RIGOL

Programming Guide

DP700 Series Programmable Linear DC Power Supply

Jun. 2016
RIGOL TECHNOLOGIES, INC.

Guaranty and Declaration

Copyright

Trademark Information

RIGOL is a registered trademark of **RIGOL** TECHNOLOGIES, INC.

Publication Number

PGH05101-1110

Software Version

00.01.02

Software upgrade might change or add product features. Please acquire the latest version of the manual from **RIGOL** website or contact **RIGOL** to upgrade the software.

Notices

- **RIGOL** products are covered by P.R.C. and foreign patents, issued and pending.
- **RIGOL** reserves the right to modify or change parts of or all the specifications and pricing policies at the company's sole decision.
- Information in this publication replaces all previously released materials.
- Information in this publication is subject to change without notice.
- **RIGOL** shall not be liable for either incidental or consequential losses in connection with the furnishing, use, or performance of this manual, as well as any information contained.
- Any part of this document is forbidden to be copied, photocopied, or rearranged without prior written approval of **RIGOL**.

Product Certification

RIGOL guarantees that this product conforms to the national and industrial standards in China as well as the ISO9001:2008 standard and the ISO14001:2004 standard. Other international standard conformance certifications are in progress.

Contact Us

If you have any problem or requirement when using our products or this manual, please contact **RIGOL**.

E-mail: service@rigol.com

Website: www.rigol.com

Document Overview

This manual is your guide to programming the **RIGOL** DP700 series programmable linear DC power supply.

Main Topics in this Manual:

Chapter 1 Programming Overview
This chapter introduces how to set up remote communication between the power supply and the PC, the remote control methods, the syntax, symbols, parameters, and abbreviation rules of the SCPI commands.

Chapter 2 Command System
This chapter introduces the syntax, function, parameters, and usage of each command.

Chapter 3 Application Instances
This chapter provides some application instances of the basic functions of the power supply. Each application instance is composed of a series of commands, which are used to realize the specified basic function.

Chapter 4 Programming Examples
This chapter illustrates how to control the DP700 series by programming in LabVIEW and Visual C#.

Chapter 5 Appendix
This chapter provides default settings and warranty information for the DP700 series.

Tip
For the latest version of this manual, download it from the official website of **RIGOL** (www.rigol.com).

Format Conventions in this Manual:

Key

(1) The key on the front panel is denoted by the format of “Key Name (Bold) + Text Box” in the manual. For example, **On/Off** denotes the "On/Off" key.

(2) Use the screen shot to indicate the key. For example,  denotes the Power key.

Content Conventions in this Manual:

DP700 series programmable linear DC power supply includes the following models. Unless otherwise specified, this manual takes DP711 as an example to make a detailed introduction about the commands of the DP700 series, and lists some application instances and programming examples of DP711.

Model	No. of Channels	Output Voltage/Current
DP711	1	30 V/5 A
DP712	1	50 V/3 A

Contents

Guaranty and Declaration I

Document Overview II

Chapter 1 Programming Overview 1-1
- To Build Remote Communication 1-2
- Remote Control Method 1-4
- SCPI Command Overview 1-4
  - Syntax 1-4
  - Symbol Description 1-4
  - Parameter Type 1-5
  - Command Abbreviation 1-5

Chapter 2 Command System 2-1
- :APPLy Command 2-2
  - :APPLy 2-2
- :DISPlay Command 2-3
  - :DISPlay[:WINDow][:STATe] 2-3
- IEEE488.2 Common Commands 2-4
  - *IDN? 2-4
  - *OPT? 2-4
  - *RCL 2-5
  - *RST 2-5
  - *SAV 2-5
  - *TST? 2-6
- :INSTrument Commands 2-7
  - :INSTrument:NSELect 2-7
  - :INSTrument[:SELEct] 2-7
  - :INSTrument[:SELect] 2-7
- :LIC Command 2-8
  - :LIC:SET 2-8
- :MEASure Commands 2-9
  - :MEASure:ALL[:DC]? 2-9
  - :MEASure:CURRent[:DC]? 2-9
  - :MEASure:POWEr[:DC]? 2-9
  - :MEASure[:VOLTage][:DC]? 2-9
- :MEMory Commands 2-10
  - :MEMory[:STATe]:DELete 2-10
  - :MEMory[:STATe]:LOAD 2-11
  - :MEMory[:STATe]:LOCK 2-11
  - :MEMory[:STATe]:STORe 2-12
  - :MEMory[:STATe]:VALid? 2-13
- :OUTPut Commands 2-14
  - :OUTPut:CVCC? 2-14
  - :OUTPut:MODE? 2-14
  - :OUTPut:OCP:ALAR? 2-15
  - :OUTPut:OCP:QUES? 2-15
  - :OUTPut:OCP:CLEAR 2-16
  - :OUTPut:OCP[:STATe] 2-16
  - :OUTPut:OCP:VALue 2-17
  - :OUTPut:OVP:ALAR? 2-18
  - :OUTPut:OVP:QUES? 2-18
  - :OUTPut:OVP:CLEAR 2-19
  - :OUTPut:OVP[:STATe] 2-19
  - :OUTPut:OVP:VALue 2-20
  - :OUTPut[:STATe] 2-21

:SOURce Commands 2-22
[:SOURce[<n>]]:CURRent[:LEVel][:IMMediate][:AMPLitude] 2-22
[:SOURce[<n>]]:CURRent:PROTection:CLEar 2-23
[:SOURce[<n>]]:CURRent:PROTection[:LEVel] 2-23
[:SOURce[<n>]]:CURRent:PROTection:STATe 2-24
[:SOURce[<n>]]:CURRent:PROTection:TRIPped? 2-25
[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude] 2-25
[:SOURce[<n>]]:VOLTage:PROTection:CLEar 2-26
[:SOURce[<n>]]:VOLTage:PROTection[:LEVel] 2-26
[:SOURce[<n>]]:VOLTage:PROTection:STATe 2-27
[:SOURce[<n>]]:VOLTage:PROTection:TRIPped? 2-28
:SYSTem Commands 2-29
:SYSTem:BEEPer:IMMediate 2-29
:SYSTem:BEEPer[:STATe] 2-30
:SYSTem:BRIGhtness 2-30
:SYSTem:COMMunicate:RS232:BAUD 2-31
:SYSTem:COMMunicate:RS232:DATABit 2-31
:SYSTem:COMMunicate:RS232:PARItybit 2-32
:SYSTem:COMMunicate:RS232:STOPBit 2-32
:SYSTem:ERRor? 2-33
:SYSTem:FAN? 2-33
:SYSTem:KLOCk 2-33
:SYSTem:KLOCk:STATe 2-34
:SYSTem:LANGuage:TYPE 2-35
:SYSTem:LOCal 2-35
:SYSTem:LOCK 2-36
:SYSTem:POWEron 2-36
:SYSTem:PRINT? 2-37
:SYSTem:REMote 2-37
:SYSTem:RWLock[:STATe] 2-37
:SYSTem:SAVer 2-38
:SYSTem:SELF:TEST? 2-38
:SYSTem:TRIGger:IN[:STATe] 2-38
:SYSTem:TRIGger:OUT[:STATe] 2-39
:SYSTem:VERSion? 2-40
:TIMEr Commands 2-41
:TIMEr:CYCLEs 2-41
:TIMEr:ENDState 2-42
:TIMEr:GROUPs 2-42
:TIMEr:PARAmeter 2-43
:TIMEr[:STATe] 2-44
:TIMEr:TRIGger 2-44

Chapter 3 Application Instances 3-1
Constant Voltage Output 3-2
Timing Output 3-2

Chapter 4 Programming Examples 4-1
Programming Preparations 4-1
LabVIEW Programming Example 4-2
Visual C# Programming Example 4-7

Chapter 5 Appendix 5-1
Appendix A: Default Settings 5-1
Appendix B: Warranty 5-2

Chapter 1 Programming Overview

This chapter introduces how to set up remote communication between the power supply and the PC, the remote control methods, the syntax, symbols, parameters, and abbreviation rules of the SCPI commands.

Contents in this chapter:

- To Build Remote Communication
- Remote Control Method
- SCPI Command Overview

To Build Remote Communication

DP700 series power supply can communicate with the PC via the RS232 interface. This section will illustrate how to use the Ultra Sigma software to remotely control the power supply via the RS232 interface.

Note: The RS232 protocol command ends with "\n" for the DP700 series.

Operation Procedures:

1. **Install Ultra Sigma**
   You can download Ultra Sigma from the official website of **RIGOL** (www.rigol.com) and install the software according to the installation wizard.

2. **Connect the power supply to the PC**
   Connect the power supply to the PC by using the 9-pin RS232 cable (female-to-female, straight-through) via the RS232 interface on the rear panel (as shown in the figure below). Then, connect to power and power on the power supply.

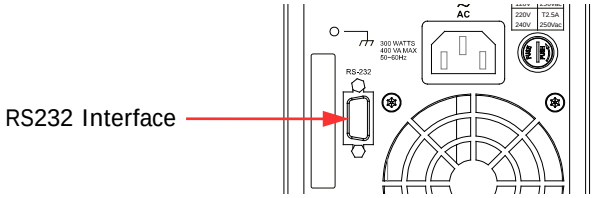
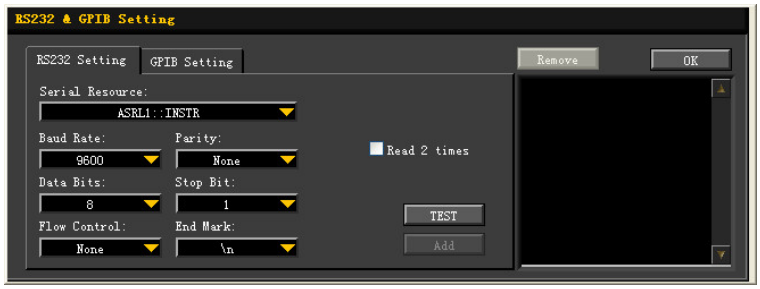


Figure 1-1 DP700 Communication Interface

3. **Set the parameters for the RS232 interface**
   Press **System**, and then press the Left/Right arrow key or use the knob to select the "Inter." tab. Then, you can set the parameters for the RS232 interface according to the help information displayed at the bottom of DP700 interface or descriptions in *DP700 User's Guide*.

4. **Add the device resource**
   Start Ultra Sigma, and then click "RS232". A window is displayed as shown in Figure (a).



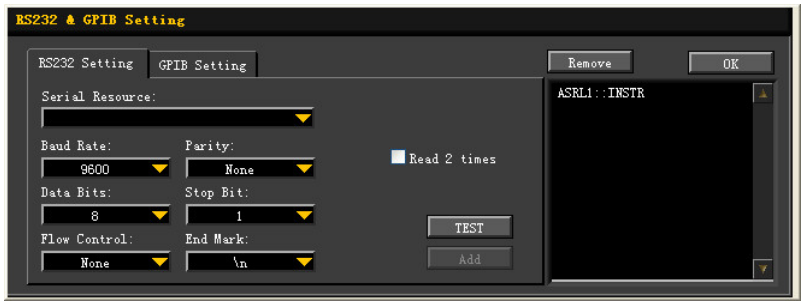
(a)

In the "RS232 Setting" tab, set Baud Rate, Parity, Data Bits, and Stop Bit that match the current parameters of the RS232 interface of DP700 (**Note:** You must select "None" under the "Flow Control" option, and select "\n" under the "End Mark" option). After completing the settings, click "TEST". If the test succeeds, a dialog box, as shown in Figure (b), is displayed.

(b)

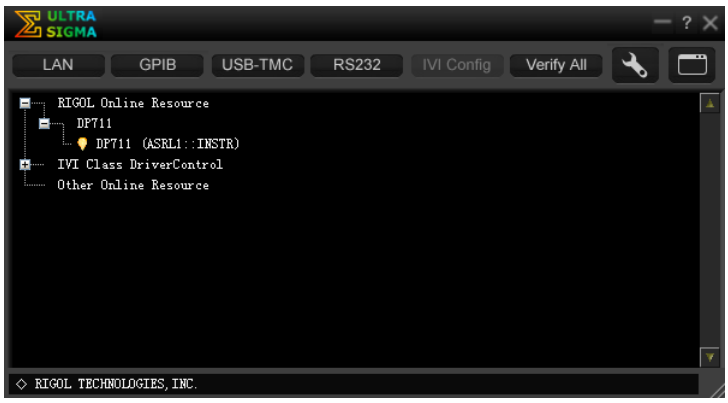
Note: If the test fails, check whether DP700 can communicate with the PC via the RS232 cable normally, whether the current RS232 settings of the software match those of DP700.

In the dialog box shown in Figure (b), click "OK". Then the "Add" button in the "RS232 Setting" tab will be enabled. Click "Add", and then the currently selected instrument resource will be displayed on the right section of the window, as shown in Figure (c). Then, click "OK" to add the instrument resource.



(c)

The added RS232 instrument resource will be displayed under the "RIGOL Online Resource" directory, as shown in Figure (d).



(d)

5. Control the instrument remotely

Right-click the resource name "DP711 (ASRL1::INSTR)", and select "SCPI Panel Control" to open the SCPI command control panel. In the SCPI command control panel, input a correct command. After that, click "Send Command" first and then click "Read Response". Or you can directly click "Send & Read" to verify whether the connection works properly.

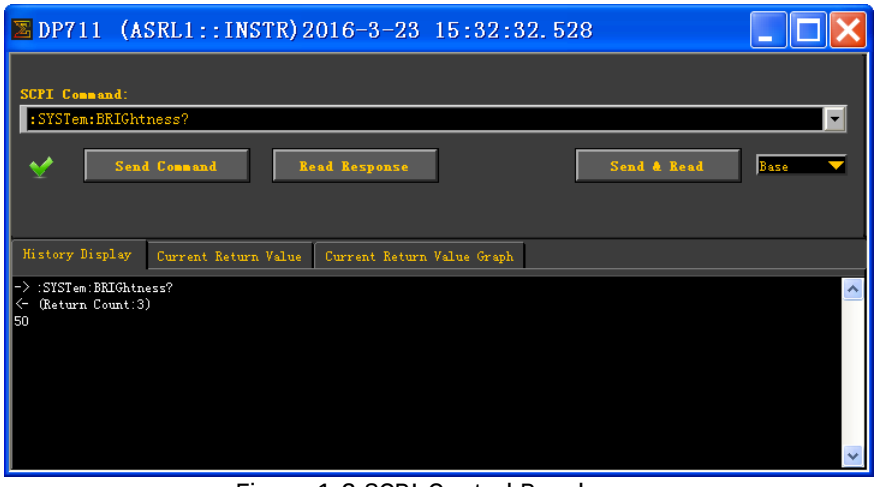


Figure 1-2 SCPI Control Panel

Remote Control Method

1. **User-defined programming**
   You can refer to Chapter 2 in this manual to use the SCPI (Standard Commands for Programmable Instruments) commands to control DP700 by programming in LabVIEW, Visual C#, and other development environments. For details, refer to Chapter 4 in this manual.

2. **Send SCPI commands via the PC software**
   You can use the PC software to send commands to control DP700 remotely. **RIGOL** Ultra Sigma is recommended. You can download the software from **RIGOL** official website (www.rigol.com).

SCPI Command Overview

SCPI (Standard Commands for Programmable Instruments) is a standardized instrument programming language that is built upon the existing standard IEEE 488.1 and IEEE 488.2 and conforms to various standards, such as the floating point operation rule in IEEE 754 standard, ISO 646 7-bit coded character set for information interchange (equivalent to ASCII programming). This chapter introduces the syntax, symbols, parameters, and abbreviation rules of the SCPI commands.

