

# **RIGOL**

## **编程手册**

### **DSA700 系列频谱分析仪**

2016 年 8 月

**RIGOL TECHNOLOGIES, INC.**



## 保证和声明

### 版权

© 2016 北京普源精电科技有限公司

### 商标信息

**RIGOL** 是北京普源精电科技有限公司的注册商标。

### 文档编号

PGD18001-1110

### 软件版本

DSA705/DSA710: 00.01.17

软件升级可能更改或增加产品功能，请关注 **RIGOL** 网站获取最新版本手册或联系 **RIGOL** 升级软件。

### 声明

- 本公司产品受中国及其它国家和地区的专利（包括已取得的和正在申请的专利）保护。
- 本公司保留改变规格及价格的权利。
- 本手册提供的信息取代以往出版的所有资料。
- 本手册提供的信息如有变更，恕不另行通知。
- 对于本手册可能包含的错误，或因手册所提供的信息及演绎的功能以及因使用本手册而导致的任何偶然或继发的损失，**RIGOL** 概不负责。
- 未经 **RIGOL** 事先书面许可，不得影印、复制或改编本手册的任何部分

### 产品认证

**RIGOL** 认证本产品符合中国国家产品标准和行业产品标准及 ISO9001:2008 标准和 ISO14001:2004 标准，并进一步认证本产品符合其它国际标准组织成员的相关标准。

### 联系我们

如您在使用此产品或本手册的过程中有任何问题或需求，可与 **RIGOL** 联系：

电子邮箱：service@rigol.com

网址：www.rigol.com

# 安全要求

## 一般安全概要

了解下列安全性预防措施，以避免受伤，并防止损坏本产品或与本产品连接的任何产品。为避免可能的危险，请务必按照规定使用本产品。

### 使用正确的电源线。

只允许使用所在国家认可的本产品专用电源线。

### 将产品接地。

本产品通过电源电缆的保护接地线接地。为避免电击，在连接本产品的任何输入或输出端子之前，请确保本产品电源电缆的接地端子与保护接地端可靠连接。

### 正确连接探头。

如果使用探头，探头地线必须连接到接地端上。请勿将探头地线连接至高电压，否则，可能会在示波器和探头的连接器、控制设备或其它表面上产生危险电压，进而对操作人员造成伤害。

### 查看所有终端额定值。

为避免起火和过大电流的冲击，请查看产品上所有的额定值和标记说明，请在连接产品前查阅产品手册以了解额定值的详细信息。

### 使用合适的过压保护。

确保没有过电压（如由雷电造成的电压）到达该产品。否则操作人员可能有遭受电击的危险。

### 请勿开盖操作。

请勿在仪器机箱打开时运行本产品。

### 请勿将异物插入排风口。

请勿将异物插入排风口以免损坏仪器。

### 使用合适的保险丝。

只允许使用本产品指定规格的保险丝。

### 避免电路外露。

电源接通后，请勿接触外露的接头和元件。

### 怀疑产品出故障时，请勿进行操作。

如果您怀疑本产品出现故障，请联络RIGOL授权的维修人员进行检测。任何维护、调整或零件更换必须由RIGOL授权的维修人员执行。

### 保持适当的通风。

通风不良会引起仪器温度升高，进而引起仪器损坏。使用时应保持有良好的通风，定期检查通风口和风扇。

### 请勿在潮湿环境下操作。

为避免仪器内部电路短路或发生电击的危险，请勿在潮湿环境下操作仪器。

### 请勿在易燃易爆的环境下操作。

为避免仪器损坏或人身伤害，请勿在易燃易爆的环境下操作仪器。

### 请保持产品表面的清洁和干燥。

为避免灰尘或空气中的水分影响仪器性能，请保持产品表面的清洁和干燥。

**防静电保护。**

静电会造成仪器损坏，应尽可能在防静电区进行测试。在连接电缆到仪器前，应将其内外导体短暂接地以释放静电。

**正确使用电池。**

如果仪器提供电池，严禁将电池暴露于高温或火中。要让儿童远离电池。不正确地更换电池可能造成爆炸（警告：锂离子电池）。必须使用 **RIGOL** 指定的电池。

**注意搬运安全。**

为避免仪器在搬运过程中滑落，造成仪器面板上的按键、旋钮或接口等部件损坏，请注意搬运安全。

## 安全术语和符号

本手册中的安全术语：

**警告**

警告性声明指出可能会造成人身伤害或危及生命安全的情况或操作。

**注意**

注意性声明指出可能导致本产品损坏或数据丢失的情况或操作。

产品上的安全术语：

**DANGER** 表示您如果不进行此操作，可能会立即对您造成危害。

**WARNING** 表示您如果不进行此操作，可能会对您造成潜在的危害。

**CAUTION** 表示您如果不进行此操作，可能会对本产品或连接到本产品的其他设备造成损坏。

产品上的安全符号：



高电压



安全警告



保护性接地端



壳体接地端



测量接地端

## 文档概述

本手册用于指导用户使用 SCPI 命令通过 USB、LAN 或 GPIB（通过 USB 转 GPIB 扩展接口）接口编程控制 RIGOL DSA700 系列频谱分析仪。

### 文档的主要内容：

#### 第 1 章 SCPI 简介

本章简介 SCPI 命令。

#### 第 2 章 命令系统

本章以字母 A-Z 的顺序逐条介绍 DSA700 各命令的格式、功能、参数以及使用说明等。

#### 第 3 章 编程实例

本章给出如何使用 Visual C++、Visual Basic 以及 LabVIEW 等开发工具编程控制 DSA700。

#### 提示

本手册的最新版本可登陆[www.rigol.com](http://www.rigol.com)进行下载。

### 文档的格式约定：

#### 1. 按键：

本手册中通常用“文本框+文字（加粗）”表示前面板上的一个按键，如 **FREQ** 表示 FREQ 键。

#### 2. 菜单：

本手册通常用“字符底纹+文字（加粗）”表示一个菜单，如 **中心频率** 表示 FREQ 功能键的中心频率菜单项。

#### 3. 连接器：

本手册中通常用“方括号+文字（加粗）”表示前面板或后面板上的一个连接器。例如：**[GEN OUTPUT 50Ω]**。

#### 4. 操作步骤：

本手册中通常用一个箭头“→”表示下一步操作。例如：**FREQ** → **中心频率** 表示按下前面板上的 **FREQ** 功能键后再按 **中心频率** 菜单软键。

### 文档的内容约定：

DSA700 系列频谱分析仪包含以下两个型号。如无特殊说明，本手册以 DSA710 为例说明 DSA700 系列各命令的使用方法。

型号	频率范围
DSA705	100 kHz 至 500 MHz
DSA710	100 kHz 至 1 GHz

# 目录

保证和声明 .....	I
安全要求 .....	II
一般安全概要 .....	II
安全术语和符号 .....	IV
文档概述 .....	V
<b>第 1 章    SCPI 简介 .....</b>	<b>1-1</b>
命令格式 .....	1-2
符号说明 .....	1-2
参数类型 .....	1-3
命令缩写 .....	1-3
<b>第 2 章    命令系统 .....</b>	<b>2-1</b>
:ABORt .....	2-2
:CALCulate 命令子系统 .....	2-2
:CALCulate:BANDwidth:NDB .....	2-4
:CALCulate:BANDwidth:RESult? .....	2-4
:CALCulate:LLINe:ALL:DELete .....	2-4
:CALCulate:LLINe:CONTRol:DOMain .....	2-5
:CALCulate:LLINe:FAIL? .....	2-5
:CALCulate:LLINe:FAIL:RATIo? .....	2-6
:CALCulate:LLINe:FAIL:STOP:STATe .....	2-6
:CALCulate:LLINe<n>:CONTRol:INTerpolate:TYPE .....	2-7
:CALCulate:LLINe<n>:DATA .....	2-7
:CALCulate:LLINe<n>:DATA:MERGe .....	2-8
:CALCulate:LLINe<n>:DELete .....	2-9
:CALCulate:LLINe<n>:RELAmpt[:STATe] .....	2-9
:CALCulate:LLINe<n>:RELFreq[:STATe] .....	2-10
:CALCulate:LLINe<n>:STATe .....	2-10
:CALCulate:MARKer:AOFF .....	2-11
:CALCulate:MARKer:FCOunt:RESolution .....	2-11
:CALCulate:MARKer:FCOunt:RESolution:AUTO .....	2-12
:CALCulate:MARKer:FCOunt:X? .....	2-12
:CALCulate:MARKer:FCOunt[:STATe] .....	2-13
:CALCulate:MARKer<n>:CPEak[:STATe] .....	2-13
:CALCulate:MARKer<n>:DELTA[:SET]:CENTer .....	2-14
:CALCulate:MARKer<n>:DELTA[:SET]:SPAN .....	2-14
:CALCulate:MARKer<n>:FUNCTion .....	2-15
:CALCulate:MARKer<n>:MAXimum:LEFT .....	2-15
:CALCulate:MARKer<n>:MAXimum:MAX .....	2-16
:CALCulate:MARKer<n>:MAXimum:NEXT .....	2-16
:CALCulate:MARKer<n>:MAXimum:RIGHT .....	2-16
:CALCulate:MARKer<n>:MINimum .....	2-17
:CALCulate:MARKer<n>:MODE .....	2-17
:CALCulate:MARKer<n>:PEAK:EXCURsion .....	2-18
:CALCulate:MARKer<n>:PEAK:SEARch:MODE .....	2-18
:CALCulate:MARKer<n>:PEAK[:SET]:CF .....	2-19
:CALCulate:MARKer<n>:PEAK:THREshold .....	2-19
:CALCulate:MARKer<n>:PTPeak .....	2-20
:CALCulate:MARKer<n>[:SET]:CENTer .....	2-20
:CALCulate:MARKer<n>[:SET]:RLEVel .....	2-21
:CALCulate:MARKer<n>[:SET]:START .....	2-21
:CALCulate:MARKer<n>[:SET]:STEP .....	2-22



:CALCulate:MARKer<n>[:SET]:STOP .....	2-22
:CALCulate:MARKer<n>:STATe .....	2-23
:CALCulate:MARKer<n>:TRACe .....	2-23
:CALCulate:MARKer<n>:TRACe:AUTO .....	2-24
:CALCulate:MARKer<n>:X .....	2-25
:CALCulate:MARKer<n>:X:CENTer .....	2-25
:CALCulate:MARKer<n>:X:POSition .....	2-26
:CALCulate:MARKer<n>:X:POSition:CENTer .....	2-27
:CALCulate:MARKer<n>:X:POSition:SPAN .....	2-27
:CALCulate:MARKer<n>:X:POSition:START .....	2-28
:CALCulate:MARKer<n>:X:POSition:STOP .....	2-28
:CALCulate:MARKer<n>:X:READout .....	2-29
:CALCulate:MARKer<n>:X:SPAN .....	2-29
:CALCulate:MARKer<n>:X:START .....	2-30
:CALCulate:MARKer<n>:X:STOP .....	2-31
:CALCulate:MARKer<n>:Y? .....	2-31
:CALCulate:MARKer:TABLE:STATe .....	2-32
:CALCulate:MARKer:TRACKing:STATe .....	2-32
:CALibration 命令子系统 .....	2-33
:CALibration:[ALL] .....	2-33
:CALibration:AUTO .....	2-33
:CONFigure 命令子系统 .....	2-34
:CONFigure? .....	2-34
:CONFigure:ACPower .....	2-34
:CONFigure:CHPower .....	2-35
:CONFigure:CNRatio .....	2-35
:CONFigure:EBWidth .....	2-35
:CONFigure:HDISt .....	2-35
:CONFigure:OBWidth .....	2-36
:CONFigure:PF .....	2-36
:CONFigure:SANalyzer .....	2-36
:CONFigure:TOI .....	2-36
:CONFigure:TPOWER .....	2-37
:COUPle 命令子系统 .....	2-38
:COUPle .....	2-38
:DISPlay 命令子系统 .....	2-39
:DISPlay:AFUnction:POSition .....	2-40
:DISPlay:ANNotation:CLOCK[:STATe] .....	2-40
:DISPlay:BRIGhtness .....	2-41
:DISPlay:ENABle .....	2-41
:DISPlay:MSGswitch:STATe .....	2-42
:DISPlay:UKEY:STATe .....	2-42
:DISPlay:WINDow:TRACe:GRATICule:GRID .....	2-43
:DISPlay:WINDow:TRACe:X[:SCALE]:SPACing .....	2-43
:DISPlay:WINDow:TRACe:Y:DLINe .....	2-44
:DISPlay:WINDow:TRACe:Y:DLINe:STATe .....	2-44
:DISPlay:WINDow:TRACe:Y[:SCALE]:PDIVision .....	2-45
:DISPlay:WINDow:TRACe:Y[:SCALE]:RLEVel .....	2-45
:DISPlay:WINDow:TRACe:Y[:SCALE]:RLEVel:OFFSet .....	2-46
:DISPlay:WINDow:TRACe:Y[:SCALE]:SPACing .....	2-46
:FETCh 命令子系统 .....	2-47
:FETCh:ACPower? .....	2-48
:FETCh:ACPower:LOWer? .....	2-48
:FETCh:ACPower:MAIN? .....	2-49
:FETCh:ACPower:UPPer? .....	2-49
:FETCh:CHPower? .....	2-50
:FETCh:CHPower:CHPower? .....	2-50

:FETCh:CHPower:DENSity?	2-51
:FETCh:CNRatio?	2-51
:FETCh:CNRatio:CARRier?	2-52
:FETCh:CNRatio:CNRatio?	2-52
:FETCh:CNRatio:NOISe?	2-53
:FETCh:EBWidth?	2-53
:FETCh:HARMonics:AMPLitude:ALL?	2-54
:FETCh:HARMonics:AMPLitude? <n>	2-54
:FETCh:HARMonics[:DISTortion]?	2-55
:FETCh:HARMonics:FREQuency:ALL?	2-55
:FETCh:HARMonics:FREQuency? <n>	2-56
:FETCh:HARMonics:FUNDamental?	2-56
:FETCh:OBWidth?	2-57
:FETCh:OBWidth:OBWidth?	2-57
:FETCh:OBWidth:OBWidth:FERRor?	2-58
:FETCh:TOIntercept?	2-58
:FETCh:TOIntercept:IP3?	2-59
:FETCh:TPOWer?	2-59
:FORMat 命令子系统	2-60
:FORMat:BORDER	2-60
:FORMat[:TRACe][:DATA]	2-61
:HCOPy 命令子系统	2-62
:HCOPy:ABORT	2-62
:HCOPy:IMAGe:COLor[:STATe]	2-62
:HCOPy:IMAGe:FTYPe	2-63
:HCOPy:IMAGe:INVert	2-63
:HCOPy:IMAGe:PTIME	2-64
:HCOPy:IMAGe:QUALity	2-65
:HCOPy[:IMMediate]	2-65
:HCOPy:PAGE:ORientation	2-66
:HCOPy:PAGE:PRINTs	2-66
:HCOPy:PAGE:SIZE	2-67
:HCOPy:RESume	2-67
IEEE 488.2 公用命令	2-68
*CLS	2-68
*ESE	2-68
*ESR?	2-69
*IDN?	2-69
*OPC	2-70
*RST	2-70
*SRE	2-70
*STB?	2-71
*TRG	2-71
*TST?	2-71
*WAI	2-71
:INITiate 命令子系统	2-72
:INITiate:CONTInuous	2-72
:INITiate[:IMMediate]	2-73
:INITiate:PAUSE	2-73
:INITiate:REStart	2-73
:INITiate:RESume	2-73
:INPut 命令子系统	2-74
:INPut:IMPedance	2-74
:MMEMory 命令子系统	2-75
:MMEMory:DELeTe	2-75
:MMEMory:DISK:INFormation?	2-76
:MMEMory:LOAD:CORRection	2-76

:MMEMory:LOAD:LIMit .....	2-77
:MMEMory:LOAD:MTABLE .....	2-77
:MMEMory:LOAD:SETUp .....	2-78
:MMEMory:LOAD:STATe .....	2-78
:MMEMory:LOAD:TRACe .....	2-79
:MMEMory:MOVE .....	2-79
:MMEMory:STORe:CORRection .....	2-80
:MMEMory:STORe:LIMit .....	2-80
:MMEMory:STORe:MTABLE .....	2-81
:MMEMory:STORe:PTABLE .....	2-81
:MMEMory:STORe:RESults .....	2-82
:MMEMory:STORe:SCReen .....	2-82
:MMEMory:STORe:SETUp .....	2-83
:MMEMory:STORe:STATe .....	2-83
:MMEMory:STORe:TRACe .....	2-84
:READ 命令子系统 .....	2-85
:READ:ACPower? .....	2-86
:READ:ACPower:LOWer? .....	2-86
:READ:ACPower:MAIN? .....	2-86
:READ:ACPower:UPPer? .....	2-87
:READ:CHPower? .....	2-87
:READ:CHPower:CHPower? .....	2-87
:READ:CHPower:DENSity? .....	2-88
:READ:CNRatio? .....	2-88
:READ:CNRatio:CARRier? .....	2-88
:READ:CNRatio:CNRatio? .....	2-89
:READ:CNRatio:NOISe? .....	2-89
:READ:EBWidth? .....	2-89
:READ:HARMonics:AMPLitude:ALL? .....	2-90
:READ:HARMonics:AMPLitude? <n> .....	2-90
:READ:HARMonics[:DISTortion]? .....	2-91
:READ:HARMonics:FREQuency:ALL? .....	2-91
:READ:HARMonics:FREQuency? <n> .....	2-91
:READ:HARMonics:FUNDamental? .....	2-92
:READ:OBWidth? .....	2-92
:READ:OBWidth:OBWidth? .....	2-92
:READ:OBWidth:OBWidth:FERRor? .....	2-93
:READ:TOIntercept? .....	2-93
:READ:TOIntercept:IP3? .....	2-93
:READ:TPOWER? .....	2-94
[:SENSe]命令子系统 .....	2-95
[:SENSe]:ACPower:AVERage:COUNT .....	2-98
[:SENSe]:ACPower:AVERage[:STATe] .....	2-98
[:SENSe]:ACPower:AVERage:TCONtrol .....	2-99
[:SENSe]:ACPower:BANDwidth:ACHannel .....	2-99
[:SENSe]:ACPower:BANDwidth:INTegration .....	2-100
[:SENSe]:ACPower:CSPacing .....	2-101
[:SENSe]:BANDwidth:EMIFilter:STATe .....	2-101
[:SENSe]:BANDwidth[:RESolution] .....	2-102
[:SENSe]:BANDwidth[:RESolution]:AUTO .....	2-102
[:SENSe]:BANDwidth:VIDeo .....	2-103
[:SENSe]:BANDwidth:VIDeo:AUTO .....	2-103
[:SENSe]:BANDwidth:VIDeo:RATio .....	2-104
[:SENSe]:CHPower:AVERage:COUNT .....	2-104
[:SENSe]:CHPower:AVERage[:STATe] .....	2-105
[:SENSe]:CHPower:AVERage:TCONtrol .....	2-105
[:SENSe]:CHPower:BANDwidth:INTegration .....	2-106

