the subsequent function CmtNA_PerformCalibrationAction.
CmtNA_GetStdSubclassNumber	Gets the subclass number of calibration standard used for measurement by the subsequent function CmtNA_PerformCalibrationAction.
CmtNA_CalKitRestore	Resets the calibration kit to the factory settings. Restores the predefined calibration kit. Removes the user defined calibration kit.
CmtNA_CalKitSaveToFile	Saves the definition file for the calibration kit.

Name	Description
CmtNA_CalKitLoadFromFile	Recalls the definition file for the calibration kit. The file must be saved by the CmtNA_CalKitSaveToFile function.
CmtNA_ExportLossTable	Saves the loss compensation table into a file.
CmtNA_ImportLossTable	Recalls the loss compensation file. The file must be saved by the CmtNA_ExportLossTable function.
CmtNA_TakePowerCalSweep	Measures the power calibration data for the port using the power meter controlled via USB or USB/GPIB. Calculates calibration coefficients on completion of the measurement, and turns ON the power correction for the port.
CmtNA_PowerSensorZeroing	Executes zeroing procedure of the power sensor. Although the Analyzer automatically turns off the RF power during this procedure, it is recommended to disconnect the power sensor from the analyzer port.
CmtNA_GetPowerCalibrationTable	Gets the power correction array (result of power calibration executed by CmtNA_PerformCalibrationAction function).
CmtNA_SetPowerCalibrationTable	Sets the power correction array (result of power calibration executed by CmtNA_PerformCalibrationAction function).

Name	Description
CmtNA_GetPowerLossCompensationTable	Gets the loss compensation table used when the power calibration is executed by CmtNA_PerformCalibrationAction function.
CmtNA_SetPowerLossCompensationTable	Sets the loss compensation table used when the power calibration is executed by CmtNA_PerformCalibrationAction function.
CmtNA_MeasurePortExtensionShort	Performs measurement of the standard "SHORT" or "OPEN", automatically calculates and sets the parameters of the Port Extension.
CmtNA_MeasurePortExtensionOpen	Performs measurement of the standard "SHORT" or "OPEN", automatically calculates and sets the parameters of the Port Extension.
CmtNA_ExecuteAutoOrientation	Executes the Auto-Orientation procedure of the AutoCal Module. The AutoCal Module must be connected to the ports of Analyzer.
CmtNA_ExecuteAutoCalCalibration	Executes 1-port calibration of the specified port of specified channel or full 2-port calibration between the specified 2 ports of specified channel or one path 2–port calibration between the specified 2 ports of specified channel using the AutoCal module.
CmtNA_ExecuteAutoCalNPortCalibration	Executes 1-port calibration of the specified port of specified channel or full 2, 3, 4-port calibration between the specified 2, 3, 4 ports of

Name	Description
	specified channel or one path 2–port calibration between the specified 2 ports of specified channel using the AutoCal module.
CmtNA_ExecuteAutoCalConfidenceCheck	Executes the confidence check of the calibration coefficients of specified channel using the AutoCal module.
CmtNA_GetMarkerValue	Gets the response value of the marker.
CmtNA_MarkerFunctionExecute	Executes the marker search according to the specified criterion. The type of the marker search is set by CMTNA_ATTR_MARKER_SEARCH_TYPE.
CmtNA_QueryMarkerMathStatistics	Gets the math statistics values.
CmtNA_QueryMarkerMathBandwidth	Gets the bandwidth search result.
CmtNA_QueryMarkerMathFlatness	Gets FLATNESS function data array. The FLATNESS function is applied within the range determined by two markers.
CmtNA_MarkerFunctionsMarkerStart	Sets the value of the specified item to the value of the position of the marker. Sweep start value set to the stimulus value of the marker position.

Name	Description
CmtNA_MarkerFunctionsMarkerStop	Sets the value of the specified item to the value of the position of the marker. Sweep stop value set to the stimulus value of the marker position.
CmtNA_MarkerFunctionsMarkerCenter	Sets the value of the specified item to the value of the position of the marker. Sweep center value set to the stimulus value of the marker position.
CmtNA_MarkerFunctionsMarkerRefValue	Sets the value of the specified item to the value of the position of the marker. Reference value set to the response value of the marker position.
CmtNA_MarkerFunctionsMarkerDelay	Sets the value of the specified item to the value of the position of the marker. Delay value set to the response value of the marker position.
CmtNA_QueryMarkerTableData	Gets the data array of all turned ON markers.
CmtNA_ConfigureTimeDomainStartStop	Configures the start and stop values for the time-domain analysis. This function configures the start and stop points in terms of time.
CmtNA_ConfigureTimeDomainCenterSpan	Configures the start and stop values for the time-domain analysis. This function configures the start and stop points in terms of time.

Name	Description
CmtNA_TimeDomainSetFrequencyLowPass	Changes the frequency range to match with the low-pass type of the time domain transformation function.
CmtNA_ConfigureGatingStartStop	Configures the start and stop values of the gating function. This function configures the center and span points in terms of time.
CmtNA_ConfigureGatingCenterSpan	Configures the center and span values of the gating function. This function configures the center and span points in terms of time.
CmtNA_SetLimitTestData	Sets the data array, which is the limit line in the limit test function.
CmtNA_GetLimitTestData	Gets the data array, which is the limit line in the limit test function.
CmtNA_SaveLimitTable	Saves the limit table into a file.
CmtNA_RestoreLimitTable	Recalls the limit table file. The file must be saved by CmtNA_SaveLimitTable function.
CmtNA_MarkerSetLimitLineResponseOffset	Sets the value of the limit line offset along Y–axis to the active marker value.
CmtNA_GetLimitTestStatus	Gets the limit test result.

Name	Description
CmtNA_GetLimitTestReport	Gets the data array, which is the stimulus values of the measurement points that failed the limit test. The array size is got by CMTNA_ATTR_ANALYSIS_LIMIT_TEST_POINTS.
CmtNA_GetLimitTestReportAll	Gets the data array, which is the limit test result.
CmtNA_SetRippleLimitData	Sets the data array, which is the limit line for the ripple limit function.
CmtNA_GetRippleLimitData	Gets the data array, which is the limit line for the ripple limit function.
CmtNA_SaveRippleLimitTable	Saves the ripple limit table into a file.
CmtNA_RestoreRippleLimitTable	Recalls the ripple limit table file. The file must be saved by CmtNA_SaveRippleLimitTable function.
CmtNA_GetRippleLimitTestStatus	Gets the ripple limit test result.
CmtNA_GetRippleLimitTestReport	Gets the data array, which is the ripple limit test result.
CmtNA_SaveStateToFile	Saves the Analyzer state into a file.
CmtNA_RecallFromFile	Recalls the specified Analyzer state file. The file must be saved by CmtNA_SaveStateToFile function.

Name	Description
CmtNA_SaveChannelToRegister	Saves the Analyzer state of the items set for the active channel into one of the four memory registers.
CmtNA_ClearRegisterStates	Clears the memory of the channel state saved by CmtNA_SaveChannelToRegister function.
CmtNA_RecallChannelFromRegister	Recalls the Analyzer state for the active channel. The file must be saved in one of the four memory registers by CmtNA_SaveChannelToRegister function.
CmtNA_SaveTraceData	Saves the CSV formatted data into a file.
CmtNA_SetTouchstoneFileType	Sets the Touchstone file type and the port numbers, when saving S–parameters by CmtNA_SaveTouchstoneFile function.
CmtNA_GetTouchstoneFileType	Gets the Touchstone file type and the port numbers, when saving S–parameters by CmtNA_SaveTouchstoneFile function.
CmtNA_SaveTouchstoneFile	Saves the measured S–parameters of the active channel into a Touchstone file. The file type (1 port to 4 port) is set by CmtNA_SetTouchstoneFileType function.
CmtNA_LoadTouchstoneFile	Loads the Touchstone file with the specified name to the measured S–parameters of the active channel. The Touchstone file types 1, 2, 3

Name	Description
	or 4 port (file extensions s1p, s2p, s3p or s4p) are supported. On completion of the command, the channel goes to the hold state.
CmtNA_SystemPreset	Resets the Analyzer to the factory settings.
CmtNA_PrintOut	Prints out the image displayed on the screen without previewing.
CmtNA_AbortPrint	Aborts the printout.
CmtNA_SaveImage	Saves the display image in BMP or PNG format into a file.
CmtNA_TestBeepComplete	Generates a beep to notify of the completion of the operation.
CmtNA_TestBeepWarning	Generates a beep to notify of warning.
CmtNA_SystemShow	Restores the analyzer main window hidden by CmtNA_SystemHide function.
CmtNA_SystemHide	Minimizes the analyzer main window removing it from desktop.

Attributes

Name	Data Type	Description
CMTNA_ATTR_STIMULUS_FREQUENCY_START	ViReal64	Sets/Gets the start value of a frequency sweep range. The value is expressed in hertz (Hz).
CMTNA_ATTR_STIMULUS_FREQUENCY_STOP	ViReal64	Sets/Gets the stop value of a frequency sweep range.The value is expressed in hertz (Hz).
CMTNA_ATTR_STIMULUS_FREQUENCY_CENTER	ViReal64	Sets/Gets the center value of a frequency sweep range. The value is expressed in hertz (Hz).
CMTNA_ATTR_STIMULUS_FREQUENCY_SPAN	ViReal64	Sets/Gets the span value of a frequency sweep range. The value is expressed in hertz (Hz).
CMTNA_ATTR_STIMULUS_POINTS	ViInt32	Sets/Gets the number of sweep points.
CMTNA_ATTR_STIMULUS_SWEEP_TYPE	ViInt32	Sets/Gets the sweep type of channel.
CMTNA_ATTR_STIMULUS_IF_BANDWIDTH	ViReal64	Sets/Gets the bandwidth of the digital. IF filter to be used in the measurement. IF Bandwidth in Hz. The list of valid IF Bandwidths is different depending on the analyzer model. If an invalid number is specified, the analyzer will round up to the closest valid number. The value is expressed in hertz (Hz).

Name	Data Type	Description
CMTNA_ATTR_STIMULUS_OUTPUT_POWER	ViReal64	Sets/Gets the power level for the frequency sweep type. This value is expressed in decibels above 1 milliwatt (dBm).
CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT_COUPLE	ViBoolean	Turns ON/OFF the port power couple.
CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT	ViReal64	Sets/Gets the power level of port for the frequency sweep type when the port couple feature is set to OFF. This value is expressed in decibels above 1 milliwatt (dBm).
CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE	ViReal64	Sets/Gets the power slope value for the frequency sweep. This value is expressed in decibels/gigahertz (dB/GHz).
CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE_STATE	ViBoolean	Turns ON/OFF the power slope. The power slope is valid for the frequency sweep type: Linear, Logarithmic, Segment.
CMTNA_ATTR_STIMULUS_OUTPUT_POWER_RFOUT	ViBoolean	Turns ON/OFF the RF signal output. Measurements cannot be performed when the RF signal output is turned OFF.
CMTNA_ATTR_STIMULUS_POWER_CW_FREQUENCY	ViReal64	Sets/Gets the fixed frequency value when the power sweep type selected. The value is expressed in hertz (Hz).
CMTNA_ATTR_STIMULUS_POWER_START	ViReal64	Sets/Gets the power sweep start value when the power sweep type is active. The value is expressed in decibels above 1 milliwatt (dBm).

Name	Data Type	Description
CMTNA_ATTR_STIMULUS_POWER_STOP	ViReal64	Sets/Gets the power sweep stop value when the power sweep type is active.The value is expressed in decibels above 1 milliwatt (dBm).
CMTNA_ATTR_STIMULUS_POWER_CENTER	ViReal64	Sets/Gets the center value of the power sweep type. The value is expressed in decibels above 1 milliwatt (dBm).
CMTNA_ATTR_STIMULUS_POWER_SPAN	ViReal64	Sets/Gets the power span when the power sweep type is active. The value is expressed in decibels above 1 milliwatt (dBm).
CMTNA_ATTR_STIMULUS_SEGMENT_DISPLAY_ORDER	ViInt32	Sets/Gets the display method of the graph horizontal axis for the segment sweep.
CMTNA_ATTR_STIMULUS_SWEEP_MEASURE_DELAY	ViReal64	Sets/Gets the delay before measurement in each measurement point. The value is expressed in seconds (sec).
CMTNA_ATTR_STIMULUS_TRIGGER_MODE	ViInt32	Sets/Gets the trigger mode.
CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE	ViInt32	Sets/Gets the trigger source.
CMTNA_ATTR_STIMULUS_TRIGGER_STATE	ViInt32	Gets the the current state of the analyzer.

Name	Data Type	Description
CMTNA_ATTR_STIMULUS_EXT_TRIGGER_EVENT	ViInt32	Turns ON/OFF the point trigger feature for external trigger source.
CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POLARITY	ViInt32	Sets/Gets out the polarity of the external trigger.
CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POSITION	ViInt32	Sets/Gets the position of the external trigger.
CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT	ViBoolean	Turns ON/OFF the trigger output.
CMTNA_ATTR_STIMULUS_EXT_TRIGGER_DELAY	ViReal64	Sets/Gets the response delay with respect to the external trigger signal. The value is expressed in seconds (sec).
CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_POLARITY	ViInt32	Sets/Gets the polarity of the trigger output.
CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_FUNCTION	ViInt32	Sets/Gets the trigger output function. The trigger output outputs various waveforms depending on the setting of the Output Trigger Function.
CMTNA_ATTR_STIMULUS_TRIGGER_SCOPE	ViInt32	Sets/Gets the trigger scope. The trigger scope determines the response on the trigger signal arrival: either starts a sweep of all waiting channels in turn or starts a sweep in the active channel only.

Name	Data Type	Description
CMTNA_ATTR_STIMULUS_FREQUENCY_OFFSET	ViBoolean	Turns ON/OFF the frequency offset feature.
CMTNA_ATTR_FREQUENCY_OFFSET_TYPE	ViInt32	Sets/Gets the frequency offset type when the frequency offset feature is ON.
CMTNA_ATTR_FREQUENCY_MULTIPLIER	ViReal64	Sets/Gets the basic frequency range multiplier of port when the frequency offset feature is ON and offset type is "PORT".
CMTNA_ATTR_FREQUENCY_DIVIDER	ViReal64	Sets/Gets the basic frequency range divisor to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_OFFSET	ViReal64	Sets/Gets the basic frequency range offset to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).
CMTNA_ATTR_FREQUENCY_OFFSET_START	ViReal64	Sets/Gets the frequency sweep start of the receivers when the frequency offset feature is ON and offset type is "SRCRcv".The value is expressed in hertz (Hz).
CMTNA_ATTR_FREQUENCY_OFFSET_STOP	ViReal64	Sets/Gets the frequency sweep stop of the receivers when the frequency offset feature is ON and offset type is "SRCRcv".The value is expressed in hertz (Hz).

Name	Data Type	Description
CMTNA_ATTR_FREQUENCY_RECEIVER_MULTIPLIER	ViReal64	Sets/Gets the basic frequency range multiplier to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_RECEIVER_DIVIDER	ViReal64	Sets/Gets the basic frequency range divisor to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET	ViReal64	Sets/Gets the basic frequency range offset to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_START	ViReal64	Sets/Gets the frequency sweep start of the receivers when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_STOP	ViReal64	Sets/Gets the frequency sweep stop of the receivers when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_SOURCE_MULTIPLIER	ViReal64	Sets/Gets the basic frequency range multiplier to get the source frequency when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_SOURCE_DIVIDER	ViReal64	Sets/Gets the basic frequency range divisor to get the source frequency when the frequency offset feature is ON and offset type is "SRCRcv".

Name	Data Type	Description
CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET	ViReal64	Sets/Gets the basic frequency range offset to get the source frequency when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_START	ViReal64	Sets/Gets the frequency sweep start of the source when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_STOP	ViReal64	Sets/Gets the frequency sweep stop of the source when the frequency offset feature is ON and offset type is "SRCRcv".
CMTNA_ATTR_STIMULUS_MAX_POINTS	ViInt32	Gets the maximum number of the measurement points.
CMTNA_ATTR_STIMULUS_MAX_FREQUENCY	ViReal64	Gets the upper limit of the measurement frequency. The value is expressed in hertz (Hz).
CMTNA_ATTR_STIMULUS_MIN_FREQUENCY	ViReal64	Gets the lower frequency of the measurement frequency. The value is expressed in hertz (Hz).
CMTNA_ATTR_EXTERNAL_TRIGGER_ROUTE	ViInt32	Sets/Gets the connector to use for the external trigger input in a PXI system (command valid for PXIe-S5090 model only).
CMTNA_ATTR_OUTPUT_TRIGGER_ROUTE	ViInt32	Sets/Gets the connector to use for the trigger output in a PXI system (command valid for PXIe-S5090 model only).

Name	Data Type	Description
CMTNA_ATTR_MEASUREMENT_FORMAT	ViInt32	Sets/Gets the display format specified by Measurement Format Enum for the measurement.
CMTNA_ATTR_MEASUREMENT_SCALE_DIV	ViReal64	Sets/Gets the trace scale.
CMTNA_ATTR_MEASUREMENT_REF_VALUE	ViReal64	Sets/Gets the value of the reference line (response value on the reference line). For the rectangular format only. The value is depending on the format.
CMTNA_ATTR_MEASUREMENT_REF_POSITION	ViInt32	Sets/Gets the position of the reference line. For the rectangular format only.
CMTNA_ATTR_MEASUREMENT_DIVISIONS	ViInt32	Sets/Sets the number of the vertical scale divisions. For the rectangular format only.
CMTNA_ATTR_MEASUREMENT_ELEC_DELAY	ViReal64	Sets/Gets the value of the electrical delay. The value is expressed in seconds (sec).
CMTNA_ATTR_MEASUREMENT_PHASE_OFFSET	ViReal64	Sets/Gets the value of the phase offset. The value is expressed in degrees (°).
CMTNA_ATTR_MEASUREMENT_AVERAGING	ViBoolean	Turns ON/OFF the measurement averaging function.

Name	Data Type	Description
CMTNA_ATTR_MEASUREMENT_AVERAGING_FACTOR	ViInt32	Sets/Gets the averaging factor, when the averaging function is turned on.
CMTNA_ATTR_MEASUREMENT_AVG_TRIGGER	ViBoolean	Turns ON/OFF the averaging trigger function. The function executes a sweep the number of times specified by the averaging factor with a single trigger for the channels with the averaging enabled.
CMTNA_ATTR_MEASUREMENT_SMOOTHING	ViBoolean	Turns ON/OFF the trace smoothing.
CMTNA_ATTR_MEASUREMENT_SMOOTHING_APERTURE	ViReal64	Sets/Gets the smoothing aperture, when performing smoothing function. The value is expressed in percentages (%).
CMTNA_ATTR_FUNCTION_POINTS	ViInt32	Gets the number of points (data pairs) of the analysis result by CmtNA_FunctionExecute function.
CMTNA_ATTR_FUNCTION_DOMAIN_COUPLING	ViBoolean	If the arbitrary range is turned ON by CMTNA_ATTR_FUNCTION_DOMAIN_STATE, specifies whether all traces of channel use the same range (coupling) or each trace uses individual range when CmtNA_FunctionExecute function is executed.
CMTNA_ATTR_FUNCTION_DOMAIN_STATE	ViBoolean	Specifies whether an arbitrary range or the entire sweep range is used when CmtNA_FunctionExecute function is executed.

Name	Data Type	Description
CMTNA_ATTR_FUNCTION_DOMAIN_START	ViReal64	Sets/Gets the start value of the analysis range of CmtNA_FunctionExecute function.
CMTNA_ATTR_FUNCTION_DOMAIN_STOP	ViReal64	Sets/Gets the time domain stop value, when the time domain transformation function is turned ON. The value is expressed in seconds (sec).
CMTNA_ATTR_FUNCTION_PEAK_EXCURSION	ViReal64	Sets/Gets the lower limit for the peak excursion value when executing the peak search with CmtNA_FunctionExecute function.
CMTNA_ATTR_FUNCTION_PEAK_POLARITY	ViInt32	Sets/Gets the polarity when performing the peak search with CmtNA_FunctionExecute function.
CMTNA_ATTR_FUNCTION_TARGET_LEVEL	ViReal64	Sets/Gets the target level when performing the search for the trace and the target level crosspoints with CmtNA_FunctionExecute function.
CMTNA_ATTR_FUNCTION_TRANSITION_TYPE	ViInt32	Sets/Gets the transition type when performing the search for the trace and the target level crosspoints with CmtNA_FunctionExecute function.
CMTNA_ATTR_FUNCTION_TYPE	ViInt32	Sets/Gets the type of analysis executed with CmtNA_FunctionExecute function.
CMTNA_ATTR_DISPLAY_ACTIVE_TRACE	ViInt32	Sets/Gets the active trace in channel.

Name	Data Type	Description
CMTNA_ATTR_DISPLAY_ACTIVE_CHANNEL	ViInt32	Sets/Gets the active channel.
CMTNA_ATTR_DISPLAY_CHANNEL_ALLOCATION	ViInt32	Sets/Gets the layout of the channel windows on the screen.
CMTNA_ATTR_DISPLAY_TRACE_ALLOCATION	ViInt32	Sets/Gets the layout of the graph in the channel window.
CMTNA_ATTR_DISPLAY_TRACE_COUNT	ViInt32	Sets/Gets the number of traces in the channel.
CMTNA_ATTR_DISPLAY_MAXIMIZE_CHANNEL	ViBoolean	Turns ON/OFF of the maximization of the active channel window.
CMTNA_ATTR_DISPLAY_MAXIMIZE_TRACE	ViBoolean	Turn ON/OFF the active trace maximization inside the specified channel.
CMTNA_ATTR_DISPLAY_TRACE_TYPE	ViInt32	Turns ON/OFF the memory trace display.
CMTNA_ATTR_DISPLAY_TRACE_DATA_MATH	ViInt32	Sets/Gets the math operation between the data trace and the memory trace. The math result replaces the data trace. If the memory trace does not exist, the command is ignored.

Name	Data Type	Description
CMTNA_ATTR_DISPLAY_TRACE_HOLD_TYPE	ViInt32	Sets/Gets the type of the trace hold function. The function holds the trace at the maximum or minimum point.
CMTNA_ATTR_DISPLAY_DATA_TRACE_COLOR	ViInt32	Sets/Gets the data trace color.
CMTNA_ATTR_DISPLAY_MEMORY_TRACE_COLOR	ViInt32	Sets/Gets the memory trace color.
CMTNA_ATTR_DISPLAY_BACKGROUND_COLOR	ViInt32	Sets/Gets the background color for trace display.
CMTNA_ATTR_DISPLAY_GRID_COLOR	ViInt32	Sets/Gets the grid and the graticule label color for trace display.
CMTNA_ATTR_DISPLAY_INVERT_COLOR	ViBoolean	Turns ON/OFF the inversion of display colors of the traces area.
CMTNA_ATTR_DISPLAY_SYSTEM_DATE	ViString	Sets/Gets the current date.
CMTNA_ATTR_DISPLAY_SYSTEM_TIME	ViString	Sets/Gets the current time.

Name	Data Type	Description
CMTNA_ATTR_DISPLAY_CYCLE_TIME_VALUE	ViReal64	Gets the measured cycle time. The cycle time is the interval between the start of two adjacent sweeps. The value is expressed in seconds (sec).
CMTNA_ATTR_DISPLAY_TITLE_LABEL	ViBoolean	Turns ON/OFF the channel title display.
CMTNA_ATTR_DISPLAY_TITLE_DATA	ViString	Sets/Gets the channel title label.
CMTNA_ATTR_DISPLAY_UPDATE	ViBoolean	Turns ON/OFF the display update.
CMTNA_ATTR_CALIBRATION_CORRECTION	ViBoolean	Turns ON/OFF the S-parameter error correction.
CMTNA_ATTR_CALIBRATION_TYPE	ViInt32	Gets the calibration type for the calculation of the calibration coefficients on completion of the calibration executed by CmtNA_ApplyCalibration function.
CMTNA_ATTR_CORRECTION_TYPE	ViString	Gets the applied calibration type and the port numbers for the specified trace.

Name	Data Type	Description
CMTNA_ATTR_CORRECTION_STATUS	ViString	Gets the interpolation/extrapolation status of the error correction.
CMTNA_ATTR_ADAPTER_REMOVAL_PORT	ViInt32	Sets the port number and sets the adapter removal/insertion function for the calculation of the calibration coefficients when CmtNA_ApplyCalibration function has been executed.
CMTNA_ATTR_ADAPTER_REMOVAL_DELAY	ViReal64	Sets/Gets the approximate delay value of an adapter in the adapter removal/insertion function. The value is expressed in seconds (sec).
CMTNA_ATTR_ADAPTER_REMOVAL_LENGTH	ViReal64	Sets/Gets the approximate value of the mechanical length of the adapter in the adapter removal/insertion function. The value is expressed in meters (m).
CMTNA_ATTR_ADAPTER_REMOVAL_MEDIA	ViInt32	Sets/Gets the adapter media in the adapter removal/insertion function.
CMTNA_ATTR_ADAPTER_REMOVAL_UNIT	ViInt32	Sets/Gets the display units of the adapter delay (length) in the adapter removal/insertion function.
CMTNA_ATTR_ADAPTER_REMOVAL_PERMITTIVITY	ViReal64	Sets/Gets the value of the permittivity of an adapter media in the adapter removal/insertion function.

Name	Data Type	Description
CMTNA_ATTR_ADAPTER_REMOVAL_CUTOFF_FREQ	ViReal64	Sets/Gets the value of the cutoff frequency of the waveguide adapter.
CMTNA_ATTR_THRU_ADDITION_DELAY	ViReal64	Sets/Gets the approximate delay value of an unknown thru in the thru addition function. The value is expressed in seconds (sec).
CMTNA_ATTR_THRU_ADDITION_LENGTH	ViReal64	Sets/Gets the approximate value of the mechanical length of an unknown thru in the thru addition function. The value is expressed in meters (m).
CMTNA_ATTR_THRU_ADDITION_UNIT	ViInt32	Sets/Gets the display units of the thru delay (length) in the thru addition function.
CMTNA_ATTR_THRU_ADDITION_MEDIA	ViInt32	Sets/Gets the media of the thru in the thru addition function.
CMTNA_ATTR_THRU_ADDITION_PERMITTIVITY	ViReal64	Sets/Gets the value of the permittivity of the thru media in the thru addition function.
CMTNA_ATTR_THRU_ADDITION_CUTOFF_FREQ	ViReal64	Sets/Gets the value of the cutoff frequency of the waveguide thru in the thru addition function.
CMTNA_ATTR_CALKIT_SELECTED	ViInt32	Sets/Gets the number of the selected calibration kit in the table of calibration kits. The selected calibration kit is used in the subsequent

Name	Data Type	Description
		calibration and is used for editing by CMTNA_ATTR_CALKIT_XXX.
CMTNA_ATTR_CALKIT_LABEL	ViString	Sets/Gets the calibration kit label.
CMTNA_ATTR_CALKIT_DESCRIPTION	ViString	Sets/Gets the calibration kit description string.
CMTNA_ATTR_CALKIT_STANDARD_TYPE	ViInt32	Sets/Gets the type of calibration standard.
CMTNA_ATTR_CALKIT_STANDARD_LABEL	ViString	Sets/Gets the label for the calibration standard.
CMTNA_ATTR_CALKIT_STANDARD_MIN_FREQUENCY	ViReal64	Sets/Gets the minimum frequency limit of the calibration standard. The value is expressed in hertz (Hz).
CMTNA_ATTR_CALKIT_STANDARD_MAX_FREQUENCY	ViReal64	Sets/Gets the maximum frequency limit of the calibration standard. The value is expressed in hertz (Hz).
CMTNA_ATTR_CALKIT_STANDARD_OFFSET_DELAY	ViReal64	Sets/Gets the offset delay value for the calibration standard. The value is expressed in seconds (sec).
CMTNA_ATTR_CALKIT_STANDARD_OFFSET_Z0	ViReal64	Sets/Gets the offset Z0 value for the calibration standard. The value is expressed in Ohm (Ω).

Name	Data Type	Description
CMTNA_ATTR_CALKIT_STANDARD_OFFSET_LOSS	ViReal64	Sets/Gets the offset loss value for the calibration standard. The value is expressed in Ohm/seconds (Ω/s).
CMTNA_ATTR_CALKIT_STANDARD_ARBITRARY	ViReal64	Sets/Gets the value of the arbitrary impedance for the load standard. The value is expressed in Ohm (Ω).
CMTNA_ATTR_CALKIT_STANDARD_C0	ViReal64	Sets/Gets the C0 value for the open calibration standard. The value is expressed in Farad (F).
CMTNA_ATTR_CALKIT_STANDARD_C1	ViReal64	Sets/Gets the C1 value for the open calibration standard. The value is expressed in Farad/Hertz (F/Hz).
CMTNA_ATTR_CALKIT_STANDARD_C2	ViReal64	Sets/Gets the C2 value for the open calibration standard. The value is expressed in Farad/Hertz2 (F/Hz2).
CMTNA_ATTR_CALKIT_STANDARD_C3	ViReal64	Sets/Gets the C3 value for the open calibration standard. The value is expressed in Farad/Hertz3 (F/Hz3).
CMTNA_ATTR_CALKIT_STANDARD_L0	ViReal64	Sets/Gets the L0 value for the short calibration standard. The value is expressed in Henry (H).
CMTNA_ATTR_CALKIT_STANDARD_L1	ViReal64	Sets/Gets the L1 value for the short calibration standard. The value is expressed in Henry/Hertz (H/Hz).

Name	Data Type	Description
CMTNA_ATTR_CALKIT_STANDARD_L2	ViReal64	Sets/Gets the L2 value for the short calibration standard. The value is expressed in Henry/Hertz2 (H/Hz2).
CMTNA_ATTR_CALKIT_STANDARD_L3	ViReal64	Sets/Gets the L3 value for the short calibration standard. The value is expressed in Henry/Hertz3 (H/Hz3).
CMTNA_ATTR_POWER_CALIBRATION_CORRECTION	ViBoolean	Turns ON/OFF the power correction.
CMTNA_ATTR_POWER_CALIBRATION_LOSS_COMPENSATION	ViBoolean	Turns ON/OFF the state of the loss compensation used when the power calibration is executed by CmtNA_TakePowerCalSweep function.
CMTNA_ATTR_PORT_EXTENSIONS	ViBoolean	Turns ON/OFF the port extension function.
CMTNA_ATTR_PORT_EXTENSIONS_TIME	ViReal64	Sets/Gets the electrical delay value for the port extension function. The value is expressed in seconds (sec).
CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_STATE	ViBoolean	Turns ON/OFF the loss compensation of the loss 1 and loss 2 for the port extension function.
CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_VALUE	ViReal64	Sets/Gets the values of the loss 1 for the port extension function. The value is expressed in decibels (dB).

Name	Data Type	Description
CMTNA_ATTR_PORT_EXTENSIONS_FREQ1_VALUE	ViReal64	Sets/Gets the values of the frequency 1 to calculate the loss for the port extension function. The value is expressed in Hertz (Hz).
CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_STATE	ViBoolean	Turns ON/OFF the loss compensation of the loss 1 and loss 2 for the port extension function.
CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_VALUE	ViReal64	Sets/Gets the values of the loss 2 for the port extension function. The value is expressed in decibels (dB).
CMTNA_ATTR_PORT_EXTENSIONS_FREQ2_VALUE	ViReal64	Sets/Gets the values of the frequency 2 to calculate the loss for the port extension function. The value is expressed in Hertz (Hz).
CMTNA_ATTR_PORT_EXTENSIONS_LDC_VALUE	ViReal64	Sets/Gets the loss value at DC for the port extension function. The value is expressed in decibels (dB).
CMTNA_ATTR_AUTO_PORT_EXTENSION_METHOD	ViInt32	Sets/Gets the frequency range used for calculation of the results of the Auto Port Extension function.
CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_START	ViReal64	Sets/Gets the start value of the user span of the auto port extension function.
CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_STOP	ViReal64	Sets/Gets the stop value of the user span of the auto port extension function.

Name	Data Type	Description
CMTNA_ATTR_AUTO_PORT_EXTENSION_INCLUDE_LOSS	ViBoolean	Turns ON/OFF the usage of "Loss1"and "Loss2" values for the results of the auto port extension function.
CMTNA_ATTR_AUTO_PORT_EXTENSION_ADJUST_MISMATCH	ViBoolean	Turns ON/OFF the usage of "Loss at DC" value for the results of the auto port extension function.
CMTNA_ATTR_CALIBRATION_TRIGGER_SOURCE	ViInt32	Enables/Disables the internal trigger source for calibration.
CMTNA_ATTR_CALIBRATION_PORT_Z0	ViReal64	Sets/Gets the system impedance Z0 or the the impedance Z0 of port. The value is expressed in Ohm (Ω).
CMTNA_ATTR_CALIBRATION_AUTO_SELECT_Z0	ViBoolean	Turns ON/OFF the auto-select Z0 function. When enabled the function sets the port impedance Z0 to the corresponding value of measuring calibration standard.
CMTNA_ATTR_AUTOCAL_CHARACTERIZATION	ViInt32	Sets/Gets the characterization number used when executing AutoCal (factory or user characterizations).
CMTNA_ATTR_AUTOCAL_AUTO_ORIENTATION	ViBoolean	Turns ON/OFF the Auto-Orientation function when the AutoCal Module calibration is executed.
CMTNA_ATTR_AUTOCAL_ORIENTATION_PORT	ViInt32	Sets/Gets the AutoCal module port number which is connected to a specified port of Network Analyzer.

Name	Data Type	Description
CMTNA_ATTR_AUTOCAL_UNKNOWN_THRU	ViBoolean	Turns ON/OFF the Unknown Thru feature when the AutoCal Module calibration is executed.
CMTNA_ATTR_AUTOCAL_TEMPERATURE	ViReal64	Gets the temperature of the AutoCal module connected to the Analyzer. The value is expressed in Celsius degrees (°C).
CMTNA_ATTR_REFERENCE_MARKER	ViBoolean	Turns ON/OFF the reference marker. When the reference marker is turned ON, all the values of the other markers turn to relative values.
CMTNA_ATTR_MARKERS_COUNT	ViInt32	Sets/Gets the number of the turned ON markers.
CMTNA_ATTR_SELECTED_MARKER	ViInt32	Sets the active marker.
CMTNA_ATTR_MARKER_STIMULUS	ViReal64	Sets/Gets the stimulus value of the marker.
CMTNA_ATTR_MARKER_SEARCH_TYPE	ViInt32	Sets/Gets the type of the marker search, which is performed by CmtNA_MarkerFunctionExecute function.
CMTNA_ATTR_MARKER_SEARCH_PEAK_EXCURSION	ViReal64	Sets/Gets the peak excursion value, when the marker search for peak is performed by CmtNA_MarkerFunctionExecute function.