Syntax

The SCPI commands provide a hierarchical tree structure, and consist of multiple subsystems. Each command subsystem consists of one root keyword and one or more sub-keywords. The command line usually starts with a colon; the keywords are separated by colons, and following the keywords are the parameter settings available. The keywords of the command and the first parameter is separated by a space. The command ending with a quotation mark indicates querying a certain function.

For example,
:SYSTem:COMMunicate:RS232:BAUD <baud>
:SYSTem:COMMunicate:RS232:BAUD?

SYSTem is the root keyword of the command, COMMunicate, RS232, and BAUD are the level-2, level-3, and level-4 keywords, respectively. The command line starts with a colon, and different levels of keywords are also separated by colons. <baud> indicates a settable parameter. The command keyword :SYSTem:COMMunicate:RS232:BAUD and the parameter <baud> is separated by a space. The quotation mark (?) indicates querying.

In some commands with multiple parameters, commas are often used to separate these parameters. For example,
:TIMEr:PARAmeter <num>,<voltage>,<current>,<time>

Symbol Description

The following four symbols are not part of the command, and they are not sent with the commands, but taken as delimiters to better describe the parameters in the command.

1. **Braces** { }
   The contents enclosed in the braces can contain multiple optional parameters. When sending the command, you must select one of the parameters.

2. **Vertical Bar** |
   The vertical bar is used to separate multiple parameter options. When sending the command, you must select one of the parameters.

3. **Square Brackets** [ ]
   The contents (keywords or parameters) in the square brackets can be omitted. If the parameter is omitted, it will be set to the default. For example, when sending the :MEASure[:VOLTage][:DC]? command, you can select any one of the following four commands, as they can achieve the same effects as the :MEASure[:VOLTage][:DC]? command.
   :MEASure?
   :MEASure:DC?
   :MEASure:VOLTage?
   :MEASure:VOLTage:DC?

4. **Angle Brackets** < >
   The parameter enclosed in the angle brackets must be replaced by an effective value. For example, the :SYSTem:SAVer <state> command must be sent in the form of :SYSTem:SAVer ON, in which <state> is replaced by an effective value ON.

Parameter Type

The parameters contained in this manual can be divided into the following five types: Bool, Integer, Real, Discrete, and ASCII String.

1. **Bool**
   The parameter can be set to ON (1) or OFF (0).

2. **Integer**
   Unless otherwise specified, the parameter can be any integer within the effective value range.
   Note: Do not set the parameter to a decimal, otherwise, errors will occur.

3. **Real**
   Unless otherwise specified, the parameter can be any real-value (in decimal form or in scientific notation) within the effective value range.

4. **Discrete**
   The parameters can only be the specified numerical values or characters.

5. **ASCII String**
   The parameter can be the combinations of ASCII characters.

Besides, many commands support the MINimum, MAXimum, or DEF parameter. MINimum indicates setting the parameter to a minimum value or querying the minimum value of the parameter; MAXimum indicates setting the parameter to a maximum value or querying the maximum value of the parameter; DEF indicates setting the parameter to a default value.

Command Abbreviation

The letters in the commands are case-insensitive. The commands can be input in uppercase letters or in lowercase letters. If abbreviation is used, you must enter all the uppercase letters that exist in the command syntax. For example, :SYSTem:ERRor? can be abbreviated as :SYST:ERR?

Chapter 2 Command System

This chapter introduces the syntax, functions, parameters, and usage of each command.

Contents in this chapter:

- :APPLy Command
- :DISPlay Command
- IEEE488.2 Common Commands
- :INSTrument Commands
- :LIC Command
- :MEASure Commands
- :MEMory Commands
- :OUTPut Commands
- :SOURce Commands
- :SYSTem Commands
- :TIMEr Commands

Note: In the command system, the commands with the time, voltage, current, or power parameter are allowed to be sent with the corresponding unit. Unless otherwise specified, the unit and the default unit for the parameters are listed in the following table.

Parameter Type	Unit Supported	Default Unit
Time	second (s)	second (s)
Voltage	volt (V), millivolt (mV)	volt (V)
Current	ampere (A), milliampere (mA)	ampere (A)
Power	watt (W), milliwatt (mW)	watt (W)

:APPLy Command

The :APPLy command provides the most straightforward method to program the power supply over the remote interface. You can set the output voltage and current in one command. As long as their setting values are within the settable range, after you execute the command, the output voltage and current will make changes accordingly.

:APPLy

Syntax

:APPLy [<channel>,]{<voltage>|MINimum|MAXimum|DEF}[,<current>|MINimum|MAXimum|DEF]
:APPLy? [<channel>[,<item>]]

Description

Sets the channel output voltage and/or current.
Queries the channel voltage and/or current setting value.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711：{CH1|P30V} DP712：{CH1|P50V}	CH1
<voltage>	Real	DP711: 0 V to 32 V DP712: 0 V to 53 V	0 V
<current>	Real	DP711: 0 A to 5.3 A DP712: 0 A to 3.2 A	DP711: 5 A DP712: 3 A
<item>	Discrete	{VOLTage|CURRent}	--

Remarks

- <voltage> indicates the channel output voltage, and <current> indicates the channel output current. If you specify only one parameter for the command, by default, the power supply regards it as the voltage setting value.
- MINimum indicates setting the channel output voltage/current to a minimum value; MAXimum indicates setting the channel output voltage/current to a maximum value; DEF indicates setting the channel output voltage/current to a default value.
- If <item> is omitted, the command queries both the channel voltage and current setting values.
- You can also use the [:SOURce[<n>]:VOLTage[:LEVel][:IMMediate][:AMPLitude] command to set the channel output voltage, and use the [:SOURce[<n>]:CURRent[:LEVel][:IMMediate][:AMPLitude] command to set the channel output current.

Example

```
:APPLy CH1,5,1          /*Sets the channel output voltage to 5 V, the output current 1 A*/
:APPLy?                 /*Queries the channel voltage and current setting value*/
:APPLy 3                /*Sets the channel output voltage to 3 V*/
:APPLy? CH1,VOLTage     /*Queries the channel voltage setting value*/
```

Return Format

A string
If all parameters are omitted or only the parameter <item> is omitted, the command returns both the channel voltage and current setting values. For example, 5.00,1.00
Otherwise, only the channel voltage or current setting value is returned. For example, 3.00

Related Commands

[:SOURce[<n>]:VOLTage[:LEVel][:IMMediate][:AMPLitude]
[:SOURce[<n>]:CURRent[:LEVel][:IMMediate][:AMPLitude]

:DISPlay Command

The :DISPlay command is used to turn on or off the display.

:DISPlay[:WINDow][:STATe]

Syntax
:DISPlay[:WINDow][:STATe] <state>
:DISPlay[:WINDow][:STATe]?

Description
Turns on or off the display.
Queries the on/off state of the display.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	ON

Remarks
Only when DP700 is in remote mode, can the :DISPlay:WINDow:STATe OFF command be valid. After you turn off the display, if DP700 returns to the local mode, the display is automatically on.

Example
:DISPlay ON /*Turns on the display*/
:DISPlay? /*Queries the on/off state of the display*/

Return Format
ON or OFF

Related Command
:SYSTem:LOCal

IEEE488.2 Common Commands

Command List:

- *IDN?
- *OPT?
- *RCL
- *RST
- *SAV
- *TST?

*IDN?

Syntax
*IDN?

Description
Queries the identification string.

Return Format
RIGOL TECHNOLOGIES,<model>,<serial number>,<software version>
Wherein,
<model>: indicates the model number of the instrument.
<serial number>: indicates the serial number of the instrument.
<software version>: indicates the software version of the instrument.

*OPT?

Syntax
*OPT?

Description
Queries the installation status of the option.

Remarks
- DP700 series provides the following options: Trigger, Timer, and High Resolution.
- If you need any options, please purchase them and refer to the :LIC:SET command to install them properly.

Return Format
<state1>,<state2>,<state3>
Wherein,
<state1>: indicates the installation status of the trigger option. If it has been installed, the command returns DP7-TRIGGER; if not, the command returns 0.
<state2>: indicates the installation status of the timer option. If it has been installed, the command returns DP7-TIMER; if not, the command returns 0.
<state3>: indicates the installation status of the high resolution option. If it has been installed, the command returns DP7-HIRES; if not, the command returns 0.

Related Command
:LIC:SET

*RCL

Syntax
*RCL <position>

Description
Recalls the state file or timer file stored in the specified location in the internal non-volatile memory of the power supply.

Parameter

Name	Type	Range	Default
<position>	Discrete	{1|2|3|4|5|6|7|8|9|10|11|12}	--

Remarks
- The internal non-volatile memory of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- <position>:
  {1|2|3|4|5|6|7|8|9|10}: indicates the storage location for the state file, which corresponds to "State1...State10" in the storage and recall interface.
  {11|12}: indicates the storage location for the timer file, which corresponds to "Timer1 and Timer2" in the storage and recall interface.
- After the files have been read, the status information of the instrument or the timer parameters will change accordingly.
- If no file is found to be saved in the selected storage location, the command is invalid.
- You can also send the :MEMory[:STATe]:LOAD command to recall the saved state files or timer files from the internal memory.

Example
*RCL 2 /*Recalls the state file stored in the storage location State2*/

Related Command
:MEMory[:STATe]:LOAD

*RST

Syntax
*RST

Description
Restores the instrument to the default settings (refer to Appendix A: Default Settings).

Related Command
:SYSTem:ERRor?

*SAV

Syntax
*SAV <position>

Description
Saves the current state or timer parameters of the power supply with the default file names to the specified location of the internal memory.

Parameter

Name	Type	Range	Default
<position>	Discrete	{1|2|3|4|5|6|7|8|9|10|11|12}	--

Remarks

- The internal non-volatile memory of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- <position>:
  {1|2|3|4|5|6|7|8|9|10}: indicates the storage location for the state file, which corresponds to "State1...State10" in the storage and recall interface of the instrument.
  {11|12}: indicates the storage location for the timer file, which corresponds to "Timer1 and Timer2" in the storage and recall interface of the instrument.
- When the state file is saved, the default file name is Rigoln, in which, "n" should be consistent with the parameter <position>.
  When the timer file is saved and <position> is set to 11 or 12, the default file name is Rigol11 or Rigol12.
- If a file is found to be saved to the specified storage location, but the file is not locked, at this time, if you use the command, the current state or timer parameters of the power supply will be directly saved to the specified location (the original file will be overwritten). If a file is found to be saved to the specified storage location, and the file is locked, the command is invalid (the original file will not be overwritten).
- You can also send the :MEMory[:STATe]:STORe command to save the current state or timer parameters of the power supply to the internal memory.

Example

```
*SAV 5     /*Saves the current state of the instrument to the storage location State5,
              with the file name Rigol5*/
```

Related Commands

:MEMory[:STATe]:LOCK
:MEMory[:STATe]:LOAD

*TST?

Syntax

*TST?

Description

Queries the self-test result.

Remarks

- The power supply performs the self-test operation when it is powered on. Sending this command can query the self-test result.
- You can also send the :SYSTem:SELF:TEST? command to query the self-test result of the instrument.

Return Format

A string. For example, Fan:PASS

Related Command

:SYSTem:SELF:TEST?

:INSTrument Commands

Command List:

- :INSTrument:NSELect
- :INSTrument[:SELEct]
- :INSTrument[:SELect]

:INSTrument:NSELect

Syntax
:INSTrument:NSELect <channel>
:INSTrument:NSELect?

Description
Selects the current channel.
Queries the currently selected channel.

Parameter

Name	Type	Range	Default
<channel>	Discrete	{1}	1

Return Format
1

:INSTrument[:SELEct]
:INSTrument[:SELect]

Syntax
:INSTrument[:SELEct] <channel>
:INSTrument[:SELect] <channel>
:INSTrument[:SELEct]?
:INSTrument[:SELect]?

Description
Selects the current channel.
Queries the currently selected channel.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Return Format
CH1

:LIC Command

The :LIC command is used to install the options. DP700 series provides the following options: Trigger, Timer, and High Resolution. If you need any options, please purchase them and install them properly.

- Trigger: provides the trigger input and output function; the order No. is TRIGGER-DP700.
- Timer: outputs based on the preset voltage and current values; the order No. is TIMER-DP700.
- High Resolution: improves the resolution of the instrument; the order No. is HIRES-DP700.

:LIC:SET

Syntax
:LIC:SET <license>

Description
Installs an option.

Parameter

Name	Type	Range	Default
<license>	ASCII String	Refer to **Remarks**	--

Remarks
To install an option, you need an option license (<license>, and each instrument has a unique license). The option license is a 28-character string, which can only consist of English letters and numbers. After you purchase an option, you will obtain a key (used for obtaining the license). Then, you can install the option according to the following steps.

1. **Obtain the option license**
   (1) Log in to the **RIGOL** official website (www.rigol.com), click **SERVICE → Software License Register** to enter the "Registered product license code" interface.
   (2) In the interface, enter the correct key, serial number (press **System**, and select the "Info." tab to obtain the serial number of the instrument), and verification code. Then click **Generate** to obtain the option license for the parameter <license>.
   Note: The hyphens in the option license should be omitted for the parameter <license>.
2. **Install the option**
   Send the :LIC:SET <license> command to install the option.

Example
:LIC:SET 4SN2M2HBB2H4PLHTJFM4PFCEUAVB

Related Command
*OPT?

:MEASure Commands

Command List:

- :MEASure:ALL[:DC]?
- :MEASure:CURRent[:DC]?
- :MEASure:POWEr[:DC]?
- :MEASure[:VOLTage][:DC]?

:MEASure:ALL[:DC]?
:MEASure:CURRent[:DC]?
:MEASure:POWEr[:DC]?
:MEASure[:VOLTage][:DC]?

Syntax

:MEASure:ALL[:DC]? [<channel>]
:MEASure:CURRent[:DC]? [<channel>]
:MEASure:POWEr[:DC]? [<channel>]
:MEASure[:VOLTage][:DC]? [<channel>]

Description

Queries the voltage, current, and power value measured on the channel output terminal.
Queries the current value measured on the channel output terminal.
Queries the power value measured on the channel output terminal.
Queries the voltage value measured on the channel output terminal.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Example

```
:MEASure:ALL? CH1          /*Queries the voltage, current, and power value measured on the channel
                             output terminal*/
:MEASure:CURRent?          /*Queries the current value measured on the channel output terminal*/
:MEASure:POWEr? CH1        /*Queries the power value measured on the channel output terminal*/
:MEASure?                  /*Queries the voltage value measured on the channel output terminal*/
```

Return Format

The :MEASure:ALL[:DC]? command returns three real numbers that are separated by commas. The three real numbers from the left to right indicate the voltage, current, and power, respectively.
For other commands, they return one real number.

:MEMory Commands

The :MEMory commands are used to save the state of the instrument or the timer parameters to the specified location of the internal memory, delete, read, lock, and unlock the saved file in the internal memory.

Command List:

- :MEMory[:STATe]:DELete
- :MEMory[:STATe]:LOAD
- :MEMory[:STATe]:LOCK
- :MEMory[:STATe]:STORe
- :MEMory[:STATe]:VALid?

:MEMory[:STATe]:DELete

Syntax
:MEMory[:STATe]:DELete <file_type>,<position>

Description
Deletes the state files (RSF) or timer files (RTF) stored in the specified locations of the internal non-volatile memory.