[ :SENSe]:CHPower:FREQuency:SPAN .....	2-106
[ :SENSe]:CNRatio:AVERAge:COUNT .....	2-107
[ :SENSe]:CNRatio:AVERAge[:STATe] .....	2-108
[ :SENSe]:CNRatio:AVERAge:TCONtrol .....	2-108
[ :SENSe]:CNRatio:BANDwidth:INTegration .....	2-109
[ :SENSe]:CNRatio:BANDwidth:NOISe .....	2-110
[ :SENSe]:CNRatio:OFFSet .....	2-110
[ :SENSe]:CORRection:CSET:ALL:DELeTe .....	2-111
[ :SENSe]:CORRection:CSET:ALL[:STATe] .....	2-111
[ :SENSe]:CORRection:CSET<n>:DATA .....	2-112
[ :SENSe]:CORRection:CSET<n>:DATA:MERGe .....	2-112
[ :SENSe]:CORRection:CSET<n>:DELeTe .....	2-113
[ :SENSe]:CORRection:CSET<n>[:STATe] .....	2-113
[ :SENSe]:CORRection:CSET<n>:X:SPACing .....	2-114
[ :SENSe]:CORRection:CSET:TABLE:STATe .....	2-114
[ :SENSe]:DEMod .....	2-115
[ :SENSe]:DEMod:GAIN:AUTO .....	2-115
[ :SENSe]:DEMod:GAIN:INCRement .....	2-116
[ :SENSe]:DEMod:STATE .....	2-117
[ :SENSe]:DEMod:TIME .....	2-117
[ :SENSe]:DETector[:FUNCTion] .....	2-118
[ :SENSe]:EBWidth:AVERAge:COUNT .....	2-118
[ :SENSe]:EBWidth:AVERAge[:STATe] .....	2-119
[ :SENSe]:EBWidth:AVERAge:TCONtrol .....	2-120
[ :SENSe]:EBWidth:FREQuency:SPAN .....	2-120
[ :SENSe]:EBWidth:MAXHold:STATe .....	2-121
[ :SENSe]:EBWidth:XDB .....	2-122
[ :SENSe]:EXTRef[:STATe]? .....	2-122
[ :SENSe]:FREQuency:CENTer .....	2-123
[ :SENSe]:FREQuency:CENTer:DOWN .....	2-123
[ :SENSe]:FREQuency:CENTer:SET:STEP .....	2-123
[ :SENSe]:FREQuency:CENTer:STEP:AUTO .....	2-124
[ :SENSe]:FREQuency:CENTer:STEP[:INCRement] .....	2-124
[ :SENSe]:FREQuency:CENTer:UP .....	2-125
[ :SENSe]:FREQuency:OFFSet .....	2-125
[ :SENSe]:FREQuency:SPAN .....	2-125
[ :SENSe]:FREQuency:SPAN:FULL .....	2-126
[ :SENSe]:FREQuency:SPAN:PREVious .....	2-126
[ :SENSe]:FREQuency:SPAN:ZIN .....	2-126
[ :SENSe]:FREQuency:SPAN:ZOUT .....	2-126
[ :SENSe]:FREQuency:STARt .....	2-127
[ :SENSe]:FREQuency:STOP .....	2-127
[ :SENSe]:HDISt:AVERAge:COUNT .....	2-128
[ :SENSe]:HDISt:AVERAge[:STATe] .....	2-128
[ :SENSe]:HDISt:AVERAge:TCONtrol .....	2-129
[ :SENSe]:HDISt:NUMBers .....	2-129
[ :SENSe]:HDISt:TIME .....	2-130
[ :SENSe]:HDISt:TIME:AUTO[:STATe] .....	2-131
[ :SENSe]:OBWidth:AVERAge:COUNT .....	2-131
[ :SENSe]:OBWidth:AVERAge[:STATe] .....	2-132
[ :SENSe]:OBWidth:AVERAge:TCONtrol .....	2-132
[ :SENSe]:OBWidth:FREQuency:SPAN .....	2-133
[ :SENSe]:OBWidth:MAXHold:STATe .....	2-134
[ :SENSe]:OBWidth:PERCent .....	2-134
[ :SENSe]:POWER:ARANge .....	2-135
[ :SENSe]:POWER:ASCale .....	2-135
[ :SENSe]:POWER:ATUNe .....	2-135
[ :SENSe]:POWER[:RF]:ATTenuation .....	2-135

[:SENSe]:POWer[:RF]:ATTenuation:AUTO .....	2-136
[:SENSe]:POWer[:RF]:GAIN[:STATe] .....	2-137
[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer] .....	2-137
[:SENSe]:SIGCapture[:STATe] .....	2-138
[:SENSe]:SIGCapture:SIGC[:STATe] .....	2-138
[:SENSe]:SIGCapture:MAXHold[:STATe] .....	2-139
[:SENSe]:SIGCapture:RESet .....	2-139
[:SENSe]:SIGCapture:2FSK[:STATe] .....	2-139
[:SENSe]:SIGCapture:2FSK:RESet .....	2-140
[:SENSe]:SIGCapture:2FSK:MAXHold[:STATe] .....	2-140
[:SENSe]:SIGCapture:2FSK:PFSWitch .....	2-141
[:SENSe]:SIGCapture:2FSK:SIGNal .....	2-141
[:SENSe]:SIGCapture:2FSK:AMPUp .....	2-142
[:SENSe]:SIGCapture:2FSK:AMPDown .....	2-142
[:SENSe]:SIGCapture:2FSK:MARK1[:FREQ] .....	2-143
[:SENSe]:SIGCapture:2FSK:MARK1:Switch[:STATe] .....	2-143
[:SENSe]:SIGCapture:2FSK:MARK2[:FREQ] .....	2-144
[:SENSe]:SIGCapture:2FSK:MARK2:Switch[:STATe] .....	2-144
[:SENSe]:SWEep:COUNT .....	2-145
[:SENSe]:SWEep:COUNT:CURRent? .....	2-145
[:SENSe]:SWEep:TIME .....	2-146
[:SENSe]:SWEep:TIME:AUTO .....	2-146
[:SENSe]:SWEep:TIME:AUTO:RULEs .....	2-147
[:SENSe]:TOI:AVERAge:COUNT .....	2-147
[:SENSe]:TOI:AVERAge[:STATe] .....	2-148
[:SENSe]:TOI:AVERAge:TCONtrol .....	2-148
[:SENSe]:TOI:FREQuency:SPAN .....	2-149
[:SENSe]:TPOWer:AVERAge:COUNT .....	2-150
[:SENSe]:TPOWer:AVERAge[:STATe] .....	2-150
[:SENSe]:TPOWer:AVERAge:TCONtrol .....	2-151
[:SENSe]:TPOWer:LLIMit .....	2-151
[:SENSe]:TPOWer:MODE .....	2-152
[:SENSe]:TPOWer:RLIMit .....	2-153
:STATus 命令子系统 .....	2-154
:STATus:OPERation:CONDition? .....	2-155
:STATus:OPERation:ENABle .....	2-155
:STATus:OPERation[:EVENT]? .....	2-156
:STATus:PRESet .....	2-156
:STATus:QUEStionable:CONDition? .....	2-156
:STATus:QUEStionable:ENABle .....	2-157
:STATus:QUEStionable[:EVENT]? .....	2-157
:SYSTem 命令子系统 .....	2-158
:SYSTem:BEEPer:STATe .....	2-159
:SYSTem:CLEar .....	2-159
:SYSTem:COMMunicate:APORT .....	2-160
:SYSTem:COMMunicate:BRMT .....	2-160
:SYSTem:COMMunicate:GPIB[:SELF]:ADDReSS .....	2-161
:SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe .....	2-161
:SYSTem:COMMunicate:LAN[:SELF]:DHCP:STATe .....	2-162
:SYSTem:COMMunicate:LAN[:SELF]:IP:ADDReSS .....	2-162
:SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer .....	2-163
:SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway .....	2-163
:SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMAsk .....	2-164
:SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe .....	2-165
:SYSTem:COMMunicate:LAN[:SELF]:RESet .....	2-165
:SYSTem:COMMunicate:USB[:SELF]:ADDReSS? .....	2-165
:SYSTem:COMMunicate:USB[:SELF]:CLASs .....	2-166

:SYSTem:CONFigure:INFormation?	2-166
:SYSTem:CONFigure:MESSAge?	2-167
:SYSTem:DATE	2-167
:SYSTem:ERRor[:NEXT]?	2-168
:SYSTem:FSWItch[:STATe]	2-168
:SYSTem:KLOCK	2-169
:SYSTem:LANGUage	2-170
:SYSTem:LINemod:STATe?	2-170
:SYSTem:LINemod:TYPe	2-171
:SYSTem:LKEY	2-171
:SYSTem:OPTions?	2-172
:SYSTem:PON:TYPe	2-172
:SYSTem:PRESet	2-173
:SYSTem:PRESet:SAVE	2-173
:SYSTem:PRESet:TYPe	2-173
:SYSTem:SPEaker[:STATe]	2-174
:SYSTem:SPEaker:VOLume	2-174
:SYSTem:TIME	2-175
:SYSTem:TX:STATe?	2-175
:SYSTem:TX:SWset	2-176
:SYSTem:TX:SWSTa?	2-176
:SYSTem:USERkey:CONFirm	2-177
:SYSTem:USERkey:KEYCmd	2-177
:SYSTem:USERkey:STATe	2-178
:SYSTem:VERSion?	2-178
:TRACe 命令子系统	2-179
:TRACe:AVERAge:CLEar	2-179
:TRACe:AVERAge:COUNt	2-179
:TRACe:AVERAge:COUNt:CURRent?	2-180
:TRACe:AVERAge:RESet	2-180
:TRACe:CLEar:ALL	2-180
:TRACe[:DATA]	2-181
:TRACe:MATH:A	2-182
:TRACe:MATH:B	2-182
:TRACe:MATH:CONSt	2-183
:TRACe:MATH:PEAK[:DATA]?	2-183
:TRACe:MATH:PEAK:POINts?	2-184
:TRACe:MATH:PEAK:SORT	2-184
:TRACe:MATH:PEAK:TABLE:STATe	2-184
:TRACe:MATH:PEAK:THReshold	2-185
:TRACe:MATH:STATe	2-185
:TRACe:MATH:TYPe	2-186
:TRACe<n>:AVERAge:TYPe	2-186
:TRACe<n>:MODE	2-187
:TRIGger 命令子系统	2-188
:TRIGger:SEQuence:EXTernal:READy?	2-188
:TRIGger:SEQuence:EXTernal:SLOPe	2-188
:TRIGger:SEQuence:SOURce	2-189
:TRIGger:SEQuence:VIDeo:LEVel	2-190
:UNIT 命令子系统	2-191
:UNIT:POWer	2-191
<b>第 3 章 编程实例</b>	<b>3-1</b>
编程说明	3-2
编程准备	3-2
Visual C++ 6.0 编程实例	3-4
Visual Basic 6.0 编程实例	3-12

---

LabVIEW 8.6 编程实例 .....	3-16
Linux 编程实例 .....	3-19
Linux 编程准备 .....	3-19
Linux 编程步骤 .....	3-22





# 第1章 SCPI 简介

SCPI (Standard Commands for Programmable Instruments, 即可编程仪器标准命令集) 是一种建立在现有标准 IEEE488.1 和 IEEE 488.2 基础上, 并遵循了 IEEE754 标准中浮点运算规则、ISO646 信息交换 7 位编码符号 (相当于 ASCII 编程) 等多种标准的标准化仪器编程语言。

本章主要内容:

- ◆ [命令格式](#)
- ◆ [符号说明](#)
- ◆ [参数类型](#)
- ◆ [命令缩写](#)

## 命令格式

SCPI 命令为树状层次结构，包括多个子系统，每个子系统由一个根关键字和一个或数个层次关键字构成。命令行通常以冒号“:”开始；关键字之间用冒号“:”分隔，关键字后面跟随可选的参数设置；命令行后面添加问号“?”，表示对此功能进行查询；命令和参数以空格分开。

例如：

```
:CALCulate:BANDwidth:NDB <rel_ampl>  
:CALCulate:BANDwidth:NDB?
```

CALCulate 是命令的根关键字，BANDwidth 和 NDB 分别是第二级、第三级关键字。命令行以冒号“:”开始，同时将各级关键字分开，<rel\_ampl>表示可设置的参数；问号“?”表示查询；命令:CALCulate:BANDwidth:NDB 和参数<rel\_ampl>之间用空格分开。

在一些带参数的命令中，通常用逗号“,”分隔多个参数，例如：

```
:SYSTem:DATE <year>,<month>,<day>
```

## 符号说明

下面四种符号不是 SCPI 命令中的内容，不随命令发送，但是通常用于辅助说明命令中的参数。

### 大括号 { }

大括号中的参数是可选项，可以不设置，也可以设置一次或多次。例如：

```
[[:SENSe]:CORRection:CSET<n>:DATA <freq>,<rel_ampl>{,<freq>,<rel_ampl>}命令中，  
{,<freq>,<rel_ampl>}中的频率和幅度可以省略，也可以设置一对或多对频率、幅度参数。
```

### 竖线 |

竖线用于分隔多个参数选项，发送命令时必须选择其中一个参数。例如：

```
:DISPlay:ANNotation:CLOCK[:STATe] OFF|ON|0|1 命令中，可选择的命令参数为“OFF”、“ON”、“0”或“1”。
```

### 方括号 [ ]

方括号中的内容（命令关键字）是可省略的。例如：

对于[:SENSe]:POWer[:RF]:ATTenuation?命令，发送下面四条命令的效果是一样的：

```
:POWer:ATTenuation?  
:POWer:RF:ATTenuation?  
:SENSe:POWer:ATTenuation?  
:SENSe:POWer:RF:ATTenuation?
```

### 三角括号 < >

三角括号中的参数必须用一个有效值来替换。例如：

以:DISPlay:BRIGhtness 5 的形式发送:DISPlay:BRIGhtness <integer>命令。

## 参数类型

本手册介绍的命令中所含的参数可以分为以下 6 种类型：布尔型、关键字、整型、连续实型、离散型、ASCII 字符串。

### 布尔型

参数取值为“OFF”、“ON”、“0”或“1”。例如：

```
:DISPlay:ANNotation:CLOCK[:STATe] OFF|ON|0|1
```

### 关键字

参数取值为所列举的值。例如：

```
:DISPlay:AFUnction:POSition BOTTOm|CENTer|TOP
```

参数为“BOTTOm”、“CENTer”或“TOP”。

### 整型

除非另有说明，参数在有效值范围内可以取任意整数。注意：此时请不要设置参数为小数格式，否则将出现异常。例如：

```
:DISPlay:BRIGhtness <integer>
```

参数<integer>可取 1 到 10 范围内的任一整数。

### 连续实型

参数在有效值范围内按精度要求（通常默认精度为小数点以后取六位有效值），可以任意进行取值。例如：

```
:CALCulate:BANDwidth:NDB <rel_ampl>
```

参数<rel\_ampl>可取-100 到 100 之间的实数。

### 离散型

参数只能取指定的数值，并且这些数值不是连续的。例如：

```
:CALCulate:MARKer<n>:MAXimum:MAX
```

参数<n>只能取值为 1、2、3 或 4。

### ASCII 字符串

参数取值为 ASCII 字符的组合。例如：

```
:SYSTem:DATE <year>,<month>,<day>
```

参数为设定的日期格式字符串。

## 命令缩写

所有命令对大小写不敏感，你可以全部采用大写或小写。但是如果缩写，必须输完命令格式中的所有大写字母，例如：

```
:CALCulate:BANDwidth:NDB? 可缩写成：:CALC:BAND:NDB?
```



