SDM Series Digital Multimeter

Programming guide

EN02A

SIGLENT TECHNOLOGIES CO.,LTD

Table of Contents

Copyright and Declaration

Trademark Information

SIGLENT is the registered trademark of SIGLENT TECHNOLOGIES CO., LTD.

Declaration

SIGLENT products are protected by patent law in and outside of P.R.C.

SIGLENT reserves the right to modify or change parts of or all the specifications or pricing policies at the company's sole decision.

Information in this publication replaces all previously corresponding material.

Product Certification

SIGLENT guarantees this product conforms to the national and industrial standards in China and other international standard conformance certifications are in progress.

Software Requirement

This chapter introduces how to build communication between the instrument and the PC. It also introduces how to configure a system for remote instrument control.

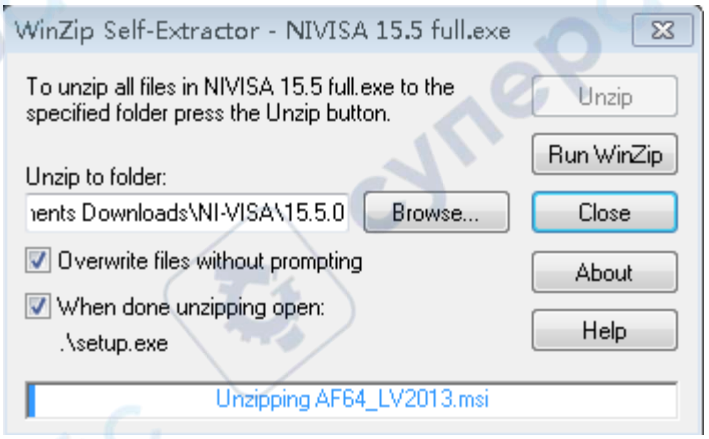
Users can remotely control the instrument through USB and LAN interfaces, in combination with National Instruments NI VISA and programming languages .Through the LAN interface, users can communicate using VXI 11, Sockets and Telnet protocols , depending on the capabilities of the specific instrument

Install NI-MAX

Currently, NI-VISA is packaged in two versions: Full version and Run-Time Engine version. The full version includes the NI device drivers and a tool named NI-MAX which is a user interface to control and test remotely connected devices. You need to install the full version of NI-VISA.

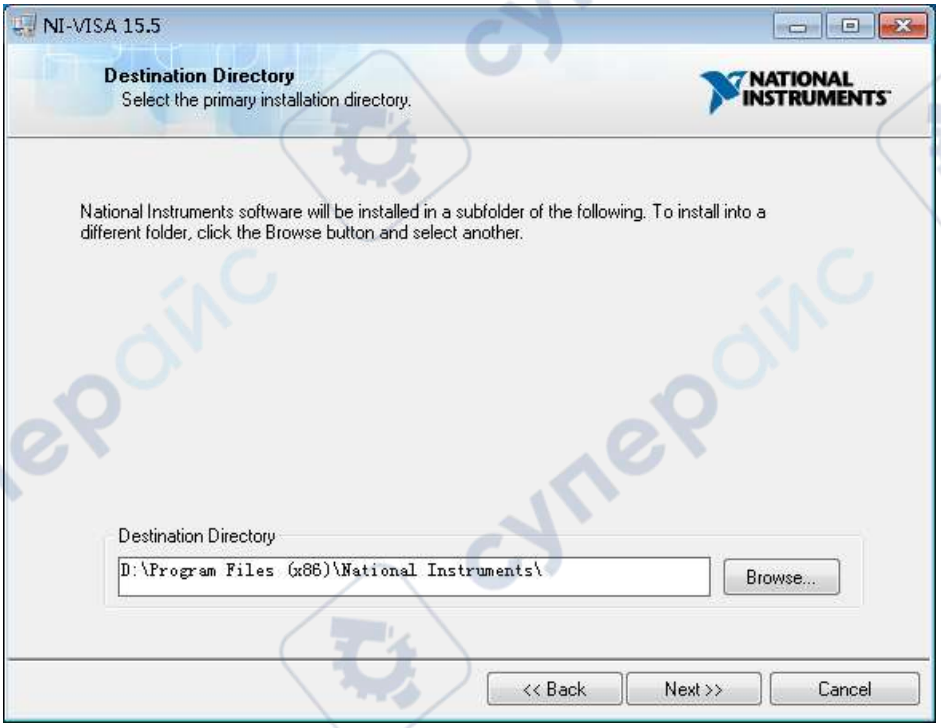
You can get the NI-VISA 15.5 full version or higher version from *https://www.ni.com/en-us/support/downloads/drivers/download.ni-visa.html#306031*.

a. Double click the NIVISA 15.5 full.exe, a dialog will be shown as below:

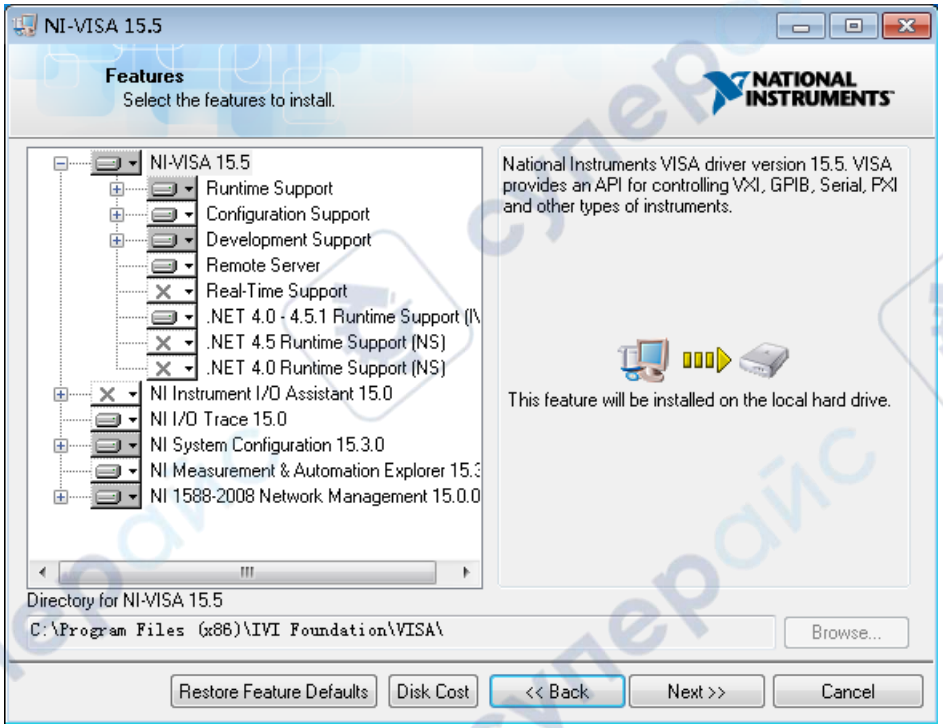


b. Click Unzip, the installation process will automatically launch after unzipping files. If your computer needs to install .NET Framework 4, it may auto start.

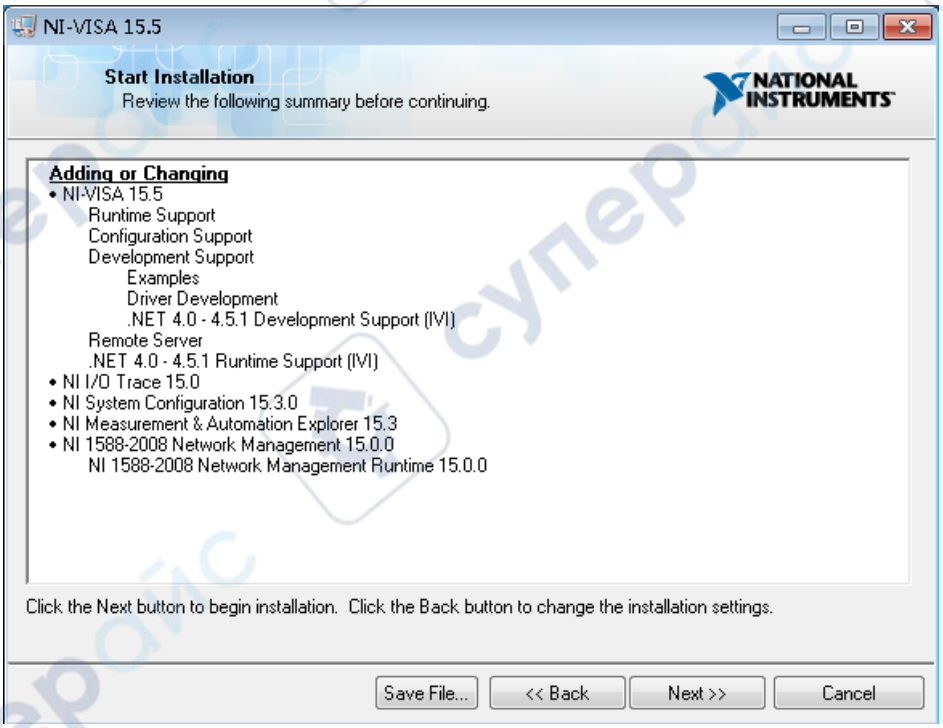
c. The NI-VISA installing dialog is shown above. Click Next to start the installation process.



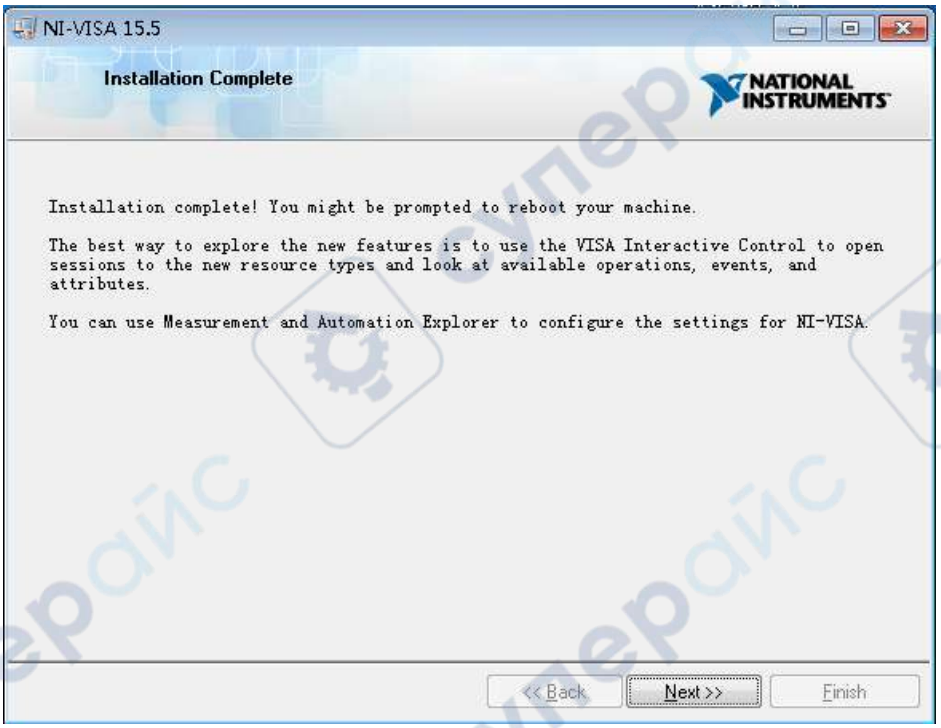
d. Set the install path. The default path is “C:\Program Files\National Instruments\”. You can change it. Click Next.



e. Click Next twice, in the License Agreement dialog, select “I accept the above 2 License Agreement(s).”,and click Next.



f. Click Next to begin the installation.



g. Wait until the installation is completed, and then reboot your PC.

1. SCPI command

1.1. Introduction to the SCPI Language

SCPI (Standard Commands for Programmable Instruments) is an ASCII-based programming language for test and measurement instruments. SCPI commands use a hierarchical structure known as a tree system. Associated commands are grouped under a common node or root, thus forming subsystems. A portion of the SENSe subsystem illustrates this, below.

SENSe:

VOLTage:

DC:RANGe {<range>|MIN|MAX|DEF}

DC:RANGe? [MINimum|MAXimum|DEFault]

SENSe is the root keyword of the command, **VOLTage** is a second-level keyword, and **DC** is a third-level keyword. A colon (**:**) separates consecutive keywords.

1.2. Syntax Conventions

The command syntax format is illustrated below:

[SENSe:]VOLTage:DC:RANGe {<range>|MIN|MAX|DEF}

Most commands (and some parameters) are a mixture of upper- and lower-case letters. The upper-case letters indicate the command's abbreviated spelling, which yields shorter program lines. For better program readability, use the long-form.

For example, consider the keyword VOLTage, above. You can type VOLT or VOLTage in any combination of upper- or lower-case letters. Therefore, VolTaGe, volt, and Volt are all acceptable. Other forms, such as VOL and VOLTAG, will generate an error.

- Braces (**{ }**) enclose the parameter choices. The braces are not sent with the command string.
- A vertical bar (**|**) separates parameter choices. For example, {<*range*>|MIN|MAX|DEF} in the above command indicates that you can specify a numeric range parameter, or "MIN", "MAX" or "DEF". The bar is not sent with the command string.

- Angle brackets (**< >**) indicate that you must specify a value for the enclosed parameter. For example, the above syntax statement shows the *<range>* parameter in angle brackets. Do not send the brackets with the command string. You must specify a value for the parameter (for example "VOLT:DC:RANG 10") unless you select one of the other options shown in the syntax (for example "VOLT:DC:RANG MIN").
- Optional parameters are enclosed in square brackets (**[ ]**). The brackets are not sent with the command string. If you do not specify a value for an optional parameter, the instrument uses a default value.

1.3. Command Separators

A colon (:) separates consecutive different levels of keywords.. You must insert a blank space to separate a parameter from a command keyword. If a command requires more than one parameter, separate adjacent parameters using a comma:

```
MEAS:TEMP? RTD,PT100
```

A semicolon (;) separates commands within the same subsystem and can also minimize typing. For example, the following string:

```
TRIG:COUN 2;SOUR EXT
```

Equivalent to the following two commands:

```
TRIG:COUN 2
TRIG:SOUR EXT
```

Use a semicolon and a colon to link commands from different subsystems. For example, in the following example, an error is generated if you do not use both the colon and semicolon:

```
TRIG:COUN 2;:SAMP:COUN 2
```

1.4. Using the MIN, MAX and DEF Parameters

For many commands, you can substitute "MIN" or "MAX" in place of a parameter. In some cases, you may also substitute "DEF". For example, consider the following example:

VOLTage:DC:RANGe {<range>|MIN|MAX|DEF}

Instead of selecting a specific value for the *<range>* parameter, you can substitute MIN to set the range to its minimum value, MAX to set the range to its maximum value, or DEF to set the range to its default value.

1.5. Querying Parameter Settings

You can query the current value of most parameters by adding a question mark (**?**) to the command. For example, The following example sets the trigger count to 10 measurements:

```
TRIG:COUN 10
```

You can then query the count value by sending:

```
TRIG:COUN?
```

You can also query the minimum or maximum count allowed as follows:

```
TRIG:COUN? MIN
TRIG:COUN? MAX
```

1.6. IEEE-488.2 Common Commands

The IEEE-488.2 standard defines a set of common commands that perform functions such as reset, self-test and status operations. Common commands always begin with an asterisk (*****), are three characters in length and may include one or more parameters. The command keyword is separated from the first parameter by a blank space. Use a semicolon (**;**) to separate multiple commands as shown below:

*RST : Restore instrument to factory default state.

*IDN? : Identification Query. Returns the instrument's identification string.

*CLS : Clear Status Command. Clears the event registers in all register groups. Also clears the error queue.

1.7. SCPI Parameter Types

The SCPI language defines several data formats to be used in program messages and response messages.

1.7.1 Numeric Parameters

Commands that require numeric parameters will accept all commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation. Special values for numeric parameters such as MIN,MAX, and DEF are also accepted. You can also send engineering unit suffixes with numeric parameters (e.g., M, k, m, or u). If a command accepts only certain specific values, the instrument will automatically round the input numeric parameters to the accepted values. The following command requires a numeric parameter for the range value:

VOLTage:DC:RANGe {<range>|MIN|MAX|DEF}

Note Because the SCPI parser is case-insensitive, there is some confusion over the letter "M"(or "m"). For your convenience, the instrument interprets "mV" (or "MV") as millivolts, but "MHZ" (or "mhz") as megahertz. Likewise "MΩ" (or "mΩ") is interpreted as megΩ. You can use the prefix "MA" for mega. For example, "MAV" is interpreted as megavolts.

1.7.2 Discrete Parameters

Discrete parameters are used to program settings that have a limited number of values (like IMMediate, EXTernal, or BUS). They have a short form and a long-form just like command keywords. You can mix upper- and lower-case letters. Query responses will always return the short form in all upper-case letters. The following example requires discrete parameters for the temperature units:

UNIT:TEMPerature {C|F|K}.

1.7.3 Boolean Parameters

Boolean parameters represent a single binary condition that is either true or false. For a false condition, the instrument will accept "OFF" or "0". For a true condition, the instrument will accept "ON" or "1". When you query a Boolean setting, the instrument will always return "0" or "1". The following example requires a Boolean parameter:

CALCulate:AVERage[:STATe] {ON|1|OFF|0}

1.8. ABORt

Aborts a measurement in progress, returning the instrument to the trigger idle state.

Parameter	Typical Return
(none)	(none)
Abort a measurement in progress: TRIG:SOUR IMM //Set the trigger source for immediate trigger TRIG:COUN 10 //Set the trigger source for 10 times INIT //Set the trigger state for “wait for trigger” ABOR //Interrupt the measurement	

- This command may be used to abort a measurement when the instrument is waiting for a trigger, or for aborting a long measurement or series of measurements.

1.9. FETCh?

Waits for measurements to complete and copies all available measurements to the instrument's output buffer. The readings remain in reading memory.

Parameter	Typical Return
(none)	(none)
Set the trigger source for an immediate trigger, the INIT command will instrument in "waiting for trigger" state, in the immediate trigger condition, measurements will immediately be triggered and the measurement results are sent to the measurement of memory. The FETCh? query transfers the measurement from reading memory to the instrument's output buffer. CONF:VOLT:DC TRIG:SOUR IMM //Set the trigger source for immediate trigger TRIG:COUN 3 //Set the trigger source for 3 times INIT //Set the trigger state for “et the trigger st FETC? //Read the resulting measurement value	

Typical Response: +1.50545692E+00,+1.50545984E+00,+1.50545984E+00

- The **FETch?** query does not erase measurements from the reading memory. You can send the query multiple times to retrieve the same data.
- You can store up to 1,0000 measurements in the reading memory of the SDM . If reading memory overflows, new measurements will overwrite the oldest measurements stored; the most recent measurements are always preserved. The instrument clears all measurements from reading memory when the measurement configuration changes, or when any of these commands are executed:

  INITiate

  MEASure:<function>?

  Read?

  *RST

  SYSTem:PRESet

1.10. INITiate[:IMMediate]

Set the trigger state for “wait for trigger”. Measurements will begin when the specified trigger conditions are satisfied following the receipt of the INITiate command. This command also clears the previous set of measurements from reading memory.

Parameter	**Typical Return**
(none)	(none)
Set the trigger source for “wait for trigger”, it can effectively receive the trigger signal: CONF:VOLT:DC TRIG:SOUR IMM // Set the trigger source to trigger IMM TRIG:COUN 5 // Set the trigger count for 5 times INIT // Set the trigger state for “wait for trigger” FETCh? // read the measured from reading memory	

Typical Response: +1.20302544E+00,+1.20302953E+00,+1.20302866E+00, +1.20302661E+00,+1.20303246E+00

- Storing measurements in reading memory with I**NITiate** is faster than sending measurements to the instrument's output buffer using **READ?** (provided you do not send **FETCh?** until done). The INITiate command is also an "overlapped" command. This means that after executing INITiate, you can send other commands that do not affect the measurements. Otherwise, if you send any other commands during Initiate's execution, the machine will pop up and tell you that the commands you sent are invalid.
- You can store up to 1,000 measurements in the reading memory of the SDM3055. If reading memory overflows, new measurements will overwrite the oldest measurements stored; the most recent measurements are always preserved. To retrieve the measurements from the reading memory, please use **FETCh?.** Use **DATA:REMove?** or **R?** to read and erase all or part of the available measurements.
- Use **ABORt** command to return to idle.

1.11. OUTPut:TRIGger:SLOPe {POSitive|NEGative}

OUTPut:TRIGger:SLOPe?

Selects the slope of the voltmeter complete output signal on the rear-panel VM Comp BNC connector.

Parameter	**Typical Return**
{POSitive|NEGative}, Default: NEGative	POS or NEG
Configure DC voltage measurements and make two measurements. The signal on the rear-panel VM Comp connector will output a positive pulse as each measurement is completed: CONF:VOLT:DC 10 SAMP:COUN 2 OUTP:TRIG:SLOP POS	

INIT

◆ This parameter is set to its default value after a Factory Reset.

1.12. R? [<max_readings>]

Reads and erases all measurements from the reading memory up to the specified <*max_readings*>. The measurements are read and erased from the reading memory starting with the oldest measurement first.