Parameter

Name	Type	Range	Default
<file_type>	Discrete	{RSF|RTF}	--
<position>	Discrete	When <file_type> is RSF, the range of <position> is {1|2|3|4|5|6|7|8|9|10}. When <file_type> is RTF, the range of <position> is {1|2}.	--

Remarks
- The internal NVM of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- **Storage location of the state file (RSF)**
  The range of <position> is {1|2|3|4|5|6|7|8|9|10}, which corresponds to "State1...State10" in the storage and recall interface of the instrument.
  Storage location of the timer file (RTF)
  The range of <position> is {1|2}, which corresponds to "Timer1 and Timer2" in the storage and recall interface of the instrument.
- This command is only valid when the file is found to be saved to the specified storage location and the file is not locked.

Example
:MEMory:DELete RTF,2 /*Deletes the file stored in the storage location Timer2*/

Related Command
:MEMory[:STATe]:LOCK

:MEMory[:STATe]:LOAD

Syntax
:MEMory[:STATe]:LOAD <file_type>,<position>

Description
Reads the state files (RSF) or timer files (RTF) stored in the specified locations of the internal non-volatile memory.

Parameter

Name	Type	Range	Default
<file_type>	Discrete	{RSF|RTF}	--
<position>	Discrete	When <file_type> is RSF, the range of <position> is {1|2|3|4|5|6|7|8|9|10}. When <file_type> is RTF, the range of <position> is {1|2}.	--

Remarks
- The internal NVM of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- **Storage location of the state file (RSF)**
  The range of <position> is {1|2|3|4|5|6|7|8|9|10}, which corresponds to "State1...State10" in the storage and recall interface of the instrument.
  Storage location of the timer file (RTF)
  The range of <position> is {1|2}, which corresponds to "Timer1 and Timer2" in the storage and recall interface of the instrument.
- This command is only valid when the file is found in the specified storage location.
- You can also send the *RCL command to recall the saved state file or timer file from the internal non-volatile memory.

Example
:MEMory:LOAD RSF,2 /*Reads the file stored in the storage location Timer2*/

Related Command
*RCL

:MEMory[:STATe]:LOCK

Syntax
:MEMory[:STATe]:LOCK <file_type>,<position>,<state>
:MEMory[:STATe]:LOCK? <file_type>,<position>

Description
Locks or unlocks the state files (RSF) or timer files (RTF) stored in the specified locations of the internal non-volatile memory.
Queries whether the file found to be stored in the specified location of the internal non-volatile memory has been locked.

Parameter

Name	Type	Range	Default
<file_type>	Discrete	{RSF|RTF}	--
<position>	Discrete	When <file_type> is RSF, the range of <position> is {1|2|3|4|5|6|7|8|9|10}. When <file_type> is RTF, the range of <position> is {1|2}.	--
<state>	Bool	{ON|OFF}	--

Remarks

- The internal non-volatile memory of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- **Storage location of the state file (RSF)**
  The range of <position> is {1|2|3|4|5|6|7|8|9|10}, which corresponds to "State1...State10" in the storage and recall interface of the instrument.
  Storage location of the timer file (RTF)
  The range of <position> is {1|2}, which corresponds to "Timer1 and Timer2" in the storage and recall interface of the instrument.
- This command is only valid when the file is found to be stored in the specified storage location.
- The locked file is allowed to be read, but not allowed to be saved and deleted.

Example

```
:MEMory:LOCK RSF,2,ON     /*Locks the file stored in the storage location State2*/
:MEMory:LOCK? RSF,2       /*Queries whether the file stored in the storage location State2 has been
                            locked*/
```

Return Format

YES or NO

Related Command

:MEMory[:STATe]:LOAD

:MEMory[:STATe]:STORe

Syntax

:MEMory[:STATe]:STORe <file_type>,<position>

Description

Saves the current state or timer parameters of the power supply with the default file names to the specified location of the internal non-volatile memory.

Parameter

Name	Type	Range	Default
<file_type>	Discrete	{RSF|RTF}	--
<position>	Discrete	When <file_type> is RSF, the range of <position> is {1|2|3|4|5|6|7|8|9|10}. When <file_type> is RTF, the range of <position> is {1|2}.	--

Remarks

- The internal non-volatile memory of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- **Storage location of the state file (RSF)**
  The range of <position> is {1|2|3|4|5|6|7|8|9|10}, which corresponds to "State1...State10" in the storage and recall interface of the instrument.
  Storage location of the timer file (RTF)
  The range of <position> is {1|2}, which corresponds to "Timer1 and Timer2" in the storage and recall interface of the instrument.
- The state file or timer file is saved with a default file name Rigoln. Wherein, "n" should be consistent with the parameter <position>.
- If a file is found to be saved to the specified storage location, but the file is not locked, at this time, if you use the command, the current state or timer parameters of the power supply will be directly saved to the specified location (the original file will be overwritten). If a file is found in the specified storage location, and the file has been locked, the command is invalid (the original file will not be overwritten).

- You can also send the *SAV command to save the current state or timer parameters of the power supply to the internal nonvolatile memory.

Example

:MEMory:STORe RSF,1 /*Saves the current state of the power supply to State1, with the file name Rigol1*/

Related Commands

:MEMory[:STATe]:LOCK
*SAV

:MEMory[:STATe]:VALid?

Syntax

:MEMory[:STATe]:VALid? <file_type>,<position>

Description

Queries whether a valid file has been saved to the specified location of the internal non-volatile memory.

Parameter

Name	Type	Range	Default
<file_type>	Discrete	{RSF|RTF}	--
<position>	Discrete	When <file_type> is RSF, the range of <position> is {1|2|3|4|5|6|7|8|9|10}. When <file_type> is RTF, the range of <position> is {1|2}.	--

Remarks

- The internal non-volatile memory of DP700 series power supply provides ten storage locations for state files and two for timer files. They are used to store the state information of the instrument and the timer parameters, respectively.
- **Storage location of the state file (RSF)**
  The range of <position> is {1|2|3|4|5|6|7|8|9|10}, which corresponds to "State1...State10" in the storage and recall interface of the instrument.
  Storage location of the timer file (RTF)
  The range of <position> is {1|2}, which corresponds to "Timer1 and Timer2" in the storage and recall interface of the instrument.
- Only when a valid file is found to be saved to the specified location of the internal non-volatile memory, can the saved file be read, deleted, and locked.

Example

:MEMory:VALid? RSF,1 /*Queries whether a valid state file has been saved to the storage location State1*/

Related Commands

:MEMory[:STATe]:DELete

:MEMory[:STATe]:LOAD

:MEMory[:STATe]:LOCK

:OUTPut Commands

The :OUTPut commands are used to enable or disable the channel output, query the channel output mode, set and query the overvoltage/overcurrent protection information.

Command List:

- :OUTPut:CVCC?
- :OUTPut:MODE?
- :OUTPut:OCP:ALAR?
- :OUTPut:OCP:QUES?
- :OUTPut:OCP:CLEAR
- :OUTPut:OCP[:STATe]
- :OUTPut:OCP:VALue
- :OUTPut:OVP:ALAR?
- :OUTPut:OVP:QUES?
- :OUTPut:OVP:CLEAR
- :OUTPut:OVP[:STATe]
- :OUTPut:OVP:VALue
- :OUTPut[:STATe]

:OUTPut:CVCC?
:OUTPut:MODE?

Syntax
:OUTPut:CVCC? [<channel>]
:OUTPut:MODE? [<channel>]

Description
Queries the channel output mode.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Remarks
DP700 series power supply can work in the constant voltage (CV) or constant current (CC) mode. In the CV mode, the output voltage equals to the voltage setting value, and the output current is determined by the load; whereas in the CC mode, the output current equals to the current setting value, and the output voltage is determined by the load. If the power supply should go into a mode of operation that is neither CV nor CC, the power supply is unregulated (UR). In this mode, the output is not predictable. The unregulated condition may be caused by the fact that the AC line voltage is below the specifications. This condition may occur temporarily. For example, when the output voltage is programmed for a large change, the output capacitor or a large capacitive load will charge up until the current reaches the setting value. During the ramp-up of the output voltage, the power supply will be in the unregulated mode. During the transition from CV to CC (e.g. transition caused by a short circuit on the output), the unregulated condition may occur temporarily.

Example
:OUTPut:CVCC? CH1
:OUTPut:MODE? /*Queries the channel output mode*/

Return Format
CV, CC, or UR

:OUTPut:OCP:ALAR?
:OUTPut:OCP:QUES?

Syntax
:OUTPut:OCP:ALAR? [<channel>]
:OUTPut:OCP:QUES? [<channel>]

Description
Queries whether the overcurrent protection (OCP) has occurred.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Remarks
- Overcurrent protection indicates that when the channel output current equals to the overcurrent protection value, the internal OCP flag is set to 1, and a message is displayed on the screen, prompting you that overcurrent protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OCP:CLEAR or [:SOURce[<n>]]:CURRent:PROTection:CLEar command to clear the internal OCP flag and the OCP prompt message.
- You can also send the [:SOURce[<n>]]:CURRent:PROTection:TRIPped? command to query whether the overcurrent protection has currently occurred.

Example
:OUTPut:OCP:ALAR? CH1
:OUTPut:OCP:QUES? /*Queries whether the overcurrent protection has currently occurred*/

Return Format
YES or NO

Related Commands
[:SOURce[<n>]]:CURRent:PROTection:TRIPped?
:OUTPut:OCP:CLEAR
[:SOURce[<n>]]:CURRent:PROTection:CLEar

:OUTPut:OCP:CLEAR

Syntax
:OUTPut:OCP:CLEAR [<channel>]

Description
Clears the internal OCP flag and the OCP prompt message.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Remarks
- Before executing the command, ensure that you have located and resolved the problem for causing the overcurrent protection. To resolve the problem, reduce the output current to below the overcurrent protection value or increase the overcurrent protection value to above the output current.
- You can send the [:SOURce[<n>]]:CURRent:PROTection:CLEar command to clear the internal OCP flag and the OCP prompt message.
- You can send the :OUTPut:OCP:ALAR?, :OUTPut:OCP:QUES?, or [:SOURce[<n>]]:CURRent:PROTection:TRIPped? command to query whether the overcurrent protection has currently occurred.

Example
:OUTPut:OCP:CLEAR /*Clears the internal OCP flag and the OCP prompt message*/

Related Commands
[:SOURce[<n>]]:CURRent:PROTection:CLEar
:OUTPut:OCP:ALAR?
:OUTPut:OCP:QUES?
[:SOURce[<n>]]:CURRent:PROTection:TRIPped?

:OUTPut:OCP[:STATe]

Syntax
:OUTPut:OCP[:STATe] [<channel>,]<state>
:OUTPut:OCP[:STATe]? [<channel>]

Description
Enable or disable the overcurrent protection (OCP) function.
Queries the on/off state of the overcurrent protection (OCP) function.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1
<state>	Bool	{ON|OFF}	OFF

Remarks
- Overcurrent protection indicates that when the channel output current equals to the overcurrent protection value, the internal OCP flag is set to 1, and a message is displayed on the screen, prompting you that overcurrent protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OCP:CLEAR or [:SOURce[<n>]]:CURRent:PROTection:CLEar command to clear the internal OCP flag and the OCP prompt message.
- You can also send the [:SOURce[<n>]]:CURRent:PROTection:STATe command to enable or disable the

overcurrent protection function.

- You can send the :OUTPut:OCP:VALue or [:SOURce[<n>]]:CURRent:PROTection[:LEVel] command to set the channel overcurrent protection value.

Example

:OUTPut:OCP CH1,ON /*Enable the overcurrent protection function*/
:OUTPut:OCP? /*Queries the on/off state of the overcurrent protection function*/

Return Format

ON or OFF

Related Commands

:OUTPut:OCP:CLEAR
[:SOURce[<n>]]:CURRent:PROTection:CLEar
[:SOURce[<n>]]:CURRent:PROTection:STATe
:OUTPut:OCP:VALue
[:SOURce[<n>]]:CURRent:PROTection[:LEVel]

:OUTPut:OCP:VALue

Syntax

:OUTPut:OCP:VALue [<channel>,]{<value>|MINimum|MAXimum}
:OUTPut:OCP:VALue? {[<channel>]|[<channel>,]MINimum|MAXimum}

Description

Sets the channel overcurrent protection (OCP) value.
Queries the channel overcurrent protection (OCP) value.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1
<value>	Real	Standard: DP711: 0.01 A to 5.5 A DP712: 0.01 A to 3.3 A High resolution option (ordering number: HIRES-DP700) installed: DP711: 0.001 A to 5.5 A DP712: 0.001 A to 3.3 A	DP711: 5.3 A DP712: 3.2 A

Remarks

- If the overcurrent protection function is enabled and the current of the output terminal equals to the currently set overcurrent protection value, then the internal OCP flag is set to 1, and a message is displayed on the screen, prompting you that overcurrent protection occurs and the channel output is automatically disabled.
- For the setting command, MINimum indicates setting the channel overcurrent protection to a minimum value; MAXimum indicates setting the channel overcurrent protection to a maximum value.
  For the query command, MINimum indicates querying the minimum settable value of the channel overcurrent protection; MAXimum indicates querying the maximum settable value of the channel overcurrent protection.
- You can also send the [:SOURce[<n>]]:CURRent:PROTection[:LEVel] command to set the channel overcurrent protection value.

Example

:OUTPut:OCP:VALue 5 /*Sets the channel overcurrent protection value to 5 A*/
:OUTPut:OCP:VALue? CH1 /*Queries the channel overcurrent protection value*/

Return Format
A real number

Related Commands
:OUTPut:OVP[:STATe]
[:SOURce[<n>]]:CURRent:PROTection:STATe
[:SOURce[<n>]]:CURRent:PROTection[:LEVel]

:OUTPut:OVP:ALAR?
:OUTPut:OVP:QUES?

Syntax
:OUTPut:OVP:ALAR? [<channel>]
:OUTPut:OVP:QUES? [<channel>]

Description
Queries whether the overvoltage protection (OVP) has occurred.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Remarks
- Overvoltage protection indicates that when the channel output voltage equals to the overvoltage protection value, the internal OVP flag is set to 1, and a message is displayed on the screen, prompting you that overvoltage protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OVP:CLEAR or [:SOURce[<n>]]:VOLTage:PROTection:CLEar command to clear the internal OVP flag and the OVP prompt message.
- You can send the [:SOURce[<n>]]:VOLTage:PROTection:TRIPped? command to query whether the overvoltage protection has currently occurred.

Example
:OUTPut:OVP:ALAR? CH1
:OUTPut:OVP:QUES? /*Queries whether the overvoltage protection has currently occurred*/

Return Format
YES or NO

Related Commands
:OUTPut:OVP:CLEAR
[:SOURce[<n>]]:VOLTage:PROTection:CLEar
[:SOURce[<n>]]:VOLTage:PROTection:TRIPped?

:OUTPut:OVP:CLEAR

Syntax
:OUTPut:OVP:CLEAR [<channel>]

Description
Clears the internal OVP flag and the OVP prompt message.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1

Remarks
- Before executing the command, ensure that you have located and resolved the problem for causing the overvoltage protection. To resolve the problem, reduce the output voltage to below the overvoltage protection value or increase the overvoltage protection value to above the output voltage.
- You can also send the [:SOURce[<n>]]:VOLTage:PROTection:CLEar command to clear the internal OVP flag and the OVP prompt message.
- You can send the :OUTPut:OVP:ALAR?, :OUTPut:OVP:QUES?, or [:SOURce[<n>]]:VOLTage:PROTection:TRIPped? command to query whether the overvoltage protection has currently occurred.

