Siglent
Programming Guide

SDL1000X Programmable DC Electronic Load

PG0801X-C01A

2019 SIGLENT TECHNOLOGIES CO., LTD

Copyright and Statement

Copyright

SIGLENT TECHNOLOGIES CO., LTD. All rights reserved.

Trademark Information

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

Statement

- **SIGLENT** products are protected by patent laws in and outside of the P.R. China.
- **SIGLENT** reserves the rights to change the specification and price.
- Information in this publication replaces all previous corresponding published material.
- Contents in this manual are not allowed to be copied, extracted or translated in any form or by any means without **SIGLENT**'s permission.

Content

Copyright and Statement 2

1. Programming Overview 1

1.1 Establishing Communications 1

1.1.1 Install NI-VISA 1

1.1.2 Connect the Instrument 3

1.2 Remote Control 4

1.2.1 User-defined Programming 4

1.2.2 Send SCPI Commands via NI_MAX 4

2. SCPI Overview 4

2.1 Command Format 4

2.2 Symbol Instruction 5

2.3 Parameter Type 5

2.4 Command Abbreviation 6

2.5 Special instructions 7

3. System Commands 8

3.1 IEEE Common Commands 8

3.2 Measure Subsystem command 10

3.3 Source Subsystem Command 11

3.3.1 Source Common Subsystem Command 11

3.3.2 Source Current Subsystem Command 12

3.3.3 Source Voltage Subsystem Command 17

3.3.4 Source Power Subsystem Command 19

3.3.5 Source Resistance Subsystem Command 22

3.3.6 Source LED Subsystem Command 26

3.3.7 Source Battery Subsystem Command 27

3.3.8 Source List Subsystem Command 31

3.3.9 Source OCPT Subsystem Command 34

3.3.10 Source OPPT Subsystem Command 37
3.3.11 Source Program Subsystem Command 40
3.3.12 Source Wave Subsystem Command 45
3.3.13 Source Utility Subsystem Command 47
3.4 Subsystem Command 50
3.5 LAN Interface Subsystem Command 54
4. Programming Examples 57
4.1 Examples of Using VISA 57
4.1.1 Example of VC++ 57
4.1.2 Example of VB 64
4.1.3 Example of MATLAB 69
4.1.4 LabVIEW sample 70

1. Programming Overview

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. By using USB and LAN interfaces, in combination with NI-VISA and programming languages, users can remotely control the instruments. Through LAN interface, VXI-11, Sockets and Telnet protocols can be used to communicate with the instruments.

1.1 Establishing Communications

1.1.1 Install NI-VISA

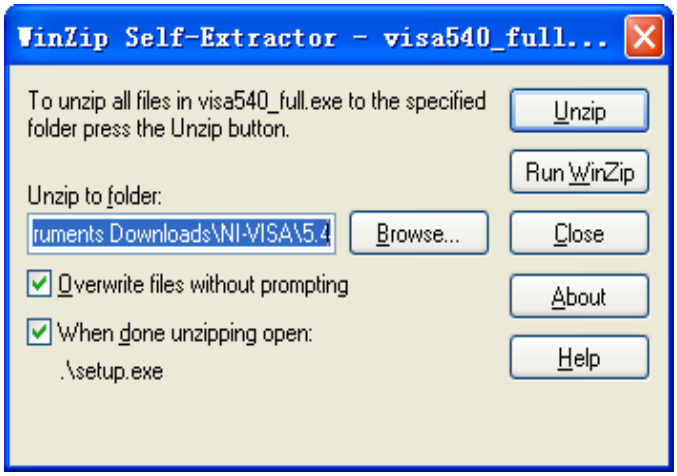
Before programming, you need to install the National Instruments NI-VISA library, which you can download from the National Instruments web site. Currently, NI-VISA is packaged in two versions: a full version and a Run-Time Engine version. The full version includes the NI device drivers and a tool named NI MAX which is a user interface to control and test remotely connected devices. The Run-Time Engine is recommended, as it is a much smaller download than the full version and includes the necessary tools for basic communication to instruments.

For example, you can get the NI-VISA 5.4 full version from: http://www.ni.com/download/ni-visa-5.4/4230/en/.

You also can download NI-VISA Run-Time Engine 5.4 to your PC and install it as the default selection. Its installation process is similar with the full version.

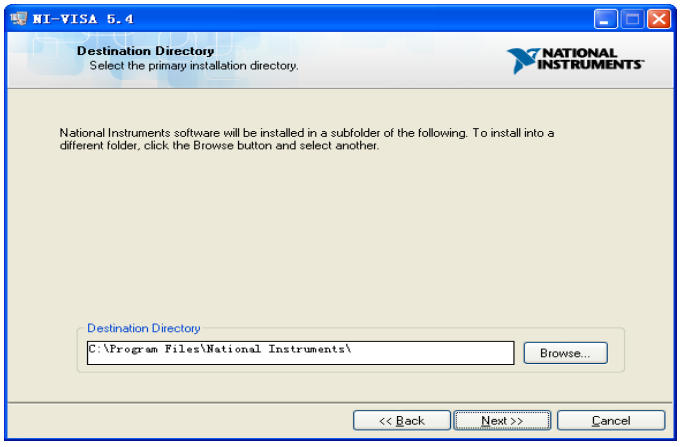
After you downloaded the file, follow these steps to install NI-VISA (The full version of NI-VISA 5.4 is used in this example. Newer versions are likely, and should be compatible with SIGLENT instrumentation. Download the latest version available for the operating system being used by the controlling computer):

a. Double click the visa540_full.exe, dialog shown as below:

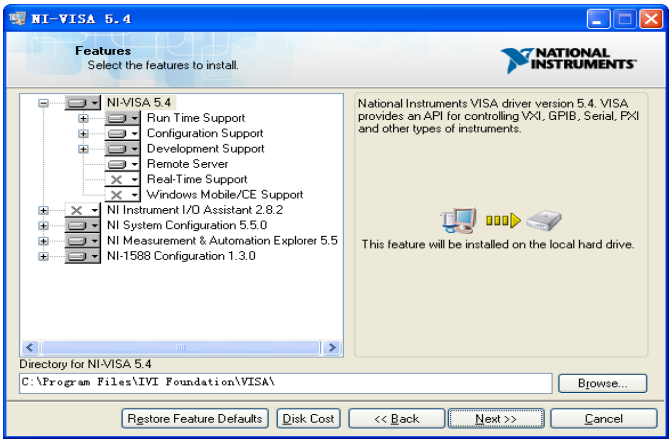


b. Click Unzip, the installation process will automatically launch after unzipping files. If your computer needs to install .NET Framework 4, it may auto start.

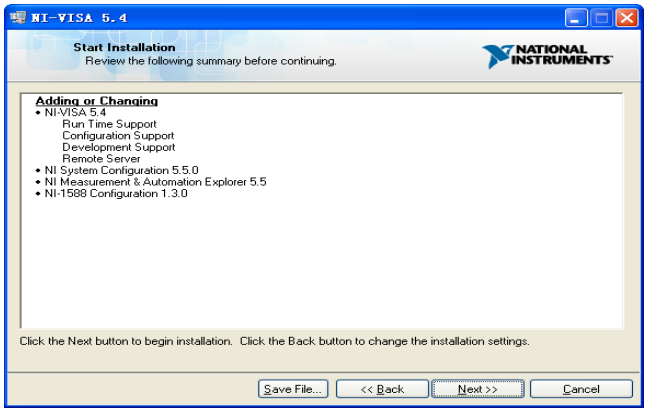
c. The NI-VISA installing dialog is shown above. Click Next to start the installation process.



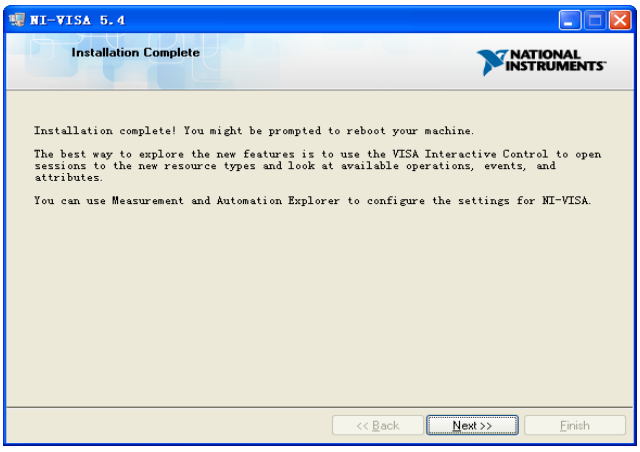
d. Set the install path, default path is “C:\Program Files\National Instruments\”, you can change it. Click Next, dialog shown as above.



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



f. Click Next to begin installation.



g. Now the installation is complete. Reboot your PC.

1.1.2 Connect the Instrument

Depending on the specific model, your electronic load may be able to communicate with a PC through the USB or LAN interface.

Connect the instrument and the USB Host interface of the PC using a USB

cable. Assuming your PC is already turned on, turn on your electronic load, and then the PC will display the “Device Setup” screen as it automatically installs the device driver as shown below.
Wait for the installation to complete and then proceed to the next step.

1.2 Remote Control

1.2.1 User-defined Programming

Users can use SCPI commands via a computer to program and control the electronic load. For details, refer to the introductions in "Programming Examples".

1.2.2 Send SCPI Commands via NI_MAX

NI-Measurement and Automation eXplorer (NI-MAX) is a program created and maintained by National Instruments. It provides a basic remote control interface for VXI, LAN, USB, GPIB, and Serial communications. It is a utility that enables you to send commands one-at-a-time and also retrieve data from connected devices. It is a great tool for troubleshooting and testing command sequences. The oscilloscopes can be controlled remotely by sending SCPI commands via NI-MAX.

2. SCPI Overview

2.1 Command Format

SCPI commands present a hierarchical tree structure containing multiple subsystems, each of the subsystems is made up of a root keyword and several subkeywords. The command string usually starts with “:”, the keywords are separated by “:” and the followed parameter settings are separated by space. Query commands add “?” at the end of the string.
For example:
:SENSe:FREQuency:CENTer <freq>
:SENSe:FREQuency:CENTer?
SENSe is the root key of the command, FREQuency and CENTer are second and third keywords. The command begins with “:”, and separates the keywords at the same time, <freq> separated by space and represents the parameter available for setting; “?” represents a query.

2.2 Symbol Instruction

The following four symbols are not the content of SCPI commands and can not be sent with the commands, but are usually used in the commands.

1、Triangle Brackets < >

The parameter in the triangle brackets can be ignored.When parameter is ignored, the instrument will set parameter to its default. For example: In the “[:SOURce]:BATTery:LEVel <value>” command in “**:**SOURce:BATTery: LEVel 4”.

2、Braces { }

The parameters in the braces are optional which can be ignored or set for one or more times. For example:

The “{ON | OFF | 0 | 1}” parameter in “[:SOURce]:INPut[:STATe] {ON | OFF | 0 | 1}” command.

3、**Vertical Bar |**

The vertical bar is used to separate multiple parameters and when sending the command,you can choose one of the parameters. For example, in the “[:SOURce]:INPut[:STATe] {ON | OFF | 0 | 1}” command, the parameters availiable are “OFF”、“ON”、“0” or “1”; such as “:SOURce:INPut:STATe ON” or “:SOURce:INPut:STATe 0”.