Parameter	**Typical Return**
1 to 10,000(readings) Default is all readings in memory	(none)
Read and remove the three oldest readings: TRIG:COUN 3 INIT R? 3 Typical Response : #247-1.06469770E-03,-1.08160033E-03,-1.22469433E-03 The "#2" means that the next 2 digits indicate how many characters will be in the returned memory string. These two digits are the "47" after the "#2". Therefore, the remainder of the string is 47 digits long: -1.06469770E-03,-1.08160033E-03,-1.22469433E-03	

The R? and DATA:REMove? queries allow you to periodically remove measurements from the reading memory that would normally cause the reading memory to overflow. R? does not wait for all readings to complete. It sends the readings that are complete at the time the instrument receives the command. Use Read? or Fetch? if you want the instrument to wait until all readings are complete before sending readings.

- You can store up to 1,0000 measurements in the reading memory of the SDM. If reading memory overflows, new measurements will overwrite the oldest measurements stored; the most recent measurements are always preserved. The instrument clears all measurements from reading memory when the measurement configuration changes, or when any of these commands are executed:

  INITiate

  MEASure:<function>?

  READ?

  *RST

  SYSTem:PRESet

1.13. READ?

Read and erase measurement results from reading storage ,up to the specified < max_readings >.This command reads the data will begin with the first measurement results.

Parameter	**Typical Return**
(none)	(none)
Transfer measurements from the reading memory: CONF:VOLT:DC TRIG:COUN 4 SAMP:COUN 1 READ? Typical Response:-1.23006735E-03,-1.30991641E-03,-1.32756530E-03, -1.32002814E-03	

- The FETch? query does not erase measurements from the reading memory. You can send the query multiple times to retrieve the same data.

- You can store up to 1,0000 measurements in the reading memory of the SDM. If reading memory overflows, new measurements will overwrite the oldest measurements stored; the most recent measurements are always preserved. The instrument clears all measurements from reading memory when the measurement configuration changes, or when any of these commands are executed:

  INITiate

  MEASure:<function>?

  READ?

  *RST

  SYSTem:PRESet

1.14. SAMPle:COUNt {<count>|MIN|MAX|DEF}

SAMPle:COUNt? [{MIN|MAX|DEF}]

Specifies the number of measurements (samples) the instrument will take per trigger.

Parameter	**Typical Return**
1 to 599999999, Default: 1	1
Configure DC voltage measurement, Set the sample count to 10,and trigger 2 times, then make and read the measurement : CONF:VOLT:DC SAMP:COUN 10 //Set the sample count to 10 TRIG:COUN2 //Set the time of trigger to 2 TRIG:SOUR IMM //Set the trigger source to IMM READ? //Start the wheel measurement and reading Typical Response：+1.20302544E+00,…（20 measurements）	

- You can use the specified sample count in conjunction with a trigger count (TRIGger:COUNt), which

sets the number of triggers to be accepted before returning to the "idle" trigger state. The total number of measurements returned will be the product of the sample count and trigger count.

- You can store up to 10,000 measurements in the reading memory of the SDM. If reading memory overflows, new measurements will overwrite the oldest measurements stored; the most recent measurements are always preserved.
- This parameter is set to its default value after a Factory Reset.

1.15. UNIT:TEMPerature {C|F|K}
UNIT:TEMPerature?

Selects the units (°C, °F, or Kelvin) to be used for all temperature measurements.

Parameter	**Typical Return**
{C|F|K}, Default: C	C, F, or K
Perform RTD measurement and return the measurement results in ℉: CONF:TEMP UNIT:TEMP F //Set the temperature measurement unit in °F MEAS:TEMP? RTD,PT100 // read a set of temperature measurements Typical Response: +9.90000000E+37	

- The command also accepts CEL or FAR, but the query returns C or F.
- This parameter is set to its default value after a Factory Reset.

2. CALCulate Subsystem Introduction

The CALCulate subsystem receives read-time data from the measurement hardware and sends them to reading memory. It also optionally performs the following math operations: Limit checking, Histogram, Statistics.

CALCulate subsystem

CALCulate:CLEar[:IMMediate]

CALCulate:LIMit Subsystem

CALCulate:TRANsform:HISTogram Subsystem

CALCulate:SCALe Subsystem

CALCulate:AVERage Subsystem

2.1. CALCulate:CLEar[:IMMediate]

Clears all limits, histogram data, statistics, and measurements.

Parameter	**Typical Return**
(none)	(none)
Clear all limits, histogram data, statistics, and measurements: CALC:CLE:IMM	

- The items cleared by this command are cleared synchronously, so that the histogram, statistics, and limit data all restart at the same time that measurements restart.

2.2. CALCulate:LIMit Subsystem

This subsystem specifies measurements and indicates when a limit has been exceeded.

Command Summary

CALCulate:LIMit:CLEar[:IMMediate]

CALCulate:LIMit:{LOWer|UPPer}[:DATA]

CALCulate:LIMit[:STATe]

Example

The following example enables limit testing of 100 DC voltage measurements and indicates whether measurements were outside the range of 2.4 to 3.6 V.

```
*CLS
CONF:VOLT:DC
SAMP:COUN 100
CALC:LIM:LOW 2.4
CALC:LIM:UPP 3.6
CALC:LIM:STAT ON
INIT
```

2.2.1 CALCulate:LIMit:CLEar[:IMMediate]

Clear all Limit test results (Low Failures, High Failures, and Status), but do not clear the setting conditions of the Low Limit and High Limit.

Parameter	Typical Return
(none)	(none)
Clear the limit test results: CALC:LIM:CLE	

- This command does not clear measurements in reading memory.

  The instrument clears front-panel indications of limits being exceeded

  CALCulate:LIMit:STATe ON

INITiate

MEASure:<function>?

READ?

CALCulate:LIMit:CLEar

*RST

- To clear statistics, limits, histogram data, and measurement data, use CALCulate:CLEar[:IMMediate].

2.2.2 CALCulate:LIMit:{LOWer|UPPer}[:DATA] {<value>|MIN|MAX|DEF}

CALCulate:LIMit:{LOWer|UPPer}[:DATA]? [{MIN|MAX|DEF}]

Sets an upper or lower limit.

Parameter	**Typical Return**
-1.0E+15 to -1.0E-15, or 0.0(default) or +1.0E-15 to 1.0E+15	+1.00000000E+00
See Example.	

- You can assign a lower limit, an upper limit, or both. Do not set the lower limit above the upper limit. If the limit set is higher than the upper limit, the limit value is set to the same limit will force the same value.
- The CONFigure command will reset the two limits to 0.
- This parameter is set to its default value after a Factory Reset.

2.2.3 CALCulate:LIMit[:STATe]{ON|1|OFF|0}

CALCulate:LIMit[:STATe]?

Enables or disables limit testing.

Parameter	Typical Return
{ON|1|OFF|0}, default: OFF	0 (OFF) or 1 (ON)
See Example.	

- The instrument clears front-panel indications of limits being exceeded and clears bits
- when the measurement function changes, or when any of the following commands are executed:

  CALCulate:LIMit:STATe ON

  INITiate

  MEASure:<function>?

  READ?

  CALCulate:LIMit:CLEar

  *RST

  SYSTem:PRESet
- This parameter is set to its default value after a Factory Reset.

2.3. CALCulate:TRANsform:HISTogram Subsystem

Configure the relevant parameters of histogram.

Command Summary

CALCulate:TRANsform:HISTogram:ALL?

CALCulate:TRANsform:HISTogram:DATA?

CALCulate:TRANsform:HISTogram:CLEar[:IMMediate]

CALCulate:TRANsform:HISTogram:COUNt?

CALCulate:TRANsform:HISTogram:POINts

CALCulate:TRANsform:HISTogram:RANGe:AUTO

CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer}

CALCulate:TRANsform:HISTogram[:STATe]

Example

This example enables an automatically scaled, 100-bin histogram of 1000 DCV measurements. It then returns the computed histogram, including the lower and upper range values, the total measurement count, and the bin data.

CONF:VOLT:DC 20

SAMP:COUN 1000

CALC:TRAN:HIST:RANG:AUTO ON

CALC:TRAN:HIST:POIN 100

CALC:TRAN:HIST:STAT ON

INIT

*WAI

CALC:TRAN:HIST:ALL?

Typical Response: +9.99383828E+00,+1.00513398E+01,+1000,<*102 bin counts*>

Note: The above response indicates 102 bin counts because the histogram includes bins for values below and above the histogram range.

2.3.1 CALCulate:TRANsform:HISTogram:ALL?

CALCulate:TRANsform:HISTogram:DATA?

The ALL form of the query returns a comma-separated list of the lower and upper range values, the number of measurements, and the bin data collected since the last time the histogram data was cleared. The DATA form returns only the bin data.

Parameter	Typical Return
(none)	(none).
See Example.	

- The bin data includes the following, in order:
  - The number of measurementsless than the lower range value.
  - The number of measurements in each of the bins, starting at the lower range value bin
  - The number of measurements greater than the upper range value
- Range values are real numbers returned in the form +1.00000000E+00. The number of measurements and bin data are signed, positive integers returned in the form +100.

2.3.2 CALCulate:TRANsform:HISTogram:CLEar[:IMMediate]

Clears the histogram data and restarts histogram ranging only if histogram automatic range selection is enabled (CALCulate:TRANsform:HISTogram:RANGe:AUTO ON).

Parameter	Typical Return
(none)	(none)
Clear the histogram data: CALC:TRAN:HIST:CLE	

- This command does not clear measurements in reading memory.

- To clear statistics, limits, histogram data, and measurement data, use CALCulate:CLEar[:IMMediate].
- The HISTogram subsystem configures the histogram display. The instrument clears histogram data when the measurement function changes and when any of the following commands is sent:

  CALCulate:TRANsform:HISTogram:CLEar[:IMMediate]

  CALCulate:TRANsform:HISTogram:POINts

  CALCulate:TRANsform:HISTogram:RANGe:AUTO

  CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer}

  CALCulate:TRANsform:HISTogram[:STATe]

  INITiate[:IMMediate]

  MEASure:<*function*>?

  READ?

  *RST

  SYSTem:PRESet

2.3.3 CALCulate:TRANsform:HISTogram:COUNt?

Returns the number of measurements collected since the last time the histogram was cleared.

Parameter	**Typical Return**
（none）	+96
Return the number of measurements used to compute the current histogram: CALC:TRAN:HIST:COUN?	

2.3.4 CALCulate:TRANsform:HISTogram:POINts{<value>|MIN|MAX|DEF}

CALCulate:TRANsform:HISTogram:POINts?[{MIN|MAX|DEF}]

Sets the number of bins between the lower and upper range values for the histogram. Two additional bins always exist one for measurements below the lower range and one for measurements above the upper range.

Parameter	Typical Return
{10|20|40|100|200|400|MIN|MAX|DEF},default 100	+100
See Example.	

- You can specify the lower and upper range values using CALCulate:TRANsform:HISTogram:RANGe: {LOWer|UPPer}. Lower and upper range values are computed automatically if CALCulate:TRANsform:HISTogram:RANGe:AUTO is ON.
- This parameter is set to its default value after a Factory Reset.

2.3.5 CALCulate:TRANsform:HISTogram:RANGe:AUTO{ON|1|OFF|0}

CALCulate:TRANsform:HISTogram:RANGe:AUTO?

Enables or disables automatic selection of the histogram's lower and upper range values.

Parameter	Typical Return
{ON|1|OFF|0}，Default: ON	0 (OFF) or 1 (ON)
See Example.	

- **ON:** the instrument sets the lower and upper range values automatically.
- **OFF:** the lower and upper range values are specified by CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer}.
- Setting the lower or upper range value (CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer) disables automatic selection of the lower and upper range values (CALCulate:TRANsform:HISTogram:RANGe:AUTO OFF).
- The instrument restarts automatic range value selection (if enabled) when INITiate, MEASure? or READ? is executed.

- This parameter is set to its default value after a Factory Reset.

2.3.6 CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer}{<value>|MIN|MAX|DEF}

CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer}?[{MIN|MAX|DEF}]

Sets the histogram's lower and upper range values. Setting the lower or upper range value (CALCulate:TRANsform:HISTogram:RANGe:{LOWer|UPPer}) disables automatic selection of the lower and upper range values (CALCulate:TRANsform:HISTogram:RANGe:AUTO OFF).

Parameter	Typical Return
-1.0E+15 to -1.0E-15, or 0.0 (default) or +1.0E-15 to 1.0E+15	+1.00000000E+06
See Example.	

- If automatic range value selection is enabled (CALCulate:TRANsform:HISTogram:RANGe:AUTO ON), the query returns the computed range value. If no histogram data exists, 9.91E37 (Not a Number) is returned.
- Lower and upper range values are computed automatically if CALCulate:TRANsform:HISTogram:RANGe:AUTO is ON.
- This parameter is set to its default value after a Factory Reset.

2.3.7 CALCulate:TRANsform:HISTogram[:STATe]{ON|1|OFF|0}

CALCulate:TRANsform:HISTogram[:STATe]?

Enables or disables histogram computation.

Parameter	Typical Return
{ON|1|OFF|0}, default OFF	0 (OFF) or 1 (ON)
See Example.	

◆ This parameter is set to its default value after a Factory Reset.

2.4. CALCulate:SCALe Subsystem

This subsystem controls ACV and DCV measurements math scaling.

Command Summary

CALCulate:SCALe:DB:REFerence

CALCulate:SCALe:DBM:REFerence

CALCulate:SCALe:FUNCtion

CALCulate:SCALe:REFerence:AUTO

CALCulate:SCALe[:STATe]

2.4.1 CALCulate:SCALe:DB:REFerence {<reference>|MIN|MAX|DEF}

CALCulate:SCALe:DB:REFerence? [{MIN|MAX}]

Stores a relative value in the multimeter's dB Relative Register, which is used for the dB function in CALCulate:SCALe:FUNCtion. When the dB function is enabled, this value will be subtracted from each voltage measurement after the measurement is converted to dBm.

Note: This command applies only to ACV and DCV measurement functions.

Parameter	Typical Return
-200.0 dBm to +200.0 dBm, Default :0	+5.00000000E+02
Enable dB scaling with a -10 dB reference and a 300Ω reference resistance: CALC:SCAL:DBM:REF 300 CALC:SCAL:DB:REF -10.0 CALC:SCAL:FUNC DB CALC:SCAL:STAT ON	

- Specifying a reference value disables automatic reference selection (CALCulate:SCALe:REFerence:AUTO OFF).
- The dB relative value parameter is relative to the dBm reference set with CALCulate:SCALe:DBM:REFerence.
- The instrument sets the reference value to 0.0 with automatic reference selection enabled after a Factory Reset, a change in math function, or a change in measurement function.

2.4.2 CALCulate:SCALe:DBM:REFerence {<reference>|MIN|MAX|DEF}

CALCulate:SCALe:DBM:REFerence? [{MIN|MAX}]

Selects the reference resistance for converting voltage measurements to dBm. This reference value affects the dBm and dB scaling functions.

Note: This command applies only to ACV and DCV measurement functions.

Parameter	Typical Return
{50|75|93|110|124|125|135|150|250|300|500|600|800|900|1000| 1200| 8000Ω}. Default:600	+6.00000000E+02
Enable dBm scaling with a reference resistance of 600 Ω: CALC:SCAL:DBM:REF 600 CALC:SCAL:FUNC DBM CALC:SCAL:STAT ON	

- The instrument sets the reference value to its default value after a Factory Reset, or a change in measurement function.

2.4.3 CALCulate:SCALe:FUNCtion {DB|DBM}
CALCulate:SCALe:FUNCtion?

Selects the operation that will be performed by the scaling function:

Note: This command applies only to ACV and DCV measurements.

Parameter	**Typical Return**
{DB|DBM}	DB or DBM
Enable the DB scaling function referenced to the next measurement taken: CALC:SCAL:FUNC DBM CALC:SCAL:STAT ON	

- DB performs a relative dB computation. The result will be the difference between the input signal and the stored DB relative value (CALCulate:SCALe:DB:REFerence), with both values converted to dBm (dB = measurement in dBm – relative value in dBm).
  For the dB function, the reference value can be automatically selected using the first measurement converted to dBm as the reference value (CALCulate:SCALe:REFerence:AUTO), or it can be specified by CALCulate:SCALe:DB:REFerence.
- DBM performs a dBM computation. The result is logarithmic and is based on a calculation of power delivered to a reference resistance

(CALCulate:SCALe:DBM:REFerence), relative to 1 milliwatt. (dBm = 10 × log10 (measurement 2 / reference resistance / 1 mW)).

- Scaling function results must be in the range of -1.0E+24 to -1.0E-24, or +1.0E-24 to 1.0E+24. Results outside these limits will be replaced with -9.9E37 (negative infinity), 0, or 9.9E37 (positive infinity).
- This parameter is set to its default value after a Factory Reset.
- Scaling is set to OFF when you change measurement functions (for example, changing from DCV to ACV). You must re-enable scaling after changing measurement functions.

2.4.4 CALCulate:SCALe:REFerence:AUTO {ON|1|OFF|0}

CALCulate:SCALe:REFerence:AUTO?

Enables or disables automatic reference selection for the dB scaling functions

Note: This command applies only to ACV and DCV measurement functions.

Parameter	**Typical Return**
{ON|1|OFF|0}, default ON	0(OFF)or1(ON)
Enable the DB function with automatic reference selection and use the first measurement as the reference value: CALC:SCAL:DBM:REF 50 CALC:SCAL:FUNC DB CALC:SCAL:REF:AUTO ON CALC:SCAL:STAT ON READ?	

- **ON:** the first measurement made will be used as the reference for all subsequent measurements, and

automatic reference selection will be disabled:

- For the dB scaling function, the first measurement is converted to dBm, and CALCulate:SCALe:DB:REFerence is set to the result.

- **OFF:**CALCulate:SCALe:DB:REFerence specifies the reference for DB scaling.
- The instrument enables automatic reference selection when the scaling function is enabled (CALCulate:SCALe:STATe ON).
- This parameter is set to its default value after a Factory Reset or a change in measurement function

2.4.5 CALCulate:SCALe[:STATe] {ON|1|OFF|0}

CALCulate:SCALe[:STATe]?

Enables or disables the scaling function.

Note: This command applies only to ACV and DCV measurement functions.

Parameter	Typical Return
{ON|1|OFF|0}，Default :ON	0 (OFF) or 1 (ON)
Enable the DB function with automatic reference selection and use the first measurement as the reference value: CALC:SCAL:DBM:REF 50 CALC:SCAL:FUNC DB CALC:SCAL:REF:AUTO ON CALC:SCAL:STAT ON READ?	

- Enabling the scaling function also enables automatic null value selection (CALCulate:SCALe:REFerence:AUTO).

- Scaling is set to OFF when you change measurement functions (for example, changing from DCV to ACV). You must re-enable scaling after changing measurement functions.

2.5. CALCulate:AVERage Subsystem

This subsystem calculates measurement statistics.

Command Summary

CALCulate:AVERage[:STATe]

CALCulate:AVERage:CLEar[:IMMediate]

CALCulate:AVERage:ALL?

CALCulate:AVERage:AVERage?

CALCulate:AVERage:COUNt?

CALCulate:AVERage:MAXimum?

CALCulate:AVERage:MINimum?

CALCulate:AVERage:PTPeak?

CALCulate:AVERage:SDEViation?

2.5.1 CALCulate:AVERage[:STATe]{ON|1|OFF|0}

CALCulate:AVERage[:STATe]?

Enables or disables statistics computation.

Parameter	**Typical Return**
{ON|1|OFF|0}, Default: OFF	0 (OFF) or 1 (ON)
Return the statistics of 100 frequency measurements: CONF:FREQ	

```
SAMP:COUN 100
CALC:AVER:STAT ON
INIT
CALC:AVER:ALL?
Typical Response: -4.10466677E-04,+3.13684184E-04,+1.75743178E-02,
-6.74799085E-04
```

- Statistics are cleared when the measurement function changes or when any of these commands are executed:

  CALCulate:AVERage:STATe ON

  CALCulate:AVERage:CLEar

  INITiate

  MEASure:<*function*>?

  READ?

- The instrument turns this setting OFF when the measurement function is changed or after a Factory Reset.

2.5.2 CALCulate:AVERage:ALL?

CALCulate:AVERage:AVERage?

CALCulate:AVERage:COUNt?

CALCulate:AVERage:MAXimum?

CALCulate:AVERage:MINimum?

CALCulate:AVERage:PTPeak?

CALCulate:AVERage:SDEViation?

The CALCulate:AVERage:ALL? query returns the arithmetic mean (average), standard deviation, minimum value, and maximum value of all measurements taken since the statistics were last cleared. The count and

peak-to-peak statistics are not returned by CALCulate:AVERage:ALL?.

The other six queries listed above return individual values.

Parameter	**Typical Return**
(none)	(see below)
Return the statistics of 100 frequency measurements: CONF:FREQ SAMP:COUN 100 CALC:AVER:STAT ON INIT CALC:AVER:ALL? Typical Response: -4.10466677E-04,+3.13684184E-04, +1.75743178E-02, -6.74799085E-04	

- Statistics are cleared when the measurement function changes or when any of these commands are executed:

  CALCulate:AVERage:STATe ON

  CALCulate:AVERage:CLEar

  INITiate

  MEASure:<*function*>?