Name	Data Type	Description
CMTNA_ATTR_MARKER_SEARCH_PEAK_POLARITY	ViInt32	Sets/Gets the peak polarity, when the marker search for peak is performed by CmtNA_MarkerFunctionExecute function.
CMTNA_ATTR_MARKER_SEARCH_TARGET_VALUE	ViReal64	Sets/Gets the target value, when the marker search for target is performed by CmtNA_MarkerFunctionExecute function.
CMTNA_ATTR_MARKER_SEARCH_TARGET_TRANSITION	ViInt32	Sets/Gets the type of the target transition, when the marker search for transition is performed by by CmtNA_MarkerFunctionExecute function.
CMTNA_ATTR_MARKER_SEARCH_TRACKING	ViBoolean	Turns ON/OFF the marker search tracking.
CMTNA_ATTR_MARKER_SEARCH_RANGE	ViBoolean	Specifies whether an arbitrary range or the entire sweep range is used when the marker search is performed.
CMTNA_ATTR_MARKER_SEARCH_START	ViReal64	Sets/Gets out the start value of the marker search range.
CMTNA_ATTR_MARKER_SEARCH_STOP	ViReal64	Sets/Gets out the stop value of the marker search range.
CMTNA_ATTR_MARKER_SEARCH_COUPLE	ViBoolean	If the arbitrary search range turned ON by CMTNA_ATTR_MARKER_SEARCH_RANGE, specifies whether all

Name	Data Type	Description
		traces of channel use the same range (coupling) or each trace uses individual range when the marker search is performed.
CMTNA_ATTR_MARKER_MATH_STATISTICS	ViBoolean	Turns ON/OFF the math statistics display.
CMTNA_ATTR_MARKER_MATH_STATISTIC_RANGE	ViBoolean	Sets/Gets either partial frequency range or entire frequency range is used for math statistic calculation. The partial frequency range is limited by two markers.
CMTNA_ATTR_MARKER_MATH_STATISTIC_START	ViInt32	Sets/Gets the number of the marker, which specifies the start frequency of the math statistics range.
CMTNA_ATTR_MARKER_MATH_STATISTIC_STOP	ViInt32	Sets/Gets the number of the marker, which specifies the stop frequency of the math statistics range.
CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH	ViBoolean	Turns ON/OFF the bandwidth search function.
CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_TYPE	ViInt32	Sets/Gets the type of the bandwidth search function.
CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_REF	ViInt32	Selects the reference point for the bandwidth search function: reference marker or absolute maximum value of the trace.

Name	Data Type	Description
CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_VALUE	ViReal64	Sets/Gets the bandwidth definition value.
CMTNA_ATTR_MARKER_MATH_FLATNESS	ViBoolean	Turns ON/OFF the marker FLATNESS function.
CMTNA_ATTR_MARKER_MATH_FLATNESS_START	ViInt32	Sets/Gets the number of the marker, which specifies the start frequency of the FLATNESS function domain.
CMTNA_ATTR_MARKER_MATH_FLATNESS_STOP	ViInt32	Sets/Gets the number of the marker, which specifies the stop frequency of the FLATNESS function domain.
CMTNA_ATTR_MARKER_PROPERTIES_DISCRETE	ViBoolean	Turns ON/OFF the marker discrete mode.
CMTNA_ATTR_MARKER_PROPERTIES_MARKER_COUPLE	ViBoolean	Turns ON/OFF the marker coupling between traces. When coupled the markers of different traces with same number track the X-axis position.
CMTNA_ATTR_MARKER_PROPERTIES_MARKER_TABLE	ViBoolean	Turns ON/OFF of the marker table.
CMTNA_ATTR_MARKER_PROPERTIES_MARKER_ACTIVE_ONLY	ViBoolean	Sets/Gets display either the active trace markers or the all trace markers.

Name	Data Type	Description
CMTNA_ATTR_MARKER_PROPERTIES_ALIGN	ViInt32	Sets/Gets the alignment mode of the marker display position of each trace, when the only active trace display feature is turned OFF by CMTNA_ATTR_MARKER_PROPERTIES_MARKER_ACTIVE_ONLY.
CMTNA_ATTR_MARKER_PROPERTIES_DATA_X_POSITION	ViInt32	Sets/Gets the display position of the marker annotation on the X–axis by a percentage of the display width. The value is expressed in percentages (%).
CMTNA_ATTR_MARKER_PROPERTIES_DATA_Y_POSITION	ViInt32	Sets/Gets the display position of the marker annotation on the Y–axis by a percentage of the display height. The value is expressed in percentages (%).
CMTNA_ATTR_ANALYSIS_FIXTURE_SIMULATOR	ViBoolean	Turns ON/OFF the fixture simulation function.
CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION	ViBoolean	Turns ON/OFF the port impedance conversion function.
CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_Z0	ViReal64	Sets/Gets the value of the impedance for port impedance conversion function.
CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_REAL	ViReal64	Sets/ Gets the real part of the impedance of the port impedance conversion function. The value is expressed in Ohm (Ω).

Name	Data Type	Description
CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_IMAG	ViReal64	Sets/ Gets the imaginary part of the impedance of the port impedance conversion function. The value is expressed in Ohm (Ω).
CMTNA_ATTR_ANALYSIS_DEEMBEDDING	ViBoolean	Turns ON/OFF the 2-port network de–embedding function.
CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT	ViBoolean	Turns ON/OFF the 2-port network de–embedding function for specified port.
CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT_FILE	ViString	Sets/Gets the name of *.s2p file of the de–embedded circuit of the 2-port network de–embedding function. The file contains the circuit S–parameters in Touchstone format.
CMTNA_ATTR_ANALYSIS_EMBEDDING	ViBoolean	Turns ON/OFF the 2-port network embedding function.
CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT	ViBoolean	Turns ON/OFF the 2-port network embedding function for each port.
CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT_FILE	ViString	Sets/Gets the name of *.s2p file of the embedded circuit of the 2-port network embedding function. The file contains the circuit S–parameters in Touchstone format.

Name	Data Type	Description
CMTNA_ATTR_ANALYSIS_TIME_DOMAIN	ViBoolean	Turns ON/OFF the time domain transformation function.
CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_UNITS	ViInt32	Sets/Gets the the transformation unit for the time domain transformation function: seconds, meters, feet.
CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_REFLECTION_TYPE	ViInt32	Sets/Gets the reflection distance either one way or round trip for the time domain transformation function.
CMTNA_ATTR_TIME_DOMAIN_CABLE_CORRECTION	ViBoolean	Turns ON/OFF the cable correction when the time domain transformation function is turned ON.
CMTNA_ATTR_TIME_DOMAIN_CABLE_VELOCITY_FACTOR	ViReal64	Sets/Gets the cable relative wave speed velocity for the cable correction function, when the time domain transformation function is turned ON.
CMTNA_ATTR_TIME_DOMAIN_CABLE_LOSS	ViReal64	Sets/Gets the cable loss value for the cable correction function, when the time domain transformation function is turned ON. The value is expressed in decibell/meter (dB/m).
CMTNA_ATTR_TIME_DOMAIN_CABLE_FREQUENCY	ViReal64	Sets/Gets the frequency value at which the cable loss specified for the cable correction function, when the time domain transformation function is turned ON. The value is expressed in Hertz (Hz).

Name	Data Type	Description
CMTNA_ATTR_TIME_DOMAIN_START	ViReal64	Sets/Gets the time domain start value, when the time domain transformation function is turned ON. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_STOP	ViReal64	Sets/Gets the time domain stop value, when the time domain transformation function is turned ON. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_CENTER	ViReal64	Sets/Gets the time domain center value, when the time domain transformation function is turned ON. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_SPAN	ViReal64	Sets/Gets the time domain span value, when the time domain transformation function is turned ON. The value is expressed in second (sec).
CMTNA_ATTR_TIME_DOMAIN_TRANSFORM_TYPE	ViInt32	Sets/Gets the stimulus type for the time domain transformation function.
CMTNA_ATTR_TIME_DOMAIN_WINDOW_SHAPE	ViInt32	Sets/Gets the window shape the Kaiser–Bessel window shape, when performing time domain transformation.
CMTNA_ATTR_TIME_DOMAIN_IMPULSE_WIDTH	ViReal64	Sets/Gets the impulse width (time domain transformation resolution), coupled with the Kaiser–Bessel window shape β parameter. The impulse width setting changes the β parameter, and setting of β

Name	Data Type	Description
		parameter changes the impulse width. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_KAISER_BETA	ViReal64	Sets/Gets the β parameter, which controls the Kaiser–Bessel window shape, when performing time domain transformation.
CMTNA_ATTR_TIME_DOMAIN_GATING	ViBoolean	Turns ON/OFF the gating function.
CMTNA_ATTR_TIME_DOMAIN_GATING_START	ViReal64	Sets/Gets the gate start value of the gating function. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_GATING_STOP	ViReal64	Sets/Gets the gate stop value of the gating function. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_GATING_CENTER	ViReal64	Sets/Gets the gate center value of the gating function. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_GATING_SPAN	ViReal64	Sets/Gets the gate span value of the gating function. The value is expressed in seconds (sec).
CMTNA_ATTR_TIME_DOMAIN_GATING_TYPE	ViInt32	Sets/Gets the gate type of the gating function.

Name	Data Type	Description
CMTNA_ATTR_TIME_DOMAIN_GATING_SHAPE	ViInt32	Sets/Gets the gate shape of the gating function.
CMTNA_ATTR_ANALYSIS_CONVERSION	ViBoolean	Turns ON/OFF the S–parameter conversion function.
CMTNA_ATTR_ANALYSIS_CONVERSION_FUNCTION	ViInt32	Sets/Gets the S–parameter conversion function type.
CMTNA_ATTR_ANALYSIS_LIMIT_TEST	ViBoolean	Turns ON/OFF the limit test.
CMTNA_ATTR_ANALYSIS_LIMIT_LINE_DISPLAY	ViBoolean	Turns ON/OFF the limit line display of the limit test function.
CMTNA_ATTR_LIMIT_LINE_STIMULUS_OFFSET	ViReal64	Sets/Gets the value of the limit line offset along X–axis.
CMTNA_ATTR_LIMIT_LINE_RESPONSE_OFFSET	ViReal64	Sets/Gets the value of the limit line offset along Y–axis.
CMTNA_ATTR_ANALYSIS_LIMIT_TEST_FAIL_SIGN	ViBoolean	Turns ON/OFF the "Fail" sign display, when performing limit test.

Name	Data Type	Description
CMTNA_ATTR_ANALYSIS_LIMIT_TEST_POINTS	ViInt32	Gets the number of the measurement points that failed the limit test. The stimulus data array of these points can be read out by CmtNA_GetLimitTestReport function.
CMTNA_ATTR_ANALYSIS_RIPPLE_TEST	ViBoolean	Turns ON/OFF the ripple limit test.
CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_DISPLAY	ViBoolean	Turns ON/OFF the ripple limit line display.
CMTNA_ATTR_ANALYSIS_RIPPLE_VALUE_TYPE	ViInt32	Sets/Gets the display type of the ripple value in the specified band.
CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_FAIL_SIGN	ViBoolean	Turns ON/OFF the "Fail" sign display, when ripple limit test.
CMTNA_ATTR_SAVE_TYPE	ViInt32	Sets/Gets the type of the Analyzer or channel state saving by CmtNA_SaveChannelToRegister function.
CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_FORMAT	ViInt32	Sets/Gets the data format for the S–parameter saving by CmtNA_SaveTouchstoneFile function.
CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARAT	ViInt32	Sets/Gets the Touchstone file separator symbol when the S–parameters are saved by CmtNA_SaveTouchstoneFile function.

Name	Data Type	Description
OR		
CMTNA_ATTR_PRINT_COLOR	ViInt32	Sets/Gets the color chart for the image printout.
CMTNA_ATTR_PRINT_INVERT_IMAGE	ViBoolean	Turns ON/OFF the inverted color image printout.
CMTNA_ATTR_PRINT_DATE_AND_TIME	ViBoolean	Turns ON/OFF the date and time printout in the upper right corner of the image.
CMTNA_ATTR_REFERENCE_FREQUENCY_SOURCE	ViInt32	Sets/Gets the internal or external source of the reference frequency of 10 MHz.
CMTNA_ATTR_EXTERNAL_REFERENCE_ROUTE	ViInt32	Sets/Gets the route of the external 10 MHz reference frequency. (PXIe-S5090 model only).
CMTNA_ATTR_SYSTEM_CORRECTION	ViBoolean	Turns ON/OFF the system correction. The system correction is the factory full 1-port calibration performed at the port connectors.
CMTNA_ATTR_POWER_TRIP_AT_OVERLOAD	ViBoolean	Turns ON/OFF the Power Trip at Overload function. Except for Planar-804/808/304 Models.
CMTNA_ATTR_POWER_SENSOR_TYPE	ViInt32	Sets/Gets the power sensor type to be used in a source power calibration.

Name	Data Type	Description
CMTNA_ATTR_BEEP_COMPLETE_ON	ViBoolean	Turns ON/OFF the beeper notifying of the completion of the operation.
CMTNA_ATTR_BEEP_WARNING	ViBoolean	Turns ON/OFF the beeper notifying of warning.
CMTNA_ATTR_SYSTEM_READ_WRITE_LOCK	ViBoolean	Sets the Analyzer to the local operation mode or remote operation mode, when all the keys on the front panel, mouse and the touch screen are active.
CMTNA_ATTR_SYSTEM_TIME_OUT_MILLISECONDS	ViInt32	I/O timeout value in milliseconds. This value is expressed in milliseconds (ms).
CMTNA_ATTR_VERIFICATION_INTERVAL	ViInt32	Sets/Gets the interval between Instrument Performance Verifications. One year (365 days) is recommended.
CMTNA_ATTR_VERIFICATION_LAST_DATE	ViString	Sets/Gets the date of the last Instrument Performance Verification.
CMTNA_ATTR_VERIFICATION_NEXT_DATE	ViString	Gets the date of the next Instrument Performance Verification.
CMTNA_ATTR_DEVICE_READY	ViBoolean	Gets the ready state of the Analyzer. The state is True when analyzer hardware is connected, powered and the boot process is completed (about 15 sec).

Name	Data Type	Description
CMTNA_ATTR_DEVICE_PXI_CHASSIS	ViInt32	Gets the identifier for the PXI chassis in which the NI vector network analyzer is installed.
CMTNA_ATTR_DEVICE_PXI_SLOT	ViInt32	Gets the number for the PXI slot in which the NI vector network analyzer is installed.
CMTNA_ATTR_DEVICE_TEMPERATURE	ViReal64	Gets the specified sensor temperature inside the Analyzer. The value is expressed in Celsius degrees (°C).
CMTNA_ATTR_DEVICE_SERIAL_NUMBER	ViString	Gets the instrument serial number.
CMTNA_ATTR_NUMBER_OF_PORTS	ViInt32	Gets the number of the ports.
CMTNA_ATTR_LOGICAL_NAME	ViString	Logical Name identifies a driver session in the Configuration Store. If Logical Name is not empty, the driver was initialized from information in the driver session. If it is empty, the driver was initialized without using the Configuration Store.
CMTNA_ATTR_AUTOCAL_MODULE_READY	ViBoolean	Gets the ready state of the AutoCal module. The state is True when the AutoCal module is connected.

Name	Data Type	Description
CMTNA_ATTR_POWER_SENSOR_READY	ViBoolean	Gets the ready state of the Power sensor. The state is True when the Power sensor is ready.
CMTNA_ATTR_RANGE_CHECK	ViBoolean	Drivers may choose to always validate some property/parameter values, never validate others, and optionally validate others, to avoid sending invalid commands to the instrument. If True, the driver performs optional validations.
CMTNA_ATTR_QUERY_INSTRUMENT_STATUS	ViBoolean	If True, the driver queries the instrument status at the end of each method or property that performs I/O to the instrument. If an error is reported, use ErrorQuery to retrieve error messages one at a time from the instrument.
CMTNA_ATTR_CACHE	ViBoolean	Drivers may choose to always cache some instrument settings, never cache others, and optionally cache others, to avoid unnecessary I/O to the instrument. If True, the driver caches optionally cached instrument settings.
CMTNA_ATTR_SIMULATE	ViBoolean	If True, the driver does not perform I/O to the instrument, and returns simulated values for output parameters.
CMTNA_ATTR_RECORD_COERCIONS	ViBoolean	If True, the driver keeps a list of the value coercions it makes for ViInt32 and ViReal64 attributes. If the driver does not support coercion recording, attempts to set RecordCoercions to True will return an error.

Name	Data Type	Description
CMTNA_ATTR_DRIVER_SETUP	ViString	The DriverSetup string is used to set the initial values for attributes that are specific to driver. The format of this string is "AttributeName=Value," where AttributeName is the name of the attribute and Value is the value to which the attribute is set. To set multiple attributes, separate their assignments with a comma.
CMTNA_ATTR_INTERCHANGE_CHECK	ViBoolean	If True, the driver maintains a record of interchangeability warnings. If the driver does not support interchangeability checking, attempts to set InterchangeCheck to True return an error.
CMTNA_ATTR_SUPPORTED_INSTRUMENT_MODELS	ViString	A comma-separated list of instrument models that the IVI specific driver can control. The string does not include an abbreviation for the manufacturer if it is the same for all models.
CMTNA_ATTR_INSTRUMENT_MANUFACTURER	ViString	Gets the Analyzer identification string.
CMTNA_ATTR_INSTRUMENT_MODEL	ViString	Gets the Analyzer identification string.
CMTNA_ATTR_INSTRUMENT_FIRMWARE_REVISION	ViString	Gets the Analyzer identification string.
CMTNA_ATTR_SPECIFIC_DRIVER_PREFIX	ViString	Returns the case-sensitive prefix of the user-callable functions that the IVI-C specific driver exports. The string that this attribute returns

Name	Data Type	Description
		contains a maximum of 32 bytes including the NUL byte.
CMTNA_ATTR_SPECIFIC_DRIVER_REVISION	ViString	Returns version information about the IVI specific driver. The string that this attribute returns contains a maximum of 256 bytes including the NUL byte.
CMTNA_ATTR_SPECIFIC_DRIVER_VENDOR	ViString	Returns the name of the vendor that supplies the IVI specific driver. The string that this attribute returns contains a maximum of 256 bytes including the NUL byte.
CMTNA_ATTR_SPECIFIC_DRIVER_DESCRIPTION	ViString	Returns a brief description of the IVI specific driver. The string that this attribute returns contains a maximum of 256 bytes including the NUL byte.
CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MAJOR_VERSION	ViInt32	Returns the major version number of the class specification in accordance with which the IVI specific driver was developed. Zero is returned if the driver is not compliant with a class specification.
CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MINOR_VERSION	ViInt32	Returns the minor version number of the class specification in accordance with which the IVI specific driver was developed. Zero is returned if the driver is not compliant with a class specification.

Creating an Application with Microsoft Visual Studio 2010

This section describes how to use an IVI-C driver from Visual C++.

Referencing the Driver

In order to access any of the driver functions, the proper header file (.h) must be included in the project and the proper import library (.lib) must be referenced. This section demonstrates usage of the driver using instrument-specific references.

All IVI-C driver programs must do the following:

- #include "CmtNA.h".
- Link to CmtNA.lib.
- Prefix function calls with "CmtNA_".

To use the CmtNA specific driver, perform the following steps in Visual Studio.

1. In solution explorer, right-click on the project and choose Properties.
2. In the property pages dialog, expand the Linker node and select Input.
3. In the Additional Dependencies field, enter "CmtNA.lib".
4. Click OK.
5. In the main application source file (.cpp), add the following line to the top of the file:

```
#include "CmtNA.h"
```

Visual Studio must know the path to the driver's library (.lib) and header (.h) files. The following paths can be entered using the Tools, Options, Projects and Solutions, VC++ Directories dialog.

For Include Files (Win32 & x64) add:	C:\Program	Files	(x86)\IVI Foundation\IVI\Include
For Library Files (Win32) add:	C:\Program	Files	(x86)\IVI Foundation\IVI\Lib_\msc
For Library Files (x64) add:	C:\Program	Files	(x86)\IVI Foundation\IVI\Lib_x64\msc

NOTE For 32 bit operating systems paths start with: C:\Program Files\

Alternately, these paths may be entered in the Project Properties dialog, Configuration Properties, C++ and Linker panes.

Initializing the Driver

Calling **CmtNA_InitWithOptions** will establish an I/O connection to an instrument (often referred to as an "I/O session") or setup the driver to work in simulation mode. Calling **CmtNA_close** at the end of the program is required by the IVI specifications, else unpredictable driver behavior could result. Any resources held by the driver will not be properly released if **close** is not called.

For more details on initializing the driver, see the topic: Initializing the Driver.

The driver is typically initialized using a resource descriptor, often a VISA resource descriptor, as shown below.

C++

```
ViSession session;
ViStatus status;
status = CmtNA_InitWithOptions("PXI1SLOT3", VI_TRUE, VI_TRUE, "", &session);
status = CmNA_close(session);
```

Initializing Using Options

This example shows how IVI-defined initialization options and driver-specific options can be passed to the Initialize function.

For more details on initializing the driver, see the topic: Initializing the Driver.

C++

```
// If true, this will query the instrument model and fail initialization
// if the model is not supported by the driver
ViBoolean idQuery = VI_FALSE;

// If true, the instrument is reset at initialization
ViBoolean reset = VI_FALSE;

// Setup IVI-defined initialization options
ViConstString standardInitOptions =
```

C++

```
"Cache=true, InterchangeCheck=false, QueryInstrStatus=true, RangeCheck=true, RecordCoercions=false, Simulate=false";


status = CmtNA_InitWithOptions("PXI1SLOT3", idQuery, reset, standardInitOptions, &session);

status = CmtNA_close(vi);
```

Accessing Attributes

Accessing attributes in an IVI-C driver is accomplished via a set of IVI-defined accessor functions. There are two accessor functions for each attribute type -- one accessor for reading (Get) attribute values and another accessor for writing (Set) attribute values.

The standard attribute accessors for reading attribute values are:

- GetAttributeViInt32
- GetAttributeViInt64
- GetAttributeViReal64
- GetAttributeViBoolean
- GetAttributeViString

Correspondingly, the standard attribute accessors for writing attribute values are:

- SetAttributeViInt32
- SetAttributeViInt64
- SetAttributeViReal64
- SetAttributeViBoolean
- SetAttributeViString

Each attribute accessor takes an attribute ID that uniquely identifies the attribute to access. These attribute IDs are #define'd constants listed in the CmtNA.h header file and documented in the "Attributes by Name" section of the help file.. The following example demonstrates basic usage of attribute accessors to read and write IVI-C driver attribute values.

The following code is for illustration only and may not apply directly to this driver.

C++

```
ViStatus status;

// Set the value of an attribute
status = CmtNA_SetAttributeViReal64(vi, VI_NULL, CMTNA_ATTR_VOLTAGE_RANGE, 20.0);

// Get the value of a property
ViReal64 voltageRange;
status = CmtNA_GetAttributeViReal64(vi, VI_NULL, CMTNA_ATTR_VOLTAGE_RANGE, &voltageRange);

// Set the value of a property (enum)
// Note use of the #define'd constant for the attribute value
status = CmtNA_SetAttributeViInt32(vi, VI_NULL, CMTNA_ATTR_TEMP_TRANSDUCER_TYPE, CMTNA_VAL_THERMOCOUPLE);
```

Complete Example

Complete, working example programs may be found by clicking the Windows Start Menu shortcut under

All Programs, CMT IVI Drivers, CmtNA, Examples.

C++

```
#include "CmtNA.h"

void main()
{
    ViSession session;
    ViStatus status;
    ViChar str[128];
    ViInt32 ErrorCode;
```

C++

```
ViChar ErrorMessage[256];

// Initialize the driver
status = CmtNA_InitWithOptions("PXI1SLOT3", VI_TRUE, VI_TRUE, "", &session);
if(status)
{
    // Initialization failed
    CmtNA_GetError(session, &ErrorCode, 255, ErrorMessage);
    printf("** InitWithOptions() Error: %d, %s\n", ErrorCode, ErrorMessage);
    printf("\nDone - Press Enter to Exit");
    getchar();
    return;
}
printf("Driver Initialized \n");

// Read and output a few attributes
status = CmtNA_GetAttributeViString(session, "", CMTNA_ATTR_SPECIFIC_DRIVER_PREFIX, 127, str);
printf("DRIVER_PREFIX:     %s\n", str);
status = CmtNA_GetAttributeViString(session, "", CMTNA_ATTR_SPECIFIC_DRIVER_REVISION, 127, str);
printf("DRIVER_REVISION:   %s\n", str);
status = CmtNA_GetAttributeViString(session, "", CMTNA_ATTR_SPECIFIC_DRIVER_VENDOR, 127, str);
printf("DRIVER_VENDOR:     %s\n", str);
status = CmtNA_GetAttributeViString(session, "", CMTNA_ATTR_SPECIFIC_DRIVER_DESCRIPTION, 127, str);
printf("DRIVER_DESCRIPTION: %s\n", str);
status = CmtNA_GetAttributeViString(session, "", CMTNA_ATTR_INSTRUMENT_MODEL, 127, str);
```

C++

```
printf("INSTRUMENT_MODEL:   %s\n", str);
status = CmtNA_GetAttributeViString(session, "", CMTNA_ATTR_INSTRUMENT_FIRMWARE_REVISION, 127, str);
printf("FIRMWARE_REVISION:  %s\n", str);

// TODO: Exercise driver functions and attributes

// Check instrument for errors
ErrorCode = 1;
printf("\n");
while(ErrorCode)
{
    status = CmtNA_error_query( session, &ErrorCode, ErrorMessage);
    printf("error_query: %d, %s\n", ErrorCode, ErrorMessage);
}

// Close the session
status = CmtNA_close(session);
if(status)
{
    CmtNA_GetError(session, &ErrorCode, 255, ErrorMessage);
    printf("** Close() Error: %d, %s\n", ErrorCode, ErrorMessage);
}
else
    printf("Driver Closed \n");
}
```

Functions By Name

CmtNA_init

Description

Opens the I/O session to the instrument. Driver methods and properties that access the instrument are only accessible after Initialize is called. Initialize optionally performs a Reset and queries the instrument to validate the instrument model.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus    CmtNA_init(
        ViRsrc resourceName,
        ViBoolean idQuery,
        ViBoolean reset,
        ViSession *vi
);
```

Parameters

resourceName

An IVI logical name or an instrument specific string that identifies the address of the instrument, such as a VISA resource descriptor string.

idQuery

Specifies whether to verify the ID of the instrument.

VI_TRUE - Perform an ID query. This value is the default.

VI_FALSE - Do not perform an ID query.

reset

Specifies whether to reset the instrument.

VI_TRUE - The device is reset.

VI_FALSE - The device is not reset. This value is the default.

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_close

Description

Closes the I/O session to the instrument. Driver methods and properties that access the instrument are not accessible after Close is called.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_close(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_InitWithOptions

Description

Opens the I/O session to the instrument. Driver methods and properties that access the instrument are only accessible after Initialize is called. Initialize optionally performs a Reset and queries the instrument to validate the instrument model.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_InitWithOptions(
        ViRsrc resourceName,
        ViBoolean idQuery,
        ViBoolean reset,
        ViConstString optionsString,
        ViSession *vi
);
```

Parameters

resourceName

An IVI logical name or an instrument specific string that identifies the address of the instrument, such as a VISA resource descriptor string.

idQuery

Specifies whether to verify the ID of the instrument.

VI_TRUE - Perform an ID query. This value is the default.

VI_FALSE - Do not perform an ID query.

reset

Specifies whether to reset the instrument.

VI_TRUE - The device is reset.

VI_FALSE - The device is not reset. This value is the default.

optionsString

The user can use the OptionsString parameter to specify the initial values of certain IVI inherent attributes for the session. The format of an assignment in the OptionsString parameter is "Name=Value", where Name is one of: RangeCheck, QuerytInstrStatus, Cache, Simulate, RecordCoercions, InterchangeCheck or DriverSetup. Value is either true or false except for DriverSetup. If the Options String parameter contains an assignment for the Driver Setup attribute, the Initialize function assumes that everything following "DriverSetup=" is part of the assignment. For a detail description of OptionsString parameter see in Initializing the IVI-C Driver.

The following attributes are used in this parameter:

CMTNA_ATTR_DEVICE_DRIVER

CMTNA_ATTR_SIMULATE

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_self_test

Description

Performs an instrument self test, waits for the instrument to complete the test, and queries the instrument for the results. If the instrument passes the test, TestResult is zero and TestMessage is 'Self test passed'.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_self_test(
        ViSession vi,
        ViInt16 *testResult,
        ViChar testMessage[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

testResult

The numeric result from the self test operation. 0 = no error (test passed).

testMessage

The self test status message.

Return value

Success or failure code.

SCPI Command

*TST?

SYSTem:TEST?

Back to Functions

CmtNA_reset

Description

Resets the device to a known initialization state.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_reset(
        ViSession vi,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Back to Functions

CmtNA_revision_query

Description

Queries the revision of the device.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_revision_query(
        ViSession vi,
        ViChar driverRevision[],
        ViChar firmwareRevision[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

driverRevision

Returns the revision of the IVI specific driver, which is the value held in the Specific Driver Revision attribute. Refer to the Specific Driver Revision attribute for more information.

firmwareRevision

Returns the firmware revision of the instrument, which is the value held in the Instrument Firmware Revision attribute. Refer to the Instrument Firmware Revision attribute for more information.

Return value

Success or failure code.

Back to Functions

CmtNA_ResetWithDefaults

Description

Resets the device to a known initialization state with defaults as specified for the device in the optionString parameter in the CmtNA_InitWithOptions function.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_ResetWithDefaults(
        ViSession vi,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Back to Functions

CmtNA_Disable

Description

Places the instrument in a quiescent state where it has minimal or no impact on the system to which it is connected.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_Disable(
        ViSession vi,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Back to Functions

CmtNA_LockSession

Description

Obtains a multithread lock on the instrument session. Before doing so, this function waits until all other execution threads have released their locks on the instrument session.

Other threads might have obtained a lock on this session in the following ways:

- Application already called this function.
- A call to driver locked the session.

After the call to this function returns successfully, no other threads can access the instrument session until you call the CmtNA_UnlockSession function. Use the CmtNA_LockSession function and the CmtNA_UnlockSession function around a sequence of calls to driver functions if you require that the device driver retains its settings through the end of the sequence.

You can safely make nested calls to the CmtNA_LockSession function within the same thread. To completely unlock the session, balance each call to the CmtNA_LockSession function with a call to the CmtNA_UnlockSession function. If, however, you use *callerHasLock* in all calls to the CmtNA_LockSession function and the CmtNA_UnlockSession function within a function, the IVI Library locks the session only once within the function regardless of the number of calls you make to the CmtNA_LockSession function. Locking the session only once allows you to call the CmtNA_UnlockSession function just once at the end of the function.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_LockSession(
        ViSession vi,
        ViBoolean *callerHasLock
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

callerHasLock

Keeps track of whether you obtain a lock and therefore need to unlock the session in complex functions. Pass the address of a local ViBoolean variable. In the declaration of the local variable, initialize it to VI_FALSE. Pass the address of the same local variable to any other calls you make to this function or the CmtNA_UnlockSession function in the same function.

This parameter serves as a convenience. If you do not want to use this parameter, pass VI_NULL.

The CmtNA_LockSession function and the CmtNA_UnlockSession function each inspect the current value and take the following actions:

CmtNA_LockSession	VI_TRUE	The CmtNA_LockSession function does not lock the session again.
	VI_FALSE	The CmtNA_LockSession function obtains the lock and sets the value of the parameter to VI_TRUE.
CmtNA_UnlockSession	VI_FALSE	The CmtNA_UnlockSession function does not attempt to unlock the session.
	VI_TRUE	The CmtNA_UnlockSession function releases the lock and sets the value of the parameter to VI_FALSE.

Thus, you can call the CmtNA_UnlockSession function at the end of your function regardless of whether you actually have the lock.

Return value

Success or failure code.

Back to Functions

CmtNA_UnlockSession

Description

Releases a lock obtained on an device driver session by calling the CmtNA_LockSession function. Refer to the CmtNA_LockSession function for additional information on session locks.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_UnlockSession(
        ViSession vi,
        ViBoolean *callerHasLock
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

callerHasLock

Keeps track of whether is obtained a lock and therefore need to unlock the session in complex functions. Pass the address of a local ViBoolean variable. In the declaration of the local variable, initialize it to VI_FALSE. Pass the address of the same local variable to any other calls you make to this function or the CmtNA_UnlockSession function in the same function.

This parameter serves as a convenience. If you do not want to use this parameter, pass VI_NULL.

The CmtNA_LockSession function and the CmtNA_UnlockSession function each inspect the current value and take the following actions:

CmtNA_LockSession	VI_TRUE	The CmtNA_LockSession function does not lock the session again.

	VI_FALSE	The CmtNA_LockSession function obtains the lock and sets the value of the parameter to VI_TRUE.
CmtNA_UnlockSession	VI_FALSE	The CmtNA_UnlockSession function does not attempt to unlock the session.
	VI_TRUE	The CmtNA_UnlockSession function releases the lock and sets the value of the parameter to VI_FALSE.

Thus, you can call the CmtNA_UnlockSession function at the end of your function regardless of whether you actually have the lock.

Return value

Success or failure code.

Back to Functions

CmtNA_error_query

Description

Queries the instrument and returns instrument specific error information. This function can be used when QueryInstrumentStatus is True to retrieve error details when the driver detects an instrument error.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_error_query(
        ViSession vi,
        ViStatus *errorCode,
        ViChar errorMessage[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

errorCode

Passes the return Value parameter that is returned from any driver function. The default value is 0 (VI_SUCCESS).

errorMessage

Returns the user-readable message string that corresponds to the status code you specify.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:ERRor[:NEXT]?

Back to Functions

CmtNA_error_message

Description

Translates the error return value from an IVI driver function to a user-readable string. The user should pass a buffer with at least 256 bytes for the ErrorMessage parameter.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_error_message(
        ViSession vi,
        ViStatus errorCode,
        ViChar errorMessage[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

errorCode

Passes the returnValue parameter that is returned from any driver function. The default value is 0 (VI_SUCCESS).

errorMessage

Returns the user-readable message string that corresponds to the status code you specify.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_GetError

Description

This function retrieves and then clears the IVI error information for the session or the current execution thread. If the user specifies a valid IVI session for the Vi parameter, Get Error retrieves and then clears the error information for the session. If the user passes VI_NULL for the Vi parameter, Get Error retrieves and then clears the error information for the current execution thread. If the Vi parameter is an invalid session, the function does nothing and returns an error. Normally, the error information describes the first error that occurred since the user last called CmtNA_GetError or CmtNA_ClearError function.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_GetError(
        ViSession vi,
        ViStatus errorCode,
        ViInt32 bufferSize,
        ViChar description[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

errorCode

Passes the returnValue parameter that is returned from any driver function. The default value is 0 (VI_SUCCESS).

bufferSize

Specifies the size of the buffer into which to acquire the error message.

description

Returns the error description for the IVI session or execution thread. If there is no description, the CmtNA_GetError function returns an empty string.

The buffer must contain at least as many elements as the value you specify with bufferSize. If the error description, including the terminating NULL byte, contains more bytes than you indicate with bufferSize, the function copies bufferSize - 1 bytes into the buffer, places an ASCII NULL byte at the end of the buffer, and returns the size of the buffer that you must pass to get the entire value.