4、**Square Brackets [ ]**

The content in the square brackets can be ignored. When the parameter is ignored, the instrument will set the parameter to its default. For exmple, In the “[:SOURce]:INPut[:STATe]?” command, sending any of the four commands below can generate the same effect:

:INPut?

:INPut:STATe?

:SOURce:INPut?

:SOURce:INPut:STATe?

2.3 Parameter Type

The parameters in the commands introduced in this manual include 5 types: boolean, enumeration, integer, float, specific format.

1、Boolean

The parameters in the commands could be “OFF”, “ON”, “0” or “1”. For example:

[:SOURce]:SHORt[:STATe] {ON | OFF | 0 | 1}

2、Enumeration

The parameter could be any of the values listed. For example:

[:SOURce]:FUNCtion {CURRent | VOLTage | POWer | RESistance | LED}

The parameter is “CURRent”、“VOLTage”、“POWer”、“RESistance”、“LED”

3、Integer

All parameters expressed by <number> and <step> are integer. It can be any integer within the effective value range.

Notes: The command is invalid if the parameter value beyond the scope. The parameter will be rounded to integer when it is decimal.

For example:

[:SOURce]:LIST:LEVel? <step>

The parameter <step> can be set to any integer in effective range. It wil l be rounded to 3 when it is set to 3.988.

4、Float

All parameters expressed by <value> are float. It can be any float within the effective value range.

Notes: The command is invalid if the parameter value beyond the scope. The parameter is also regarded as float when it is integer.

For example:

[:SOURce]:CURRent:IRANGe < value >

The parameter < value > can be set to any real number between 0 and 30.

5、Specific format type

The parameter should be input in specific format and with specified symbol.

For example: LAN:IPADdress <aaa.bbb.ccc.ddd>

The parameter is separated by “.”. It is also take effect when lack numeric characters. The parameter will be set in progressive type.

For example, if the oringnal IP address is “12.13.14.2”, IP address will be set to “15.6.0.2” when the parameter is “15.6.0”.

2.4 Command Abbreviation

All of the commands are not case sensitive, so you can use any of them. But if abbreviation is used, all the capital letters in the command must be

written completely. For example:
:SOURce:CURRent:SLEW:POSitive?
Can be abbreviated to：
:SOUR:CURR:SLEW:POS?

2.5 Special instructions

1、When sets the current range，if <value> is greater than 5, the current range will be set to 30A; if <value> is less than 5, the current range will be set to 5A.

2、When sets the voltage range，if <value> is greater than 36, the voltage range will be set to 150V; if <value> is less than 36, the voltage range will be set to 36V.

3、The parameters is {<value> | MINimum | MAXimum | DEFault} enumerations. Its meanings are as follows

<value> ：Any setting value
MINimum ：The minimum value in the setting range
MAXimum ：The maximum value in the setting range
DEFault ：The default value for the setting value

3. System Commands

3.1 IEEE Common Subsystem Commands

Command Format	*IDN?
Description	Returns an instrument identification information string. The string will contain the manufacturer, model number, serial number and software number.
Example	*IDN?
Response	Siglent\sTechnologies,SDL1020X,0123456789,1.01.01.15.

Command Format	*RST
Description	Rstore the equipment state to be initial state.
Example	*RST

Command Format	*CLS
Description	Clears all bits in all of the event registers and the error list.
Example	*CLS

Command Format	*ESE < number >
Description	Set the bits in the standard event status enable register.
Example	*ESE 16

Command Format	*ESE?
Description	Query the standard event status enable register. The value returned reflects the current state of all the bits in the register.
Example	*ESE?
Response	64

Command Format	*ESR?
Description	Query and clears the standard event status register. The value returned reflects the current state of all the bits in the register.
Example	*ESR?

Response	0

Command Format	*OPC
Description	Set bit 0 in the standard event status register to “1” when all operations have finished.
Example	*OPC

Command Format	*OPC?
Description	Query whether the current operation has been finished. Returns a “1” if the operation is complete.
Example	*OPC?
Response	1

Command Format	*SRE <number>
Description	Set the bits in the status byte enable register.
Example	*SRE 24

Command Format	*SRE?
Description	Query the status byte enable register. The value returned reflects the current state of all the bits in the register.
Example	*SRE?
Response	24

Command Format	*STB?
Description	Query the status byte event register. The value returned reflects the current state of all the bits in the register.
Example	*STB?
Response	72

Command Format	*TST?
Description	This query is used by some instruments for a self test.
Example	*TST?
Response	0

Command Format	*WAI
Description	This command causes the instrument to wait until all pending commands are completed before executing any additional commands.
Example	*WAI

3.2 Measure Subsystem command

Command Format	MEASure:VOLTage[:DC]?
Description	Gets the real-time voltage measurement value
Example	MEASure:VOLTage:DC?
Response	7.924678

Command Format	MEASure:CURRent[:DC]?
Description	Gets the real-time current measurement value
Example	MEASure: CURRent:DC?
Response	3.986634

Command Format	MEASure:POWer[:DC]?
Description	Gets the real-time power measurement value
Example	MEASure:POWer:DC?
Response	31.867329

Command Format	MEASure:RESistance[:DC]?
Description	Gets the real-time resistor measurement value
Example	MEASure: RESistance:DC?
Response	5.842789

Command Format	MEASure:EXT?
Description	Gets the real-time external measurement value in external sink mode.
Example	MEASure: EXT?
Response	3.863521

Command Format	MEASure:WAVEdata? {CURRent | VOLTage | POWer | RESistance}
Description	Gets the waveform data of the waveform display interface in CC/CV/CP/CR mode. Totally include 200 float data .
Example	MEASure:WAVEdata? VOLTage
Response	3.947389,3.928473,3.197493,3.573992…

3.3 Source Subsystem Command

3.3.1 Source Common Subsystem Command

Command Format	[:SOURce]:INPut[:STATe] {ON | OFF | 0 | 1}
Description	Sets the input status of the load (ON or OFF)
Example	:SOURce:INPut:STATe OFF

Command Format	[:SOURce]:INPut[:STATe]?
Description	Query the input status of the load. Return “1” if input status is ON. Otherwise, return “0”
Example	:SOURce:INPut:STATe?
Response	0

Command Format	[:SOURce]:SHORt[:STATe] {ON | OFF | 0 | 1}
Description	Sets the short circuit status of the load (ON or OFF)
Example	:SOURce:SHORt:STATe ON

Command Format	[:SOURce]:SHORt[:STATe]?
Description	Query the short circuit status in current mode of the load. Return “1” if short circuit status is ON. Otherwise, return “0”
Example	:SOURce:SHORt:STATe?
Response	1

Command Format	[:SOURce]:FUNCtion:TRANsient {CURRent | VOLTage | POWer | RESistance}
Description	Sets mode in transient operation (CC/CV/CP/CR)

Example	:SOURce:FUNCtion:TRANsient VOLTage

Command Format	[:SOURce]:FUNCtion:TRANsient?
Description	Query current mode in transient operation.
Example	:SOURce:FUNCtion:TRANsient?
Response	1

Command Format	[:SOURce]:FUNCtion {CURRent | VOLTage | POWer | RESistance | LED}
Description	Sets mode in static operation (CC/CV/CP/CR/LED)
Example	:SOURce:FUNCtion LED

Command Format	[:SOURce]:FUNCtion?
Description	Query current mode in static operation.
Example	:SOURce:FUNCtion?
Response	RESISTANCE

Command Format	[:SOURce]:TEST:STEP?
Description	Query the number of running step in the LIST/PROGRAM test sequence.
Example	:SOURce:TEST:STEP?
Response	3

Command Format	[:SOURce]:TEST:STOP?
Description	Query whether the running steps of the test sequence stop or not. Resturns “1” if test stop or return “0” if test stop.
Example	:SOURce:TEST:STOP?
Response	1

3.3.2 Source Current Subsystem Command

Command Format	[:SOURce]:CURRent[:LEVel][:IMMediate] {<value> | MINimum | MAXimum | DEFault}
Description	Sets the sink current value of CC mode in static operation.
Example	:SOURce:CURRent:LEVel:IMMediate 0.845

Command Format	[:SOURce]:CURRent[:LEVel][:IMMediate]?
Description	Query the preset current value of CC mode in static operation.
Example	:SOURce:CURRent:LEVel:IMMediate?
Response	2.674

Command Format	[:SOURce]:CURRent:IRANGe <value>
Description	Sets the current range of CC mode in static operation.
Example	:SOURce:CURRent:IRANGe 10

Command Format	[:SOURce]:CURRent:IRANGe?
Description	Query the current range of CC mode in static operation.
Example	:SOURce:CURRent:IRANGe?
Response	30

Command Format	[:SOURce]:CURRent:VRANGe <value>
Description	Sets the voltage range of CC mode in static operation.
Example	:SOURce:CURRent:VRANGe 65

Command Format	[:SOURce]:CURRent:VRANGe?
Description	Query the voltage range of CC mode in static operation.
Example	:SOURce:CURRent:VRANGe?
Response	36

Command Format	[:SOURce]:CURRent:SLEW[:BOTH] {<value> | MINimum | MAXimum | DEFault}
Description	Sets the slope of CC mode in static operation. The rise slope and descending slope will be set synchronously.
Example	:SOURce:CURRent:SLEW:BOTH 0.652

Command Format	[:SOURce]:CURRent:SLEW:POSitive {<value> | MINimum | MAXimum | DEFault}
Description	Sets the rise slope of CC mode in static operation.
Example	:SOURce:CURRent:SLEW:POSitive 0.258

Command	[:SOURce]:CURRent:SLEW:POSitive?

Format	
Description	Query the rise slope of CC mode in static operation.s
Example	:SOURce:CURRent:SLEW:POSitive?
Response	0.498

Command Format	[:SOURce]:CURRent:SLEW:NEGative {<value> | MINimum | MAXimum | DEFault}
Description	Sets the descending slope of CC mode in static operation.
Example	:SOURce:CURRent:SLEW: NEGative 1.986

Command Format	[:SOURce]:CURRent:SLEW:NEGative?
Description	Query the descending slope of CC mode in static operation.
Example	:SOURce:CURRent:SLEW: NEGative?
Response	0.187

Command Format	[:SOURce]:CURRent:TRANsient:MODE {CONTinuous | PULSe | TOGGle}
Description	Sets the waveform mode of CC mode in transient operation.
Example	:SOURce:CURRent:SLEW: NEGative 1.986

Command Format	[:SOURce]:CURRent:TRANsient:MODE?
Description	Query the waveform mode of CC mode in static operation.
Example	:SOURce:CURRent:TRANsient:MODE?
Response	CONTINUOUS

Command Format	[:SOURce]:CURRent:TRANsient:IRANGe <value>
Description	Sets the current range of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:IRANGe 9

Command Format	[:SOURce]:CURRent:TRANsient:IRANGe?
Description	Query the current range of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:IRANGe?
Response	30

Command	[:SOURce]:CURRent:TRANsient:VRANGe <value>

Format	
Description	Sets the voltage range of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:VRANGe 80

Command Format	[:SOURce]:CURRent:TRANsient:VRANGe?
Description	Query the voltage range of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:VRANGe?
Response	150

Command Format	[:SOURce]:CURRent:TRANsient:ALEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:ALEVel 4.653

Command Format	[:SOURce]:CURRent:TRANsient:ALEVel?
Description	Query the A Level of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:ALEVel?
Response	6.000

Command Format	[:SOURce]:CURRent:TRANsient:BLEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the B Level of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:BLEVel 5.000

Command Format	[:SOURce]:CURRent:TRANsient:BLEVel?
Description	Query the B Level of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient:BLEVel?
Response	8.000

Command Format	[:SOURce]:CURRent:TRANsient:AWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level pulse width time value of CC mode in transient operation. Its unit is “s”
Example	:SOURce:CURRent:TRANsient:AWIDth 0.700

Command Format	[:SOURce]:CURRent:TRANsient:AWIDth?