Example
:OUTPut:OVP:CLEAR /*Clears the internal OVP flag and the OVP prompt message*/

Related Commands
[:SOURce[<n>]]:VOLTage:PROTection:CLEar
:OUTPut:OVP:ALAR?
:OUTPut:OVP:QUES?
[:SOURce[<n>]]:VOLTage:PROTection:TRIPped?

:OUTPut:OVP[:STATe]

Syntax
:OUTPut:OVP[:STATe] [<channel>,]<state>
:OUTPut:OVP[:STATe]? [<channel>]

Description
Enable or disable the overvoltage protection (OVP) function.
Queries the on/off state of the overvoltage protection (OVP) function.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1
<state>	Bool	{ON|OFF}	OFF

Remarks
- Overvoltage protection indicates that when the channel output voltage equals to the overvoltage protection value, the internal OVP flag is set to 1, and a message is displayed on the screen, prompting you that overvoltage protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OVP:CLEAR or [:SOURce[<n>]]:VOLTage:PROTection:CLEar command to clear the internal OVP flag and the OVP prompt message.
- You can also send the [:SOURce[<n>]]:VOLTage:PROTection:STATe command to enable or disable the overvoltage protection function.

- You can send the :OUTPut:OVP:VALue or [:SOURce[<n>]]:VOLTage:PROTection[:LEVel] command to set the channel overvoltage protection value.

Example
:OUTPut:OVP:STATe CH1,ON /*Enables the overvoltage protection function*/
:OUTPut:OVP:STATe? /*Queries the on/off state of the overvoltage protection function*/

Return Format
ON or OFF

Related Commands
:OUTPut:OVP:CLEAR
[:SOURce[<n>]]:VOLTage:PROTection:CLEar
[:SOURce[<n>]]:VOLTage:PROTection:STATe
:OUTPut:OVP:VALue
[:SOURce[<n>]]:VOLTage:PROTection[:LEVel]

:OUTPut:OVP:VALue

Syntax
:OUTPut:OVP:VALue [<channel>,]{<value>|MINimum|MAXimum}
:OUTPut:OVP:VALue? {[<channel>]|[<channel>,]MINimum|MAXimum}

Description
Sets the channel overvoltage protection (OVP) value.
Queries the channel overvoltage protection (OVP) value.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1
<value>	Real	Standard: DP711: 0.01 V to 33 V DP712: 0.01 V to 55 V High resolution option (ordering number: HIRES-DP700) installed: DP711: 0.001 V to 33 V DP712: 0.001 V to 55 V	DP711: 32 V DP712: 53 V

Remarks
- If the overvoltage protection function is enabled and the voltage of the output terminal equals to the currently set overvoltage protection value, then the internal OVP flag is set to 1, and a message is displayed on the screen, prompting you that overvoltage protection occurs and the channel output is automatically disabled.
- For the setting command, MINimum indicates setting the channel overvoltage protection to a minimum value; MAXimum indicates setting the channel overvoltage protection to a maximum value.
  For the query command, MINimum indicates querying the minimum settable value of the channel overvoltage protection; MAXimum indicates querying the maximum settable value of the channel overvoltage protection.
- You can also send the [:SOURce[<n>]]:VOLTage:PROTection[:LEVel] command to set the channel overvoltage protection value.

Example
:OUTPut:OVP:VALue 30 /*Sets the channel overvoltage protection value to 30 V*/
:OUTPut:OVP:VALue? CH1 /*Queries the channel overvoltage protection value*/

Return Format
A real number

Related Commands
:OUTPut:OVP[:STATe]
[:SOURce[<n>]]:VOLTage:PROTection:STATe
[:SOURce[<n>]]:VOLTage:PROTection[:LEVel]

:OUTPut[:STATe]

Syntax
:OUTPut[:STATe] [<channel>,]<state>
:OUTPut[:STATe]? [<channel>]

Description
Enable or disable the channel output.
Queries the on/off state of the channel output.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1
<state>	Bool	{ON|OFF}	OFF

Remarks
Before enabling the channel output, ensure that the current settings will not affect the equipment connected to the channel output terminal.

Example
:OUTPut:STATe CH1,ON /*Enable the channel output*/
:OUTPut:STATe? CH1 /*Queries the on/off state of the channel output*/

Return Format
ON or OFF

:SOURce Commands

The :SOURce commands are used to set and query the channel output voltage/current, and the overvoltage/overcurrent protection information. These commands can be used to modify a single parameter, with a great flexibility.

Command List:

- [:SOURce[<n>]:CURRent[:LEVel][:IMMediate][:AMPLitude]
- [:SOURce[<n>]:CURRent:PROTection:CLEar
- [:SOURce[<n>]:CURRent:PROTection[:LEVel]
- [:SOURce[<n>]:CURRent:PROTection:STATe
- [:SOURce[<n>]:CURRent:PROTection:TRIPped?
- [:SOURce[<n>]:VOLTage[:LEVel][:IMMediate][:AMPLitude]
- [:SOURce[<n>]:VOLTage:PROTection:CLEar
- [:SOURce[<n>]:VOLTage:PROTection[:LEVel]
- [:SOURce[<n>]:VOLTage:PROTection:STATe
- [:SOURce[<n>]:VOLTage:PROTection:TRIPped?

[:SOURce[<n>]:CURRent[:LEVel][:IMMediate][:AMPLitude]

Syntax

[:SOURce[<n>]:CURRent[:LEVel][:IMMediate][:AMPLitude] {<current>|MINimum|MAXimum}
[:SOURce[<n>]:CURRent[:LEVel][:IMMediate][:AMPLitude]? [MINimum|MAXimum]

Description

Sets the channel output current.
Queries the channel current setting value.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1
<current>	Real	DP711: 0 A to 5.3 A DP712: 0 A to 3.2 A	DP711: 5 A DP712: 3 A

Remarks

- For the setting command, MINimum indicates setting the channel output current to a minimum value; MAXimum indicates setting the channel output current to a maximum value.
  For the query command, MINimum indicates querying the minimum settable value of the current channel current; MAXimum indicates querying the maximum settable value of the current channel current.
- You can send the :APPLy command to set the channel output voltage and/or current.

Example

:CURRent 3 /*Sets the channel output current to 3 A*/
:CURRent? /*Queries the channel current setting value*/

Return Format

A real number

Related Command

:APPLy

[:SOURce[<n>]]:CURRent:PROTection:CLEar

Syntax
[:SOURce[<n>]]:CURRent:PROTection:CLEar

Description
Clears the internal OCP flag and the OCP prompt message.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1

Remarks
- Before executing the command, ensure that you have located and resolved the problem for causing the overcurrent protection. To resolve the problem, reduce the output current to below the overcurrent protection value or increase the overcurrent protection value to above the output current value. Executing the command will clear the internal OCP flag and the OCP prompt message.
- You can send the :OUTPut:OCP:CLEAR command to clear the internal OCP flag and the OCP prompt message.
- You can also send the :OUTPut:OCP:ALAR?, :OUTPut:OCP:QUES?, or [:SOURce[<n>]]:CURRent:PROTection:TRIPped? command to query whether the overcurrent protection has currently occurred.

Example
CURRent:PROTection:CLEar /*Clears the internal OCP flag and the OCP prompt message*/

Related Commands
:OUTPut:OCP:CLEAR
:OUTPut:OCP:ALAR?
:OUTPut:OCP:QUES?
[:SOURce[<n>]]:CURRent:PROTection:TRIPped?

[:SOURce[<n>]]:CURRent:PROTection[:LEVel]

Syntax
[:SOURce[<n>]]:CURRent:PROTection[:LEVel] {<value>|MINimum|MAXimum}
[:SOURce[<n>]]:CURRent:PROTection[:LEVel]? [MINimum|MAXimum]

Description
Sets the channel overcurrent protection (OCP) value.
Queries the overcurrent protection (OCP) value for the channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1
<value>	Real	Standard: 0.01 A to 5.5 A High resolution option (ordering number: HIRES-DP700) installed: 0.001 A to 5.5 A	5.3 A

Remarks
- If the overcurrent protection function is enabled and the current of the output terminal equals to the currently set overcurrent protection value, then the internal OCP flag is set to 1, and a message is displayed on the screen, prompting you that overcurrent protection occurs and the channel output is automatically disabled.
- For the setting command, MINimum indicates setting the channel overcurrent protection to a minimum value; MAXimum indicates setting the channel overcurrent protection to a maximum value.

For the query command, MINimum indicates querying the minimum settable value of the channel overcurrent protection; MAXimum indicates querying the maximum settable value of the channel overcurrent protection.

- You can also send the :OUTPut:OCP:VALue command to set the channel overcurrent protection value.

Example
:CURRent:PROTection 5 /*Sets the channel overcurrent protection value to 5 A*/
:CURRent:PROTection? /*Queries the channel overcurrent protection value*/

Return Format
A real number

Related Commands
:OUTPut:OVP[:STATe]
[:SOURce[<n>]]:CURRent:PROTection:STATe
:OUTPut:OCP:VALue

[:SOURce[<n>]]:CURRent:PROTection:STATe

Syntax
[:SOURce[<n>]]:CURRent:PROTection:STATe <state>
[:SOURce[<n>]]:CURRent:PROTection:STATe?

Description
Enable or disable the overcurrent protection (OCP) function.
Queries the on/off state of the overcurrent protection (OCP) function.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1
<state>	Bool	{ON|OFF}	OFF

Remarks

- Overcurrent protection indicates that when the channel output current equals to the overcurrent protection value, the internal OCP flag is set to 1, and a message is displayed on the screen, prompting you that overcurrent protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OCP:CLEAR or [:SOURce[<n>]]:CURRent:PROTection:CLEar command to clear the internal OCP flag and the OCP prompt message.
- You can also send the :OUTPut:OCP[:STATe] command to enable or disable the overcurrent protection function.
- You can send the :OUTPut:OCP:VALue or [:SOURce[<n>]]:CURRent:PROTection[:LEVel] command to set the channel overcurrent protection value.

Example
:CURRent:PROTection:STATe ON /*Enables the overcurrent protection function*/
:CURRent:PROTection:STATe? /*Queries the on/off state of the overcurrent protection function*/

Return Format
ON or OFF

Related Commands
:OUTPut:OCP:CLEAR
[:SOURce[<n>]]:CURRent:PROTection:CLEar
:OUTPut:OCP[:STATe]
:OUTPut:OCP:VALue
[:SOURce[<n>]]:CURRent:PROTection[:LEVel]

[:SOURce[<n>]]:CURRent:PROTection:TRIPped?

Syntax
[:SOURce[<n>]]:CURRent:PROTection:TRIPped?

Description
Queries whether the overcurrent protection (OCP) has occurred.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1

Remarks
- Overcurrent protection indicates that when the channel output current equals to the overcurrent protection value, the internal OCP flag is set to 1, and a message is displayed on the screen, prompting you that overcurrent protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OCP:CLEAR or [:SOURce[<n>]]:CURRent:PROTection:CLEar command to clear the internal OCP flag and the OCP prompt message.
- You can also send the :OUTPut:OCP:ALAR? or :OUTPut:OCP:QUES? command to query whether the overcurrent protection has currently occurred.

Example
:CURRent:PROTection:TRIPped? /*Queries whether the overcurrent protection has currently occurred*/

Return Format
YES or NO

Related Commands
:OUTPut:OCP:CLEAR
[:SOURce[<n>]]:CURRent:PROTection:CLEar
:OUTPut:OCP:ALAR?
:OUTPut:OCP:QUES?

[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]

Syntax
[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude] {<voltage>|MINimum|MAXimum}
[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]? [MINimum|MAXimum]

Description
Sets the channel output voltage.
Queries the channel voltage setting value.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1
<current>	Real	0 V to 30 V	0 V

Remarks
- For the setting command, MINimum indicates setting the channel output voltage to a minimum value; MAXimum indicates setting the channel output voltage to a maximum value.
  For the query command, MINimum indicates querying the minimum settable value of the channel voltage; MAXimum indicates querying the maximum settable value of the channel voltage.
- You can also send the :APPLy command to set the channel output voltage and/or current.

Example
:VOLTage 20 /*Sets the channel output voltage to 20 V*/
:VOLTage? /*Queries the channel voltage setting value*/

Return Format
A real number

Related Command
:APPLy

[:SOURce[<n>]]:VOLTage:PROTection:CLEar

Syntax
[:SOURce[<n>]]:VOLTage:PROTection:CLEar

Description
Clears the internal OVP flag and the OVP prompt message.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1

Remarks
- Before executing the command, ensure that you have located and resolved the problem for causing the overvoltage protection. To resolve the problem, reduce the output voltage to below the overvoltage protection value or increase the overvoltage protection value to above the output voltage. Executing the command will clear the internal OVP flag and the OVP prompt message.
- You can send the :OUTPut:OVP:CLEAR command to clear the internal OVP flag and the OVP prompt message.
- You can also send the :OUTPut:OVP:ALAR?, :OUTPut:OVP:QUES?, or [:SOURce[<n>]]:VOLTage:PROTection:TRIPped? command to query whether the overvoltage protection has currently occurred.

Example
:VOLTage:PROTection:CLEar /*Clears the internal OVP flag and the OVP prompt message*/

Related Commands
:OUTPut:OVP:CLEAR
:OUTPut:OVP:ALAR?
:OUTPut:OVP:QUES?
[:SOURce[<n>]]:VOLTage:PROTection:TRIPped?

[:SOURce[<n>]]:VOLTage:PROTection[:LEVel]

Syntax
[:SOURce[<n>]]:VOLTage:PROTection[:LEVel] {<value>|MINimum|MAXimum}
[:SOURce[<n>]]:VOLTage:PROTection[:LEVel]? [MINimum|MAXimum]

Description
Sets the channel overvoltage protection (OVP) value.
Queries the channel overvoltage protection (OVP) value.

Parameter

Name	Type	Range	Default
<channel>	Discrete	DP711: {CH1|P30V} DP712: {CH1|P50V}	CH1
<value>	Real	Standard: 0.01 V to 33 V High resolution option (ordering number: HIRES-DP700) installed: 0.001 V to 33 V	32 V

Remarks

- If the overvoltage protection function is enabled and the voltage of the output terminal equals to the currently set overvoltage protection value, then the internal OVP flag is set to 1, and a message is displayed on the screen, prompting you that overvoltage protection occurs and the channel output is automatically disabled.
- For the setting command, MINimum indicates setting the channel overvoltage protection to a minimum value; MAXimum indicates setting the channel overvoltage protection to a maximum value.
  For the query command, MINimum indicates querying the minimum settable value of the channel overvoltage protection; MAXimum indicates querying the maximum settable value of the channel overvoltage protection.
- You can also send the :OUTPut:OVP:VALue command to set the channel overvoltage protection value.

Example

```
:VOLTage:PROTection 30     /*Sets the channel overvoltage protection value to 30 V*/
:VOLTage:PROTection?       /*Queries the channel overvoltage protection value*/
```

Return Format

A real number.

Related Commands

:OUTPut:OVP[:STATe]
[:SOURce[<n>]]:VOLTage:PROTection:STATe
:OUTPut:OVP:VALue

[:SOURce[<n>]]:VOLTage:PROTection:STATe

Syntax

```
[:SOURce[<n>]]:VOLTage:PROTection:STATe <state>
[:SOURce[<n>]]:VOLTage:PROTection:STATe?
```

Description

Enable or disable the overvoltage protection (OVP) function.
Queries the on/off state of the overvoltage protection (OVP) function.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1
<state>	Bool	{ON|OFF}	OFF

Remarks

- Overvoltage protection indicates that when the channel output voltage equals to the overvoltage protection value, the internal OVP flag is set to 1, and a message is displayed on the screen, prompting you that overvoltage protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OVP:CLEAR or [:SOURce[<n>]]:VOLTage:PROTection:CLEar command to clear the internal OCP flag and the OCP prompt message.
- You can also send the :OUTPut:OVP[:STATe] command to enable or disable the overvoltage protection function.