If you pass 0 to bufferSize, you can pass VI_NULL for this parameter.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ClearError

Description

This function clears the error code and error description for the current execution thread and for the IVI session. If the user specifies a valid IVI session for the Vi parameter, this function clears the error information for the session. If the user passes VI_NULL for the Vi parameter, this function clears the error information for the current execution thread. If the Vi parameter is an invalid session, the function does nothing and returns an error.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_ClearError(
        ViSession vi,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_GetAttributeViInt32

Description

Queries the value of a ViInt32 attribute. Use this low-level function to get the values of inherent IVI attributes, and instrument-specific attributes.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Returns the current value of the attribute. Pass the address of a ViInt32 variable.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_GetAttributeViReal64

Description

Queries the value of a ViReal64 attribute. Use this low-level function to get the values of inherent IVI attributes, and instrument-specific attributes.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Returns the current value of the attribute. Pass the address of a ViReal64 variable.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_GetAttributeViString

Description

Queries the value of a ViString attribute.Use this low-level function to get the values of inherent IVI attributes, and instrument-specific attributes.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViSession bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

bufferSize

Pass the number of bytes in the ViChar buffer you specify for the attribute value parameter.

If you pass a negative number, the function copies the value to the buffer regardless of the number of bytes in the value.

If you pass 0, you can pass VI_NULL for the attribute value buffer parameter.

value

Returns the current value of the attribute. Pass the address of a ViString variable.

Return *value*

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_GetAttributeViBoolean

Description

Queries the value of a ViBoolean attribute.Use this low-level function to get the values of inherent IVI attributes, and instrument-specific attributes.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Returns the current value of the attribute. Pass the address of a ViBoolean variable.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_GetAttributeViSession

Description

Queries the value of a ViSession attribute.Use this low-level function to get the values of inherent IVI attributes, and instrument-specific attributes.

Declaration: CmtNA.h

Syntax

C++

```
ViStatus  CmtNA_GetAttributeViSession(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViSession *value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Returns the current value of the attribute. Pass the address of a ViSession variable.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_SetAttributeViInt32

Description

Sets the value of a ViInt32 attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViSession value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Pass the value to which you want to set the attribute.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_SetAttributeViReal64

Description

Sets the value of a ViReal64 attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Pass the value to which you want to set the attribute.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_SetAttributeViString

Description

Sets the value of a ViString attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Pass the value to which you want to set the attribute.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_SetAttributeViBoolean

Description

Sets the value of a ViBoolean attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Pass the value to which you want to set the attribute.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_SetAttributeViSession

Description

Sets the value of a ViSession attribute. Use this low-level function to set the values of inherent IVI attributes and instrument-specific attributes.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetAttributeViSession(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViSession value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. For valid values, see the Accessing Repeated Capabilities.

attributeID

Pass the ID of an attribute.

value

Pass the value to which you want to set the attribute.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ReadString

Description

Reads the entire contents of the buffer until the termination character /n and return the data as string.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ReadString(
        ViSession vi,
        ViSession bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

bufferSize

The number of bytes in the ViChar array that the user specifies for the Val parameter.

value

Value read from the instrument.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_WriteString

Description

Writes a string to the I/O Stream and flushes the buffer.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_WriteString(
        ViSession vi,
         ViConstString command
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

command

(String): Any SCPI Command supported by the Analyzer.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_Queryf

Description

Performs a formatted write and read through a single call to an operation. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_WriteString(
        ViSession vi,
        ViString writeFmt,
        ViString readFmt,
        ...
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

writeFmt

String describing the format of write arguments.

readFmt

String describing the format of read arguments.

...

Parameters to which write and read format strings are applied.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_Printf

Description

Converts, formats and sends the parameters (designated by ...) to the device as specified by the format string. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_Printf(
        ViSession vi,
        ViString writeFmt,
        ...
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

writeFmt

String describing the format of write arguments.

...

Parameters to which write and read format string is applied.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_Scanf

Description

Reads, converts and formats data using the format specifier. Stores the formatted data in the parameters (designated by ...). This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_Scanf(
        ViSession vi,
        ViString readFmt,
        ...
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

readFmt

String describing the format of read arguments.

...

Parameters to which write and read format string is applied.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_VQueryf

Description

Performs a formatted write and read through a single call to an operation. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_VQueryf(
        ViSession vi,
        ViString writeFmt,
        ViString readFmt,
        ViVAList params
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

writeFmt

String describing the format of write arguments.

readFmt

String describing the format of read arguments.

params

A list containing the variable number of write and read parameters. The write parameters are formatted and written to the specified device. The read parameters store the data read from the device after the format string is applied to the data.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_VPrintf

Description

Converts, formats and sends the parameters (designated by *params*) to the device or interface as specified by the format string. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_VPrintf(
        ViSession vi,
        ViString writeFmt,
        ViVAList params
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

writeFmt

String describing the format of write arguments.

params

A list containing the variable number of write and read parameters. The write parameters are formatted and written to the specified device. The read parameters store the data read from the device after the format string is applied to the data.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_VScanf

Description

Reads, converts and formats data using the format specifier. Stores the formatted data in the parameters designated by *params*. This function uses a direct call to the viPrintf function from the VISA library and saves its formatting and arguments.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_VPrintf(
        ViSession vi,
        ViString readFmt,
        ViVAList params
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

readFmt

String describing the format of read arguments.

params

A list containing the variable number of write and read parameters. The write parameters are formatted and written to the specified device. The read parameters store the data read from the device after the format string is applied to the data.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ConfigureSweep

Description

Configures the number of points for the sweep measurement and the sweep mode.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureSweep(
        ViSession vi,
        ViConstString repCapID,
        ViSession numberOfPoints,
        ViSession sweepMode
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

numberOfPoints

The number of points to acquire as part of the sweep measurement.

See more details: CMTNA_ATTR_STIMULUS_POINTS.

sweepMode

The mode for the sweep measurement.

See more details: CMTNA_ATTR_STIMULUS_SWEEP_TYPE.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ConfigureFrequencyCenterSpan

Description

Sets the sweep range for the channel using center and span value of the frequencies.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureFrequencyCenterSpan(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 centerFrequency,
        ViReal64 frequencySpan
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

centerFrequency

The center frequency in a frequency sweep.

See more details: CMTNA_ATTR_STIMULUS_FREQUENCY_CENTER.

frequencySpan

The frequency span for the frequency sweep. This frequency span is centered at the value you select for *centerFrequency*.

See more details: CMTNA_ATTR_STIMULUS_FREQUENCY_SPAN.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ConfigureFrequencyStartStop

Description

Sets the sweep range for the channel using start and stop value of the frequencies.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureFrequencyStartStop(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 startFrequency,
        ViReal64 stopFrequency
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

startFrequency

The lower limit of a span of frequencies.

See more details: CMTNA_ATTR_STIMULUS_FREQUENCY_START.

stopFrequency

The upper limit of a span of frequencies.

See more details: CMTNA_ATTR_STIMULUS_FREQUENCY_STOP.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ConfigurePowerStartStop

Description

Sets the sweep range for the channel using start and stop value of the output power.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigurePowerStartStop(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 startPower,
        ViReal64 stopPower
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

startPower

The start power for power sweep.

See more details: CMTNA_ATTR_STIMULUS_POWER_START.

stopPower

The stop power for power sweep.

See more details: CMTNA_ATTR_STIMULUS_POWER_STOP.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_ConfigurePowerCenterSpan

Description

Sets the sweep range for the channel using center and span value of the output power.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigurePowerCenterSpan(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 centerPower,
        ViReal64 spanPower
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

centerPower

The center value of the output power for power sweep.

See more details: CMTNA_ATTR_STIMULUS_POWER_CENTER.

spanPower

The span value of the output power for power sweep.

See more details: CMTNA_ATTR_STIMULUS_POWER_SPAN.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_SetSegmentData

Description

Sets the array of the segment sweep table.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetSegmentData(
        ViSession vi,
        ViConstString repCapID,
        ViSession segmentCount,
        ViReal64 startFrequencies[],
        ViReal64 stopFrequencies[],
        ViSession numberPoints[],
        ViBoolean listIFBW,
        ViReal64 ifbwValues[],
        ViBoolean listPower,
        ViReal64 powerValues[],
        ViBoolean listDelay,
        ViReal64 delayValues[],
        ViBoolean listTime,
        ViReal64 timeValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

segmentCount

The number of segments.

startFrequencies

The array of start value.

stopFrequencies

The array of stop value.

numberPoints

The array of number of points.

listIFBW

Setting of *ifbwValues* (0 – disabled, 1 – enabled).

ifbwValues

The array of IF bandwidth value (if enabled).

listPower

Setting of *powerValues* (0 – disabled, 1 – enabled).

powerValues

The array of power value (if enabled).

listDelay

Setting of *delayValues* (0 – disabled, 1 – enabled).

delayValues

The array of measurement delay value (if enabled).

listTime

Setting of *timeValues* (0 – disabled, 1 – enabled).

timeValues

Reserved for future use (if enabled).

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:SEGMent:DATA <numeric list>

Back to Functions

CmtNA_GetSegmentData

Description

Gets the array of the segment sweep table.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetSegmentData(
        ViSession vi,
        ViConstString repCapID,
        ViSession anyBuffersSize,
        ViInt32 *segmentCount,
        ViReal64 startFrequencies[],
        ViReal64 stopFrequencies[],
        ViSession numberPoints[],
        ViBoolean *listIFBW,
        ViReal64  ifbwValues[],
        ViBoolean *listPower,
        ViReal64 powerValues[],
        ViBoolean *listDelay,
        ViReal64 delayValues[],
        ViBoolean *listTime,
        ViReal64 timeValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *startFrequencies*, *stopFrequencies*, *numberPoints*, *ifbwValues*, *powerValues*, *delayValues*, *timeValues* parameters.

segmentCount

The number of segments.

startFrequencies

The array of start value.

stopFrequencies

The array of stop value.

numberPoints

The array of number of points.

listIFBW

Setting of *ifbwValues* (0 – disabled, 1 – enabled).

ifbwValues

The array of IF bandwidth value (if enabled).

listPower

Setting of *powerValues* (0 – disabled, 1 – enabled).

powerValues

The array of power value (if enabled).

listDelay

Setting of *delayValues* (0 – disabled, 1 – enabled).

delayValues

The array of measurement delay value (if enabled).

listTime

Setting of *timeValues* (0 – disabled, 1 – enabled).

timeValues

Reserved for future use (if enabled).

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:SEGMent:DATA?

Back to Functions

CmtNA_SaveSegmentTable

Description

Saves the segment table in a file.

The function sets attribute for the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveSegmentTable(
        ViSession vi,
        ViConstString repCapID,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \Segment subdirectory of the application directory will be searched for the file. The segment file has *.seg extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:SEGMent <string>

Back to Functions

CmtNA_RecallSegmentTable

Description

Recalls the segment table file. The file must be saved by CmtNA_SaveSegmentTable function.

The function sets attribute for the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_RecallSegmentTable(
        ViSession vi,
        ViConstString repCapID,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

fileName

The file path into which the data can be saved . If the full path of the file is not specified, the \Segment subdirectory of the application directory will be searched for the file. The segment file has *.seg extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:SEGMent <string>

Back to Functions

CmtNA_HoldAllChannels

Description

Turns OFF the continuous trigger initiation mode for all channels.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_HoldAllChannels(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

INITiate:CONTinuous:ALL {OFF|0}

Back to Functions

CmtNA_ContinuousAllChannels

Description

Turns ON the continuous trigger initiation mode for all channels.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ContinuousAllChannels(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

INITiate:CONTinuous:ALL {ON|1}

Back to Functions

CmtNA_TriggerInit

Description

Puts the channel to the Trigger Waiting state for the one trigger event. The channel should be in the hold state, otherwise an error occurs and the command is ignored.

The channel goes into the Hold as a result of CMTNA_ATTR_STIMULUS_TRIGGER_MODE (set to CMTNA_ATTR_TRIGGER_MODE_HOLD).

If the Internal trigger source is selected by CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE (set to CMTNA_ATTR_TRIGGER_SOURCE_INTERNAL), then the command initiates a sweep in the single channel, otherwise the channel goes to the Waiting for a Single Trigger mode.

Upon receipt of a trigger from the selected source, the sweep starts for the channels awaiting trigger. On completion of the sweep the channel goes to the Hold state.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TriggerInit(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

INITiate<Ch>[:IMMediate]

Back to Functions

CmtNA_WaitForTriggerState

Description

Waits for the specified state of the analyzer to be reached. The function can be used to wait for the end of the sweep. If the continuous initiation mode is turned ON - the parameter of the function must be WAIT, otherwise HOLD.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_WaitForTriggerState(
        ViSession vi,
        ViInt32 state,
        ViInt32 timeoutMilliseconds
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

state

Specifies state of the analyzer.

See more details:CMTNA_ATTR_STIMULUS_TRIGGER_STATE.

timeoutMilliseconds

Maximum time out. This value is expressed in milliseconds.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

TRIGger[:SEQuence]:WAIT <char>

Back to Functions

CmtNA_WaitForSweepComplete

Description

Minimizes the analyzer main window removing it from the desktop.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_WaitForSweepComplete(
        ViSession vi,
        ViInt32 timeoutMilliseconds
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

timeoutMilliseconds

Maximum time out. This value is expressed in milliseconds.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_WaitForOperationComplete

Description

Method returns when all pending operations are complete or maxTimeMilliseconds exceeded.

The function can be used for waiting for the end of a sweep initiated by CmtNA_TriggerSingle, CmtNA_PowerSensorZeroing, CmtNA_TakePowerCalSweep function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_WaitForOperationComplete(
        ViSession vi,
        ViInt32 timeoutMilliseconds
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

timeoutMilliseconds

Maximum time out. This value is expressed in milliseconds.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

Back to Functions

CmtNA_TriggerSingle

Description

Generates a trigger signal and initiates a sweep under the following conditions:

- Trigger source is set to the BUS (set by the CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE), otherwise an error occurs and the function is ignored.
- Analyzer must be in the trigger waiting state, otherwise (the analyzer is in the measurement state or in the hold state) an error occurs and the function is ignored.

As opposed to the function CmtNA_TriggerImmediate this function is pending till the end of the sweep.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TriggerSingle(
        ViSession vi,
        ViConstString repCapID,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

TRIGger[:SEQuence]:SINGle

Back to Functions

CmtNA_TriggerSingleAndWait

Description

Generates a trigger signal and initiates a sweep under the following conditions:

- Trigger source is set to the BUS (set by the CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE), otherwise an error occurs and the function is ignored.
- Analyzer must be in the trigger waiting state, otherwise (the analyzer is in the measurement state or in the hold state) an error occurs and the function is ignored.

As opposed to the function CmtNA_TriggerImmediate this function is pending till the end of the sweep.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TriggerSingleAndWait(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 timeoutMilliseconds
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

timeoutMilliseconds

Maximum time out. This value is expressed in milliseconds.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

TRIGger[:SEQuence]:SINGle

Back to Functions

CmtNA_TriggerImmediate

Description

Generates a trigger signal and initiates a sweep under the following conditions:

- Trigger source is set to the BUS (set by the command TRIG:SOUR BUS), otherwise an error occurs and the command is ignored.
- Analyzer must be in the trigger waiting state, otherwise (the analyzer is in the measurement state or in the hold state) an error occurs and the function is ignored.

The function is completed immediately after the generation of the trigger signal (does not wait the end of a sweep).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TriggerImmediate(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

TRIGger[:SEQuence][:IMMediate]

Back to Functions

CmtNA_TriggerRestart

Description

Aborts the sweep. The channels in the Single trigger initiation mode transit to the Hold state. The channels in the Continuous trigger initiation mode transit to the trigger waiting state, if the trigger source is set to Internal, the channel immediately starts a new sweep.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TriggerRestart(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

ABORt

Back to Functions

CmtNA_MeasurementSetParameter

Description

Sets the measurement parameter of the trace.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementSetParameter(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 measurementType,
        ViInt32 newReceiverPort,
        ViInt32 newSourcePort
);
```

Parameters

vi

Unique identifier for an IVI session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

measurementType

The type measurement.

Defined Values

Name	Description
CMTNA_ATTR_MEASUREMENT_TYPE_S_PARAM	S – parameter: S11, S12, S13, S14, S21, S22, S23, S24, S31, S32, S33, S34, S41, S42, S43, S44
CMTNA_ATTR_MEASUREMENT_TYPE_ABSOLUTE_T	Test receiver: A, B, C, D or T1, T2, T3, T4
CMTNA_ATTR_MEASUREMENT_TYPE_ABSOLUTE_R	Reference receiver: R1, R2, R3, R4
CMTNA_ATTR_MEASUREMENT_TYPE_DC_VOLTAGE	DC Voltage: AUX1, AUX2 V1, V2

newReceiverPort

The number of port from which the measurement is performed.

newSourcePort

The number of port which is the source.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>:PARameter<Tr>:DEFine <char>