Description	Query the A Level pulse width time value of CC mode in transient operation. Its unit is “s”
Example	:SOURce:CURRent:TRANsient:AWIDth?
Response	1.000

Command Format	[:SOURce]:CURRent:TRANsient:BWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the B Level pulse width time value of CC mode in transient operation. Its unit is “s”
Example	:SOURce:CURRent:TRANsient:BWIDth 0.800

Command Format	[:SOURce]:CURRent:TRANsient:BWIDth?
Description	Query the B Level pulse width time value of CC mode in transient operation. Its unit is “s”
Example	:SOURce:CURRent:TRANsient:BWIDth?
Response	1.500

Command Format	[:SOURce]:CURRent:TRANsient:SLEW:POSitive {<value> | MINimum | MAXimum | DEFault}
Description	Sets the rise slope of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient: SLEW:POSitive 0.400

Command Format	[:SOURce]:CURRent:TRANsient:SLEW:POSitive?
Description	Query the rise slope of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient: SLEW:POSitive?
Response	0.600

Command Format	[:SOURce]:CURRent:TRANsient:SLEW:NEGative {<value> | MINimum | MAXimum | DEFault}
Description	Sets the descending slope of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient: SLEW: NEGative 0.300

Command Format	[:SOURce]:CURRent:TRANsient:SLEW:NEGative?
Description	Query the descending slope of CC mode in transient operation.
Example	:SOURce:CURRent:TRANsient: SLEW: NEGative?
Response	0.900

3.3.3 Source Voltage Subsystem Command

Command Format	[:SOURce]:VOLTage[:LEVel][:IMMediate] {<value> | MINimum | MAXimum | DEFault}
Description	Sets the preset voltage value of CV mode in static operation.
Example	:SOURce:VOLTage:LEVel:IMMediate 3.000

Command Format	[:SOURce]:VOLTage[:LEVel][:IMMediate]?
Description	Query the preset voltage value of CV mode in static operation.
Example	:SOURce:VOLTage:LEVel:IMMediate?
Response	4.000

Command Format	[:SOURce]:VOLTage:IRANGe <value>
Description	Sets the current range of CV mode in static operation.
Example	:SOURce:VOLTage:IRANGe 10

Command Format	[:SOURce]:VOLTage:IRANGe?
Description	Query the current range of CV mode in static operation.
Example	:SOURce:VOLTage:IRANGe?
Response	30

Command Format	[:SOURce]:VOLTage:VRANGe <value>
Description	Sets the voltage range of CV mode in static operation.
Example	:SOURce:VOLTage:VRANGe 60

Command Format	[:SOURce]:VOLTage:VRANGe?
Description	Query the voltage range of CV mode in static operation.
Example	:SOURce:VOLTage:VRANGe?
Response	36

Command Format	[:SOURce]:VOLTage:TRANsient:MODE {CONTinuous | PULSe | TOGGle}
Description	Sets the waveform mode of CV mode in transient operation.
Example	:SOURce:VOLTage:TRANsient:MODE TOGGle

Command Format	[:SOURce]:VOLTage:TRANsient:MODE?
Description	Query the waveform mode of CV mode in static operation.
Example	:SOURce:VOLTage:TRANsient:MODE?
Response	PULSe

Command Format	[:SOURce]:VOLTage:TRANsient:IRANGe <value>
Description	Sets the current range of CV mode in transient operation.
Example	:SOURce:VOLTage:TRANsient:IRANGe 5

Command Format	[:SOURce]:VOLTage:TRANsient:IRANGe?
Description	Query the current range of CV mode in transient operation.
Example	:SOURce:VOLTage:TRANsient:IRANGe?
Response	30

Command Format	[:SOURce]:VOLTage:TRANsient:VRANGe <value>
Description	Sets the voltage range of CV mode in transient operation.
Example	:SOURce:VOLTage:TRANsient:VRANGe 36

Command Format	[:SOURce]:VOLTage:TRANsient:VRANGe?
Description	Query the voltage range of CV mode in transient operation.
Example	:SOURce:VOLTage:TRANsient:VRANGe?
Response	150

Command Format	[:SOURce]: VOLTage:TRANsient:ALEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level of CV mode in transient operation.
Example	:SOURce: VOLTage:TRANsient:ALEVel 4.653

Command Format	[:SOURce]: VOLTage:TRANsient:ALEVel?
Description	Query the A Level of CV mode in transient operation.
Example	:SOURce: VOLTage:TRANsient:ALEVel?
Response	8.000

Command	[:SOURce]:VOLTage:TRANsient:BLEVel {<value> | MINimum |

Format	MAXimum | DEFault}
Description	Sets the B Level of CV mode in transient operation.
Example	:SOURce:VOLTage:TRANsient:BLEVel MAXimum

Command Format	[:SOURce]: VOLTage:TRANsient:BLEVel?
Description	Query the B Level of CV mode in transient operation.
Example	:SOURce: VOLTage:TRANsient:BLEVel?
Response	10.000

Command Format	[:SOURce]:VOLTage:TRANsient:AWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level pulse width time value of CV mode in transient operation. Its unit is “s”
Example	:SOURce:VOLTage:TRANsient:AWIDth DEFault

Command Format	[:SOURce]:VOLTage:TRANsient:AWIDth?
Description	Query the A Level pulse width time value of CV mode in transient operation. Its unit is “s”
Example	:SOURce:VOLTage:TRANsient:AWIDth?
Response	1.000

Command Format	[:SOURce]:VOLTage:TRANsient:BWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the B Level pulse width time value of CV mode in transient operation. Its unit is “s”
Example	:SOURce:VOLTage:TRANsient:BWIDth 0.800

Command Format	[:SOURce]:VOLTage:TRANsient:BWIDth?
Description	Query the B Level pulse width time value of CV mode in transient operation. Its unit is “s”
Example	:SOURce:VOLTage:TRANsient:BWIDth?
Response	1.500

3.3.4 Source Power Subsystem Command

Command Format	[:SOURce]:POWer[:LEVel][:IMMediate] {<value> | MINimum | MAXimum | DEFault}

Description	Sets the preset power value of CP mode in static operation.
Example	:SOURce:POWer:LEVel:IMMediate 3.000

Command Format	[:SOURce]:POWer[:LEVel][:IMMediate]?
Description	Query the preset power value of CP mode in static operation.
Example	:SOURce:POWer:LEVel:IMMediate?
Response	4.000

Command Format	[:SOURce]:POWer:IRANGe <value>
Description	Sets the current range of CP mode in static operation.
Example	:SOURce:POWer:IRANGe 10

Command Format	[:SOURce]:POWer:IRANGe?
Description	Query the current range of CP mode in static operation.
Example	:SOURce:POWer:IRANGe?
Response	30

Command Format	[:SOURce]:POWer:VRANGe <value>
Description	Sets the voltage range of CP mode in static operation.
Example	:SOURce:POWer:VRANGe 60

Command Format	[:SOURce]:POWer:VRANGe?
Description	Query the voltage range of CP mode in static operation.
Example	:SOURce:POWer:VRANGe?
Response	30

Command Format	[:SOURce]:POWer:TRANsient:MODE {CONTinuous | PULSe | TOGGle}
Description	Sets the waveform mode of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:MODE TOGGle

Command Format	[:SOURce]:POWer:TRANsient:MODE?
Description	Query the waveform mode of CP mode in static operation.
Example	:SOURce:POWer:TRANsient:MODE?

Response	PULSe

Command Format	[:SOURce]:POWer:TRANsient:IRANGe <value>
Description	Sets the current range of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:IRANGe 5

Command Format	[:SOURce]:POWer:TRANsient:IRANGe?
Description	Query the current range of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:IRANGe?
Response	30

Command Format	[:SOURce]:POWer:TRANsient:VRANGe <value>
Description	Sets the voltage range of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:VRANGe 36

Command Format	[:SOURce]:POWer:TRANsient:VRANGe?
Description	Query the voltage range of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:VRANGe?
Response	150

Command Format	[:SOURce]:POWer:TRANsient:ALEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:ALEVel MINimum

Command Format	[:SOURce]:POWer:TRANsient:ALEVel?
Description	Query the A Level of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:ALEVel?
Response	8.000

Command Format	[:SOURce]:POWer:TRANsient:BLEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the B Level of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:BLEVel MAXimum

Command Format	[:SOURce]:POWer:TRANsient:BLEVel?
Description	Query the B Level of CP mode in transient operation.
Example	:SOURce:POWer:TRANsient:BLEVel?
Response	10.000

Command Format	[:SOURce]: POWer:TRANsient:AWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level pulse width time value of CP mode in transient operation. Its unit is “s”
Example	:SOURce: POWer:TRANsient:AWIDth 0.700

Command Format	[:SOURce]:POWer:TRANsient:AWIDth?
Description	Query the A Level pulse width time value of CP mode in transient operation. Its unit is “s”
Example	:SOURce:POWer:TRANsient:AWIDth?
Response	2.000

Command Format	[:SOURce]: POWer:TRANsient:BWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the B Level pulse width time value of CP mode in transient operation. Its unit is “s”
Example	:SOURce: POWer:TRANsient:BWIDth 0.800

Command Format	[:SOURce]:POWer:TRANsient:BWIDth?
Description	Query the B Level pulse width time value of CP mode in transient operation. Its unit is “s”
Example	:SOURce:POWer:TRANsient:BWIDth?
Response	1.500

3.3.5 Source Resistance Subsystem Command

Command Format	[:SOURce]: RESistance [:LEVel][:IMMediate] {<value> | MINimum | MAXimum | DEFault}
Description	Sets the preset resistor value of CR mode in static operation.
Example	:SOURce: RESistance:LEVel:IMMediate 3.000

Command Format	[:SOURce]:RESistance[:LEVel][:IMMediate]?
Description	Query the preset resistor value of CR mode in static operation.
Example	:SOURce:RESistance:LEVel:IMMediate?
Response	4.000

Command Format	[:SOURce]:RESistance:IRANGe <value>
Description	Sets the current range of CR mode in static operation.
Example	:SOURce:RESistance:IRANGe 10

Command Format	[:SOURce]:RESistance:IRANGe?
Description	Query the current range of CR mode in static operation.
Example	:SOURce:RESistance:IRANGe?
Response	30

Command Format	[:SOURce]:RESistance:IRANGe <value>
Description	Sets the voltage range of CR mode in static operation.
Example	:SOURce:RESistance:IRANGe 60

Command Format	[:SOURce]:RESistance:VRANGe?
Description	Query the voltage range of CR mode in static operation.
Example	:SOURce:RESistance:VRANGe?
Response	150

Command Format	[:SOURce]:RESistance:RRANGe {LOW | MIDDLE | HIGH | UPPER}
Description	Sets the resistor range of CR mode in static operation.
Example	:SOURce:RESistance:RRANGe MIDDLE

Command Format	[:SOURce]:RESistance:RRANGe?
Description	Query the resistor range of CR mode in static operation.
Example	:SOURce:RESistance:RRANGe?
Response	UPPER

Command	[:SOURce]:RESistance:TRANsient:MODE {CONTinuous |

Format	PULSe | TOGGle}
Description	Sets the waveform mode of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:MODE TOGGle

Command Format	[:SOURce]:RESistance:TRANsient:MODE?
Description	Query the waveform mode of CR mode in static operation.
Example	:SOURce:RESistance:TRANsient:MODE?
Response	PULSe

Command Format	[:SOURce]:RESistance:TRANsient:IRANGe <value>
Description	Sets the current range of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:IRANGe 5

Command Format	[:SOURce]:RESistance:TRANsient:IRANGe?
Description	Query the current range of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:IRANGe?
Response	30

Command Format	[:SOURce]:RESistance:TRANsient:VRANGe <value>
Description	Sets the voltage range of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:VRANGe 36

Command Format	[:SOURce]:RESistance:TRANsient:VRANGe?
Description	Query the voltage range of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:VRANGe?
Response	150

Command Format	[:SOURce]:RESistance:TRANsient:RRANGe {LOW | MIDDLE | HIGH | UPPER}
Description	Sets the resistor range of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:VRANGe HIGH

Command Format	[:SOURce]:RESistance:TRANsient:RRANGe?
Description	Query the resistor range of CR mode in transient operation.