## 第2章 命令系统

本章按照字母 A-Z 顺序介绍 DSA700 系列频谱分析仪的命令系统。

本章主要内容：

- ◆ [:ABORt](#)
- ◆ [:CALCulate 命令子系统](#)
- ◆ [:CALibration 命令子系统](#)
- ◆ [:CONFigure 命令子系统](#)
- ◆ [:COUPle 命令子系统](#)
- ◆ [:DISPlay 命令子系统](#)
- ◆ [:FETCh 命令子系统](#)
- ◆ [:FORMat 命令子系统](#)
- ◆ [:HCOPy 命令子系统](#)
- ◆ [IEEE 488.2 公用命令](#)
- ◆ [:INITiate 命令子系统](#)
- ◆ [:INPut 命令子系统](#)
- ◆ [:MMEMory 命令子系统](#)
- ◆ [:READ 命令子系统](#)
- ◆ [\[:SENSe\]命令子系统](#)
- ◆ [:STATus 命令子系统](#)
- ◆ [:SYSTem 命令子系统](#)
- ◆ [:TRACe 命令子系统](#)
- ◆ [:TRIGger 命令子系统](#)
- ◆ [:UNIT 命令子系统](#)

说明：

1. 本命令集中有关准峰值检波、EMI 滤波、高级测量、Sig Capture 和 TX1000 的相关命令仅适用于已安装相关选件的 DSA700，具体请见每个子命令系统中的说明。
2. 在本命令集中，如无特殊说明，所查询功能对应的选件未安装时，将返回“N/A”（不含引号），而当所查询的功能未开启或者类型不匹配时，则返回“ERR”（不含引号）。
3. 本手册以 DSA710 为例说明命令中各参数的范围。

## :ABORt

### 命令格式

:ABORt

### 功能描述

放弃当前操作，重新扫频。

## :CALCulate命令子系统

### 命令列表:

- ◆ [:CALCulate:BANDwidth:NDB](#)
- ◆ [:CALCulate:BANDwidth:RESult?](#)
- ◆ [:CALCulate:LLINe:ALL:DELeTe](#)
- ◆ [:CALCulate:LLINe:CONTRol:DOMain](#)
- ◆ [:CALCulate:LLINe:FAIL?](#)
- ◆ [:CALCulate:LLINe:FAIL:RATIo?](#)
- ◆ [:CALCulate:LLINe:FAIL:STOP:STATe](#)
- ◆ [:CALCulate:LLINe<n>:CONTRol:INTerpolate:TYPE](#)
- ◆ [:CALCulate:LLINe<n>:DATA](#)
- ◆ [:CALCulate:LLINe<n>:DATA:MERGe](#)
- ◆ [:CALCulate:LLINe<n>:DELeTe](#)
- ◆ [:CALCulate:LLINe<n>:RELAmpt\[:STATe\]](#)
- ◆ [:CALCulate:LLINe<n>:RELFreq\[:STATe\]](#)
- ◆ [:CALCulate:LLINe<n>:STATe](#)
- ◆ [:CALCulate:MARKer:AOFF](#)
- ◆ [:CALCulate:MARKer:FCOunt:RESolution](#)
- ◆ [:CALCulate:MARKer:FCOunt:RESolution:AUTO](#)
- ◆ [:CALCulate:MARKer:FCOunt:X?](#)
- ◆ [:CALCulate:MARKer:FCOunt\[:STATe\]](#)
- ◆ [:CALCulate:MARKer<n>:CPEak\[:STATe\]](#)
- ◆ [:CALCulate:MARKer<n>:DELTA\[:SET\]:CENTer](#)
- ◆ [:CALCulate:MARKer<n>:DELTA\[:SET\]:SPAN](#)
- ◆ [:CALCulate:MARKer<n>:FUNCTion](#)
- ◆ [:CALCulate:MARKer<n>:MAXimum:LEFT](#)
- ◆ [:CALCulate:MARKer<n>:MAXimum:MAX](#)
- ◆ [:CALCulate:MARKer<n>:MAXimum:NEXT](#)
- ◆ [:CALCulate:MARKer<n>:MAXimum:RIGHT](#)
- ◆ [:CALCulate:MARKer<n>:MINimum](#)

- ◆ [:CALCulate:MARKer<n>:MODE](#)
- ◆ [:CALCulate:MARKer<n>:PEAK:EXCursion](#)
- ◆ [:CALCulate:MARKer<n>:PEAK:SEARch:MODE](#)
- ◆ [:CALCulate:MARKer<n>:PEAK\[:SET\]:CF](#)
- ◆ [:CALCulate:MARKer<n>:PEAK:THReshold](#)
- ◆ [:CALCulate:MARKer<n>:PTPeak](#)
- ◆ [:CALCulate:MARKer<n>\[:SET\]:CENTer](#)
- ◆ [:CALCulate:MARKer<n>\[:SET\]:RLEVel](#)
- ◆ [:CALCulate:MARKer<n>\[:SET\]:STARt](#)
- ◆ [:CALCulate:MARKer<n>\[:SET\]:STEP](#)
- ◆ [:CALCulate:MARKer<n>\[:SET\]:STOP](#)
- ◆ [:CALCulate:MARKer<n>:STATe](#)
- ◆ [:CALCulate:MARKer<n>:TRACe](#)
- ◆ [:CALCulate:MARKer<n>:TRACe:AUTO](#)
- ◆ [:CALCulate:MARKer<n>:X](#)
- ◆ [:CALCulate:MARKer<n>:X:CENTer](#)
- ◆ [:CALCulate:MARKer<n>:X:POSition](#)
- ◆ [:CALCulate:MARKer<n>:X:POSition:CENTer](#)
- ◆ [:CALCulate:MARKer<n>:X:POSition:SPAN](#)
- ◆ [:CALCulate:MARKer<n>:X:POSition:STARt](#)
- ◆ [:CALCulate:MARKer<n>:X:POSition:STOP](#)
- ◆ [:CALCulate:MARKer<n>:X:READout](#)
- ◆ [:CALCulate:MARKer<n>:X:SPAN](#)
- ◆ [:CALCulate:MARKer<n>:X:STARt](#)
- ◆ [:CALCulate:MARKer<n>:X:STOP](#)
- ◆ [:CALCulate:MARKer<n>:Y?](#)
- ◆ [:CALCulate:MARKer:TABLE:STATe](#)
- ◆ [:CALCulate:MARKer:TRACking:STATe](#)

## :CALCulate:BANDwidth:NDB

### 命令格式

```
:CALCulate:BANDwidth:NDB <rel_ampl>
:CALCulate:BANDwidth:NDB?
```

### 功能描述

设置 N dB 带宽测量时 N 的值。  
 查询 N dB 带宽测量时 N 的值。

### 参数

名称	类型	范围	默认值
<rel_ampl>	连续实型	-100 dB 至 100 dB	-3 dB

### 返回格式

查询以科学计数形式返回 N 值。

### 举例

下面的命令设置 N 的值为-4。

```
:CALCulate:BANDwidth:NDB -4
```

下面的查询返回-4.000000E+00。

```
:CALCulate:BANDwidth:NDB?
```

## :CALCulate:BANDwidth:RESult?

### 命令格式

```
:CALCulate:BANDwidth:RESult?
```

### 功能描述

查询 N dB 带宽的测量结果，单位为 Hz。

### 返回格式

以整数形式返回带宽数值（单位为 Hz）。

若没有查找到当前光标频点左、右与其相差 N dB 幅度的两个频点，返回----。

## :CALCulate:LLINe:ALL:DELeTe

### 命令格式

```
:CALCulate:LLINe:ALL:DELeTe
```

### 功能描述

删除当前已编辑的限制线。



## :CALCulate:LLINe:CONTrol:DOMain

### 命令格式

```
:CALCulate:LLINe:CONTrol:DOMain FREQuency|TIME
:CALCulate:LLINe:CONTrol:DOMain?
```

### 功能描述

设置通过/失败测试时横坐标表示频率或时间。  
查询通过/失败测试时横坐标的类型。

### 参数

名称	类型	范围	默认值
——	关键字	FREQuency TIME	FREQuency

### 说明

该设置对上、下限制线同时生效。  
更改横坐标的类型时会删除当前限制线所编辑的所有点。

### 返回格式

查询返回 FREQ 或 TIME。

### 举例

下面的命令设置横坐标为时间。

```
:CALCulate:LLINe:CONTrol:DOMain TIME
```

下面的查询返回 TIME。

```
:CALCulate:LLINe:CONTrol:DOMain?
```

## :CALCulate:LLINe:FAIL?

### 命令格式

```
:CALCulate:LLINe:FAIL?
```

### 功能描述

查询通过/失败测试的结果。

### 返回格式

查询返回 PASS 或 FAIL。如果测试未完成则返回 UNMEAS。

## :CALCulate:LLINe:FAIL:RATIo?

### 命令格式

:CALCulate:LLINe:FAIL:RATIo?

### 功能描述

查询通过/失败测试结果中失败次数的几率。

### 返回格式

查询以科学计数形式返回失败次数的几率。

### 举例

下面的查询返回 5.490000E+00。

```
:CALCulate:LLINe:FAIL:RATIo?
```

## :CALCulate:LLINe:FAIL:STOP:STATe

### 命令格式

```
:CALCulate:LLINe:FAIL:STOP:STATe OFF|ON|0|1
```

```
:CALCulate:LLINe:FAIL:STOP:STATe?
```

### 功能描述

设置如果测试失败是否停止测试。

查询如果测试失败是否停止测试。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 说明

若参数取值为 ON 或 1，则测试失败时停止测试；若参数取值为 OFF 或 0，则测试失败时不停止测试。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令设置测试结果为失败时停止测试。

```
:CALCulate:LLINe:FAIL:STOP:STATe ON 或 :CALCulate:LLINe:FAIL:STOP:STATe 1
```

下面的查询返回 1。

```
:CALCulate:LLINe:FAIL:STOP:STATe?
```

## :CALCulate:LLINE<n>:CONTrol:INTerpolate:TYPE

### 命令格式

```
:CALCulate:LLINE<n>:CONTrol:INTerpolate:TYPE LOGarithmic|LINear
:CALCulate:LLINE<n>:CONTrol:INTerpolate:TYPE?
```

### 功能描述

选择通过/失败测试时的频率插补方式为对数或线性。  
查询频率插补方式。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2	——
——	关键字	LOGarithmic LINear	LINear

### 说明

<n>表示下限制线（1）或上限制线（2）。

选择对数模式时，频率和幅度都以对数方式做插补计算。

选择线性模式时，频率以线性方式、幅度以对数方式做插补计算。

### 返回格式

查询返回 LOG 或 LIN。

### 举例

下面的命令设置下限制线频率插补方式为对数。

```
:CALCulate:LLINE1:CONTrol:INTerpolate:TYPE LOGarithmic
```

下面的查询返回 LOG。

```
:CALCulate:LLINE1:CONTrol:INTerpolate:TYPE?
```

## :CALCulate:LLINE<n>:DATA

### 命令格式

```
:CALCulate:LLINE<n>:DATA <x-axis>,<ampl>,<connected>{,<x-axis>,<ampl>,<connected>}
:CALCulate:LLINE<n>:DATA?
```

### 功能描述

创建一条通过/失败测试的限制线。

查询限制线的信息。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2	2
<x-axis>	连续实型	0 Hz 至 1 GHz（X 轴表示频率） 0 us 至 1 ks（X 轴表示时间）	——
<ampl>	连续实型	-400 dBm 至 320 dBm	——
<connected>	离散型	0 1	0

### 说明

<n>表示下限制线（1）或上限制线（2）。

<x-axis>：所编辑点的频率或时间（取决于 X 轴的类型）。

<ampl>: 所编辑点的幅度。

<connected>: 是否将当前点与上一点相连, 1 代表连接, 0 代表断开。注意: 第 1 个点只可为断开。  
每条限制线最多可编辑 200 个点。

### 返回格式

以<x-axis>,<ampl>,<connected>{,<x-axis>,<ampl>,<connected>}格式返回指定限制线的信息。  
如果指定的限制线未编辑任何点, 返回 NULL。

### 举例

下面的命令创建一条包含 3 个点的上限制线。

```
:CALCulate:LLINe2:DATA 50,100,0,100,150,1,200,200,1
```

下面的查询返回 50,100.000000,0,100,150.000000,1,200,200.000000,1。

```
:CALCulate:LLINe2:DATA?
```

## :CALCulate:LLINe<n>:DATA:MERGe

### 命令格式

```
:CALCulate:LLINe<n>:DATA:MERGe <x-axis>,<ampl>,<connected>{,<x-axis>,<ampl>,<connected>}
```

### 功能描述

通过/失败测试时, 追加编辑限制线上的点。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2	2
<x-axis>	连续实型	0 Hz 至 1 GHz (X 轴表示频率) 0 us 至 1 ks (X 轴表示时间)	——
<ampl>	连续实型	-400 dBm 至 320 dBm	——
<connected>	离散型	0 1	0

### 说明

<n>表示下限制线 (1) 或上限制线 (2)。

<x-axis>: 所编辑点的频率或时间 (取决于 X 轴的类型)。

<ampl>: 所编辑点的幅度。

<connected>: 是否将当前点与上一点相连, 1 表示连接, 0 表示断开。注意: 第 1 个点只可为断开。  
每条限制线最多可编辑 200 个点。

### 举例

下面的命令追加两点至上限制线。

```
:CALCulate:LLINe2:DATA:MERG 250,200,1,300,250,1
```

**:CALCulate:LLINe<n>:DELeTe****命令格式**

:CALCulate:LLINe<n>:DELeTe

**功能描述**

删除通过/失败测试中指定的限制线。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2	——

**说明**

<n>表示下限制线（1）或上限制线（2）。

**举例**

下面的命令删除下限制线。

```
:CALCulate:LLINe1:DELeTe
```

**:CALCulate:LLINe<n>:RELAmpT[:STATe]****命令格式**

:CALCulate:LLINe<n>:RELAmpT[:STATe] OFF|ON|0|1

:CALCulate:LLINe<n>:RELAmpT?

**功能描述**

打开或关闭相对幅度。

查询相对幅度的状态。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2	——
——	布尔型	OFF ON 0 1	OFF 0

**说明**

<n>表示下限制线（1）或上限制线（2）。

打开相对幅度时，您设置的幅度值为当前点的幅度值相对于当前参考电平的差值。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开相对幅度。

```
:CALCulate:LLINe1:RELAmpT:STATe ON 或 :CALCulate:LLINe1:RELAmpT:STATe 1
```

下面的查询返回 1。

```
:CALCulate:LLINe1:RELAmpT?
```

## :CALCulate:LLINe<n>:RELFreq[:STATe]

### 命令格式

```
:CALCulate:LLINe<n>:RELFreq[:STATe] OFF|ON|0|1
:CALCulate:LLINe<n>:RELFreq?
```

### 功能描述

打开或关闭相对频率。  
查询相对频率的状态。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2	——
——	布尔型	OFF ON 0 1	OFF 0

### 说明

<n>表示下限制线（1）或上限制线（2）。  
打开相对频率时，您设置的频率值为当前点的频率值相对于当前中心频率的差值。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开相对频率。

```
:CALCulate:LLINe1:RELFreq:STATe ON 或 :CALCulate:LLINe1:RELFreq:STATe 1
```

下面的查询返回 1。

```
:CALCulate:LLINe1:RELFreq?
```

## :CALCulate:LLINe<n>:STATe

### 命令格式

```
:CALCulate:LLINe<n>:STATe OFF|ON|0|1
:CALCulate:LLINe<n>:STATe?
```

### 功能描述

打开或关闭上/下限制线。  
查询限制线状态。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2	——
——	布尔型	OFF ON 0 1	OFF 0

### 说明

<n>表示下限制线（1）或上限制线（2）。

### 返回格式

查询返回 0 或 1。

**举例**

下面的命令打开上限制线。

```
:CALCulate:LLINe2:STATe ON 或 :CALCulate:LLINe2:STATe 1
```

下面的查询返回 1。

```
:CALCulate:LLINe2:STATe?
```

**:CALCulate:MARKer:AOff****命令格式**

```
:CALCulate:MARKer:AOff
```

**功能描述**

关闭所有打开的光标以及所有基于光标的功能。

**:CALCulate:MARKer:FCOunt:RESolution****命令格式**

```
:CALCulate:MARKer:FCOunt:RESolution <freq>  
:CALCulate:MARKer:FCOunt:RESolution?
```

**功能描述**

设置频率计数器的分辨率。

查询频率计数器的分辨率。

**参数**

名称	类型	范围	默认值
<freq>	离散型	1 Hz   10 Hz   100 Hz   1 kHz   10 kHz   100 kHz	1 kHz

**说明**

<freq>的值以“数字”形式设置时，默认设置单位为 Hz。另外，<freq>的值还可以“数值+单位”的形式设置，如 1 kHz。

使用 [:CALCulate:MARKer:FCOunt:RESolution:AUTO](#) 命令可将分辨率设置为自动模式。

**返回格式**

查询以整数形式返回分辨率，单位为 Hz。

**举例**

下面的命令设置频率计数器的分辨率为 1 kHz。

```
:CALCulate:MARKer:FCOunt:RESolution 1000 或 :CALCulate:MARKer:FCOunt:RESolution 1kHz
```

下面的查询返回 1000。

```
:CALCulate:MARKer:FCOunt:RESolution?
```

## :CALCulate:MARKer:FCOunt:RESolution:AUTO

### 命令格式

```
:CALCulate:MARKer:FCOunt:RESolution:AUTO OFF|ON|0|1
:CALCulate:MARKer:FCOunt:RESolution:AUTO?
```

### 功能描述

自动设置频率计数器的分辨率。  
查询频率计数器分辨率的自动设置状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令关闭频率计数器分辨率的自动设置。

```
:CALCulate:MARKer:FCOunt:RESolution:AUTO OFF 或 :CALCulate:MARKer:FCOunt:RESolution:AUTO 0
```