CALCulate<Ch>:PARameter<Tr>:SPORt <port>

Back to Functions

CmtNA_MeasurementGetParameter

Description

Gets the measurement parameter of the trace.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementGetParameter(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *measurementType,
        ViInt32 *newReceiverPort,
        ViInt32 *newSourcePort
);
```

Parameters

vi

Unique identifier for an IVI session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

measurementType

The type measurement.

Defined Values

Name	Description
CMTNA_ATTR_MEASUREMENT_TYPE_S_PARAM	S – parameter: S11, S12, S13, S14, S21, S22, S23, S24, S31, S32, S33, S34, S41, S42, S43, S44
CMTNA_ATTR_MEASUREMENT_TYPE_ABSOLUTE_T	Test receiver: A, B, C, D or T1, T2, T3, T4
CMTNA_ATTR_MEASUREMENT_TYPE_ABSOLUTE_R	Reference receiver: R1, R2, R3, R4
CMTNA_ATTR_MEASUREMENT_TYPE_DC_VOLTAGE	DC Voltage: AUX1, AUX2 V1, V2

newReceiverPort

The number of port from which the measurement is performed.

newSourcePort

The number of port which is the source.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>:PARameter<Tr>:DEFine?

Back to Functions

CmtNA_MeasurementAutoScale

Description

Executes the auto scale function for the trace.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementAutoScale(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The physical or virtual repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. 'Channel1:Measurement3' and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:AUTO

Back to Functions

CmtNA_MeasurementAutoRefValue

Description

Executes the auto reference function for the trace. The function automatically sets the RLEVel value.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementAutoRefValue(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The physical or virtual repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. 'Channel1:Measurement3' and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RLEVel:AUTO

Back to Functions

CmtNA_ClearAverage

Description

Clears and restarts the averaging process, when the averaging function is turned on.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ClearAverage(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:AVERage:CLEar

Back to Functions

CmtNA_ChannelFrequencyData

Description

The array size is N, where N is the number of measurement points.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ChannelFrequencyData(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 bufferSize,
        ViReal64 retValues[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

bufferSize

The size of the array to hold in the *retValues* parameter.

retValues

The array of the frequency value at the n–th measurement point.

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:FREQuency:DATA?

Back to Functions

CmtNA_MeasurementFetchX

Description

Gets the X-axis values array.

The X-axis values array is the frequency, power or time values array depending on the trace setup. The array contains real values.

The array size is N, where N is the number of measurement points.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementFetchX(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 bufferSize,
        ViReal64 retValues[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

bufferSize

The size of the array to hold in the *retValues* parameter.

retValues

The array of the the X-axis value.

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:DATA:XAXis?

CALCulate<Ch>:TRACe<Tr>:DATA:XAXis?

Back to Functions

CmtNA_MeasurementFetchFormatted

Description

Returns the formatted data array.

The formatted data array is the data, which processing is completed including the formatting as the last step. Such data represent the data trace values as they are shown on the screen.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementFetchFormatted(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 realBufferSize,
        ViReal64 realValues[],
        ViInt32 *realReturnSize,
        ViInt32 imagBufferSize,
        ViReal64 imagValues[],
        ViInt32 *imagReturnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

realBufferSize

The size of the array to hold in the *realValues* parameter.

realValues

The array of the real number in rectangular format, real part in polar and Smith chart formats.

realReturnSize

The actual number of elements filled by the query.

imagBufferSize

The size of the array to hold in the *imagValues* parameter.

imagValues

The array of 0 in rectangular format, imaginary part in polar and Smith chart formats.

imagReturnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:DATA:FDATa?

CALCulate<Ch>:TRACe<Tr>:DATA:FDATa?

Back to Functions

CmtNA_MeasurementFetchComplex

Description

Returns the corrected data array.

The corrected data array is the data, which processing is completed excluding the formatting as the last step. Such data represent S–parameter complex values.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementFetchComplex(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 realBufferSize,
        ViReal64 realValues[],
        ViInt32 *realReturnSize,
        ViInt32 imagBufferSize,
        ViReal64 imagValues[],
        ViInt32 *imagReturnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so

on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

realBufferSize

The size of the array to hold in the *realValues* parameter.

realValues

The array of the real part of corrected measurement.

realReturnSize

The actual number of elements filled by the query.

imagBufferSize

The size of the array to hold in the *imagValues* parameter.

imagValues

The array of the the imaginary part of corrected measurement.

imagReturnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:DATA:SDATa?

CALCulate<Ch>:TRACe<Tr>:DATA:SDATa?

Back to Functions

CmtNA_MeasurementFetchMemoryFormatted

Description

Returns the formatted memory array.

The formatted memory array is the data, which processing is completed including the formatting as the last step. Such data represent the memory trace values as they are shown on the screen.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementFetchMemoryFormatted(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 realBufferSize,
        ViReal64 realValues[],
        ViInt32 *realReturnSize,
        ViInt32 imagBufferSize,
        ViReal64 imagValues[],
        ViInt32 *imagReturnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

realBufferSize

The size of the array to hold in the *realValues* parameter.

realValues

The array of the real number in rectangular format, real part in polar and Smith chart formats.

realReturnSize

The actual number of elements filled by the query.

imagBufferSize

The size of the array to hold in the *imagValues* parameter.

imagValues

The array of 0 in rectangular format, imaginary part in polar and Smith chart formats.

imagReturnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:DATA:FMEMory?

CALCulate<Ch>:TRACe<Tr>:DATA:FMEMory?

Back to Functions

CmtNA_MeasurementFetchMemoryComplex

Description

Returns the corrected memory array.

The corrected memory array is the data, which processing is completed excluding the formatting as the last step. Such data represent S–parameter complex values.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementFetchMemoryComplex(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 realBufferSize,
        ViReal64 realValues[],
        ViInt32 *realReturnSize,
        ViInt32 imagBufferSize,
        ViReal64 imagValues[],
        ViInt32 *imagReturnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so

on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

realBufferSize

The size of the array to hold in the *realValues* parameter.

realValues

The array of the real part of corrected measurement memory.

realReturnSize

The actual number of elements filled by the query.

imagBufferSize

The size of the array to hold in the *imagValues* parameter.

imagValues

The array of the imaginary part of corrected measurement memory.

imagReturnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:DATA:SMEMory?

CALCulate<Ch>:TRACe<Tr>:DATA:SMEMory?

Back to Functions

CmtNA_FunctionExecute

Description

Executes the analysis specified by the CMTNA_ATTR_FUNCTION_TYPE.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_FunctionExecute(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:EXECute

CALCulate<Ch>:TRACe<Tr>:FUNCtion:EXECute

Back to Functions

CmtNA_GetCalculatedFunctionData

Description

Gets the data array, which is the CmtNA_FunctionExecute function analysis result.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalculatedFunctionData(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 bufferSize,
        ViReal64 responseValues[],
        ViReal64 stimulusValues[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

bufferSize

The size of the arrays to hold in the *responseValues*, *stimulusValues* parameter.

responseValues

The array of the response value.

stimulusValues

The array of the stimulus value. Always set to 0 for the analysis of mean value, standard deviation, and peak–to–peak value.

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:DATA?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DATA?

Back to Functions

CmtNA_GetMaxChannelCount

Description

Gets the maximum number of channels.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_GetMaxChannelCount(
        ViSession vi,
        ViInt32 *channelCount
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

channelCount

The maximum number of channels.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SERVice:CHANnel:COUNt?

Back to Functions

CmtNA_GetMaxTraceCount

Description

Gets the maximum number of traces in the channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_GetMaxTraceCount(
        ViSession vi,
        ViInt32 *traceCount
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

traceCount

The maximum number of traces in the channel.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SERVice:CHANnel:TRACe:COUNt?

Back to Functions

CmtNA_MeasurementDataToMemory

Description

Saves measurement trace data into memory.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurementDataToMemory(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MATH:MEMorize

CALCulate<Ch>:TRACe<Tr>:MATH:MEMorize

Back to Functions

CmtNA_TraceHoldRestart

Description

This command resets the trace hold function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TraceHoldRestart(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:HOLD:CLEar

CALCulate<Ch>:TRACe<Tr>:HOLD:CLEar

Back to Functions

CmtNA_DisplaySetDefaults

Description

Restores the display settings to the default values.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_DisplaySetDefaults(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

DISPlay:COLor:RESet

Back to Functions

CmtNA_InitUserCal

Description

Initializes user calibration.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_InitUserCal(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 calibrationType,
        ViInt32 receiverPort,
        ViInt32 sourcePort
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

calibrationType

The type of calibration that is performed.

See more details: CMTNA_ATTR_CALIBRATION_TYPE

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:SHORt <port>

SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:OPEN <port>

SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:THRU <rcvport>,<srcport>

SENSe<Ch>:CORRection:COLLect:METHod:ERESponse <rcvport>,<srcport>

SENSe<Ch>:CORRection:COLLect:METHod:SOLT1 <port>

SENSe<Ch>:CORRection:COLLect:METHod:SOLT2 <port1>,<port2>

Back to Functions

CmtNA_InitNPortUserCal

Description

Initializes user calibration.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_InitNPortUserCal(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 calibrationType,
        ViInt32 portList[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

calibrationType

The type of calibration that is performed.

See more details: CMTNA_ATTR_CALIBRATION_TYPE

portList

The list of ports.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:SHORt <port>

SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:OPEN <port>

SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:THRU <rcvport>,<srcport>

SENSe<Ch>:CORRection:COLLect:METHod:ERESponse <rcvport>,<srcport>

SENSe<Ch>:CORRection:COLLect:METHod:SOLT1 <port>

SENSe<Ch>:CORRection:COLLect:METHod:SOLT2 <port1>,<port2>

SENSe<Ch>:CORRection:COLLect:METHod:SOLT3 <port1>,<port2>,<port3>

SENSe<Ch>:CORRection:COLLect:METHod:SOLT4 <port1>,<port2>,<port3>,<port4>

Back to Functions

CmtNA_PerformCalibrationAction

Description

Performs one calibration action on the specified ports or the type of calibration action.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_PerformCalibrationAction(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 receiverPort
        ViInt32 sourcePort,
        ViInt32 actionType
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Port index for the n–th trace of channel i.e. "Channel1:Port2" and so on.For valid values, see the Accessing Repeated Capabilities.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

actionType

The number of port which is the source.

Defined Values

Name	Description
CMTNA_CALIBRATION_ACTION_OPEN	OPEN calibration
CMTNA_CALIBRATION_ACTION_SHORT	SHOT calibration
CMTNA_CALIBRATION_ACTION_LOAD	LOAD calibration
CMTNA_CALIBRATION_ACTION_THRU	THRU calibration
CMTNA_CALIBRATION_ACTION_ISOLATION	Isolation calibration
CMTNA_CALIBRATION_ACTION_POWER_CAL	The power calibration
CMTNA_CALIBRATION_ACTION_TEST_RECEIVER_CAL	The test receiver calibration
CMTNA_CALIBRATION_ACTION_REFERENCE_RECEIVER_CAL	The reference receiver calibration

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect[:ACQuire]:OPEN <port>

SENSe<Ch>:CORRection:COLLect[:ACQuire]:SHORt <port>

SENSe<Ch>:CORRection:COLLect[:ACQuire]:LOAD <port>

SENSe<Ch>:CORRection:COLLect[:ACQuire]:THRU <rcvport>,<srcport>

SENSe<Ch>:CORRection:COLLect[:ACQuire]:ISOLation <rcvport>,<srcport>

SOURce<Ch>:POWer:PORT<Pt>:CORRection:COLLect[:ACQuire]

SENSe<Ch>:CORRection:RECeiver<Pt>:COLLect:TCHannel:ACQuire <srcport>

SENSe<Ch>:CORRection:RECeiver<Pt>:COLLect:RCHannel:ACQuire

Back to Functions

CmtNA_ApplyCalibration

Description

Applies calibration to channels. This method must be executed after acquiring all the Standards during calibration session.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ApplyCalibration(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:SAVE

Back to Functions

CmtNA_ApplySimplefiedCalibration

Description

Calculates the calibration coefficients for the simplified 3 or 4 port calibration from the calibration standards measurements when the 3 or 4 port calibration is selected as the calibration type. The calibration type is selected by CmtNA_InitUserCal function.

The simplified 3 port calibration allows omit one THRU measurement. The simplified 4 port calibration allows omit up to three THRU measurements. If full set of calibration standard measurement is made this command behaves like the CmtNA_ApplyCalibration function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_ApplySimplefiedCalibration(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:SIMPlified:SAVE

Back to Functions

CmtNA_CancelCalibration

Description

Clears the measurement data of the calibration standards.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_CancelCalibration(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:CLEar

Back to Functions

CmtNA_ResetCalibration

Description

Clears the calibration coefficient table.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus    CmtNA_ResetCalibration(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:CLEar

Back to Functions

CmtNA_GetCalibrationInfo

Description

Gets the information string of the calibration acting between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalibrationInfo(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 receiverPort,
        ViInt32 sourcePort,
        ViInt32 bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

bufferSize

The size of the array to hold in the *value* parameter.

value

The information string of the calibration acting between the source and receiver ports.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:INFormation? <rcvport>,<srcport>

Back to Functions

CmtNA_PerformUnkThru2PortCal

Description

Completes the full 2-port calibration between the specified ports provided so that each port was calibrated using full 1-port calibration:

- Measures an arbitrary thru between the ports;
- Calculates the error terms Et and El using the unknown thru algorithm;
- Saves the Et and El error terms to the existing calibration getting the full 2-port calibration from the two 1-port calibrations.

If the full 2-port calibration already existed between the specified ports, updates the Et and El error terms.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_PerformUnkThru2PortCal(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 port1,
        ViInt32 port2
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

port1

The first port number from 1 to 4.

port2

The second port number from 1 to 4.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:FULL2:COMPlete <port1>,<port2>

Back to Functions

CmtNA_CalStandardInsert

Description

Inserts the calibration standard into the selected calibration kit. The existing standards with indices greater than or equal to inserted standard are shifted by +1.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_CalStandardInsert(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:INSert

Back to Functions

CmtNA_CalStandardRemove

Description

Deletes the calibration standard into the selected calibration kit. The existing standards with indices greater than inserted standard are shifted by –1.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_CalStandardRemove(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:REMove

Back to Functions

CmtNA_SetCalStandardS1PData

Description

Sets the data array of the data-based calibration standard. The first element of the array is 1 and determines the number of ports of the calibration standard.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalStandardS1PData(
        ViSession vi,
        ViConstString repCapID
        ViInt32 anyBuffersSize,
        ViReal64 stimulusValues[],
        ViReal64 realS11Values[],
        ViReal64 imageS11Values[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *stimulusValues*, *realS11Values*, *imageS11Values* parameters.

stimulusValues

The array of stimulus value.

realS11Values

The array of value real part S11.

imageS11Values

The array of value image part S11.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:DATA <numeric list>

Back to Functions

CmtNA_GetCalStandardS1PData

Description

Gets the data array of the data-based calibration standard. The first element of the array is 1 and determines the number of ports of the calibration standard.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalStandardS1PData(
        ViSession vi,
        ViConstString repCapID
        ViInt32 anyBuffersSize,
        ViReal64 stimulusValues[],
        ViReal64 realS11Values[],
        ViReal64 imageS11Values[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *stimulusValues*, *realS11Values*, *imageS11Values* parameters.

stimulusValues

The array of stimulus value.

realS11Values

The array of value real part S11.

imageS11Values

The array of value image part S11.

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:DATA?

Back to Functions

CmtNA_SetCalStandardS2PData

Description

Sets the data array of the data-based calibration standard. The first element of the array is 2 and determines the number of ports of the calibration standard.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalStandardS2PData(
        ViSession vi,
        ViConstString repCapID
        ViInt32 anyBuffersSize,
        ViReal64 stimulusValues[],
        ViReal64 realS11Values[],
        ViReal64 imageS11Values[],
        ViReal64 realS21Values[],
        ViReal64 imageS21Values[],
        ViReal64 realS12Values[],
        ViReal64 imageS12Values[],
        ViReal64 realS22Values[],
        ViReal64 imageS22Values[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *stimulusValues*, *realS11Values*, *imageS11Values*, *realS21Values*, *imageS21Values*, *realS12Values*, *imageS12Values*, *realS22Values*, *imageS22Values* parameters.

stimulusValues

The array of stimulus value.

realS11Values

The array of value real part S11.

imageS11Values

The array of value image part S11.

realS21Values

The array of value real part S21.

imageS21Values

The array of value image part S21.

realS12Values

The array of value real part S12.

imageS12Values

The array of value image part S12.

realS22Values

The array of value real part S22.

imageS22Values

The array of value image part S22.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:DATA <numeric list>

Back to Functions

CmtNA_GetCalStandardS2PData

Description

Gets the data array of the data-based calibration standard. The first element of the array is 2 and determines the number of ports of the calibration standard.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalStandardS2PData(
        ViSession vi,
        ViConstString repCapID
        ViInt32 anyBuffersSize,
        ViReal64 stimulusValues[],
        ViReal64 realS11Values[],
        ViReal64 imageS11Values[],
        ViReal64 realS21Values[],
        ViReal64 imageS21Values[],
        ViReal64 realS12Values[],
        ViReal64 imageS12Values[],
        ViReal64 realS22Values[],
        ViReal64 imageS22Values[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *stimulusValues*, *realS11Values*, *imageS11Values*, *realS21Values*, *imageS21Values*, *realS12Values*, *imageS12Values*, *realS22Values*, *imageS22Values* parameters.

stimulusValues

The array of stimulus value.

realS11Values

The array of value real part S11.

imageS11Values

The array of value image part S11.

realS21Values

The array of value real part S21.

imageS21Values

The array of value image part S21.

realS12Values

The array of value real part S12.

imageS12Values

The array of value image part S12.

realS22Values

The array of value real part S22.

imageS22Values

The array of value image part S22.

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:DATA?

Back to Functions

CmtNA_SetCalKitOrderOpen

Description

Sets the number of the calibration standard of the open type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderOpen(
        ViSession vi,
        ViInt32 port,
        ViInt32 stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:OPEN <port>,<numeric>

Back to Functions

CmtNA_SetCalKitOrderShort

Description

Sets the number of the calibration standard of the short type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderShort(
        ViSession vi,
        ViInt32 port,
        ViInt32 stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:SHORt <port>,<numeric>

Back to Functions

CmtNA_SetCalKitOrderLoad

Description

Sets the number of the calibration standard of the load type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderLoad(
        ViSession vi,
        ViInt32 port,
        ViInt32 stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:LOAD <port>,<numeric>

Back to Functions

CmtNA_SetCalKitOrderThru

Description

Sets the number of the calibration standard of the thru type, used for the measurement between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderThru(
        ViSession vi,
        ViSession receiverPort,
        ViSession sourcePort,
        ViSession stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:THRU <port1>,<port2>,<numeric>

Back to Functions

CmtNA_SetCalKitOrderTRLLine

Description

Sets the number of the calibration standard of the TRL line type, used for the measurement between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderTRLLine(
        ViSession vi,
        ViSession port,
        ViSession stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:TRLLine <port1>,<port2>,<numeric>

Back to Functions

CmtNA_SetCalKitOrderTRLThru

Description

Sets the number of the calibration standard of the TRL thru type, used for the measurement between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderTRLThru(
        ViSession vi,
        ViSession receiverPort,
        ViSession sourcePort,
        ViSession stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:TRLThru <port1>,<port2>,<numeric>

Back to Functions

CmtNA_SetCalKitOrderTRLReflect

Description

Sets the number of the calibration standard of the TRL Reflect type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetCalKitOrderTRLReflect(
        ViSession vi,
        ViSession port,
        ViSession stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:TRLReflect <port>,<numeric>

Back to Functions

CmtNA_GetCalKitOrderOpen

Description

Gets the number of the calibration standard of the open type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderOpen(
        ViSession vi,
        ViSession port,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:OPEN? <port>

Back to Functions

CmtNA_GetCalKitOrderShort

Description

Gets the number of the calibration standard of the short type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderShort(
        ViSession vi,
        ViSession port,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:SHORt? <port>

Back to Functions

CmtNA_GetCalKitOrderLoad

Description

Gets the number of the calibration standard of the load type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderLoad(
        ViSession vi,
        ViSession port,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:LOAD? <port>

Back to Functions

CmtNA_GetCalKitOrderThru

Description

Gets the number of the calibration standard of the thru type, used for the measurement between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderThru(
        ViSession vi,
        ViSession receiverPort,
        ViSession sourcePort,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:THRU? <port1>,<port2>

Back to Functions

CmtNA_GetCalKitOrderTRLLine

Description

Gets the number of the calibration standard of the TRL line type, used for the measurement between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderTRLLine(
        ViSession vi,
        ViSession receiverPort,
        ViSession sourcePort,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:TRLLine? <port1>,<port2>

Back to Functions

CmtNA_GetCalKitOrderTRLThru

Description

Gets the number of the calibration standard of the TRL thru type, used for the measurement between the source and receiver ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderTRLThru(
        ViSession vi,
        ViSession receiverPort,
        ViSession sourcePort,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:TRLThru? <port1>,<port2>

Back to Functions

CmtNA_GetCalKitOrderTRLReflect

Description

Gets the number of the calibration standard of the TRL Reflect type, used for the measurement of the specified port.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetCalKitOrderTRLReflect(
        ViSession vi,
        ViSession port,
        ViInt32 *stdNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

port

The number of port from which the measurement is performed.

stdNumber

The number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:TRLReflect? <port>

Back to Functions

CmtNA_SetSubclassStdOrder

Description

Sets the subclass used to specify classes of calibration standards by functions:

CmtNA_SetCalKitOrderOpen,

CmtNA_SetCalKitOrderShort,

CmtNA_SetCalKitOrderLoad,

CmtNA_SetCalKitOrderThru,

CmtNA_SetCalKitOrderTRLLine,

CmtNA_SetCalKitOrderTRLThru,

CmtNA_SetCalKitOrderTRLReflect.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetSubclassStdOrder(
        ViSession vi,
        ViSession subclassNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

subclassNumber

The subclass number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:SELect <numeric>

Back to Functions

CmtNA_GetSubclassStdOrder

Description

Gets the subclass used to specify classes of calibration standards by functions:

CmtNA_GetCalKitOrderOpen,

CmtNA_GetCalKitOrderShort,

CmtNA_GetCalKitOrderLoad,

CmtNA_GetCalKitOrderThru,

CmtNA_GetCalKitOrderTRLLine,

CmtNA_GetCalKitOrderTRLThru,

CmtNA_GetCalKitOrderTRLReflect.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetSubclassStdOrder(
        ViSession vi,
        ViInt32 *subclassNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

subclassNumber

The subclass number of the calibration standard.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:ORDer:SELect?

Back to Functions

CmtNA_SetStdSubclassNumber

Description

Sets the subclass number of calibration standard used for measurement by the subsequent function CmtNA_PerformCalibrationAction. If the calibration kit contains several calibration standards of the same type, say SHORTs, this allows to select the particular SHORT. The subclasses must be set in advance by functions:

CmtNA_SetCalKitOrderOpen,

CmtNA_SetCalKitOrderShort,

CmtNA_SetCalKitOrderLoad,

CmtNA_SetCalKitOrderThru,

CmtNA_SetCalKitOrderTRLLine,

CmtNA_SetCalKitOrderTRLThru,

CmtNA_SetCalKitOrderTRLReflect

or in the user interface “Specify Classes”.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetStdSubclassNumber(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *subclassNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

subclassNumber

The subclass number of the calibration standard from 1 to 8.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect[:ACQuire]:SUBClass <numeric>

Back to Functions

CmtNA_GetStdSubclassNumber

Description

Gets the subclass number of calibration standard used for measurement by the subsequent function CmtNA_PerformCalibrationAction. If the calibration kit contains several calibration standards of the same type, say SHORTs, this allows to select the particular SHORT. The subclasses must be set in advance by functions:

CmtNA_GetCalKitOrderOpen,

CmtNA_GetCalKitOrderShort,

CmtNA_GetCalKitOrderLoad,

CmtNA_GetCalKitOrderThru,

CmtNA_GetCalKitOrderTRLLine,

CmtNA_GetCalKitOrderTRLThru,

CmtNA_GetCalKitOrderTRLReflect.

or in the user interface “Specify Classes”.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetStdSubclassNumber(
        ViSession vi,
        ViConstString repCapID,
        ViSession subclassNumber
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

subclassNumber

The subclass number of the calibration standard from 1 to 8.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect[:ACQuire]:SUBClass?

Back to Functions

CmtNA_CalKitRestore

Description

Resets the calibration kit to the factory settings. Restores the predefined calibration kit. Removes the user defined calibration kit.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_CalKitRestore(
        ViSession vi,
        ViSession kitIndex
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

kitIndex

The index of the calibration kit.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:CKIT:RESet

Back to Functions

CmtNA_CalKitSaveToFile

Description

Saves the definition file for the calibration kit.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_CalKitSaveToFile(
        ViSession vi,
        ViSession kitIndex,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

kitIndex

The index of the calibration kit.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \CalKit subdirectory of the application directory will be searched for the file. The calibration kit definition file has *.ckd extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:CKIT<Ck> <string>

Back to Functions

CmtNA_CalKitLoadFromFile

Description

Recalls the definition file for the calibration kit. The file must be saved by the CmtNA_CalKitSaveToFile function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_CalKitLoadFromFile(
        ViSession vi,
        ViSession kitIndex,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

kitIndex

The index of the calibration kit.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \CalKit subdirectory of the application directory will be searched for the file. The calibration kit definition file has *.ckd extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:CKIT<Ck> <string>

Back to Functions

CmtNA_ExportLossTable

Description

Saves the loss compensation table into a file.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ExportLossTable(
        ViSession vi,
        ViConstString repCapID,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Port index i.e. "Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \CalKit subdirectory of the application directory will be searched for the file. The loss compensation file has *.lct extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:PLOSs<Pt> <string>

Back to Functions

CmtNA_ImportLossTable

Description

Recalls the loss compensation file. The file must be saved by the CmtNA_ExportLossTable function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ImportLossTable(
        ViSession vi,
        ViConstString repCapID,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Port index i.e. "Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \CalKit subdirectory of the application directory will be searched for the file. The loss compensation file has *.lct extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:PLOSs<Pt> <string>

Back to Functions

CmtNA_TakePowerCalSweep

Description

Measures the power calibration data for the port using the power meter controlled via USB or USB/GPIB. Calculates calibration coefficients on completion of the measurement, and turns ON the power correction for the port.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_TakePowerCalSweep(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection:COLLect[:ACQuire]

Back to Functions

CmtNA_PowerSensorZeroing

Description

Executes zeroing procedure of the power sensor. Although the Analyzer automatically turns off the RF power during this procedure, it is recommended to disconnect the power sensor from the analyzer port.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_PowerSensorZeroing(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:COMMunicate:PSENsor:ZEROing

Back to Functions

CmtNA_GetPowerCalibrationTable

Description

Gets the power correction array (result of power calibration executed by CmtNA_PerformCalibrationAction function).

The array size is NOP, where NOP is the number of measurement points.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetPowerCalibrationTable(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 buffersSize,
        ViReal64 powerValues[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

buffersSize

The size of the arrays to hold in the *powerValues* parameters.

powerValues

The array of the power correction value.

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection:DATA?

Back to Functions

CmtNA_SetPowerCalibrationTable

Description

Sets the power correction array (result of power calibration executed by CmtNA_PerformCalibrationAction function).

The array size is NOP, where NOP is the number of measurement points.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetPowerCalibrationTable(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 buffersSize,
        ViReal64 powerValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

buffersSize

The size of the arrays to hold in the *powerValues* parameters.

powerValues

The array of the power correction value.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection:DATA <numeric list>

Back to Functions

CmtNA_GetPowerLossCompensationTable

Description

Gets the loss compensation table used when the power calibration is executed by CmtNA_PerformCalibrationAction function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetPowerLossCompensationTable(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 anyBuffersSize,
        ViReal64 frequencyValues[],
        ViReal64 lossValues[],
        ViInt32 *returnSize
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *frequencyValues*, *lossValues* parameters.

frequencyValues

The array of frequency value.

lossValues

The array of the loss compensation value (from –100 to +100 dB).

returnSize

The actual number of elements filled by the query.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection:COLLect:TABLe:LOSS:DATA?

Back to Functions

CmtNA_SetPowerLossCompensationTable

Description

Sets the loss compensation table used when the power calibration is executed by CmtNA_PerformCalibrationAction function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetPowerLossCompensationTable(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 anyBuffersSize,
        ViReal64 frequencyValues[],
        ViReal64 lossValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

anyBuffersSize

The size of the arrays to hold in the *frequencyValues*, *lossValues* parameters.

frequencyValues

The array of value of frequency.

lossValues

The array of value of the loss compensation (from –100 to +100 dB).

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection:COLLect:TABLe:LOSS:DATA <numeric list>

Back to Functions

CmtNA_MeasurePortExtensionShort

Description

Performs measurement of the standard "SHORT", automatically calculates and sets the parameters of the Port Extension.

When two consecutive measurements of "SHORT" are performed the results of these measurements are averaged.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurePortExtensionShort(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:MEASure {SHORt|OPEN}

SENSe<Ch>:CORRection:EXTension:AUTO:PORT<Pt> {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension:AUTO:PORT<Pt>?

Back to Functions

CmtNA_MeasurePortExtensionOpen

Description

Performs measurement of the standard "OPEN", automatically calculates and sets the parameters of the Port Extension.

When two consecutive measurements of "OPEN" are performed the results of these measurements are averaged.

It is recommended to use CmtNA_WaitForOperationComplete function to check if operation is completed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MeasurePortExtensionOpen(
        ViSession vi,
        ViConstString repCapID,
        ViSession portNum
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

portNum

The port of the active channel.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:MEASure {SHORt|OPEN}

SENSe<Ch>:CORRection:EXTension:AUTO:PORT<Pt> {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension:AUTO:PORT<Pt>?

Back to Functions

CmtNA_ExecuteAutoOrientation

Description

Executes the Auto-Orientation procedure of the AutoCal Module. The AutoCal Module must be connected to the ports of Analyzer.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_ExecuteAutoOrientation(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe:CORRection:COLLect:ECAL:ORIentation:EXECute

Back to Functions

CmtNA_ExecuteAutoCalCalibration

Description

Executes 1-port calibration of the specified port of specified channel or full 2-port calibration between 2 specified ports of specified channel or one path 2–port calibration between 2 specified ports of specified channel using the AutoCal module.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_ExecuteAutoCalCalibration(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 autoCalType,
        ViInt32 receiverPort,
        ViInt32 sourcePort
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

autoCalType

The type AutoCal Module calibration.

Default Values

Name	Description
CMTNA_AUTOCAL_1PORT	1-port calibration
CMTNA_AUTOCAL_2PORT	Full 2-port calibration
CMTNA_AUTOCAL_ONE_PATH_2PORT	One path 2–port calibration

receiverPort

The number of port from which the measurement is performed.

sourcePort

The number of port which is the source.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ECAL:SOLT1 <port>

SENSe<Ch>:CORRection:COLLect:ECAL:SOLT2 <port1>,<port2>

SENSe<Ch>:CORRection:COLLect:ECAL:ERESponse <rcvport>,<srcport>

Back to Functions

CmtNA_ExecuteAutoCalNPortCalibration

Description

Executes 1-port calibration of the specified port of specified channel or full 2, 3, 4-port calibration between the specified 2, 3, 4 ports of specified channel or one path 2–port calibration between the specified 2 ports of specified channel using the AutoCal module.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ExecuteAutoCalNPortCalibration(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 autoCalType,
        ViInt32 portList[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

autoCalType

The type AutoCal Module calibration.

Default Values

Name	Description
CMTNA_AUTOCAL_1PORT	1-port calibration
CMTNA_AUTOCAL_2PORT	Full 2-port calibration
CMTNA_AUTOCAL_3PORT	Full 3-port calibration
CMTNA_AUTOCAL_4PORT	Full 4-port calibration
CMTNA_AUTOCAL_ONE_PATH_2PORT	One path 2–port calibration

portList

The list of ports.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ECAL:SOLT1 <port>

SENSe<Ch>:CORRection:COLLect:ECAL:SOLT2 <port1>,<port2>

SENSe<Ch>:CORRection:COLLect:ECAL:SOLT3 <port1>,<port2>,<port3>

SENSe<Ch>:CORRection:COLLect:ECAL:SOLT4 <port1>,<port2>,<port3>,<port4>

SENSe<Ch>:CORRection:COLLect:ECAL:ERESponse <rcvport>,<srcport>

Back to Functions

CmtNA_ExecuteAutoCalConfidenceCheck

Description

Executes the confidence check of the calibration coefficients of specified channel using the AutoCal module.

The function sets the AutoCal Module to the special internal state, reads the S-parameters of this state from the AutoCal Module and sets memory traces so that they can be compared with actual measured data. Comparison is carried out visually by the user.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ExecuteAutoCalConfidenceCheck(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ECAL:CCHeck[:ACQuire]

Back to Functions

CmtNA_GetMarkerValue

Description

Gets the response value of the marker.

If the reference marker is turned ON, the values of the markers from 1 to 15 are read out as relative values to the reference marker.

The data include 2 elements:

- <numeric 1> real number in rectangular format, real part in polar and Smith chart formats;

- <numeric 2> 0 in rectangular format, imaginary part in polar and Smith chart formats.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus   CmtNA_GetMarkerValue(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 *realPart,
        ViReal64 *imagePart
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

realPart

The real number in rectangular format, real part in polar and Smith chart formats.

imagePart

0 in rectangular format, imaginary part in polar and Smith chart formats.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:Y?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:Y?

Back to Functions

CmtNA_MarkerFunctionExecute

Description

Executes the marker search according to the specified criterion. The type of the marker search is set by CMTNA_ATTR_MARKER_SEARCH_TYPE.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_MarkerFunctionExecute(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:EXECute

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:EXECute

Back to Functions

CmtNA_QueryMarkerMathStatistics

Description

Gets the math statistics values.

The statistics function is applied either over the whole range, or within the range specified by CMTNA_ATTR_MARKER_MATH_STATISTICS (the range limits are determined by two markers).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_QueryMarkerMathStatistics(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 *mean,
        ViReal64 *sdev,
        ViReal64 *peakpeak
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

mean

The mean value.

sdev

The standard deviation value.

peakpeak

The peak–to–peak value (difference between the maximum value and the minimum value).

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MSTatistics:DATA?

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DATA?

Back to Functions

CmtNA_QueryMarkerMathBandwidth

Description

Gets the bandwidth search result.

The bandwidth search can performed relatively to the marker, or relatively to the absolute maximum value of the trace (in this case the number of the marker is ignored), that is set by CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_REF.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_QueryMarkerMathBandwidth(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 *bandwidth,
        ViReal64 *centerFreq,
        ViReal64 *qvalue,
        ViReal64 *loss
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th

trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

bandwidth

The bandwidth value.

centerFreq

The center frequency value.

qvalue

The Q value.

loss

The loss value.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:BWIDth:DATA?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:BWIDth:DATA?

Back to Functions

CmtNA_QueryMarkerMathFlatness

Description

Gets FLATNESS function data array. The FLATNESS function is applied within the range determined by two markers.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_QueryMarkerMathFlatness(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 *span,
        ViReal64 *gain,
        ViReal64 *slope,
        ViReal64 *flatness
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

span

The span value.

gain

The gain value.

slope

The slope value.

flatness

The flatness value.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:DATA?

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:DATA?

Back to Functions

CmtNA_MarkerFunctionsMarkerStart

Description

Sets the value of the specified item to the value of the position of the marker.

Sweep start value is set to the stimulus value of the marker position.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MarkerFunctionsMarkerStart(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index, Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:SET <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:SET <char>

Back to Functions

CmtNA_MarkerFunctionsMarkerStop

Description

Sets the value of the specified item to the value of the position of the marker.

Sweep stop value is set to the stimulus value of the marker position.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MarkerFunctionsMarkerStop(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index, Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:SET <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:SET <char>

Back to Functions

CmtNA_MarkerFunctionsMarkerCenter

Description

Sets the value of the specified item to the value of the position of the marker.

Sweep center value is set to the stimulus value of the marker position.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MarkerFunctionsMarkerCenter(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index, Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:SET <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:SET <char>

Back to Functions

CmtNA_MarkerFunctionsMarkerRefValue

Description

Sets the value of the specified item to the value of the position of the marker.

Reference value is set to the response value of the marker position.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MarkerFunctionsMarkerRefValue(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index, Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:SET <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:SET <char>

Back to Functions

CmtNA_MarkerFunctionsMarkerDelay

Description

Sets the value of the specified item to the value of the position of the marker.

Delay value is set to the response value of the marker position.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MarkerFunctionsMarkerDelay(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index, Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:SET <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:SET <char>

Back to Functions

CmtNA_QueryMarkerTableData

Description

Gets the data array of all turned ON markers.

The array size is 3N + 1, where N is the number of turned ON markers including the reference marker. If the reference marker is turned ON the last three elements of the array contain the reference marker data and the rest elements of array contain the relative values.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_QueryMarkerTableData(
        ViSession vi,
        ViConstString repCapID
        ViInt32 *markerONCount,
        ViInt32 anyBuffersSize,
        ViReal64 stimulusValues[],
        ViReal64 realValues[],
        ViReal64 imageValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so

on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

markerONCount

The number of turned ON markers including the reference marker (N).

anyBuffersSize

The size of the arrays to hold in the *stimulusValues*, *realValues*, *imageValues* parameters.

stimulusValues

The stimulus value of the n–th marker.

realValues

The real data in rectangular format, real part in polar and Smith chart formats of the n–th marker.

imageValues

0 in rectangular format, imaginary part in polar and Smith chart formats of the n–th marker

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:DATA?

CALCulate<Ch>:TRACe<Tr>:MARKer:DATA?

Back to Functions

CmtNA_ConfigureTimeDomainStartStop

Description

Configures the start and stop values for the time-domain analysis. This function configures the start and stop points in terms of time.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureTimeDomainStartStop(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 startPoint,
        ViReal64 stopPoint
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

startPoint

The time domain start value.

See more details: CMTNA_ATTR_TIME_DOMAIN_START.

stopPoint

The time domain stop value.

See more details: CMTNA_ATTR_TIME_DOMAIN_STOP.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:STARt <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STARt <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STOP <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STOP <time>

Back to Functions

CmtNA_ConfigureTimeDomainCenterSpan

Description

Configures the center and span values for the time-domain analysis. This function configures the center and span points in terms of time.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureTimeDomainCenterSpan(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 centerPoint,
        ViReal64 span
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

centerPoint

The time domain center value.

See more details: CMTNA_ATTR_TIME_DOMAIN_CENTER.

span

The time domain span value.

See more details: CMTNA_ATTR_TIME_DOMAIN_SPAN.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:CENTer <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:CENTer <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:SPAN <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:SPAN <time>

Back to Functions

CmtNA_TimeDomainSetFrequencyLowPass

Description

Changes the frequency range to match the low-pass type of the time domain transformation function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_TimeDomainSetFrequencyLowPass(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:LPFRequency

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:LPFRequency

Back to Functions

CmtNA_ConfigureGatingStartStop

Description

Configures the start and stop values of the gating function. This function configures the start and stop points in terms of time.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureGatingStartStop(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 gatingStart,
        ViReal64 gatingStop
        ViInt32 type
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

gatingStart

The gating function start value.

See more details: CMTNA_ATTR_TIME_DOMAIN_GATING_START.

gatingStop

The gating function stop value.

See more details: CMTNA_ATTR_TIME_DOMAIN_GATING_STOP.

type

The gating function type.

See more details: CMTNA_ATTR_TIME_DOMAIN_GATING_TYPE.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STARt <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STOP <time>

Back to Functions

CmtNA_ConfigureGatingCenterSpan

Description

Configures the center and span values of the gating function. This function configures the center and span points in terms of time.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ConfigureGatingCenterSpan(
        ViSession vi,
        ViConstString repCapID,
        ViReal64 gatingCenter,
        ViReal64 gatingSpan
        ViInt32 type
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

gatingStart

The gating function center value.

See more details: CMTNA_ATTR_TIME_DOMAIN_GATING_CENTER.

gatingStop

The gating function span value.

See more details: CMTNA_ATTR_TIME_DOMAIN_GATING_SPAN.

type

The gating function type.

See more details: CMTNA_ATTR_TIME_DOMAIN_GATING_TYPE.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:CENTer <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SPAN <time>

Back to Functions

CmtNA_SetLimitTestData

Description

Sets the data array, which is the limit line in the limit test function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetLimitTestData(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 segmentCount,
        ViInt32 limitLineType[],
        ViReal64 beginLimitArgument[],
        ViReal64 endLimitArgument[],
        ViReal64 beginLimitValues[],
        ViReal64 endLimitValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

segmentCount

The number of limit line segments N is from 0 to 100. Setting 0 clears the limit line.

limitLineType

The type of the n–th limit line segment.

Defined Values

Name	Description
CMTNA_ANALYSIS_LIMIT_LINE_OFF	Off
CMTNA_ANALYSIS_LIMIT_LINE_MAX	Upper limit
CMTNA_ANALYSIS_LIMIT_LINE_MIN	Lower limit
CMTNA_ANALYSIS_LIMIT_LINE_SINGLE	Single Point limit

beginLimitArgument

The stimulus value in the start point of the n–th segment.

endLimitArgument

The stimulus value in the end point of the n–th segment.

beginLimitValues

The response value in the start point of the n–th segment.

endLimitValues

The response value in the end point of the n–th segment.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:DATA <numeric list>

CALCulate<Ch>:TRACe<Tr>:LIMit:DATA <numeric list>

Back to Functions

CmtNA_GetLimitTestData

Description

Gets the data array, which is the limit line in the limit test function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetLimitTestData(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *segmentCount,
        ViInt32 anyBuffersSize,
        ViInt32 limitLineType[],
        ViReal64 beginLimitArgument[],
        ViReal64 endLimitArgument[],
        ViReal64 beginLimitValues[],
        ViReal64 endLimitValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index

and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

segmentCount

The number of limit line segments N is from 0 to 100.

anyBuffersSize

The size of the arrays to hold in the *beginLimitArgument*, *endLimitArgument*, *beginLimitValues*, *endLimitValues* parameters.

limitLineType

The type of the n–th limit line segment.

Defined Values

Name	Description
CMTNA_ANALYSIS_LIMIT_LINE_OFF	Off
CMTNA_ANALYSIS_LIMIT_LINE_MAX	Upper limit
CMTNA_ANALYSIS_LIMIT_LINE_MIN	Lower limit
CMTNA_ANALYSIS_LIMIT_LINE_SINGLE	Single Point limit

beginLimitArgument

The stimulus value in the start point of the n–th segment.

endLimitArgument

The stimulus value in the end point of the n–th segment.

beginLimitValues

The response value in the start point of the n–th segment.

endLimitValues

The response value in the end point of the n–th segment.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:DATA?

CALCulate<Ch>:TRACe<Tr>:LIMit:DATA?

Back to Functions

CmtNA_SaveLimitTable

Description

Gets the data array, which is the limit line in the limit test function.

The function is used for the active trace of the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveLimitTable(
        ViSession vi,
        ViConstString repCapID,
        ViString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \Limit subdirectory of the application directory will be searched for the file. The file has *.lim extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:LIMit <string>

Back to Functions

CmtNA_RestoreLimitTable

Description

Recalls the limit table file. The file must be saved by CmtNA_SaveLimitTable function.

The function is used for the active trace of the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_RestoreLimitTable(
        ViSession vi,
        ViConstString repCapID,
        ViString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \Limit subdirectory of the application directory will be searched for the file. The file has *.lim extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:LIMit <string>

Back to Functions

CmtNA_MarkerSetLimitLineResponseOffset

Description

Sets the value of the limit line offset along Y–axis to the active marker value.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_MarkerSetLimitLineResponseOffset(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:OFFSet:MARKer

CALCulate<Ch>:TRACe<Tr>:LIMit:OFFSet:MARKer

Back to Functions

CmtNA_GetLimitTestStatus

Description

Gets the limit test result.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetLimitTestStatus(
        ViSession vi,
        ViConstString repCapID,
        ViBoolean *result
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

result

The limit test result.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:FAIL?

CALCulate<Ch>:TRACe<Tr>:LIMit:FAIL?

Back to Functions

CmtNA_GetLimitTestReport

Description

Gets the data array, which is the stimulus values of the measurement points that failed the limit test.

The array size is got by CMTNA_ATTR_ANALYSIS_LIMIT_TEST_POINTS.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetLimitTestReport(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *points,
        ViInt32 bufferSize,
        ViReal64 dataValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

points

The number of the measurement points that failed the limit test.

bufferSize

The size of the array to hold in the *dataValues* parameter.

dataValues

The stimulus values of the measurement points that failed the limit test.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:REPort[:DATA]?

CALCulate<Ch>:TRACe<Tr>:LIMit:REPort[:DATA]?

Back to Functions

CmtNA_GetLimitTestReportAll

Description

Gets the data array, which is the limit test result.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetLimitTestReportAll(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 stimulusBufferSize,
        ViReal64 stimulusValues[],
        ViInt32 resultBufferSize,
        ViInt32 resultValues[],
        ViInt32 upperLimitBufferSize,
        ViReal64 upperLimitValues[],
        ViInt32 lowerLimitBufferSize,
        ViReal64 lowerLimitValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so

on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

stimulusBufferSize

The size of the array to hold in the *stimulusValues* parameter.

stimulusValues

The stimulus value in the n–th point.

resultBufferSize

The size of the array to hold in the *resultValues* parameter.

resultValues

The limit test result in the n–th point.

–1: No limit

0: Fail

1: Pass

upperLimitBufferSize

The size of the array to hold in the *upperLimitValues* parameter.

upperLimitValues

The upper limit value in the n–th point (0 – if there is no limit).

lowerLimitBufferSize

The size of the array to hold in the *lowerLimitValues* parameter.

lowerLimitValues

The lower limit value in the n–th point (0 – if there is no limit).

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:REPort:ALL?

CALCulate<Ch>:TRACe<Tr>:LIMit:REPort:ALL?

Back to Functions

CmtNA_SetRippleLimitData

Description

Sets the data array, which is the limit line for the ripple limit function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetRippleLimitData(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 segmentCount,
        ViInt32 rippleLimitType[],
        ViReal64 beginLimitArgument[],
        ViReal64 endLimitArgument[],
        ViReal64 limitValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

segmentCount

The number of limit line segments N is the integer from 0 to 12. Setting 0 clears the limit line.

rippleLimitType

The type of the n–th limit line segment.

Defined Values

Name	Description
CMTNA_ANALYSIS_RIPPLE_LIMIT_OFF	Off
CMTNA_ANALYSIS_RIPPLE_LIMIT_ON	On

beginLimitArgument

The stimulus value in the beginning point of the n–th segment.

endLimitArgument

The stimulus value in the end point of the n–th segment.

LimitValues

The ripple limit value of the n–th segment.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit:DATA <numeric list>

CALCulate<Ch>:TRACe<Tr>:RLIMit:DATA <numeric list>

Back to Functions

CmtNA_GetRippleLimitData

Description

Gets the data array, which is the limit line for the ripple limit function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetRippleLimitData(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *segmentCount,
        ViInt32 limitType[],
        ViReal64 beginLimitArgument[],
        ViReal64 endLimitArgument[],
        ViReal64 limitValues[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

segmentCount

The number of limit line segments N is the integer from 0 to 12. Setting 0 clears the limit line.

limitType

The type of the n–th limit line segment.

Defined Values

Name	Description
CMTNA_ANALYSIS_RIPPLE_LIMIT_OFF	Off
CMTNA_ANALYSIS_RIPPLE_LIMIT_ON	On

beginLimitArgument

The stimulus value in the beginning point of the n–th segment.

endLimitArgument

The stimulus value in the end point of the n–th segment.

LimitValues

The ripple limit value of the n–th segment.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit:DATA?

CALCulate<Ch>:TRACe<Tr>:RLIMit:DATA?

Back to Functions

CmtNA_SaveRippleLimitTable

Description

Saves the ripple limit table into a file.

The function is used for the active trace of the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveRippleLimitTable(
        ViSession vi,
        ViConstString repCapID,
        ViString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \Limit subdirectory of the application directory will be searched for the file. The file has *.lim extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:RLIMit <string>

Back to Functions

CmtNA_RestoreRippleLimitTable

Description

Recalls the ripple limit table file. The file must be saved by CmtNA_SaveRippleLimitTable function.

The function is used for the active trace of the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_RestoreRippleLimitTable(
        ViSession vi,
        ViConstString repCapID,
        ViString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \Limit subdirectory of the application directory will be searched for the file. The file has *.lim extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:RLIMit <string>

Back to Functions

CmtNA_GetRippleLimitTestStatus

Description

Gets the ripple limit test result.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetRippleLimitTestStatus(
        ViSession vi,
        ViConstString repCapID,
        ViBoolean *result
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

result

The ripple limit test result.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit:FAIL?

CALCulate<Ch>:TRACe<Tr>:RLIMit:FAIL?

Back to Functions

CmtNA_GetRippleLimitTestReport

Description

Gets the data array, which is the ripple limit test result.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetRippleLimitTestReport(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *bandsCount,
        ViInt32 anyBuffersSize,
        ViInt32 bandNumbers[],
        ViReal64 bandRippleValues[],
        ViInt32 bandResults[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

bandsCount

N total number of the bands.

anyBuffersSize

The size of the arrays to hold in the *bandNumbers*, *bandRippleValues*, *bandResults* parameters.

bandNumbers

n number of the band

bandRippleValues

The ripple value in the n–th band.

bandResults

The ripple limit test result in the n–th band: 0- Pass, 1- Fail.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit:REPort[:DATA]?

CALCulate<Ch>:TRACe<Tr>:RLIMit:REPort[:DATA]?

Back to Functions

CmtNA_SaveStateToFile

Description

Saves the Analyzer state into a file.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveStateToFile(
        ViSession vi,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \State subdirectory of the application directory will be searched for the file. The state file has *.sta extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe[:STATe] <string>

Back to Functions

CmtNA_RecallFromFile

Description

Recalls the specified Analyzer state file. The file must be saved by CmtNA_SaveStateToFile function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_RecallFromFile(
        ViSession vi,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \State subdirectory of the application directory will be searched for the file. The Analyzer state file has *.sta extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe[:STATe] <string>

Back to Functions

CmtNA_SaveChannelToRegister

Description

Saves the Analyzer state of the items set for the active channel into one of the four memory registers.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveChannelToRegister(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 stateRegister
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The memory register for saving the Analyzer state of the items set for the active channel into one of the four memory registers

stateRegister

The memory register for saving the file.

Defined Values

Name	Description
CMTNA_CHANNEL_STATE_REGISTER_A	Save to register A
CMTNA_CHANNEL_STATE_REGISTER_B	Save to register B
CMTNA_CHANNEL_STATE_REGISTER_C	Save to register C
CMTNA_CHANNEL_STATE_REGISTER_D	Save to register D

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:CHANnel[:STATe] <char>

Back to Functions

CmtNA_ClearRegisterStates

Description

Clears the memory of the channel state saved by CmtNA_SaveChannelToRegister function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_ClearRegisterStates(
        ViSession vi,
        ViConstString repCapID
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:CHANnel:CLEar

Back to Functions

CmtNA_RecallChannelFromRegister

Description

Recalls the Analyzer state for the active channel. The file must be saved in one of the four memory registers by CmtNA_SaveChannelToRegister function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_RecallChannelFromRegister(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 stateRegister
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

stateRegister

The memory register whence recall the Analyzer state for the active channel.

Defined Values

Name	Description
CMTNA_CHANNEL_STATE_REGISTER_A	Recall from register A
CMTNA_CHANNEL_STATE_REGISTER_B	Recall from register B
CMTNA_CHANNEL_STATE_REGISTER_C	Recall from register C
CMTNA_CHANNEL_STATE_REGISTER_D	Recall from register D

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:CHANnel[:STATe] <char>

Back to Functions

CmtNA_SaveTraceData

Description

Saves the CSV formatted data into a file.

The function is used for the active trace of the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveTraceData(
        ViSession vi,
        ViConstString repCapID,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \CSV subdirectory of the application directory will be searched for the file. The file has *.csv extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:FDATa <string>

Back to Functions

CmtNA_SetTouchstoneFileType

Description

Sets the Touchstone file type and the port numbers, when saving S–parameters by CmtNA_SaveTouchstoneFile function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SetTouchstoneFileType(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 touchstoneType,
        ViInt32 ports[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

touchstoneType

The type of Touchstone file.

ports

The array of port numbers. The number of ports must match *touchstoneType*.

Defined Values

Name	Description	Number of ports
CMTNA_SAVE_TOUCHSTONE_FILE_S1P	The file type S1P	1
CMTNA_SAVE_TOUCHSTONE_FILE_S2P	The file type S2P	2
CMTNA_SAVE_TOUCHSTONE_FILE_S3P	The file type S3P	3
CMTNA_SAVE_TOUCHSTONE_FILE_S4P	The file type S4P	4

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:SNP:TYPE:S1P <port>

MMEMory:STORe:SNP:TYPE:S2P <port1>,<port2>

MMEMory:STORe:SNP:TYPE:S3P <port1>,<port2>,<port3>

MMEMory:STORe:SNP:TYPE:S4P <port1>,<port2>,<port3>,<port4>

Back to Functions

CmtNA_GetTouchstoneFileType

Description

Gets the Touchstone file type and the port numbers, when saving S–parameters by CmtNA_SaveTouchstoneFile function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_GetTouchstoneFileType(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 *touchstoneType,
        ViInt32  bufferSize,
        ViInt32 ports[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

touchstoneType

The type of Touchstone file.

bufferSize

The size of the arrays to hold in the *ports* parameter.

ports

The array of port numbers. The number of ports must match *touchstoneType*.

Defined Values

Name	Description	Number of ports
CMTNA_SAVE_TOUCHSTONE_FILE_S1P	The file type S1P	1
CMTNA_SAVE_TOUCHSTONE_FILE_S2P	The file type S2P	2
CMTNA_SAVE_TOUCHSTONE_FILE_S3P	The file type S3P	3
CMTNA_SAVE_TOUCHSTONE_FILE_S4P	The file type S4P	4

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:SNP:TYPE:S1P?

MMEMory:STORe:SNP:TYPE:S2P?

MMEMory:STORe:SNP:TYPE:S3P?

MMEMory:STORe:SNP:TYPE:S4P?

Back to Functions

CmtNA_SaveTouchstoneFile

Description

Saves the measured S–parameters of the active channel into a Touchstone file. The file type (1 port to 4 port) is set by CmtNA_SetTouchstoneFileType function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveTouchstoneFile(
        ViSession vi,
        ViConstString repCapID,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \FixtureSim subdirectory of the application directory will be searched for the file. The file has *.sNp extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:SNP[:DATA] <string>

Back to Functions

CmtNA_LoadTouchstoneFile

Description

Loads the Touchstone file with the specified name to the measured S–parameters of the active channel. The Touchstone file types 1, 2, 3 or 4 port (file extensions s1p, s2p, s3p or s4p) are supported. On completion of the command, the channel goes to the hold state.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_LoadTouchstoneFile(
        ViSession vi,
        ViConstString repCapID,
        ViInt32 destination,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier.

Must be VI_NULL or an empty string if loads the Touchstone file to the measured S–parameters of the active channel.

If loads the Touchstone file with the specified name to the memory trace, the physical names are supported along with the corresponding Measurement index i.e. "Measurement1" and so on. For valid values, see the Accessing Repeated Capabilities.

destination

The destination.

Defined Values

Name	Description
CMTNA_UPLOAD_TOUCHSTONE_FILE_TO_ACTIVE_TRACE_MEMORY	Loads to the memory trace
CMTNA_UPLOAD_TOUCHSTONE_FILE_TO_S_PARAMETERS	Loads to S-parameters

fileName

The file path into which to save the data. If the full path of the file is not specified, the \FixtureSim subdirectory of the application directory will be searched for the file. The file has *.sNp extension by default.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:LOAD:SNP[:DATA] <string>

MMEMory:LOAD:SNP:TRACe<Tr>:MEMory <string>

Back to Functions

CmtNA_SystemPreset

Description

Resets the Analyzer to the factory settings.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SystemPreset(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:PRESet

Back to Functions

CmtNA_PrintOut

Description

Prints out the image displayed on the screen without previewing.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_PrintOut(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

HCOPy[:IMMediate]

Back to Functions

CmtNA_AbortPrint

Description

Aborts the printout.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_AbortPrint(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

HCOPy:ABORt

Back to Functions

CmtNA_SaveImage

Description

Saves the display image in BMP or PNG format into a file.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus  CmtNA_SaveImage(
        ViSession vi,
        ViConstString fileName
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

fileName

The file path into which to save the data. If the full path of the file is not specified, the \Image subdirectory of the application directory will be searched for the file. If the file has *.png extension, the file had PNG format, in all the other cases the file has BMP format.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

MMEMory:STORe:IMAGe <string>

Back to Functions

CmtNA_TestBeepComplete

Description

Generates a beep to notify of the completion of the operation.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_TestBeepComplete(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:BEEPer:COMPlete:IMMediate

Back to Functions

CmtNA_TestBeepWarning

Description

Generates a beep to notify of warning.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_TestBeepWarning(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:BEEPer:WARNing:IMMediate

Back to Functions

CmtNA_SystemShow

Description

Restores the analyzer main window hidden by CmtNA_SystemHide function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_SystemShow(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:SHOW

Back to Functions

CmtNA_SystemHide

Description

Minimizes the analyzer main window removing it from the desktop.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
ViStatus CmtNA_SystemHide(
        ViSession vi
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

Return value

Success or failure code.

Remarks

For detailed parameter usage see: Initializing the IVI-C Driver.

SCPI Command

SYSTem:HIDE

Back to Functions

Attributes By Name

CMTNA_ATTR_STIMULUS_FREQUENCY_START

Description

Sets/Gets the start value of the frequency sweep range. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_FREQUENCY_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus    CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_FREQUENCY_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:FREQuency:STARt <frequency>

SENSe<Ch>:FREQuency:STARt?

Back to Attributes

CMTNA_ATTR_STIMULUS_FREQUENCY_STOP

Description

Sets/Gets the stop value of the frequency sweep range. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_FREQUENCY_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_FREQUENCY_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:FREQuency:STOP <frequency>

SENSe<Ch>:FREQuency:STOP?

Back to Attributes

CMTNA_ATTR_STIMULUS_FREQUENCY_CENTER

Description

Sets/Gets the center value of the frequency sweep range. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_FREQUENCY_CENTER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_FREQUENCY_CENTER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:FREQuency:CENTer <frequency>

SENSe<Ch>:FREQuency:CENTer?

Back to Attributes

CMTNA_ATTR_STIMULUS_FREQUENCY_SPAN

Description

Sets/Gets the span value of the frequency sweep range. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_FREQUENCY_SPAN

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_FREQUENCY_SPAN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:FREQuency:SPAN <frequency>

SENSe<Ch>:FREQuency:SPAN?

Back to Attributes

CMTNA_ATTR_STIMULUS_POINTS

Description

Sets/Gets the number of sweep points.

Default value: 201

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_POINTS

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CmtNA_ATTR_STIMULUS_POINTS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:SWEep:POINts <numeric>

SENSe<Ch>:SWEep:POINts?

Back to Attributes

CMTNA_ATTR_STIMULUS_SWEEP_TYPE

Description

Sets/Gets the sweep type of channel.

Default value: CMTNA_ATTR_SWEEP_TYPE_LIN_FREQUENCY

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_SWEEP_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CmtNA_ATTR_STIMULUS_SWEEP_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_SWEEP_TYPE_LIN_FREQUENCY	Linear frequency sweep
CMTNA_ATTR_SWEEP_TYPE_LOG_FREQUENCY	Logarithmic frequency sweep
CMTNA_ATTR_SWEEP_TYPE_SEGMENT	Segment frequency sweep
CMTNA_ATTR_SWEEP_TYPE_POWER	Power sweep

SCPI Command

SENSe<Ch>:SWEep:TYPE {LINear|LOGarithmic|SEGMent|POWer}

SENSe<Ch>:SWEep:TYPE?

Back to Attributes

CMTNA_ATTR_STIMULUS_IF_BANDWIDTH

Description

Sets/Gets the bandwidth of the digital IF filter to be used in the measurement. IF Bandwidth is in Hz. The list of valid IF Bandwidths is different depending on the analyzer model. If an invalid number is specified, the analyzer will round up to the closest valid number. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Default value: 10000

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_IF_BANDWIDTH

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CmtNA_ATTR_STIMULUS_IF_BANDWIDTH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:BANDwidth[:RESolution] <frequency>

SENSe<Ch>:BANDwidth[:RESolution]?

SENSe<Ch>:BWIDth[:RESolution] <frequency>

SENSe<Ch>:BWIDth[:RESolution]?

Back to Attributes

CMTNA_ATTR_STIMULUS_OUTPUT_POWER

Description

Sets/Gets the power level for the frequency sweep type. This value is expressed in decibels above 1 milliwatt (dBm).

Unit: dBm (decibels above 1 milliwatt)

Default value: 0 dBm

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_OUTPUT_POWER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_OUTPUT_POWER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer[:LEVel][:IMMediate][:AMPLitude] <power>

SOURce<Ch>:POWer[:LEVel][:IMMediate][:AMPLitude]?

Back to Attributes

CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT_COUPLE

Description

Turns ON/OFF the port power couple.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT_COUPLE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT_COUPLE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:PORT:COUPle {OFF|ON|0|1}

SOURce<Ch>:POWer:PORT:COUPle?

Back to Attributes

CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT

Description

Sets/Gets the power level of port for the frequency sweep type when the port couple feature is set to OFF. This value is expressed in decibels above 1 milliwatt (dBm).

Unit: dBm (decibels above 1 milliwatt)

Default value: 0 dBm

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_OUTPUT_POWER_PORT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>[:LEVel][:IMMediate][:AMPLitude] <power>

SOURce<Ch>:POWer:PORT<Pt>[:LEVel][:IMMediate][:AMPLitude]?

Back to Attributes

CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE

Description

Sets/Gets the power slope value for the frequency sweep. This value is expressed in decibels/gigahertz (dB/GHz).

Unit: dB/GHz (decibel/gigahertz)

Default value: 0 dBm

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer[:LEVel]:SLOPe[:DATA] <numeric>

SOURce<Ch>:POWer[:LEVel]:SLOPe[:DATA]?

Back to Attributes

CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE_STATE

Description

Turns ON/OFF the power slope. The power slope is valid for the frequency sweep type: Linear, Logarithimic, Segment.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE_STATE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_OUTPUT_POWER_SLOPE_STATE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer[:LEVel]:SLOPe:STATe {OFF|ON|0|1}

SOURce<Ch>:POWer[:LEVel]:SLOPe:STATe?

Back to Attributes

CMTNA_ATTR_STIMULUS_OUTPUT_POWER_RFOUT

Description

Turns ON/OFF the RF signal output. Measurements cannot be performed when the RF signal output is turned OFF.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_OUTPUT_POWER_RFOUT

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_OUTPUT_POWER_RFOUT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

OUTPut[:STATe] {OFF|ON|0|1}

OUTPut[:STATe]?

Back to Attributes

CMTNA_ATTR_STIMULUS_POWER_CW_FREQUENCY

Description

Sets/Gets the fixed frequency value when the power sweep type is selected. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Default value: The minimum frequency limit of the analyzer.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_POWER_CW_FREQUENCY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_POWER_CW_FREQUENCY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:FREQuency[:CW] <frequency>

SENSe<Ch>:FREQuency[:FIXed] <frequency>

SENSe<Ch>:FREQuency[:CW]?

SENSe<Ch>:FREQuency[:FIXed]?

Back to Attributes

CMTNA_ATTR_STIMULUS_POWER_START

Description

Sets/Gets the power sweep start value when the power sweep type is active. The value is expressed in decibels above 1 milliwatt (dBm).

Unit: dBm (decibels above 1 milliwatt)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_POWER_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_POWER_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:STARt <power>

SOURce<Ch>:POWer:STARt?

Back to Attributes

CMTNA_ATTR_STIMULUS_POWER_STOP

Description

Sets/Gets the power sweep stop value when the power sweep type is active. The value is expressed in decibels above 1 milliwatt (dBm).

Unit: dBm (decibels above 1 milliwatt)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_POWER_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_POWER_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:STOP <power>

SOURce<Ch>:POWer:STOP?

Back to Attributes

CMTNA_ATTR_STIMULUS_POWER_CENTER

Description

Sets/Gets the center value of the power sweep type. The value is expressed in decibels above 1 milliwatt (dBm).

Unit: dBm (decibels above 1 milliwatt)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_POWER_CENTER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_POWER_CENTER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:CENTer <power>

SOURce<Ch>:POWer:CENTer?

Back to Attributes

CMTNA_ATTR_STIMULUS_POWER_SPAN

Description

Sets/Gets the power span when the power sweep type is active. The value is expressed in decibels above 1 milliwatt (dBm).

Unit: dBm (decibels above 1 milliwatt)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_POWER_CENTER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_POWER_CENTER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:SPAN <power>

SOURce<Ch>:POWer:SPAN?

Back to Attributes

CMTNA_ATTR_STIMULUS_SEGMENT_DISPLAY_ORDER

Description

Sets/Gets the display method of the graph horizontal axis for the segment sweep.

Default value: CMTNA_ATTR_SEGMENT_FREQUENCY_ORDER

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_SEGMENT_DISPLAY_ORDER

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_SEGMENT_DISPLAY_ORDER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_SEGMENT_FREQUENCY_ORDER	Frequency base (linear frequency axis)
CMTNA_ATTR_SEGMENT_INDEX_ORDER	Order base (linear axis of the point number)

SCPI Command

DISPlay:WINDow<Ch>:X:SPACing <char>

DISPlay:WINDow<Ch>:X:SPACing?

Back to Attributes

CMTNA_ATTR_STIMULUS_SWEEP_MEASURE_DELAY

Description

Sets/Gets the delay before measurement in each measurement point. The value is expressed in seconds (sec).

Unit: sec (second)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_SWEEP_MEASURE_DELAY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_SWEEP_MEASURE_DELAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:SWEep:POINt:TIME <time>

SENSe<Ch>:SWEep:POINt:TIME?

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_MODE

Description

Sets/Gets the trigger mode.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_TRIGGER_MODE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_MODE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_MODE_HOLD	Sweep actuation is off in the channel, trigger signals do not affect the channel
CMTNA_ATTR_TRIGGER_MODE_SINGLE	One sweep actuation occurs with trigger signal detection after the mode has been enabled; after the sweep is complete the channel modes changes to hold
CMTNA_ATTR_TRIGGER_MODE_CONTINUOUS	A sweep actuation occurs every time a trigger signal is detected

SCPI Command

INITiate<Ch>:CONTinuous {OFF|ON|0|1}

INITiate<Ch>:CONTinuous?

INITiate<Ch>[:IMMediate]

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE

Description

Sets/Gets the trigger source.

Default value: CMTNA_ATTR_TRIGGER_SOURCE_INTERNAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_SOURCE_INTERNAL	The next trigger signal is generated by the Analyzer on completion of each sweep
CMTNA_ATTR_TRIGGER_SOURCE_EXTERNAL	The external trigger input is used as a trigger signal source
CMTNA_ATTR_TRIGGER_SOURCE_MANUAL	The trigger signal is generated by pressing the corresponding softkey
CMTNA_ATTR_TRIGGER_SOURCE_BUS	The trigger signal is generated by a command communicated from an external computer from a program controlling the Analyzer via COM/DCOM

SCPI Command

TRIGger[:SEQuence]:SOURce <char>

TRIGger[:SEQuence]:SOURce?

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_STATE

Description

Gets the the current state of the analyzer.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_TRIGGER_STATE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_STATE.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

Defined Values

Name	Description
CMTNA_STIMULUS_TRIGGER_STATE_HOLD	Hold
CMTNA_STIMULUS_TRIGGER_STATE_MEASURE	Measure (sweep in progress)
CMTNA_STIMULUS_TRIGGER_STATE_WAIT	Waiting for trigger

SCPI Command

TRIGger[:SEQuence]:STATus?

Back to Attributes

CMTNA_ATTR_STIMULUS_EXT_TRIGGER_EVENT

Description

Turns ON/OFF the point trigger feature for external trigger source.

Default value: CMTNA_ATTR_EXT_TRIGGER_EVENT_ON_SWEEP

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_EXT_TRIGGER_EVENT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_EXT_TRIGGER_EVENT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_EXT_TRIGGER_EVENT_ON_SWEEP	The external trigger response is the entire sweep
CMTNA_ATTR_EXT_TRIGGER_EVENT_ON_POINT	The external trigger response is the single point

SCPI Command

TRIGger[:SEQuence]:POINt {OFF|ON|0|1}

TRIGger[:SEQuence]:POINt?

Back to Attributes

CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POLARITY

Description

Sets/Gets out the polarity of the external trigger.

Default value: CMTNA_ATTR_EXT_TRIGGER_POLARITY_NEGATIVE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POLARITY

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POLARITY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_EXT_TRIGGER_POLARITY_NEGATIVE	Negative edge
CMTNA_ATTR_EXT_TRIGGER_POLARITY_POSITIVE	Positive edge

SCPI Command

TRIGger[:SEQuence]:EXTernal:SLOPe <char>

TRIGger[:SEQuence]:EXTernal:SLOPe?

Back to Attributes

CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POSITION

Description

Sets/Gets the position of the external trigger.

Default value: CMTNA_ATTR_EXT_TRIGGER_POSITION_BSAMPLING

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POSITION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_EXT_TRIGGER_POSITION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_EXT_TRIGGER_POSITION_BSAMPLING	Before sampling
CMTNA_ATTR_EXT_TRIGGER_POSITION_BSETUP	Before frequency setup

SCPI Command

TRIGger[:SEQuence]:EXTernal:POSition <char>

TRIGger[:SEQuence]:EXTernal:POSition?

Back to Attributes

CMTNA_ATTR_STIMULUS_EXT_TRIGGER_DELAY

Description

Sets/Gets the response delay with respect to the external trigger signal. The value is expressed in seconds (sec).

Unit: sec (second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_EXT_TRIGGER_DELAY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_EXT_TRIGGER_DELAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

TRIGger[:SEQuence]:EXTernal:DELay <time>

TRIGger[:SEQuence]:EXTernal:DELay?

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT

Description

Turns ON/OFF the trigger output.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);


ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

TRIGger:OUTPut:STATe {OFF|ON|0|1}

TRIGger:OUTPut:STATe?

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_POLARITY

Description

Sets/Gets the polarity of the trigger output.

Default value: CMTNA_ATTR_TRIGGER_OUTPUT_POLARITY_NEGATIVE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_POLARITY

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_POLARITY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_OUTPUT_POLARITY_NEGATIVE	Negative edge
CMTNA_ATTR_TRIGGER_OUTPUT_POLARITY_POSITIVE	Positive edge

SCPI Command

TRIGger:OUTPut:POLarity <char>

TRIGger:OUTPut:POLarity?

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_FUNCTION

Description

Sets/Gets the trigger output function. The trigger output outputs various waveforms depending on the setting of the Output Trigger Function.

Default value: CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_READY_FOR_TRIGGER

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_FUNCTION

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_FUNCTION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_BSETUP	Before frequency setup pulse
CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_BSAMPLING	Before sampling pulse
CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_ASAMPLING	After sampling pulse
CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_READY_FOR_TRIGGER	Ready for trigger signal
CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_SWEEP_END	End of sweep pulse
CMTNA_ATTR_TRIGGER_OUTPUT_FUNCTION_MEASUREMENT	Measurement sweep signal

SCPI Command

TRIGger:OUTPut:FUNCtion <char>

TRIGger:OUTPut:FUNCtion?

Back to Attributes

CMTNA_ATTR_STIMULUS_TRIGGER_SCOPE

Description

Sets/Gets the trigger scope. The trigger scope determines the response on the trigger signal arrival: either starts a sweep of all waiting repCapIDs in turn or starts a sweep in the active channel only.

Default value: CMTNA_ATTR_TRIGGER_SCOPE_ALL_CHANNEL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_TRIGGER_SCOPE

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_TRIGGER_SCOPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_SCOPE_ALL_CHANNEL	All channels
CMTNA_ATTR_TRIGGER_SCOPE_ACTIVE_CHANNEL	Active channel

SCPI Command

TRIGger[:SEQuence]:SCOPe <char>

TRIGger[:SEQuence]:SCOPe?

Back to Attributes

CMTNA_ATTR_STIMULUS_FREQUENCY_OFFSET

Description

Turns ON/OFF the frequency offset feature.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_FREQUENCY_OFFSET

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_STIMULUS_FREQUENCY_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet[:STATe] {OFF|ON|0|1}

SENSe<Ch>:OFFSet[:STATe]?

Back to Attributes

CMTNA_ATTR_FREQUENCY_OFFSET_TYPE

Description

Sets/Gets the frequency offset type when the frequency offset feature is ON. There are two frequency offset types: "Port1/Port2" and "Source/Receivers". First offset type offsets ports against each other. Second offset type offsets source against receivers.

Default value: CMTNA_ATTR_FREQUENCY_OFFSET_TYPE_PORT_PORT

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_OFFSET_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_OFFSET_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_FREQUENCY_OFFSET_TYPE_PORT_PORT	Port1/Port2 offset
CMTNA_ATTR_FREQUENCY_OFFSET_TYPE_SOURCE_RECEIVER	Source/Receivers offset

SCPI Command

SENSe<Ch>:OFFSet:TYPE <char>

SENSe<Ch>:OFFSet:TYPE?

Back to Attributes

CMTNA_ATTR_FREQUENCY_MULTIPLIER

Description

Sets/Gets the basic frequency range multiplier of port when the frequency offset feature is ON and offset type is "PORT".

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_MULTIPLIER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_MULTIPLIER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:MULTiplier <numeric>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:MULTiplier?

Back to Attributes

CMTNA_ATTR_FREQUENCY_DIVIDER

Description

Sets/Gets the basic frequency range divisor of current port when the frequency offset feature is ON and offset type is "PORT".

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_DIVIDER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_DIVIDER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:DIVisor <numeric>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:DIVisor?

Back to Attributes

CMTNA_ATTR_FREQUENCY_OFFSET

Description

Sets/Gets the basic frequency range offset of current port when the frequency offset feature is ON and offset type is "PORT". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_OFFSET

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:OFFSet <frequency>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:OFFSet?

Back to Attributes

CMTNA_ATTR_FREQUENCY_OFFSET_START

Description

Sets/Gets the frequency sweep start of current port when the frequency offset feature is ON and offset type is "PORT". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_OFFSET_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_OFFSET_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STARt <frequency>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STARt?

Back to Attributes

CMTNA_ATTR_FREQUENCY_OFFSET_STOP

Description

Sets/Gets the frequency sweep stop of current port when the frequency offset feature is ON and offset type is "PORT". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_OFFSET_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_OFFSET_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STOP <frequency>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STOP?

Back to Attributes

CMTNA_ATTR_FREQUENCY_RECEIVER_MULTIPLIER

Description

Sets/Gets the basic frequency range multiplier to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv".

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_RECEIVER_MULTIPLIER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_RECEIVER_MULTIPLIER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:MULTiplier <numeric>

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:MULTiplier?

Back to Attributes

CMTNA_ATTR_FREQUENCY_RECEIVER_DIVIDER

Description

Sets/Gets the basic frequency range divisor to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv".

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_RECEIVER_DIVIDER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_RECEIVER_DIVIDER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:DIVisor <numeric>

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:DIVisor?

Back to Attributes

CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET

Description

Sets/Gets the basic frequency range offset to get the receiver frequency when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:OFFSet <frequency>

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:OFFSet?

Back to Attributes

CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_START

Description

Sets/Gets the frequency sweep start of the receivers when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:STARt <frequency>

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:STARt?

Back to Attributes

CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_STOP

Description

Sets/Gets the frequency sweep stop of the receivers when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:STOP <frequency>

SENSe<Ch>:OFFSet:RECeiver[:FREQuency]:STOP?

Back to Attributes

CMTNA_ATTR_FREQUENCY_SOURCE_MULTIPLIER

Description

Sets/Gets the basic frequency range multiplier to get the source frequency when the frequency offset feature is ON and offset type is "SRCRcv".

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_SOURCE_MULTIPLIER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_SOURCE_MULTIPLIER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:MULTiplier <numeric>

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:MULTiplier?

Back to Attributes

CMTNA_ATTR_FREQUENCY_SOURCE_DIVIDER

Description

Sets/Gets the basic frequency range divisor to get the source frequency when the frequency offset feature is ON and offset type is "SRCRcv".

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_SOURCE_DIVIDER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_SOURCE_DIVIDER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:DIVisor <numeric>

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:DIVisor?

Back to Attributes

CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET

Description

Sets/Gets the basic frequency range offset to get the source frequency when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_RECEIVER_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:OFFSet <frequency>

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:OFFSet?

Back to Attributes

CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_START

Description

Sets/Gets the frequency sweep start of the source when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:STARt <frequency>

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:STARt?

Back to Attributes

CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_STOP

Description

Sets/Gets the frequency sweep stop of the source when the frequency offset feature is ON and offset type is "SRCRcv". The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FREQUENCY_SOURCE_OFFSET_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:STOP <frequency>

SENSe<Ch>:OFFSet:SOURce[:FREQuency]:STOP?

Back to Attributes

CMTNA_ATTR_STIMULUS_MAX_POINTS

Description

Gets the maximum number of the measurement points.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_STIMULUS_MAX_POINTS

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_MAX_POINTS.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SERVice:SWEep:POINts?

Back to Attributes

CMTNA_ATTR_STIMULUS_MAX_FREQUENCY

Description

Gets the upper limit of the measurement frequency. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_MAX_FREQUENCY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_MAX_FREQUENCY.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SERVice:SWEep:FREQuency:MAXimum?

Back to Attributes

CMTNA_ATTR_STIMULUS_MIN_FREQUENCY

Description

Gets the lower frequency of the measurement frequency. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_STIMULUS_MIN_FREQUENCY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_STIMULUS_MIN_FREQUENCY.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SERVice:SWEep:FREQuency:MINimum?

Back to Attributes

CMTNA_ATTR_EXTERNAL_TRIGGER_ROUTE

Description

Sets/Gets the connector to use for the external trigger input in a PXI system (command valid for PXIe-S5090 model only). The trigger source must be set to the EXTernal. One of the 10 routes can be selected.

The same line cannot be selected as input and output trigger route.

Default value: CMTNA_ATTR_TRIGGER_ROUTE_SMB

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_EXTERNAL_TRIGGER_ROUTE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_EXTERNAL_TRIGGER_ROUTE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_ROUTE_SMB	Front panel connector "Ext Trig In"
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG0	Backplane Trigger Line (PXI TRIG0)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG1	Backplane Trigger Line (PXI TRIG1)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG2	Backplane Trigger Line (PXI TRIG2)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG3	Backplane Trigger Line (PXI TRIG3)

Name	Description
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG4	Backplane Trigger Line (PXI TRIG4)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG5	Backplane Trigger Line (PXI TRIG5)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG6	Backplane Trigger Line (PXI TRIG6)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG7	Backplane Trigger Line (PXI TRIG7)
CMTNA_ATTR_TRIGGER_ROUTE_STAR	Backplane Trigger Line (PXI STAR)

SCPI Command

TRIGger[:SEQuence]:EXTernal:ROUTe <char>

TRIGger[:SEQuence]:EXTernal:ROUte?

Back to Attributes

CMTNA_ATTR_OUTPUT_TRIGGER_ROUTE

Description

Sets/Gets the connector to use for the trigger output in a PXI system (command valid for PXIe-S5090 model only).

The trigger output must be activated and configured by

CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_FUNCTION,

CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT_POLARITY,

CMTNA_ATTR_STIMULUS_TRIGGER_OUTPUT.

One of the 9 routes can be selected.

The same line cannot be selected as input and output trigger route.

Default value: CMTNA_ATTR_TRIGGER_ROUTE_SMB

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_OUTPUT_TRIGGER_ROUTE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
```

```
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
    );
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_OUTPUT_TRIGGER_ROUTE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRIGGER_ROUTE_SMB	Front panel connector "Ext Trig Out"
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG0	Backplane Trigger Line (PXI TRIG0)

Name	Description
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG1	Backplane Trigger Line (PXI TRIG1)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG2	Backplane Trigger Line (PXI TRIG2)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG3	Backplane Trigger Line (PXI TRIG3)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG4	Backplane Trigger Line (PXI TRIG4)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG5	Backplane Trigger Line (PXI TRIG5)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG6	Backplane Trigger Line (PXI TRIG6)
CMTNA_ATTR_TRIGGER_ROUTE_PXI_TRIG7	Backplane Trigger Line (PXI TRIG7)

SCPI Command

TRIGger:OUTPut:ROUTe <char>

TRIGger:OUTPut:ROUte?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_FORMAT

Description

Sets/Gets the display format specified by Measurement Format Enum for the measurement.

Default value: CMTNA_MEASUREMENT_FORMAT_LOG_MAG

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_FORMAT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_FORMAT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MEASUREMENT_FORMAT_LOG_MAG	Logarithmic magnitude
CMTNA_MEASUREMENT_FORMAT_SWR	Voltage standing wave ratio
CMTNA_MEASUREMENT_FORMAT_PHASE	Phase
CMTNA_MEASUREMENT_FORMAT_EXPAND_PHASE	Expanded phase
CMTNA_MEASUREMENT_FORMAT_GROUP_DELAY	Group delay time
CMTNA_MEASUREMENT_FORMAT_LIN_MAG	Linear magnitude

Name	Description
CMTNA_MEASUREMENT_FORMAT_REAL	Real part
CMTNA_MEASUREMENT_FORMAT_IMAG	Imaginary part
CMTNA_MEASUREMENT_FORMAT_SMITH_LOG	Smith chart format (Log)
CMTNA_MEASUREMENT_FORMAT_SMITH_LIN	Smith chart format (Lin)
CMTNA_MEASUREMENT_FORMAT_SMITH_REAL_IMAGE	Smith chart format (Real/Imag)
CMTNA_MEASUREMENT_FORMAT_SMITH_COMPLEX	Smith chart format (R + jX)
CMTNA_MEASUREMENT_FORMAT_SMITH_ADMITTANCE	Smith chart format (G + jB)
CMTNA_MEASUREMENT_FORMAT_POLAR_LOG	Polar format (Log)
CMTNA_MEASUREMENT_FORMAT_POLAR_LIN	Polar format (Lin)
CMTNA_MEASUREMENT_FORMAT_POLAR_REAL_IMAGE	Polar format (Real/Imag)

SCPI Command

CALCulate<Ch>[:SELected]:FORMat <char>

CALCulate<Ch>[:SELected]:FORMat?

CALCulate<Ch>:TRACe<Tr>:FORMat <char>

CALCulate<Ch>:TRACe<Tr>:FORMat?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_SCALE_DIV

Description

Sets/Gets the trace scale. Sets the scale per division, when the data format is the rectangular format. Sets the full scale value, when the data format is the Smith chart format or the polar format. The value is depending on the format.

Unit: dB| ° | sec (decibel | degree | second)

Default value: Varies depending on the format.

Logarithmic Magnitude: 10 dB/Div

Phase: 40 °/Div

Expand Phase: 100 °/Div

Group Delay: 10e–9 sec/Div

Smith Chart, Polar, SWR: 1 /Div

Linear Magnitude: 0.1 /Div

Real part, Imaginary part: 0.2 /Div

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MEASUREMENT_SCALE_DIV

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