Example	:SOURce:RESistance:TRANsient:VRANGe?
Response	LOW

Command Format	[:SOURce]:RESistance:TRANsient:ALEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:ALEVel MINimum

Command Format	[:SOURce]:RESistance:TRANsient:ALEVel?
Description	Query the A Level of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:ALEVel?
Response	8.000

Command Format	[:SOURce]:RESistance:TRANsient:BLEVel {<value> | MINimum | MAXimum | DEFault}
Description	Sets the B Level of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:BLEVel MINimum

Command Format	[:SOURce]:RESistance:TRANsient:BLEVel?
Description	Query the B Level of CR mode in transient operation.
Example	:SOURce:RESistance:TRANsient:BLEVel?
Response	10.000

Command Format	[:SOURce]:RESistance:TRANsient:AWIDth {<value> | MINimum | MAXimum | DEFault}
Description	Sets the A Level pulse width time value of CR mode in transient operation. Its unit is “s”.
Example	:SOURce:RESistance:TRANsient:AWIDth DEFault

Command Format	[:SOURce]:RESistance:TRANsient:AWIDth?
Description	Query the A Level pulse width time value of CR mode in transient operation. Its unit is “s”.
Example	:SOURce:RESistance:TRANsient:AWIDth?
Response	2.000

Command Format	[:SOURce]:RESistance:TRANsient:BWIDth {< value > | MINimum | MAXimum | DEFault}

Description	Sets the B Level pulse width time value of CR mode in transient operation. Its unit is “s”.
Example	:SOURce:RESistance:TRANsient:BWIDth 0.800

Command Format	[:SOURce]:RESistance:TRANsient:BWIDth?
Description	Query the B Level pulse width time value of CR mode in transient operation. Its unit is “s”.
Example	:SOURce:RESistance:TRANsient:BWIDth?
Response	1.500

3.3.6 Source LED Subsystem Command

Command Format	[:SOURce]:LED:IRANGe <value>
Description	Sets the current range of LED mode in static operation.
Example	:SOURce:LED:IRANGe 12

Command Format	[:SOURce]:LED:IRANGe?
Description	Query the current range of LED mode in static operation.
Example	:SOURce:LED:IRANGe?
Response	30

Command Format	[:SOURce]:LED:VRANGe <value>
Description	Sets the voltage range of LED mode in static operation.
Example	:SOURce:LED:VRANGe 40

Command Format	[:SOURce]:LED:VRANGe?
Description	Query the voltage range of LED mode in static operation.
Example	:SOURce:LED:VRANGe?
Response	150

Command Format	[:SOURce]:LED:VOLTage {< value > | MINimum | MAXimum | DEFault}
Description	Sets the “Vo” preset voltage of LED mode in static operation.
Example	:SOURce:LED: VOLTage 10

Command Format	[:SOURce]:LED:VOLTage?
Description	Query the “Vo” preset voltage of LED mode in static operation.
Example	:SOURce:LED: VOLTage?
Response	60.000

Command Format	[:SOURce]:LED:CURRent {< value > | MINimum | MAXimum | DEFault}
Description	Sets the “Io” preset current of LED mode in static operation.
Example	:SOURce:LED: CURRent 3

Command Format	[:SOURce]:LED:CURRent?
Description	Query the “Io” preset current of LED mode in static operation.
Example	:SOURce:LED: CURRent?
Response	2.000

Command Format	[:SOURce]:LED: RCOnf {< value > | MINimum | MAXimum | DEFault}
Description	Sets the “Rco” preset value of LED mode in static operation.
Example	:SOURce:LED: CURRent DEFault

Command Format	[:SOURce]:LED: RCOnf?
Description	Query the “Rco” preset value of LED mode in static operation.
Example	:SOURce:LED: RCOnf?
Response	0.200

3.3.7 Source Battery Subsystem Command

Command Format	[:SOURce]:BATTery:FUNC
Description	Enter the BATTERY function of the eclectronic load.
Example	:SOURce:BATTery:FUNC

Command Format	[:SOURce]:BATTery:FUNC?
Description	Query whether the electronic load is in BATTERY function.
Example	:SOURce:BATTery:FUNC?
Response	1

Command Format	[:SOURce]:BATTery:MODE {CURRent | POWer | RESistance}
Description	Sets CC/CP/CR mode in BATTERY function.
Example	:SOURce:BATTery:MODE POWer

Command Format	[:SOURce]:BATTery:MODE?
Description	Query the current mode in BATTERY function.
Example	:SOURce:BATTery:MODE?
Response	CURRENT

Command Format	[:SOURce]:BATTery:IRANGe <value>
Description	Sets the current range in BATTERY function.
Example	:SOURce:BATTery:IRANGe 6.000

Command Format	[:SOURce]:BATTery:IRANGe?
Description	Query the current range in BATTERY function.
Example	:SOURce:BATTery:IRANGe?
Response	30

Command Format	[:SOURce]:BATTery:VRANGe <value>
Description	Sets the voltage range in BATTERY function.
Example	:SOURce:BATTery:IRANGe 22.000

Command Format	[:SOURce]:BATTery:VRANGe?
Description	Query the voltage range in BATTERY function.
Example	:SOURce:BATTery: VRANGe?
Response	150

Command Format	[:SOURce]:BATTery:RRANGe {LOW | MIDDLE | HIGH | UPPER}
Description	Sets the resistor range in BATTERY function.
Example	:SOURce:BATTery:IRANGe UPPER

Command	[:SOURce]:BATTery:RRANGe?

Format	
Description	Query the resistor range in BATTERY function.
Example	:SOURce:BATTery:IRANGe?
Response	MIDDLE

Command Format	[:SOURce]:BATTery:LEVel <value>
Description	Sets the preset discharging value in BATTERY CC/CR/CP mode.
Example	:SOURce:BATTery:LEVel 5.000

Command Format	[:SOURce]:BATTery:LEVel?
Description	Query the discharging value in BATTERY CC/CR/CP mode.
Example	:SOURce:BATTery:LEVel?
Response	8.000

Command Format	[:SOURce]:BATTery:VOLTage {< value > | MINimum | MAXimum | DEFault}
Description	Sets the cut-off voltage value in BATTERY function.
Example	:SOURce:BATTery:VOLTage 9.000

Command Format	[:SOURce]:BATTery:VOLTage?
Description	Query the cut-off voltage value in BATTERY function.
Example	:SOURce:BATTery:VOLTage?
Response	20.000

Command Format	[:SOURce]:BATTery:CAPability < value >
Description	Sets the cut-off capacitance values in BATTERY function.
Example	:SOURce:BATTery: CAPability 100.00

Command Format	[:SOURce]:BATTery:CAPability?
Description	Query the cut-off capacitance values in BATTERY function.
Example	:SOURce:BATTery: CAPability?
Response	199.000

Command Format	[:SOURce]:BATTery:TIMer {< value > | MINimum | MAXimum | DEFault}

Description	Sets the cut-off discharging time value in BATTERY function.
Example	:SOURce:BATTery:TIMer DEFault

Command Format	[:SOURce]:BATTery:TIMer?
Description	Query the cut-off discharging time value in BATTERY function.
Example	:SOURce:BATTery:TIMer?
Response	999.0000

Command Format	[:SOURce]:BATTery:VOLTage:STATe {ON | OFF | 0 | 1}
Description	Sets whether use the cut-off voltage as the terminating condiation in BATTERY function.
Example	:SOURce:BATTery:VOLTage:STATe ON

Command Format	[:SOURce]:BATTery:VOLTage:STATe?
Description	Query whether the cut-off voltage is the terminating condiation in BATTERY function.
Example	:SOURce:BATTery:VOLTage:STATe?
Response	0

Command Format	[:SOURce]:BATTery:CAPability:STATe {ON | OFF | 0 | 1}
Description	Sets whether use the cut-off capacitance as the terminating condiation in BATTERY function.
Example	:SOURce:BATTery: CAPability:STATe OFF

Command Format	[:SOURce]:BATTery:CAPability:STATe?
Description	Query whether the cut-off capacitance is the terminating condiation in BATTERY function.
Example	:SOURce:BATTery: CAPability:STATe?
Response	1

Command Format	[:SOURce]:BATTery:TIMer:STATe{ON | OFF | 0 | 1}
Description	Sets whether use the discharging time as the terminating condiation in BATTERY function.
Example	:SOURce:BATTery: TIMer:STATe ON

Command Format	[:SOURce]:BATTery:TIMer:STATe?
Description	Query whether the discharging time is the terminating condiation in BATTERY function.
Example	:SOURce:BATTery: TIMer:STATe?
Response	1

Command Format	[:SOURce]:BATTery:DISCHArg:CAPability?
Description	Gets the discharging capacity after user start the BATTERY test.
Example	:SOURce:BATTery:DISCHArg:CAPability?
Response	13

Command Format	[:SOURce]:BATTery:DISCHArg:TIMer?
Description	Gets the discharging time after user start the BATTERY test.
Example	:SOURce:BATTery:DISCHArg: TIMer?
Response	162

3.3.8 Source List Subsystem Command

Command Format	[:SOURce]:LIST:MODE {CURRent | VOLTage | POWer | RESistance}
Description	Sets the run mode in LIST function.
Example	:SOURce:LIST:MODE VOLTage

Command Format	[:SOURce]:LIST:MODE?
Description	Query the run mode in LIST function.
Example	:SOURce:LIST:MODE?
Response	CURRENT

Command Format	[:SOURce]:LIST:IRANGe <value>
Description	Sets the current range in LIST function.
Example	:SOURce:LIST:IRANGe 5

Command Format	[:SOURce]:LIST:IRANGe?