  READ?

  *RST

- All values except the COUNt are returned in the form +1.00000000E+01. The COUNt query return is as a signed, positive integer: +129.
- When dB or dBm scaling is used, the CALC:AVER:AVER and CALC:AVER:SDEV queries return +9.91000000E+37 (not a number).

2.5.3 CALCulate:AVERage:CLEar[:IMMediate]

Clears all computed statistics: minimum, maximum, average, peak-to-peak, count and standard deviation.

Parameter	Typical Return
(none)	(none)
Clear the stored statistical data: CALC:AVER:CLE	

- This command does not clear measurements in reading memory.
- Statistics are cleared when the measurement function changes or when any of these commands are executed:

  CALCulate:AVERage:STATe ON

  CALCulate:AVERage:CLEar

  INITiate

  MEASure:<*function*>?

  READ?

  *RST
- To clear statistics, limits, histogram data, and measurement data, use CALCulate:CLEar[:IMMediate].

3. CONFigure Subsystem

The CONFigure commands are the most concise way to configure measurements. These commands use default measurement configuration values. However, these commands do not automatically start measurements, so you can modify measurement attributes before initiating the measurement.

Note:Use INITiate or READ? to initiate measurements.

Command Summary

CONFigure?

CONFigure:CONTinuity

CONFigure:CURRent:{AC|DC}

CONFigure:DIODe

CONFigure:{FREQuency|PERiod}

CONFigure:{RESistance|FRESistance}

CONFigure:TEMPerature

CONFigure[:VOLTage]:{AC|DC}

CONFigure:CAPacitance

Default Settings for the CONFigure Command

The CONFigure commands select the function, range, and resolution (only applies to the SDM3065X) in one command. All other parameters are set to their default values (below). If no range or resolution settings, which will use the default value.

Measurement Parameter	Default Setting
Autozero	OFF (only apply to SDM3065X)
Range	AUTO
Samples per Trigger	1 sample
Trigger Count	1 trigger

Trigger Delay	Automatic delay
Trigger Source	Immediate
Trigger Slope	NEGative
Math Functions	Disabled
Per-function Null State	Disabled
AC Input Filter(bandwidth)	20Hz (only apply to SDM3065X)
DCV/DCI Filter	OFF (apply to SDM3045X,SDM3055,)

Using CONFigure

The following example uses CONFigure and READ? to make an externally-triggered measurement. The CONFigure command configures DC voltage measurements but does not place the instrument in the "wait-for-trigger" state.

The READ? query places the instrument in the "wait-for-trigger" state, initiates a measurement when the rear-panel **Ext Trig** input is pulsed (low by default), stores the measurement in reading memory, and transfers the measurement to the instrument's output buffer. The default range (auto-range) and resolution 10 PLC(**PLCs only apply to SDM3065X models**) are used for the measurement.

```
CONF:VOLT:DC
TRIG:SOUR EXT
READ?
```

Typical Response: -5.21391630E-04

The following example is like the previous one, but it uses INITiate and FETCh? instead of READ?. The INITiate command places the instrument in the "wait-for-trigger" state, triggers a measurement when the rear-panel **Ext Trig** input is pulsed (low by default), and sends the measurement to reading memory. The FETCh? query transfers the measurement from reading memory to the instrument's output buffer.

```
CONF:VOLT:DC
TRIG:SOUR EXT
```

```
INIT
FETC?
```

Typical Response: -5.21205366E-04

Storing measurements in reading memory with INITiate is faster than sending measurements to the instrument's output buffer using READ? (provided you do not send FETCh? until done). The INITiate command is also an "overlapped" command. This means that after executing INITiate, you can send other commands that do not affect the measurements. This allows you to check for data availability before initiating a read attempt that might otherwise time out. Note that the FETCh? query waits until all measurements are complete to terminate. You can store up to 1,0000 measurements in the reading memory of the SDM.

The following example configures the instrument for 2-wire resistance measurements, triggers the instrument to make one measurement using INITiate, and stores the measurement in reading memory. The 20 kΩ range is selected.

```
CONF:RES 20000
INIT
FETC?
```

Typical Response: +5.21209585E+04

3.1. CONFigure?

Returns a quoted string indicating the present function, range. When the model is SDM3065X, the corresponding resolution is returned. The present function name is returned in the short format, such as VOLT, FREQ.

Parameter	**Typical Return**

(none)	“none)l Returnat,is al
For the SDM3065X models, return the present function, range, and resolution: CONF? Typical Response："VOLT +2.00000000E-01,+1.00000000E-07" For the SDM3055 models, Return the present function and range: CONF? Typical Response："VOLT +2.00000000E-01"	

3.2. CONFigure:CONTinuity

Sets all measurement parameters and trigger parameters to their default values for continuity measurements.

Parameter	**Typical Return**
(none)	(none)
Configure the instrument for continuity measurements. and read the measurement: CONF:CONT READ? Typical Response：+9.90000000E+37 (display open)	

- The range and resolution are fixed at 2 kΩ for continuity tests (a 2-wire resistance measurement).
- The instrument beeps (if the beeper is enabled) for each measurement less than or equal to the continuity threshold (The threshold can be set by [:SENSe]:CONTinuity:THReshold:VALue command), and the actual resistance measurement appears on the display.
- From threshold to 2kΩ, the instrument displays the actual resistance measurement with no beep.

Above 2kΩ, the instrument displays "OPEN" with no beep.

- The FETCh?, READ?, and MEASure:CONTinuity? queries return the measured resistance, When the screen display “open”, return “+9.90000000E+37”.
- Use READ? Or INITiate to start the measurement.

3.3. CONFigure:CURRent:{AC|DC} [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values for AC or DC current measurements. Also specifies the range through the incoming parameters.

Parameter	**Typical Return**
SDM3045X: <range>：{600uA|6mA|60mA|600mA|6A|10A|AUTO}，Default: AUTO The 600uA and 6mA ranges can only be set in DC mode, and the minimum range is 60mA in AC mode SDM3055: <range>：{200uA|2mA|20mA|200mA|2A|10A|AUTO}，Default: AUTO The 200uA and 2mA ranges can only be set in DC mode, and the minimum range is 20mA in AC mode SDM3065X: <range>：{200uA|2mA|20mA|200mA|2A|10A|AUTO}，Default: AUTO	(none)
Take SDM3065X as an example, Configure AC current measurements using the 2A range. And read two measurements: CONF:CURR:AC 2 SAMP:COUN 2 READ? Typical Response：+4.32133675E-04,+4.18424606E-04	

- You can let auto-ranging select the measurement range, or you can manually select a fixed range. Autoranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- Use READ? or INITiate to start the measurement.

3.4. CONFigure:DIODe

Sets all measurement parameters and trigger parameters to their default values for diode measurements.

Parameter	Typical Return
（none）	（none）
Configure diode measurement, and read the measurement: CONF:DIOD READ? Typical Response： +9.90000000E+37(display open)	

- The range is fixed for diode tests is 2 VDC.
- The FETCh?, READ?, and MEASure:DIODe? queries return the measured voltage, When the screen display “open”, return “+9.90000000E+37”.
- If the voltage value is less than the threshold, the instrument sounds a buzzer (unless the buzzer is disabled) and the instrument displays the actual voltage measurement value. If the voltage is above

the threshold, the instrument indicates "OPEN"(overload), no buzzing.

- Use READ? or INITiate to start the measurement.

3.5. CONFigure:{FREQuency|PERiod}

Sets all measurement parameters and trigger parameters to their default values in the frequency/period measurements.

Parameter	**Typical Return**
(none)	(none)
Configure frequency measurement and read the measurement: CONF:FREQ READ? Typical Response: +7.79645018E+01	

- The input signal for frequency or period measurements has an AC voltage component. Use [SENSe:]{FREQuency|PERiod}:VOLTage:RANGe:AUTO to disable or enable voltage auto-ranging or use CONFigure:{FREQuency|PERiod} to select voltage auto-ranging by default. Use [SENSe:]{FREQuency|PERiod}:VOLTage:RANGe to select a fixed voltage range for frequency and period measurements.
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- Use READ? or INITiate to start the measurement.

3.6. CONFigure:{RESistance|FRESistance} [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values for 4-wire (FRESistance) or 2-wire (RESistance) resistance measurements. Also specifies the range and resolution.

Parameter	Typical Return
SDM3045X: <range>: {600 Ω|6 kΩ|60 kΩ|600 kΩ|6 MΩ|60 MΩ|100 MΩ}, Default: AUTO SDM3055/: <range>: {200 Ω|2 kΩ|20 kΩ|200 kΩ|2 MΩ|10 MΩ|100 MΩ}, Default: AUTO SDM3065X: <range>: {200 Ω|2 kΩ|20 kΩ|200 kΩ|1 MΩ|10 MΩ|100 MΩ}, Default: AUTO	(none)
Take SDM3065X as an example, Configure 4-wire resistance measurements using the 200 Ω range with default resolution. Make and read two measurements CONF:FRES 200 SAMP:COUN 2 READ? Typical Response: +6.71881065E+01,+6.83543086E+01	

- You can let auto-ranging select the measurement range, or you can manually select a fixed range. Autoranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- Use READ? or INITiate to start the measurement.

3.7. CONFigure:TEMPerature [{RTD|THER|DEFault}[,{<type>|DEFault}]]

Sets all measurement parameters and trigger parameters to their default values for temperature measurements.

Parameter	Typical Return
<probe_type>：{RTD|THER}，default THER. The default command can only choose the built-in sensor manufacturers	none
RTD<type>： { PT100|PT1000} THER<type>： {BITS90|EITS90|JITS90|KITS90|NITS90|RITS90|SITS90|TITS90}	
Configuration RTD measurement. Then read measurement results: CONF:TEMP RTD,PT100 READ? Typical Response： -2.00000000E+02	

- To change temperature units, use UNIT:TEMPerature.
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "9.9E37" from the remote interface.
- Use READ? or INITiate to start the measurement.

3.8. CONFigure[:VOLTage]:{AC|DC} [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values for AC or DC voltage measurements. Also specifies the range through the incoming parameters.

Parameter	Typical Return
SDM3045X: <range>: {600 mV|6 V|60 V|600 V|1000 V(DC)/750V(AC)}, Default: AUTO SDM3055//SDM3065X: <range>: {200 mV|2 V|20 V|200 V|1000 V(DC)/750V(AC)}, Default: AUTO	(none)
Take SDM3065X as an example, Configure AC voltage measurements using the 200 V range. Make and read two measurements: CONF:VOLT:AC 200 SAMP:COUN 2 READ? Typical Response: +2.43186951E-02,+2.56896019E-02	

- You can let auto-ranging select the measurement range, or you can manually select a fixed range. Autoranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "9.9E37" from the remote interface.
- Use READ? or INITiate to start the measurement.

3.9. CONFigure:CAPacitance [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values for capacitance measurement. Also specifies the range through the incoming parameters.

Parameter	Typical Return

SDM3045X/ SDM3055/: <range>：{2nF|20nF|200nF|2uF |20uF| 200uF|10000uF}，Default AUTO SDM3065X： <range>:{2nF|20nF|200nF|2uF |20uF| 200uF|2mF|20mF|100mF}，Default AUTO	（none）
Configure capacitance measurement using the 2uF range. Make and read two measurements: CONF:CAP 2uF SAMP:COUN 2 READ? Typical Response： +7.26141264E-10,+7.26109188E-10	

- You can let auto-ranging select the measurement range, or you can manually select a fixed range. Autoranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "9.9E37" from the remote interface.
- Use READ? or INITiate to start the measurement.

4. DATA Subsystem

This subsystem allows you to configure and remove data from reading memory. The instrument clears all measurements from reading memory when the measurement configuration changes, or when any of these commands are executed: INITiate, MEASure:<*function*>?, READ?, *RST.

Command Summary

DATA:LAST?

DATA:POINts?

DATA:REMove?

4.1. DATA:LAST?

Returns the last measurement taken. You can execute this query at any time, even during a series of measurements.

Parameter	**Typical Return**
(none)	(none)
Return the last measurement: DATA:LAST? Typical Response: One measurement with units. If no data is available, 9.91E37 (Not a Number) is returned with units For Example -4.79221344E-04 VDC	

4.2. DATA:POINts?

Returns the total number of measurements currently in reading memory. You can execute this query at any time, even during a series of measurements.

Parameter	**Typical Return**
（none）	(none)
Return the number of measurements in reading memory: DATA:POIN? Typical Response：+20	

- You can store up to 1,0000 measurements in the reading memory on the SDM

4.3. DATA:REMove? <num_readings>

Reads and erases <*num_readings*> measurements from the reading memory. If fewer than <*num_readings*> measurements are available, the query will return an error.

Parameter	**Typical Return**
1~10000	none
Read and erase the five oldest readings from reading memory:： DATA:REMove? 5	

Typical Response：-4.55379486E-04,-4.55975533E-04,-4.56273556E-04,
-4.53591347E-04,-4.55379486E-04

- The R? and DATA:REMove? queries allow you to periodically remove measurements from the reading memory that would normally cause the reading memory to overflow.

5. MEASure Subsystem

The MEASure queries are the easiest way to program measurements because they always use default measurement parameters. You set the function, range in one command, but you cannot change other parameters from their default values. The results are sent directly to the instrument's output buffer.

Note: A MEASure query is functionally equivalent to sending CONFigure followed immediately by READ? The difference is that CONFigure commands allow you to change parameters between the CONFigure and the READ?

Command Summary

MEASure:CONTinuity?

MEASure:CURRent:{AC|DC}?

MEASure:DIODe?

MEASure:{FREQuency|PERiod}?

MEASure:{RESistance|FRESistance}?

MEASure:TEMPerature?

MEASure[:VOLTage]:{AC|DC}?

MEASure:CAPacitance?

Default Settings for MEASure?

With the MEASure? queries, you can select the function, range, and resolution(only applicable to SDM3065X models) in one command. All other parameters are set to their default values (below).

Parameter	**Default Setting**
Autozero	OFF
Range	AUTO

Samples per Trigger	1 sample
Trigger Count	1 trigger
Trigger Delay	Automatic delay
Trigger Source	Immediate
Trigger Slope	NEGative
Math Functions	Disabled
Per-function Null State	Disabled
AC Input Filter (bandwidth)	20 Hz (SDM3065X)
DC Filter	Disabled (SDM3045|SDM3055)

Using the MEASure? Query

The following example configures DC voltage measurements, internally triggers the instrument to take a measurement, and reads the measurement. The default range (auto-range) is used for the measurement.

```
MEAS:VOLT:DC?
```

Typical Response: +4.23450000E-03

The following takes the SDM3065X model as an example, configures the instrument for 2-wire resistance measurements, triggers the instrument to take a measurement, and reads the measurement with the 2kΩ range.

```
MEAS:RES? 2000
```

Typical Response: +3.27150000E+02

5.1. MEASure:CONTinuity?

Sets all measurement parameters and trigger parameters to their default values for continuity test and immediately triggers a measurement. The results are sent directly to the instrument's output buffer.

Parameter	Typical Return
（none）	none
Configure the instrument for continuity measurements. Then make and read one measurement: MEAS:CONT? Typical Response:+9.84739065E+02	

- The range and resolution are fixed at 2kΩ for continuity tests (a 2-wire resistance measurement).
- The instrument beeps (if the beeper is enabled) for each measurement less than or equal to the continuity threshold, and the actual resistance measurement appears on the display.
- From threshold to 2 kΩ, the instrument displays the actual resistance measurement with no beep. Above 2 kΩ, the instrument displays "OPEN" with no beep.
- The FETCh?, READ?, and MEASure:CONTinuity? queries return the measured resistance, When the screen display “open”, return “+9.90000000E+37”.

5.2. MEASure:CURRent:{AC|DC}? [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values for AC or DC current measurements and immediately triggers a measurement. Also specifies the range through the incoming parameters.

Parameter	Typical Return
SDM3045X: <range>：{600uA|6mA|60mA|600mA|6A|10A|AUTO}，Default: AUTO	None

The 600uA and 6mA ranges can only be set in DC mode, and the minimum range is 60mA in AC mode SDM3055: <range>：{200uA|2mA|20mA|200mA|2A|10A|AUTO}，Default: AUTO The 200uA and 2mA ranges can only be set in DC mode, and the minimum range is 20mA in AC mode SDM3065X: <range>：{200uA|2mA|20mA|200mA|2A|10A|AUTO}，Default: AUTO	
Take the SDM3065X model as an example: configure AC current measurement using the 2A range. Make and read two measurements: MEAS:CURR:AC? 2 Typical Response： +4.32133675E-04	

- You can let auto-ranging select the measurement range, or you can manually select a fixed range. Auto-ranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "+9.90000000E+37" from the remote interface.

5.3. MEASure:DIODe?

Sets all measurement parameters and trigger parameters to their default values for diode test measurements and immediately triggers a measurement. The results are sent directly to the instrument's output buffer.

Parameter	Typical Return
（none）	none
Configure and read a default diode measurement: MEAS:DIOD? Typical Response：+9.84733701E-01	

- The range and resolution are fixed for diode tests: the range is 2 VDC.
- The FETCh?, READ?, and MEASure:DIODe? queries return the measured voltage, When the screen display “open”, return “+9.90000000E+37”.

5.4. MEASure:{FREQuency|PERiod}?

Sets all measurement parameters and trigger parameters to their default values for frequency or period measurements and immediately triggers a measurement. The results are sent directly to the instrument's output buffer.

Parameter	Typical Return
（none）	none
configure and read the default frequency measurements： MEAS:FREQ? Typical Response：+7.19480528E+01	

- The input signal for frequency or period measurements has an AC voltage component. By default, this command uses auto-range to select the voltage range.

5.5. MEASure:{RESistance|FRESistance}? [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement and trigger parameters to their default values for 4-wire resistance (FRESistance) or 2-wire (RESistance) measurements, and immediately triggers a measurement. The results are sent directly to the instrument's output buffer. Also specifies the range through the incoming parameters.

Parameter	Typical Return
SDM3045X: <range>: {600 Ω|6 kΩ|60 kΩ|600 kΩ|6 MΩ|60 MΩ|100 MΩ}, Default: AUTO SDM3055/: <range>: {200 Ω|2 kΩ|20 kΩ|200 kΩ|2 MΩ|10 MΩ|100 MΩ}, Default: AUTO SDM3065X: <range>: {200 Ω|2 kΩ|20 kΩ|200 kΩ|1 MΩ|10 MΩ|100 MΩ}, Default: AUTO	(none)
Take the SDM3065X model as an example: Configure 4-wire resistance measurements using the 200Ω range. Then make and read one measurement: MEAS:FRES? 200 Typical Response: +6.71881065E+01	

- You can let -auto-ranging select the measurement range, or you can manually select a fixed range. Auto-ranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.

5.6. MEASure:TEMPerature?[{RTD|THER|DEFault}[,{<type>|DEFault}]]

Sets all measurement parameters and trigger parameters to their default values and immediately triggers a measurement. The results are sent directly to the instrument's output buffer.

Parameter	**Typical Return**
<probe_type>: {RTD|THER}，default THER. The default command can only choose the built-in sensor manufacturers.	（none）
For RTD: <type>:{ PT100|PT1000} For THER: <type>:{BITS90|EITS90|JITS90|KITS90|NITS90|RITS90|SITS90|TITS90}	
Configure the RTD measurement. Then read out the measuring results: MEAS:TEMP? RTD,PT100 Typical Response： -2.00000000E+02	

- To change temperature units, use UNIT:TEMPerature.
- If the input signal is greater than can be measured, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.

5.7. MEASure[:VOLTage]:{AC|DC}? [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values and immediately triggers a measurement. The results are sent directly to the instrument's output buffer. Also specifies the range through the incoming parameters.

Parameter	Typical Return
SDM3045X: <range>：{600 mV|6 V|60 V|600 V|1000 V(DC)/750V(AC)}，Default :AUTO SDM3055//SDM3065X： <range>：{200 mV|2 V|20 V|200 V|1000 V(DC)/750V(AC)}，Default :AUTO	（none）
Take the SDM3065X model as an example: Configure AC voltage measurements using the 200 V range. Then make and read one measurement： MEAS:VOLT:AC? 200 Typical Response: +2.43186951E-02	

- You can let auto-ranging select the measurement range, or you can manually select a fixed range. Auto-ranging conveniently selects the range for each measurement based on the input signal. For the fastest measurements, use manual ranging (auto-ranging may require additional time for range selection).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.

5.8. MEASure:CAPacitance? [{<range>|AUTO|MIN|MAX|DEF}]

Sets all measurement parameters and trigger parameters to their default values for capacitance measurement. Also specifies the range through the incoming parameters

.

Parameter	Typical Return
SDM3045X/ SDM3055/: <range>：{2nF|20nF|200nF|2uF |20uF| 200uF|10000uF}，Default: AUTO SDM3065X： <range>：{2nF|20nF|200nF|2uF |20uF| 200uF|2mF|20mF|100mF}，Default: AUTO	（none）
Configure DC voltage ratio measurements using the 2uF resolution. Then make and read one measurement: MEAS:CAP? 2uF Typical Response： +7.26141264E-10	

- For the Input terminals, you can allow the instrument to select the measurement range by auto-ranging or you can select a fixed range using manual ranging. Auto-ranging decides which range to use for each measurement based on the input signal. For faster measurements, use manual ranging on each measurement (additional time is required for auto-ranging to select a range).
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.