```
ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_SCALE_DIV.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:PDIVision <numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:PDIVision?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_REF_VALUE

Description

Sets/Gets the value of the reference line (response value on the reference line). For the rectangular format only. The value is depending on the format.

Unit: dB| ° | sec (decibel | degree | second)

Default value: 0 (except for SWR: 1)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_REF_VALUE

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_REF_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RLEVel <numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RLEVel?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_REF_POSITION

Description

Sets/Gets the position of the reference line. For the rectangular format only.

Default value: 5 (except for SWR: 0)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_REF_POSITION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_REF_POSITION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RPOSition <numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RPOSition?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_DIVISIONS

Description

Sets/Gets the number of the vertical scale divisions. For the rectangular format only.

Default value: 10

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_DIVISIONS

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_DIVISIONS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:Y[:SCALe]:DIVisions <numeric>

DISPlay:WINDow<Ch>:Y[:SCALe]:DIVisions?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_ELEC_DELAY

Description

Sets/Gets the value of the electrical delay. The value is expressed in seconds (sec).

Unit: sec (second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_ELEC_DELAY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_ELEC_DELAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:CORRection:EDELay:TIME <time>

CALCulate<Ch>[:SELected]:CORRection:EDELay:TIME?

CALCulate<Ch>:TRACe<Tr>:CORRection:EDELay:TIME <time>

CALCulate<Ch>:TRACe<Tr>:CORRection:EDELay:TIME?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_PHASE_OFFSET

Description

Sets/Gets the value of the phase offset.The value is expressed in seconds degrees (°).

Unit: ° (degree)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_PHASE_OFFSET

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_PHASE_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:CORRection:OFFSet:PHASe <phase>

CALCulate<Ch>[:SELected]:CORRection:OFFSet:PHASe?

CALCulate<Ch>:TRACe<Tr>:CORRection:OFFSet:PHASe <phase>

CALCulate<Ch>:TRACe<Tr>:CORRection:OFFSet:PHASe?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_AVERAGING

Description

Turns ON/OFF the measurement averaging function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_AVERAGING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_AVERAGING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:AVERage[:STATe] {OFF|ON|0|1}

SENSe<Ch>:AVERage[:STATe]?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_AVERAGING_FACTOR

Description

Sets/Gets the averaging factor, when the averaging function is turned on.

Default value: 10

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_AVERAGING_FACTOR

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_AVERAGING_FACTOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:AVERage:COUNt <numeric>

SENSe<Ch>:AVERage:COUNt?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_AVG_TRIGGER

Description

Turns ON/OFF the averaging trigger function. The function executes a sweep the number of times specified by the averaging factor with a single trigger for the channels with the averaging enabled.

The averaging process begins again with each trigger.

Note: The point trigger function has priority against this command. When the point trigger is enabled the number of pulses equal to (number of points) x (averaging factor) is needed to complete the averaging.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_AVG_TRIGGER

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_AVG_TRIGGER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

TRIGger[:SEQuence]:AVERage {OFF|ON|0|1}

TRIGger[:SEQuence]:AVERage?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_SMOOTHING

Description

Turns ON/OFF the trace smoothing.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_SMOOTHING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_SMOOTHING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:SMOothing[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:SMOothing[:STATe]?

CALCulate<Ch>:TRACe<Tr>:SMOothing[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:SMOothing[:STATe]?

Back to Attributes

CMTNA_ATTR_MEASUREMENT_SMOOTHING_APERTURE

Description

Sets/Gets the smoothing aperture, when performing smoothing function. The value is expressed in percentages (%).

Unit: % (percent)

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MEASUREMENT_SMOOTHING_APERTURE

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MEASUREMENT_SMOOTHING_APERTURE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:SMOothing:APERture <numeric>

CALCulate<Ch>[:SELected]:SMOothing:APERture?

CALCulate<Ch>:TRACe<Tr>:SMOothing:APERture <numeric>

CALCulate<Ch>:TRACe<Tr>:SMOothing:APERture?

Back to Attributes

CMTNA_ATTR_FUNCTION_POINTS

Description

Gets the number of points (data pairs) of the analysis result by CmtNA_FunctionExecute function.

Always reads out 1, when the search is executed for the maximum, minimum, mean, standard deviation, peak, and peak–to–peak values. The actual number of points is read out, when the search is executed for all peak or all targets.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_FUNCTION_POINTS

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index

and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_POINTS.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:POINts?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:POINts?

Back to Attributes

CMTNA_ATTR_FUNCTION_DOMAIN_COUPLING

Description

If the arbitrary range is turned ON by CMTNA_ATTR_FUNCTION_DOMAIN_STATE, it specifies whether all traces of repCapID use the same range (coupling) or each trace uses individual range when CmtNA_FunctionExecute function is executed.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_FUNCTION_DOMAIN_COUPLING

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_DOMAIN_COUPLING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:COUPle {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:COUPle?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain:COUPle {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain:COUPle?

Back to Attributes

CMTNA_ATTR_FUNCTION_DOMAIN_STATE

Description

Specifies whether an arbitrary range or the entire sweep range is used when CmtNA_FunctionExecute function is executed.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_DOMAIN_STATE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_DOMAIN_STATE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:DOMain[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:FUNCtion:DOMain[:STATe]?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain[:STATe]?

Back to Attributes

CMTNA_ATTR_FUNCTION_DOMAIN_START

Description

Sets/Gets the start value of the analysis range of CmtNA_FunctionExecute function.

Unit: Hz | s |dBm (Herz | second | decibels above 1 milliwatt)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_DOMAIN_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_DOMAIN_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STARt <stimulus>

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STARt?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain:STARt <stimulus>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain:STARt?

Back to Attributes

CMTNA_ATTR_FUNCTION_DOMAIN_STOP

Description

Sets/Gets the stop value of the analysis range of CmtNA_FunctionExecute function.

Unit: Hz | sec |dBm (Herz | second | decibels above 1 milliwatt)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_DOMAIN_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_DOMAIN_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STOP <stimulus>

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STOP?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain:STOP <stimulus>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:DOMain:STOP?