Description	Query the current range in LIST function.
Example	:SOURce:LIST:IRANGe?
Response	30

Command Format	[:SOURce]:LIST:VRANGe <value>
Description	Sets the voltage range in LIST function.
Example	:SOURce:LIST:VRANGe 150

Command Format	[:SOURce]:LIST:VRANGe?
Description	Query the voltage range in LIST function.
Example	:SOURce:LIST:VRANGe?
Response	36

Command Format	[:SOURce]:LIST:RRANGe {LOW | MIDDLE | HIGH |UPPER}
Description	Sets the resistor range in LIST function.
Example	SOURce:LIST:RRANGe HIGH

Command Format	[:SOURce]:LIST:RRANGe?
Description	Query the resistor range in LIST function.
Example	:SOURce:LIST:RRANGe?
Response	UPPER

Command Format	[:SOURce]:LIST:COUNt {< number > | MINimum | MAXimum | DEFault}
Description	Sets the number of running loops in LIST function.
Example	:SOURce:LIST:COUNt 10

Command Format	[:SOURce]:LIST:COUNt?
Description	Query the number of running loops in LIST function.
Example	:SOURce:LIST:COUNt?
Response	255

Command Format	[:SOURce]:LIST:STEP {< number > | MINimum | MAXimum | DEFault}

Description	Sets the execution of steps in LIST function.
Example	:SOURce:LIST: STEP 5

Command Format	[:SOURce]:LIST:STEP?
Description	Query the execution of steps in LIST function.
Example	:SOURce:LIST:STEP?
Response	15

Command Format	[:SOURce]:LIST:LEVel <step,value>
Description	Sets the set value of the step which is set in this command in LIST sequence.
Example	:SOURce:LIST:LEVel 3,4.500

Command Format	[:SOURce]:LIST:LEVel? <step>
Description	Query the set value of the step which is set in this command in LIST sequence.
Example	[:SOURce]:LIST:LEVel? 5
Response	2.000

Command Format	[:SOURce]:LIST:SLEW[:BOTH] <step,value>
Description	Sets the slope of the step which is set in this command in LIST CC mode.
Example	:SOURce:LIST:SLEW:BOTH 4,0.500

Command Format	[:SOURce]:LIST:SLEW[:BOTH]? <step>
Description	Query the slope of the step which is set in this command in LIST CC mode.
Example	:SOURce:LIST:SLEW:BOTH? 5
Response	0.100

Command Format	[:SOURce]:LIST:WIDth <step,value>
Description	Sets the run time of the step which is set in this command in LIST sequence.
Example	:SOURce:LIST:WIDth 3,1.000

Command Format	[:SOURce]:LIST:WIDth? <step>
Description	Query the run time of the step which is set in this command in LIST sequence.
Example	:SOURce:LIST:WIDth? 5
Response	0.100

Command Format	[:SOURce]:LIST:STATe:ON
Description	Enter the LIST function of the eclectronic load.
Example	:SOURce:LIST:STATe:ON

Command Format	[:SOURce]:LIST:STATe?
Description	Query whether the electronic load is in LIST test mode.
Example	:SOURce:LIST:STATe?
Response	0.100

3.3.9 Source OCPT Subsystem Command

Command Format	[:SOURce]: OCP:FUNC
Description	Enter the OCPT function of the eclectronic load.
Example	:SOURce: OCP:FUNC

Command Format	[:SOURce]: OCP:FUNC?
Description	Query whether the electronic load is in OCPT test mode.
Example	:SOURce: OCP:FUNC?
Response	0

Command Format	[:SOURce]:OCP:IRANGe <value>
Description	Sets the current range in OCPT function.
Example	:SOURce:OCP:IRANGe 30

Command Format	[:SOURce]:OCP:IRANGe?
Description	Query the current range in OCPT function.

Example	:SOURce:OCP:IRANGe?
Response	5

Command Format	[:SOURce]:OCP:VRANGe <value>
Description	Sets the voltage range in OCPT function.
Example	:SOURce:OCP:VRANGe 36

Command Format	[:SOURce]:OCP:VRANGe?
Description	Query the voltage range in OCPT function.
Example	:SOURce:OCP:VRANGe?
Response	150

Command Format	[:SOURce]:OCP:STARt {< value > | MINimum | MAXimum | DEFault}
Description	Sets the current value when the load starts in OCPT test.
Example	:SOURce:OCP:STARt 1.000

Command Format	[:SOURce]:OCP:STARt?
Description	Query the current value when the load starts in OCPT test.
Example	:SOURce:OCP:STARt?
Response	1.000

Command Format	[:SOURce]:OCP:STEP {< value > | MINimum | MAXimum | DEFault}
Description	Sets the step current value in OCPT function.
Example	:SOURce:OCP: STEP 0.500

Command Format	[:SOURce]:OCP:STEP?
Description	Query the step current value in OCPT function.
Example	:SOURce:OCP: STEP?
Response	1.000

Command Format	[:SOURce]:OCP:STEP:DELay {< value > | MINimum | MAXimum | DEFault}
Description	Sets the delay time of each step in OCPT function.
Example	:SOURce:OCP:STEP:DELay 1.000

Command Format	[:SOURce]:OCP:STEP:DELay?
Description	Query the delay time of each step in OCPT function.
Example	:SOURce:OCP:STEP:DELay?
Response	2.000

Command Format	[:SOURce]:OCP:END {< value > | MINimum | MAXimum | DEFault}
Description	Sets the stop current in OCPT function.
Example	:SOURce:OCP: END 5.000

Command Format	[:SOURce]:OCP:END?
Description	Query the stop current in OCPT function.
Example	:SOURce:OCP: END?
Response	4.000

Command Format	[:SOURce]:OCP:MIN {< value > | MINimum | MAXimum | DEFault}
Description	Sets the minimum value of the protection current in OCPT function.
Example	:SOURce:OCP: MIN 2.000

Command Format	[:SOURce]:OCP:MIN?
Description	Query the minimum value of the protection current in OCPT function.
Example	:SOURce:OCP: MIN?
Response	3.000

Command Format	[:SOURce]:OCP:MAX {< value > | MINimum | MAXimum | DEFault}
Description	Sets the maximum value of the protection current in OCPT function.
Example	:SOURce:OCP: MAX 10.000

Command Format	[:SOURce]:OCP:MAX?
Description	Query the maximum value of the protection current in OCPT

	function.
Example	:SOURce:OCP: MAX?
Response	8.000

Command Format	[:SOURce]:OCP:VOLTage {< value > | MINimum | MAXimum | DEFault}
Description	Sets the protection voltage in OCPT function.
Example	:SOURce:OCP: VOLTage 8.000

Command Format	[:SOURce]:OCP:VOLTage?
Description	Query the value of the protection voltage in OCPT function.
Example	:SOURce:OCP: VOLTage?
Response	10.000

3.3.10 Source OPPT Subsystem Command

Command Format	[:SOURce]: OPP:FUNC
Description	Enter the OPPT function of the eclectronic load.
Example	:SOURce: OPP:FUNC

Command Format	[:SOURce]: OPP:FUNC?
Description	Query whether the electronic load is in OPPT test mode.
Example	:SOURce: OPP:FUNC?
Response	1

Command Format	[:SOURce]:OPP:IRANGe <value>
Description	Sets the current range in OPPT function.
Example	:SOURce:OPP:IRANGe 30

Command Format	[:SOURce]:OPP:IRANGe?
Description	Query the current range in OPPT function.
Example	:SOURce:OPP:IRANGe?
Response	5

Command Format	[:SOURce]:OPP:VRANGe <value>
Description	Sets the voltage range in OPPT function.
Example	:SOURce:OPP:VRANGe 36

Command Format	[:SOURce]:OPP:VRANGe?
Description	Query the voltage range in OPPT function.
Example	:SOURce:OPP:VRANGe?
Response	150

Command Format	[:SOURce]:OPP:STARt {< value > | MINimum | MAXimum | DEFault}
Description	Sets the power value when the load starts in OPPT test.
Example	:SOURce:OPP:STARt 1.000

Command Format	[:SOURce]:OPP:STARt?
Description	Query the power value when the load starts in OPPT test.
Example	:SOURce:OPP:STARt?
Response	1.000

Command Format	[:SOURce]:OPP:STEP {< value > | MINimum | MAXimum | DEFault}
Description	Sets the step power value in OPPT function.
Example	:SOURce:OPP: STEP 0.500

Command Format	[:SOURce]:OPP:STEP?
Description	Query the step power value in OPPT function.
Example	:SOURce:OPP: STEP?
Response	1.000

Command Format	[:SOURce]:OPP:STEP:DELay {< value > | MINimum | MAXimum | DEFault}
Description	Sets the delay time of each step in OPPT function.
Example	:SOURce:OPP:STEP:DELay 1.000

Command Format	[:SOURce]:OPP:STEP:DELay?
Description	Query the delay time of each step in OPPT function.
Example	:SOURce:OPP:STEP:DELay?
Response	2.000

Command Format	[:SOURce]:OPP:END {< value > | MINimum | MAXimum | DEFault}
Description	Sets the stop power value in OCPT function.
Example	:SOURce:OPP: END 5.000

Command Format	[:SOURce]:OPP:END?
Description	Query the stop power value in OPPT function.
Example	:SOURce:OPP: END?
Response	4.000

Command Format	[:SOURce]:OPP:MIN {< value > | MINimum | MAXimum | DEFault}
Description	Sets the minimum value of the protection power in OPPT function.
Example	:SOURce:OPP: MIN 2.000

Command Format	[:SOURce]:OPP:MIN?
Description	Query the minimum value of the protection power in OPPT function.
Example	:SOURce:OPP: MIN?
Response	3.000

Command Format	[:SOURce]:OPP:MAX {< value > | MINimum | MAXimum | DEFault}
Description	Sets the maximum value of the protection power in OPPT function.
Example	:SOURce:OPP: MAX 10.000

Command Format	[:SOURce]:OPP:MAX?
Description	Query the maximum value of the protection power in OPPT function.
Example	:SOURce:OPP: MAX?
Response	8.000

Command Format	[:SOURce]:OPP:VOLTage {< value > | MINimum | MAXimum | DEFault}
Description	Sets the protection voltage in OPPT function.
Example	:SOURce:OPP: VOLTage 8.000

Command Format	[:SOURce]:OPP:VOLTage?
Description	Query the value of the protection voltage in OPPT function.
Example	:SOURce:OPP: VOLTage?
Response	10.000

3.3.11 Source Program Subsystem Command

Command Format	[:SOURce]:PROGram:STEP {< number > | MINimum | MAXimum | DEFault}
Description	Sets the execution of steps in PROGRAM function.
Example	:SOURce:PROGram:STEP 8

Command Format	[:SOURce]:PROGram:STEP?
Description	Query the execution of steps in PROGRAM function.
Example	:SOURce:PROGram:STEP?
Response	10

Command Format	[:SOURce]:PROGram:MODE <step, {CURRent | VOLTage | POWer | RESistance | LED}>
Description	Sets the mode of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:MODE 2,VOLTage

Command Format	[:SOURce]:PROGram:MODE? <step>
Description	Query the mode of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:MODE? 3
Response	2.000

Command	[:SOURce]:PROGram:IRANGe <step, value>

Format	
Description	Sets the current range of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:IRANGe 4,5

Command Format	[:SOURce]:PROGram:IRANGe? <step>
Description	Query the current range of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:IRANGe? 5
Response	30

Command Format	[:SOURce]:PROGram:VRANGe <step, value>
Description	Sets the voltage range of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:VRANGe 3,36