6. SENSe Subsystem Introduction

The SENSe subsystem configures measurements. The most basic SENSe command is [SENSe:]FUNCtion[:ON], which selects the measurement function. All other SENSe commands are associated with specific measurement types:

Current

Frequency and Period

Resistance

Temperature

Voltage

Capacitance

6.1. [SENSe:]FUNCtion[:ON] "*<function>*"

[SENSe:]FUNCtion[:ON]?

Selects the measurement function (all function-related measurement attributes are retained).

Parameter	Typical Return
CONTinuity CURRent:AC CURRent[:DC] DIODe FREQuency FRESistance PERiod RESistance TEMPerature	The short form of the selected function is returned in quotation marks, with no optional keywords: "CONT"、"CURR:AC"、"CURR"、"DIOD" and so on.

VOLTage:AC VOLTage[:DC] CAPacitance default is VOLTage [: DC].	
Select the AC voltage function: FUNC "VOLT:AC"	

- If you change the measurement function, all measurement attributes of the previous function (range, resolution (SDM3065X), speed (SDM3045X/SDM3055)) are remembered. If you return to the original function, those measurement attributes will be restored.
- This parameter is set to its default value after a Factory Reset.

6.2. [SENSe:]CURRent Subsystem

This subsystem configures AC and DC current measurements.

Command Summary

[SENSe:]CURRent:{AC|DC}:NULL[:STATe]

[SENSe:]CURRent:{AC|DC}:NULL:VALue

[SENSe:]CURRent:{AC|DC}:NULL:VALue:AUTO

[SENSe:]CURRent:{AC|DC}:RANGe

[SENSe:]CURRent:{AC|DC}:RANGe:AUTO

[SENSe:]CURRent[:DC]:NPLC

[SENSe:]CURRent[:AC]: BANDwidth

[SENSe:]CURRent[:DC]: AZ[:STATe]

[SENSe]:CURRent[:DC]:FILTer[:STATe]

6.2.1 [SENSe:]CURRent:{AC|DC}:NULL[:STATe] {ON|1|OFF|0}

[SENSe:]CURRent:{AC|DC}:NULL[:STATe]?

Enables or disables the null function for AC or DC current measurements.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default OFF	0 (OFF) or 1 (ON)
Take SDM3065X as an example, Configure AC current measurements, Provide 1.5A AC current using the null function to subtract 1A from the measurements. Then make two measurements and send them to the instrument's output buffer: CONF:CURR:AC CURR:AC:NULL:STAT ON;VAL 1A SAMP:COUN 2 READ? Typical Response： +4.39291265E-01,+4.96965953E-01	

- Enabling the scaling function also enables automatic null value selection ([SENSe:]CURRent: {AC|DC}:NULL:VALue:AUTO ON).
- To set a fixed null value, use this command: [SENSe:]CURRent:{AC|DC}:NULL:VALue.
- The instrument disables the null function after a Factory Reset or CONFigure function..

6.2.2 [SENSe:]CURRent:{AC|DC}:NULL:VALue {<value>|MIN|MAX|DEF}

[SENSe:]CURRent:{AC|DC}:NULL:VALue? [{MIN|MAX|DEF}]

Sets the null value for AC or DC current measurements.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
-11 to 11 A, default 0	+0.00000000E+00
Take SDM3065X as an example, set a null value to measure: Configure AC current measurements, Provide 1.5A AC current, using the null function to subtract 0.5A from the measurements. Then make two measurements and send them to the instrument's output buffer. CONF:CURR:AC CURR:AC:NULL:STAT ON; CURR:AC:NULL:VAL 0.5A SAMP:COUN 2 READ? Typical Response： +9.98962286E-01,+9.98959327E-01	

- Specifying a null value disables automatic null value selection ([SENSe:]CURRent:{AC|DC}:NULL:VALue:AUTO OFF).
- To use the null value, the null state must be on ([SENSe:]CURRent:{AC|DC}:NULL:STATe ON).

◆ This parameter is set to its default value after a Factory Reset or CONFigure function.

6.2.3 [SENSe:]CURRent:{AC|DC}:NULL:VALue:AUTO{ON|1|OFF|0}

[SENSe:]CURRent:{AC|DC}:NULL:VALue:AUTO?

Enables or disables automatic null value selection for AC or DC current measurements.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default ON	0 (OFF) or 1 (ON)
Take SDM3065X as an example, set a null value to measure: Provide 1.5A DC current and Configure DC current measurements, using the null function to subtract 1A from the measurements. Then make two measurements and send them to the instrument's output buffer. CONF:CURR:DC CURR:DC:NULL:STAT ON CURR:DC:NULL:VAL 1 SAMP:COUN 2 READ? Typical Response：+5.00043460E-01,+5.00041377E-01 Using automatic null value selection to measure: CONF:CURR:DC CURR:DC:NULL:STAT ON CURR:DC:NULL:VAL:AUTO ON	

SAMP:COUN 2 READ? Typical Response：+0.00000000E+00,-2.29304902E-06

- When automatic reference selection is ON, the first measurement made is used as the null value for all subsequent measurements.

[SENSe:]CURRent:{AC|DC}:NULL:VALue is set to this value. Automatic null value selection will be disabled.

- When automatic null value selection is disabled (OFF), the null value is specified by this command: [SENSe:] CURRent:{AC|DC}:NULL:VALue.
- The instrument enables automatic null value selection when the null function is enabled ([SENSe:]CURRent:{AC|DC}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.2.4 [SENSe:]CURRent:{AC|DC}:RANGe {<range>|MIN|MAX|DEF}

[SENSe:]CURRent:{AC|DC}:RANGe? [{MIN|MAX|DEF}]

Selects a fixed measurement range for AC or DC current measurements.

Note: This parameter setting is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
SDM3045X models: <range>：{600uA|6mA|60mA|600mA|6A|10A|AUTO}，Default: AUTO	+2.00000000E-01

The 600uA and 6mA ranges can only be set in DC mode, and the minimum range is 60mA in AC mode SDM3055 models: <range>：{200uA|2mA|20mA|200mA|2A|10A|AUTO}，Default: AUTO The 200uA and 2mA ranges can only be set in DC mode, and the minimum range is 20mA in AC mode SDM3065X models: <range>：{200uA|2mA|20mA|200mA|2A|10A|AUTO}，Default: AUTO	
Take SDM3065X as an example, Configure AC current measurements using the 2A range. Make and read two measurements: CONF:CURR:AC CURR:AC:RANG 2 SAMP:COUN 2 READ? Typical Response： +3.53049833E-04,+3.54828343E-04	

- Selecting a fixed range ([SENSe:]<*function*>:RANGe) disables autoranging.
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- Unlike CONFigure and MEASure?, this command does not support the 10 A range
- This parameter is set to its default value after a Factory Reset

6.2.5 [SENSe:]CURRent:{AC|DC}:RANGe:AUTO {OFF|ON|ONCE}

[SENSe:]CURRent:{AC|DC}:RANGe:AUTO?

Disables or enables auto-ranging for AC or DC current measurements. Autoranging is convenient because it automatically selects the range for each measurement based on the input signal. Specifying ONCE performs an immediate auto-range and then turns auto-ranging off.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default ON	0 (OFF) or 1 (ON)
Configure AC current measurements and perform an immediate autorange. Make and read two measurements: CONF:CURR:AC CURR:AC:RANG:AUTO ONCE SAMP:COUN 2 READ? Typical Response： +5.79294185E-06,+5.79294185E-06	

- With auto-ranging enabled, the instrument selects the range based on the input signal.
- ONCE performs an immediate auto-range, and then sets auto-ranging to OFF. (Thus, the query returns"0")
- Selecting a fixed range ([SENSe:]<function>:RANGe) disables autoranging.
- This parameter is set to its default value after a Factory Reset.

6.2.6 [SENSe:]CURRent[:DC]:NPLC {<PLC>|MIN|MAX|DEF}

[SENSe:]CURRent[:DC]:NPLC? [{MIN|MAX|DEF}]

Sets the integration time in number of power line cycles (PLC) for DC current measurements. Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a

measurement. A longer integration time gives better measurement resolution but slower measurement speed.

Parameter	Typical Return
SDM3045X//SDM3055 models: <PLC> : {0.3|1|10}，Default:10 On the front panel,0.3|1|10 corresponds to the Speed menu under Fast|Middle|Slow SDM3065X models： <PLC> :{100|10|1|0.5|0.05|0.005}，Default:10 On the front panel,100|10|1|0.5|0.05|0.005 corresponds to the NPLC menu under 100PLC|10PLC|1PLC|0.5PLC|0.05PLC|0.005PLC(50Hz power supply)	+10.00000000E+01
Configure DC current measurements using a 10 PLC integration time. Then make and read one measurement: CONF:CURR:DC CURR:DC:NPLC 10 READ? Typical Response: +1.49999321E+00	

◆ This parameter is set to its default value after a Factory Reset.

6.2.7 [SENSe:]CURRent[:AC]:BANDwidth{|MIN|MAX|DEF}

[SENSe:]CURRent[:AC]:BANDwidth? [{MIN|MAX|DEF}]

Sets the bandwidth for AC current measurements. This command only supports SDM3065X models.

The instrument uses three different AC filters that enable you either to optimize low-frequency accuracy or

to achieve faster AC settling times following a change in input signal amplitude.

Parameter	Typical Return
{3Hz|20Hz|200Hz}，default 20Hz.	20Hz
Configure AC current measurements using a 20Hz,then make and read an AC current measurement: CONF:CURR:AC CURR:AC: BAND 20 READ? Typical Response: +1.49950478E+00	

- This parameter is set to its default value after a Factory Reset.

6.2.8 [SENSe:]CURRent[:DC]:AZ[:STATe] {ON|1|OFF|0}

[SENSe:]CURRent[:DC]:AZ[:STATe]?

Enables or disables the Auto-Zero function for DC current measurements. This command only supports SDM3065X models

Parameter	Typical Return
{ON|1|OFF|0}，default ON	0 (OFF) or 1 (ON)
Provide 1.5A DC current and Configure DC current measurements and perform an immediate autozero. Make and read two measurements CONF:CURR:DC CURR:DC:AZ ON SAMP:COUN 2	

READ? Typical Response: +1.49938088E+00,+1.49938994E+00

- ON (default): The SDM internally measures the offset following each measurement. It then subtracts that measurement from the preceding reading. This prevents offset voltages present on the SDM's input circuitry from affecting measurement accuracy.
- OFF: The instrument uses the last measured zero measurement and subtracts it from each measurement. It takes a new zero measurement each time you change the function, range, or integration time.
- This parameter is set to its default value after a Factory Reset.

6.2.9 [SENSe]:CURRent[:DC]:FILTer[:STATe] {ON|1|OFF|0}

[SENSe]:CURRent[:DC]:FILTer[:STATe]?

Enables or disables the filter for DC current measurements. This command only supports SDM3045X, and SDM3055 models.

Parameter	**Typical Return**
{ON|1|OFF|0}, Default: OFF	0(OFF) or1(ON)
Enables the filter for DC current measurements: CONF:CURR:DC CURR:FILT ON	

- This parameter is set to its default value after a Factory Reset.

6.3. [SENSe:]{FREQuency|PERiod} Subsystem

This subsystem configures frequency and period measurements.

Command Summary

[SENSe:]{FREQuency|PERiod}:NULL[:STATe]

[SENSe:]{FREQuency|PERiod}:NULL:VALue

[SENSe:]{FREQuency|PERiod}:NULL:VALue:AUTO

[SENSe:]{FREQuency|PERiod}:VOLTage:RANGe

[SENSe:]{FREQuency|PERiod}:VOLTage:RANGe:AUTO

[SENSe:]{FREQuency|PERiod}:APERture

6.3.1 [SENSe:]{FREQuency|PERiod}:NULL[:STATe] {ON|1|OFF|0}

[SENSe:]{FREQuency|PERiod}:NULL[:STATe]?

Enables or disables the null function for frequency and period measurements.

This parameter is shared between frequency and period measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default OFF	0 (OFF) or 1 (ON)
Configure frequency measurements, and provide 120 Hz signal, using the null function to subtract 100 Hz from the measurements. Make and read two measurements: CONF:FREQ FREQ:NULL:STAT ON	

FREQ:NULL:VAL 100 SAMP:COUN 2 READ? Typical Response： +2.00016058E+01,+2.00010927E+01

◆ Enabling the scaling function also enables automatic null value selection ([SENSe:] {FREQuency|PERiod}:NULL:VALue:AUTO ON).

◆ To set a fixed null value, use this command: [SENSe:]{FREQuency|PERiod}:NULL:VALue.

◆ The instrument disables the null function after a Factory Reset or CONFigure function.

6.3.2 [SENSe:]{FREQuency|PERiod}:NULL:VALue {<value>|MIN|MAX|DEF}

[SENSe:]{FREQuency|PERiod}:NULL:VALue? [{MIN|MAX|DEF}]

Sets the null value for frequency or period measurements.

This parameter is shared between frequency and period measurements.

Parameter	**Typical Return**
-1.2E6 to +1.2E6, Default：0	+1.00000000E-02
Configure frequency measurements, and provide 120 Hz signal, using the null function to subtract 100 Hz from the measurements. Make and read two measurements: CONF:FREQ FREQ:NULL:STAT ON FREQ:NULL:VAL 100	

SAMP:COUN 2 READ? Typical Response： +2.00016058E+01,+2.00010927E+01

- Specifying a null value disables automatic null value selection ([SENSe:]{FREQuency|PERiod}:NULL:VALue:AUTO OFF).
- To use the null value, the null state must be on ([SENSe:]{FREQuency|PERiod}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.3.3 [SENSe:]{FREQuency|PERiod}:NULL:VALue:AUTO {ON|1|OFF|0}

[SENSe:]{FREQuency|PERiod}:NULL:VALue:AUTO?

Enables or disables automatic null value selection for frequency and period measurements. This parameter is shared between frequency and period measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default ON	0 (OFF) or 1 (ON)
Configure frequency measurements, and provide 150 Hz signal, using the null function to subtract 50Hz from the measurements. Make and read two measurements: CONF:FREQ FREQ:NULL:STAT ON FREQ:NULL:VAL 50	

SAMP:COUN 2

READ?

Typical Response：+1.00001821E+02,+1.00001744E+02

Using automatic null value selection to measure:

CONF:FREQ

FREQ:NULL:STAT ON

FREQ:NULL:VAL:AUTO ON

READ?

Typical Response： +0.00000000E+00,-3.98579024E-04

- When automatic reference selection is ON, the first measurement made is used as the null value for all subsequent measurements. [SENSe:]{FREQuency|PERiod}:NULL:VALue is set to this value. Automatic null value selection will be disabled.
- When automatic null value selection is disabled (OFF), the null value is specified by this command: [SENSe:]{FREQuency|PERiod}:NULL:VALue.
- The instrument enables automatic null value selection when the null function is enabled ([SENSe:]{FREQuency|PERiod}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.3.4 [SENSe:]{FREQuency|PERiod}:VOLTage:RANGe {<range>|MIN|MAX|DEF}

[SENSe:]{FREQuency|PERiod}:VOLTage:RANGe? [{MIN|MAX|DEF}]

Selects a fixed voltage range for frequency and period measurements:

Note：This parameter is shared between frequency and period measurements.

Parameter	Typical Return
SDM3045X models: <range>：{600 mV|6 V|60 V|600 V|750V}，Default :60V SDM3055//SDM3065X models: <range>：{200 mV|2 V|20 V|200 V|750V}，Default :20V	+2.00000000E+01
Take SDM3065X as an example, Provide 50 Hz signal, and configures frequency measurements using the 20 V AC voltage range. Make and read two measurements. CONF:FREQ FREQ:VOLT:RANG 20 SAMP:COUN 2 READ? Typical Response：+5.00332541E+01,+5.00086975E+01	

- Selecting a fixed range ([SENSe:]<*function*>:RANGe) disables autoranging.
- The input signal for frequency or period measurements has an AC voltage component. Use this command to select a fixed voltage range for frequency and period measurements. Use [SENSe:]{FREQuency|PERiod}:VOLTage:RANGe:AUTO to disable or enable voltage auto-ranging or use CONFigure:{FREQuency|PERiod} to select voltage auto-ranging by default.
- If the input voltage is too large for the selected voltage range (manual ranging), the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.3.5 [SENSe:]{FREQuency|PERiod}:VOLTage:RANGe:AUTO {OFF|ON|ONCE}

[SENSe:]{FREQuency|PERiod}:VOLTage:RANGe:AUTO?

Disables or enables voltage auto-ranging for frequency and period measurements. Autoranging is convenient because it automatically selects the range for each measurement based on the input signal. Specifying ONCE performs an immediate auto-range and then turns auto-ranging off.

Note：This parameter is shared between frequency and period measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default: ON	0 (OFF) or 1 (ON)
Provide 1.5kHz signal and configure frequency measurements and perform an immediate AC voltage auto-range. Make and read two measurements: CONF:FREQ FREQ:VOLT:RANG:AUTO ONCE SAMP:COUN 2 READ? Typical Response：+1.50001619E+03,+1.50001513E+03	

- With auto-ranging enabled, the instrument selects the range based on the input signal.
- The input signal for frequency or period measurements has an AC voltage component. Use this command to disable or enable voltage auto-ranging or use CONFigure:{FREQuency|PERiod} to select voltage auto-ranging by default. Use [SENSe:]{FREQuency|PERiod}:VOLTage:RANGe to select a fixed voltage range for frequency and period measurements.
- Selecting a fixed range ([SENSe:]<*function*>:RANGe) disables autoranging.
- ONCE performs an immediate auto-range, and then sets auto-ranging to OFF. (Thus, the query returns"0")
- This parameter is set to its default value after a Factory Reset.

6.3.6 [SENSe:]{FREQuency|PERiod}:APERture {<value>|MIN|MAX|DEF}

[SENSe:]{FREQuency|PERiod}:APERture? [{MIN|MAX|DEF}]

Set the gate time for frequency and period measurements. This command only supports SDM3065X models.

Note：This parameter is shared between frequency and period measurements.

Parameter	**Typical Return**
<value>：{1ms|10ms|100ms|1s}，default :100ms	+1.00000000E-01
Provide 1.5kHz signal and configures frequency measurements using the 100ms gate time. Make and read two measurements CONF:FREQ FREQ: APER 0.1 SAMP:COUN 2 READ? Typical Response：+1.50001619E+03,+1.50001513E+03	

- ONCE performs an immediate auto-range, and then sets auto-ranging to OFF. (Thus, the query returns"0".)
- This parameter is set to its default value after a Factory Reset.

6.4. [SENSe:]{RESistance|FRESistance} Subsystem

This subsystem configures two- and four-wire resistance measurements.

Command Summary

6.4.1 [SENSe:]{RESistance|FRESistance}:NPLC {<PLC>|MIN|MAX|DEF}

[SENSe:]{RESistance|FRESistance}:NPLC? [{MIN|MAX|DEF}]

Sets the integration time in number of power line cycles (PLC) for all alternating current measurements. Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a measurement. A longer integration time gives better measurement resolution but slower measurement speed.

Parameter	Typical Return
SDM3045X//SDM3055 models: <PLC> : {0.3|1|10}，Default:10 On the front panel,0.3|1|10 corresponds to the Speed menu under Fast|Middle|Slow SDM3065X models： <PLC> :{100|10|1|0.5|0.05|0.005}，Default:10 On the front panel,100|10|1|0.5|0.05|0.005 corresponds to the NPLC menu under 100PLC|10PLC|1PLC|0.5PLC|0.05PLC|0.005PLC(50 Hz power supply) or	+1.00000000E+01

100PLC|10PLC|1PLC|0.6PLC|0.05PLC|0.006PLC(60 Hz power supply)	
Configure 2-wire resistance measurements using a 10 PLC integration time. CONF:RES RES:NPLC 10 READ? Typical Response：+1.00000060E+03	

- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.4.2 [SENSe:]{RESistance|FRESistance}:NULL[:STATe] {ON|1|OFF|0}

[SENSe:]{RESistance|FRESistance}:NULL[:STATe]?

Enables or disables the null function for all resistance measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default OFF	0 (OFF) or 1 (ON)
Configure 2-wire resistance measurements, provide 1.5KΩ measurement resistance, and using the null function to subtract the relative reference resistance of 500Ω. Make and read two measurements CONF:RES RES:NULL:STAT ON RES:NULL:VAL 500 SAMP:COUN 2 READ?	

Typical Response：+1.00007049E+03,+1.00006631E+03

◆ Enabling the scaling function also enables automatic null value selection ([SENSe:]{RESistance/FRESistance}:NULL:VALue:AUTO ON).

◆ To set a fixed null value, use this command: [SENSe:]{RESistance/FRESistance}:NULL:VALue.

◆ The instrument disables the null function after a Factory Reset or CONFigure function.