Back to Attributes

CMTNA_ATTR_FUNCTION_PEAK_EXCURSION

Description

Sets/Gets the lower limit for the peak excursion value when executing the peak search with CmtNA_FunctionExecute function.

Unit: dB| ° | sec (decibel | degree | second)

Default value: 3.0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_FUNCTION_PEAK_EXCURSION

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_PEAK_EXCURSION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:PEXCursion <numeric>

CALCulate<Ch>[:SELected]:FUNCtion:PEXCursion?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:PEXCursion <numeric>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:PEXCursion?

Back to Attributes

CMTNA_ATTR_FUNCTION_PEAK_POLARITY

Description

Sets/Gets the polarity when performing the peak search with CmtNA_FunctionExecute function.

Default value: CMTNA_ATTR_FUNCTION_PEAK_POLARITY_POSITIVE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_PEAK_POLARITY

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_PEAK_POLARITY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_FUNCTION_PEAK_POLARITY_POSITIVE	Positive peaks
CMTNA_ATTR_FUNCTION_PEAK_POLARITY_NEGATIVE	Negative peaks
CMTNA_ATTR_FUNCTION_PEAK_POLARITY_BOTH	Both positive peaks and negative peaks

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:PPOLarity <char>

CALCulate<Ch>[:SELected]:FUNCtion:PPOLarity?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:PPOLarity <char>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:PPOLarity?

Back to Attributes

CMTNA_ATTR_FUNCTION_TARGET_LEVEL

Description

Sets/Gets the target level when performing the search for the trace and the target level crosspoints with CmtNA_FunctionExecute function.

Unit: dB| ° | sec (decibel | degree | second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_TARGET_LEVEL

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_TARGET_LEVEL.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:TARGet <numeric>

CALCulate<Ch>[:SELected]:FUNCtion:TARGet?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:TARGet <numeric>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:TARGet?

Back to Attributes

CMTNA_ATTR_FUNCTION_TRANSITION_TYPE

Description

Sets/Gets the transition type when performing the search for the trace and the target level crosspoints with CmtNA_FunctionExecute function.

Default value: CMTNA_ATTR_FUNCTION_TRANSITION_TYPE_POSITIVE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_TRANSITION_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_FUNCTION_TRANSITION_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_FUNCTION_TRANSITION_TYPE_POSITIVE	Positive peaks
CMTNA_ATTR_FUNCTION_TRANSITION_TYPE_NEGATIVE	Negative peaks
CMTNA_ATTR_FUNCTION_TRANSITION_TYPE_BOTH	Both positive peaks and negative peaks

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:TTRansition <char>

CALCulate<Ch>[:SELected]:FUNCtion:TTRansition?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:TTRansition <char>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:TTRansition?

Back to Attributes

CMTNA_ATTR_FUNCTION_TYPE

Description

Sets/Gets the type of analysis executed by CmtNA_FunctionExecute function.

Default value: CMTNA_ATTR_FUNCTION_PEAK_TO_PEAK

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_FUNCTION_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

repCapID

Must be CMTNA_ATTR_FUNCTION_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_FUNCTION_PEAK_TO_PEAK	Peak–to–peak (difference between the maximum value and the minimum value)
CMTNA_ATTR_FUNCTION_STANDARD_DEVIATION	Standard deviation
CMTNA_ATTR_FUNCTION_MEAN	Mean value
CMTNA_ATTR_FUNCTION_MAXIMUM	Maximum value
CMTNA_ATTR_FUNCTION_MINIMUM	Minimum value
CMTNA_ATTR_FUNCTION_PEAK	Search for peak

Name	Description
CMTNA_ATTR_FUNCTION_ALL_PEAKS	Search for all the peaks
CMTNA_ATTR_FUNCTION_ALL_TARGET	Search for all targets

SCPI Command

CALCulate<Ch>[:SELected]:FUNCtion:TYPE <char>

CALCulate<Ch>[:SELected]:FUNCtion:TYPE?

CALCulate<Ch>:TRACe<Tr>:FUNCtion:TYPE <char>

CALCulate<Ch>:TRACe<Tr>:FUNCtion:TYPE?

Back to Attributes

CMTNA_ATTR_DISPLAY_ACTIVE_TRACE

Description

Sets/Gets the active trace in channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_ACTIVE_TRACE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace number of the channel i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_ACTIVE_TRACE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:PARameter<Tr>:SELect

SERVice:CHANnel<Ch>:TRACe:ACTive?

Back to Attributes

CMTNA_ATTR_DISPLAY_ACTIVE_CHANNEL

Description

Sets/Gets the active channel.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_ACTIVE_CHANNEL

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability. The physical names are supported along with the corresponding Channel index for the active trace number of the channel i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_ACTIVE_CHANNEL.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:ACTivate

SERVice:CHANnel:ACTive?

Back to Attributes

CMTNA_ATTR_DISPLAY_CHANNEL_ALLOCATION

Description

Sets/Gets the layout of the channel windows on the screen. The channel window layout on the screen see below.

1: 	2: 	3: 	4: 
5: 	6: 	7: 	8: 
9: 	10: 	11: 	12: 
13: 	14: 	15: 	16: 

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_CHANNEL_ALLOCATION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
```

```
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
    );
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_CHANNEL_ALLOCATION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:SPLit <numeric>

DISPlay:SPLit?

Back to Attributes

CMTNA_ATTR_DISPLAY_TRACE_ALLOCATION

Description

Sets/Gets the layout of the graph in the channel window. The graph layout in the channel window see below.

1: ×1	2: ×2	3: ×2	4: ×3
5: ×3	6: ×3	7: ×4	8: ×4
9: ×6	10: ×6	11: ×8	12: ×8
13: ×9	14: ×12	15: ×12	16: ×16

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DISPLAY_TRACE_ALLOCATION

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
```

```
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
    );
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TRACE_ALLOCATION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:SPLit <numeric>

DISPlay:WINDow<Ch>:SPLit?

Back to Attributes

CMTNA_ATTR_DISPLAY_TRACE_COUNT

Description

Sets/Gets the number of traces in the channel.

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_TRACE_COUNT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TRACE_COUNT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:PARameter:COUNt <numeric>

CALCulate<Ch>:PARameter:COUNt?

Back to Attributes

CMTNA_ATTR_DISPLAY_MAXIMIZE_CHANNEL

Description

Turns ON/OFF of the maximization of the active channel window.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_MAXIMIZE_CHANNEL

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_MAXIMIZE_CHANNEL.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:MAXimize {OFF|ON|0|1}

DISPlay:MAXimize?

Back to Attributes

CMTNA_ATTR_DISPLAY_MAXIMIZE_TRACE

Description

Turns ON/OFF the active trace maximization inside the specified channel.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_MAXIMIZE_TRACE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_MAXIMIZE_TRACE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:MAXimize {OFF|ON|0|1}

DISPlay:WINDow<Ch>:MAXimize?

Back to Attributes

CMTNA_ATTR_DISPLAY_TRACE_TYPE

Description

Turns ON/OFF the memory trace display.

Default value: CMTNA_ATTR_DISPLAY_TRACE_DATA

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_TRACE_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TRACE_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_DISPLAY_TRACE_DATA	Only data trace is displayed
CMTNA_ATTR_DISPLAY_TRACE_MEMORY	Only memory trace is displayed
CMTNA_ATTR_DISPLAY_TRACE_DATA_AND_MEMORY	Data trace and memory trace are displayed
CMTNA_ATTR_DISPLAY_TRACE_OFF	Both traces are not displayed

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:MEMory[:STATe] {OFF|ON|0|1}

DISPlay:WINDow<Ch>:TRACe<Tr>:MEMory[:STATe]?

DISPlay:WINDow<Ch>:TRACe<Tr>:STATe {OFF|ON|0|1}

DISPlay:WINDow<Ch>:TRACe<Tr>:STATe?

Back to Attributes

CMTNA_ATTR_DISPLAY_TRACE_DATA_MATH

Description

Sets/Gets the math operation between the data trace and the memory trace. The math result replaces the data trace. If the memory trace does not exist, the command is ignored.

Default value: CMTNA_ATTR_TRACE_DATA_MATH_OFF

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DISPLAY_TRACE_DATA_MATH

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TRACE_DATA_MATH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRACE_DATA_MATH_DIV	Division Data / Mem.
CMTNA_ATTR_TRACE_DATA_MATH_MULT	Multiplication Data x Mem.
CMTNA_ATTR_TRACE_DATA_MATH_SUBT	Subtraction Data – Mem.
CMTNA_ATTR_TRACE_DATA_MATH_ADD	Addition Data + Mem.
CMTNA_ATTR_TRACE_DATA_MATH_OFF	No math

SCPI Command

CALCulate<Ch>[:SELected]:MATH:FUNCtion <char>

CALCulate<Ch>[:SELected]:MATH:FUNCtion?

CALCulate<Ch>:TRACe<Tr>:MATH:FUNCtion <char>

CALCulate<Ch>:TRACe<Tr>:MATH:FUNCtion?

Back to Attributes

CMTNA_ATTR_DISPLAY_TRACE_HOLD_TYPE

Description

Sets/Gets the type of the trace hold function. The function holds the trace at the maximum or minimum point.

Default value: CMTNA_ATTR_TRACE_HOLD_OFF

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_TRACE_HOLD_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TRACE_HOLD_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ATTR_TRACE_HOLD_OFF	Turns off the trace hold function.
CMTNA_ATTR_TRACE_HOLD_MAX	Maximum hold.
CMTNA_ATTR_TRACE_HOLD_MIN	Minimum hold.

SCPI Command

CALCulate<Ch>[:SELected]:HOLD:TYPE <char>

CALCulate<Ch>[:SELected]:HOLD:TYPE?

CALCulate<Ch>:TRACe<Tr>:HOLD:TYPE <char>

CALCulate<Ch>:TRACe<Tr>:HOLD:TYPE?

Back to Attributes

CMTNA_ATTR_DISPLAY_DATA_TRACE_COLOR

Description

Sets/Gets the data trace color. The color is defined by its mix of red (RR), green (GG) and blue (BB) components each represented by 16-bit hexadecimal number.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_DATA_TRACE_COLOR

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Measurement index i.e. "Measurement1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_DATA_TRACE_COLOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:COLor:TRACe<Tr>:DATA <numeric 1>,<numeric 2>,<numeric 3>

DISPlay:COLor:TRACe<Tr>:DATA?

Back to Attributes

CMTNA_ATTR_DISPLAY_MEMORY_TRACE_COLOR

Description

Sets/Gets the memory trace color. The color is defined by its mix of red (RR), green (GG) and blue (BB) components each represented by 16-bit hexadecimal number.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_MEMORY_TRACE_COLOR

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Measurement index i.e. "Measurement1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_MEMORY_TRACE_COLOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:COLor:TRACe<Tr>:MEMory <numeric 1>,<numeric 2>,<numeric 3>

DISPlay:COLor:TRACe<Tr>:MEMory?

Back to Attributes

CMTNA_ATTR_DISPLAY_BACKGROUND_COLOR

Description

Sets/Gets the background color for trace display. The color is defined by its mix of red (RR), green (GG) and blue (BB) components each represented by 16-bit hexadecimal number.

Default value: 0, 0, 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DISPLAY_BACKGROUND_COLOR

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_BACKGROUND_COLOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:COLor:BACK <numeric 1>,<numeric 2>,<numeric 3>

DISPlay:COLor:BACK?

Back to Attributes

CMTNA_ATTR_DISPLAY_GRID_COLOR

Description

Sets/Gets the grid and the graticule label color for trace display. The color is defined by its mix of red (RR), green (GG) and blue (BB) components each represented by 16-bit hexadecimal number.

Default value: 160, 160, 164

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_GRID_COLOR

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_GRID_COLOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:COLor:GRATicule <numeric 1>,<numeric 2>,<numeric 3>

DISPlay:COLor:GRATicule?

Back to Attributes

CMTNA_ATTR_DISPLAY_INVERT_COLOR

Description

Turns ON/OFF the inversion of display colors of the traces area.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_INVERT_COLOR

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

repCapID

Must be CMTNA_ATTR_DISPLAY_INVERT_COLOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:IMAGe <char>

DISPlay:IMAGe?

Back to Attributes

CMTNA_ATTR_DISPLAY_SYSTEM_DATE

Description

Sets/Gets the current date.

The date format: "YYYY,MM,DD" (YYYY - year, MM - month, DD - day).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_SYSTEM_DATE

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_SYSTEM_DATE.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:DATE <numeric 1>,<numeric 2>,<numeric 3>

SYSTem:DATE?

Back to Attributes

CMTNA_ATTR_DISPLAY_SYSTEM_TIME

Description

Sets/Gets the current time.

The time format: "hh,mm,ss" (hh - hour, mm - minute, ss - second).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_SYSTEM_TIME

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_SYSTEM_TIME.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:TIME <numeric 1>,<numeric 2>,<numeric 3>

SYSTem:TIME?

Back to Attributes

CMTNA_ATTR_DISPLAY_CYCLE_TIME_VALUE

Description

Gets the measured cycle time. The cycle time is the interval between the start of two adjacent sweeps. The cycle time is averaged by an exponential window with a time constant of about 0.5 sec. If the cycle time is changed more than 100 usec in comparison with the averaged time, the averaging starts anew. The value is expressed in seconds (sec).

Unit: sec (second)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DISPLAY_CYCLE_TIME_VALUE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_CYCLE_TIME_VALUE.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SYSTem:CYCLe:TIME:MEASurement?

Back to Attributes

CMTNA_ATTR_DISPLAY_TITLE_LABEL

Description

Turns ON/OFF the channel title display.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_TITLE_LABEL

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TITLE_LABEL.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TITLe[:STATe] {OFF|ON|0|1}

DISPlay:WINDow<Ch>:TITLe[:STATe]?

Back to Attributes

CMTNA_ATTR_DISPLAY_TITLE_DATA

Description

Sets/Gets the channel title label.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_TITLE_DATA

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_DISPLAY_TITLE_DATA.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TITLe:DATA <string>

DISPlay:WINDow<Ch>:TITLe:DATA?

Back to Attributes

CMTNA_ATTR_DISPLAY_UPDATE

Description

Turns ON/OFF the display update.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DISPLAY_UPDATE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_DISPLAY_UPDATE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:ENABle {OFF|ON|0|1}

DISPlay:ENABle?

Back to Attributes

CMTNA_ATTR_CALIBRATION_CORRECTION

Description

Turns ON/OFF the S-parameter error correction.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALIBRATION_CORRECTION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALIBRATION_CORRECTION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:STATe {OFF|ON|0|1}

SENSe<Ch>:CORRection:STATe?

Back to Attributes

CMTNA_ATTR_CALIBRATION_TYPE

Description

Gets the calibration type for the calculation of the calibration coefficients on completion of the calibration executed by CmtNA_ApplyCalibration function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALIBRATION_TYPE

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALIBRATION_TYPE.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

Defined Values

Name	Description
CMTNA_CALIBRATION_TYPE_RESPONSE_OPEN	OPEN response calibration
CMTNA_CALIBRATION_TYPE_RESPONSE_SHORT	SHORTresponse calibration
CMTNA_CALIBRATION_TYPE_RESPONSE_THRU	THRU response calibration
CMTNA_CALIBRATION_TYPE_1PATH_2PORT	One-path 2-port calibration
CMTNA_CALIBRATION_TYPE_1PORT_SOL	Full 1-port calibration
CMTNA_CALIBRATION_TYPE_2PORT_SOLT	Full 2-port calibration
CMTNA_CALIBRATION_TYPE_NOT_DEFINED	Not defined

SCPI Command

SENSe<Ch>:CORRection:COLLect:METHod:TYPE?

Back to Attributes

CMTNA_ATTR_CORRECTION_TYPE

Description

Gets the applied calibration type and port numbers for the specified trace.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CORRECTION_TYPE

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index end Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CORRECTION_TYPE.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SENSe<Ch>:CORRection:TYPE<Tr>?

Back to Attributes

CMTNA_ATTR_CORRECTION_STATUS

Description

Gets the interpolation/extrapolation status of the error correction.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CORRECTION_STATUS

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CORRECTION_STATUS.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:CORRection:STATus?

CALCulate<Ch>:TRACe<Tr>:CORRection:STATus?

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_PORT

Description

Sets the port number and sets the adapter removal/insertion function for the calculation of the calibration coefficients when CmtNA_ApplyCalibration function has been executed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_ADAPTER_REMOVAL_PORT

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_PORT.

value

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:METHod:ADAPter:REMoval <port>

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_DELAY

Description

Sets/Gets the approximate delay value of an adapter in the adapter removal/insertion function. This value is used to eliminate the uncertainty of ± 180 ° when calculating the phase response of the adapter.

The sign of the value depends on the type of the removal / insertion function. The value must be negative for the adapter removal function and must be positive for the adapter insertion function.

If this value is set to zero, the analyzer uses an algorithm to automatically determine the delay of the adapter. In most cases setting this value to zero is enough. Setting this value to non zero is required when:

$$Frequency\ Step > \frac{1}{2Delay}$$

The delay and the length of the adapter can be set mutually:

$$Delay = \frac{Length\sqrt{Permittivity}}{C}$$

The value is expressed in seconds (sec).

Unit: sec (second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_ADAPTER_REMOVAL_DELAY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
```

```
        ViReal64 *value
    );

    ViStatus    CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
    );
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_DELAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ADAPter:DELay <numeric>

SENSe<Ch>:CORRection:COLLect:ADAPter:DELay?

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_LENGTH

Description

Sets/Gets the approximate value of the mechanical length of the adapter in the adapter removal/insertion function. This value is used to eliminate the uncertainty of ± 180 ° when calculating the phase response of the adapter.

The sign of the value depends on the type of the removal / insertion function. The value must be negative for the adapter removal function and must be positive for the adapter insertion function.

If this value is set to zero, the analyzer uses an algorithm to automatically determine the delay of the adapter. In most cases setting this value to zero is enough. Setting this value to non zero is required when:

$$Frequency\ Step > \frac{1}{2Delay}$$

The delay and the length of the adapter can be set mutually:

$$Delay = \frac{Length\sqrt{Permittivity}}{C}$$

The value is expressed in meters (m).

Unit: m (meters)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_ADAPTER_REMOVAL_LENGTH

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
```

```
        ViReal64 *value
    );

    ViStatus    CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
    );
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_LENGTH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ADAPter:LENGth <numeric>

SENSe<Ch>:CORRection:COLLect:ADAPter:LENGth?

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_MEDIA

Description

Sets/Gets the adapter media in the adapter removal/insertion function.

Default value: CMTNA_ADAPTER_REMOVAL_MEDIA_COAXIAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ADAPTER_REMOVAL_MEDIA

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus    CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_MEDIA.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ADAPTER_REMOVAL_MEDIA_COAXIAL	Specifies the coaxial adapter
CMTNA_ADAPTER_REMOVAL_MEDIA_WAVEGUIDE	Specifies the waveguide adapter

SCPI Command

SENSe<Ch>:CORRection:COLLect:ADAPter:MEDia {COAXial|WAVeguide}

SENSe<Ch>:CORRection:COLLect:ADAPter:MEDia?

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_UNIT

Description

Sets/Gets the display units of the adapter delay (length) in the adapter removal/insertion function.

Default value: CMTNA_ADAPTER_REMOVAL_UNIT_SECONDS

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ADAPTER_REMOVAL_UNIT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
       ViInt32  *value
);

ViStatus   CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_UNIT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ADAPTER_REMOVAL_UNIT_SECONDS	Selects seconds
CMTNA_ADAPTER_REMOVAL_UNIT_METERS	Selects meters

SCPI Command

SENSe<Ch>:CORRection:COLLect:ADAPter:UNIT {SEConds|METers}

SENSe<Ch>:CORRection:COLLect:ADAPter:UNIT?

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_PERMITTIVITY

Description

Sets/Gets the value of the permittivity of an adapter media in the adapter removal/insertion function.

When setting the adapter length, this parameter is used to calculate the adapter delay; therefore this parameter must be set before setting of the adapter length. When setting the adapter delay, this parameter is not used.

Default value: 1.000649 (air)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ADAPTER_REMOVAL_PERMITTIVITY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus    CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_PERMITTIVITY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ADAPter:PERMittivity <numeric>

SENSe<Ch>:CORRection:COLLect:ADAPter:PERMittivity?

Back to Attributes

CMTNA_ATTR_ADAPTER_REMOVAL_CUTOFF_FREQ

Description

Sets/Gets the value of the cutoff frequency of the waveguide adapter.

Default value: 1.0 GHz

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ADAPTER_REMOVAL_CUTOFF_FREQ

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus    CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ADAPTER_REMOVAL_CUTOFF_FREQ.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:ADAPter:WAVeguide:CUToff <numeric>

SENSe<Ch>:CORRection:COLLect:ADAPter:WAVeguide:CUToff?

Back to Attributes

CMTNA_ATTR_THRU_ADDITION_DELAY

Description

Sets/Gets the approximate delay value of an unknown thru in the thru addition function. This value is used to eliminate the uncertainty of ± 180 ° when calculating the phase response of the thru.

If this value is set to zero, the analyzer uses an algorithm to automatically determine the delay of the thru. In most cases setting this value to zero is enough. Setting this value to non zero is required when:

$$Frequency\ Step > \frac{1}{2Delay}$$

The delay and the length of the adapter can be set mutually:

$$Delay = \frac{Length\sqrt{Permittivity}}{C}$$

The value is expressed in seconds (sec).

Unit: sec (second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_THRU_ADDITION_DELAY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

```
ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_THRU_ADDITION_DELAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:DELay <numeric>

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:DELay?

Back to Attributes

CMTNA_ATTR_THRU_ADDITION_LENGTH

Description

Sets/Gets the approximate value of the mechanical length of an unknown thru in the thru addition function. This value is used to eliminate the uncertainty of ± 180 ° when calculating the phase response of the thru.

If this value is set to zero, the analyzer uses an algorithm to automatically determine the delay of the thru. In most cases setting this value to zero is enough. Setting this value to non zero is required when:

$$Frequency\ Step > \frac{1}{2Delay}$$

The delay and the length of the adapter can be set mutually:

$$Delay = \frac{Length\sqrt{Permittivity}}{C}$$

The value is expressed in meters (m).

Unit: m (meters)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_THRU_ADDITION_LENGTH

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

```
ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_THRU_ADDITION_LENGTH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:LENGth <numeric>

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:LENGth?

Back to Attributes

CMTNA_ATTR_THRU_ADDITION_UNIT

Description

Sets/Gets the display units of the thru delay (length) in the thru addition function.

Default value: CMTNA_THRU_ADDITION_UNIT_SECONDS

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_THRU_ADDITION_UNIT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_THRU_ADDITION_UNIT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_THRU_ADDITION_UNIT_SECONDS	Selects seconds
CMTNA_THRU_ADDITION_UNIT_METERS	Selects meters

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:UNIT {SEConds|METers}

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:UNIT?

Back to Attributes

CMTNA_ATTR_THRU_ADDITION_MEDIA

Description

Sets/Gets the media of the thru in the thru addition function.

Default value: CMTNA_THRU_ADDITION_MEDIA_COAXIAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_THRU_ADDITION_MEDIA

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_THRU_ADDITION_MEDIA.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_THRU_ADDITION_MEDIA_COAXIAL	Specifies the coaxial
CMTNA_THRU_ADDITION_MEDIA_WAVEGUIDE	Specifies the waveguide

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:MEDia {COAXial|WAVeguide}

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:MEDia?

Back to Attributes

CMTNA_ATTR_THRU_ADDITION_PERMITTIVITY

Description

Sets/Gets the value of the permittivity of the thru media in the thru addition function.

This parameter is used to calculate the adapter delay when the thru length is being set; therefore this parameter must be set before setting of the thru length.

Default value: 1.000649 (air)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_THRU_ADDITION_PERMITTIVITY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_THRU_ADDITION_PERMITTIVITY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:PERMittivity <numeric>

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:PERMittivity?

Back to Attributes

CMTNA_ATTR_THRU_ADDITION_CUTOFF_FREQ

Description

Sets/Gets the value of the cutoff frequency of the waveguide thru in the thru addition function.

Default value: 1.0 GHz

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_THRU_ADDITION_CUTOFF_FREQ

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_THRU_ADDITION_CUTOFF_FREQ.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:WAVeguide:CUToff <numeric>

SENSe<Ch>:CORRection:COLLect:THRU:ADDition:WAVeguide:CUToff?

Back to Attributes

CMTNA_ATTR_CALKIT_SELECTED

Description

Sets/Gets the number of the selected calibration kit in the table of calibration kits. The selected calibration kit is used in the subsequent calibration and is used for editing by CMTNA_ATTR_CALKIT_XXX.

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_SELECTED

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_CALKIT_SELECTED.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT[:SELect] <numeric>

SENSe:CORRection:COLLect:CKIT[:SELect]?

Back to Attributes

CMTNA_ATTR_CALKIT_LABEL

Description

Sets/Gets the calibration kit label.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_LABEL

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_CALKIT_LABEL.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:LABel <string>

SENSe:CORRection:COLLect:CKIT:LABel?

Back to Attributes

CMTNA_ATTR_CALKIT_DESCRIPTION

Description

Sets/Gets the calibration kit description string.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_DESCRIPTION

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_CALKIT_DESCRIPTION.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute

SCPI Command

SENSe:CORRection:COLLect:CKIT:DESCription <string>

SENSe:CORRection:COLLect:CKIT:DESCription ?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_TYPE

Description

Sets/Gets the type of calibration standard.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_CALKIT_STANDARD_OPEN	Open
CMTNA_CALKIT_STANDARD_SHORT	Short
CMTNA_CALKIT_STANDARD_LOAD	Load
CMTNA_CALKIT_STANDARD_THRU_DELAY	Thru
CMTNA_CALKIT_STANDARD_UNKN_THRU	Unknown Thru
CMTNA_CALKIT_STANDARD_SLIDING_LOAD	Sliding Load
CMTNA_CALKIT_STANDARD_DATA_BASED	Data Based
CMTNA_CALKIT_STANDARD_NONE	Not defined

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:TYPE <char>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:TYPE?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_LABEL

Description

Sets/Gets the label for the calibration standard.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_LABEL

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_LABEL.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:LABel <string>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:LABel?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_MIN_FREQUENCY

Description

Sets/Gets the minimum frequency limit of the calibration standard. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_MIN_FREQUENCY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_MIN_FREQUENCY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:FMINimum <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:FMINimum?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_MAX_FREQUENCY

Description

Sets/Gets the maximum frequency limit of the calibration standard. The value is expressed in hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_MAX_FREQUENCY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_MAX_FREQUENCY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:FMAXimum <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:FMAXimum?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_OFFSET_DELAY

Description

Sets/Gets the offset delay value for the calibration standard. The value is expressed in seconds (sec).

Unit: sec (second)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_OFFSET_DELAY

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_OFFSET_DELAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:DELay <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:Delay?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_OFFSET_Z0

Description

Sets/Gets the offset Z0 value for the calibration standard. The value is expressed in Ohm (Ω).

Unit: Ω (Ohm)

Default value: 50 or 75, depending on the selected calibration kit.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_OFFSET_Z0

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_OFFSET_Z0.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:Z0 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:Z0?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_OFFSET_LOSS

Description

Sets/Gets the offset loss value for the calibration standard. The value is expressed in Ohm/seconds (Ω/sec).

Unit: Ω/sec (Ohm/second)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_OFFSET_LOSS

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_OFFSET_LOSS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:LOSS <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:LOSS?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_ARBITRARY

Description

Sets/Gets the value of the arbitrary impedance for the load standard. The value is expressed in Ohm (Ω).

Unit: Ω (Ohm)

Defaut value: 50 or 75, depending on the selected calibration kit

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_ARBITRARY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_ARBITRARY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:ARBitrary <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:ARBitrary?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_C0

Description

Sets/Gets the C0 value for the open calibration standard.The value is expressed in Farad (F).

Unit: 1E–15 F (Farad)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_C0

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_C0.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C0 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C0?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_C1

Description

Sets/Gets the C1 value for the open calibration standard. The value is expressed in Farad/Hertz (F/Hz).

Unit: 1E–27 F/Hz (Farad/Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_C1

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

Must be CMTNA_ATTR_CALKIT_STANDARD_C1.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C1 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C1?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_C2

Description

Sets/Gets the C2 value for the open calibration standard. The value is expressed in Farad/Hertz^2 (F/Hz^2).

Unit: 1E–36 F/Hz^2 (Farad/Hertz^2)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_C2

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_C2.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C2 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C2?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_C3

Description

Sets/Gets the C3 value for the open calibration standard. The value is expressed in Farad/Hertz^3 (F/Hz^3).

Unit: 1E–45 F/Hz^3 (Farad/Hertz^3)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_C3

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_C3.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C3 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:C3?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_L0

Description

Sets/Gets the L0 value for the short calibration standard. The value is expressed in Henry (H).

Unit: 1E–12 H (Henry)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_L0

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_L0.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L0 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L0?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_L1

Description

Sets/Gets the L1 value for the short calibration standard. The value is expressed in Henry/Hertz (H/Hz).

Unit: 1E–24 H/Hz (Henry/Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_L1

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_L1.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L1 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L1?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_L2

Description

Sets/Gets the L2 value for the short calibration standard. The value is expressed in Henry/Hertz^2 (H/Hz^2).

Unit: 1E–33 H/Hz^2 (Henry/Hertz^2)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALKIT_STANDARD_L2

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_L2.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L2 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L2?

Back to Attributes

CMTNA_ATTR_CALKIT_STANDARD_L3

Description

Sets/Gets the L3 value for the short calibration standard. The value is expressed in Henry/Hertz^3 (H/Hz^3).

Unit: 1E–42 H/Hz^3 (Henry/Hertz^3)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALKIT_STANDARD_L3

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Standard index i.e. "Standard1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALKIT_STANDARD_L3.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L3 <numeric>

SENSe:CORRection:COLLect:CKIT:STAN<Std>:L3?

Back to Attributes

CMTNA_ATTR_POWER_CALIBRATION_CORRECTION

Description

Turns ON/OFF the power correction.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_POWER_CALIBRATION_CORRECTION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_POWER_CALIBRATION_CORRECTION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection[:STATe] {OFF|ON|0|1}

SOURce<Ch>:POWer:PORT<Pt>:CORRection[:STATe]?

Back to Attributes

CMTNA_ATTR_POWER_CALIBRATION_LOSS_COMPENSATION

Description

Turns ON/OFF the state of the loss compensation used when the power calibration is executed by CmtNA_TakePowerCalSweep function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_POWER_CALIBRATION_LOSS_COMPENSATION

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_POWER_CALIBRATION_LOSS_COMPENSATION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SOURce<Ch>:POWer:PORT<Pt>:CORRection:COLLect:TABLe:LOSS[:STATe] {OFF|ON|0|1}

SOURce<Ch>:POWer:PORT<Pt>:CORRection:COLLect:TABLe:LOSS[:STATe]?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS

Description

Turns ON/OFF the port extension function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension[:STATe] {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension[:STATe]?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_TIME

Description

Sets/Gets the electrical delay value for the port extension function. The value is expressed in seconds (sec).

Unit: sec (second)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS_TIME

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_TIME.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:TIME <time>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:TIME?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_STATE

Description

Turns ON/OFF the loss compensation of the loss 1 for the port extension function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_STATE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_STATE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:INCLude{[1]|2}[:STATe] {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:INCLude{[1]|2}[:STATe]?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_VALUE

Description

Sets/Gets the values of the loss 1 for the port extension function. The value is expressed in decibels (dB).

Unit: dB (decibel)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_VALUE

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_LOSS1_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LOSS{[1]|2} <numeric>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LOSS{[1]|2}?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_FREQ1_VALUE

Description

Sets/Gets the values of the frequency 1 to calculate the loss for the port extension function. The value is expressed in Hertz (Hz).

Unit: Hz (Hertz)

Default value: 1E9

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_PORT_EXTENSIONS_FREQ1_VALUE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_FREQ1_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:FREQuency{[1]|2} <frequency>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:FREQuency{[1]|2}?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_STATE

Description

Turns ON/OFF the loss compensation of the loss 2 for the port extension function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_STATE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_STATE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:INCLude{[1]|2}[:STATe] {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:INCLude{[1]|2}[:STATe]?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_VALUE

Description

Sets/Gets the values of the loss 2 for the port extension function. The value is expressed in decibels (dB).

Unit: dB (decibel)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_VALUE

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_LOSS2_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LOSS{[1]|2} <numeric>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LOSS{[1]|2}?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_FREQ2_VALUE

Description

Sets/Gets the values of the frequency 2 to calculate the loss for the port extension function. The value is expressed in Hertz (Hz).

Unit: Hz (Hertz)

Default value: 1E9

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_PORT_EXTENSIONS_FREQ2_VALUE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_FREQ2_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:FREQuency{[1]|2} <frequency>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:FREQuency{[1]|2}?

Back to Attributes

CMTNA_ATTR_PORT_EXTENSIONS_LDC_VALUE

Description

Sets/Gets the loss value at DC for the port extension function. The value is expressed in decibels (dB).

Unit: dB (decibel)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PORT_EXTENSIONS_LDC_VALUE

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_PORT_EXTENSIONS_LDC_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LDC <numeric>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LDC?

Back to Attributes

CMTNA_ATTR_AUTO_PORT_EXTENSION_METHOD

Description

Sets/Gets the frequency range used for calculation of the results of the Auto Port Extension function.

Default value: CMTNA_AUTO_PORT_EXTENSION_CURRENT_SPAN

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTO_PORT_EXTENSION_METHOD

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_AUTO_PORT_EXTENSION_METHOD.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_AUTO_PORT_EXTENSION_CURRENT_SPAN	Uses current frequency span
CMTNA_AUTO_PORT_EXTENSION_ACTIVE_MARKER	Uses the frequency of the active marker. This is applied to Loss 1 and Loss 2 is ignored.
CMTNA_AUTO_PORT_EXTENSION_USER_SPAN	Uses arbitrary frequency range set by user

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:CONFig {CSPN|AMKR|USPN}

SENSe<Ch>:CORRection:EXTension:AUTO:CONFig?

Back to Attributes

CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_START

Description

Sets/Gets the start value of the user span of the auto port extension function. The value is expressed in Hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_START

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:STARt <frequency>

SENSe<Ch>:CORRection:EXTension:AUTO:STARt?

Back to Attributes

CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_STOP

Description

Sets/Gets the stop value of the user span of the auto port extension function. The value is expressed in Hertz (Hz).

Unit: Hz (Hertz)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_STOP

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_AUTO_PORT_EXTENSION_USER_SPAN_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:STOP <frequency>

SENSe<Ch>:CORRection:EXTension:AUTO:STOP?

Back to Attributes

CMTNA_ATTR_AUTO_PORT_EXTENSION_INCLUDE_LOSS

Description

Turns ON/OFF the usage of "Loss1"and "Loss2" values for the results of the auto port extension function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTO_PORT_EXTENSION_INCLUDE_LOSS

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_AUTO_PORT_EXTENSION_INCLUDE_LOSS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:LOSS {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension:AUTO:LOSS?

Back to Attributes

CMTNA_ATTR_AUTO_PORT_EXTENSION_ADJUST_MISMATCH

Description

Turns ON/OFF the usage of "Loss at DC" value for the results of the auto port extension function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTO_PORT_EXTENSION_ADJUST_MISMATCH

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_AUTO_PORT_EXTENSION_ADJUST_MISMATCH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:EXTension:AUTO:DCOFfset {OFF|ON|0|1}

SENSe<Ch>:CORRection:EXTension:AUTO:DCOFfset?