Command Format	[:SOURce]:PROGram:VRANGe? <step>
Description	Query the voltage range of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:VRANGe? 2
Response	150

Command Format	[:SOURce]:PROGram:RRANGe <step, {LOW | MIDDLE | HIGH | UPPER}>
Description	Sets the resistor range of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram: RRANGe 3,LOW

Command Format	[:SOURce]:PROGram:RRANGe? <step>
Description	Query the resistor range of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram: RRANGe? 3
Response	MIDDLE

Command Format	[:SOURce]:PROGram:SHORt <step, {ON | OFF | 0 | 1 }>

Description	Whether set the step of the electronic load which is set in this command to be short circuit in PROGRAM test list.
Example	:SOURce:PROGram: SHORt 3,ON

Command Format	[:SOURce]:PROGram:SHORt? <step>
Description	Query Whether the step of the electronic load which is set in this command is short circuit in PROGRAM test list.
Example	:SOURce:PROGram: SHORt? 4
Response	0

Command Format	[:SOURce]:PROGram:PAUSE <step, {ON | OFF | 0 | 1 }>
Description	Whether pause the step of the electronic load which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram: PAUSE 2,1

Command Format	[:SOURce]:PROGram:PAUSE? <step>
Description	Query Whether the step of the electronic load which is set in this command is paused in PROGRAM test list.
Example	:SOURce:PROGram: PAUSE? 5
Response	0

Command Format	[:SOURce]:PROGram:TIME:ON <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets the loading time of the step which is set in this command in PROGRAM test list. Its unit is “s”
Example	:SOURce:PROGram:TIME:ON 4,1.000

Command Format	[:SOURce]:PROGram:TIME:ON? <step>
Description	Query the loading time of the step which is set in this command in PROGRAM test list. Its unit is “s”
Example	:SOURce:PROGram:TIME:ON? 5
Response	0.500

Command Format	[:SOURce]:PROGram:TIME:OFF <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets the unloading time of the step which is set in this command in

	PROGRAM test list. Its unit is “s”
Example	:SOURce:PROGram:TIME:OFF 4,0.500

Command Format	[:SOURce]:PROGram:TIME:OFF? <step>
Description	Query the unloading time of the step which is set in this command in PROGRAM test list. Its unit is “s”
Example	:SOURce:PROGram:TIME:OFF? 6
Response	2.000

Command Format	[:SOURce]:PROGram:TIME:DELay <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets test delay time of the step which is set in this command in PROGRAM test list. Its unit is “s”
Example	:SOURce:PROGram:TIME: DELay 3,0.500

Command Format	[:SOURce]:PROGram:TIME:DELay? <step>
Description	Query test delay time of the step which is set in this command in PROGRAM test list. Its unit is “s”
Example	:SOURce:PROGram:TIME: DELay? 1
Response	0.600

Command Format	[:SOURce]:PROGram:MIN <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets the minimum allowed value of the step which is set in this command in PROGRAM test list. The value is current value in CV and is voltage value in CC/CR/CP/LED.
Example	:SOURce:PROGram:MIN 3,1.000

Command Format	[:SOURce]:PROGram:MIN? <step>
Description	Query the minimum allowed value of the step which is set in this command in PROGRAM test list. The value is current value in CV and is voltage value in CC/CR/CP/LED.
Example	:SOURce:PROGram:MIN? 5
Response	2.000

Command Format	[:SOURce]:PROGram:MAX <step, {< value > | MINimum | MAXimum | DEFault}>

Description	Sets the maximum allowed value of the step which is set in this command in PROGRAM test list. The value is current value in CV and is voltage value in CC/CR/CP/LED.
Example	:SOURce:PROGram: MAX 2,10.000

Command Format	[:SOURce]:PROGram:MAX? <step>
Description	Query the minimum allowed value of the step which is set in this command in PROGRAM test list. The value is current value in CV and is voltage value in CC/CR/CP/LED.
Example	:SOURce:PROGram:MAX? 3
Response	15.000

Command Format	[:SOURce]:PROGram:LEVel <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets the sink value of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:LEVel 6,7.000

Command Format	[:SOURce]:PROGram:LEVel? <step>
Description	Query the set value of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:LEVel? 2
Response	6.000

Command Format	[:SOURce]:PROGram:LED:CURRent <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets the “Io” value of the step which is set in this command in PROGRAM function when the step is in LED mode.
Example	:SOURce:PROGram: LED:CURRent 6,7.000

Command Format	[:SOURce]:PROGram:LED:CURRent? <step>
Description	Query the “Io” value of the step which is set in this command in PROGRAM function when the step is in LED mode.
Example	:SOURce:PROGram: LED:CURRent? 4
Response	1.000

Command Format	[:SOURce]:PROGram:LED:RCOnf <step, {< value > | MINimum | MAXimum | DEFault}>
Description	Sets the “Rco” value of the step which is set in this command in PROGRAM function when the step is in LED mode.
Example	:SOURce:PROGram: LED: RCOnf 6,0.300

Command Format	[:SOURce]:PROGram:LED:RCOnf? <step>
Description	Query the “Rco” value of the step which is set in this command in PROGRAM function when the step is in LED mode.
Example	:SOURce:PROGram: LED: RCOnf? 5
Response	0.200

Command Format	[:SOURce]:PROGram:STATe:ON
Description	Enter the PROGRAM test mode of the eclectronic load .
Example	:SOURce:PROGram:STATe:ON

Command Format	[:SOURce]:PROGram:STATe?
Description	Query whether the load is in PROGRAM test mode.
Example	:SOURce:PROGram:STATe?
Response	0

Command Format	[:SOURce]:PROGram:TEST? <step>
Description	Query the test rusult of the step which is set in this command in PROGRAM test list.
Example	:SOURce:PROGram:TEST? 3
Response	3.584720

3.3.12 Source Wave Subsystem Command

Command Format	[:SOURce]:WAVE:TIME < number >
Description	Sets the window time in Waveform Display Function. Its unit is “s”
Example	:SOURce:WAVE:TIME 8

Command Format	[:SOURce]:WAVE:TIME?
Description	Query the window time in Waveform Display Function. Its unit is “s”
Example	:SOURce:WAVE:TIME?
Response	3600

Command Format	[:SOURce]:WAVE:MODE {CURRent | VOLTage | POWer | RESistance }
Description	Sets different data type include I, U, R and P displayed in Waveform Display Function.
Example	:SOURce]:WAVE:MODE CURRent

Command Format	[:SOURce]:WAVE:MODE?
Description	Query the displayed data type in Waveform Display Function.
Example	:SOURce]:WAVE:MODE?
Response	VOLTAGE

Command Format	[:SOURce]:WAVE:PAUSE {ON | OFF | 0 | 1}
Description	Sets whether pause the waveform displayed in the Waveform Display Function.
Example	:SOURce:WAVE:PAUSE ON

Command Format	[:SOURce]:WAVE:PAUSE?
Description	Query whether the waveform displayed in the Waveform Display Function is paused.
Example	:SOURce:WAVE:PAUSE?
Response	1

Command Format	[:SOURce]:WAVE:DISPlay {ON|OFF|0|1}
Description	Enter the Waveform Display Function of the electronic load.
Example	:SOURce:WAVE:DISPlay ON

Command Format	[:SOURce]:WAVE:DISPlay?
Description	Query whether the electronic load is in Waveform Display Function.
Example	:SOURce:WAVE:DISPlay?
Response	ON

3.3.13 Source Utility Subsystem Command

Command Format	[:SOURce]:VOLTage[:LEVel]:ON <value>
Description	Sets the breakover voltage of electronic load.
Example	:SOURce:VOLTage:LEVel:ON 6.000

Command Format	[:SOURce]:VOLTage[:LEVel]:ON?
Description	Query the value of the breakover voltage of electronic load.
Example	:SOURce:VOLTage:LEVel:ON?
Response	4.000

Command Format	[:SOURce]:VOLTage:LATCh[:STATe] {ON | OFF | 0 | 1}
Description	Sets whether enable the Von Latch switch of the electronic load.
Example	:SOURce:VOLTage:LATCh:STATe OFF

Command Format	[:SOURce]:VOLTage:LATCh[:STATe]?
Description	Query whether the Von Latch switch of the electronic load is enabled.
Example	:SOURce:VOLTage:LATCh:STATe?
Response	0

Command Format	[:SOURce]:EXT:INPUT[:StATe] {ON | OFF | 0 | 1}
Description	Sets whether enable the external control switch on the rear panel of the electronic load.
Example	:SOURce: EXT:INPUT:STATe OFF

Command Format	[:SOURce]: EXT:INPUT [:STATe]?
Description	Query whether the external control switch of the electronic load is enabled.
Example	:SOURce: EXT:INPUT:STATe?
Response	0

Command Format	[:SOURce]:CURRent:PROTection:STATe {ON | OFF | 0 | 1}
Description	Sets whether enable the current protection switch of the electronic load.
Example	:SOURce:CURRent:PROTection:STATe ON

Command Format	[:SOURce]: EXT:INPUT [:STATe]?
Description	Query whether the current protection switch of the electronic load is enabled.
Example	:SOURce: EXT:INPUT:STATe?
Response	0

Command Format	[:SOURce]:CURRent:PROTection:LEVel {< value > | MINimum | MAXimum | DEFault}
Description	Sets the threhold value of the current protection of the electronic load after enable the current protection function.
Example	:SOURce:CURRent:PROTection:LEVel 7.00

Command Format	[:SOURce]:CURRent:PROTection:LEVel?
Description	Query the threhold value of the current protection of the electronic load.
Example	:SOURce:CURRent:PROTection:LEVel?
Response	8.000

Command Format	[:SOURce]:CURRent:PROTection:DELay {< value > | MINimum | MAXimum | DEFault}
Description	Sets the delay time of the current protection of the electronic load .
Example	:SOURce:CURRent:PROTection:DELay 2.00

Command Format	[:SOURce]:CURRent:PROTection:DELay?
Description	Query the delay time of the current protection of the electronic load.
Example	:SOURce:CURRent:PROTection:DELay?
Response	3.000

Command Format	[:SOURce]:POWer:PROTection:STATe {ON | OFF | 0 | 1}
Description	Sets whether enable the power protection switch of the electronic load.
Example	:SOURce:POWer:PROTection:STATe ON

Command Format	[:SOURce]:POWer:PROTection:STATe?
Description	Query whether the power protection switch of the electronic load is enabled.
Example	:SOURce:POWer:PROTection:STATe?
Response	1

Command Format	[:SOURce]:POWer:PROTection:LEVel {< value > | MINimum | MAXimum | DEFault}
Description	Sets the threhold value of the power protection of the electronic load after enable the power protection function.
Example	:SOURce:POWer:PROTection:LEVel 7.00

Command Format	[:SOURce]:POWer:PROTection:LEVel?
Description	Query the threhold value of the power protection of the electronic load.
Example	:SOURce:POWer:PROTection:LEVel?
Response	8.000

Command Format	[:SOURce]:POWer:PROTection:DELay {< value > | MINimum | MAXimum | DEFault}
Description	Sets the delay time of the power protection of the electronic load after enable the power protection function.
Example	:SOURce:POWer:PROTection:DELay 2.00

Command Format	[:SOURce]:POWer:PROTection:DELay?
Description	Query the delay time of the power protection of the electronic load.
Example	:SOURce:POWer:PROTection:DELay?
Response	3.000