下面的查询返回 0。

```
:CALCulate:MARKer:FCOunt:RESolution:AUTO?
```

## :CALCulate:MARKer:FCOunt:X?

### 命令格式

```
:CALCulate:MARKer:FCOunt:X?
```

### 功能描述

查询频率计数器的读数。

### 返回格式

以整数形式返回读数，单位为 Hz。  
当频率计数器关闭时，返回 9000000000000000000。



**:CALCulate:MARKer:FCOunt[:STATe]****命令格式**

```
:CALCulate:MARKer:FCOunt[:STATe] OFF|ON|0|1
:CALCulate:MARKer:FCOunt[:STATe]?
```

**功能描述**

打开或关闭频率计数器。  
查询频率计数器的状态。

**参数**

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**说明**

使用 [:CALCulate:MARKer:FCOunt:X?](#) 命令查询频率计数器读数。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开频率计数器。

```
:CALCulate:MARKer:FCOunt:STATe ON 或 :CALCulate:MARKer:FCOunt:STATe 1
```

下面的查询返回 1。

```
:CALCulate:MARKer:FCOunt:STATe?
```

**:CALCulate:MARKer<n>:CPEak[:STATe]****命令格式**

```
:CALCulate:MARKer<n>:CPEak[:STATe] OFF|ON|0|1
:CALCulate:MARKer<n>:CPEak[:STATe]?
```

**功能描述**

打开连续峰值搜索并用指定光标标记峰值，或关闭连续峰值搜索。  
查询连续峰值搜索功能状态。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	1
——	布尔型	OFF ON 0 1	OFF 0

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开连续峰值搜索，用光标 2 标记。

```
:CALCulate:MARKer2:CPEak:STATe ON 或 :CALCulate:MARKer2:CPEak:STATe 1
```

下面的查询返回 1。

```
:CALCulate:MARKer2:CPEak:STATe?
```

## :CALCulate:MARKer<n>:DELTA[:SET]:CENTer

### 命令格式

:CALCulate:MARKer<n>:DELTA[:SET]:CENTer

### 功能描述

将指定的差值、差值对或跨度对光标的频率差值设置为频谱仪的中心频率。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

使用:[:CALCulate:MARKer<n>:MODE](#) 命令选择差值、差值对或跨度对型光标。  
零扫宽下该功能无效。

### 举例

下面的命令将差值对型光标 1 的频率差值设置为频谱仪的中心频率。

```
:CALCulate:MARKer1:DELTA:SET:CENTer
```

## :CALCulate:MARKer<n>:DELTA[:SET]:SPAN

### 命令格式

:CALCulate:MARKer<n>:DELTA[:SET]:SPAN

### 功能描述

将指定的差值、差值对或跨度对光标的频率差值设置为频谱仪的扫宽。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

使用:[:CALCulate:MARKer<n>:MODE](#) 命令选择差值、差值对或跨度对型光标。  
零扫宽下该功能无效。

### 举例

下面的命令将差值对型光标 1 的频率差值设置为频谱仪的扫宽。

```
:CALCulate:MARKer1:DELTA:SET:SPAN
```

## :CALCulate:MARKer<n>:FUNction

### 命令格式

```
:CALCulate:MARKer<n>:FUNction NDB|NOISe|OFF
:CALCulate:MARKer<n>:FUNction?
```

### 功能描述

为指定的光标选择特殊测量类型。  
查询指定光标的特殊测量类型。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
—	关键字	NDB NOISe OFF	OFF

### 说明

NDB: N dB 带宽。  
NOISe: 噪声光标。  
OFF: 关闭所有测量。

### 返回格式

查询返回 NDB、NOIS 或 OFF。

### 举例

下面的命令设置光标 1 的测量类型为 N dB 带宽。

```
:CALCulate:MARKer1:FUNction NDB
```

下面的查询返回 NDB。

```
:CALCulate:MARKer1:FUNction?
```

## :CALCulate:MARKer<n>:MAXimum:LEFT

### 命令格式

```
:CALCulate:MARKer<n>:MAXimum:LEFT
```

### 功能描述

查找迹线上处于当前峰值左侧，并且与之距离最近的满足搜索条件的峰值，并用指定光标标记。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—

### 说明

当无法找到峰值时，屏幕中将提示“没有找到峰值”。

## :CALCulate:MARKer<n>:MAXimum:MAX

### 命令格式

:CALCulate:MARKer<n>:MAXimum:MAX

### 功能描述

根据 :CALCulate:MARKer<n>:PEAK:SEARCh:MODE 命令设置的搜索模式执行一次峰值搜索，并用指定光标标记。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	---

### 举例

下面的命令执行一次峰值搜索，并用光标 2 标记。

```
:CALCulate:MARKer2:MAXimum:MAX
```

## :CALCulate:MARKer<n>:MAXimum:NEXT

### 命令格式

:CALCulate:MARKer<n>:MAXimum:NEXT

### 功能描述

查找迹线上幅度仅次于当前峰值且满足搜索条件的峰值，并用指定光标标记。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	---

### 说明

当无法找到峰值时，屏幕中将提示“没有找到峰值”。

## :CALCulate:MARKer<n>:MAXimum:RIGHT

### 命令格式

:CALCulate:MARKer<n>:MAXimum:RIGHT

### 功能描述

查找迹线上处于当前峰值右侧，并且与之距离最近的满足搜索条件的峰值，并用指定光标标记。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	---

### 说明

当无法找到峰值时，屏幕中将提示“没有找到峰值”。

**:CALCulate:MARKer<n>:MINimum****命令格式**

:CALCulate:MARKer&lt;n&gt;:MINimum

**功能描述**

查找迹线上的最小幅度值，并用指定光标标记。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

**说明**

当无法找到峰值时，屏幕中将提示“没有找到峰值”。

**:CALCulate:MARKer<n>:MODE****命令格式**

:CALCulate:MARKer&lt;n&gt;:MODE POSition|DELTA|BAND|SPAN

:CALCulate:MARKer&lt;n&gt;:MODE?

**功能描述**

设置指定光标的类型。

查询指定光标的类型。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
——	关键字	POSition DELTA BAND SPAN	POSition

**说明**

POSition: 常规。

DELTA: 差值。

BAND: 差值对。

SPAN: 跨度对。

**返回格式**

查询返回 POS、DELT、BAND 或 SPAN。

**举例**

下面的命令设置光标 1 的类型为常规。

:CALCulate:MARKer1:MODE POSition

下面的查询返回 POS。

:CALCulate:MARKer1:MODE?

## :CALCulate:MARKer<n>:PEAK:EXCursion

### 命令格式

```
:CALCulate:MARKer<n>:PEAK:EXCursion <rel_ampl>
:CALCulate:MARKer<n>:PEAK:EXCursion?
```

### 功能描述

设置峰值偏移，单位为 dB。  
查询峰值偏移。

### 参数

名称	类型	范围	默认值
<n>	离散型	1   2   3   4	——
<rel_ampl>	连续实型	0 dB 至 200 dB	10 dB

### 返回格式

查询以科学计数形式返回峰值偏移。

### 举例

下面的命令设置峰值偏移为 12 dB。

```
:CALCulate:MARKer1:PEAK:EXCursion 12
```

下面的查询返回 1.200000E+01。

```
:CALCulate:MARKer1:PEAK:EXCursion?
```

## :CALCulate:MARKer<n>:PEAK:SEARch:MODE

### 命令格式

```
:CALCulate:MARKer<n>:PEAK:SEARch:MODE PARameter|MAXimum
:CALCulate:MARKer<n>:PEAK:SEARch:MODE?
```

### 功能描述

设置峰值搜索模式。  
查询峰值搜索模式。

### 参数

名称	类型	范围	默认值
<n>	离散型	1   2   3   4	——
——	关键字	PARameter MAXimum	MAXimum

### 说明

PARameter: 参数，查找迹线上满足搜索参数（峰值偏移和峰值极限）条件的峰值，并用光标标记。

MAXimum: 最大值，查找迹线上的最大值，并用光标标记。

该命令只对发送 [:CALCulate:MARKer<n>:MAXimum:MAX](#) 对应的峰值搜索功能有效。下一峰值、右峰值、左峰值及最小峰值不受该命令的限制。

### 返回格式

查询返回 PAR 或 MAX。

### 举例

下面的命令设置迹线 1 峰值搜索模式为参数，并用光标 1 标记。

```
:CALCulate:MARKer1:PEAK:SEARch:MODE PARAmeter
```

下面的查询返回 PAR。

```
:CALCulate:MARKer1:PEAK:SEARch:MODE?
```

## :CALCulate:MARKer<n>:PEAK[:SET]:CF

### 命令格式

```
:CALCulate:MARKer<n>:PEAK[:SET]:CF
```

### 功能描述

执行一次峰值搜索（使用指定的光标标记峰值），并将当前峰值信号的频率设置为频谱仪的中心频率。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—

### 举例

下面的命令执行一次峰值搜索（使用光标 1 标记峰值），并将当前峰值信号的频率设置为频谱仪的中心频率。

```
:CALCulate:MARKer1:PEAK:SET:CF
```

## :CALCulate:MARKer<n>:PEAK:THReshold

### 命令格式

```
:CALCulate:MARKer<n>:PEAK:THReshold <ampl>  
:CALCulate:MARKer<n>:PEAK:THReshold?
```

### 功能描述

设置峰值极限，单位默认为 dBm。

查询峰值极限。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<ampl>	连续实型	-200 dBm 至 0 dBm	-90 dBm

### 返回格式

查询以科学计数形式返回峰值极限。

### 举例

下面的命令设置光标 1 对应的峰值极限值为 -100 dBm。

```
:CALCulate:MARKer1:PEAK:THReshold -100
```

下面的查询返回 -1.000000E+02。

```
:CALCulate:MARKer1:PEAK:THReshold?
```

## :CALCulate:MARKer<n>:PTPeak

### 命令格式

```
:CALCulate:MARKer<n>:PTPeak
```

### 功能描述

执行峰峰搜索，并用指定光标标记。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

执行该命令后，指定光标的光标类型自动选择差值对，参考光标和差值光标分别标记峰峰位置。

### 举例

下面的命令执行峰峰搜索，并用参考光标 1R 和差值光标 1 分别标记峰峰位置。

```
:CALCulate:MARKer1:PTPeak
```

## :CALCulate:MARKer<n>[:SET]:CENTer

### 命令格式

```
:CALCulate:MARKer<n>[:SET]:CENTer
```

### 功能描述

将指定光标处的频率设置为频谱仪的中心频率。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

若指定的光标类型为常规，则中心频率被设为光标处的频率。

若指定的光标类型为差值、差值对或跨度对，则中心频率被设为差值光标处的频率。

该命令仅当指定的光标打开时有效。

零扫宽下此功能无效。

### 举例

下面的命令设置光标 1 (常规型) 处的频率为频谱仪的中心频率。

```
:CALCulate:MARKer1:SET:CENTer
```



## :CALCulate:MARKer<n>[:SET]:RLEVel

### 命令格式

```
:CALCulate:MARKer<n>[:SET]:RLEVel
```

### 功能描述

将指定光标处的幅度设置为频谱仪的参考电平。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

若指定的光标类型为常规，则参考电平被设为光标处的幅度。

若指定的光标类型为差值、差值对或跨度对，则参考电平被设为差值光标处的幅度。

该命令仅当指定的光标打开时有效。

### 举例

下面的命令设置光标 2 (常规型) 处的幅度为频谱仪的参考电平。

```
:CALCulate:MARKer2:SET:RLEVel
```

## :CALCulate:MARKer<n>[:SET]:START

### 命令格式

```
:CALCulate:MARKer<n>[:SET]:START
```

### 功能描述

将指定光标处的频率设置为频谱仪的起始频率。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

若指定的光标类型为常规，则起始频率被设为光标处的频率。

若指定的光标类型为差值、差值对或跨度对，则起始频率被设为差值光标处的频率。

该命令仅当指定的光标打开时有效。

零扫宽下此功能无效。

### 举例

下面的命令设置光标 3 (常规型) 处的频率为频谱仪的起始频率。

```
:CALCulate:MARKer3:SET:START
```

## :CALCulate:MARKer<n>[:SET]:STEP

### 命令格式

```
:CALCulate:MARKer<n>[:SET]:STEP
```

### 功能描述

将指定光标处的频率设置为频谱仪的中心频率步长。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

若指定的光标类型为常规，则中心频率步长被设为光标处的频率。

若指定的光标类型为差值、差值对或跨度对，则中心频率步长被设为差值光标处的频率。

该命令仅当指定的光标打开时有效。

零扫宽下此功能无效。

### 举例

下面的命令设置光标 4（常规型）处的频率为频谱仪的中心频率步长。

```
:CALCulate:MARKer4:SET:STEP
```

## :CALCulate:MARKer<n>[:SET]:STOP

### 命令格式

```
:CALCulate:MARKer<n>[:SET]:STOP
```

### 功能描述

将指定光标处的频率设置频谱仪的终止频率。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——

### 说明

若指定的光标类型为常规，则终止频率被设为光标处的频率。

若指定的光标类型为差值、差值对或跨度对，则终止频率被设为差值光标处的频率。

该命令仅当指定的光标打开时有效。

零扫宽下此功能无效。

### 举例

下面的命令设置光标 2（常规型）处的频率为频谱仪的终止频率。

```
:CALCulate:MARKer2:SET:STOP
```

## :CALCulate:MARKer<n>:STATe

### 命令格式

```
:CALCulate:MARKer<n>:STATe OFF|ON|0|1
:CALCulate:MARKer<n>:STATe?
```

### 功能描述

打开或关闭指定的光标。  
查询指定光标的状态。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
——	布尔型	OFF ON 0 1	OFF 0

### 说明

如果指定的光标当前未打开，发送命令打开此光标时，默认选择常规型（POSition）。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开光标 1。

```
:CALCulate:MARKer1:STATe ON 或 :CALCulate:MARKer1:STATe 1
```

下面的查询返回 1。

```
:CALCulate:MARKer1:STATe?
```

## :CALCulate:MARKer<n>:TRACe

### 命令格式

```
:CALCulate:MARKer<n>:TRACe <integer>
:CALCulate:MARKer<n>:TRACe?
```

### 功能描述

为指定的光标设置标记迹线。  
查询指定光标的标记迹线。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
<integer>	离散型	1 2 3 4	1

### 说明

<integer>表示选择的迹线，可选迹线 1、迹线 2、迹线 3 或迹线 4（数学迹线）。所选择的迹线必须已打开。您也可以使用 [:CALCulate:MARKer<n>:TRACe:AUTO](#) 命令设置指定光标的标记迹线为“自动”。

### 返回格式

查询返回 1、2、3 或 4。

如果标记迹线为“自动”，查询返回光标所标记的具体迹线号。

**举例**

下面的命令将光标 1 的标记迹线设置为迹线 2。

```
:CALCulate:MARKer1:TRACe 2
```

下面的查询返回 2。

```
:CALCulate:MARKer1:TRACe?
```

**:CALCulate:MARKer<n>:TRACe:AUTO****命令格式**

```
:CALCulate:MARKer<n>:TRACe:AUTO OFF|ON|0|1
```

```
:CALCulate:MARKer<n>:TRACe:AUTO?
```

**功能描述**

设置指定光标的标记迹线为“自动”。

查询标记迹线是否为“自动”。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
—	布尔型	OFF ON 0 1	ON 1

**说明**

关闭 AUTO 时，当前光标将保持在相应的迹线上。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令设置光标 1 的标记迹线为“自动”。

```
:CALCulate:MARKer1:TRACe:AUTO ON 或 :CALCulate:MARKer1:TRACe:AUTO 1
```

下面的查询返回 1。

```
:CALCulate:MARKer1:TRACe:AUTO?
```

## :CALCulate:MARKer<n>:X

### 命令格式

```
:CALCulate:MARKer<n>:X <param>
:CALCulate:MARKer<n>:X?
```

### 功能描述

设置指定光标横坐标的值，单位默认为 Hz。  
查询指定光标的横坐标值。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
<param>	连续实型	参考“说明”	——

### 说明

<param>取值为当前 X 轴可选的范围。

如果读数方式是“频率”，则不能超过扫描的频率范围，支持 Hz（默认）、kHz、MHz 和 GHz 单位输入。  
如果读数方式是“时间”，则不能超过扫描的时间范围，支持 s（默认）、us、ms 和 ks 单位输入。

如果指定的光标类型为常规，则设置光标处的 X 值。

如果指定的光标类型为差值，则设置差值光标相对于参考光标的 X 值。

如果指定的光标类型为差值对，则根据当前菜单状态，设置参考光标或差值光标的 X 值。

如果指定的光标类型为跨度对，则根据当前菜单状态，设置参考光标和差值光标的跨度或中心的 X 值。

### 返回格式

读数方式是“频率”或“时间倒数”时，查询以整数形式返回光标横坐标的值。

读数方式是“时间”或“周期”时，查询以科学计数形式返回光标横坐标的值。

### 举例

下面的命令设置光标 1 处横坐标的值（读数方式是“频率”）为 150 MHz。

```
:CALCulate:MARKer1:X 150000000
```

下面的查询返回 150000000。

```
:CALCulate:MARKer1:X?
```

## :CALCulate:MARKer<n>:X:CENTer

### 命令格式

```
:CALCulate:MARKer<n>:X:CENTer <param>
:CALCulate:MARKer<n>:X:CENTer?
```

### 功能描述

设置指定跨度对型光标的横坐标中心值。

查询指定跨度对型光标的横坐标中心值。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
<param>	连续实型	参考“说明”	——