- You can send the :OUTPut:OVP:VALue or [:SOURce[<n>]]:VOLTage:PROTection[:LEVel] command to set the channel overvoltage protection value.

Example
:VOLTage:PROTection:STATe ON /*Enables the overvoltage protection function*/
:VOLTage:PROTection:STATe? /*Queries the on/off state of the overvoltage protection function*/

Return Format
ON or OFF

Related Commands
:OUTPut:OVP:CLEAR
[:SOURce[<n>]]:VOLTage:PROTection:CLEar
:OUTPut:OVP[:STATe]
:OUTPut:OVP:VALue
[:SOURce[<n>]]:VOLTage:PROTection[:LEVel]

[:SOURce[<n>]]:VOLTage:PROTection:TRIPped?

Syntax
[:SOURce[<n>]]:VOLTage:PROTection:TRIPped?

Description
Queries whether the overvoltage protection (OVP) has occurred.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1}	1

Remarks
- Overvoltage protection indicates that when the channel output voltage equals to the overvoltage protection value, the internal OVP flag is set to 1, and a message is displayed on the screen, prompting you that overvoltage protection occurs and the channel output is automatically disabled. At this time, you can send the :OUTPut:OVP:CLEAR or [:SOURce[<n>]]:VOLTage:PROTection:CLEar command to clear the internal OVP flag and the OVP prompt message.
- You can also send the :OUTPut:OVP:ALAR? or :OUTPut:OVP:QUES? command to query whether the overvoltage protection has currently occurred.

Example
:VOLTage:PROTection:TRIPped? /*Queries whether the overvoltage protection has currently occurred*/

Return Format
YES or NO

Related Commands
:OUTPut:OVP:CLEAR
[:SOURce[<n>]]:VOLTage:PROTection:CLEar
:OUTPut:OVP:ALAR?
:OUTPut:OVP:QUES?

:SYSTem Commands

Command List:

- :SYSTem:BEEPer:IMMediate
- :SYSTem:BEEPer[:STATe]
- :SYSTem:BRIGhtness
- :SYSTem:COMMunicate:RS232:BAUD
- :SYSTem:COMMunicate:RS232:DATABit
- :SYSTem:COMMunicate:RS232:PARItybit
- :SYSTem:COMMunicate:RS232:STOPBit
- :SYSTem:ERRor?
- :SYSTem:FAN?
- :SYSTem:KLOCk
- :SYSTem:KLOCk:STATe
- :SYSTem:LANGuage:TYPE
- :SYSTem:LOCal
- :SYSTem:LOCK
- :SYSTem:POWEron
- :SYSTem:PRINT?
- :SYSTem:REMote
- :SYSTem:RWLock[:STATe]
- :SYSTem:SAVer
- :SYSTem:SELF:TEST?
- :SYSTem:TRIGger:IN[:STATe]
- :SYSTem:TRIGger:OUT[:STATe]
- :SYSTem:VERSion?

:SYSTem:BEEPer:IMMediate

Syntax
:SYSTem:BEEPer:IMMediate

Description
Enables the beeper to sound immediately.

:SYSTem:BEEPer[:STATe]

Syntax
:SYSTem:BEEPer[:STATe] <state>
:SYSTem:BEEPer[:STATe]?

Description
Turns on/off the beeper.
Queries the on/off state of the beeper.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	ON

Remarks
When the beeper is turned on, the beeper sounds when you perform the following operations.
- When you press the keys on the front panel
- When you rotate the knob
- When an error occurs in remote operation

Example
:SYSTem:BEEPer ON /*Turns on the beeper*/
:SYSTem:BEEPer? /*Queries the on/off state of the beeper*/

Return Format
ON or OFF

:SYSTem:BRIGhtness

Syntax
:SYSTem:BRIGhtness {<bright>|MINimum|MAXimum}
:SYSTem:BRIGhtness? [MINimum|MAXimum]

Description
Sets screen brightness.
Queries screen brightness.

Parameter

Name	Type	Range	Default
<bright>	Integer	1 to 100	50

Remarks
For the setting command, MINimum indicates setting screen brightness to a minimum value; MAXimum indicates setting screen brightness to a maximum value.
For the query command, MINimum indicates querying the minimum settable value of screen brightness; MAXimum indicates querying the maximum settable value of screen brightness.

Example
:SYSTem:BRIGhtness 55 /*Sets screen brightness to 55%*/
:SYSTem:BRIGhtness? /*Queries screen brightness*/

Return Format
An integer between 1 and 100

:SYSTem:COMMunicate:RS232:BAUD

Syntax
:SYSTem:COMMunicate:RS232:BAUD <baud>
:SYSTem:COMMunicate:RS232:BAUD?

Description
Sets the baud rate of the RS232 interface.
Queries the baud rate of the RS232 interface.

Parameter

Name	Type	Range	Default
<baud>	Discrete	{7200|9600|14400|19200|38400|57600|115200}	9600

Remarks
Communication can be set up between the DP700 series power supply and the PC via the RS232 interface. Use the 9-pin RS232 cable (female-to-female, straight-through) to connect the power supply to the PC via the RS232 interface on the rear panel.

Example
:SYSTem:COMMunicate:RS232:BAUD 14400 /*Sets the baud rate of the RS232 interface to 14400*/
:SYSTem:COMMunicate:RS232:BAUD? /*Queries the baud rate of the RS232 interface*/

Return Format
7200, 9600, 14400, 19200, 38400, 57600, or 115200

:SYSTem:COMMunicate:RS232:DATABit

Syntax
:SYSTem:COMMunicate:RS232:DATABit <bit>
:SYSTem:COMMunicate:RS232:DATABit?

Description
Sets the number of the data bits of the RS232 interface.
Queries the number of the data bits of the RS232 interface.

Parameter

Name	Type	Range	Default
<bit>	Discrete	{8}	8

Remarks
- Communication can be set up between the DP700 series power supply and the PC via the RS232 interface. Use the 9-pin RS232 cable (female-to-female, straight-through) to connect the power supply to the PC via the RS232 interface on the rear panel.
- The number of the data bits of the RS232 interface is 8, and cannot be modified.

Return Format
8

:SYSTem:COMMunicate:RS232:PARItybit

Syntax
:SYSTem:COMMunicate:RS232:PARItybit <parity>
:SYSTem:COMMunicate:RS232:PARItybit?

Description
Sets the parity of the RS232 interface.
Queries the parity of the RS232 interface.

Parameter

Name	Type	Range	Default
<parity>	Discrete	{NONE|ODD|EVEN}	NONE

Remarks
- Communication can be set up between the DP700 series power supply and the PC via the RS232 interface. Use the 9-pin RS232 cable (female-to-female, straight-through) to connect the power supply to the PC via the RS232 interface on the rear panel.
- <parity>:
  NONE: indicates none.
  ODD: indicates the odd check.
  EVEN: indicates the even check.

Example
:SYSTem:COMMunicate:RS232:PARItybit EVEN /*Sets the parity of the RS232 interface to EVEN*/
:SYSTem:COMMunicate:RS232:PARItybit? /*Queries the parity of the RS232 interface*/

Return Format
NONE, ODD, or EVEN

:SYSTem:COMMunicate:RS232:STOPBit

Syntax
:SYSTem:COMMunicate:RS232:STOPBit <bit>
:SYSTem:COMMunicate:RS232:STOPBit?

Description
Sets the number of the stop bits of the RS232 interface.
Queries the number of the stop bits of the RS232 interface.

Parameter

Name	Type	Range	Default
<bit>	Discrete	{1}	1

Remarks
- Communication can be set up between the DP700 series power supply and the PC via the RS232 interface. Use the 9-pin RS232 cable (female-to-female, straight-through) to connect the power supply to the PC via the RS232 interface on the rear panel.
- The number of the stop bits of the RS232 interface is 1, and cannot be modified.

Return Format
1

:SYSTem:ERRor?

Syntax
:SYSTem:ERRor?

Description
Queries the last error message in the error queue and clears the error message.

Remarks
DP700 series allows you to view the last 5 errors.

Return Format
The number and contents of the error message, such as -113,"Undefined header; keyword cannot be found"

:SYSTem:FAN?

Syntax
:SYSTem:FAN?

Description
Queries the self-test result of the fan.

Remarks
- The power supply performs the self-test operation when it is powered on. Sending this command can query the self-test result of the fan.
- You can also send the *TST? or :SYSTem:SELF:TEST? command to query the self-test result of the instrument.

Return Format
A string. For example, FAN:PASS

Related Commands
*TST?
:SYSTem:SELF:TEST?

:SYSTem:KLOCk

Syntax
:SYSTem:KLOCk <key>,<state>
:SYSTem:KLOCk? <key>

Description
Locks or unlocks the specified key on the front panel.
Queries whether the specified key on the front panel is locked or not.

Parameter

Name	Type	Range	Default
<key>	Discrete	Refer to **Remarks**	--
<state>	Bool	{ON|OFF|1|0}	0

Remarks
- DP700 series power supply provides the key locking functions to avoid any loss caused by misoperation.

- The keys locked cannot be used. You are allowed to lock the specified keys or all keys on the front panel (including the knob, but excluding ).

- The parameter <key> indicates a specified key, and its range is as follows:

SYSTEM|MEMORY|TIMER|	/*Function keys*/
OUTPUT|	/*Output on/off key*/
NUM0|NUM1|NUM2|NUM3|NUM4|NUM5| NUM6|NUM7|NUM8|NUM9|DOT|	/*Num keys*/
BACK|	/*Return key*/
KNOB|	/*Knob*/
LEFT|RIGHT|UP|DOWN|	/*Arrow keys*/
OK|	/*Confirmation key*/
ALL	/*All keys on the front panel (excluding ) and the knob*/

- <state>:
  ON|1: locks the specified key.
  OFF|0: unlocks the specified key.

Example
:SYSTem:KLOCk MEMORY,ON /*Locks the **Memory** key*/
:SYSTem:KLOCk? MEMORY /*Queries whether the **Memory** key is locked*/

Return Format
1 or 0

:SYSTem:KLOCk:STATe

Syntax
:SYSTem:KLOCk:STATe <state>
:SYSTem:KLOCk:STATe?

Description
Enables or disables the remote locking.
Queries the on/off state of the remote locking.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	OFF

Remarks
- When the remote locking is enabled, the power supply is locked in remote mode. Then,
  - all keys (except **On/Off** and ) and the knob on the front panel are locked;
  - you cannot press  on the front panel of the power supply to return to the local mode;
  - the icon  is displayed on the status bar of the user interface ( is not displayed);
- You can also send the :SYSTem:RWLock[:STATe] command to enable or disable the remote locking.
- When the remote locking is enabled, you can send the :SYSTem:LOCal command to return to the local mode. At this time, the remote locking will be disabled.

Example
:SYSTem:KLOCk:STATe ON /*Enables the remote locking*/
:SYSTem:KLOCk:STATe? /*Queries the on/off state of the remote locking*/

Return Format
ON or OFF

Related Commands
:SYSTem:RWLock[:STATe]
:SYSTem:LOCal

:SYSTem:LANGuage:TYPE

Syntax
:SYSTem:LANGuage:TYPE <type>
:SYSTem:LANGuage:TYPE?

Description
Selects the system language.
Queries the currently selected language.

Parameter

Name	Type	Range	Default
<type>	Discrete	{EN|CH}	--

Remarks
DP700 series power supply provides the help information, prompt message, and interface display in both Chinese and English version.

Example
:SYSTem:LANGuage:TYPE EN /*Selects English as the system language*/
:SYSTem:LANGuage:TYPE? /*Queries the currently selected language*/

Return Format
English or Chinese

:SYSTem:LOCal

Syntax
:SYSTem:LOCal

Description
Enables the power supply to shift from the remote mode to the local mode.

Remarks
- When the power supply is in the remote mode, sending this command will enable the power supply to shift from the remote mode to the local mode.
- When the power supply is locked in the remote mode, you can send the command to enable the power supply to return to the local mode. At this time, the remote locking will be disabled.
- After the power supply returns to the local mode, all the keys and the knob on the front panel are available to use. The icons  and  will not be displayed on the status bar of the user interface.

Related Commands
:SYSTem:KLOCk:STATe
:SYSTem:RWLock[:STATe]
:SYSTem:REMote

:SYSTem:LOCK

Syntax
:SYSTem:LOCK <state>
:SYSTem:LOCK?

Description
Locks or unlocks the front panel.
Queries whether the front panel is locked or not.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF|1|0}	OFF

Remarks
- DP700 series power supply provides the key locking functions to avoid any loss caused by misoperation.
- When the front panel is locked, it indicates that all keys (except **On/Off**,  , and ) and the knob on the front panel are locked. The icons  and  will be displayed on the status bar of the user interface.

Example
:SYSTem:LOCK ON /*Locks the front panel*/
:SYSTem:LOCK? /*Queries whether the front panel is locked or not*/

Return Format
ON or OFF

:SYSTem:POWEron

Syntax
:SYSTem:POWEron <state>
:SYSTem:POWEron?

Description
Selects the power-on setting (the configurations for the instrument when the instrument is powered on.)
Queries the currently selected power-on setting.

Parameter

Name	Type	Range	Default
<state>	Discrete	{DEFault|LAST}	DEFault

Remarks
<state>:
DEFault: restores defaults when the instrument is powered on. For default settings, refer to Appendix A: Default Settings.
LAST: restores its last state when the instrument is powered on.

Example
:SYSTem:POWEron DEFault /*Selects "DEFault" as the power-on setting*/
:SYSTem:POWEron? /*Queries the currently selected power-on setting*/

Return Format
DEFAULT or LAST

:SYSTem:PRINT?

Syntax
:SYSTem:PRINT?

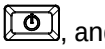
Description
Reads the data stream of the image currently displayed on the screen (screen shot).

Return Format
A string

:SYSTem:REMote

Syntax
:SYSTem:REMote

Description
Enables the power supply to shift from the local mode to the remote mode.

Remarks
When the power supply returns to the remote mode, all keys (except **On/Off**,  , and) and the knob on the front panel are locked. The icon  will be displayed on the status bar of the user interface.

Related Command
:SYSTem:LOCal

:SYSTem:RWLock[:STATe]

Syntax
:SYSTem:RWLock[:STATe] [<state>]

Description
Enables or disables the remote locking.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	OFF

Remarks
- When the remote locking is enabled, the power supply is locked in remote mode. Then,
  - all keys (except **On/Off** and ) and the knob on the front panel are locked;
  - you cannot press on the front panel of the power supply to return to the local mode;
  - the icon  is displayed on the status bar of the user interface (is not displayed).
- You can also send the :SYSTem:KLOCk:STATe command to enable or disable the remote locking.
- When the remote locking is enabled, you can send the :SYSTem:LOCal command to return to the local mode. At this time, the remote locking will be disabled.

Example
:SYSTem:RWLock ON /*Enables the remote locking*/

Related Commands
:SYSTem:KLOCk:STATe
:SYSTem:LOCal

:SYSTem:SAVer

Syntax
:SYSTem:SAVer <state>
:SYSTem:SAVer?

Description
Enables or disables the screen saver function.
Queries the on/off state of the screen saver function.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	OFF

Remarks
When the screen saver function is enabled, if no operation is performed on the front panel for more than 25 minutes, the instrument automatically enters the screen saver mode; if it persists for another 12.5 minutes, the screen of the instrument stays black.