3.4 Subsystem Command

Command Format	SYSTem:SENSe[:STATe] {ON | OFF | 0 | 1}
Description	Sets whether enable the Sense function switch of the electronic load.
Example	SYSTem:SENSe:STATe OFF

Command Format	SYSTem:SENSe[:STATe]?
Description	Query whether the Sense function switch of the electronic load is enabled.
Example	SYSTem:SENSe:STATe?
Response	1

Command Format	SYSTem:IMONItor[:STATe] {ON | OFF | 0 | 1}
Description	Sets whether enable the current monitoring terminal switch of the electronic load.
Example	SYSTem:IMONItor:STATe OFF

Command Format	SYSTem: IMONItor [:STATe]?
Description	Query whether the current monitoring terminal switch of the electronic load is enabled.
Example	SYSTem: IMONItor:STATe?
Response	1

Command Format	SYSTem:VMONItor[:STATe] {ON | OFF | 0 | 1}

Description	Sets whether enable the voltage monitoring terminal switch of the electronic load.
Example	SYSTem:VMONItor:STATe OFF

Command Format	SYSTem: VMONItor [:STATe]?
Description	Query whether the voltage monitoring terminal switch of the electronic load is enabled.
Example	SYSTem: VMONItor:STATe?
Response	1

Command Format	STOP:ON:FAIL[:STATe] {ON | OFF | 0 | 1}
Description	Sets whether stop the PROGRAM test when test step is failed.
Example	STOP:ON:FAIL:STATe OFF

Command Format	STOP:ON:FAIL[:STATe]?
Description	Query whether the SOF switch is enabled.
Example	STOP:ON:FAIL:STATe?
Response	1

Command Format	*TRG
Description	Generate a trigger in the electronic load.
Example	*TRG

Command Format	TRIGger:SOURce {MANUal | EXTernal | BUS}
Description	Sets the trigger source of the electronic load.
Example	TRIGger:SOURce BUS

Command Format	TRIGger:SOURce?
Description	Query the trigger source of the electronic load.
Example	TRIGger:SOURce?
Response	MANUAL

Command Format	SENSe:AVERage:COUNt {6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14}
Description	Sets the average point number of the read-back current and voltage of the electronic load. The value is the index of “2”.
Example	SENSe:AVERage:COUNt 10

Command Format	SENSe:AVERage:COUNt?
Description	Query the average point number of the read-back current and voltage of the electronic load. The value is the index of “2”
Example	SENSe:AVERage:COUNt?
Response	9

Command Format	EXT:MODE {INT | EXTI | EXTV}
Description	Sets the sink mode include external or internal mode of the electronic load.
Example	EXT:MODE INT

Command Format	EXT:MODE?
Description	Query the sink mode include external or internal mode of the electronic load.
Example	EXT:MODE?
Response	EXTV

Command Format	EXT:IRANGe <value>
Description	Sets the current range in external sink mode of the electronic load.
Example	EXT:IRANGe 5

Command Format	EXT:IRANGe?
Description	Query the current range in external sink mode of the electronic load.
Example	EXT:IRANGe?
Response	30

Command Format	EXT:VRANGe <value>
Description	Sets the voltage range in external sink mode of the electronic load.
Example	EXT:VRANGe 36

Command Format	EXT:VRANGe?
Description	Query the voltage range in external sink mode of the electronic load.
Example	EXT:VRANGe?
Response	150

Command Format	TIME:TEST[:STATe] {ON | OFF | 0 | 1}
Description	Sets whether enable the time measurement switch.
Example	TIME:TEST:STATe OFF

Command Format	TIME:TEST[:STATe]?
Description	Query whether the time measurement switch is enabled.
Example	TIME:TEST:STATe?
Response	0

Command Format	TIME:TEST:VOLTage:LOW {< value > | MINimum | MAXimum | DEFault}
Description	Sets the V_Low voltage in the time measurement function (SLMT).
Example	TIME:TEST:VOLTage:LOW 1.000

Command Format	TIME:TEST:VOLTage:LOW?
Description	Query the V_Low voltage in the time measurement function (SLMT).
Example	TIME:TEST:VOLTage:LOW?
Response	3.00

Command Format	TIME:TEST:VOLTage:HIGH {< value > | MINimum | MAXimum | DEFault}
Description	Sets the V_High voltage in the time measurement function (SLMT).
Example	TIME:TEST:VOLTage: HIGH 10.000

Command Format	TIME:TEST:VOLTage:HIGH?
Description	Query the V_High voltage in the time measurement function (SLMT).
Example	TIME:TEST:VOLTage: HIGH?
Response	150.00

Command Format	TIME:TEST:RISE?
Description	Query the voltage rise time in the time measurement function (SLMT).
Example	TIME:TEST:RISE?
Response	0.020

Command Format	TIME:TEST:FALL?
Description	Query the voltage descending time in the time measurement function (SLMT).
Example	TIME:TEST: FALL?
Response	0.030

3.5 LAN Interface Subsystem Command

Command Format	LAN:LINK?
Description	Query whether the LAN interface of the electronic load had been connect to the network.
Example	LAN:LINK?
Response	0

Command Format	DHCP {ON | OFF | 0 | 1}
Description	Sets whether enable the DHCP switch of the electronic load.
Example	DHCP ON

Command Format	DHCP?
Description	Query whether the DHCP switch of the electronic load is enabled
Example	DHCP?
Response	1

Command Format	LAN:IPADdress <aaa.bbb.ccc.ddd>
Description	Sets IP address of the electronic load when DHCP is disabled.
Example	LAN:IPADdress 10.12.15.64

Command Format	LAN:IPADdress?
Description	Query the IP address of the electronic load.
Example	LAN:IPADdress?
Response	10.11.13.76

Command Format	LAN:SMASk <aaa.bbb.ccc.ddd>
Description	Sets the subnet mask of the electronic load when DHCP is disabled.
Example	LAN: SMASk 255.255.255.255

Command Format	LAN:SMASk?
Description	Sets the subnet mask of the electronic load.
Example	LAN: SMASk?
Response	255.255.255.0

Command Format	LAN:GATeway <aaa.bbb.ccc.ddd>

Description	Sets the gateway of the electronic load when DHCP is disabled.
Example	LAN: GATeway 10.11.13.1

Command Format	LAN:GATeway?
Description	Query the subnet mask of the electronic load.
Example	LAN: GATeway?
Response	10.12.16.1

Command Format	LAN:MAC?
Description	Query the MAC address of the electronic load.
Example	LAN:MAC?
Response	00.80.e1.00.00.00

4. Programming Examples

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

4.1 Examples of Using VISA

4.1.1 Example of VC++

Environment: Win7 32bit system, Visual Studio

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.

Follow the steps to finish the example:

1、 Open Visual Studio, create a new VC++ win32 console project.

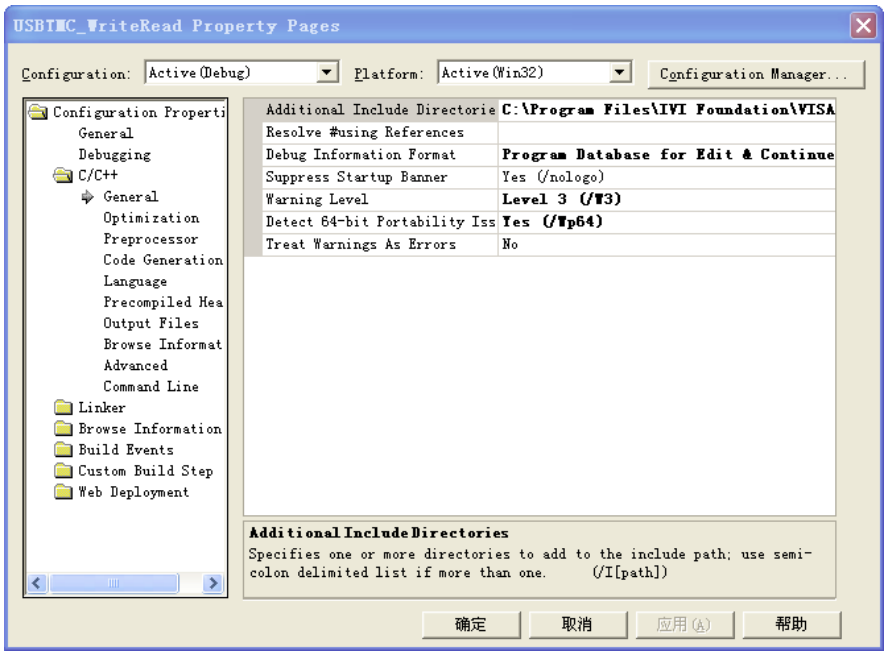
2、 Set the project environment to use the NI-VISA lib, there are two ways to use NI-VISA, static or automatic:

(1) Static: find files: visa.h, visatype.h, visa32.lib in NI-VISA install path. Copy them to your project, and add them into project. In the projectname.cpp file, add the follow two lines:

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

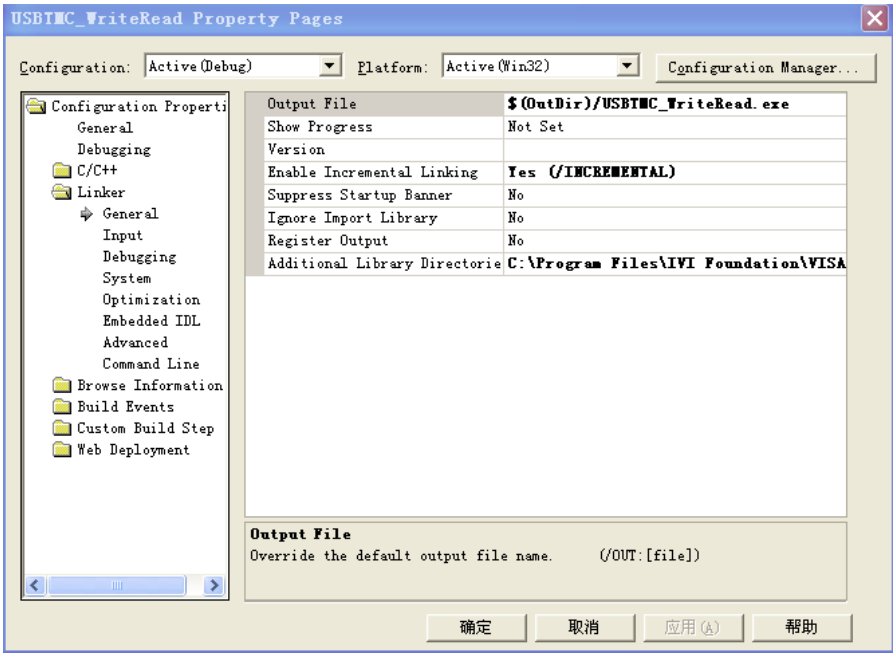
(2) Automatic:

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.