6.4.3 [SENSe:]{RESistance|FRESistance}:NULL:VALue {<value>|MIN|MAX|DEF}

[SENSe:]{RESistance|FRESistance}:NULL:VALue? [{MIN|MAX|DEF}]

Set a fixed null value for all resistance measurements.

Parameter	**Typical Return**
-110 M 0 cal110 M ， default: 0	+0.00000000E+00
Configure 2-wire resistance measurements, provide 1.5KΩ measurement resistance, using the null function to subtract the relative reference resistance of 500Ω.Make and read two measurements: CONF:RES RES:NULL:STAT ON RES:NULL:VAL 500 SAMP:COUN 2 READ?	

Typical Response：+1.00007049E+03,+1.00006631E+03

- Specifying a null value disables automatic null value selection ([SENSe:]{RESistance/FRESistance}:NULL:VALue:AUTO OFF).
- To use the null value, the null state must be on ([SENSe:]{RESistance|FRESistance}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.4.4 [SENSe:]{RESistance|FRESistance}:NULL:VALue:AUTO {ON|1|OFF|0}

[SENSe:]{RESistance|FRESistance}:NULL:VALue:AUTO?

Enables or disables automatic null value selection for all resistance measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default on	0 (OFF) or 1 (ON)
Configure 2-wire resistance measurements, provide 1.5KΩ measurement resistance, using the null function to subtract the relative reference resistance of 500Ω. Make and read two measurements: CONF:RES RES:NULL:STAT ON RES:NULL:VAL 500 SAMP:COUN 2 READ? Typical Response：+1.00007049E+03,+1.00006631E+03	

Using automatic null value selection to measure: CONF:RES RES:NULL:STAT ON RES:NULL:VAL:AUTO ON SAMP:COUN 2 READ? Typical Response : +0.00000000E+00,-5.06651893E-03

- When automatic reference selection is ON, the first measurement made is used as the null value for all subsequent measurements.

  [SENSe:]{RESistance|FRESistance}:NULL:VALue will be set to this value. Automatic null value selection will be disabled.
- When automatic null value selection is disabled (OFF), the null value is specified by this command: [SENSe:] {RESistance|FRESistance}:NULL:VALue.
- The instrument enables automatic null value selection when the null function is enabled ([SENSe:] {RESistance|FRESistance}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.4.5 [SENSe:]{RESistance|FRESistance}:RANGe {<range>|MIN|MAX|DEF}

[SENSe:]{RESistance|FRESistance}:RANGe? [{MIN|MAX|DEF}]

Selects a fixed measurement range for all resistance measurements.

Parameter	Typical Return
SDM3045X models: <range>: {600 Ω|6 kΩ|60 kΩ|600 kΩ|6 MΩ|60 MΩ|100 MΩ}, default: 6kΩ	+2.00000000E+03

SDM3055/: <range>: {200 Ω|2 kΩ|20 kΩ|200 kΩ|2 MΩ|10 MΩ|100 MΩ}, default: 2kΩ SDM3065X models: <range>: {200 Ω|2 kΩ|20 kΩ|200 kΩ|1 MΩ|10 MΩ|100 MΩ}, default :2kΩ	
Take SDM3065X as an example, provide 1.5KΩ measurement resistance and configure 2-wire resistance measurements using the 2kΩ range. Then make and read one measurement: CONF:RES RES:RANG 2000 SAMP:COUN 2 READ? Typical Response :+1.50000104E+03,+1.50000224E+03	

- Selecting a fixed range ([SENSe:]<*function*>:RANGe) disables auto-ranging.
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.

6.4.6 [SENSe:]{RESistance|FRESistance}:RANGe:AUTO {OFF|ON|ONCE}

[SENSe:]{RESistance|FRESistance}:RANGe:AUTO?

Disables or enables auto-ranging for all resistance measurements. Autoranging is convenient because it automatically selects the range for each measurement based on the input signal. Specifying ONCE performs an immediate auto-range and then turns auto-ranging off.

Parameter	**Typical Return**

{ON|1|OFF|0}，default ON	0(OFF) or 1(ON)
Configure 2-wire resistance measurements and perform an immediate auto-range. Make and read two measurements: CONF:RES RES:RANG:AUTO ONCE SAMP:COUN 2 READ? Typical Response :+1.50000104E+03,+1.50000224E+03	

- With auto-ranging enabled, the instrument selects the range based on the input signal.
- ONCE performs an immediate auto-range, and then sets auto-ranging to OFF. (Thus, the query returns"0")
- Selecting a fixed range ([SENSe:]<*function*>:RANGe) disables autoranging.
- This parameter is set to its default value after a Factory Reset.

6.4.7 [SENSe:]{RESistance|FRESistance}:AZ[:STATe] {ON|1|OFF|0}

[SENSe:]{RESistance|FRESistance}:AZ[:STATe]?

Enables or disables the Auto-Zero for RESistance and FRESistance measurements. This command only supports SDM3065X models

Parameter	**Typical Return**
{ON|1|OFF|0}，default: ON	0 (OFF) or 1 (ON)
Configure RESistance measurements, enables the Auto-Zero function to RESistance measurements. CONF:RES	

```
RES:AZ ON
SAMP:COUN 2
READ?

Typical Response :+1.50000104E+03,+1.50000224E+03
```

- ON (default): the SDM internally measures the offset following each measurement. It then subtracts that measurement from the preceding reading. This prevents offset voltages present on the SDM's input circuitry from affecting measurement accuracy.
- OFF: the instrument uses the last measured zero measurement and subtracts it from each measurement. It takes a new zero measurement each time you change the function, range, or integration time.
- This parameter is set to its default value after a Factory Reset.

6.5. [SENSe:]TEMPerature Subsystem

This subsystem configures temperature measurements.

Command Summary

[SENSe:]TEMPerature:NULL[:STATe]

[SENSe:]TEMPerature:NULL:VALue

[SENSe:]TEMPerature:NULL:VALue:AUTO

[SENSe:]TEMPerature:TRANsducer?

[SENSe:]TEMPerature:{UDEFine|MDEFine}:{THER|RTD}:TRANsducer:LIST?

[SENSe:]TEMPerature:{UDEFine|MDEFine}:{THER|RTD}:TRANsducer

[SENSe:]TEMPerature:{UDEFine|MDEFine}:{THER|RTD}:TRANsducer:POINt?

6.5.1 [SENSe:]TEMPerature:NULL[:STATe] {ON|1|OFF|0}

[SENSe:]TEMPerature:NULL[:STATe]?

Enables or disables the null function for temperature measurements.

Parameter	Typical Return
{ON|1|OFF|0}，Default :OFF	0(OFF) or 1(ON)
Configure RTD PT100 measurements, using the null function to subtract the relative reference temperature of 30°.Make and read two measurements: CONF:TEMP RTD,PT100 TEMP:NULL:STAT ON	

TEMP:NULL:VAL 30

SAMP:COUN 2

READ?

Typical Response : +7.01418194E+01,+7.01417827E+01

- Enabling the scaling function also enables automatic null value selection ([SENSe:]TEMPerature:NULL:VALue:AUTO ON).
- To set a fixed null value, use this command: [SENSe:]TEMPerature:NULL:VALue.
- The instrument disables the null function after a Factory Reset or CONFigure function.

6.5.2 [SENSe:]TEMPerature:NULL:VALue {<value>|MIN|MAX|DEF}

[SENSe:]TEMPerature:NULL:VALue? [{MIN|MAX|DEF}]

Set a null value for temperature measurements.

Parameter	**Typical Return**
-1.0E15 to +1.0E15，default :0	+0.00000000E+00

Configure RTD PT100 measurements, using the null function to subtract the relative reference temperature of 30°. Make and read two measurements:

CONF:TEMP RTD,PT100

TEMP:NULL:STAT ON

TEMP:NULL:VAL 30

SAMP:COUN 2

READ?

Typical Response : +7.01418194E+01,+7.01417827E+01

- Specifying a null value disables automatic null value selection ([SENSe:]TEMPerature:NULL:VALue:AUTO OFF).
- To use the null value, the null state must be on ([SENSe:]TEMPerature:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.5.3 [SENSe:]TEMPerature:NULL:VALue:AUTO {ON|1|OFF|0}

[SENSe:]TEMPerature:NULL:VALue:AUTO?

Enables or disables automatic null value selection for temperature measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default OFF	0(OFF)or1(ON)
Configure RTD PT100 measurements, using the null function to subtract the relative reference temperature of 30°. Make and read two measurements: CONF:TEMP RTD,PT100 TEMP:NULL:STAT ON TEMP:NULL:VAL 30 SAMP:COUN 2 READ? Typical Response : +7.01418194E+01,+7.01417827E+01 Using automatic null value selection to measure: CONF:TEMP RTD,PT100 TEMP:NULL:STAT ON	

TEMP:NULL:VAL:AUTO ON SAMP:COUN 2 READ? Typical Response : +0.00000000E+00,-9.36733100E-05

- When automatic reference selection is ON, the first measurement made is used as the null value for all subsequent measurements. [SENSe:]TEMPerature:NULL:VALue will be set to this value. Automatic null value selection will be disabled.
- When automatic null value selection is disabled (OFF), the null value is specified by this command: [SENSe:] TEMPerature:NULL:VALue.
- The instrument enables automatic null value selection when the null function is enabled ([SENSe:] TEMPerature:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.5.4 [SENSe:]TEMPerature:TRANsducer?

Query current effective sensor

Parameter	**Typical Return**
(none)	MANU DEFINE,THER,BITS90
Query the current effective sensor of the temperature measurement function: TEMP:TRAN? Typical Response：PT100	

- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.5.5 [SENSe:]TEMPerature:MDEFine:{THER|RTD}:TRANsducer: LIST?

Query user-defined/manufacturer curing RTD/THER sensor list.

Parameter	Typical Return
(none)	BITS90,EITS90,JITS90,KITS90,NITS90,RITS90,SITS90,TITS90
Query manufacturer curing RTD sensor list: TEMP:MDEF:RTD:TRAN:LIST? Typical Response：PT100	

6.5.6 [SENSe:]TEMPerature:MDEFine:{THER|RTD}:TRANsducer <transducer>

Set the current effective sensor.

Parameter	Typical Return
RTD: <transducer>:PT100|PT1000 THER: <transducer>: BITS90|EITS90|JITS90|KITS90|NITS90|RITS90|SITS90|TITS90	(none)
Set the THER KITS90 as the current sensor： CONF:TEMP TEMP:MDEF:THER:TRAN KITS90	

- The information can be obtained through the

[SENSe:]TEMPerature:{UDEFine|MDEFine}:{THER|RTD}:TRANsducer:LIST?

- This command will respond to the default sensor(KITS90) after a Factory Reset.

6.5.7 [SENSe:]TEMPerature:MDEFine:{THER|RTD}:TRANsducer: POINt?

Inquiry the definition of the sensor.

Parameter	Typical Return
RTD: <transducer>:PT100|PT1000 THER: <transducer>: BITS90|EITS90|JITS90|KITS90|NITS90|RITS90|SITS90|TITS90	1|-6.45800|-270.0000,2|-6.44100|-260.0000 … The return value to serial number voltage value | | temperature (point) of the format arrangement, comma-separated between different points.
Inquiry detailed information of THER KITS90 definition. TEMP:MDEF:THER:TRAN:POIN? KITS90	

6.6. [SENSe:]VOLTage Subsystem

This subsystem configures AC voltage measurements and DC voltage measurements.

Command Summary

[SENSe:]VOLTage:{AC|DC}:NULL[:STATe]

[SENSe:]VOLTage:{AC|DC}:NULL:VALue

6.6.1 [SENSe:]VOLTage:{AC|DC}:NULL[:STATe] {ON|1|OFF|0}

[SENSe:] VOLTage:{AC|DC}:NULL[:STATe]?

Enables or disables the null function for AC or DC voltage measurements.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default OFF	0(OFF)or1(ON)
Configure AC voltage measurements, Provide 1.5V AC voltage signal, using the null function to subtract the relative reference voltage of 1V from the measurements. Make and read two measurements: CONF:VOLT:AC VOLT:AC:NULL:STAT ON VOLT:AC:NULL:VAL 1 SAMP:COUN 2	

READ? Typical Response： +4.97335411E-01,+4.97355425E-01

- Enabling the scaling function also enables automatic null value selection ([SENSe:]VOLTage:{AC|DC}:NULL:VALue:AUTO ON).
- To set a fixed null value, use this command: [SENSe:]VOLTage:{AC|DC}:NULL:VALue.
- The instrument disables the null function after a Factory Reset or CONFigure function.

6.6.2 [SENSe:]VOLTage:{AC|DC}:NULL:VALue {<value>|MIN|MAX|DEF}

[SENSe:]VOLTage:{AC|DC}:NULL:VALue? [{MIN|MAX|DEF}]

Set a null value for voltage measurements.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
SDM3055,SDM3065X models: DCV NULL range:-1100 to +1,100 V，default :0 ACV NULL range:-825 to +825 V，default :0	+0.00000000E+00
Configure AC voltage measurements, Provide 1.5V AC voltage, using the null function to subtract the relative reference voltage of 1V from the measurements. Make and read two measurements: CONF:VOLT:AC VOLT:AC:NULL:STAT ON VOLT:AC:NULL:VAL 1V	

SAMP:COUN 2 READ? Typical Response： +4.98368721E-01,+4.98375337E-01

- Specifying a null value disables automatic null value selection ([SENSe:]VOLTage:{AC|DC}:NULL:VALue:AUTO OFF).
- To use the null value, the null state must be on ([SENSe:]VOLTage:{AC|DC}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.6.3 [SENSe:]VOLTage:{AC|DC}:NULL:VALue:AUTO {ON|1|OFF|0} [SENSe:]VOLTage:{AC|DC}:NULL:VALue:AUTO?

Enables or disables automatic null value selection for AC voltage or DC voltage measurements.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}, default :OFF	0 (OFF) or 1 (ON)
Configure AC voltage measurements, Provide 1.5 V AC voltage, using the null function to subtract the relative reference voltage of 500 mV from the measurements. Make and read two measurements: CONF:VOLT:AC VOLT:AC:NULL:STAT ON VOLT:AC:NULL:VAL 0.5	

SAMP:COUN 2

READ?

Typical Response： +1.00003621E+00,+1.00003746E+00

Using automatic null value selection to measure:

CONF:VOLT:AC

VOLT:AC:NULL:STAT ON

VOLT:AC:NULL:VAL:AUTO ON

SAMP:COUN 2

READ?

Typical Response： +0.00000000E+00,+0.01230000E+00

- When automatic reference selection is ON, the first measurement made is used as the null value for all subsequent measurements.
  [SENSe:]VOLTage:{AC|DC}:NULL:VALue will be set to this value. Automatic null value selection will be disabled.
- When automatic null value selection is disabled (OFF), the null value is specified by this command:[SENSe:]VOLTage:{AC|DC}:NULL:VALue.
- The instrument enables automatic null value selection when the null function is enabled ([SENSe:]VOLTage: {AC|DC}:NULL:STATe ON).
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.6.4 [SENSe:]VOLTage:{AC|DC}:RANGe {<range>|MIN|MAX|DEF}

[SENSe:]VOLTage:{AC|DC}:RANGe? [{MIN|MAX|DEF}]

Selects a fixed measurement range for AC and DC voltage measurements. Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC

measurements.

Parameter	Typical Return
SDM3045X models: <range>: {600 mV|6 V|60 V|600 V|1000 V(DC)/750V(AC)} AC default range: 60V; DC default range: 1000V SDM3055//SDM3065X models: <range>: {200 mV|2 V|20 V|200 V|1000 V(DC)/750V(AC)} AC default range: 20V;DC default range: 1000V	+2.00000000E+00
Take SDM3065X as an example, Configure AC voltage measurements using the 2V range. Make and read two measurements: CONF:VOLT:AC VOLT:AC:RANG 2 SAMP:COUN 2 READ? Typical Response: +8.21650028E-03,+8.17775726E-03	

- Selecting a fixed range ([SENSe:]<*function*>:RANGe) disables autoranging.

  If the input signal is greater than can be measured on the specified manual range, the instrument displays the word overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- The instrument is set to the default range, with auto-ranging enabled ([SENSe:]VOLTage:{AC|DC}:RANGe:AUTO ON), after a Factory Reset.

6.6.5 [SENSe:]VOLTage:{AC|DC}:RANGe:AUTO {OFF|ON|ONCE}

[SENSe:]VOLTage:{AC|DC}:RANGe:AUTO?

Disables or enables auto-ranging for AC and DC voltage measurements. Autoranging is convenient because it automatically selects the range for each measurement based on the input signal. Specifying ONCE performs an immediate auto-range and then turns auto-ranging off.

Note: This parameter is not shared between AC and DC measurements. The parameters are independent for AC and DC measurements.

Parameter	**Typical Return**
{ON|1|OFF|0}，default: ON	0(OFF)or1(ON)
Configure AC voltage measurements and perform an immediate auto-range. Make and read two measurements: CONF:VOLT:AC VOLT:AC:RANG:AUTO ONCE SAMP:COUN 2 READ? Typical Response： +8.36187601E-03,+8.34387541E-03	

- With auto-ranging enabled, the instrument selects the range based on the input signal.
- ONCE performs an immediate auto-range, and then sets auto-ranging to OFF. (Thus, the query returns"0")
- Selecting a fixed range ([SENSe:]<function>:RANGe) disables autoranging.
- This parameter is set to its default value after a Factory Reset.

6.6.6 [SENSe:]VOLTage[:DC]:NPLC {<PLC>|MIN|MAX|DEF}

[SENSe:]VOLTage[:DC]:NPLC? [{MIN|MAX|DEF}]

Sets the integration time in number of power line cycles (PLC) for DC voltage and ratio measurements. Integration time is the period that the instrument's analog-to-digital (A/D) converter samples the input signal for a measurement. A longer integration time gives better measurement resolution but slower measurement speed.

Parameter	Typical Return
SDM3045X//SDM3055 models: <PLC> : {0.3|1|10}，Default:10 On the front panel,0.3|1|10 corresponds to the Speed menu under Fast|Middle|Slow SDM3065X models： <PLC> :{100|10|1|0.5|0.05|0.005}，Default:10 On the front panel, 100|10|1|0.5|0.05|0.005 corresponds to the NPLC menu under 100PLC|10PLC|1PLC|0.5PLC|0.05PLC|0.005PLC	+1.00000000E+01
Configure DC voltage measurements using a 10 PLC integration time. CONF:VOLT:DC VOLT:DC:NPLC 10 READ? Typical Response：+2.00630075E+00	

- This parameter is set to its default value after a Factory Reset.

6.6.7 [SENSe:]VOLTage[:DC]:IMPedance <impedance>

[SENSe:]VOLTage[:DC]:IMPedance?

Select the input impedance of DC voltage measurement.

Parameter	**Typical Return**
{10M|10G}，default : 1OM	10M
Select 10M as the input impedance: VOLT:DC:IMP 10M	

- As for the SDM3045X, This parameter is only valid in the 600 mV range.
- As for the SDM3055/, This parameter is only valid in the 200 mV and 2 V ranges.
- As for the SDM3065X, This parameter is only valid in the 200 mV, 2 V and 20 V ranges.
- This parameter is set to its default value after a Factory Reset.

6.6.8 [SENSe:]VOLTage[:AC]:BANDwidth{|MIN|MAX|DEF}

[SENSe:]VOLTage[:AC]:BANDwidth? [{MIN|MAX|DEF}]

Sets the bandwidth for AC voltage measurements. This command only supports SDM3065X models

Parameter	Typical Return
{3|20|200}，default 20. On the front panel,3|20|200 corresponds to the NPLC menu under 3Hz|20Hz|200Hz	20 Hz
Configure AC voltage measurements using a 20 Hz: CONF:VOLT:AC VOLT:AC: BAND 20 READ? Typical Response：+2.00630075E+00	

- This parameter is set to its default value after a Factory Reset.

6.6.9 [SENSe:]VOLTage[:DC]:AZ[:STATe] {ON|1|OFF|0}

[SENSe:]VOLTage[:DC]:AZ[:STATe]?

Enables or disables the Auto-Zero function for DC voltage measurements. This command only supports SDM3065X models

Parameter	Typical Return
{ON|1|OFF|0}，default OFF	0 (OFF) or 1 (ON)
Configure DC voltage measurements and perform an immediate autozero. Make and read the measurement: CONF:VOLT:DC VOLT:DC:AZ ON READ? Typical Response：+2.00630075E+00	

- ON (default): the SDM internally measures the offset following each measurement. It then subtracts that measurement from the preceding reading. This prevents offset voltages present on the SDM's input circuitry from affecting measurement accuracy.
- OFF: the instrument uses the last measured zero measurement and subtracts it from each measurement. It takes a new zero measurement each time you change the function, range, or integration time.
- This parameter is set to its default value after a Factory Reset.

6.6.10 [SENSe]:VOLTage[:DC]:FILTer[:STATe] {ON|1|OFF|0}

[SENSe]:VOLTage[:DC]:FILTer[:STATe]?

Enables or disables the filter for DC voltage measurements. This command only supports SDM3045X //SDM3055 models.