Example
:SYSTem:SAVer ON /*Enables the screen saver function*/
:SYSTem:SAVer? /*Queries the on/off state of the screen saver function*/

Return Format
ON or OFF

:SYSTem:SELF:TEST?

Syntax
:SYSTem:SELF:TEST?

Description
Queries the self-test results of the instrument.

Remarks
- The power supply performs the self-test operation when it is powered on. Sending this command can query the self-test results of the instrument.
- You can also send the *TST? command to query the self-test results of the instrument.

Return Format
A string. For example, Fan:PASS

Related Commands
*TST?

:SYSTem:TRIGger:IN[:STATe]

Syntax
:SYSTem:TRIGger:IN[:STATe] <state>
:SYSTem:TRIGger:IN[:STATe]?

Description
Turns on or off the trigger input.
Queries the on/off state of the trigger input.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	OFF

Remarks

- Trigger input indicates that the external trigger input signal controls the on/off state of the channel output. Pin 8 of the RS232 interface on the rear panel is used to receive the external trigger input signal. When it receives a high level signal (≥ 2.1 V, 10 mA), the channel output will be turned on; when it receives a low level signal (≤0.7 V, 10 mA), the channel output will be turned off.
- Trigger (including the trigger input and output) is an optional function. If you want to use the function, order the trigger option (with the order No. TRIGGER-DP700) first, and then install the option according to the instructions in the :LIC:SET command.
- When multiple power supplies are in serial or parallel connection, using the external trigger function can realize the synchronous output for multiple power supplies.

Return Format

ON or OFF

Related Command

:SYSTem:TRIGger:OUT[:STATe]

:SYSTem:TRIGger:OUT[:STATe]

Syntax

:SYSTem:TRIGger:OUT[:STATe] <state>
:SYSTem:TRIGger:OUT[:STATe]?

Description

Turns on or off the trigger output.
Queries the on/off status of the trigger output.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	OFF

Remarks

- Trigger output indicates that controlling the on/off state of the channel output can enable the instrument to output the specified signal. Pin 7 of the RS232 interface on the rear panel is used to output the trigger output signals. When the channel output is turned on, the high level signal (≥2.1 V, 10 mA) is output from Pin 7; when off, the low level signal (≤0.7 V, 10 mA) is output from Pin 7.
- Trigger (including the trigger input and output) is an optional function. If you want to use the function, order the trigger option (with the order No. TRIGGER-DP700) first, and then install the option according to the instructions in the :LIC:SET command.
- When multiple power supplies are in serial or parallel connection, using the external trigger function can realize the synchronous output for multiple power supplies.

Return Format

ON or OFF

Related Command

:SYSTem:TRIGger:IN[:STATe]

:SYSTem:VERSion?

Syntax
:SYSTem:VERSion?

Description
Queries the SCPI version of the system.

Return Format
YYYY.V, for example, 1999.0
Wherein,
YYYY: indicates the version year.
V: indicates the number of edition times in the version year.

:TIMEr Commands

Command List:

- :TIMEr:CYCLEs
- :TIMEr:ENDState
- :TIMEr:GROUPs
- :TIMEr:PARAmeter
- :TIMEr[:STATe]
- :TIMEr:TRIGger

:TIMEr:CYCLEs

Syntax
:TIMEr:CYCLEs <cycles>
:TIMEr:CYCLEs?

Description
Sets the number of cycles for the timer.
Queries the number of cycles selected for the timer.

Parameter

Name	Type	Range	Default
<cycles>	Discrete	{N[,<value>]|I}	N,1
<value>	Integer	1 to 99999	1

Remarks
- "Cycles" refers to the number of cycle times that the instrument performs timing output according to the preset voltage and current values.
- <cycles>:
  N[,<value>]: sets "cycles" to a finite value. The specified value of the parameter is determined by <value>. When <value> is omitted, it indicates that "cycles" is, by default, set to 1.
  I: indicates setting "cycles" to Infinite.
- Total Number of Output Groups for the Timer = Outp Groups x Cycles. Wherein, for "Outp Groups", refer to :TIMEr:GROUPs.
- After the instrument has completed outputting operation for a specified number of times (total number of output groups), the timing output will be terminated. At this time, the state of the instrument is determined by the return value that you have set. When "cycles" is set to I (Infinite), "End State" is invalid.

Example
```
:TIMEr:CYCLEs N,20     /*Sets "cycles" to 20*/
:TIMEr:CYCLEs?         /*Queries the number of cycles selected for the timer*/
```

Return Format
N,<value> or I
Wherein, <value> is an integer between 1 and 99999.

Related Commands
:TIMEr:GROUPs
:TIMEr:ENDState

:TIMEr:ENDState

Syntax
:TIMEr:ENDState <state>
:TIMEr:ENDState?

Description
Sets the end state of the timer.
Queries the end state selected for the timer.

Parameter

Name	Type	Range	Default
<state>	Discrete	{OFF|LAST}	OFF

Remarks
- "End State" refers to the state of the instrument after it has completed outputting groups of voltage and current values for a specified number of times (the total number of output groups) when "cycles" is N[,<value>] (a finite value). When "cycles" is set to I (Infinite), "End State" is invalid.
- <state>:
  OFF: indicates that the output will be turned off automatically after the output is completed.
  LAST: keeps the output state (voltage/current) of the last group after the output is completed.
- Total Number of Output Groups for the Timer = Outp Groups x Cycles. Wherein, for "Outp Groups", refer to the :TIMEr:GROUPs command; for "Cycles", refer to the :TIMEr:CYCLEs command.

Example
:TIMEr:ENDState LAST /*Sets the end state of the timer to "LAST"*/
:TIMEr:ENDState? /*Queries the end state selected for the timer*/

Return Format
OFF or LAST

Related Commands
:TIMEr:CYCLEs
:TIMEr:GROUPs

:TIMEr:GROUPs

Syntax
:TIMEr:GROUPs <value>
:TIMEr:GROUPs?

Description
Sets the number of output groups.
Queries the number of output groups selected for the timer.

Parameter

Name	Type	Range	Default
<value>	Integer	1 to 2048	1

Remarks
- "Outp Groups" indicates the number of groups of the preset voltage and current values that the instrument outputs in each cycle.
- Total Number of Output Groups for the Timer = Outp Groups x Cycles. Wherein, for "Outp Groups", refer to :TIMEr:GROUPs.
- After the instrument has completed outputting operation for a specified number of times (total number

of output groups), the timing output will be terminated. At this time, the state of the instrument is determined by the return value that you have set. When "cycles" is set to I (Infinite), "End State" is invalid.

Example
:TIMEr:GROUPs 25 /*Sets the number of output groups to 25*/
:TIMEr:GROUPs? /*Queries the number of output groups selected for the timer*/

Return Format
An integer between 1 and 2048

Related Commands
:TIMEr:GROUPs
:TIMEr:ENDState
:TIMEr:CYCLEs

:TIMEr:PARAmeter

Syntax
:TIMEr:PARAmeter <num>,<voltage>,<current>,<time>
:TIMEr:PARAmeter? <firnum>[,<count>]

Description
Sets the timing parameters for a specified group.
Queries the timing parameters.

Parameter

Name	Type	Range	Default
<num>	Integer	1 to 2048	1
<voltage>	Real	0 V to 32 V	1 V
<current>	Real	0 A to 5.3 A	1 A
<time>	Real	10 ms to 99999 s	1 s
<firnum>	Integer	1 to 2048	--
<count>	Integer	1 to 2048	--

Remarks
- The timing parameters are used to control the output of the instrument. The parameters consist of the group ID, the output voltage, the output current, and duration time.
  <num>: indicates the group ID.
  <voltage>: indicates the output voltage. The default unit is V.
  <current>: indicates the output current. The default unit is A.
  <time>: indicates the duration time. The default unit is s.
- <firnum>: indicates the start group ID of the timing parameters to be queried.
  <count>: indicates the number of groups of the timing parameters to be queried. When <count> is omitted, it indicates that the number of groups of the timing parameters is set to 1.

Example
:TIMEr:PARAmeter 1,8,2,10 /*Sets the first group of timing parameters to 8 V, 2 A, and 10 s*/
:TIMEr:PARAmeter? 1,2 /*Queries two groups of the timing parameters beginning from the first group (i.e. Group 1 and Group 2)*/

Return Format
A string beginning with **#**
The string is composed of a data block header and the specified timing parameters. For example, #90000000361,20.00,2.00,5.00;2,18.00,1.80,3.00;

- The data block header describes the data length, beginning with a symbol #. For example, in the above string, #9000000036 indicates the data block header. The 9 figures following the symbol # indicate the 9-digit data (000000036), which describes the length of the data stream (36 bytes).
- Each group of the timing parameters is in the format of "group ID,output voltage,output current,duration time". Each group of parameters is separated by a semi-colon. In the above example, 1,20.00,2.00,5.00;2,18.00,1.80,3.00; indicates the specific timing parameters, that is, two groups of timing parameters are listed here. For the first group, the group ID is 1; the output voltage is 20 V, the output current is 2 A, and the duration time is 5 s. For the second group, the group ID is 2; the output voltage is 18 V, the output current is 1.8 A, and the duration time is 3 s.

:TIMEr[:STATe]

Syntax
:TIMEr[:STATe] <state>
:TIMEr[:STATe]?

Description
Enables or disables the timing output.
Queries the on/off state of the timing output.

Parameter

Name	Type	Range	Default
<state>	Bool	{ON|OFF}	OFF

Remarks
- Enabling the timing output will modify the channel output state. Therefore, before enabling the timing output, ensure that the change of the channel output state will not affect the devices that are connected to the power supply.
- Complete timer parameter settings before enabling the timing output. During the timing output, the timer parameters are not allowed to be modified.

Example
:TIMEr ON /*Enables the timing output*/
:TIMEr? /*Queries the on/off state of the timing output*/

Return Format
ON or OFF

:TIMEr:TRIGger

Syntax
:TIMEr:TRIGger <mode>
:TIMEr:TRIGger?

Description
Sets the trigger mode of the timer (timing output mode).
Queries the trigger mode selected for the timer.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{DEFault|SINGle}	DEFault

Remarks
<mode>:

DEFault: when the timing output is enabled, the instrument automatically outputs based on the parameter configuration of the timer.
SINGle: when the timing output is enabled, one single press on the **OK** enables a single output based on a group of timing parameters, until the instrument has completed outputting operation for a specified number of times (total number of output groups).

Example

```
:TIMEr:TRIGger DEFault    /*Sets the trigger mode of the timer to "DEFault"*/
:TIMEr:TRIGger?           /*Queries the trigger mode selected for the timer*/
```

Return Format

DEFAULT or SINGLE

Chapter 3 Application Instances

This chapter provides the application instances of the SCPI commands. The main functions of the power supply can be realized through a series of SCPI commands.

Note:

1. The instances in this chapter take DP711 as an example. The range of certain parameters for other models may be different. Therefore, you need to adjust the parameter range for the model that you use if necessary.
2. Before using the instances in this chapter, refer to To Build Remote Communication to set up remote communication between the power supply and the PC. In addition, you have to install Ultra Sigma or other PC software that can be used to send commands.
3. In each instance, every command is followed by contents enclosed by two slashes ("/*" and "*/"). They are the descriptions of the command and not part of the command, which help you understand the command better.

Contents in this chapter:

- Constant Voltage Output
- Timing Output

Constant Voltage Output

Requirement

Functions to be realized: constant voltage output, the output voltage is 30 V, the output current is 2 A, the overcurrent protection function is enabled and the overcurrent protection value is 2 A.

Method 1:

(1) *IDN? /*Queries the identification string of the power supply to test whether the remote communication works normally*/
(2) :APPLy CH1,30,2 /*Sets the channel output voltage to 30 V, the output current 2 A*/
(3) :OUTPut:OCP:VALue 2 /*Sets the channel overcurrent protection value to 2 A*/
(4) :OUTPut:OCP:STATe CH1,ON /*Enables the overcurrent protection function*/
(5) :OUTPut:STATe CH1,ON /*Turns on the channel output*/

Method 2:

(1) *IDN? /*Queries the identification string of the power supply to test whether the remote communication works normally*/
(2) :SOURce1:VOLTage:LEVel:IMMediate:AMPLitude 30 /*Sets the channel output voltage to 30 V*/
(3) :SOURce1:CURRent:LEVel:IMMediate:AMPLitude 2 /*Sets the channel output current to 2 A*/
(4) :SOURce1:CURRent:PROTection:LEVel 2 /*Sets the channel overcurrent protection value to 2 A*/
(5) :SOURce1:CURRent:PROTection:STATe ON /*Enables the overcurrent protection function*/
(6) :OUTPut:STATe CH1,ON /*Turns on the channel output*/

Timing Output

Requirement

Functions to be realized:

1. Timing parameters: Output Groups is 3, Cycles is 2, Trig Mode is Auto, End State is Last. The three groups of timing parameters are as follows:

Group ID	Output Voltage	Output Current	Duration Time
1	10 V	1 A	3 s
2	12 V	1.2 A	5 s
3	18 V	1.8 A	3.5 s

2. Save the edited timing parameters.
3. Enable the timing output.

Method

(1) *IDN? /*Queries the identification string of the power supply to test whether the remote communication works normally*/
(2) :TIMEr:GROUPs 3 /*Sets the number of output groups to 3*/
(3) :TIMEr:CYCLEs N,2 /*Sets the number of cycles for the timer to 2*/
(4) :TIMEr:TRIGer DEFault /*Sets the trigger mode of the timer to "Auto"*/
(5) :TIMEr:ENDState LAST /*Sets the end state of the timer to "Last"*/
(6) :TIMEr:PARAmeter 1,10,1,3 /*Sets the first group of timing parameters to 10 V, 1 A, and 3 s*/
(7) :TIMEr:PARAmeter 2,12,1.2,5 /*Sets the second group of timing parameters to 12 V, 1.2 A, and 5 s*/
(8) :TIMEr:PARAmeter 3,18,1.8,3.5 /*Sets the third group of timing parameters to 18 V, 1.8 A, and 3.5 s*/
(9) :MEMory:STATe:STORe RTF,1 /*Saves the current timer parameters to the state file storage location Timer1, with the file name Rigol1*/
(10) :TIMEr:STATe ON /*Enables the timing output*/

Chapter 4 Programming Examples

This chapter lists some examples by programming based on NI-VISA in the LabVIEW and Visual C# to remotely control the power supply.

NI-VISA (National Instrument-Virtual Instrument Software Architecture), developed by NI (National Instrument), provides an advanced programming interface to communicate with various instruments through their bus lines. NI-VISA enables you to communicate with the instrument in the same way, without considering the interface type of the instrument.

Contents in this chapter:

- Programming Preparations
- LabVIEW Programming
- Visual C# Programming Example

Programming Preparations

Before programming, you need to prepare the following tasks:

1. Install Ultra Sigma (PC) software. You can download Ultra Sigma from the official website of **RIGOL** (www.rigol.com) and install the software according to the installation wizard. After Ultra Sigma is installed successfully, NI-VISA library will be completely installed automatically. In this manual, the default installation path is C:\Program Files\IVI Foundation\VISA.

2. Use the 9-pin RS232 cable (female-to-female, straight-through) to connect the power supply to the PC via the RS232 interface on the rear panel. Then, connect to power and power on the power supply.

   Note: The RS232 protocol command ends with "\n" for the DP700 series.

3. Set the parameters for the RS232 interface of the power supply. The parameter settings for the RS232 interface are as follows:
   - Baud Rate: 9600
   - Stop Bits: 8
   - Data Bits: 1
   - Parity: None

Then, the preparation work for the programming is completed.