Back to Attributes

CMTNA_ATTR_CALIBRATION_TRIGGER_SOURCE

Description

Enables/Disables the internal trigger source for calibration.

If the internal trigger source for calibration is enabled then a command of the calibration standard measurement starts the measurement immediately.

If the internal trigger source for calibration is disabled then the system trigger source is used (which is set for regular measurement with CMTNA_ATTR_STIMULUS_TRIGGER_SOURCE) to start the calibration standard measurement.

The system trigger source also enables the averaging trigger CMTNA_ATTR_MEASUREMENT_AVG_TRIGGER and the point trigger CMTNA_ATTR_STIMULUS_EXT_TRIGGER_EVENT for calibration.

NOTE When the system trigger source is selected you should avoid the program trigger source (BUS), otherwise the program deadlock is possible.

NOTE The command isn't applied to the electronic calibration, the power calibration and the receiver calibration. The internal trigger is always used in these cases.

Default value: CMTNA_CALIBRATION_TRIGGER_SOURCE_INTERNAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALIBRATION_TRIGGER_SOURCE

ViStatus CmtNA_GetAttributeViInt32(
    ViSession vi,
```

```
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
    );

    ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
    );
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALIBRATION_TRIGGER_SOURCE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_CALIBRATION_TRIGGER_SOURCE_INTERNAL	Internal
CMTNA_CALIBRATION_TRIGGER_SOURCE_SYSTEM	System

SCPI Command

SENSe<Ch>:CORRection:TRIGger:FREE[:STATe] {OFF|ON|0|1}

SENSe<Ch>:CORRection:TRIGger:FREE[:STATe]?

Back to Attributes

CMTNA_ATTR_CALIBRATION_PORT_Z0

Description

Gets the system impedance Z0 or the the impedance Z0 of port. The value is expressed in Ohm (Ω).

Unit: Ω (Ohm)

Default value: 50

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CALIBRATION_PORT_Z0

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier.

Must be VI_NULL or an empty string if is used the system impedance Z0.

If id used the the impedance Z0 of port, the physical names are supported along with the corresponding Port index i.e. "Port1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_CALIBRATION_PORT_Z0.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SENSe:CORRection:IMPedance[:INPut][:MAGNitude] <numeric>

SENSe:CORRection:IMPedance[:INPut][:MAGNitude]?

SENSe:CORRection:PORT<Pt>:IMPedance[:INPut][:MAGNitude] <numeric>

SENSe:CORRection:PORT<Pt>:IMPedance[:INPut][:MAGNitude]?

Back to Attributes

CMTNA_ATTR_CALIBRATION_AUTO_SELECT_Z0

Description

Turns ON/OFF the auto-select Z0 function. When enabled the function sets the port impedance Z0 to the corresponding value of measuring calibration standard.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_CALIBRATION_AUTO_SELECT_Z0

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_CALIBRATION_AUTO_SELECT_Z0.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:IMPedance[:INPut]:SELect:AUTO

SENSe:CORRection:IMPedance[:INPut]:SELect:AUTO?

Back to Attributes

CMTNA_ATTR_AUTOCAL_CHARACTERIZATION

Description

Sets/Gets the characterization number used when executing AutoCal (factory or user characterizations).

Default value: CMTNA_AUTOCAL_CHARACTERIZATION_FACTORY

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTOCAL_CHARACTERIZATION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_AUTOCAL_CHARACTERIZATION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_AUTOCAL_CHARACTERIZATION_FACTORY	Factory characterization
CMTNA_AUTOCAL_CHARACTERIZATION_USER1	User characterization 1
CMTNA_AUTOCAL_CHARACTERIZATION_USER2	User characterization 2
CMTNA_AUTOCAL_CHARACTERIZATION_USER3	User characterization 3

SCPI Command

SENSe:CORRection:COLLect:ECAL:UCHar <char>

SENSe:CORRection:COLLect:ECAL:UCHar?

Back to Attributes

CMTNA_ATTR_AUTOCAL_AUTO_ORIENTATION

Description

Turns ON/OFF the Auto-Orientation function when the AutoCal Module calibration is executed.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTOCAL_AUTO_ORIENTATION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_AUTOCAL_AUTO_ORIENTATION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:ECAL:ORIentation:STATe {OFF|ON|0|1}

SENSe:CORRection:COLLect:ECAL:ORIentation:STATe?

Back to Attributes

CMTNA_ATTR_AUTOCAL_ORIENTATION_PORT

Description

Sets/Gets the AutoCal module port number which is connected to a specified port of Network Analyzer.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTOCAL_ORIENTATION_PORT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_AUTOCAL_ORIENTATION_PORT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_AUTOCAL_ORIENTATION_PORT_A	Port A of AutoCal Module
CMTNA_AUTOCAL_ORIENTATION_PORT_B	Port B of AutoCal Module
CMTNA_AUTOCAL_ORIENTATION_PORT_C	Port C of AutoCal Module
CMTNA_AUTOCAL_ORIENTATION_PORT_D	Port D of AutoCal Module

SCPI Command

SENSe:CORRection:COLLect:ECAL:PATH <numeric1>,<numeric2>

SENSe:CORRection:COLLect:ECAL:PATH? <numeric1>

Back to Attributes

CMTNA_ATTR_AUTOCAL_UNKNOWN_THRU

Description

Turns ON/OFF the Unknown Thru feature when the AutoCal Module calibration is executed.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_AUTOCAL_UNKNOWN_THRU

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_AUTOCAL_UNKNOWN_THRU.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe:CORRection:COLLect:ECAL:UTHRu:STATe {OFF|ON|0|1}

SENSe:CORRection:COLLect:ECAL:UTHRu:STATe?

Back to Attributes

CMTNA_ATTR_AUTOCAL_TEMPERATURE

Description

Gets the temperature of the AutoCal module connected to the Analyzer. The value is expressed in Celsius degrees (°C).

Unit: °C (Celsius degree)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_AUTOCAL_TEMPERATURE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_AUTOCAL_TEMPERATURE.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SYSTem:COMMunicate:ECAL:TEMPerature:SENSor?

Back to Attributes

CMTNA_ATTR_REFERENCE_MARKER

Description

Turns ON/OFF the reference marker.

When the reference marker is turned ON, all the values of the other markers turn to relative values.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_REFERENCE_MARKER

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_REFERENCE_MARKER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:REFerence[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:REFerence[:STATe]?

CALCulate<Ch>:TRACe<Tr>:MARKer:REFerence[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:REFerence[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKERS_COUNT

Description

Sets\Gets the number of the turned ON markers.

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKERS_COUNT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKERS_COUNT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:COUNt <numeric>

CALCulate<Ch>[:SELected]:MARKer:COUNt?

CALCulate<Ch>:TRACe<Tr>:MARKer:COUNt <numeric>

CALCulate<Ch>:TRACe<Tr>:MARKer:COUNt?

Back to Attributes

CMTNA_ATTR_SELECTED_MARKER

Description

Sets the active marker.

If the marker is not ON, this function will turn the marker ON. Turning ON a marker with the number from 1 to 15 will turn ON all the markers of smaller numbers. Turning ON the reference marker with number 16 does not turn ON the markers with the numbers from 1 to 15, but switches these markers to the relative measurement mode.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_SELECTED_MARKER

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th

trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_SELECTED_MARKER.

value

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:ACTivate

CALCulate<Ch>:TRACe<Tr>MARKer<Mk>:ACTivate

Back to Attributes

CMTNA_ATTR_MARKER_STIMULUS

Description

Sets/Gets the stimulus value of the marker.

Unit: Hz | sec | dBm (Hertz | second | decibels above 1 milliwatt)

Default value: Stimulus center *value*

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_STIMULUS

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_STIMULUS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:X <stimulus>

CALCulate<Ch>[:SELected]:MARKer<Mk>:X?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:X <stimulus>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:X?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_TYPE

Description

Sets the type of the marker search, which is performed by CmtNA_MarkerFunctionExecute function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MARKER_SEARCH_TYPE

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MARKER_SEARCH_TYPE_MAXIMUM	Maximum
CMTNA_MARKER_SEARCH_TYPE_MINIMUM	Minimum
CMTNA_MARKER_SEARCH_TYPE_PEAK	Search Peak
CMTNA_MARKER_SEARCH_TYPE_PEAK_LEFT	Search Peak Left
CMTNA_MARKER_SEARCH_TYPE_PEAK_RIGHT	Search Peak Right
CMTNA_MARKER_SEARCH_TYPE_TARGET	Search Target

Name	Description
CMTNA_MARKER_SEARCH_TYPE_TARGET_LEFT	Search Target Left
CMTNA_MARKER_SEARCH_TYPE_TARGET_RIGHT	Search Target Right

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TYPE <char>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TYPE?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TYPE <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TYPE?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_PEAK_EXCURSION

Description

Sets/Gets the start value of the marker search range.

Unit: Hz | ° | sec (Hertz | degree | second)

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_PEAK_EXCURSION

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_PEAK_EXCURSION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PEXCursion <numeric>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PEXCursion?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:PEXCursion <numeric>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:PEXCursion?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_PEAK_POLARITY

Description

Sets/Gets the peak polarity, when the marker search for peak is performed by CmtNA_MarkerFunctionExecute function.

Default value: CMTNA_MARKER_SEARCH_PEAK_POLARITY_POSITIVE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_PEAK_POLARITY

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_PEAK_POLARITY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MARKER_SEARCH_PEAK_POLARITY_POSITIVE	Positive polarity
CMTNA_MARKER_SEARCH_PEAK_POLARITY_NEGATIVE	Negative polarity
CMTNA_MARKER_SEARCH_PEAK_POLARITY_BOTH	Both positive polarity and negative polarity

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PPOLarity <char>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PPOLarity?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:PPOLarity <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:PPOLarity?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_TARGET_VALUE

Description

Sets/Gets the target value, when the marker search for target is performed by CmtNA_MarkerFunctionExecute function.

Unit: dB | ° | sec (decibels | second | degree)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MARKER_SEARCH_TARGET_VALUE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_TARGET_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TARGet <numeric>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TARGet?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TARGet <numeric>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TARGet?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_TARGET_TRANSITION

Description

Sets/Gets the type of the target transition, when the marker search for transition is performed by CmtNA_MarkerFunctionExecute function.

Default value: CMTNA_MARKER_SEARCH_TARGET_TRANSITION_BOTH

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_TARGET_TRANSITION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_TARGET_TRANSITION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MARKER_SEARCH_TARGET_TRANSITION_POSITIVE	Positive target transition
CMTNA_MARKER_SEARCH_TARGET_TRANSITION_NEGATIVE	Negative target transition
CMTNA_MARKER_SEARCH_TARGET_TRANSITION_BOTH	Both positive target transition and negative target transition

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TTRansition <char>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TTRansition?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TTRansition <char>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TTRansition?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_TRACKING

Description

Turns ON/OFF the marker search tracking.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_TRACKING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_TRACKING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TRACking {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TRACking?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TRACking {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:FUNCtion:TRACking?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_RANGE

Description

Specifies whether an arbitrary range or the entire sweep range is used when the marker search is performed.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_RANGE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_RANGE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain[:STATe]?

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_START

Description

Sets/Gets the start value of the marker search range.

Unit: Hz | sec | dBm (Hertz | second | decibels above 1 milliwatt)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STARt <stimulus>

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STARt?

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain:STARt <stimulus>

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain:STARt?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_STOP

Description

Sets/Gets the stop value of the marker search range.

Unit: Hz | sec | dBm (Hertz | second | decibels above 1 milliwatt)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STOP <stimulus>

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STOP?

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain:STOP <stimulus>

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain:STOP?

Back to Attributes

CMTNA_ATTR_MARKER_SEARCH_COUPLE

Description

If the arbitrary search range is turned ON by CMTNA_ATTR_MARKER_SEARCH_RANGE, specifies whether all traces of repCapID use the same range (coupling) or each trace uses individual range when the marker search is performed.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_SEARCH_COUPLE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_SEARCH_COUPLE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:COUPle {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:COUPle?

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain:COUPle {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:FUNCtion:DOMain:COUPle?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_STATISTICS

Description

Turns ON/OFF the math statistics display.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_STATISTICS

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_STATISTICS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MSTatistics[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MSTatistics[:STATe]?

CALCulate<Ch>:TRACe<Tr>:MSTatistics[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MSTatistics[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_STATISTIC_RANGE

Description

Sets/Gets either partial frequency range or entire frequency range is used for math statistic calculation. The partial frequency range is limited by two markers.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_STATISTIC_RANGE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_STATISTIC_RANGE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:STATe]?

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DOMain[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DOMain[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_STATISTIC_START

Description

Sets/Gets the number of the marker, which specifies the start frequency of the math statistics range.

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_STATISTIC_START

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_STATISTIC_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STARt <numeric>

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STARt?

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DOMain[:MARKer]:STARt <numeric>

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DOMain[:MARKer]:STARt?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_STATISTIC_STOP

Description

Sets/Gets the number of the marker, which specifies the stop frequency of the math statistics range.

Default value : 2

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_STATISTIC_STOP

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_STATISTIC_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STOP <numeric>

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STOP?

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DOMain[:MARKer]:STOP <numeric>

CALCulate<Ch>:TRACe<Tr>:MSTatistics:DOMain[:MARKer]:STOP?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH

Description

Turns ON/OFF the bandwidth search function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:BWIDth[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:BWIDth[:STATe]?

CALCulate<Ch>:TRACe<Tr>:MARKer:BWIDth[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:BWIDth[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_TYPE

Description

Sets/Gets the type of the bandwidth search function.

Default value: CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_BANDPASS

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_BANDPASS	Bandpass
CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_NOTCH	Notch

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:BWIDth:TYPE <char>

CALCulate<Ch>[:SELected]:MARKer:BWIDth:TYPE?

CALCulate<Ch>:TRACe<Tr>:MARKer:BWIDth:TYPE <char>

CALCulate<Ch>:TRACe<Tr>:MARKer:BWIDth:TYPE?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_REF

Description

Selects the reference point for the bandwidth search function: reference marker or absolute maximum value of the trace.

Default value: CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_REF_MAXIMUM

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_REF

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_REF.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_REF_MARKER	Bandwidth search relative to the reference marker
CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_REF_MAXIMUM	Bandwidth search relative to the absolute maximum of the trace
CMTNA_MARKER_MATH_BANDWIDTH_SEARCH_REF_MINIMUM	Bandwidth search relative to the absolute minimum of the trace

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:BWIDth:REFerence <char>

CALCulate<Ch>[:SELected]:MARKer:BWIDth:REFerence?

CALCulate<Ch>:TRACe<Tr>:MARKer:BWIDth:REFerence <char>

CALCulate<Ch>:TRACe<Tr>:MARKer:BWIDth:REFerence?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_VALUE

Description

Sets/Gets the bandwidth definition value.

Unit: dB | ° | sec (decibel | degree | second)

Default value: -3.0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_VALUE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Marker index for the active trace of channel i.e. "Channel1:Marker2" and so on. The physical names are supported along with the corresponding Channel index and Measurement index and Marker index for the n–th trace of channel i.e. "Channel1:Measurement3:Marker2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_BANDWIDTH_SEARCH_VALUE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer<Mk>:BWIDth:THReshold <numeric>

CALCulate<Ch>[:SELected]:MARKer<Mk>:BWIDth:THReshold?

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:BWIDth:THReshold <numeric>

CALCulate<Ch>:TRACe<Tr>:MARKer<Mk>:BWIDth:THReshold?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_FLATNESS

Description

Turns ON/OFF the marker FLATNESS function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_RF_FILTER_STATS

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_RF_FILTER_STATS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:STATe {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:STATe?

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:STATe {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:STATe?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_FLATNESS_START

Description

Sets/Gets the number of the marker, which specifies the start frequency of the FLATNESS function domain.

Default value: 1

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define
 CMTNA_ATTR_MARKER_MATH_RF_FILTER_PASSBAND_START

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_RF_FILTER_PASSBAND_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:DOMain:STARt <numeric>

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:DOMain:STARt?

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:STARt <numeric>

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:STARt?

Back to Attributes

CMTNA_ATTR_MARKER_MATH_FLATNESS_STOP

Description

Sets/Gets the number of the marker, which specifies the stop frequency of the FLATNESS function domain.

Default value: 2

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_MATH_RF_FILTER_PASSBAND_STOP

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on.The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_MATH_RF_FILTER_PASSBAND_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:DOMain:STOP <numeric>

CALCulate<Ch>[:SELected]:MARKer:MATH:FLATness:DOMain:STOP?

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:STOP <numeric>

CALCulate<Ch>:TRACe<Tr>:MARKer:MATH:FLATness:STOP?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_DISCRETE

Description

Turns ON/OFF the marker discrete mode.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_PROPERTIES_DISCRETE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_DISCRETE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:DISCrete {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:DISCrete?

CALCulate<Ch>:TRACe<Tr>:MARKer:DISCrete {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:DISCrete?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_MARKER_COUPLE

Description

Turns ON/OFF the marker coupling between traces. When coupled the markers of different traces with same number track the X-axis position.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_PROPERTIES_MARKER_COUPLE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_MARKER_COUPLE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:MARKer:COUPle {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:MARKer:COUPle?

CALCulate<Ch>:TRACe<Tr>:MARKer:COUPle {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:MARKer:COUPle?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_MARKER_TABLE

Description

Turns ON/OFF of the marker table.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_PROPERTIES_MARKER_TABLE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_MARKER_TABLE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:MARKer:TABLe[:STATe] {OFF|ON|0|1}

DISPlay:MARKer:TABLe[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_MARKER_ACTIVE_ONLY

Description

Sets/Gets display either of the active trace markers or all trace markers.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define
 CMTNA_ATTR_MARKER_PROPERTIES_MARKER_ACTIVE_ONLY

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_MARKER_ACTIVE_ONLY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:ANNotation:MARKer:SINGle[:STATe] {OFF|ON|0|1}

DISPlay:WINDow<Ch>:ANNotation:MARKer:SINGle[:STATe]?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_ALIGN

Description

Sets/Gets the alignment mode of the marker display position of each trace, when the only active trace display feature is turned OFF by CMTNA_ATTR_MARKER_PROPERTIES_MARKER_ACTIVE_ONLY.

Default value: CMTNA_MARKER_PROPERTIES_ALIGN_OFF

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MARKER_PROPERTIES_ALIGN

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_ALIGN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_MARKER_PROPERTIES_ALIGN_VERTICAL	Vertical alignment
CMTNA_MARKER_PROPERTIES_ALIGN_HIRIZONTAL	Horizontal alignment
CMTNA_MARKER_PROPERTIES_ALIGN_OFF	No alignment

SCPI Command

DISPlay:WINDow<Ch>:ANNotation:MARKer:ALIGn[:TYPE] <char>

DISPlay:WINDow<Ch>:ANNotation:MARKer:ALIGn[:TYPE]?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_DATA_Y_POSITION

Description

Sets/Gets the display position of the marker annotation on the Y–axis by a percentage of the display height. The value is expressed in percentages (%).

Unit: % (percent)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_MARKER_PROPERTIES_DATA_Y_POSITION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_DATA_Y_POSITION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:Y <numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:Y?

Back to Attributes

CMTNA_ATTR_MARKER_PROPERTIES_DATA_X_POSITION

Description

Sets/Gets the display position of the marker annotation on the X–axis by a percentage of the display width. The value is expressed in percentages (%).

Unit: % (percent)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_MARKER_PROPERTIES_DATA_X_POSITION

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Measurement index i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_MARKER_PROPERTIES_DATA_X_POSITION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:X <numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:X?

Back to Attributes

CMTNA_ATTR_ANALYSIS_FIXTURE_SIMULATOR

Description

Turns ON/OFF the fixture simulation function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_FIXTURE_SIMULATOR

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_FIXTURE_SIMULATOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:STATe {OFF|ON|0|1}

CALCulate<Ch>:FSIMulator:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION

Description

Turns ON/OFF the port impedance conversion function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:STATe {OFF|ON|0|1}

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_Z0

Description

Sets/Gets the value of the impedance for port impedance conversion function. The value is expressed in Ohm (Ω).

Unit: Ω (Ohm)

Default value: 50

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_Z0

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus    CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_Z0.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0[:R] <numeric>

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0[:R]?

Back to Attributes

CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_REAL

Description

Sets/ Gets the real part of the impedance of the port impedance conversion function. The value is expressed in Ohm (Ω).

Unit: Ω (Ohm)

Default value: 50

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_REAL

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_REAL.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0:REAL

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0:REAL?

Back to Attributes

CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_IMAG

Description

Sets/ Gets the imaginary part of the impedance of the port impedance conversion function. The value is expressed in Ohm (Ω).

Unit: Ω (Ohm)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_IMAG

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_PORT_ZCONVERSION_IMAG.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0:IMAGinary

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0:IMAGinary?

Back to Attributes

CMTNA_ATTR_ANALYSIS_DEEMBEDDING

Description

Turns ON/OFF the 2-port network de–embedding function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_DEEMBEDDING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_DEEMBEDDING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:STATe {OFF|ON|0|1}

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT

Description

Turns ON/OFF the 2-port network de–embedding function for specified port.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:STATe {OFF|ON|0|1}

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT_FILE

Description

Sets/Gets the name of *.s2p file of the de–embedded circuit of the 2-port network de–embedding function. The file contains the circuit S–parameters in Touchstone format.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT_FILE

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus   CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_DEEMBEDDING_PORT_FILE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:USER:FILename <string>

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:USER:FILename?

Back to Attributes

CMTNA_ATTR_ANALYSIS_EMBEDDING

Description

Turns ON/OFF the 2-port network embedding function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_EMBEDDING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_EMBEDDING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:STATe {OFF|ON|0|1}

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT

Description

Turns ON/OFF the 2-port network embedding function for each port.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:STATe {OFF|ON|0|1}

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT_FILE

Description

Sets/Gets the name of *.s2p file of the embedded circuit of the 2-port network embedding function. The file contains the circuit S–parameters in Touchstone format.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT_FILE

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize
        ViChar value[]
);

ViStatus    CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index and Port index i.e. "Channel1:Port2" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_EMBEDDING_PORT_FILE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:USER:FILename <string>

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:USER:FILename?

Back to Attributes

CMTNA_ATTR_ANALYSIS_TIME_DOMAIN

Description

Turns ON/OFF the time domain transformation function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_TIME_DOMAIN

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

ci

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_TIME_DOMAIN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:STATe {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:TRANsform:TIME:STATe?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STATe {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STATe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_UNITS

Description

Sets/Gets the transformation unit for the time domain transformation function: seconds, meters, feet.

Default value: CMTNA_TIME_DOMAIN_UNITS_SECONDS

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_UNITS

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_UNITS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_TIME_DOMAIN_UNITS_SECONDS	Seconds
CMTNA_TIME_DOMAIN_UNITS_METERS	Meters
CMTNA_TIME_DOMAIN_UNITS_FEET	Feet

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:UNIT <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME:UNIT?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:UNIT <char>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:UNIT?

Back to Attributes

CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_REFLECTION_TYPE

Description

Sets/Gets the reflection distance either one way or round trip for the time domain transformation function.

Default value: CMTNA_TIME_DOMAIN_REFLECTION_ROUND_TRIP

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_REFLECTION_TYPE

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_TIME_DOMAIN_REFLECTION_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_TIME_DOMAIN_REFLECTION_ROUND_TRIP	Round Trip
CMTNA_TIME_DOMAIN_REFLECTION_ONE_WAY	One Way

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:REFLection:TYPE <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME:REFLection:TYPE?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:REFLection:TYPE <char>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:REFLection:TYPE?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_CABLE_CORRECTION

Description

Turns ON/OFF the cable correction when the time domain transformation function is turned ON.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_CABLE_CORRECTION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_CABLE_CORRECTION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:TRANsform:TIME:STATe {OFF|ON|0|1}

SENSe<Ch>:CORRection:TRANsform:TIME:STATe?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_CABLE_VELOCITY_FACTOR

Description

Sets/Gets the cable relative wave speed velocity for the cable correction function, when the time domain transformation function is turned ON.

Default value: 1.0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_TIME_DOMAIN_CABLE_VELOCITY_FACTOR

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_CABLE_VELOCITY_FACTOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:TRANsform:TIME:RVELocity <numeric>

SENSe<Ch>:CORRection:TRANsform:TIME:RVELocity?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_CABLE_LOSS

Description

Sets/Gets the cable loss value for the cable correction function, when the time domain transformation function is turned ON. The value is expressed in decibell/meter (dB/m).

Unit: dB/m (decibell/meter)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_TIME_DOMAIN_CABLE_LOSS

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_CABLE_LOSS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:TRANsform:TIME:LOSS <numeric>

SENSe<Ch>:CORRection:TRANsform:TIME:LOSS?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_CABLE_FREQUENCY

Description

Sets/Gets the frequency value at which the cable loss is specified for the cable correction function, when the time domain transformation function is turned ON. The value is expressed in Hertz (Hz).

Unit: Hz (Hertz)

Default value: 1 GHz

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_TIME_DOMAIN_CABLE_FREQUENCY

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index i.e. "Channel1" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_CABLE_FREQUENCY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SENSe<Ch>:CORRection:TRANsform:TIME:FREQuency <frequency>

SENSe<Ch>:CORRection:TRANsform:TIME:FREQuency?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_START

Description

Sets/Gets the time domain start value, when the time domain transformation function is turned ON. The value is expressed in second (sec).

Unit: sec (second)

Default value: -1e-8

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_START

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:STARt <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STARt?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STARt <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STARt?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_STOP

Description

Sets/Gets the time domain stop value, when the time domain transformation function is turned ON. The value is expressed in second (sec).

Unit: sec (second)

Default value: +1e-8

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:STOP <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STOP?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STOP <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STOP?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_CENTER

Description

Sets/Gets the time domain center value, when the time domain transformation function is turned ON. The value is expressed in second (sec).

Unit: sec (second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_CENTER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_CENTER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:CENTer <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:CENTer?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:CENTer <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:CENTer?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_SPAN

Description

Sets/Gets the time domain span value, when the time domain transformation function is turned ON. The value is expressed in second (sec).

Unit: sec (second)

Default value: 2e-8

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_SPAN

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_SPAN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:SPAN <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:SPAN?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:SPAN <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:SPAN?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_TRANSFORM_TYPE

Description

Sets/Gets the stimulus type for the time domain transformation function.

Default value: CMTNA_TIME_DOMAIN_TRANSFORM_TYPE_BANDPASS

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_TRANSFORM_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_TRANSFORM_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_TIME_DOMAIN_TRANSFORM_TYPE_BANDPASS	Bandpass
CMTNA_TIME_DOMAIN_TRANSFORM_TYPE_LOWPASS_STEP	Lowpass Step
CMTNA_TIME_DOMAIN_TRANSFORM_TYPE_LOWPASS_IMPULSE	Lowpass Impulse

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:STIMulus <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STIMulus?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STIMulus <char>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STIMulus?

CALCulate<Ch>[:SELected]:TRANsform:TIME[:TYPE] <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME[:TYPE]?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME[:TYPE] <char>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME[:TYPE]?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_WINDOW_SHAPE

Description

Sets/Gets the Kaiser–Bessel window shape, when performing time domain transformation.

Default value: CMTNA_TIME_DOMAIN_WINDOW_SHAPE_NORMAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_TIME_DOMAIN_WINDOW_SHAPE

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_WINDOW_SHAPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_TIME_DOMAIN_WINDOW_SHAPE_MAXIMUM	Maximum (β parameter equals 13)
CMTNA_TIME_DOMAIN_WINDOW_SHAPE_NORMAL	Normal (β parameter equals 6)
CMTNA_TIME_DOMAIN_WINDOW_SHAPE_MINIMUM	Minimum (rectangular, β parameter equals 0)
CMTNA_TIME_DOMAIN_WINDOW_SHAPE_ARBITRARY	Arbitrary (β not equal to a fixed value)

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:KBESsel <numeric>

CALCulate<Ch>[:SELected]:TRANsform:TIME:KBESsel?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:KBESsel <numeric>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:KBESsel?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_IMPULSE_WIDTH

Description

Sets/Gets the impulse width (time domain transformation resolution), coupled with the Kaiser–Bessel window shape β parameter. The impulse width setting changes the β parameter, and setting of β parameter changes the impulse width. The value is expressed in seconds (sec).

Unit: sec (second)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_IMPULSE_WIDTH

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_IMPULSE_WIDTH.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:IMPulse:WIDTh <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:IMPulse:WIDTh?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:IMPulse:WIDTh <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:IMPulse:WIDTh?

CALCulate<Ch>[:SELected]:TRANsform:TIME:STEP:RTIMe <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STEP:RTIMe?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STEP:RTIMe <time>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:STEP:RTIMe?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_KAISER_BETA

Description

Sets/Gets the β parameter, which controls the Kaiser–Bessel window shape, when performing time domain transformation.

Default value: 6

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_KAISER_BETA

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_KAISER_BETA.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:TRANsform:TIME:KBESsel <numeric>

CALCulate<Ch>[:SELected]:TRANsform:TIME:KBESsel?

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:KBESsel <numeric>

CALCulate<Ch>:TRACe<Tr>:TRANsform:TIME:KBESsel?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING

Description

Turns ON/OFF the gating function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_GATING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STATe {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STATe?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:STATe {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:STATe?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING_START

Description

Sets/Gets the gate start value of the gating function. The value is expressed in seconds (sec).

Unit: sec (second)

Default value: -1e-8

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_TIME_DOMAIN_GATING_START

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING_START.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STARt <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STARt?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:STARt <time>

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:STARt?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING_STOP

Description

Sets/Gets the gate stop value of the gating function. The value is expressed in seconds (sec).

Unit: sec (second)

Default value: +1e-8

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_GATING_STOP

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING_STOP.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STOP <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STOP?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:STOP <time>

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:STOP?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING_CENTER

Description

Sets/Gets the gate center value of the gating function. The value is expressed in seconds (sec).

Unit: sec (second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_GATING_CENTER

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING_CENTER.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:CENTer <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:CENTer?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:CENTer <time>

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:CENTer?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING_SPAN

Description

Sets/Gets the gate span value of the gating function. The value is expressed in seconds (sec).

Unit: sec (second)

Default value: 2e-8

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_GATING_SPAN

ViStatus  CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus  CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING_SPAN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SPAN <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SPAN?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:SPAN <time>

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:SPAN?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING_TYPE

Description

Sets/Gets the gate type of the gating function.

Default value: CMTNA_TIME_DOMAIN_GATING_BANDPASS

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_GATING_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_TIME_DOMAIN_GATING_BANDPASS	Bandpass type
CMTNA_TIME_DOMAIN_GATING_NOTCH	Notch type

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME[:TYPE] <char>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME[:TYPE]?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME[:TYPE] <char>

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME[:TYPE]?

Back to Attributes

CMTNA_ATTR_TIME_DOMAIN_GATING_SHAPE

Description

Sets/Gets the gate shape of the gating function.

Default value: CMTNA_TIME_DOMAIN_GATING_SHAPE_NORMAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_TIME_DOMAIN_GATING_SHAPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_TIME_DOMAIN_GATING_SHAPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_TIME_DOMAIN_GATING_SHAPE_MAXIMUM	Maximum shape
CMTNA_TIME_DOMAIN_GATING_SHAPE_WIDE	Wide shape
CMTNA_TIME_DOMAIN_GATING_SHAPE_NORMAL	Normal shape
CMTNA_TIME_DOMAIN_GATING_SHAPE_MINIMUM	Minimum shape

SCPI Command

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SHAPe <char>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SHAPe?

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:SHAPe <char>

CALCulate<Ch>:TRACe<Tr>:FILTer[:GATE]:TIME:SHAPe?

Back to Attributes

CMTNA_ATTR_ANALYSIS_CONVERSION

Description

Turns ON/OFF the S–parameter conversion function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_CONVERSION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_CONVERSION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:CONVersion[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:CONVersion[:STATe]?

CALCulate<Ch>:TRACe<Tr>:CONVersion[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:CONVersion[:STATe]?

Back to Attributes

CMTNA_ATTR_ANALYSIS_CONVERSION_FUNCTION

Description

Sets/Gets the S–parameter conversion function type.

Default value: CMTNA_CONVERSION_FUNCTION_Z_REFLECTION

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_CONVERSION_FUNCTION

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus    CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_CONVERSION_FUNCTION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_CONVERSION_FUNCTION_Z_REFLECTION	Reflection equivalent impedance
CMTNA_CONVERSION_FUNCTION_Z_TRANSMISSION	Transmission equivalent impedance
CMTNA_CONVERSION_FUNCTION_Y_REFLECTION	Reflection equivalent admittance
CMTNA_CONVERSION_FUNCTION_Y_TRANSMISSION	Transmission equivalent admittance
CMTNA_CONVERSION_FUNCTION_S_INVERSE	Inverse S–parameter

Name	Description
CMTNA_CONVERSION_FUNCTION_Z_TRANS_SHUNT	Shunt equivalent impedance
CMTNA_CONVERSION_FUNCTION_Y_TRANS_SHUNT	Shunt equivalent admittance
CMTNA_CONVERSION_FUNCTION_CONJUGATION	S–parameter conjugate

SCPI Command

CALCulate<Ch>[:SELected]:CONVersion:FUNCtion <char>

CALCulate<Ch>[:SELected]:CONVersion:FUNCtion?

CALCulate<Ch>:TRACe<Tr>:CONVersion:FUNCtion <char>

CALCulate<Ch>:TRACe<Tr>:CONVersion:FUNCtion?

Back to Attributes

CMTNA_ATTR_ANALYSIS_LIMIT_TEST

Description

Turns ON/OFF the limit test.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_LIMIT_TEST

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_LIMIT_TEST.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:LIMit[:STATe]?

CALCulate<Ch>:TRACe<Tr>:LIMit[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:LIMit[:STATe]?

Back to Attributes

CMTNA_ATTR_ANALYSIS_LIMIT_LINE_DISPLAY

Description

Turns ON/OFF the limit line display of the limit test function.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_LIMIT_LINE_DISPLAY

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier.The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_LIMIT_LINE_DISPLAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:DISPlay[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:LIMit:DISPlay[:STATe]?

CALCulate<Ch>:TRACe<Tr>:LIMit:DISPlay[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:LIMit:DISPlay[:STATe]?

Back to Attributes

CMTNA_ATTR_LIMIT_LINE_STIMULUS_OFFSET

Description

Sets/Gets the value of the limit line offset along X–axis.

Unit: Hz | sec | dBm (second | second | decibel above 1 milliwatt)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_LIMIT_LINE_STIMULUS_OFFSET

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_LIMIT_LINE_STIMULUS_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:OFFSet:STIMulus <stimulus>

CALCulate<Ch>[:SELected]:LIMit:OFFSet:STIMulus?

CALCulate<Ch>:TRACe<Tr>:LIMit:OFFSet:STIMulus <stimulus>

CALCulate<Ch>:TRACe<Tr>:LIMit:OFFSet:STIMulus?

Back to Attributes

CMTNA_ATTR_LIMIT_LINE_RESPONSE_OFFSET

Description

Sets/Gets the value of the limit line offset along Y–axis.