Parameter	Typical Return
{ON|1|OFF|0}，default : ON	0(OFF) or 1(ON)
Enables the filter for DC voltage measurements. CONF:VOLT:DC VOLT:FILT ON	

This parameter is set to its default value after a Factory Reset.

6.7. [SENSe:]CAPacitance Subsystem

This subsystem configures capacitance measurements.

Command Summary

[SENSe:]CAPacitance:NULL[:STATe]

[SENSe:]CAPacitance:NULL:VALue

[SENSe:]CAPacitance:NULL:VALue:AUTO

[SENSe:]CAPacitance:RANGe

[SENSe:]CAPacitance:RANGe:AUTO

6.7.1 [SENSe:]CAPacitance:NULL[:STATe] {ON|1|OFF|0}

[SENSe:]CAPacitance:NULL[:STATe]?

Enable or disable the null function for capacitance measurement.

Parameter	Typical Return
{ON|1|OFF|0}，default :OFF	0(OFF)or1(ON)
Configure capacitance measurements, using the null function to subtract the relative reference capacitance of 100nF from the measurements. Make and read two measurements: CONF:CAP CAP:NULL:STAT ON CAP:NULL:VAL 100nF SAMP:COUN 2 READ? Typical Response: +4.79899595E-10,+4.79906446E-10	

- Enable the scaling function will also enable automatic null value selection.

([SENSe:]Capacitance:{AC|DC}:NULL:VALue:AUTO ON).

- To set a fixed null value, use this command:

[SENSe:]Capacitance:{AC|DC}:NULL:VALue.

- The instrument disables the null function after a Factory Reset or CONFigure function.

6.7.2 [SENSe:]CAPacitance:NULL:VALue {<value>|MIN|MAX|DEF}

[SENSe:]CAPacitance:NULL:VALue? [{MIN|MAX|DEF}]

Set a null value for capacitance measurements.

.

Parameter	Typical Return
-12 to +12 mF，Default :0.	+1.20000000E-02
Configure capacitance measurements, using the null function to subtract the relative reference capacitance of 100nF from the measurements. Make and read two measurements: CONF:CAP CAP:NULL:STAT ON CAP:NULL:VAL 100nF SAMP:COUN 2 READ? Typical Response： +4.79899595E-10,+4.79906446E-10	

- To use the null value, the null state must be on ([SENSe:]Capacitance:NULL:STATe ON).
- The instrument disables the null function after a Factory Reset or CONFigure function.
- Specifies a null value will disable automatic null value selection ([SENSe:]Capacitance:NULL:VALue:AUTO OFF).

6.7.3 [SENSe:]CAPacitance:NULL:VALue:AUTO {ON|1|OFF|0}

[SENSe:]CAPacitance:NULL:VALue:AUTO?

Enable or disable automatic null value selection for capacitance measurements.

Parameter	Typical Return
{ON|1|OFF|0}，default ON	0(OFF)or1(ON)

Configure capacitance measurements, using the null function to subtract the relative reference capacitance of 100nF from the measurements. Make and read two measurements:

CONF:CAP

CAP:NULL:STAT ON

CAP:NULL:VAL 100nF

SAMP:COUN 2

READ?

Typical Response： +4.79899595E-10,+4.79906446E-10

Using automatic null value selection to measure:

CONF:CAP

CAP:NULL:STAT ON

CAP:NULL:VAL:AUTO ON

SAMP:COUN 2

READ?

Typical Response： +0.00000000E+00,+1.02300000E-01

- When automatic reference selection is ON, the first measurement made is used as the null value for all subsequent measurements. [SENSe:]Capacitance:NULL:VALue will be set to this value. Automatic null value selection will be disabled.
- When automatic null value selection is disabled (OFF), the null value is specified by this command:[SENSe:]Capacitance:NULL:VALue.
- The instrument enables automatic null value selection when the null function is enabled

([SENSe:]Capacitance:NULL:STATe ON).

- This parameter is set to its default value after a Factory Reset or CONFigure function

6.7.4 [SENSe:]CAPacitance:RANGe {<range>|MIN|MAX|DEF}

[SENSe:]CAPacitance:RANGe? [{MIN|MAX|DEF}]

Selects a fixed range for capacitance measurements.

Parameter	Typical Return
SDM3045X/ SDM3055/ models: <range>: {2nF|20nF|200nF|2uF |20uF| 200uF|10000uF}，Default :2uF SDM3065X models： <range>: {2nF|20nF|200nF|2uF|20uF|200uF|2mF|20mF|100mF}，Default :2uF	+2.00000000E-06
Configure capacitance measurements, select 2uF range, Make and read two measurements. CONF:CAP CAP:RANG 2E-6 SAMP:COUN 2 READ? Typical Response： +7.28283777E-10,+7.28268544E-10	

- Choose the fixed range ([SENSe:]<function>:RANGe) ,disableautoranging..
- If the input signal is greater than can be measured on the specified manual range, the instrument displays the word Overload on the front panel and returns "+9.90000000E+37" from the remote interface.
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.7.5 [SENSe:]CAPacitance:RANGe:AUTO {OFF|ON|ONCE}

[SENSe:]CAPacitance:RANGe:AUTO?

Disables or enables auto-ranging for capacitance measurements. Autoranging is convenient because it automatically selects the range for each measurement based on the input signal. Specifying ONCE performs an immediate auto-range and then turns auto--ranging off.

Parameter	**Typical Return**
{ON|1|OFF|0}，default :ON	0(OFF)or1ON)
Configure capacitance measurements and perform an immediate auto-range. . Make and read two measurements. CONF:CAP CAP:RANG:AUTO ONCE SAMP:COUN 2 READ? Typical Response： +7.28283777E-10,+7.28268544E-10	

- The situation in the automatic adjustment range enabled, the input signal range based on the instrument.
- ONCE performs an immediate auto-range, and then sets auto-ranging to OFF. (Thus, the query returns"0")
- This parameter is set to its default value after a Factory Reset or CONFigure function.

6.8. [SENSe:]CONTinuity Subsystem

This subsystem configures continuity measurements.

Command Summary

[SENSe:]CONTinuity:THReshold:VALue

[SENSe:]CONTinuity:VOLume:STATe

6.8.1 [SENSe:]CONTinuity:THReshold:VALue {<value>|MIN|MAX|DEF}

[SENSe:]CONTinuity:THReshold:VALue?

Set the threshold resistors for continuity measurements.

Parameter	**Typical Return**
0~2000 Ω，default :50	+5.00000000E+01
Sets the threshold resistance to 2000Ω. CONF:CONT CONT:THR:VAL 2000 CONT:THR:VAL? Typical Response：+2.00000000E+03	

◆ This parameter is set to its default value after a Factory Reset or CONFigure function.

6.8.2 [SENSe:]CONTinuity:VOLume:STATe{<value>|LOW|MIDDLE|HIGH}

[SENSe:]CONTinuity:VOLume:STATe?

Sets the buzzer volume level for the continuity measurements.

Note: This parameter is shared between the continuity measurement and the diode measurement.

Parameter	**Typical Return**
LOW|MIDDLE|HIGH	LOW|MIDDLE|HIGH
Sets the buzzer volume to high CONF:CONT CONT:VOL:STAT HIGH CONT:VOL:STAT? Typical Response：HIGH	

- This parameter is set to its default value after a Factory Reset or CONFigure function.

7. SYSTem Subsystem

The System subsystem includes general commands and remote interface configuration commands.

7.1. General command

Command Summary

SYSTem:BEEPer:STATe

SYSTem:PRESet

7.1.1 SYSTem:BEEPer:STATe {ON|1|OFF|0}

SYSTem:BEEPer:STATe?

Disables or enables the beep heard during continuity, diode, or Probe Hold measurements.

Parameter	Typical Return
{ON|1|OFF|0}，default ON	0(OFF)or1(ON)
Disable keyboard click: SYST:BEEP:STAT OFF	

- This parameter is set to its default value after a Factory Reset.

7.1.2 SYSTem:PRESet

This command is nearly identical to *RST. The difference is that, for SCPI operations, *RST resets the instrument and restores the machine to factory default parameters, while SYSTem:PRESet restores the machine's parameter configuration to the state configuration when the power is up.

Parameter	Typical Return
Null	Null
Restore the parameter configuration of the machine to the state configuration when the power is turned on: SYSTem:PRESet	

7.2. Remote interface configuration commands.

Command Summary

SYSTem:COMMunicate:LAN:IPADdress

SYSTem:COMMunicate:LAN:SMASk

SYSTem:COMMunicate:GPIB:ADDRess

SYSTem:COMMunicate:LAN:TYPE

SYSTem:COMMunicate:LAN:GATeway

7.2.1 SYSTem:COMMunicate:LAN:IPADdress "<address>"

SYSTem:COMMunicate:LAN:IPADdress? [{CURRent|STATic}]

Assigns a static Internet Protocol (IP) address for the instrument.

Parameter	Typical Return
< address >: "nnn.nnn.nnn.nnn" Query: {CURRent|STATic}，default CURRent	“169.254.3.5”
Set a static IP address: SYST:COMM:LAN:IPAD "10.11.13.212"	

SYST:COMM:LAN:IPAD? CURRent Typical Response：“10.11.13.212”

- For the query form, specify "CURRent" (default) to read the value currently being used by the instrument. Specify "STATic" to read the value currently stored in nonvolatile memory within the instrument
- This setting is non-volatile; it is not changed after a Factory Reset.

7.2.2 SYSTem:COMMunicate:LAN:SMASk "<mask>"

SYSTem:COMMunicate:LAN:SMASk? [{CURRent|STATic}]]

Assigns a subnet mask for the instrument to use in determining whether a client's IP address is on the same local subnet.

Parameter	**Typical Return**
<mask>: "nnn.nnn.nnn.nnn",default"255.255.0.0" Query: {CURRent|STATic}，default CURRent	“255.255.255.0”
Set the subnet mask: SYST:COMM:LAN:SMAS "255.255.255.0" SYST:COMM:LAN:SMAS? CURRent Typical Response：“255.255.255.0”	

- A value of "0.0.0.0" or "255.255.255.255" indicates that the subnet is not being used.
- **CURRent:** returns address currently being used by the instrument.
- **STATic:** returns static address from non-volatile memory. If DHCP is disabled or unavailable, this address will be used.

◆ This setting is non-volatile; it is not changed after a Factory Reset.

7.2.3 SYSTem:COMMunicate:GPIB:ADDRess<address>

SYSTem:COMMunicate:GPIB:ADDRess?

Set the GPIB (IEEE-488) address for the device. Each device on the GPIB interface must have a unique address.

Parameter	Typical Return
0 to 30 , default: 18	18
Set GPIB address to 20: SYST:COMM:GPIB:ADDR 20 SYST:COMM:GPIB:ADDR? Typical Response：20	

◆ This setting is non-volatile; it is not changed after a Factory Reset.

7.2.4 SYSTem:COMMunicate:LAN:GATeway <address>

SYSTem:COMMunicate:LAN:GATeway?

Set a default gateway to the specified IP address. This allows the instrument to communicate with systems outside the local subnet.

Parameter	Typical Return

<address>: “AA:BB:CC:DD”	none
Set gateway address to “10.11.13.1”: SYST:COMM:LAN:GAT “10.11.13.1” SYST:COMM:LAN:GAT? Typical Response：”10.11.13.1”	

- This setting is non-volatile; it is not changed after a Factory Reset.

8. TRIGger Subsystem

The TRIGger subsystem configures the triggering that controls measurement acquisition.

Command Summary

TRIGger:COUNt

TRIGger:DELay

TRIGger:DELay:AUTO

TRIGger:SLOPe

TRIGger:SOURce

8.1. TRIGger:COUNt {<count>|MIN|MAX|DEF|INFinity}

TRIGger:COUNt? [{MIN|MAX|DEF}]

Selects the number of triggers that are accepted by the instrument before returning to the "idle" trigger state.

Parameter	Typical Return
1 to 1,000,000 or continuous (INFinity). Default 1	+1.00000000E+00 For a continuous trigger (INFinity), the query returns "+9.90000000E+37".
Return ten sets of five DC voltage measurements: CONF:VOLT:DC SAMP:COUN 5 TRIG:COUN 10	

READ? Typical Response: -1.85425399E-04, fi50 measurements)

- You can use the specified trigger count in conjunction with a sample count (SAMPle:COUNt), which sets the number of samples to be made per trigger. In this case, the number of measurements returned will be the sample count multiplied by the trigger count.
- You can store up to 10,000 measurements in the reading memory of the SDM. If reading memory overflows, new measurements will overwrite the oldest measurements stored; the most recent measurements are always preserved.
- This parameter is set to its default value after a Factory Reset.

8.2. TRIGger:DELay {<seconds>|MIN|MAX|DEF}
TRIGger:DELay? [{MIN|MAX|DEF}]

Set the delay between the trigger signal and the first measurement. Make sure the input signal is stable until the measurement results are obtained.

Parameter	**Typical Return**
0 to ~3600 seconds (~1µs steps), default 1 s	+1.00000000E-06
Take SDM3065X for an example. Configure DC voltage measurements, Provide 1V DC voltage signal, Return three DC voltage measurements. There is a 2-second delay before each measurement. CONF:VOLT:DC 20 SAMP:COUN 3 TRIG:DEL 2 READ?	

Typical Response：+1.00261197E+00,+1.00261197E+00,+1.00261498E+00

- Due to internal quantization, the actual delay that you set may be slightly different than your specified value. The increment is approximately 1μs. Use the query to determine the exact delay. For example, if you send TRIG:DEL 500 ms and then TRIG:DEL? the actual delay may be +5.00000753E-01.
- By default, TRIGger:DELay:AUTO is ON. The instrument automatically determines the delay based on function, range and integration time (see Automatic Trigger Delays). However, you may need to set a delay longer than the automatic delay for long cables, high capacitance or high impedance signals.
- If you specify a trigger delay with this command, that delay is used for *all* functions (*except* CONTinuity and DIODe) and ranges. The CONTinuity and DIODe tests ignore the trigger delay setting.
- If you have configured the instrument for more than one measurement per trigger (SAMPle:COUNt >1), the delay is inserted after the trigger and between consecutive measurements.
- The instrument selects automatic trigger delay after a Factory Reset.

8.3. TRIGger:DELay:AUTO {ON|1|OFF|0}
TRIGger:DELay:AUTO?

Disables or enables automatic trigger delay. If enabled, the instrument determines the delay based on function, range, and integration time or bandwidth.

Parameter	**Typical Return**
{ON|1|OFF|0}，default ON	0(OFF)or1(ON)
Take SDM3065X for an example. Configure DC voltage measurements, Provide a 1V DC voltage	

signal, enables the automatic trigger delay, and read the result. CONF:VOLT:DC 2 SAMP:COUN 3 TRIG:DEL:AUTO 1 READ? Typical Response：+1.00260896E+00,+1.00260896E+00,+1.00260896E+00

- Selecting a specific trigger delay using TRIGger:DELay disables the automatic trigger delay.
- This parameter is set to its default value after a Factory Reset.

8.4. TRIGger:SLOPe {POSitive|NEGative}

TRIGger:SLOPe?

When the trigger source is Ext, Selects whether the instrument uses the rising edge (POS) or the falling edge (NEG) of the trigger signal on the back panel *Ext Trig* BNC connector. When the trigger source is the level trigger (automatic trigger or single trigger), choose the rising or falling edge of the input signal.

Parameter	**Typical Return**
{POSitive | NEGative} ， default: NEGative	POS or NEG
Return ten sets of five DC voltage measurements, using a positive-going external trigger to start each measurement set: CONF:VOLT:DC SAMP:COUN 5 TRIG:COUN 10 TRIG:SOUR EXT;SLOP POS READ?	

Typical Response: -1.85425399E-04, (50 measurements)

◆ This parameter is set to its default value after a Factory Reset.

8.5. TRIGger:SOURce {IMMediate|EXTernal|BUS}

TRIGger:SOURce?

Selects the trigger source for measurements.

Source	Description
IMMediate	The trigger signal is always present. When you place the instrument in the "wait-for-trigger" state, the trigger is issued immediately..
BUS	The instrument is triggered by *TRG over the remote interface once the DMM is in the "wait-for-trigger" state.
EXTernal	The instrument accepts hardware triggers applied to the rear-panel Ext Trig input and takes the specified number of measurements (SAMPle:COUNt), each time a TTL pulse specified by OUTPut:TRIGger:SLOPe is received. If the instrument receives an external trigger before it is ready, it will buffer one trigger.

Parameter	Typical Return
{IMMediate|EXTernal|BUS} ， default IMMediate	IMM,EXT or BUS
Configure the trigger source to trigger immediately, and read the readings after 20 triggers: CONF:VOLT:DC TRIG:COUN 20 TRIG:SOUR IMM	

READ?

Typical Response: -1.85425399E-04, POSs20 measurements)

- After selecting the trigger source, you must place the instrument in the "wait-for-trigger" state by sending INITiate or READ? A trigger will not be accepted from the selected trigger source until the instrument is in the "wait-for-trigger" state.
- This parameter is set to its default value after a Factory Reset.

9. ROUTe Subsystem (SC model only)

The ROUTe subsystem configures the scanner that controls the scan function.

Only multimeters with scan card options are supported.

Note: Only when the scan card function is enabled, the corresponding write command will take effect.

Command Summary

ROUTe:STATe

ROUTe:SCAN

ROUTe:STARt

ROUTe:FUNCtion

ROUTe:DELay

ROUTe:COUNt:AUTO

ROUTe:COUNt

ROUTe:LIMIt:{HIGH|LOW}

ROUTe:DATA?

ROUTe:CHANnel

ROUTe:CHANnel?

ROUTe:RELAtive

ROUTe:IMPedance

ROUTe:TEMPerature:RTD

ROUTe:TEMPerature:THER

ROUTe:TEMPerature:UNIT

ROUTe:{FREQuency|PERiod}

ROUTe:{DCV|DCI}:AZ

ROUTe:{RESistance|FRESistance}:AZ

ROUTe:{FREQuency|PERiod}:APERture

9.1. ROUTe:STATe?

Inquire whether the scanner card is installed.

Parameter	Typical Return
NULL	0 (OFF) or 1 (ON)
Query whether the SDM has a scan card installed ROUTe:STATe? Typical Response：ON（Indicates that this SDM is installed with a scan card.）	

9.2. ROUTe:SCAN {ON|1|OFF|0}

ROUTe:SCAN?

Open or close the scanner card function.

Parameter	Typical Return
{ON|1|OFF|0}	0 (OFF) or 1 (ON)
Enable the scanning card function, and then query the status of the scanning function: ROUTe:SCAN ON ROUTe:SCAN? Typical Response：ON	

9.3. ROUTe:STARt {ON|1|OFF|0}

ROUTe:STARt?

Start or stop scanning card measurement.

Parameter	Typical Return
{ON|1|OFF|0}	0 (OFF) or 1 (ON)
Enable scanner card function and start scanner card measurement, and then inquire the status of scanner measurement: ROUTe:SCAN ON ROUTe:STARt ON ROUTe:STARt? Typical Response：ON	

9.4. ROUTe:FUNCtion {SCAN|STEP}

ROUTe:FUNCtion?

Configure scan card loop mode.

Parameter	Typical Return
{SCAN|STEP}	SCAN or STEP
Open the scan card function, configure the scan card to single-step mode, and query the scan cycle mode: ROUTe:SCAN ON	

ROUTe:FUNC STEP ROUTe:FUNC? Typical Response：STEP

9.5. ROUTe:DELay {<value>|MAX|MIN|DEF }

ROUTe:DELay?

Configure scan card delay time.

Parameter	**Typical Return**
value	0
Open the scanner card function, and configure the scan card delay time to 1S： ROUTe:SCAN ON ROUTe:DEL 1 ROUTe:DEL? Typical Response：1	

9.6. ROUTe:COUNt:AUTO {ON|1|OFF|0}

ROUTe:COUNt:AUTO?

Open or close scan card automatic cycle switch.

Parameter	**Typical Return**
{ON|1|OFF|0}　Default :OFF	0 (OFF) or 1 (ON)

Open the scanner card function, and open the scan card automatic cycle switch: ROUTe:SCAN ON ROUTe:COUN:AUTO ON ROUTe:COUN:AUTO? Typical Response：ON

9.7. ROUTe:COUNt {<value>|MAX|MIN|DEF}

ROUTe:COUNt?

Set the number of scanning card cycle measurements。

Parameter	Typical Return
value	1
Open the scanner card function, and set the number of scan card cycles to 2： ROUTe:SCAN ON ROUTe:COUN 2 ROUTe:COUN? Typical Response：2	

9.8. ROUTe:LIMIt:{HIGH|LOW} {<value>|MAX|MIN|DEF}

ROUTe:LIMIt:{HIGH|LOW}?

Setting the upper and lower limits of the scanning card channel.

Parameter	Typical Return
<value>: (1|2|3|4|5|6|7|8|9|10|11|12|13|14|15)	1
Setting the upper channel of the scanning card is 16, and the lower channel is 2: ROUTe:SCAN ON ROUTe:LIMI:HIGH 16 ROUTe:LIMI:LOW 2 ROUTe:LIMI:LOW? Typical Response: 2	

9.9. ROUTe:DATA? <value>

Returns the final measurement value of the scan card setup channel.