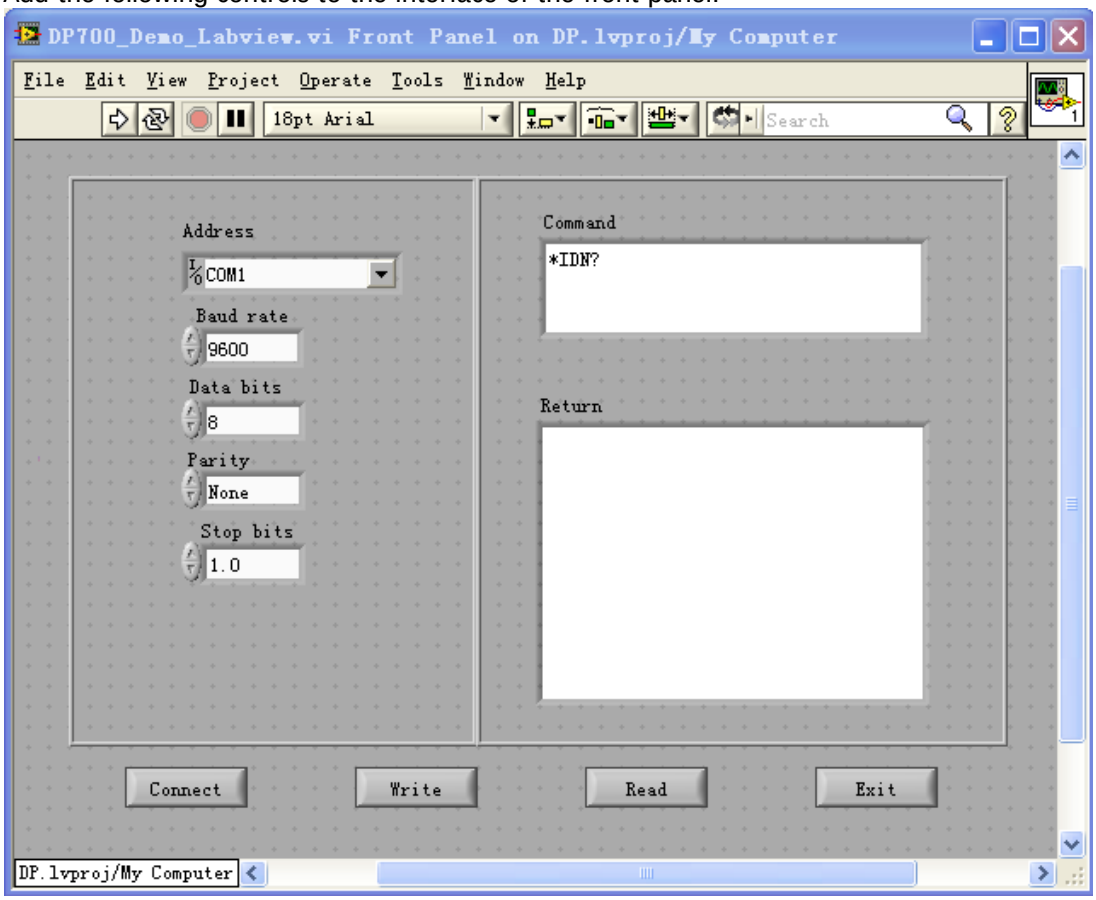
LabVIEW Programming Example

Program used in this instance: LabVIEW 2010

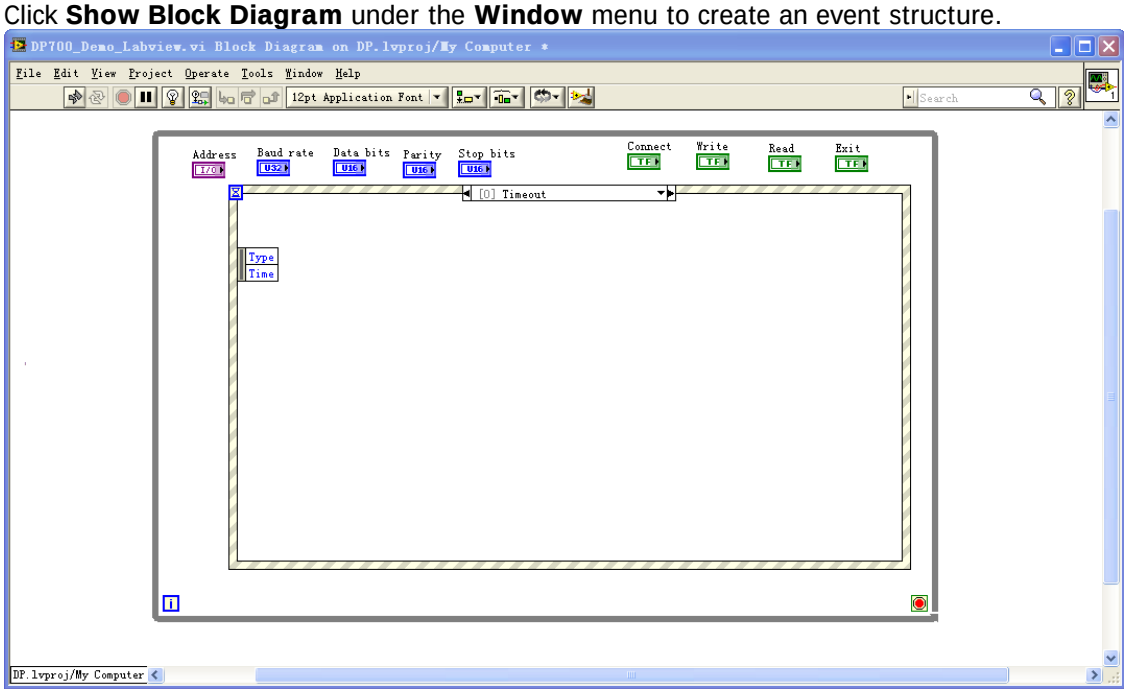
Function realized in this instance: search for the instrument address, connect to the instrument, send commands and read return values.

1. Run LabVIEW 2010, and then create a VI file named DP700_Demo_Labview.

2. Add the following controls to the interface of the front panel.

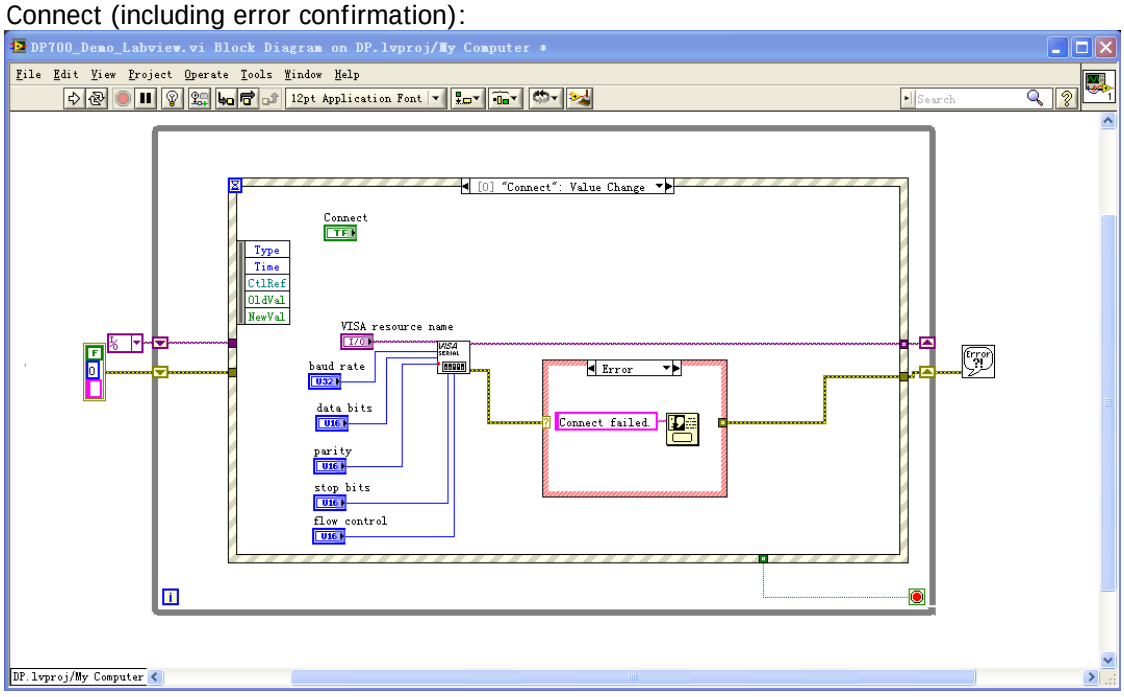


3. Click **Show Block Diagram** under the **Window** menu to create an event structure.

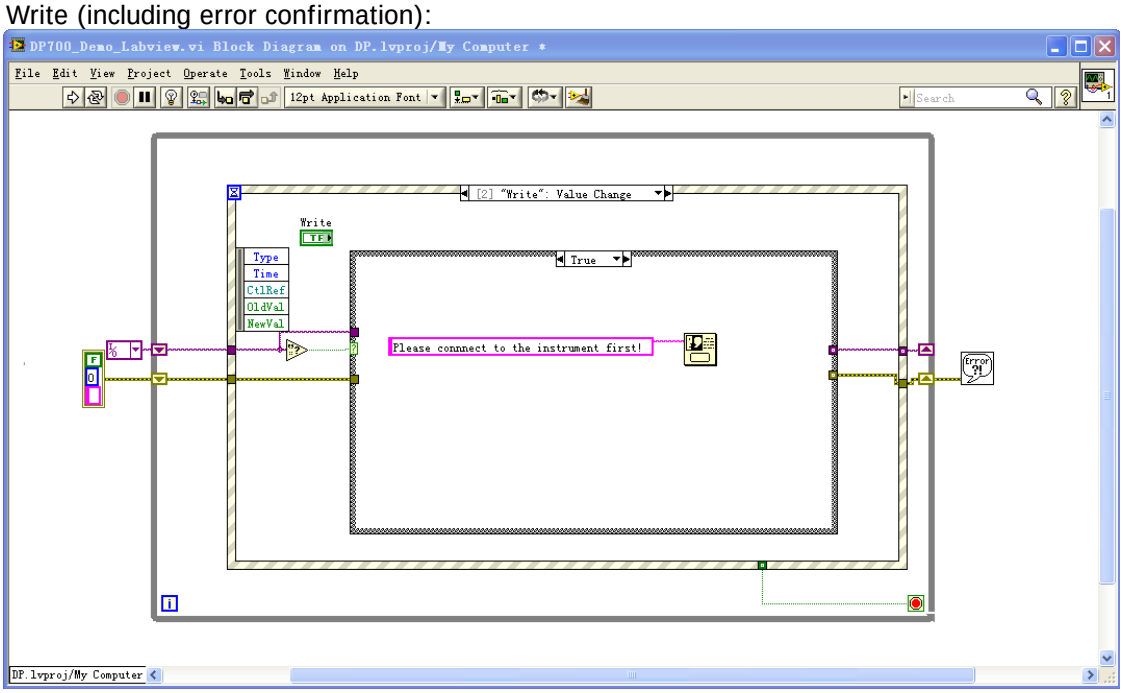


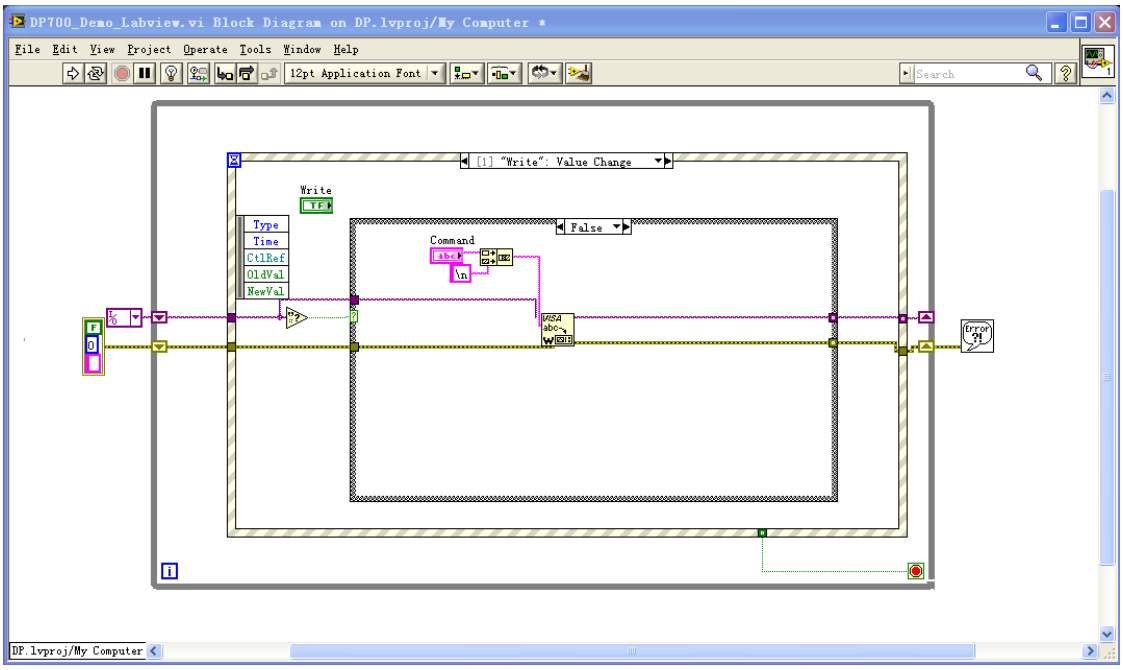
4. Add an event, including Connect, Writing, Read, and Exit.