### 说明

<param>的取值范围为当前横轴的可选范围。

如果读数方式是“频率”，则不能超过扫描的频率范围，支持 Hz（默认）、kHz、MHz 和 GHz 单位输入。

如果读数方式是“时间”，则不能超过扫描的时间范围，支持 s（默认）、us、ms 和 ks 单位输入。

### 返回格式

读数方式是“频率”时，查询以整数形式返回光标横坐标中心值，单位为 Hz。

读数方式是“时间”或“周期”时，查询以科学计数形式返回光标横坐标中心值，单位为 s。

### 举例

下面的命令设置光标 1（跨度对型）的横坐标中心值（读数方式是“频率”）为 1500000000 Hz。

```
:CALCulate:MARKer1:X:CENTer 1500000000 或 :CALCulate:MARKer1:X:CENTer 1.5GHZ
```

下面的查询返回 1500000000。

```
:CALCulate:MARKer1:X:CENTer?
```

## :CALCulate:MARKer<n>:X:POSition

### 命令格式

```
:CALCulate:MARKer<n>:X:POSition <integer>
```

```
:CALCulate:MARKer<n>:X:POSition?
```

### 功能描述

设置指定的常规型光标标记的位置。

查询指定的常规型光标标记的位置。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<integer>	整型	0 至 600	300

### 返回格式

查询以整数形式返回光标标记的位置。

### 举例

下面的命令设置光标 1（常规型）标记的位置为 100。

```
:CALCulate:MARKer1:X:POSition 100
```

下面的查询返回 100。

```
:CALCulate:MARKer1:X:POSition?
```

**:CALCulate:MARKer<n>:X:POSition:CENTer****命令格式**

```
:CALCulate:MARKer<n>:X:POSition:CENTer <param>
:CALCulate:MARKer<n>:X:POSition:CENTer?
```

**功能描述**

设置指定的跨度对型光标的中心位置。  
 查询指定的跨度对型光标的中心位置。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	整型	0 至 600	300

**返回格式**

查询以整数形式返回指定光标的中心位置。

**举例**

下面的命令设置光标 1（跨度对型）的中心位置为 200。

```
:CALCulate:MARKer1:X:POSition:CENTer 200
```

下面的查询返回 200。

```
:CALCulate:MARKer1:X:POSition:CENTer?
```

**:CALCulate:MARKer<n>:X:POSition:SPAN****命令格式**

```
:CALCulate:MARKer<n>:X:POSition:SPAN <param>
:CALCulate:MARKer<n>:X:POSition:SPAN?
```

**功能描述**

设置指定的跨度对型光标在跨度范围内对应迹线上的点数。  
 查询指定的跨度对型光标在跨度范围内对应迹线上的点数。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	整型	0 至 600	0

**返回格式**

查询以整数形式返回指定光标跨度范围对应迹线上的点数。

**举例**

下面的命令设置光标 1（跨度对型）在跨度范围内对应迹线上的点数为 150。

```
:CALCulate:MARKer1:X:POSition:SPAN 150
```

下面的查询返回 150。

```
:CALCulate:MARKer1:X:POSition:SPAN?
```

## :CALCulate:MARKer<n>:X:POSition:START

### 命令格式

```
:CALCulate:MARKer<n>:X:POSition:START <param>
:CALCulate:MARKer<n>:X:POSition:START?
```

### 功能描述

设置指定的差值对型光标的参考光标标记的位置。  
 查询指定的差值对型光标的参考光标标记的位置。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	整型	0 至 600	300

### 返回格式

查询以整数形式返回参考光标标记的位置。

### 举例

下面的命令设置光标 1（差值对型）的参考光标标记的位置为 100。

```
:CALCulate:MARKer1:X:POSition:START 100
```

下面的查询返回 100。

```
:CALCulate:MARKer1:X:POSition:START?
```

## :CALCulate:MARKer<n>:X:POSition:STOP

### 命令格式

```
:CALCulate:MARKer<n>:X:POSition:STOP <param>
:CALCulate:MARKer<n>:X:POSition:STOP?
```

### 功能描述

设置指定的差值对型光标的差值光标标记的位置。  
 查询指定的差值对型光标的差值光标标记的位置。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	整型	0 至 600	300

### 返回格式

查询以整数形式返回差值光标标记的位置。

### 举例

下面的命令设置光标 1（差值对型）的差值光标标记的位置为 300。

```
:CALCulate:MARKer1:X:POSition:STOP 300
```

下面的查询返回 300。

```
:CALCulate:MARKer1:X:POSition:STOP?
```



## :CALCulate:MARKer<n>:X:READout

### 命令格式

```
:CALCulate:MARKer<n>:X:READout FREQuency|TIME|ITIME|PERiod
:CALCulate:MARKer<n>:X:READout?
```

### 功能描述

选择指定光标在 X 轴的读数方式。  
查询指定光标在 X 轴的读数方式。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
—	关键字	FREQuency TIME ITIME PERiod	参考“说明”

### 说明

FREQuency: 频率, 非零扫宽模式下的默认读数方式。

TIME: 时间, 零扫宽模式下的默认读数方式。

ITIME: 时间倒数, 只有在零扫宽模式且使用差值型光标时可用。

PERiod: 周期, 零扫宽模式下不可用。

### 返回格式

查询返回 FREQ、TIME、ITIM 或 PER。

### 举例

下面的命令设置光标 1 在 X 轴的读数方式为时间。

```
:CALCulate:MARKer1:X:READout TIME
```

下面的查询返回 TIME。

```
:CALCulate:MARKer1:X:READout?
```

## :CALCulate:MARKer<n>:X:SPAN

### 命令格式

```
:CALCulate:MARKer<n>:X:SPAN <param>
:CALCulate:MARKer<n>:X:SPAN?
```

### 功能描述

设置指定的跨度对型光标跨度范围对应的 X 值。

查询指定的跨度对型光标跨度范围对应的 X 值。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	连续实型	参考“说明”	0

### 说明

<param>的取值范围为当前 X 轴的可选范围。

如果当前读数方式是“频率”, 则不能超过扫描的频率范围, 支持 Hz (默认)、kHz、MHz 和 GHz 单位输入。

如果当前读数方式是“时间”, 则不能超过扫描的时间范围, 支持 s (默认)、us、ms 和 ks 单位输入。

### 返回格式

读数方式是“频率”时，查询以整数形式返回光标跨度范围的 X 值，单位为 Hz。

读数方式是“时间”或“周期”时，查询以科学计数形式返回光标跨度范围的 X 值，单位为 s。

### 举例

下面的命令设置光标 1（跨度对型）在跨度范围对应的 X 值（读数方式是“频率”）为 500 MHz。

```
:CALCulate:MARKer1:X:SPAN 500000000
```

下面的查询返回 500000000。

```
:CALCulate:MARKer1:X:SPAN?
```

## :CALCulate:MARKer<n>:X:START

### 命令格式

```
:CALCulate:MARKer<n>:X:START <param>
```

```
:CALCulate:MARKer<n>:X:START?
```

### 功能描述

设置指定的差值对型光标的参考光标对应的 X 值。

查询指定的差值对型光标的参考光标对应的 X 值。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	连续实型	参考“说明”	—

### 说明

<param>的取值范围为当前 X 轴可选的范围。

如果当前读数方式是“频率”，则不能超过扫描的频率范围，支持 Hz（默认）、kHz、MHz 和 GHz 单位输入。

如果当前读数方式是“时间”，则不能超过扫描的时间范围，支持 s（默认）、us、ms 和 ks 单位输入。

### 返回格式

读数方式是“频率”时，查询以整数形式返回参考光标的 X 值，单位为 Hz。

读数方式是“时间”或“周期”时，查询以科学计数形式返回参考光标的 X 值，单位为 s。

### 举例

下面的命令设置光标 1（差值对型）的参考光标对应的 X 值（读数方式是“频率”）为 750 MHz。

```
:CALCulate:MARKer1:X:START 750000000
```

下面的查询返回 750000000。

```
:CALCulate:MARKer1:X:START?
```

## :CALCulate:MARKer<n>:X:STOP

### 命令格式

```
:CALCulate:MARKer<n>:X:STOP <param>
:CALCulate:MARKer<n>:X:STOP?
```

### 功能描述

设置指定的差值对型光标的差值光标对应的 X 值。  
查询指定的差值对型光标的差值光标对应的 X 值。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
<param>	连续实型	参考“说明”	—

### 说明

<param>的取值范围为当前 X 轴可选的范围。

如果当前读数方式是“频率”，则不能超过扫描的频率范围，支持 Hz（默认）、kHz、MHz 和 GHz 单位输入。

如果当前读数方式是“时间”，则不能超过扫描的时间范围，支持 s（默认）、us、ms 和 ks 单位输入。

### 返回格式

读数方式是“频率”时，查询以整数形式返回差值光标的 X 值，单位为 Hz。

读数方式是“时间”或“周期”时，查询以科学计数形式返回差值光标的 X 值，单位为 s。

### 举例

下面的命令设置光标 1（差值对型）的差值光标对应的 X 值（读数方式是“频率”）为 350 MHz。

```
:CALCulate:MARKer1:X:STOP 350000000
```

下面的查询返回 350000000。

```
:CALCulate:MARKer1:X:STOP?
```

## :CALCulate:MARKer<n>:Y?

### 命令格式

```
:CALCulate:MARKer<n>:Y?
```

### 功能描述

查询指定光标处的 Y 轴值，单位默认为 dBm。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—

### 说明

若指定的光标为常规型，则查询光标处的 Y 值。

若指定的光标为差值型、差值对型或跨度对型，则查询参考光标和差值光标的 Y 轴差值。

### 返回格式

查询以科学计数形式返回 Y 轴值。

### 举例

下面的查询返回 5.960000E+00。

```
:CALCulate:MARKer1:Y?
```

## :CALCulate:MARKer:TABLE:STATE

### 命令格式

```
:CALCulate:MARKer:TABLE:STATE OFF|ON|0|1
:CALCulate:MARKer:TABLE:STATE?
```

### 功能描述

打开或关闭光标表。  
查询光标表的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令关闭光标表。

```
:CALCulate:MARKer:TABLE:STATE OFF 或 :CALCulate:MARKer:TABLE:STATE 0
```

下面的查询返回 0。

```
:CALCulate:MARKer:TABLE:STATE?
```

## :CALCulate:MARKer:TRACking:STATE

### 命令格式

```
:CALCulate:MARKer:TRACking:STATE OFF|ON|0|1
:CALCulate:MARKer:TRACking:STATE?
```

### 功能描述

打开或关闭信号追踪。  
查询信号追踪的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 说明

当信号追踪打开时，每次扫描完成后将执行峰值查找，然后将当前峰值处的频率值设为中心频率，使信号始终显示在屏幕中心。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开信号追踪。

```
:CALCulate:MARKer:TRACking:STATE ON 或 :CALCulate:MARKer:TRACking:STATE 1
```

下面的查询返回 1。

```
:CALCulate:MARKer:TRACking:STATE?
```

## :CALibration命令子系统

命令列表:

- ◆ [:CALibration:\[ALL\]](#)
- ◆ [:CALibration:AUTO](#)

### :CALibration:[ALL]

命令格式

:CALibration:[ALL]

功能描述

立即执行自校准。

举例

下面的命令使仪器立即执行自校准。

```
:CALibration:ALL
```

### :CALibration:AUTO

命令格式

:CALibration:AUTO OFF|ON|0|1

:CALibration:AUTO?

功能描述

打开或关闭自动校准。

查询自动校准的状态。

参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

说明

出厂默认打开自动校准，但当用户改变该设置时，下次开机总是保持用户上次设置。

返回格式

查询返回 0 或 1。

举例

下面的命令打开自动校准功能。

```
:CALibration:AUTO ON 或 :CALibration:AUTO 1
```

下面的查询返回 1。

```
:CALibration:AUTO?
```

## :CONFigure命令子系统

命令列表:

- ◆ [:CONFigure?](#)
- ◆ [:CONFigure:ACPower\\*](#)
- ◆ [:CONFigure:CHPower\\*](#)
- ◆ [:CONFigure:CNRatio\\*](#)
- ◆ [:CONFigure:EBWidth\\*](#)
- ◆ [:CONFigure:HDISSt\\*](#)
- ◆ [:CONFigure:OBWidth\\*](#)
- ◆ [:CONFigure:PF](#)
- ◆ [:CONFigure:SANalyzer\\*](#)
- ◆ [:CONFigure:TOI\\*](#)
- ◆ [:CONFigure:TPOWer\\*](#)

说明:

带\*的命令仅适用于已安装高级测量套件选件的 DSA700。

### :CONFigure?

命令格式

:CONFigure?

功能描述

查询当前的测量功能。

返回格式

查询返回 OFF（关闭）、TPOW（时域功率）、ACP（邻道功率）、CHP（通道功率）、OBW（占用带宽）、EBW（发射带宽）、CNR（载噪比）、HD（谐波失真）、TOI（三阶互调失真）或 PF（通过/失败）。

### :CONFigure:ACPower

命令格式

:CONFigure:ACPower

功能描述

将频谱仪设置为邻道功率测量状态。

说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置后，使用:[READ](#)命令可以初始化一个测量，而不需设置它们为默认值。

## :CONFigure:CHPower

### 命令格式

:CONFigure:CHPower

### 功能描述

将频谱仪设置为通道功率测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置后，使用:[:READ](#) 命令可以初始化一个测量，而不需设置它们为默认值。

## :CONFigure:CNRatio

### 命令格式

:CONFigure:CNRatio

### 功能描述

将频谱仪设置为载噪比测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置后，使用:[:READ](#) 命令可以初始化一个测量，而不需设置它们为默认值。

## :CONFigure:EBWidth

### 命令格式

:CONFigure:EBWidth

### 功能描述

将频谱仪设置为发射带宽测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置后，使用:[:READ](#) 命令可以初始化一个测量，而不需设置它们为默认值。

## :CONFigure:HDISt

### 命令格式

:CONFigure:HDISt

### 功能描述

将频谱仪设置为谐波失真测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置后，使用:[:READ](#) 命令可以初始化一个测量，而不需设置它们为默认值。

## :CONFigure:OBWidth

### 命令格式

:CONFigure:OBWidth

### 功能描述

将频谱仪设置为占用带宽测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置后，使用:[READ](#) 命令可以初始化一个测量，而不需设置它们为默认值。

## :CONFigure:PF

### 命令格式

:CONFigure:PF

### 功能描述

打开通过/失败测试功能。

## :CONFigure:SANalyzer

### 命令格式

:CONFigure:SANalyzer

### 功能描述

该命令将关闭当前激活的测量功能，并将频谱仪设置为扫描模式。

## :CONFigure:TOI

### 命令格式

:CONFigure:TOI

### 功能描述

将频谱仪设置为三阶互调失真测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置时，使用:[READ](#) 命令可以初始化一个测量，而不需设置为它们的默认值。



## :CONFigure:TPOWer

### 命令格式

:CONFigure:TPOWer

### 功能描述

将频谱仪设置为时域功率测量状态。

### 说明

该命令中断当前的测量并将仪器设定为指定测量的出厂默认状态。

改变任何测量设置时，使用:[READ](#) 命令可以初始化一个测量，而不需设置为它们的默认值。

## :COUPle命令子系统

命令列表:

◆ [:COUPle](#)

### :COUPle

命令格式

:COUPle ALL|NONE

:COUPle?