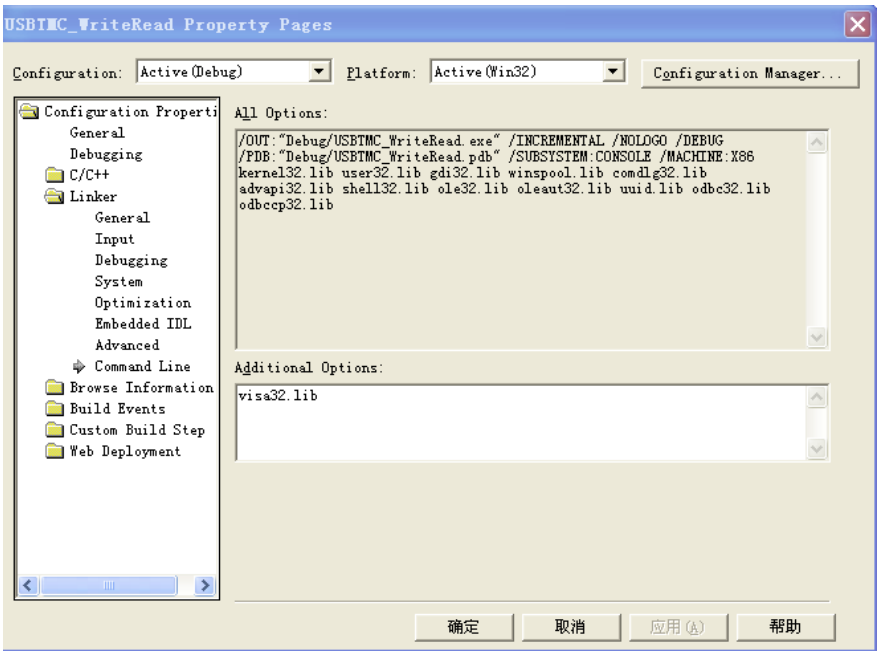


Set lib path set lib file:

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 seen 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、Add codes:

(1)USBTMC access code.

Write a function Usbtmc_test:

```
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 */
```

```
/*************************************************************/
ViSessiondefaultRM;
ViSessioninstr;
ViUInt32numInstrs;
ViFindListfindList;
ViStatus status;
char instrResourceString[VI_FIND_BUFLEN];
unsigned char buffer[100];
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.\nPress '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<int(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, 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;
}
int _tmain(int argc, _TCHAR* argv[])
{
  Usbtmc_test();
  return 0;
}
```

(2)TCP/IP access code:

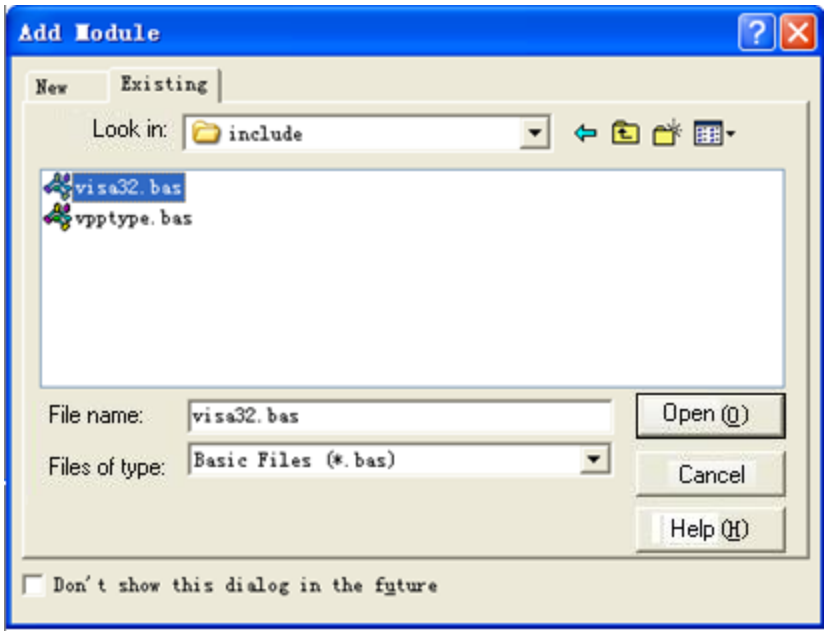
Write a function TCP_IP_Test.

```
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 */
  char head[256] ="TCPIP0::";
  char tail[] ="::INSTR";
  char resource [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);
    return 0;
}
```

4.1.2 Example of VB

Environment: Win7 32bit system, Microsoft Visual Basic 6.0
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.
Follow the steps to complete the example:
1、 Open Visual Basic, build a standard application program project (Standard EXE)
2、 Set the project environment to use the NI-VISA lib, Click the Existing tab of Project>>Add Existing Item. Search for the visa32.bas file in the include folder under the NI-VISA installation path and add the file.



This allows the VISA functions and VISA data types to be used in a program.

3、 Add codes:

(1)USBTMC access code.

Write a function Usbtmc_test:

```
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 numlnstrs 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
        Exit Function
    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)
NextFind:
    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
```

(2)TCP/IP access code:

Write a function TCP_IP_Test.

```
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
```

4.1.3 Example of MATLAB

Environment: Win7 32bit system, MATLAB R2013a
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.
Follow the steps to complete the example:
1、 Open MATLAB, 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、 Add codes:

(1)USBTMC access code
Write a function Usbtmc_test.

```
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::0xF4ED::0xEE3A::sdg2000x::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
```

2)TCP/IP access code.

Write a function TCP_IP_Test:

```
function TCP_IP_test()
% 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::','10.11.13.32','::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
```

4.1.4 LabVIEW sample

Environment: Win7 32bit system, LabVIEW 2011

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.

Follow the steps to complete the example:

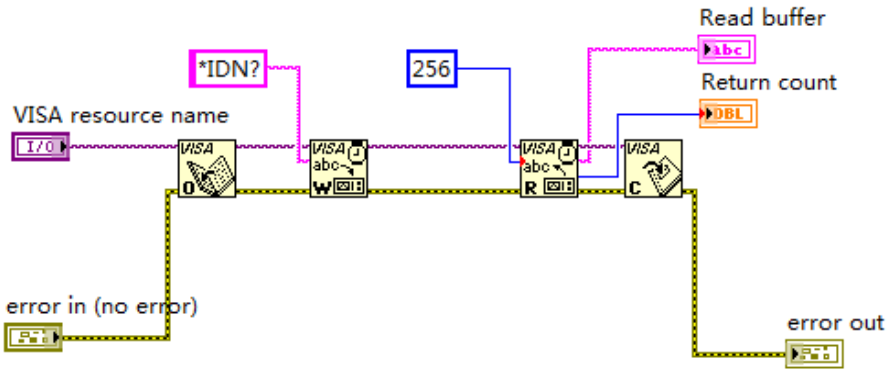
1、 Open LabVIEW, create a VI file.

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

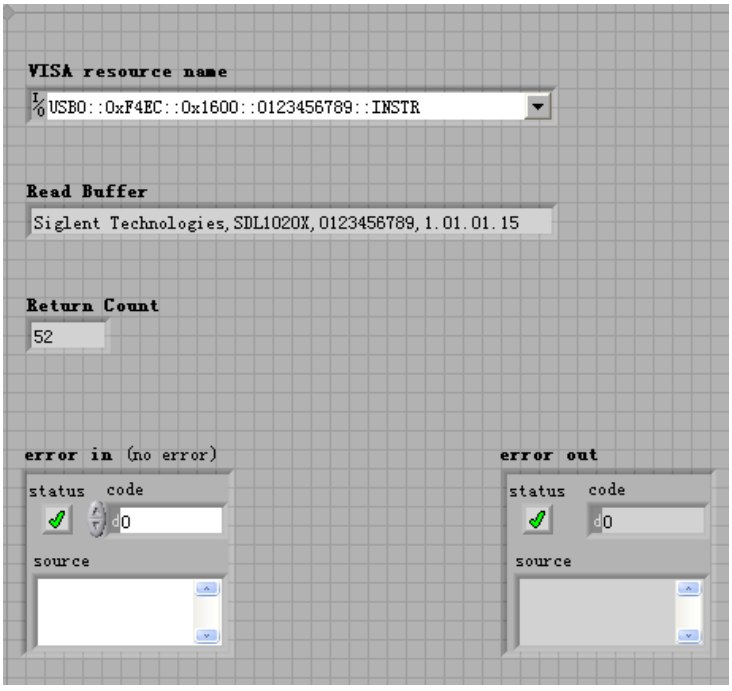
column.

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、 Connect them as shown in the figure below



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

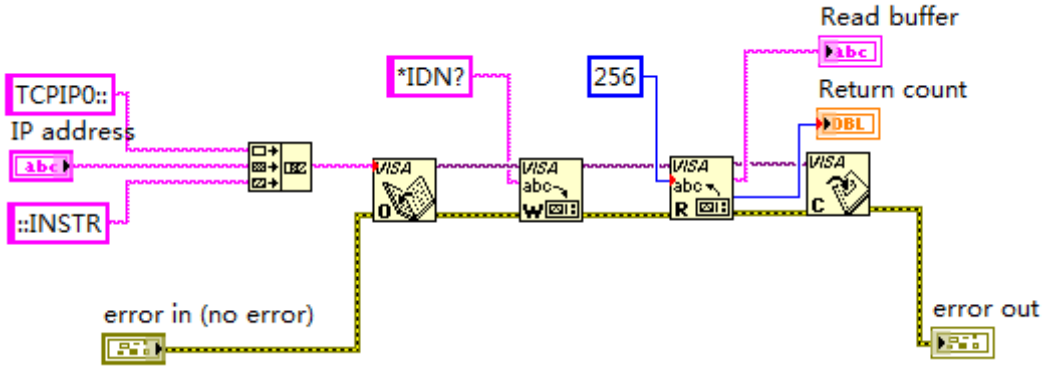
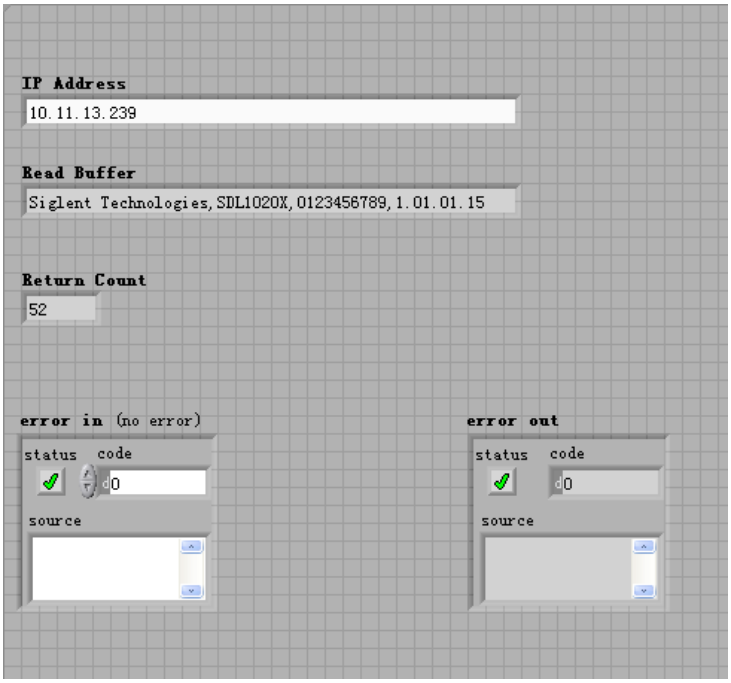


In this example, the VI opens a VISA session to a USBTMC device, writes a command to the device, and reads back the response. In this example, the specific command being sent is the device ID query. Check with your device manufacturer for the device command set. After all communication is complete, the 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、 Connect them as shown in the figure below



8、 Input the IP address and run the program.