Parameter	Typical Return
<value>: (1|2|3|4|5|6|7|8|9|10|11|12|13|14|15)	null
Open the scanner card function, and query the final measurement value of the second channel: ROUTe:SCAN ON ROUTe:DATA? 2	

Typical Response：1.79221344E-04 VDC

9.10. ROUTe:CHANnel {<channel>,<switch>,<mode>,<range>,<speed>}

Configure scan card channel parameters.

Parameter	Return
1. <channel>: (1|2|3|4|5|6|7|8|9|10|11|12|13|14|15) 2. <switch>: (ON|OFF) 3. <mode>:(DCV|DCI|ACV|ACI|2W|4W|CAP|FRQ|CONT|DIO|TEMP) 4. range DCV(AUTO, 200MLV, 2V, 20V, 200V) ACI|DCI(2A) FRQ|ACV(AUTO, 200MLV, 2V, 20V, 200V) 2W(AUTO,200OHM,2KOHM,20KOHM,200KOHM,2MGOHM, 10MGOHM,100MGOHM) 4W(AUTO,200OHM,2KOHM,20KOHM,200KOHM,2MGOHM, 10MGOHM, 100MGOHM) CAP(AUTO, 2NF, 20NF, 200NF, 2UF, 20UF, 200UF, 10000UF) SPEED (SLOW|FAST)	null
Open the scanner card function, and configure the 1,2 channel： ROUTe:SCAN ON ROUT:CHAN 1,ON,2W,2KOHM,SLOW ROUT:CHAN 2,ON,CONT	

9.11. ROUTe:CHANnel? {CHANNEL}

Return scan card channel parameters.

Parametr	Return
1-16	The return format of the query: <channel>,<switch>,<range>,<speed> 1. CHANNEL (1-16) 2. SWITCH (ON/OFF) 3. MODE(DCV/DCI/ACV/ACI/2W/4W/CAP/FRQ/CONT/DIO/TEMP) 4. RANGE DCV(AUTO, 200MLV, 2V, 20V, 200V) ACI|DCI(2A) FRQ|ACV(AUTO, 200MLV, 2V, 20V, 200V) 2W(AUTO,200OHM,2KOHM,20KOHM,200KOHM,2MGOHM, 10MGOHM,100MGOHM) 4W(AUTO,200OHM,2KOHM,20KOHM,200KOHM,2MGOHM, 10MGOHM, 100MGOHM) CAP(AUTO, 2NF, 20NF, 200NF, 2UF, 20UF, 200UF, 10000UF) SPEED (SLOW/FAST)
Open the scanner card function, and query the parameters of channel 1:	

ROUTe:SCAN ON ROUT:CHAN? 1 Typical Response：1,ON,DCV,AUTO ,SLOW

9.12. ROUTe:RELAtive {<mode>,<switch>}

ROUTe:RELAtive? <mode>

Configure the Relative value switch of the scanning card measurement mode.

Parameter	**Typical Return**
<mode>:(DCV|DCI|ACV|ACI|2W|4W|CAP|FRQ|TEMP) <switch>:(ON|OFF)	ON or OFF
Open the scanner card function, and configure the relative value switch of the voltage measurement mode to be on : ROUTe:SCAN ON ROUT:RELA DCV,ON ROUT:RELA? DCV Typical Response: ON	

9.13. ROUTe:IMPedance {10M|10G}

ROUTe:IMPedance?

Configure the input impedance of the scan card track.

Parameter	Typical Return
{10M |10G}	10M or 10G
Open the scanner card function, and configure the impedance to 10M: ROUTe:SCAN ON ROUT:IMP 10M ROUT:IMP? Typical Response: 10M	

9.14. ROUTe:TEMPerature:RTD <transducer>

ROUTe:TEMPerature: TRANsducer?

Configuration of thermal resistance sensor model.

Parameter	Typical Return
<transducer>:PT100|PT1000	RTD,PT100
Open the scanner card function, and configure the thermal resistance sensor model to PT100: ROUTe:SCAN ON ROUT:TEMP:RTD PT100 ROUT:TEMP:TRAN? Typical Response: RTD,PT100	

9.15. ROUTe:TEMPerature: THER <transducer>

ROUTe:TEMPerature: TRANsducer?

Configuration thermocouple sensor model.

Parameter	Typical Return
<transducer>: BITS90|EITS90|JITS90|KITS90|NITS90|RITS90|SITS90|TITS90	THER,KITS90
Open the scanner card function, and configure the thermal thermocouple sensor model to KITS90: ROUTe:SCAN ON ROUT:TEMP:THER KITS90 ROUT:TEMP:TRAN? Typical Response: THER,KITS90	

9.16. ROUTe:TEMPerature:UNIT {C|F|K}
ROUTe:TEMPerature:UNIT?

Configure the unit of temperature measurement mode.

Parameter	Typical Return
{C|F|K},Default :C	C
Open the scanner card function, and configure the unit of temperature measurement to K: ROUTe:SCAN ON ROUT:TEMP:UNIT K ROUT:TEMP:UNIT?	

Typical Response: K

9.17. ROUTe:{FREQuency | PERiod}

ROUTe:{FREQuency|PERiod}[:STATe]?

Configure the display mode of frequency measurement.

Parameter	**Typical Return**
NULL	0 or 1
Open scan card function, display frequency measurement mode: ROUTe:SCAN ON ROUT:FRE ROUT:FRE? Typical Response: 1	

9.18. ROUTe:{DCV|DCI}:AZ[:STATe] {ON|1|OFF|0}

ROUTe:{DCV|DCI}:AZ[:STATe]?

Configuration auto-zero for scanning card DC voltage or current. Only supports SDM3065X models.

Parameter	**Typical Return**
{ON|1|OFF|0}	0 or 1
Open the scan card function, and configure DC voltage auto-zero to ON: ROUTe:SCAN ON ROUT:DCV:AZ ON	

ROUT:DCV:AZ? Typical Response: 1

9.19. ROUTe:{RESistance| FRESistance}:AZ[:STATe] {ON|1|OFF|0}

ROUTe:{RESistance|FRESistance}:AZ[:STATe]?

Turn on or off the automatic zero setting function of the scanning card resistance measurement。

Parameter	**Typical Return**
{ON|1|OFF|0}	0 or 1
Open the scan card function, and configure the automatic zero setting to ON ： ROUTe:SCAN ON ROUT:RES:AZ ON ROUT:RES:AZ? Typical Response: 1	

9.20. ROUTe:{FREQuency| PERiod}:APERture {1|0.1|0.01|0.001}

ROUTe:{FREQuency| PERiod}:APERture?

Configure the gate time of scan card frequency measurement mode.

Parameter	**Typical Return**
{1|0.1|0.01|0.001}	+1.00000000E+00
Open the scan card function, and configure the frequency gate time to 1S： ROUTe:SCAN ON	

ROUT:FREQ:APER 1

ROUT:FREQ:APER?

Typical Response: +1.00000000E+00

Programming Examples

This chapter gives some examples for the programmer. In these examples you can see how to use VISA or sockets, in combination with the commands described above to control the oscilloscope. By following these examples, you can develop many more applications.

- VISA Examples
  - VC++ Example
  -

VB Example

  - MATLAB Example
  - LabVIEW Example
  - C# Example
- Examples of Using Sockets
  - Python Example
  - C Example

VISA Examples

VC++ Example

Environment: Win7 32-bit, Visual Studio.

Description: Use National Instruments VISA to control the device with USBTMC or TCP/IP access.

Perform a write and read operation.

Steps:

1. Open Visual Studio, create a new VC++ win32 project.
2. Set the project environment to use the NI-VISA library. There are two ways to use NI-VISA, static or automatic:

   a) Static:

   Find the files visa.h, visatype.h, visa32.lib in the NI-VISA installation path, copy them to your project, and add them into the project. In the projectname.cpp file, add the follow two lines:

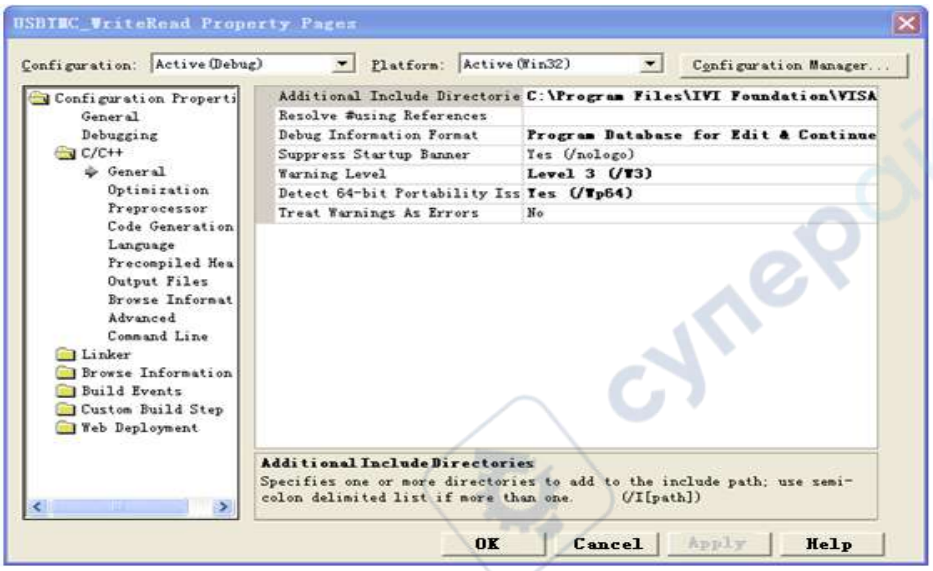
   ```
   #include "visa.h"
   #pragma comment(lib,"visa32.lib")
   ```

   b) Automatic:

   Set the .h file include directory, the NI-VISA install path, in our computer we set the path is: C:\Program Files\IVI Foundation \VISA\WinNT\include. Set this path to:

   project->properties->C/C++->General->Additional Include Directories.

   See the picture:

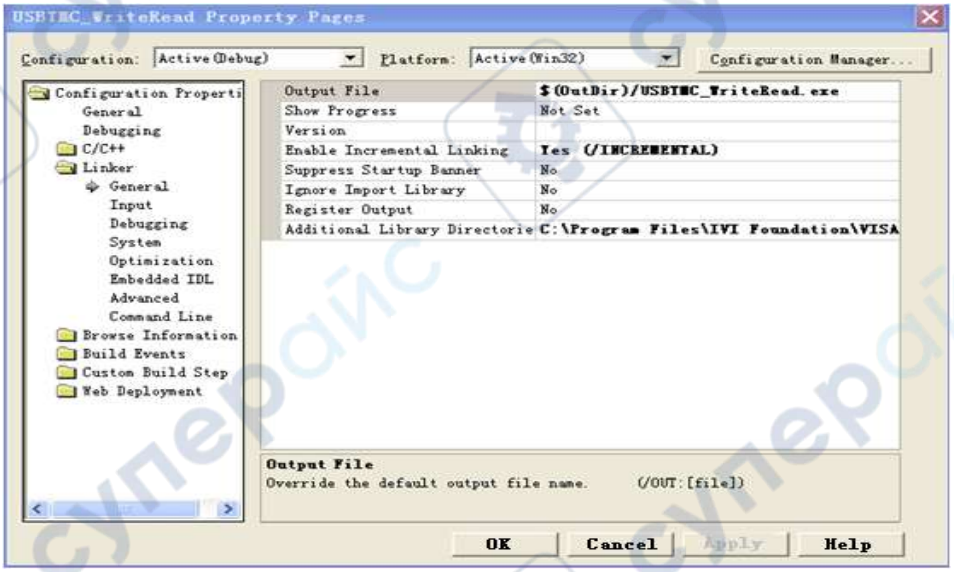


Set lib path set lib file:

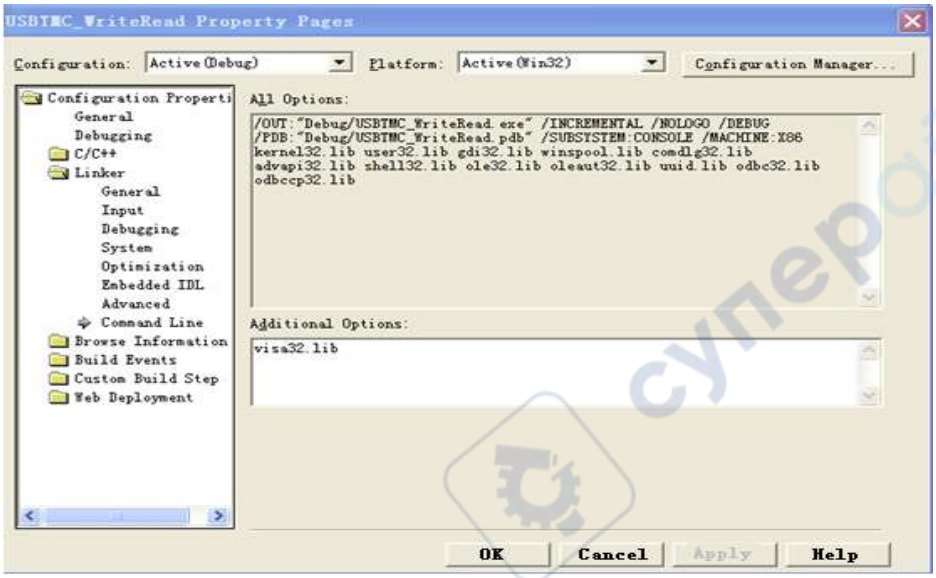
Set lib path: the NI-VISA install path, in our computer we set the path is C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc. Set this path to:

project->properties->Linker->General->Additional Library Directories.

As shown in the pictures below:



Set lib file:project->properties->Linker->Command Line->Additional Options: visa32.lib



Include visa.h file in the projectname.cpp file:

#include <visa.h>

3. Coding:

a) USBTMC:

```
Int     Usbtmc_test()
{
        /* This code demonstrates sending synchronous read & write commands */
        /* to an USB Test & Measurement Class (USBTMC) instrument using      */
        /* NI-VISA                                                           */
        /* The example writes the "*IDN?\n" string to all the USBTMC         */
        /* devices connected to the system and attempts to read back         */
        /* results using the write and read functions.                       */
        /* The general flow of the code is   */
        /*      Open Resource Manager        */
        /*      Open VISA Session to an Instrument                           */
        /*      Write the Identification Query Using viPrintf                */
        /*      Try to Read a Response With viScanf                          */
        /*      Close the VISA Session       */
        /************************************************************/
        ViSession defaultRM;
        ViSession instr;
        ViUInt32 numInstrs;
        ViFindList findList;
        ViUInt32 retCount;
        ViUInt32 writeCount;
        ViStatus status;
        char    instrResourceString[VI_FIND_BUFLEN];
        unsigned    charbuffer[100];
        charstringinput[512];
        int   i;
        /** First we must call viOpenDefaultRM to get the manager
        * handle.   We will store this handle in defaultRM.*/
        status= ViOpenDefaultRM (&defaultRM);
        if (status<VI_SUCCESS)
        {
                printf ("Could not open a session to the VISA Resource Manager!\n");
```

```
            return     status;
    }
    /* Find all the USB TMC VISA resources in our system and store the    number of resources in the
system in numInstrs.                                    */
    status = viFindRsrc (defaultRM, "USB?*INSTR", &findList, &numInstrs, instrResourceString);
    if (status<VI_SUCCESS)
    {
            printf ("An error occurred while finding resources.\nHit enter to continue.");
            fflush(stdin);
            getchar();
            viClose (defaultRM);
            return     status;
    }
    /** Now we will open VISA sessions to all USB TMC instruments.
    * We must use the handle from viOpenDefaultRM and we must
    * also use a string that indicates which instrument to open.    This
    * is called the instrument descriptor.    The format for this string
    * can be found in the function panel by right clicking on the
    * descriptor parameter. After opening a session to the
    * device, we will get a handle to the instrument which we
    * will use in later VISA functions.    The AccessMode and Timeout
    * parameters in this function are reserved for future
    * functionality.    These two parameters are given the value VI_NULL.*/
    for (i= 0; i<numInstrs; i++)
    {
            if (i> 0)
    {
        viFindNext (findList, instrResourceString);
    }
        status = viOpen (defaultRM, instrResourceString, VI_NULL, VI_NULL, &instr);
            if (status<VI_SUCCESS)
            {
                    printf ("Cannot open a session to the device %d.\n", i+1);
                    continue;
            }
            /* * At this point we now have a session open to the USB TMC instrument.
            * We will now use the viPrintf function to send the device the string "*IDN?\n",
            * asking for the device's identification.    */
            char * cmmand ="*IDN?\n";
            status = viPrintf    (instr, cmmand);
            if (status<VI_SUCCESS)
            {
                    printf ("Error writing to the device %d.\n", i+1);
                    status = viClose (instr);
                    continue;
            }
            /** Now we will attempt to read back a response from the device to
            * the identification query that was sent.    We will use the viScanf
            * function to acquire the data.
            * After the data has been read the response is displayed.*/
            status =    viScanf(instr, "%t", buffer);
            if (status<VI_SUCCESS)
    {
    printf ("Error reading a response from the device %d.\n", i+1);
    }
        else
```

```
        {
                        printf ("\nDevice %d: %*s\n", i+1,retCount, buffer);
        }
                status = viClose (instr);
    }
    /** Now we will close the session to the instrument using
    * viClose. This operation frees all system resources.                    */
    status = viClose (defaultRM);
  printf("Press 'Enter' to exit.");
    fflush(stdin);
    getchar();
    return 0;
}
```

b) TCP/IP:

```
    int     TCP_IP_Test(char *pIP)
    {
            char  outputBuffer[VI_FIND_BUFLEN];
            ViSession defaultRM, instr;
            ViStatus status;
            ViUInt32 count;
            ViUInt16 portNo;
            /* First we will need to open the default resource manager. */
            status = viOpenDefaultRM (&defaultRM);
            if (status<VI_SUCCESS)
            {
                    printf("Could not open a session to the VISA Resource Manager!\n");
            }
            /* Now we will open a session via TCP/IP device */
            charhead[256] ="TCPIP0::";
            chartail[] ="::INSTR";
            charresource [256];
            strcat(head,pIP);
            strcat(head,tail);
            status = viOpen (defaultRM, head, VI_LOAD_CONFIG, VI_NULL, &instr);
            if (status<VI_SUCCESS)
            {
                    printf ("An error occurred opening the session\n");
                    viClose(defaultRM);
            }
            status = viPrintf(instr, "*idn?\n");
            status = viScanf(instr, "%t", outputBuffer);
            if (status<VI_SUCCESS)
             {
                    printf("viRead failed with error code: %x \n",status);
                    viClose(defaultRM);
            }
          else
          {
                    printf ("\ndata read from device: %*s\n", 0,outputBuffer);
          }
            status = viClose (instr);
            status = viClose (defaultRM);
          printf("Press 'Enter' to exit.");
            fflush(stdin);
            getchar();
                    return 0;
```

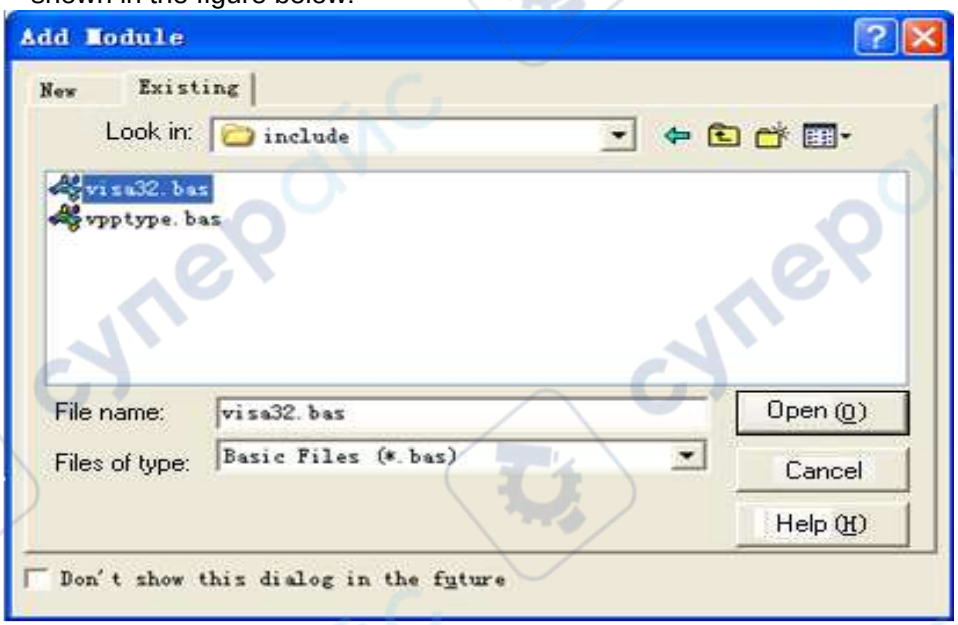
```
}
```

VB Example

Environment: Windows7 32-bit, Microsoft Visual Basic 6.0
Description: The function of this example: Use the NI-VISA, to control the device with USBTMC and TCP/IP access to do a write and read.
Steps:

1. Open Visual Basic, and build a standard application program project.

2. Set the project environment to use the NI-VISA lib: Click the Existing tab of Project->Add Module, search the visa32.bas file in the “include” folder under the NI-VISA installation path and add the file, as shown in the figure below:



3. Coding:

a) USBTMC:

```
Private Function Usbtmc_test() As Long
    ' This code demonstrates sending synchronous read & write commands
    ' to an USB Test & Measurement Class (USBTMC) instrument using
    ' NI-VISA
    ' The example writes the "*IDN?\n" string to all the USBTMC
    ' devices connected to the system and attempts to read back
    ' results using the write and read functions.
     ' The general flow of the code is
    '       Open Resource Manager
    '       Open VISA Session to an Instrument
    '       Write the Identification Query Using viWrite
    '       Try to Read a Response With viRead
    '       Close the VISA Session
    Const MAX_CNT = 200

    Dim defaultRM As Long
    Dim instrsesn As Long
    Dim numInstrs As Long
```

```
Dim findList As Long
Dim retCount As Long
Dim writeCount As Long
Dim status As Long
Dim instrResourceString As String * VI_FIND_BUFLEN
Dim buffer As String * MAX_CNT
Dim i As Integer
 ' First we must call viOpenDefaultRM to get the manager
' handle. We will store this handle in defaultRM.
status = viOpenDefaultRM(defaultRM)
If (status < VI_SUCCESS) Then
    Debug.Print "Could not open a session to the VISA Resource Manager!"
    Usbtmc_test = status
    ExitFunction
End If

' Find all the USB TMC VISA resources in our system and store the
' number of resources in the system in numInstrs.
 status= ViFindRsrc(defaultRM,"USB?*INSTR",findList,numInstrs,instrResourceString)
If (status < VI_SUCCESS) Then
    Debug.Print "An error occurred while finding resources."
    viClose (defaultRM)
    Usbtmc_test = status
    Exit Function
End If

 ' Now we will open VISA sessions to all USB TMC instruments.
 ' We must use the handle from viOpenDefaultRM and we must
' also use a string that indicates which instrument to open. This
' is called the instrument descriptor.    The format for this string
' can be found in the function panel by right clicking on the
' descriptor parameter. After opening a session to the
' device, we will get a handle to the instrument which we
' will use in later VISA functions.    The AccessMode and Timeout
' parameters in this function are reserved for future
' functionality.    These two parameters are given the value VI_NULL.
For i = 0 To numInstrs
    If (i > 0) Then
        status = viFindNext(findList, instrResourceString)
    End If
        status = viOpen(defaultRM, instrResourceString, VI_NULL, VI_NULL, instrsesn)
    If (status < VI_SUCCESS) Then
        Debug.Print "Cannot open a session to the device ", i + 1
        GoTo NextFind
    End If

    ' At this point we now have a session open to the USB TMC instrument.
    ' We will now use the viWrite function to send the device the string "*IDN?",
    ' asking for the device's identification.
    status = viWrite(instrsesn, "*IDN?", 5, retCount)
    If (status < VI_SUCCESS) Then
                Debug.Print "Error writing to the device."
                status = viClose(instrsesn)
                GoTo NextFind
    End If
```

```
        ' Now we will attempt to read back a response from the device to
        ' the identification query that was sent.  We will use the viRead
        ' function to acquire the data.
        ' After the data has been read the response is displayed.
        status = viRead(instrsesn, buffer, MAX_CNT, retCount)
        If (status < VI_SUCCESS) Then
            Debug.Print "Error reading a response from the device.", i + 1
        Else
            Debug.Print i + 1, retCount, buffer
        End If
        status = viClose(instrsesn)
    Next i

    ' Now we will close the session to the instrument using
    ' viClose. This operation frees all system resources.
     status = viClose(defaultRM)
    Usbtmc_test = 0
End Function
```

b) TCP/IP:

```
Private Function TCP_IP_Test(ip As String) As Long
    Dim outputBuffer As String * VI_FIND_BUFLEN
    Dim defaultRM As Long
    Dim instrsesn As Long
    Dim status As Long
    Dim count As Long

    ' First we will need to open the default resource manager.
    status = viOpenDefaultRM (defaultRM)
    If (status < VI_SUCCESS) Then
        Debug.Print "Could not open a session to the VISA Resource Manager!"
        TCP_IP_Test = status
        Exit Function
    End If

    ' Now we will open a session via TCP/IP device
    status = viOpen(defaultRM, "TCPIP0::" + ip + "::INSTR", VI_LOAD_CONFIG, VI_NULL, instrsesn)
     If (status < VI_SUCCESS) Then
        Debug.Print "An error occurred opening the session"
        viClose (defaultRM)
        TCP_IP_Test = status
        Exit Function
    End If

    status = viWrite(instrsesn, "*IDN?", 5, count)
     If (status < VI_SUCCESS) Then
        Debug.Print "Error writing to the device."
    End If
    status = viRead(instrsesn, outputBuffer, VI_FIND_BUFLEN, count)
    If (status < VI_SUCCESS) Then
        Debug.Print "Error reading a response from the device.", i + 1
    Else
        Debug.Print "read from device:", outputBuffer
    End If
    status = viClose(instrsesn)
    status = viClose(defaultRM)
```

```
    TCP_IP_Test = 0
End Function
```

MATLAB Example

Environment: Windows7 32-bit, MATLAB R2010b
Description: The function of this example: Use the NI-VISA, to control the device with USBTMC or TCP/IP access to do a write and read.
Steps:

1. Open MATLAB, and modify the current directory. In this demo, the current directory is modified to D:\USBTMC_TCPIP_Demo.

2. Click File>>New>>Script in the Matlab interface to create an empty M file.

3. Coding:

a) USBTMC:

```
function USBTMC_test()
% This code demonstrates sending synchronous read & write commands
% to an USB Test & Measurement Class (USBTMC) instrument using
% NI-VISA

%Create a VISA-USB object connected to a USB instrument
vu = visa('ni','USB0::0xF4EC::0xEE38::0123456789::INSTR');

%Open the VISA object created
fopen(vu);

%Send the string "*IDN?",asking for the device's identification.
fprintf(vu,'*IDN?');

%Request the data
outputbuffer = fscanf(vu);
disp(outputbuffer);

%Close the VISA object
fclose(vu);
delete(vu);
clear vu;

end
```

b) TCP/IP:

```
function TCP_IP_test( IPstr )
% This code demonstrates sending synchronous read & write commands
% to an TCP/IP instrument using NI-VISA

%Create a VISA-TCPIP object connected to an instrument
%configured with IP address.
vt = visa('ni',['TCPIP0::',IPstr,'::INSTR']);

%Open the VISA object created
fopen(vt);

%Send the string "*IDN?", asking for the device's identification.
fprintf(vt,'*IDN?');
```

```
%Request the data
outputbuffer = fscanf(vt);
disp(outputbuffer);

%Close the VISA object
fclose(vt);
delete(vt);
clear vt;

end
```

LabVIEW Example

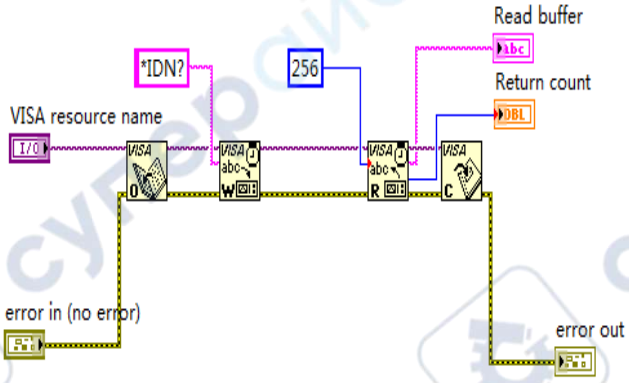
Environment: Windows7 32-bit, LabVIEW 2011
Description: The functions of this example: use the NI-VISA, to control the device with USBTMC and TCP/IP access to do a write and read.
Steps:

1. Open LabVIEW, create a VI file.

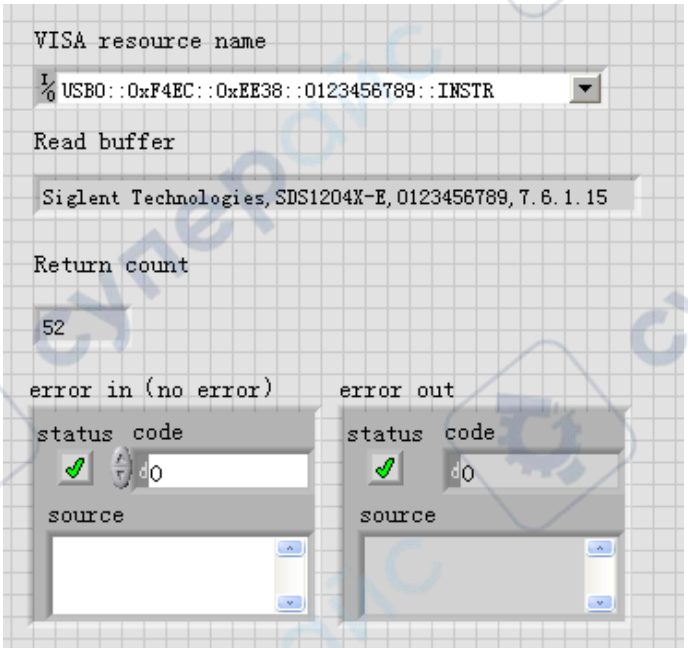
2. Add controls. Right-click in the **Front Panel** interface, select and add **VISA resource name**, error in, error out and some indicators from the Controls column.

3. Open the **Block Diagram** interface. Right-click on the **VISA resource name** and you can select and add the following functions from VISA Palette from the pop-up menu: **VISA Write**, **VISA Read**, **VISA Open** and **VISA Close**.

4. The connection is as shown in the figure below:



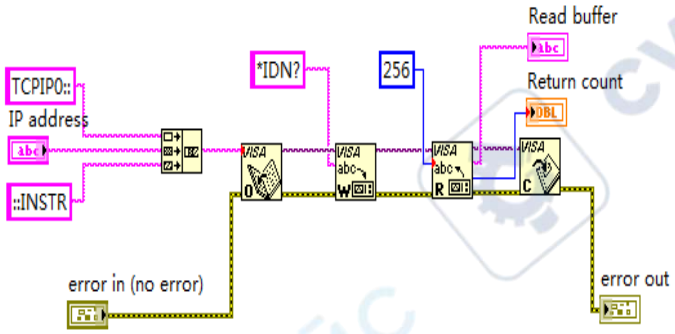
5. Select the device resource from the VISA Resource Name list box and run the program.



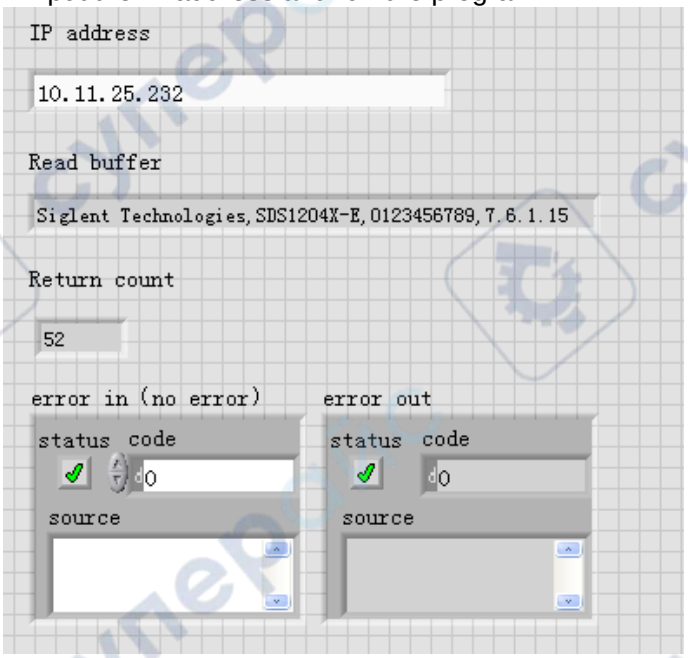
In this example, the VI opens a VISA session to a USBTMC device, writes a command to the device, and reads back the response. After all communication is complete, the VI closes the VISA session.

6. Communicating with the device via TCP/IP is similar to USBTMC. But you need to change VISA Write and VISA Read Function to Synchronous I/O. The LabVIEW default is asynchronous I/O. Right-click the node and select Synchronous I/O Mod>>Synchronous from the shortcut menu to write or read data synchronously.

7. The connection is as shown in the figure below:



8. Input the IP address and run the program.

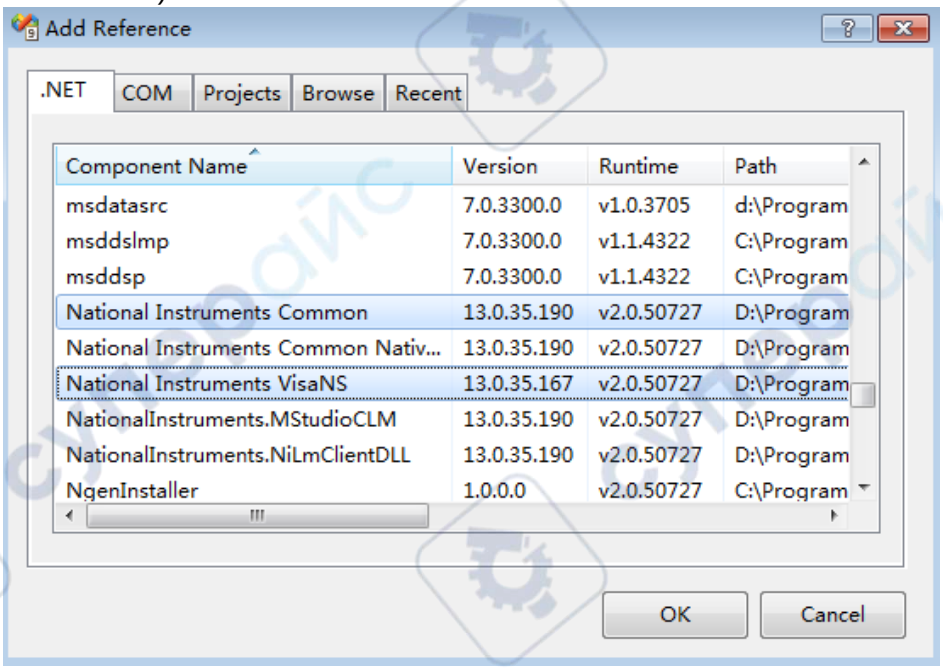


C# Example

Environment: Windows7 32-bit, Visual Studio
Description: The functions of this example: use the NI-VISA, to control the device with USBTMC or TCP/IP access to do a write and read.
Steps:

1. Open Visual Studio, create a new C# project.

2. Add References. Add NationalInstruments.Common.dll and NationalInstruments.VisaNS.dll to the project. (Notice: you must install the .NET Framework 3.5/4.0/4.5 Languages support when you install the NI-VISA.)



3. Coding:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using NationalInstruments.VisaNS;

namespace TestVisa
{
    class Program
    {
        static void Main(string[] args)
        {
            // Find all the USBTMC resources
            string[]
            usbRsrcStrings=ResourceManager.GetLocalManager().FindResources("USB?*INSTR");
            if (usbRsrcStrings.Length <= 0)
            {
                Console.WriteLine("Cannot find USBTMC Device!");
                return;
            }
```

```
            //Choose the first resource string to connect the device.
            //You can input the address manually
            //USBTMC:
            //MessageBasedSession
            mbSession=(MessageBasedSession)ResourceManager.GetLocalManager().Open("USB
            0::0xF4EC::0xEE38::0123456789::INSTR");
            /TCP IP:
            //MessageBasedSession
            mbSession=(MessageBasedSession)ResourceManager.GetLocalManager().Open("TCPI
            P0::192.168.1.100::INSTR");
            MessageBasedSession
            mbSession=(MessageBasedSession)ResourceManager.GetLocalManager().Open(usbR
            srcStrings[0]);
            mbSession.Write("*IDN?");
            string result = mbSession.ReadString();
            mbSession.Dispose();
            Console.WriteLine(result);
        }
    }
}
```

Examples of Using Sockets

Socket communication is a basic communication technology in computer network. It allows applications to communicate through the standard network protocol mechanism built by network hardware and operation system.

This method is a two-way communication between the instrument and the computer through a fixed port number.

Note that SCPI strings are terminated with a “\n” (new line) character.

Python Example

Python has a low-level networking module that provides access to the socket interface. Python scripts can be written for sockets to do a variety of test and measurement tasks.

Environment: Windows7 32-bit, Python v2.7.5
Description: Open a socket, send a query, and repeat this loop for 10 times, finally close the socket.

Below is the code of the script:

```
#!/usr/bin/env python
#-*- coding:utf-8 –*-
#--------------------------------------------------------------------------
# The short script is a example that open a socket, sends a query,
# print the return message and closes the socket.
#--------------------------------------------------------------------------
import socket    # for sockets
import sys  # for exit
import time # for sleep
#--------------------------------------------------------------------------
remote_ip = "10.12.255.209"  # should match the instrument’s IP address
port = 5025 # the port number of the instrument service
count = 0

def SocketConnect():
    try:
        #create an AF_INET, STREAM socket (TCP)
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    except socket.error:
        print ('Failed to create socket.')
        sys.exit();
```

```
    try:
        #Connect to remote server
        s.connect((remote_ip , port))
    except socket.error:
        print ('failed to connect to ip ' + remote_ip)
    return s

def SocketQuery(Sock, cmd):
    try :
        #Send cmd string
        Sock.sendall(cmd)
        Sock.sendall(b'\n')
        time.sleep(1)
    except socket.error:
        #Send failed
        print ('Send failed')
        sys.exit()
    reply = Sock.recv(4096)
    return reply

def SocketClose(Sock):
    #close the socket
    Sock.close()
    time.sleep(.300)

def main():
    global remote_ip
    global port
    global count

    # Body: send the SCPI commands *IDN? 10 times and print the return message
    s = SocketConnect()
    for i in range(10):
        qStr = SocketQuery(s, b'*IDN?')
        print (str(count) + ":: " + str(qStr))
        count = count + 1
    SocketClose(s)
    input('Press "Enter" to exit')

if __name__ == '__main__':
    proc = main()
```

C Example

```
int MySocket;
if((MySocket=socket(PF_INET,SOCK_STREAM,0))==-1)
```

```
{
    exit(1);
}
struct in_addr
{
    unsigned long s_addr;
};
struct sockaddr_in
{
    short int sin_family; // Address family
    unsigned short int sin_port; // Port number
    struct in_addr sin_addr; // Internet address
    unsigned char sin_zero[8]; // Padding
};
struct sockaddr_in MyAddress;

// Initialize the whole structure to zero
memset(&MyAddress,0,sizeof(struct sockaddr_in));
// Then set the individual fields
MyAddress.sin_family=PF_INET; // IPv4
MyAddress.sin_port=htons(5025); // Port number used by most instruments
MyAddress.sin_addr.s_addr=inet_addr("169.254.9.80"); // IP Address

// Establish TCP connection
if(connect(MySocket,(struct sockaddr*)&MyAddress,sizeof(struct sockaddr_in))==-1)
{
    exit(1);
}

// Send SCPI command
if(send(MySocket,"*IDN?\n",6,0)==-1)
{
    exit(1);
}

// Read response
char buffer[200];
int actual;
if((actual=recv(MySocket,&buffer[0],200,0))==-1)
{
    exit(1);
}
buffer[actual]= 0; // Add zero character (C string)
printf("Instrument ID: %s\n",buffer);

// Close socket
if(close(MySocket)==-1)
{
    exit(1);
}
```

About SIGLENT

SIGLENT is an international high-tech company, concentrating on R&D, sales, production and services of electronic test & measurement instruments.

SIGLENT first began developing digital oscilloscopes independently in 2002. After more than a decade of continuous development, SIGLENT has extended its product line to include digital oscilloscopes, isolated handheld oscilloscopes, function/arbitrary waveform generators, RF/MW signal generators, spectrum analyzers, vector network analyzers, digital multimeters, DC power supplies, electronic loads and other general purpose test instrumentation. Since its first oscilloscope was launched in 2005, SIGLENT has become the fastest growing manufacturer of digital oscilloscopes. We firmly believe that today SIGLENT is the best value in electronic test & measurement.

Headquarters:
SIGLENT Technologies Co., Ltd
Add: Bldg No.4 & No.5, Antongda Industrial Zone, 3rd Liuxian Road, Bao'an District, Shenzhen, 518101, China
Tel: + 86 755 3688 7876
Fax: + 86 755 3359 1582
Email: sales@siglent.com
Website: int.siglent.com

USA:
SIGLENT Technologies America, Inc
6557 Cochran Rd Solon, Ohio 44139
Tel: 440-398-5800
Toll Free: 877-515-5551
Fax: 440-399-1211
Email: info@siglent.com
Website: www.siglentna.com

Europe:
SIGLENT Technologies Germany GmbH
Add: Staetzlinger Str. 70
86165 Augsburg, Germany
Tel: +49(0)-821-666 0 111 0
Fax: +49(0)-821-666 0 111 22
Email: info-eu@siglent.com
Website: www.siglenteu.com

Follow us on
Facebook: SiglentTech