功能描述

将所有存在耦合关系的参数按照耦合关系进行联动设置。  
查询耦合联动设置的状态。

参数

名称	类型	范围	默认值
—	关键字	ALL NONE	ALL

说明

下列参数存在耦合关系:

中频步长

参考电平

输入衰减

分辨率带宽 (RBW)

视频带宽 (VBW)

扫描时间

返回格式

查询返回 ALL 或 NONE。

举例

下面的命令关闭具有耦合关系参数之间的联动设置。

```
:COUPle NONE
```

下面的查询返回 NONE。

```
:COUPle?
```

## :DISPlay命令子系统

命令列表:

- ◆ [:DISPlay:AFUnction:POStion](#)
- ◆ [:DISPlay:ANNotation:CLOCK\[:STATe\]](#)
- ◆ [:DISPlay:BRIGhtness](#)
- ◆ [:DISPlay:ENABLe](#)
- ◆ [:DISPlay:MSGswitch:STATe](#)
- ◆ [:DISPlay:UKEY:STATe](#)
- ◆ [:DISPlay:WINDow:TRACe:GRATicule:GRID](#)
- ◆ [:DISPlay:WINDow:TRACe:X\[:SCALe\]:SPACing](#)
- ◆ [:DISPlay:WINDow:TRACe:Y:DLINe](#)
- ◆ [:DISPlay:WINDow:TRACe:Y:DLINe:STATe](#)
- ◆ [:DISPlay:WINDow:TRACe:Y\[:SCALe\]:PDIVision](#)
- ◆ [:DISPlay:WINDow:TRACe:Y\[:SCALe\]:RLEVel](#)
- ◆ [:DISPlay:WINDow:TRACe:Y\[:SCALe\]:RLEVel:OFFSet](#)
- ◆ [:DISPlay:WINDow:TRACe:Y\[:SCALe\]:SPACing](#)

## :DISPlay:AFUnction:POSition

### 命令格式

```
:DISPlay:AFUnction:POSition BOTTOm|CENTer|TOP
:DISPlay:AFUnction:POSition?
```

### 功能描述

设置活动功能区在屏幕中显示的位置。  
查询活动功能区在屏幕中显示的位置。

### 参数

名称	类型	范围	默认值
—	关键字	BOTTOm CENTer TOP	TOP

### 说明

BOTTOm: 底部。  
CENTer: 中间。  
TOP: 顶部。

### 返回格式

查询返回 BOTT、CENT 或 TOP。

### 举例

下面的命令设置活动功能区显示在屏幕中间。

```
:DISPlay:AFUnction:POSition CENTer
```

下面的查询返回 CENT。

```
:DISPlay:AFUnction:POSition?
```

## :DISPlay:ANNotation:CLOCK[:STATe]

### 命令格式

```
:DISPlay:ANNotation:CLOCK[:STATe] OFF|ON|0|1
:DISPlay:ANNotation:CLOCK[:STATe]?
```

### 功能描述

打开或关闭时间和日期的显示。  
查询时间和日期的显示状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开时间和日期的显示。

```
:DISPlay:ANNotation:CLOCK:STATe ON 或 :DISPlay:ANNotation:CLOCK:STATe 1
```

下面的查询返回 1。

```
:DISPlay:ANNotation:CLOCK:STATe?
```

## :DISPlay:BRIGhtness

### 命令格式

```
:DISPlay:BRIGhtness <integer>
:DISPlay:BRIGhtness?
```

### 功能描述

设置屏幕显示亮度。  
查询屏幕显示亮度。

### 参数

名称	类型	范围	默认值
<integer>	整型	0 至 7	2

### 返回格式

查询以整数形式返回亮度值。

### 举例

下面的命令设置亮度值为 6。

```
:DISPlay:BRIGhtness 6
```

下面的查询返回 6。

```
:DISPlay:BRIGhtness?
```

## :DISPlay:ENABle

### 命令格式

```
:DISPlay:ENABle OFF|ON|0|1
:DISPlay:ENABle?
```

### 功能描述

设置屏幕的开关状态。  
查询屏幕的开关状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 说明

选择 OFF|0 时，屏幕被锁定，此时，屏幕停止刷新，测量速度提高。按 Esc 键即可解锁。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令锁定屏幕。

```
:DISPlay:ENABle OFF 或 :DISPlay:ENABle 0
```

下面的查询返回 0。

```
:DISPlay:ENABle?
```

## :DISPlay:MSGswitch:STATe

### 命令格式

```
:DISPlay:MSGswitch:STATe OFF|ON|0|1
:DISPlay:MSGswitch:STATe?
```

### 功能描述

打开或关闭消息的显示。  
查询消息的显示状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令关闭消息显示。

```
:DISPlay:MSGswitch:STATe OFF 或 :DISPlay:MSGswitch:STATe 0
```

下面的查询返回 0。

```
:DISPlay:MSGswitch:STATe?
```

## :DISPlay:UKEY:STATe

### 命令格式

```
:DISPlay:UKEY:STATe OFF|ON|0|1
:DISPlay:UKEY:STATe?
```

### 功能描述

打开或关闭 UserKey 键的定义在用户界面的显示。  
查询用户界面是否显示 UserKey 键的定义。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开 UserKey 键的定义在用户界面的显示。

```
:DISPlay:UKEY:STATe ON 或 :DISPlay:UKEY:STATe 1
```

下面的查询返回 1。

```
:DISPlay:UKEY:STATe?
```

## :DISPlay:WINDow:TRACe:GRATicule:GRID

### 命令格式

```
:DISPlay:WINDow:TRACe:GRATicule:GRID <integer>
:DISPlay:WINDow:TRACe:GRATicule:GRID?
```

### 功能描述

设置屏幕网格的亮度。  
查询屏幕网格的亮度。

### 参数

名称	类型	范围	默认值
<integer>	整型	0 至 10	3

### 返回格式

查询以整数形式返回网格亮度值。

### 举例

下面的命令设置网格亮度为 2。

```
:DISPlay:WINDow:TRACe:GRATicule:GRID 2
```

下面的查询返回 2。

```
:DISPlay:WINDow:TRACe:GRATicule:GRID?
```

## :DISPlay:WINDow:TRACe:X[:SCALe]:SPACing

### 命令格式

```
:DISPlay:WINDow:TRACe:X[:SCALe]:SPACing LINear|LOGarithmic
:DISPlay:WINDow:TRACe:X[:SCALe]:SPACing?
```

### 功能描述

设置 X 轴刻度类型。  
查询 X 轴刻度类型。

### 参数

名称	类型	范围	默认值
—	关键字	LINear LOGarithmic	LINear

### 说明

LINear: 线性。  
LOGarithmic: 对数。

### 返回格式

查询返回 LIN 或 LOG。

### 举例

下面的命令设置 X 轴刻度类型为对数。

```
:DISPlay:WINDow:TRACe:X:SCALe:SPACing LOGarithmic
```

下面的查询返回 LOG。

```
:DISPlay:WINDow:TRACe:X:SCALe:SPACing?
```

## :DISPlay:WINDow:TRACe:Y:DLINe

### 命令格式

```
:DISPlay:WINDow:TRACe:Y:DLINe <ampl>
:DISPlay:WINDow:TRACe:Y:DLINe?
```

### 功能描述

设置显示线在屏幕中的显示位置，单位默认为 dBm。  
查询显示线的显示位置。

### 参数

名称	类型	范围	默认值
<ampl>	连续实型	当前显示的幅度范围	0 dBm

### 说明

出厂默认状态为关闭显示线，首次打开时，显示线的位置为 0 dBm。该参数不受预设设置影响，始终为用户上次设置的值。

### 返回格式

查询以科学计数形式返回显示线位置。

### 举例

下面的命令设置显示线位置为-10 dBm。

```
:DISPlay:WINDow:TRACe:Y:DLINe -10
```

下面的查询返回-1.000000E+01。

```
:DISPlay:WINDow:TRACe:Y:DLINe?
```

## :DISPlay:WINDow:TRACe:Y:DLINe:STATe

### 命令格式

```
:DISPlay:WINDow:TRACe:Y:DLINe:STATe OFF|ON|0|1
:DISPlay:WINDow:TRACe:Y:DLINe:STATe?
```

### 功能描述

打开或关闭显示线的显示。  
查询显示线的显示状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开显示线的显示。

```
:DISPlay:WINDow:TRACe:Y:DLINe:STATe 1 或 :DISPlay:WINDow:TRACe:Y:DLINe:STATe ON
```

下面的查询返回 1。

```
:DISPlay:WINDow:TRACe:Y:DLINe:STATe?
```



## :DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision

### 命令格式

```
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <rel_ampl>
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?
```

### 功能描述

设置 Y 轴刻度。  
查询 Y 轴刻度。

### 参数

名称	类型	范围	默认值
<rel_ampl>	连续实型	0.1 dB 至 20 dB	10 dB

### 返回格式

查询以科学计数形式返回 Y 轴刻度值。

### 举例

下面的命令设置 Y 轴刻度值为 15 dB。

```
:DISPlay:WINDow:TRACe:Y:SCALe:PDIVision 15
```

下面的查询返回 1.500000E+01。

```
:DISPlay:WINDow:TRACe:Y:SCALe:PDIVision?
```

## :DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel

### 命令格式

```
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel <ampl>
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel?
```

### 功能描述

设置参考电平。  
查询参考电平。

### 参数

名称	类型	范围	默认值
<rel_ampl>	连续实型	-100 dBm 至 20 dBm	0 dBm

### 返回格式

查询以科学计数形式返回参考电平值。

### 举例

下面的命令设置参考电平值为 -10 dBm。

```
:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel -10
```

下面的查询返回 -1.000000E+01。

```
:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel?
```

## :DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet

### 命令格式

```
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet <rel_ampl>
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet?
```

### 功能描述

设置参考电平偏移。  
查询参考电平偏移。

### 参数

名称	类型	范围	默认值
<rel_ampl>	连续实型	-300 dB 至 300 dB	0 dB

### 说明

参考电平偏移量不改变曲线的位置，只修改参考电平和光标的幅度读数。

### 返回格式

查询以科学计数形式返回参考电平偏移量。

### 举例

下面的命令设置参考电平偏移值为 10 dB。

```
:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet 10
```

下面的查询返回 1.000000E+01。

```
:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet?
```

## :DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing

### 命令格式

```
:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing LINear|LOGarithmic
:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing?
```

### 功能描述

设置 Y 轴刻度类型。  
查询 Y 轴刻度类型。

### 参数

名称	类型	范围	默认值
——	关键字	LINear LOGarithmic	LOGarithmic

### 说明

LINear: 线性

LOGarithmic: 对数

### 返回格式

查询返回 LIN 或 LOG。

### 举例

下面的命令设置 Y 轴刻度类型为对数。

```
:DISPlay:WINDow:TRACe:Y:SCALe:SPACing LOGarithmic
```

下面的查询返回 LOG。

```
:DISPlay:WINDow:TRACe:Y:SCALe:SPACing?
```

## :FETCh命令子系统

### 命令列表:

- ◆ [:FETCh:ACPower?](#)
- ◆ [:FETCh:ACPower:LOWer?](#)
- ◆ [:FETCh:ACPower:MAIN?](#)
- ◆ [:FETCh:ACPower:UPPer?](#)
- ◆ [:FETCh:CHPower?](#)
- ◆ [:FETCh:CHPower:CHPower?](#)
- ◆ [:FETCh:CHPower:DENSity?](#)
- ◆ [:FETCh:CNRatio?](#)
- ◆ [:FETCh:CNRatio:CARRier?](#)
- ◆ [:FETCh:CNRatio:CNRatio?](#)
- ◆ [:FETCh:CNRatio:NOISe?](#)
- ◆ [:FETCh:EBWidth?](#)
- ◆ [:FETCh:HARMonics:AMPLitude:ALL?](#)
- ◆ [:FETCh:HARMonics:AMPLitude? <n>](#)
- ◆ [:FETCh:HARMonics\[:DISTortion\]?](#)
- ◆ [:FETCh:HARMonics:FREQuency:ALL?](#)
- ◆ [:FETCh:HARMonics:FREQuency? <n>](#)
- ◆ [:FETCh:HARMonics:FUNDamental?](#)
- ◆ [:FETCh:OBWidth?](#)
- ◆ [:FETCh:OBWidth:OBWidth?](#)
- ◆ [:FETCh:OBWidth:OBWidth:FERRor?](#)
- ◆ [:FETCh:TOIntercept?](#)
- ◆ [:FETCh:TOIntercept:IP3?](#)
- ◆ [:FETCh:TPOWer?](#)

### 说明:

:FETCh 命令仅适用于已安装高级测量套件选件的 DSA700。

## **:FETCh:ACPower?**

### **命令格式**

:FETCh:ACPower?

### **功能描述**

查询邻道功率测量结果。

### **说明**

该命令仅在打开邻道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回以逗号分隔的 5 个值：主信道功率、前一信道功率及其与主信道功率之差（dBc）、后一信道功率及其与主信道功率之差（dBc）。

### **举例**

下面的查询返回-5.150423E+01,-5.173441E+01, -2.301865E-01,-5.142665E+01,7.757568E-02。

```
:FETCh:ACPower?
```

## **:FETCh:ACPower:LOWer?**

### **命令格式**

:FETCh:ACPower:LOWer?

### **功能描述**

查询邻道功率测量中后一信道的功率。

### **说明**

该命令仅在打开邻道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回后一信道功率值。

### **举例**

下面的查询返回-5.142665E+01。

```
:FETCh:ACPower:LOWer?
```

## :FETCh:ACPower:MAIN?

### 命令格式

:FETCh:ACPower:MAIN?

### 功能描述

查询邻道功率测量中的主信道功率。

### 说明

该命令仅在打开邻道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回主信道功率值。

### 举例

下面的查询返回-5.150423E+01。

```
:FETCh:ACPower:MAIN?
```

## :FETCh:ACPower:UPPer?

### 命令格式

:FETCh:ACPower:UPPer?

### 功能描述

查询邻道功率测量中的前一信道功率。

### 说明

该命令仅在打开邻道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回前一信道功率值。

### 举例

下面的查询返回-5.173441E+01。

```
:FETCh:ACPower:UPPer?
```

## **:FETCh:CHPower?**

### **命令格式**

:FETCh:CHPower?

### **功能描述**

查询通道功率测量结果。

### **说明**

该命令仅在打开通道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回通道功率和功率谱密度值，以逗号间隔。

### **举例**

下面的查询返回-1.599480E+01,-7.900511E+01。

```
:FETCh:CHPower?
```

## **:FETCh:CHPower:CHPower?**

### **命令格式**

:FETCh:CHPower:CHPower?

### **功能描述**

查询通道功率值。

### **说明**

该命令仅在打开通道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回通道功率值。

### **举例**

下面的查询返回-1.599480E+01。

```
:FETCh:CHPower:CHPower?
```

## :FETCh:CHPower:DENSity?

### 命令格式

:FETCh:CHPower:DENSity?

### 功能描述

查询通道功率谱密度。

### 说明

该命令仅在打开通道功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回通道功率谱密度值。

### 举例

下面的查询返回-7.900511E+01。

```
:FETCh:CHPower:DENSity?
```

## :FETCh:CNRatio?

### 命令格式

:FETCh:CNRatio?

### 功能描述

查询载噪比测量结果。

### 说明

该命令仅在打开载噪比测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回载波功率、噪声功率和载噪比（dB），以逗号间隔。

### 举例

下面的查询返回-6.048788E+01,-6.186192E+01,1.374039E+00。

```
:FETCh:CNRatio?
```

## **:FETCh:CNRatio:CARRier?**

### **命令格式**

:FETCh:CNRatio:CARRier?

### **功能描述**

查询载波功率。

### **说明**

该命令仅在打开载噪比测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回载波功率值。

### **举例**

下面的查询返回-1.484203E+01。

```
:FETCh:CNRatio:CARRier?
```

## **:FETCh:CNRatio:CNRatio?**

### **命令格式**

:FETCh:CNRatio:CNRatio?

### **功能描述**

查询载噪比。

### **说明**

该命令仅在打开载噪比测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回载噪比。

### **举例**

下面的查询返回 8.956909E-02。

```
:FETCh:CNRatio:CNRatio?
```



## **:FETCh:CNRatio:NOISe?**

### **命令格式**

:FETCh:CNRatio:NOISe?

### **功能描述**

查询噪声功率。

### **说明**

该命令仅在打开载噪比测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回噪声功率值。

### **举例**

下面的查询返回-1.442294E+01。

```
:FETCh:CNRatio:NOISe?
```

## **:FETCh:EBWidth?**

### **命令格式**

:FETCh:EBWidth?

### **功能描述**

查询发射带宽测量结果。

### **说明**

该命令仅在打开发射带宽测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以整数形式返回发射带宽测量结果，单位为 Hz。

### **举例**

下面的查询返回 50000。

```
:FETCh:EBWidth?
```

## :FETCh:HARMonics:AMPLitude:ALL?

### 命令格式

:FETCh:HARMonics:AMPLitude:ALL?

### 功能描述

查询前 10 个谐波的幅度值。第一个谐波为基波。

### 说明

该命令仅在打开谐波失真测量时有效。

返回值中幅度单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回前 10 个谐波的幅度值，以逗号间隔。

如果测量的谐波少于 10 个，则未测量的谐波无返回值。

### 举例

下面的查询返回

```
-1.692102E+01,-6.458423E+01,-7.509421E+01,-7.924328E+01,-7.847027E+01,-7.885457E+01,-7.882358E+01,-7.921457E+01,-7.923057E+01,-7.915358E+01。
```

```
:FETCh:HARMonics:AMPLitude:ALL?
```

## :FETCh:HARMonics:AMPLitude? <n>

### 命令格式

:FETCh:HARMonics:AMPLitude? <n>

### 功能描述

查询指定谐波的幅度值。

### 参数

名称	类型	范围	默认值
<n>	整型	1 至 10	---

### 说明

该命令仅在打开谐波失真测量时有效。

返回值中幅度单位与当前 Y 轴单位一致。

所读取的谐波没有数据时，则返回---

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回指定谐波的幅度值。

### 举例

下面的查询返回-1.692102E+01。

```
:FETCh:HARMonics:AMPLitude? 1
```

## **:FETCh:HARMonics[:DISTortion]?**

### **命令格式**

:FETCh:HARMonics[:DISTortion]?

### **功能描述**

查询总谐波失真的百分数。

### **说明**

该命令仅在打开谐波失真测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回总谐波失真的百分数。

### **举例**

下面的查询返回 2.490393E+02。

```
:FETCh:HARMonics:DISToRtion?
```

## **:FETCh:HARMonics:FREQuency:ALL?**

### **命令格式**

:FETCh:HARMonics:FREQuency:ALL?

### **功能描述**

查询前 10 个谐波的频率值。第一个谐波为基波。

### **说明**

该命令仅在打开谐波失真测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以整数形式返回前 10 个谐波的频率值，以逗号间隔，单位为 Hz。

如果测量的谐波少于 10 个，则未测量的谐波无返回值。

### **举例**

下面的查询返回

```
45500000,91000000,136500000,182000000,227500000,273000000,318500000,364000000,409500000,455000000.
```

```
:FETCh:HARMonics:FREQuency:ALL?
```

## :FETCh:HARMonics:FREQuency? <n>

### 命令格式

:FETCh:HARMonics:FREQuency? <n>

### 功能描述

查询指定谐波的频率值。

### 参数

名称	类型	范围	默认值
<n>	整型	1 至 10	---

### 说明

该命令仅在打开谐波失真测量时有效。

所读取的谐波没有数据时，查询返回---

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以整数形式返回指定谐波的频率值，单位为 Hz。

### 举例

下面的查询返回 45500000。

```
:FETCh:HARMonics:FREQuency? 1
```

## :FETCh:HARMonics:FUNDamental?

### 命令格式

:FETCh:HARMonics:FUNDamental?