(1) Connect (including error confirmation):

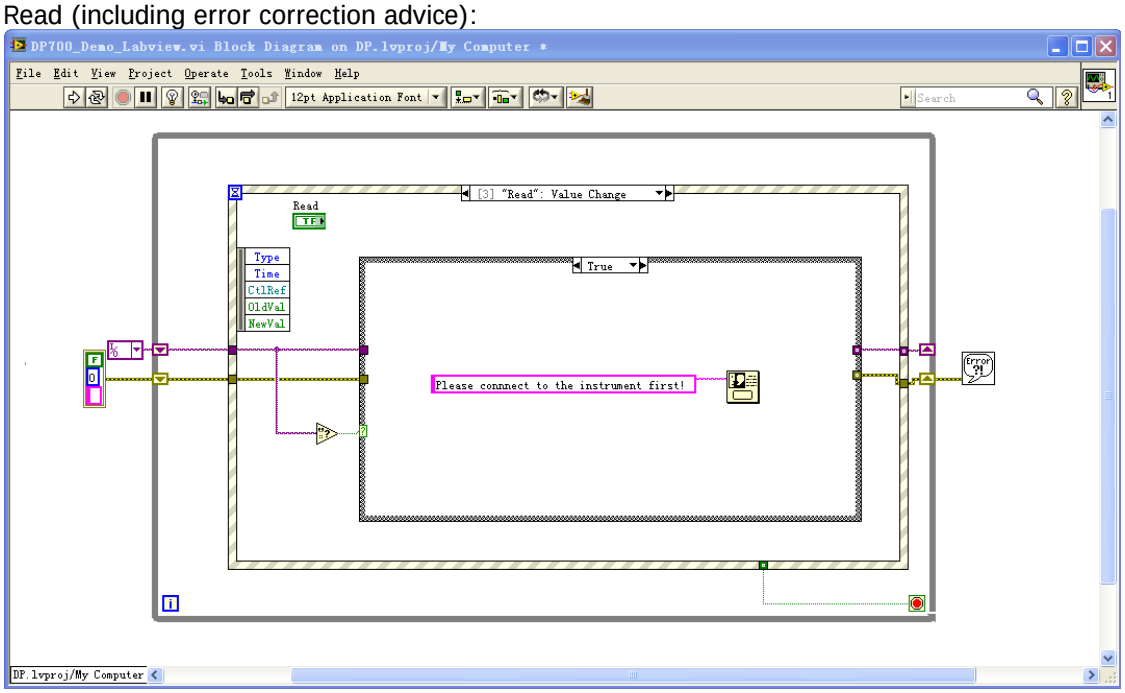


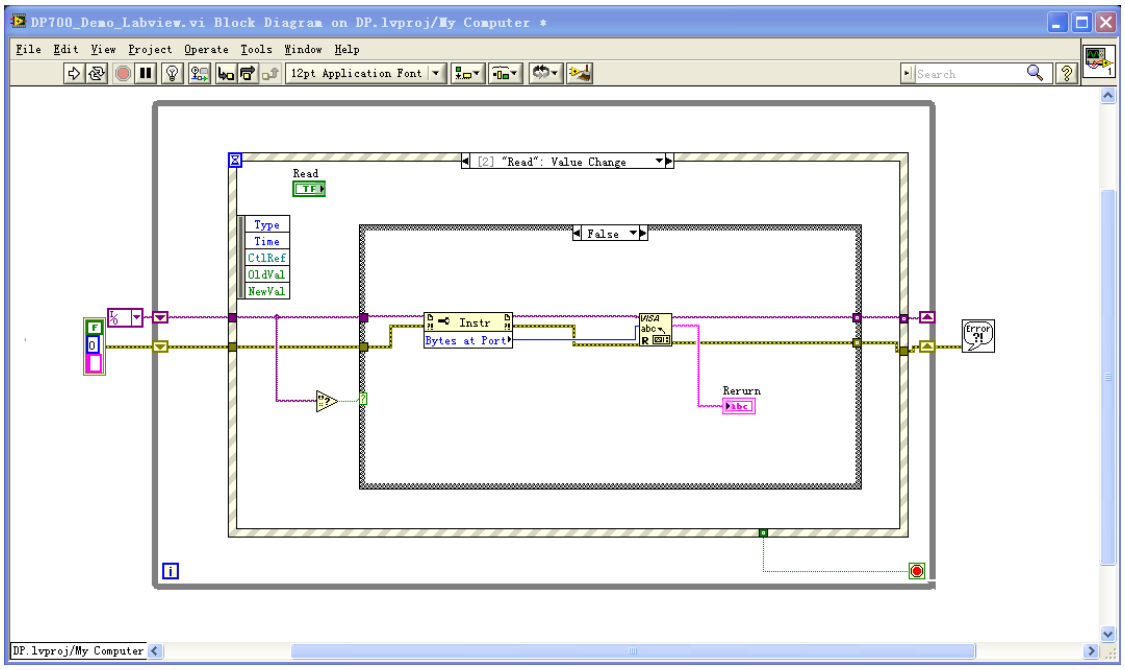
(2) Write (including error confirmation):



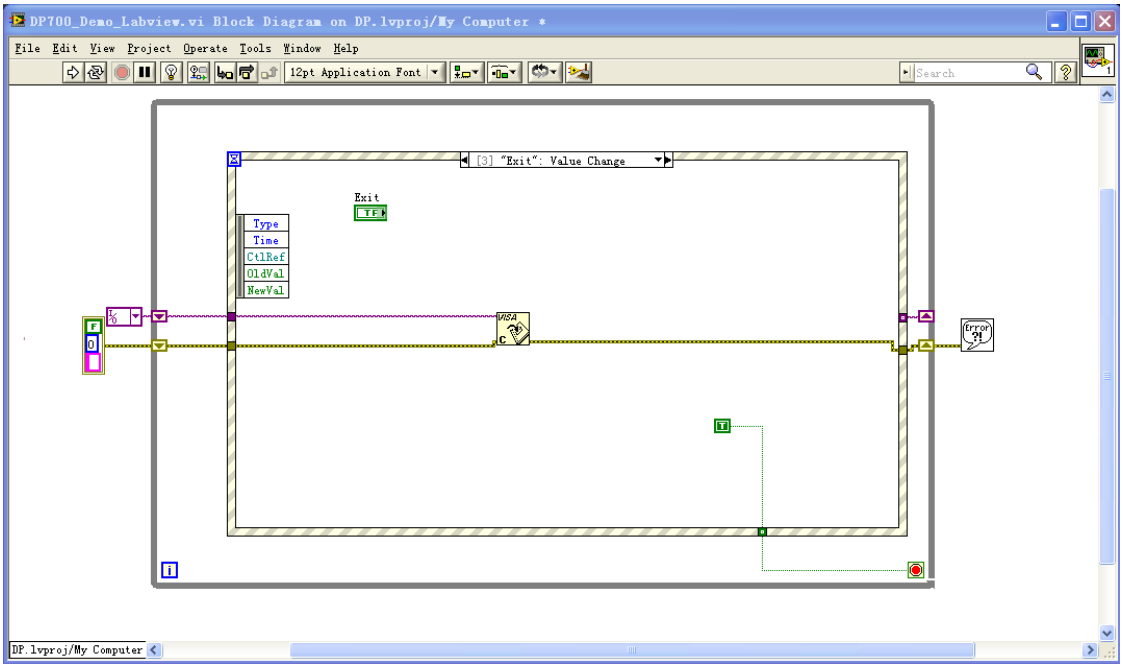


(3) Read (including error correction advice):

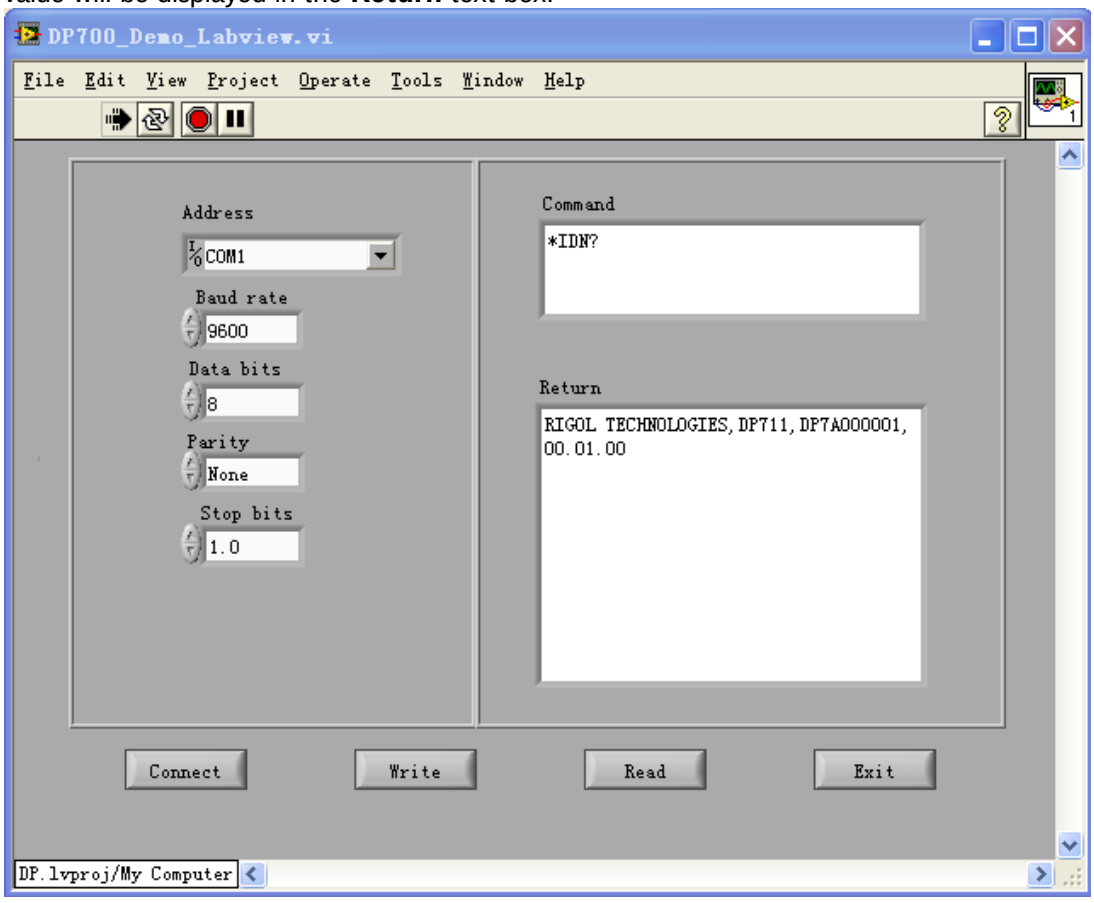




(4) Exit:



5. Run the program, and set the parameters for the RS232 interface. Click **Connect** to connect the instrument. Then, input a command into the **Command** text box, for example, *IDN?. Then, click **Write** to write the command to the instrument. If it is a query command, click **Read**. Then, the return value will be displayed in the **Return** text box.



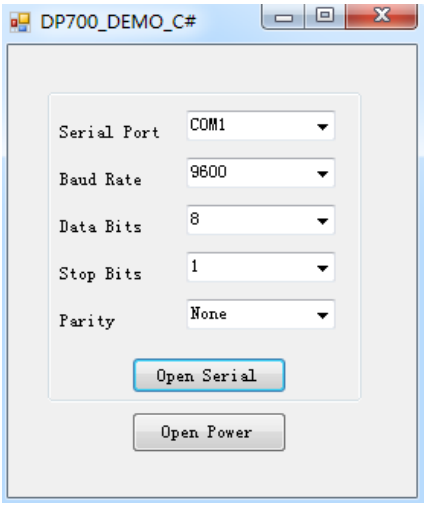
Visual C# Programming Example

Program used in this instance: Microsoft Visual Studio 2010

Functions realized in this stance: turning on the channel output.

1. Run Microsoft Visual Studio 2010 and create a standard application project (WindowsFormsApplication) named DP700_DEMO_C#.

2. Add "serialPort" and other controls.



3. Click "COM1" to add an event "comBox_MouseClick".

```
/// <summary>
/// Available serial port
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void comBox_MouseClick(object sender, MouseEventArgs e)
{
    bool bcomExist = false;
    comBox.Items.Clear();
    for (int i = 0; i < 15; i++)
    {
        try
        {
            SerialPort sp = new SerialPort("COM" + (i + 1).ToString());
            sp.Open();
            sp.Close();
            comBox.Items.Add("COM" + (i + 1).ToString());
            bcomExist = true;
        }
        catch (Exception)
        {
            continue;
        }
    }
    if (bcomExist)
    {
        comBox.SelectedIndex = 0;
    }
    else
```

```
        {
            MessageBox.Show("No Available Serial Port Found!") ;
        }
    }
```

4. Click "Open Power" to add the following program.

```
    /// <summary>
    /// Turn on the power output
    /// </summary>
    /// <param name="sender"></param>
    /// <param name="e"></param>
    private void button_Click(object sender, EventArgs e)
    {
        //Turn on/off the output
        if (bOpen)
        {
            bOpen = false;
            serialPort.Write(":OUTPut:STATe CH1,OFF\n");
            button.Text = "Open Power";
        }
        else
        {
            bOpen = true;
            serialPort.Write(":OUTPut:STATe CH1,ON\n");
        button.Text = "Close Power";
        }
    }
```

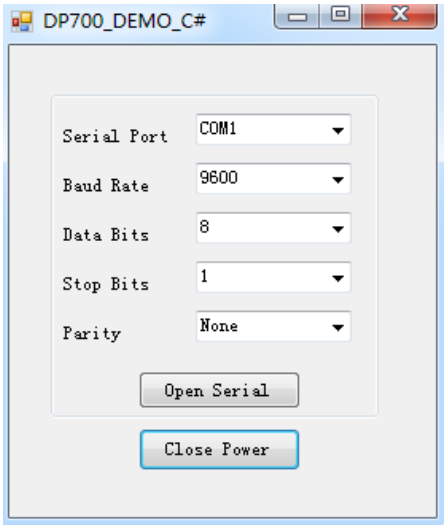
5. Double-click the window to enter the programming environment and add the following codes.

```
    /// <summary>
    /// Open the serial port
    /// </summary>
    /// <param name="sender"></param>
    /// <param name="e"></param>
    private void button1_Click(object sender, EventArgs e)
    {
        if (serialPort.IsOpen)
        {
            serialPort.Close();
        }
        try
        {
            serialPort.PortName = comBox.Text.Trim();              //Serial Port
            serialPort.BaudRate = int.Parse(baudBox.Text);         //Baud Rate
            serialPort.DataBits = int.Parse(dataBox.Text);         //Data Bits
        }
        catch (Exception)
        {
            MessageBox.Show("Error!");
        }
        //Initialize the stop bits
        float fstopValue = Convert.ToSingle(stopBox.Text.Trim());
        if (fstopValue == 0)
        {
            serialPort.StopBits = StopBits.None;
        }
        else if (fstopValue == 1)
        {
```

```
            serialPort.StopBits = StopBits.One;
        }
        //Initialize the parity
        if (checkBox.Text.Trim() == "None")
        {
            serialPort.Parity = Parity.None;
        }
        else if (checkBox.Text.Trim() == "Odd")
        {
            serialPort.Parity = Parity.Odd;
        }
        else
        {
            serialPort.Parity = Parity.Even;
        }
        try
        {
            serialPort.Open();
        }
        catch (Exception)
        {
            MessageBox.Show("Error!");
        }
    }
```

6. Run the results.
   (1) Select a serial port from the "COM1" drop-down list box.
   (2) Click "Open Serial".
   (3) Click "Open Power".

   Run the results: the output of the power supply is turned on. The "Open Power" button is changed to the "Close Power" button.



Chapter 5 Appendix

Appendix A: Default Settings

Sending the *RST command can restore the instrument to default settings, as shown in the table below.

Model	DP711	DP712
Channel Parameter		
Voltage/Current Setting Value	0 V/5 A	0 V/3 A
OVP/OCP Value	32 V/5.3 A	53 V/3.2 A
OVP/OCP On/Off	Off/Off	Off/Off
Output On/Off	Off	Off
System Setting		
Beeper	On	
Screen Saver	Off	
Trigger Input	Off	
Trigger Output	Off	
RS232 Interface Setting		
Baud Rate	9600	
Stop Bit	1	
Data Bit	8	
Parity	None	
Timer		
Number of Output Groups	1	
Number of Cycles	1	
Trigger Mode	Auto	
End State	Output Off	
Timing Parameter	Group ID: 1 Output Voltage: 1 V Output Current: 1 A Duration Time: 1 s	

Tip
When the power-on setting is set to "DEFault" (:SYSTem:POWEron DEFault), it indicates that the instrument will be restored to defaults once the instrument is powered on again.

Appendix B: Warranty

RIGOL TECHNOLOGIES, INC. (hereinafter referred to as **RIGOL**) warrants that the product will be free from defects in materials and workmanship within the warranty period. If a product proves defective within the warranty period, **RIGOL** guarantees free replacement or repair for the defective product.

To get repair service, please contact with your nearest **RIGOL** sales or service office.

There is no other warranty, expressed or implied, except such as is expressly set forth herein or other applicable warranty card. There is no implied warranty of merchantability or fitness for a particular purpose. Under no circumstances shall **RIGOL** be liable for any consequential, indirect, ensuing, or special damages for any breach of warranty in any case.