Unit: dB | ° | sec (decibel | degree | second)

Default value: 0

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_LIMIT_LINE_RESPONSE_OFFSET

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);

ViStatus CmtNA_SetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_LIMIT_LINE_RESPONSE_OFFSET.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:OFFSet:AMPLitude <numeric>

CALCulate<Ch>[:SELected]:LIMit:OFFSet:AMPLitude?

CALCulate<Ch>:TRACe<Tr>:LIMit:OFFSet:AMPLitude <numeric>

CALCulate<Ch>:TRACe<Tr>:LIMit:OFFSet:AMPLitude?

Back to Attributes

CMTNA_ATTR_ANALYSIS_LIMIT_TEST_FAIL_SIGN

Description

Turns ON/OFF the "Fail" sign display, when performing limit test.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_LIMIT_TEST_FAIL_SIGN

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_ANALYSIS_LIMIT_TEST_FAIL_SIGN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:FSIGn {OFF|ON|0|1}

DISPlay:FSIGn?

Back to Attributes

CMTNA_ATTR_ANALYSIS_LIMIT_TEST_POINTS

Description

Gets the number of the measurement points that failed the limit test.

The stimulus data array of these points can be read out by CmtNA_GetLimitTestReport function.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_ANALYSIS_LIMIT_TEST_POINTS

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_LIMIT_TEST_POINTS.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:LIMit:REPort:POINts?

CALCulate<Ch>:TRACe<Tr>:LIMit:REPort:POINts?

Back to Attributes

CMTNA_ATTR_ANALYSIS_RIPPLE_TEST

Description

Turns ON/OFF the ripple limit test.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_RIPPLE_TEST

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_RIPPLE_TEST.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit[:STATe] {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:RLIMit[:STATe]?

CALCulate<Ch>:TRACe<Tr>:RLIMit[:STATe] {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:RLIMit[:STATe]?

Back to Attributes

CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_DISPLAY

Description

Turns ON/OFF the ripple limit line display.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_DISPLAY

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_DISPLAY.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit:DISPlay:LINE {OFF|ON|0|1}

CALCulate<Ch>[:SELected]:RLIMit:DISPlay:LINE?

CALCulate<Ch>:TRACe<Tr>:RLIMit:DISPlay:LINE {OFF|ON|0|1}

CALCulate<Ch>:TRACe<Tr>:RLIMit:DISPlay:LINE?

Back to Attributes

CMTNA_ATTR_ANALYSIS_RIPPLE_VALUE_TYPE

Description

Sets/Gets the display type of the ripple value in the specified band.

Default value: CMTNA_ANALYSIS_RIPPLE_VALUE_OFF

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_RIPPLE_VALUE_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. The physical names are supported along with the corresponding Channel index for the active trace of channel i.e. "Channel1" and so on. The physical names are supported along with the corresponding Channel index and Measurement index for the n–th trace of channel i.e. "Channel1:Measurement3" and so on. For valid values, see the Accessing Repeated Capabilities.

attributeID

Must be CMTNA_ATTR_ANALYSIS_RIPPLE_VALUE_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_ANALYSIS_RIPPLE_VALUE_OFF	Ripple value display OFF
CMTNA_ANALYSIS_RIPPLE_VALUE_ABSOLUTE	Absolute value
CMTNA_ANALYSIS_RIPPLE_VALUE_MARGIN	Margin (difference between the ripple limit and the absolute value)

SCPI Command

CALCulate<Ch>[:SELected]:RLIMit:DISPlay:VALue <char>

CALCulate<Ch>[:SELected]:RLIMit:DISPlay:VALue?

CALCulate<Ch>:TRACe<Tr>:RLIMit:DISPlay:VALue <char>

CALCulate<Ch>:TRACe<Tr>:RLIMit:DISPlay:VALue?

Back to Attributes

CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_FAIL_SIGN

Description

Turns ON/OFF the "Fail" sign display, when ripple limit test.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_FAIL_SIGN

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_ANALYSIS_RIPPLE_LIMIT_FAIL_SIGN.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

DISPlay:FSIGn {OFF|ON|0|1}

DISPlay:FSIGn?

Back to Attributes

CMTNA_ATTR_SAVE_TYPE

Description

Sets/Gets the type of the Analyzer or channel state saving by CmtNA_SaveChannelToRegister function.

Default value: CMTNA_SAVE_TYPE_STATE_AND_CAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SAVE_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_SAVE_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_SAVE_TYPE_STATE	Measurement conditions
CMTNA_SAVE_TYPE_STATE_AND_CAL	Measurement conditions and calibration
CMTNA_SAVE_TYPE_STATE_AND_TRACE	Measurement conditions and data
CMTNA_SAVE_TYPE_ALL	Measurement conditions, calibration, data and memory
CMTNA_SAVE_TYPE_STATE_AND_CAL_AND_MEM	Measurement conditions, calibration and memory

SCPI Command

MMEMory:STORe:STYPe <char>

MMEMory:STORe:STYPe?

Back to Attributes

CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_FORMAT

Description

Sets/Gets the data format for the S–parameter saving by CmtNA_SaveTouchstoneFile function.

The attribute is used for the active channel.

Default value: CMTNA_SAVE_TOUCHSTONE_FILE_FORMAT_DB_ANGLE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_FORMAT

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_FORMAT.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_SAVE_TOUCHSTONE_FILE_FORMAT_REAL_IMAGE	Real part /Imaginary part format
CMTNA_SAVE_TOUCHSTONE_FILE_FORMAT_MAGNITUDE_ANGLE	Logarithmic Magnitude / Angle format
CMTNA_SAVE_TOUCHSTONE_FILE_FORMAT_DB_ANGLE	Linear Magnitude / Angle format

SCPI Command

MMEMory:STORe:SNP:FORMat <char>

MMEMory:STORe:SNP:FORMat?

Back to Attributes

CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARATOR

Description

Sets/Gets the Touchstone file separator symbol when the S–parameters are saved by CmtNA_SaveTouchstoneFile function.

The attribute is used for the active channel.

Default value: CMTNA_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARATOR_TAB

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define
 CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARATOR

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

The repeated capability identifier. Pass VI_NULL or empty string if operation does not apply to a repeated capability.

attributeID

Must be CMTNA_ATTR_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARATOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARATOR_TAB	Tab symbol (0x09)
CMTNA_SAVE_TOUCHSTONE_FILE_COLUMN_SEPARATOR_SPACE	Space symbol (0x20)

SCPI Command

MMEMory:STORe:SNP:SEParator <char>

MMEMory:STORe:SNP:SEParator?

Back to Attributes

CMTNA_ATTR_PRINT_COLOR

Description

Sets/Gets the color chart for the image printout.

Default value: CMTNA_PRINT_BLACK_AND_WHITE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PRINT_COLOR

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_PRINT_COLOR.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_PRINT_COLOR	Color printout
CMTNA_PRINT_GRAY_SCALE	Grayscale printout
CMTNA_PRINT_BLACK_AND_WHITE	Black&white printout

SCPI Command

HCOPy:PAINt <char>

HCOPy:PAINt?

Back to Attributes

CMTNA_ATTR_PRINT_INVERT_IMAGE

Description

Turns ON/OFF the inverted color image printout.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PRINT_INVERT_IMAGE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_PRINT_INVERT_IMAGE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

HCOPy:IMAGe <char>

HCOPy:IMAGe?

Back to Attributes

CMTNA_ATTR_PRINT_DATE_AND_TIME

Description

Turns ON/OFF the date and time printout in the upper right corner of the image.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_PRINT_DATE_AND_TIME

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_PRINT_DATE_AND_TIME.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

HCOPy:DATE:STAMp {OFF|ON|0|1}

HCOPy:DATE:STAMp?

Back to Attributes

CMTNA_ATTR_REFERENCE_FREQUENCY_SOURCE

Description

Sets/Gets the internal or external source of the reference frequency of 10 MHz.

Default value: CMTNA_REFERENCE_FREQUENCY_SOURCE_INTERNAL

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_REFERENCE_FREQUENCY_SOURCE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_REFERENCE_FREQUENCY_SOURCE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_REFERENCE_FREQUENCY_SOURCE_INTERNAL	Internal source of the reference frequency
CMTNA_REFERENCE_FREQUENCY_SOURCE_EXTERNAL	External source of the reference frequency

SCPI Command

SENSe<Ch>:ROSCillator:SOURce <char>

SENSe<Ch>:ROSCillator:SOURce?

Back to Attributes

CMTNA_ATTR_EXTERNAL_REFERENCE_ROUTE

Description

Sets/Gets the route of the external 10 MHz reference frequency. (PXIe-S5090 model only). The source of the reference (see CMTNA_ATTR_REFERENCE_FREQUENCY_SOURCE) must be set to the EXTernal.

Default value: CMTNA_EXTERNAL_REFERENCE_ROUTE_REAR

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_EXTERNAL_REFERENCE_ROUTE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_EXTERNAL_REFERENCE_ROUTE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_EXTERNAL_REFERENCE_ROUTE_REAR	Rear panel
CMTNA_EXTERNAL_REFERENCE_ROUTE_FRONT	Front panel

SCPI Command

SENSe:ROSCillator:EXTernal:ROUTe <char>

SENSe:ROSCillator:EXTernal:ROUTe?

Back to Attributes

CMTNA_ATTR_SYSTEM_CORRECTION

Description

Turns ON/OFF the system correction. The system correction is the factory full 1-port calibration performed at the port connectors.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_SYSTEM_CORRECTION

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SYSTEM_CORRECTION.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:CORRection[:STATe] {OFF|ON|0|1}

SYSTem:CORRection[:STATe]?

Back to Attributes

CMTNA_ATTR_POWER_TRIP_AT_OVERLOAD

Description

Turns ON/OFF the Power Trip at Overload function. Except for Planar-804/808/304 Models.

Default value: VI_FALSE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_POWER_TRIP_AT_OVERLOAD

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_POWER_TRIP_AT_OVERLOAD.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:RECeiver:OVERload:POWer[:STATe] {OFF|ON|0|1}

SYSTem:RECeiver:OVERload:POWer[:STATe]?

Back to Attributes

CMTNA_ATTR_POWER_SENSOR_TYPE

Description

Sets/Gets the power sensor type to be used in a source power calibration.

Default value: CMTNA_POWER_SENSOR_NRPZ

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_POWER_SENSOR_TYPE

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_POWER_SENSOR_TYPE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Defined Values

Name	Description
CMTNA_POWER_SENSOR_NRPZ	Rohde&Schwarz NRP-Z series Sensors
CMTNA_POWER_SENSOR_NRPXT	Rohde&Schwarz NRPxT series Sensors
CMTNA_POWER_SENSOR_NRVS	Rohde&Schwarz NRVS power meter
CMTNA_POWER_SENSOR_U848X	Keysight U848x series Sensors
CMTNA_POWER_SENSOR_U200X	Keysight U200x series Sensors

SCPI Command

SYSTem:COMMunicate:PSENsor:TYPE <char>

SYSTem:COMMunicate:PSENsor:TYPE?

Back to Attributes

CMTNA_ATTR_BEEP_COMPLETE_ON

Description

Turns ON/OFF the beeper notifying of the completion of the operation.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_BEEP_COMPLETE_ON

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_BEEP_COMPLETE_ON.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:BEEPer:COMPlete:STATe {OFF|ON|0|1}

SYSTem:BEEPer:COMPlete:STATe?

Back to Attributes

CMTNA_ATTR_BEEP_WARNING

Description

Turns ON/OFF the beeper notifying of warning.

Default value: VI_TRUE

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_BEEP_WARNING

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_BEEP_WARNING.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:BEEPer:WARNing:STATe {OFF|ON|0|1}

SYSTem:BEEPer:WARNing:STATe?

Back to Attributes

CMTNA_ATTR_SYSTEM_READ_WRITE_LOCK

Description

Sets the Analyzer to the local operation mode or remote operation mode, when all the keys on the front panel, mouse and the touch screen are active.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_SYSTEM_READ_WRITE_LOCK

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SYSTEM_READ_WRITE_LOCK.

value

The value to which to set the attribute.

SCPI Command

SYSTem:LOCal

SYSTem:RWLock

Back to Attributes

CMTNA_ATTR_SYSTEM_TIME_OUT_MILLISECONDS

Description

I/O timeout value in milliseconds. This value is expressed in milliseconds (ms).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SYSTEM_TIME_OUT_MILLISECONDS

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SYSTEM_TIME_OUT_MILLISECONDS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_VERIFICATION_INTERVAL

Description

Sets/Gets the interval between Instrument Performance Verifications.

One year (365 days) is recommended.

This value is expressed in days.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_VERIFICATION_INTERVAL

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);

ViStatus  CmtNA_SetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_VERIFICATION_INTERVAL.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:SERVice:PVERify:INTerval <numeric>

SYSTem:SERVice:PVERify:INTerval?

Back to Attributes

CMTNA_ATTR_VERIFICATION_LAST_DATE

Description

Sets/Gets the date of the last Instrument Performance Verification. The date format: "YYYY,MM,DD" (YYYY - year, MM - month, DD - day).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_VERIFICATION_LAST_DATE

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);

ViStatus  CmtNA_SetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViConstString value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_VERIFICATION_LAST_DATE.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

SCPI Command

SYSTem:SERVice:PVERify:LAST <numeric 1>,<numeric 2>,<numeric 3>

SYSTem:SERVice:PVERify:LAST?

Back to Attributes

CMTNA_ATTR_VERIFICATION_NEXT_DATE

Description

Gets the date of the next Instrument Performance Verification. The date format: "YYYY,MM,DD" (YYYY - year, MM - month, DD - day).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_VERIFICATION_NEXT_DATE

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_VERIFICATION_NEXT_DATE.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SYSTem:SERVice:PVERify:NEXT?

Back to Attributes

CMTNA_ATTR_DEVICE_READY

Description

Gets the ready state of the Analyzer. The state is True when analyzer hardware is connected, powered and the boot process is completed (about 15 sec).

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DEVICE_READY

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_DEVICE_READY.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SYSTem:READy[:STATe]?

Back to Attributes

CMTNA_ATTR_DEVICE_PXI_CHASSIS

Description

Gets the identifier for the PXI chassis in which the CMT vector network analyzer is installed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DEVICE_PXI_CHASSIS

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_DEVICE_PXI_CHASSIS.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

Back to Attributes

CMTNA_ATTR_DEVICE_PXI_SLOT

Description

Gets the number for the PXI slot in which the CMT vector network analyzer is installed.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DEVICE_PXI_SLOT

ViStatus CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_DEVICE_PXI_SLOT.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

Back to Attributes

CMTNA_ATTR_DEVICE_TEMPERATURE

Description

Gets the specified sensor temperature inside the Analyzer. The value is expressed in Celsius degrees (°C).

Unit: °C (Celsius degrees)

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DEVICE_TEMPERATURE

ViStatus CmtNA_GetAttributeViReal64(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViReal64 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_DEVICE_TEMPERATURE.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SYSTem:TEMPerature:SENSor? <numeric>

Back to Attributes

CMTNA_ATTR_DEVICE_SERIAL_NUMBER

Description

Gets the instrument serial number.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_DEVICE_SERIAL_NUMBER

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_DEVICE_SERIAL_NUMBER.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

SCPI Command

*IDN?

Back to Attributes

CMTNA_ATTR_NUMBER_OF_PORTS

Description

Gets the number of the ports.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_NUMBER_OF_PORTS

ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_NUMBER_OF_PORTS.

value

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

SCPI Command

SERVice:PORT:COUNt?

Back to Attributes

CMTNA_ATTR_LOGICAL_NAME

Description

Logical Name identifies a driver session in the Configuration Store. If Logical Name is not empty, the driver was initialized from information in the driver session. If it is empty, the driver was initialized without using the Configuration Store.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_LOGICAL_NAME

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_LOGICAL_NAME.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_AUTOCAL_MODULE_READY

Description

Gets the ready state of the AutoCal module. The state is True when the AutoCal module is connected.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_AUTOCAL_MODULE_READY

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_AUTOCAL_MODULE_READY.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_POWER_SENSOR_READY

Description

Gets the ready state of the power sensor. The state is True when the power sensor is ready.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_POWER_SENSOR_READY

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_POWER_SENSOR_READY.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_RANGE_CHECK

Description

Drivers may always validate some property/parameter values, or never validate others, or optionally validate others, to avoid sending invalid commands to the instrument. If True, the driver performs optional validations.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_RANGE_CHECK

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_RANGE_CHECK.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_QUERY_INSTRUMENT_STATUS

Description

If True, the driver queries the instrument status at the end of each method or property that performs I/O to the instrument. If an error is reported, use ErrorQuery to retrieve error messages one at a time from the instrument.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_QUERY_INSTRUMENT_STATUS

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_QUERY_INSTRUMENT_STATUS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_CACHE

Description

Drivers may always cache some instrument settings, or never cache others, or optionally cache others, to avoid unnecessary I/O to the instrument. If True, the driver caches optionally cached instrument settings.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_CACHE

ViStatus CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_CACHE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_SIMULATE

Description

If True, the driver does not perform I/O to the instrument, and returns simulated values for output parameters.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SIMULATE

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SIMULATE.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_RECORD_COERCIONS

Description

Not Supported.

If True, the driver keeps a list of the value coercions it makes for ViInt32 and ViReal64 attributes. If the driver does not support coercion recording, attempts to set RecordCoercions to True will return an error.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_RECORD_COERCIONS

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_RECORD_COERCIONS.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_DRIVER_SETUP

Description

The DriverSetup string is used to set the initial values for attributes that are specific to the driver. The format of this string is "AttributeName=Value," where AttributeName is the name of the attribute and Value is the value to which the attribute is set. To set multiple attributes, separate their assignments with a comma.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_DRIVER_SETUP

ViStatus CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be Must be CMTNA_ATTR_DRIVER_SETUP.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_INTERCHANGE_CHECK

Description

Not Supported.

If True, the driver maintains a record of interchangeability warnings. If the driver does not support interchangeability checking, attempts to set InterchangeCheck to True return an error.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_INTERCHANGE_CHECK

ViStatus  CmtNA_GetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean *value
);

ViStatus  CmtNA_SetAttributeViBoolean(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViBoolean value,
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_INTERCHANGE_CHECK.

value (GetAttribute)

Returns the current value of the attribute. The user must specify the address of a variable that has the same data type as the attribute.

value (SetAttribute)

The value to which to set the attribute.

Back to Attributes

CMTNA_ATTR_SUPPORTED_INSTRUMENT_MODELS

Description

A comma-separated list of instrument models that the IVI specific driver can control. The string does not include an abbreviation for the manufacturer if it is the same for all models.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SUPPORTED_INSTRUMENT_MODELS

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be Must be CMTNA_ATTR_SUPPORTED_INSTRUMENT_MODELS.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_INSTRUMENT_MANUFACTURER

Description

Gets the Analyzer identification string.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_INSTRUMENT_MANUFACTURER

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be Must be CMTNA_ATTR_INSTRUMENT_MANUFACTURER.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

SCPI Command

*IDN?

Back to Attributes

CMTNA_ATTR_INSTRUMENT_MODEL

Description

Gets the Analyzer identification string.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_INSTRUMENT_MODEL

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be Must be CMTNA_ATTR_INSTRUMENT_MODEL.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

SCPI Command

*IDN?

Back to Attributes

CMTNA_ATTR_INSTRUMENT_FIRMWARE_REVISION

Description

Gets the Analyzer identification string.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_INSTRUMENT_FIRMWARE_REVISION

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be Must be CMTNA_ATTR_INSTRUMENT_FIRMWARE_REVISION.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

SCPI Command

*IDN?

Back to Attributes

CMTNA_ATTR_SPECIFIC_DRIVER_PREFIX

Description

Returns the case-sensitive prefix of the user-callable functions that the IVI-C specific driver exports. The string that this attribute returns contains a maximum of 32 bytes including the NUL byte.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SPECIFIC_DRIVER_PREFIX

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SPECIFIC_DRIVER_PREFIX.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_SPECIFIC_DRIVER_REVISION

Description

Returns version information about the IVI specific driver. The string that this attribute returns contains a maximum of 256 bytes including the NUL byte.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SPECIFIC_DRIVER_REVISION

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SPECIFIC_DRIVER_REVISION.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_SPECIFIC_DRIVER_VENDOR

Description

Returns the name of the vendor that supplies the IVI specific driver. The string that this attribute returns contains a maximum of 256 bytes including the NUL byte.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define  CMTNA_ATTR_SPECIFIC_DRIVER_VENDOR

ViStatus  CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[]
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SPECIFIC_DRIVER_VENDOR.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_SPECIFIC_DRIVER_DESCRIPTION

Description

Returns a brief description of the IVI specific driver. The string that this attribute returns contains a maximum of 256 bytes including the NUL byte.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define CMTNA_ATTR_SPECIFIC_DRIVER_DESCRIPTION

ViStatus CmtNA_GetAttributeViString(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 bufferSize,
        ViChar value[],
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SPECIFIC_DRIVER_DESCRIPTION.

bufferSize

The number of bytes in the ViChar array that the user specifies for the AttributeValue parameter.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MAJOR_VERSION

Description

Returns the major version number of the class specification in accordance with which the IVI specific driver was developed. Zero is returned if the driver is not compliant with a class specification.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define
 CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MAJOR_VERSION


ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MAJOR_VERSION.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MINOR_VERSION

Description

Returns the minor version number of the class specification in accordance with which the IVI specific driver was developed. Zero is returned if the driver is not compliant with a class specification.

Declaration: CmtNA.h

Implementation: CmtNA.dll

Syntax

C++

```
#define
 CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MINOR_VERSION


ViStatus  CmtNA_GetAttributeViInt32(
        ViSession vi,
        ViConstString repCapID,
        ViAttr attributeID,
        ViInt32 *value
);
```

Parameters

vi

The ViSession handle that is obtained from the IviDriver_init or IviDriver_InitWithOptions function. The handle identifies a particular instrument session.

repCapID

Must be VI_NULL or an empty string. This attribute is not defined on a repeated capability.

attributeID

Must be CMTNA_ATTR_SPECIFIC_DRIVER_CLASS_SPEC_MINOR_VERSION.

value

The buffer in which the function returns the current value of the attribute. Can be VI_NULL if *bufferSize* is 0.

Back to Attributes

Errors and Warnings

Warning

Name	Description
CMTNA_SUCCESS	Success.
CMTNA_WARN_NSUP_ID_QUERY	ID Query is not supported by this instrument.
CMTNA_WARN_NSUP_RESET	Reset is not supported by this instrument.
CMTNA_WARN_NSUP_SELF_TEST	Self test is not supported by this instrument.
CMTNA_WARN_NSUP_ERROR_QUERY	Error query is not supported by this instrument.
CMTNA_WARN_NSUP_REV_QUERY	Firmware revision query is not supported by this instrument.

Error

Name	Description
CMTNA_ERROR_CANNOT_RECOVER	Failure cannot recover.
CMTNA_ERROR_INSTRUMENT_STATUS	Instrument error is detected. Use CmtNA_error_query to determine the error(s).
CMTNA_ERROR_CANNOT_OPEN_FILE	Cannot open file.
CMTNA_ERROR_READING_FILE	Error reading file.
CMTNA_ERROR_WRITING_FILE	Error writing file.
CMTNA_ERROR_INVALID_PATHNAME	The pathname is invalid.

Name	Description
CMTNA_ERROR_INVALID_VALUE	Invalid value for method or parameter.
CMTNA_ERROR_FUNCTION_NOT_SUPPORTED	Does not support this class-compliant feature.
CMTNA_ERROR_ATTRIBUTE_NOT_SUPPORTED	Does not support this class-compliant feature.
CMTNA_ERROR_VALUE_NOT_SUPPORTED	Does not support this class-compliant feature.
CMTNA_ERROR_NOT_INITIALIZED	A connection to the instrument has not been established.
CMTNA_ERROR_UNKNOWN_CHANNEL_NAME	Unknown channel name.
CMTNA_ERROR_TOO_MANY_OPEN_FILES	Too many files are open.
CMTNA_ERROR_CHANNEL_NAME_REQUIRED	A channel name is required.
CMTNA_ERROR_MEASUREMENT_NAME_REQUIRED	A measurement name is required.
CMTNA_ERROR_MISSING_OPTION_NAME	The option string is missing an option name.
CMTNA_ERROR_MISSING_OPTION_VALUE	The option string is missing an option value.
CMTNA_ERROR_BAD_OPTION_NAME	The name in the option string is unknown.
CMTNA_ERROR_BAD_OPTION_VALUE	The value in the option string is unknown.
CMTNA_ERROR_OUT_OF_MEMORY	Could not allocate necessary memory.

Name	Description
CMTNA_ERROR_OPERATION_PENDING	Operation is in progress.
CMTNA_ERROR_NULL_POINTER	Null pointer passed for method or parameter.
CMTNA_ERROR_UNEXPECTED_RESPONSE	Unexpected response from instrument.
CMTNA_ERROR_FILE_NOT_FOUND	File is not found.
CMTNA_ERROR_INVALID_FILE_FORMAT	Invalid file format.
CMTNA_ERROR_STATUS_NOT_AVAILABLE	The instrument status is not available.
CMTNA_ERROR_ID_QUERY_FAILED	Instrument ID query failed.
CMTNA_ERROR_RESET_FAILED	Instrument reset failed.
CMTNA_ERROR_RESOURCE_UNKNOWN	Unknown resource.
CMTNA_ERROR_ALREADY_INITIALIZED	The driver is already initialized.
CMTNA_ERROR_CANNOT_CHANGE_SIMULATION_STATE	The simulation state cannot be changed.
CMTNA_ERROR_BADLY_FORMED_SELECTOR	The repeated capability selector is badly-formed.
CMTNA_ERROR_IO_GENERAL	General.
CMTNA_ERROR_IO_TIMEOUT	Timeout.
CMTNA_ERROR_MODEL_NOT_SUPPORTED	Model is not supported.
CMTNA_ERROR_INVALID_SESSION_INSTANCE_DATA	Invalid session instance data.

Name	Description
CMTNA_ERROR_INVALID_ATTRIBUTE	Attribute ID is not recognized.
CMTNA_ERROR_TYPES_DO_NOT_MATCH	Wrong attribute accessor invoked for attribute type.
CMTNA_ERROR_IVI_ATTR_NOT_WRITABLE	Attribute is read-only.
CMTNA_ERROR_IVI_ATTR_NOT_READABLE	Attribute is write-only.
CMTNA_ERROR_COMMAND	Command error.
CMTNA_ERROR_UNMATCHED_QUOTE	Unmatched quote.
CMTNA_ERROR_UNMATCHED_BRACKET	Unmatched bracket.
CMTNA_ERROR_INVALID_NUMERIC_LIST	Invalid value in numeric list.
CMTNA_ERROR_DATA_TYPE	Data type error.
CMTNA_ERROR_PARAMETER_OVERFLOW	Numeric parameter overflow.
CMTNA_ERROR_WRONG_UNIT	Wrong units in numeric data.
CMTNA_ERROR_PARAMETER_NOT_ALLOWED	Parameter is not allowed.
CMTNA_ERROR_PARAMETER_MISSING	Missing parameter.
CMTNA_ERROR_COMMAND_HEADER	Command header error.
CMTNA_ERROR_SUFFIX_OUT_RANGE	Header suffix is out of range.
CMTNA_ERROR_INPUT_BUFFER_FULL	Input buffer is full.
CMTNA_ERROR_SUFFIX	Suffix error.
CMTNA_ERROR_EXECUTION	Execution error.

Name	Description
CMTNA_ERROR_INVALID_CHANNEL_INDEX	Invalid repCapID index.
CMTNA_ERROR_INVALID_TRACE_INDEX	Invalid trace index.
CMTNA_ERROR_INVALID_MARKER_INDEX	Invalid marker index.
CMTNA_ERROR_MARKER_NOT_ENABLED	Marker is not enabled.
CMTNA_ERROR_INVALID_SAVE_TYPE	Invalid save type specifier.
CMTNA_ERROR_INVALID_SWEEP_TYPE	Invalid sweep type specifier.
CMTNA_ERROR_TRIGGER_SOURCE_SPECIFIER	Invalid trigger source specifier.
CMTNA_ERROR_MEASUREMENT_PARAMETER_SPECIFIER	Invalid measurement parameter specifier.
CMTNA_ERROR_INVALID_FORMAT_SPECIFIER	Invalid format specifier.
CMTNA_ERROR_INVALID_DATA_MATH	Invalid data math specifier.
CMTNA_ERROR_TRIGGER_IGNORED	Trigger is ignored.
CMTNA_ERROR_INVALID_TRIGGER_SOURCE	Invalid trigger source.
CMTNA_ERROR_INIT_IGNORED	Init is ignored.
CMTNA_ERROR_INVALID_LIMIT_DATA	Invalid limit data.
CMTNA_ERROR_INVALID_SEGMENT_DATA	Invalid segment data.
CMTNA_ERROR_INVALID_STD_TYPE	Invalid standard type specifier.
CMTNA_ERROR_INVALID_CONVERSION_SPECIFIER	Invalid conversion specifier.

Name	Description
CMTNA_ERROR_INVALID_GATING_SHAPE	Invalid gating shape specifier.
CMTNA_ERROR_INVALID_GATING_TYPE	Invalid gating type specifier.
CMTNA_ERROR_IN_PARAMETER	Parameter Error.
CMTNA_ERROR_INVALID_PORT_INDEX	Invalid port index.
CMTNA_ERROR_DATA_OUT_RANGE	Data is out of range.
CMTNA_ERROR_NO_CALIBRATION_DATA	No Calibration Measurement Data.
CMTNA_ERROR_ILLEGAL_PARAMETER	Illegal parameter value.
CMTNA_ERROR_CALKIT_DEFINITION	Cal Kit Definition Error.
CMTNA_ERROR_DIFFER_THRU	Differ Forward and Reverse Thru.
CMTNA_ERROR_DIFFER_TRL_THRU	Differ Forward and Reverse TRL Thru.
CMTNA_ERROR_DIFFER_LINE	Differ Forward and Reverse Line.
CMTNA_ERROR_TRL_MATH	TRL Match Standard is not Load Type Standard.
CMTNA_ERROR_AUTOORIENTATION	ACM Auto-Orientation Error.
CMTNA_ERROR_ORIENTATION_SETTING	ACM Orientation Settings Error.
CMTNA_ERROR_AUTOCAL_EXECUTION	AutoCal Execution Error.
CMTNA_ERROR_ACM_SETTINGS	ACM Characterization Error.

Name	Description
CMTNA_ERROR_ACM_CHARACTERIZATION	ACM Characterization Error.
CMTNA_ERROR_ACM_EXCEEDS_RANGE	Frequency Range Exceeds ACM Characterization Frequency Range.
CMTNA_ERROR_ACM_READING	AutoCal Module Reading Error.
CMTNA_ERROR_INCORRECT_PARAMETERS	Incorrect set of measured parameters.
CMTNA_ERROR_CALIBRATION_EXECUTION	Calibration Execution Error.
CMTNA_ERROR_TRIG_SING_INTERRUPTED	TRIG:SING interrupted.
CMTNA_ERROR_DEVICE_NOT_READY	Analyzer is not ready.
CMTNA_ERROR_ACM_NOT_READY	AutoCal Module is not ready.
CMTNA_ERROR_INVALID_TRIGGER_SCOPE	Invalid trigger scope specifier.
CMTNA_ERROR_TRIGGER_POLARITY	Invalid trigger polarity specifier.
CMTNA_ERROR_INVALID_TRIGGER_POSITION	Invalid trigger position specifier.
CMTNA_ERROR_FILE_MISSING	File is not found.
CMTNA_ERROR_DEVICE_SPECIFIC	Device – specific error.
CMTNA_ERROR_STATUS_REPORTING	Status reporting system error.
CMTNA_ERROR_QUERY	Query error.
CMTNA_ERROR_QUERY_NO_DATA	Query error: no data.

Name	Description
CMTNA_ERROR_QUERY_TRUNCATED	Query truncated.
CMTNA_ERROR_QUERY_INTERRUPTED	Query Interrupted.
CMTNA_ERROR_PSEN_NOT_READY	Power Sensor is not ready
CMTNA_ERROR_INVALID_SESSION_HANDLE	Invalid session handle.
CMTNA_ERROR_CANNOT_CREATE_LOCK	Can not create a lock.
CMTNA_ERROR_CANNOT_CREATE_THREAD_LOCAL	Can not create a local thread.
VI_ERROR_SYSTEM_ERROR	The VISA system failed to initialize.
VI_ERROR_LIBRARY_NFOUND	VISA: Library is not found.
VI_ERROR_INV_SETUP	VISA: Setup is invalid.
VI_ERROR_NSUP_OPER	VISA: Not supported operation.
VI_ERROR_INV_RSRC_NAME	VISA: Invalid resource name.
VI_ERROR_INV_ACC_MODE	VISA: Invalid access mode.
VI_ERROR_RSRC_NFOUND	VISA: Resource is not found.
VI_ERROR_RSRC_BUSY	VISA: Resource is busy.
VI_ERROR_INTF_NUM_NCONFIG	VISA: The specified interface number is not configured.
VI_ERROR_INV_SESSION	VISA: Invalid session handle.
VI_ERROR_RSRC_LOCKED	VISA: Resource is locked.

Name	Description
VI_ERROR_IO	VISA: General I/O error.
VI_ERROR_TMO	VISA: Timeout expired before operation completed.
VI_ERROR_INV_FMT	VISA: Invalid format.
VI_ERROR_NSUP_FMT	VISA: Not supported command.
VI_ERROR_ALLOC	VISA: Could not allocate necessary memory.
VI_ERROR_RAW_WR_PROT_VIOL	VISA: Violation of raw write protocol occurred during transfer.
VI_ERROR_RAW_RD_PROT_VIOL	VISA: Violation of raw read protocol occurred during transfer.
VI_ERROR_CONN_LOST	VISA: The I/O connection for the given session has been lost.
VI_ERROR_INV_MASK	VISA: The specified mask is invalid.

Installation

Updating to a New Driver Version

It is important to note that only a single version of an IVI driver can be installed on a system at once. This rule is dictated by the IVI specifications. To ensure that only a single version of the CmtNA driver exists on the machine, the CmtNA installer will automatically uninstall a previous version of the driver. Although the driver can be always explicitly uninstalled, this is typically not necessary.

Location of Files

The CmtNA installer deploys and registers a variety of files on the target computer. The notation <IviInstallDir> is used to denote the directory in which the IVI Shared Components have been installed. By default, this is C:\Program Files\IVI Foundation\IVI. Substitute the install directory if the IVI Shared Components were installed in an alternate location.

File	Required
<IviInstallDir>\Bin\CmtNA.dll	Driver DLL.
<IviInstallDir>\Include\CmtNA.h	Driver header file.
<IviInstallDir>\Lib\msc\CmtNA.lib	Microsoft-format import library.
<IviInstallDir>\Drivers\CMTNA\CmtNA_IVI_C_Reference.chm	CHM help file.

IVI Shared Components

Before any IVI driver is installed on a system, the IVI Shared Components must be properly installed. For an introduction to the IVI Shared Components.

For additional information on the IVI Shared Components, visit the IVI Foundation web site.

To determine which version of the IVI Shared Components is installed on your system

1. Navigate to <IviInstallDir>\Bin. Typically this is C:\Program Files\IVI Foundation\IVI\Bin.

2. Locate the file IviSharedComponentVersion.dll.

3. Right-click on that file and choose Properties.

4. In the Properties Window, click the Version tab.

5. The File version field indicates which version of the IVI Shared Components is installed.

Copyright

IVI Foundation Copyright Notice