### 功能描述

查询基波的频率值。

### 说明

该命令仅在打开谐波失真测量时有效。

该命令等同于 [:FETCh:HARMonics:FREQuency? 1](#)。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以整数形式返回基波的频率值，单位为 Hz。

### 举例

下面的查询返回 45500000。

```
:FETCh:HARMonics:FUNDamental?
```

## **:FETCh:OBWidth?**

### **命令格式**

:FETCh:OBWidth?

### **功能描述**

查询占用带宽测量结果。

### **说明**

该命令仅在打开占用带宽测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以整数形式返回占用带宽 (Hz) 和传输频率误差 (Hz)，以逗号间隔。

### **举例**

下面的查询返回 1860000,20000。

```
:FETCh:OBWidth?
```

## **:FETCh:OBWidth:OBWidth?**

### **命令格式**

:FETCh:OBWidth:OBWidth?

### **功能描述**

查询占用带宽。

### **说明**

该命令仅在打开占用带宽测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### **返回格式**

查询以整数形式返回占用带宽，单位为 Hz。

### **举例**

下面的查询返回 1860000。

```
:FETCh:OBWidth:OBWidth?
```

## :FETCh:OBWidth:OBWidth:FERRor?

### 命令格式

:FETCh:OBWidth:OBWidth:FERRor?

### 功能描述

查询传输频率误差。

### 说明

该命令仅在打开占用带宽测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以整数形式返回传输频率误差，单位为 Hz。

### 举例

下面的查询返回 20000。

```
:FETCh:OBWidth:OBWidth:FERRor?
```

## :FETCh:TOIntercept?

### 命令格式

:FETCh:TOIntercept?

### 功能描述

查询三阶互调失真的测量结果。

### 说明

该命令仅在打开三阶互调失真测量时有效。

返回值中幅度单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以如下形式返回三阶互调失真的测量结果：

低基频信号 (Base Lower) 的频率 (Hz)，幅度，高基频信号 (Base Upper) 的频率 (Hz)，幅度，低频 TOI (3rd Order Lower) 的频率 (Hz)，幅度，三阶互调截止点 (Intercept)，高频 TOI (3rd Order Upper) 的频率 (Hz)，幅度，三阶互调截止点 (Intercept)。

### 举例

下面的查询返回

```
1500450000,-8.131735E+01,1500450000,-8.131735E+01,1500450000,-8.131735E+01,-8.131735E+01,1500450000,-8.131735E+01,-8.131735E+01.
```

```
:FETCh:TOIntercept?
```

## :FETCh:TOIntercept:IP3?

### 命令格式

:FETCh:TOIntercept:IP3?

### 功能描述

查询低频 TOI (Third Order Lower) 和 高频 TOI (Third Order Upper) 的三阶互调截止点 (Intercept) 中的较小值。

### 说明

该命令仅在打开三阶互调失真测量时有效。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回较小值。

### 举例

下面的查询返回-8.131735E+01。

```
:FETCh:TOIntercept:IP3?
```

## :FETCh:TPOWer?

### 命令格式

:FETCh:TPOWer?

### 功能描述

查询时域功率测量结果。

### 说明

该命令仅在打开时域功率测量时有效。

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传送至输出缓冲器中。

### 返回格式

查询以科学计数形式返回时域功率测量结果。

### 举例

下面的查询返回-1.658941E+01。

```
:FETCh:TPOWer?
```

## :FORMat命令子系统

命令列表:

- ◆ [:FORMat:BORDER](#)
- ◆ [:FORMat\[:TRACe\]\[:DATA\]](#)

### :FORMat:BORDER

命令格式

```
:FORMat:BORDER NORMal|SWAPped
:FORMat:BORDER?
```

功能描述

选择二进制数据传输的字节顺序。  
查询二进制数据传输的字节顺序。

参数

名称	类型	范围	默认值
—	关键字	NORMal SWAPped	NORMal

说明

NORMal: 字节顺序为以 MSB (Most Significant Byte) 起始, 以 LSB (Least Significant Byte) 结束。  
SWAPped: 字节顺序为以 LSB (Least Significant Byte) 起始, 以 MSB (Most Significant Byte) 结束。

返回格式

查询返回 NORM 或 SWAP。

举例

下面的命令设置二进制数据传输的字节顺序为 NORMal。

```
:FORMat:BORDER NORMal
```

下面的查询返回 NORM。

```
:FORMat:BORDER?
```



**:FORMat[:TRACe][:DATA]****命令格式**

:FORMat[:TRACe][:DATA] ASCii|REAL[,32]  
 :FORMat[:TRACe][:DATA]?

**功能描述**

设置迹线数据的输入/输出格式。  
 查询迹线数据的输入/输出格式。

**参数**

名称	类型	范围	默认值
—	关键字	ASCii REAL[,32]	ASCii

**说明**

ASCii: 数据点为 ASCII 字符, 以逗号分隔。  
 REAL[,32]: 数据点为 32 位二进制数。

**返回格式**

查询返回 ASCII 或 REAL,32。

**举例**

下面的命令设置迹线数据的输入/输出格式为 REAL,32。

```
:FORMat:TRACe:DATA REAL,32
```

下面的查询返回 REAL,32。

```
:FORMat:TRACe:DATA?
```

## :HCOPY命令子系统

命令列表:

- ◆ [:HCOPY:ABORT](#)
- ◆ [:HCOPY:IMAGE:COLor\[:STATe\]](#)
- ◆ [:HCOPY:IMAGE:FTYPE](#)
- ◆ [:HCOPY:IMAGE:INVert](#)
- ◆ [:HCOPY:IMAGE:PTIME](#)
- ◆ [:HCOPY:IMAGE:QUALity](#)
- ◆ [:HCOPY\[:IMMediate\]](#)
- ◆ [:HCOPY:PAGE:ORlentation](#)
- ◆ [:HCOPY:PAGE:PRINTs](#)
- ◆ [:HCOPY:PAGE:SIZE](#)
- ◆ [:HCOPY:RESume](#)

### :HCOPY:ABORT

命令格式

:HCOPY:ABORT

功能描述

取消正在执行的打印操作。

### :HCOPY:IMAGE:COLor[:STATe]

命令格式

:HCOPY:IMAGE:COLor[:STATe] OFF|ON|0|1

:HCOPY:IMAGE:COLor[:STATe]?

功能描述

设置打印颜色为灰色或彩色。

查询打印颜色类型。

参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

说明

OFF|0:灰色。

ON|1:彩色。

返回格式

查询返回 0 或 1。

**举例**

下面的命令设置打印颜色为灰色。

```
:HCOPY:IMAGe:COLor:STATe OFF 或 :HCOPY:IMAGe:COLor:STATe 0
```

下面的查询返回 0。

```
:HCOPY:IMAGe:COLor:STATe?
```

**:HCOPY:IMAGe:FTYPE****命令格式**

```
:HCOPY:IMAGe:FTYPE DEFault|EXIFjpeg
```

```
:HCOPY:IMAGe:FTYPE?
```

**功能描述**

设置打印的图像类型为默认值或 Exif/JPEG。

**参数**

名称	类型	范围	默认值
—	关键字	DEFault EXIFjpeg	DEFault

**返回格式**

查询返回 DEF 或 EXIF。

**举例**

下面的命令设置打印的图像类型为 Exif/JPEG。

```
:HCOPY:IMAGe:FTYPE EXIFjpeg
```

下面的查询返回 EXIF。

```
:HCOPY:IMAGe:FTYPE?
```

**:HCOPY:IMAGe:INVert****命令格式**

```
:HCOPY:IMAGe:INVert OFF|ON|0|1
```

```
:HCOPY:IMAGe:INVert?
```

**功能描述**

设置是否反色打印。

查询是否反色打印。

**参数**

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

**说明**

OFF|0: 关闭反色打印。

ON|1: 打开反色打印。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开反色打印。

```
:HCOPY:IMAGe:INVert ON 或 :HCOPY:IMAGe:INVert 1
```

下面的查询返回 1。

```
:HCOPY:IMAGe:INVert?
```

**:HCOPY:IMAGe:PTIMe****命令格式**

```
:HCOPY:IMAGe:PTIMe OFF|ON|0|1
```

```
:HCOPY:IMAGe:PTIMe?
```

**功能描述**

设置是否打印日期。

查询是否打印日期。

**参数**

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**说明**

OFF|0: 不打印日期。

ON|1: 打印日期。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令设置打印日期。

```
:HCOPY:IMAGe:PTIMe ON 或 :HCOPY:IMAGe:PTIMe 1
```

下面的查询返回 1。

```
:HCOPY:IMAGe:PTIMe?
```

## :HCOPY:IMAGe:QUALity

### 命令格式

```
:HCOPY:IMAGe:QUALity DEFault|NORMal|DRAFt|FINE
:HCOPY:IMAGe:QUALity?
```

### 功能描述

设置打印质量。  
查询打印质量。

### 参数

名称	类型	范围	默认值
——	关键字	DEFault NORMal DRAFt FINE	DEFault

### 说明

DEFault: 默认。  
NORMal: 普通。  
DRAFt: 草稿。  
FINE: 精细。

### 返回格式

查询返回 DEF、NORM、DRAF 或 FINE。

### 举例

下面的命令设置打印质量为草稿。

```
:HCOPY:IMAGe:QUALity DRAFt
```

下面的查询返回 DRAF。

```
:HCOPY:IMAGe:QUALity?
```

## :HCOPY[:IMMEDIATE]

### 命令格式

```
:HCOPY[:IMMEDIATE]
```

### 功能描述

执行打印操作。

## :HCOPY:PAGE:ORIENTATION

### 命令格式

```
:HCOPY:PAGE:ORIENTATION LANDscape|PORTRait
:HCOPY:PAGE:ORIENTATION?
```

### 功能描述

设置打印纸方向为纵向或横向。  
查询打印纸的方向。

### 参数

名称	类型	范围	默认值
—	关键字	LANDscape PORTRait	LANDscape

### 说明

LANDscape: 横向。  
PORTRait: 纵向。

### 返回格式

查询返回 LAND 或 PORT。

### 举例

下面的命令设置打印纸方向为纵向。  
:HCOPY:PAGE:ORIENTATION PORTRait

下面的查询返回 PORT。  
:HCOPY:PAGE:ORIENTATION?

## :HCOPY:PAGE:PRINTS

### 命令格式

```
:HCOPY:PAGE:PRINTS <integer>
:HCOPY:PAGE:PRINTS?
```

### 功能描述

设置打印份数。  
查询打印份数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 999	1

### 返回格式

查询以整数形式返回打印份数。

### 举例

下面的命令设置打印份数为 10。  
:HCOPY:PAGE:PRINTS 10

下面的查询返回 10。  
:HCOPY:PAGE:PRINTS?

## :HCOPY:PAGE:SIZE

### 命令格式

```
:HCOPY:PAGE:SIZE DEFault|A4|A5|A6|B5
:HCOPY:PAGE:SIZE?
```

### 功能描述

设置打印的页面尺寸。  
查询打印的页面尺寸。

### 参数

名称	类型	范围	默认值
——	关键字	DEFault A4 A5 A6 B5	DEFault

### 返回格式

查询返回 DEF、A4、A5、A6 或 B5。

### 举例

下面的命令设置打印的页面尺寸为 A5。

```
:HCOPY:PAGE:SIZE A5
```

下面的查询返回 A5。

```
:HCOPY:PAGE:SIZE?
```

## :HCOPY:RESume

### 命令格式

```
:HCOPY:RESume
```

### 功能描述

恢复被中止的打印任务。

## IEEE 488.2 公用命令

IEEE 488.2 公用命令主要用于操作或查询状态寄存器。状态寄存器的结构请参考“[:STATus 命令子系统](#)”。

命令列表:

- ◆ [\\*CLS](#)
- ◆ [\\*ESE](#)
- ◆ [\\*ESR?](#)
- ◆ [\\*IDN?](#)
- ◆ [\\*OPC](#)
- ◆ [\\*RST](#)
- ◆ [\\*SRE](#)
- ◆ [\\*STB?](#)
- ◆ [\\*TRG](#)
- ◆ [\\*TST?](#)
- ◆ [\\*WAI](#)

### \*CLS

命令格式

\*CLS

功能描述

将所有事件寄存器的值清零，同时清空错误队列。

### \*ESE

命令格式

\*ESE <value>

\*ESE?

功能描述

设置标准事件状态寄存器的使能值。

查询标准事件状态寄存器的使能值。

参数

名称	类型	范围	默认值
<value>	整型	参考“说明”	0

说明

标准事件状态寄存器的位 2 至位 4、位 7 为保留位，可以设置数值但不影响系统，位 1 和位 6 未使用，始终视为 0，因此<value>的取值范围为 00000000（十进制 0）和 11111111（十进制 255）之间位 1 和位 6 为 0 的二进制数对应的十进制数。



### 返回格式

查询返回一个整数，该整数等于寄存器中所有已设置位的权值之和。例如，如果位 4（十进制为 16）和位 7（十进制为 128）被使能，则返回 144。

### 举例

下面的命令设置标准事件状态寄存器的使能寄存器值为 16。

```
*ESE 16
```

下面的查询返回 16。

```
*ESE?
```

## \*ESR?

### 命令格式

```
*ESR?
```

### 功能描述

查询并清除标准事件状态寄存器的事件值。

### 说明

标准事件状态寄存器的位 1 和位 6 未使用，始终视为 0。因此，查询返回 00000000（十进制 0）和 11111111（十进制 255）之间位 1 和位 6 为 0 的二进制数对应的十进制数。

### 返回格式

查询返回一个整数，该整数等于寄存器中所有已设置位的权值之和。例如，如果位 4（十进制为 16）和位 7（十进制为 128）被使能，则返回 144。

### 举例

下面的查询返回 24（位 3 和位 4 已设置）。

```
*ESR?
```

## \*IDN?

### 命令格式

```
*IDN?
```

### 功能描述

查询仪器 ID 字符串。

### 返回格式

查询以如下格式返回仪器的 ID 字符串：

```
Rigol Technologies,<model>,<serial number>,XX.XX.XX.XX.XX
```

<model>：仪器的型号

<serial number>：仪器的序列号

XX.XX.XX.XX.XX：仪器的软件版本

### 举例

下面的查询返回 Rigol Technologies,DSA710,DSA8A134400008,00.01.16.00.03。

```
*IDN?
```

## \*OPC

### 命令格式

\*OPC  
\*OPC?

### 功能描述

在当前操作完成后将标准事件状态寄存器的位 0 置 1。  
查询当前操作是否完成。

### 返回格式

当前操作完成则返回 1，否则返回 0。

## \*RST

### 命令格式

\*RST

### 功能描述

将仪器恢复到出厂默认状态。

## \*SRE

### 命令格式

\*SRE <value>  
\*SRE?

### 功能描述

设置状态字节寄存器的使能值。  
查询状态字节寄存器的使能值。

### 参数

名称	类型	范围	默认值
<value>	整型	参考“说明”	0

### 说明

状态字节寄存器的位 0 和位 1 未使用，始终视为 0，因此<value>的取值范围为 00000000（十进制 0）和 11111111（十进制 255）之间位 0 和位 1 为 0 的二进制数对应的十进制数。

### 返回格式

查询返回一个整数，该整数等于寄存器中所有已设置位的权值之和。例如，如果位 4（十进制为 16）和位 7（十进制为 128）被使能，则返回 144。

### 举例

下面的命令设置状态字节寄存器的使能寄存器值为 16。

```
*SRE 16
```

下面的查询返回 16。

```
*SRE?
```

## \*STB?

### 命令格式

\*STB?

### 功能描述

查询状态字节寄存器的事件值。

### 说明

状态字节寄存器的位 0 和位 1 未使用，始终视为 0。因此，查询返回 00000000（十进制 0）和 11111111（十进制 255）之间位 0 和位 1 为 0 的二进制数对应的十进制数。

### 返回格式

查询返回一个整数，该整数等于寄存器中所有已设置位的权值之和。例如，如果位 4（十进制为 16）和位 7（十进制为 128）被使能，则返回 144。

### 举例

下面的查询返回 24（位 3 和位 4 与已设置）。

```
*STB?
```

## \*TRG

### 命令格式

\*TRG

### 功能描述

立即触发一次扫频或测量。

## \*TST?

### 命令格式

\*TST?

### 功能描述

查询仪器是否完成自检操作。

## \*WAI

### 命令格式

\*WAI

### 功能描述

等待操作完成。

## :INITiate命令子系统

命令列表:

- ◆ [:INITiate:CONTinuous](#)
- ◆ [:INITiate\[:IMMediate\]](#)
- ◆ [:INITiate:PAUSe\\*](#)
- ◆ [:INITiate:REStart\\*](#)
- ◆ [:INITiate:RESume\\*](#)

说明:

带\*的命令仅适用于已安装高级测量套件选件的 DSA700。

### :INITiate:CONTinuous

命令格式

```
:INITiate:CONTinuous OFF|ON|0|1
:INITiate:CONTinuous?
```

功能描述

在非测量状态下，选择连续 (ON|1) 或单次 (OFF|0) 扫描。在测量状态下，选择连续 (ON|1) 或单次 (OFF|0) 测量。

查询扫描或测量的方式。

参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

返回格式

查询返回 0 或 1。

举例

下面的命令使仪器进行连续扫描（频谱仪在非测量状态下）。

```
:INITiate:CONTinuous ON 或 :INITiate:CONTinuous 1
```

下面的查询返回 1。

```
:INITiate:CONTinuous?
```

## **:INITiate[:IMMediate]**

### **命令格式**

:INITiate[:IMMediate]

### **功能描述**

在非测量状态下，初始化一次扫描。

在测量状态下，触发一次测量。

### **说明**

仪器必须处在单次测量模式下。如果:INITiate:CONTinuous 命令设置为 ON 或 1，则忽略该命令。

使用:FETCh?命令可将一个测量结果从内部存储器传输到输出缓存中。

## **:INITiate:PAUSE**

### **命令格式**

:INITiate:PAUSE

### **功能描述**

将仪器测量状态从“等待触发”转换到“暂停”。

### **说明**

该命令仅在打开测量功能并且选择连续测量模式时有效。

## **:INITiate:REStart**

### **命令格式**

:INITiate:REStart

### **功能描述**

在“空闲”状态下重启当前的测量。

### **说明**

该命令仅在测量功能打开时有效。

## **:INITiate:RESume**

### **命令格式**

:INITiate:RESume

### **功能描述**

将仪器测量状态由“暂停”恢复到“等待触发”。

### **说明**

该命令仅在打开测量功能并且选择连续测量模式时有效。

## :INPut命令子系统

命令列表:

◆ [:INPut:IMPedance](#)

### :INPut:IMPedance

命令格式

```
:INPut:IMPedance 50|75
```

```
:INPut:IMPedance?
```

功能描述

设置电压转换为功率时的输入阻抗，单位为  $\Omega$ 。

查询电压转换为功率时的输入阻抗。

参数

名称	类型	范围	默认值
—	离散型	50 75	50

说明

如果被测系统的输出阻抗为  $75 \Omega$ ，则需使用 **RIGOL** 提供的  $75 \Omega$  转  $50 \Omega$  适配器（选件）将被测系统和频谱仪连接起来，并将输入阻抗设置为  $75 \Omega$ 。

返回格式

查询返回 50 或 75。

举例

下面的命令设置输入阻抗为  $75 \Omega$ 。

```
:INPut:IMPedance 75
```

下面的查询返回 75。

```
:INPut:IMPedance?
```

## :MMEMory命令子系统

命令列表:

- ◆ [:MMEMory:DELeTe](#)
- ◆ [:MMEMory:DISK:INFormation?](#)
- ◆ [:MMEMory:LOAD:CORRection](#)
- ◆ [:MMEMory:LOAD:LIMit](#)
- ◆ [:MMEMory:LOAD:MTABle](#)
- ◆ [:MMEMory:LOAD:SETUp](#)
- ◆ [:MMEMory:LOAD:STATe](#)
- ◆ [:MMEMory:LOAD:TRACe](#)
- ◆ [:MMEMory:MOVE](#)
- ◆ [:MMEMory:STORe:CORRection](#)
- ◆ [:MMEMory:STORe:LIMit](#)
- ◆ [:MMEMory:STORe:MTABle](#)
- ◆ [:MMEMory:STORe:PTABle](#)
- ◆ [:MMEMory:STORe:RESults](#)
- ◆ [:MMEMory:STORe:SCReen](#)
- ◆ [:MMEMory:STORe:SETUp](#)
- ◆ [:MMEMory:STORe:STATe](#)
- ◆ [:MMEMory:STORe:TRACe](#)

### :MMEMory:DELeTe

命令格式

```
:MMEMory:DELeTe <file_name>
```

功能描述

删除指定文件名的文件。

参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	—	—

说明

<file\_name>需包含路径和文件名，如 E:\Rigol\Trace1.trc。

如果指定的文件不存在，该操作失败。

举例

下面的命令删除 U 盘中 Rigol 文件夹下的 Trace1.trc 文件。

```
:MMEMory:DELeTe E:\Rigol\Trace1.trc
```

## :MMEMory:DISK:INFormation?

### 命令格式

:MMEMory:DISK:INFormation?

### 功能描述

查询当前 U 盘的磁盘信息。

### 返回格式

查询以字符串形式返回磁盘信息，包括磁盘名、类型、文件系统、已用空间和总容量。

### 举例

下面的查询返回值为：

磁盘名：E

类型：可移动磁盘

文件系统：FAT32

已用空间：2.15 MB

总容量：3.73 GB

```
:MMEMory:DISK:INFormation?
```

## :MMEMory:LOAD:CORRection

### 命令格式

:MMEMory:LOAD:CORRection ANTenna|CABLe|OTHer|USER,<file\_name>

### 功能描述

载入指定文件（.cbl）中的数据进行幅度校正。

### 参数

名称	类型	范围	默认值
——	关键字	ANTenna CABLe OTHer USER	——
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

如果指定的文件不存在，该操作失败。

### 举例

下面的命令载入 D 盘中的 123.cbl 文件进行幅度校正。

```
:MMEMory:LOAD:CORRection ANT,D:\Corr0:123.cbl
```



## :MMEMory:LOAD:LIMit

### 命令格式

:MMEMory:LOAD:LIMit <file\_name>

### 功能描述

将已编辑的限制线文件（.lim）载入仪器中。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 需包含路径和文件名。

如果指定的文件不存在，该操作失败。

### 举例

下面的命令将 D 盘的 upp1.lim 限制线文件载入仪器中。

```
:MMEM:LOAD:LIM D:\Limit1:upp1.lim
```

## :MMEMory:LOAD:MTABLE

### 命令格式

:MMEMory:LOAD:MTABLE <file\_name>

### 功能描述

将已存储的光标表文件（.mkr）载入仪器中。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 需包含路径和文件名。

光标表文件只可保存在外部存储器中，也只可从外部存储器载入仪器。

如果指定的文件不存在，该操作失败。

### 举例

下面的命令将 U 盘中 Mark.mkr 光标表文件载入仪器中。

```
:MMEMory:LOAD:MTABLE E:\Mak.mkr
```

## :MMEMory:LOAD:SETUp

### 命令格式

:MMEMory:LOAD:SETUp <file\_name>

### 功能描述

加载指定的设置文件（.set）。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

如果指定的文件不存在，该操作失败。

### 举例

下面的命令将 D 盘中 pf.set 设置文件载入仪器中。

```
:MMEMory:LOAD:SETUp D:\Setup0:pf.set
```

## :MMEMory:LOAD:STATe

### 命令格式

:MMEMory:LOAD:STATe 1,<file\_name>

### 功能描述

加载指定的状态文件（.sta）。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

如果指定的文件不存在，该操作失败。

### 举例

下面的命令将 D 盘中 default.sta 状态文件载入仪器中。

```
:MMEMory:LOAD:STATe 1,D:\State0:default.sta
```

## :MMEMory:LOAD:TRACe

### 命令格式

:MMEMory:LOAD:TRACe <file\_name>

### 功能描述

加载指定的迹线文件 (.trc)。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 需包含路径和文件名。  
如果指定的文件不存在，该操作失败。

### 举例

下面的命令将 D 盘中 t1.trc 迹线文件载入仪器中。

```
:MMEMory:LOAD:TRACe D:\Trace0:t1.trc
```

## :MMEMory:MOVE

### 命令格式

:MMEMory:MOVE <file\_name1>,<file\_name2>

### 功能描述

将 <file\_name1> 指定的文件重命名为 <file\_name2>。

### 参数

名称	类型	范围	默认值
<file_name1>	ASCII 字符串	——	——
<file_name2>	ASCII 字符串	——	——

### 说明

<file\_name1> 和 <file\_name2> 需包含路径和文件名。  
如果指定的文件不存在，该操作失败。

### 举例

下面的命令将 U 盘中的 file1.bmp 文件重命名为 file2.bmp。

```
:MMEM:MOVE E:\file1.bmp, E:\file2.bmp
```

## :MMEMory:STORe:CORRection

### 命令格式

:MMEMory:STORe:CORRection ANTenna|CABLe|OTHer|USER,<file\_name>

### 功能描述

以指定文件名（.cbl 或.csv）保存幅度校正数据文件。

### 参数

名称	类型	范围	默认值
——	关键字	ANTenna CABLe OTHer USER	——
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

您可以将幅度校正文件保存到 D 盘或 E 盘。.csv 格式的文件只可保存在 E 盘。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 ANT.cbl 保存幅度校正数据文件至 D 盘。

```
:MMEMory:STORe:CORRection ANT,D:\Corr1:ANT.cbl
```

## :MMEMory:STORe:LIMit

### 命令格式

:MMEMory:STORe:LIMit <file\_name>

### 功能描述

以指定文件名（.lim）保存当前编辑的限制线。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

您可以将限制文件保存到 D 盘或 E 盘。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 low.liml 保存当前编辑的限制线至 D 盘。

```
:MMEMory:STORe:LIMit D:\Limit0:low.lim
```

## :MMEMory:STORe:MTABLE

### 命令格式

:MMEMory:STORe:MTABLE <file\_name>

### 功能描述

以指定文件名（.mkr 或.csv）保存光标表到 U 盘中。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 包含路径和文件名。  
如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 MAK1.mkr 保存光标表至 U 盘。

```
:MMEMory:STORe:MTABLE E:\MAK1.mkr
```

## :MMEMory:STORe:PTABLE

### 命令格式

:MMEMory:STORe:PTABLE <file\_name>

### 功能描述

以指定文件名（.csv）保存峰值表到 U 盘中。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 需包含路径和文件名。  
如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 PT1.csv 保存峰值表到 U 盘。

```
:MMEMory:STORe:PTABLE E:\PT1.csv
```

## :MMEMory:STORe:RESults

### 命令格式

:MMEMory:STORe:RESults <file\_name>

### 功能描述

以指定文件名（.csv）保存当前的测量结果到 U 盘中。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 ACP.csv 保存当前测量结果至 U 盘。

```
:MMEMory:STORe:RESults E:\ACP.csv
```

## :MMEMory:STORe:SCReen

### 命令格式

:MMEMory:STORe:SCReen <file\_name>

### 功能描述

以指定文件名（.bmp）保存当前屏幕图像到 U 盘中。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 screen.bmp 保存当前屏幕图像至 U 盘。

```
:MMEMory:STORe:SCReen E:\screen.bmp
```

## :MMEMory:STORe:SETUp

### 命令格式

:MMEMory:STORe:SETUp <file\_name>

### 功能描述

以指定文件名（.set）保存当前的设置。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 需包含路径和文件名。

您可以将设置文件保存到 D 盘或 E 盘。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 ST1.set 保存当前的设置至 U 盘。

```
:MMEMory:STORe:SETUp D:\Setup1:ST1.set
```

## :MMEMory:STORe:STATe

### 命令格式

:MMEMory:STORe:STATe 1,<file\_name>

### 功能描述

以指定文件名（.sta）保存当前的仪器状态。

### 参数

名称	类型	范围	默认值
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name> 需包含路径和文件名。

您可以将状态文件保存到 C 盘、D 盘或 E 盘。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 state.sta 保存当前的仪器状态至 U 盘。

```
:MMEMory:STORe:STATe 1,E:\state.sta
```

## :MMEMory:STORe:TRACe

### 命令格式

:MMEMory:STORe:TRACe <label>,<file\_name>

### 功能描述

以指定文件名（.trc 或.csv）保存指定迹线。

### 参数

名称	类型	范围	默认值
<label>	关键字	TRACE1 TRACE2 TRACE3 MATH ALL	——
<file_name>	ASCII 字符串	——	——

### 说明

<file\_name>需包含路径和文件名。

您可以将迹线文件保存到 D 盘或 E 盘。 .csv 格式的文件只可保存在 E 盘。

如果指定的文件已存在，则将其覆盖。

### 举例

下面的命令以文件名 T1.trc 保存迹线 1。

```
:MMEMory:STORe:TRACe TRACE1,D:\Trace1:T1.trc
```



## :READ 命令子系统

:READ 命令与:FETCh 命令的作用都是获取测量结果。二者之间的区别是，:FETCh 命令立即获取测量结果，:READ 命令启动一次测量并等待测量结束后返回测量结果。

### 命令列表:

- ◆ [:READ:ACPower?](#)
- ◆ [:READ:ACPower:LOWer?](#)
- ◆ [:READ:ACPower:MAIN?](#)
- ◆ [:READ:ACPower:UPPer?](#)
- ◆ [:READ:CHPower?](#)
- ◆ [:READ:CHPower:CHPower?](#)
- ◆ [:READ:CHPower:DENSity?](#)
- ◆ [:READ:CNRatio?](#)
- ◆ [:READ:CNRatio:CARRier?](#)
- ◆ [:READ:CNRatio:CNRatio?](#)
- ◆ [:READ:CNRatio:NOISe?](#)
- ◆ [:READ:EBWidth?](#)
- ◆ [:READ:HARMonics:AMPLitude:ALL?](#)
- ◆ [:READ:HARMonics:AMPLitude? <n>](#)
- ◆ [:READ:HARMonics\[:DISTortion\]?](#)
- ◆ [:READ:HARMonics:FREQuency:ALL?](#)
- ◆ [:READ:HARMonics:FREQuency? <n>](#)
- ◆ [:READ:HARMonics:FUNDamental?](#)
- ◆ [:READ:OBWidth?](#)
- ◆ [:READ:OBWidth:OBWidth?](#)
- ◆ [:READ:OBWidth:OBWidth:FERRor?](#)
- ◆ [:READ:TOIntercept?](#)
- ◆ [:READ:TOIntercept:IP3?](#)
- ◆ [:READ:TPOWer?](#)

### 说明:

:READ 命令仅适用于已安装高级测量套件选件的 DSA700。

## **:READ:ACPower?**

### **命令格式**

:READ:ACPower?

### **功能描述**

执行一次邻道功率测量并返回测量结果。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回以逗号间隔的 5 个值：主信道功率、前一信道功率及其与主信道功率之差（dBc）、后一信道功率及其与主信道功率之差（dBc）。 例如：

-5.150423E+01,-5.173441E+01,-2.301865E-01,-5.142665E+01,7.757568E-02。

## **:READ:ACPower:LOWer?**

### **命令格式**

:READ:ACPower:LOWer?

### **功能描述**

执行一次邻道功率测量并返回后一信道功率。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回后一信道功率值。例如：-5.142665E+01。

## **:READ:ACPower:MAIN?**

### **命令格式**

:READ:ACPower:MAIN?

### **功能描述**

执行一次邻道功率测量并返回主信道功率。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回主信道功率值。例如：-5.150423E+01。

## **:READ:ACPower:UPPer?**

### **命令格式**

:READ:ACPower:UPPer?

### **功能描述**

执行一次邻道功率测量并返回前一信道功率。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回前一信道功率值。例如：-5.173441E+01。

## **:READ:CHPower?**

### **命令格式**

:READ:CHPower?

### **功能描述**

执行一次通道功率测量并返回测量结果。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回通道功率和功率谱密度值，以逗号间隔。例如：-1.599480E+01,-7.900511E+01。

## **:READ:CHPower:CHPower?**

### **命令格式**

:READ:CHPower:CHPower?

### **功能描述**

执行一次通道功率测量并返回通道功率值。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回通道功率值。例如：-1.599480E+01。

## **:READ:CHPower:DENSity?**

### **命令格式**

:READ:CHPower:DENSity?

### **功能描述**

执行一次通道功率测量并返回通道功率谱密度。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回通道功率谱密度值。例如：-7.900511E+01。

## **:READ:CNRatio?**

### **命令格式**

:READ:CNRatio?

### **功能描述**

执行一次载噪比测量并返回测量结果。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回以逗号间隔的 3 个值：载波功率、噪声功率和载噪比（dB）。

例如：-6.048788E+01,-6.186192E+01,1.374039E+00。

## **:READ:CNRatio:CARRier?**

### **命令格式**

:READ:CNRatio:CARRier?

### **功能描述**

执行一次载噪比测量并返回载波功率。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回载波功率值。例如：-1.484203E+01。

## **:READ:CNRatio:CNRatio?**

### **命令格式**

:READ:CNRatio:CNRatio?

### **功能描述**

执行一次载噪比测量并返回载噪比。

### **说明**

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回载噪比。例如：8.956909E-02。

## **:READ:CNRatio:NOISe?**

### **命令格式**

:READ:CNRatio:NOISe?

### **功能描述**

执行一次载噪比测量并返回噪声功率。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回噪声功率值。例如：-1.442294E+01。

## **:READ:EBWidth?**

### **命令格式**

:READ:EBWidth?

### **功能描述**

执行一次发射带宽测量并返回测量结果。

### **说明**

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以整数形式返回发射带宽测量结果，单位为 Hz。例如：50000。

**:READ:HARMonics:AMPLitude:ALL?****命令格式**

:READ:HARMonics:AMPLitude:ALL?

**功能描述**

执行一次谐波失真测量并返回前 10 个谐波的幅度值。

**说明**

如果测量的谐波少于 10 个，则未测量的谐波无返回值。

返回值中幅度单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

**返回格式**

查询以科学计数形式返回前 10 个谐波的幅度值，以逗号间隔。

例如：

-1.692102E+01,-6.458423E+01,-7.509421E+01,-7.924328E+01,-7.847027E+01,-7.885457E+01,-7.882358E+01,-7.921457E+01,-7.923057E+01,-7.915358E+01。

**:READ:HARMonics:AMPLitude? <n>****命令格式**

:READ:HARMonics:AMPLitude? &lt;n&gt;

**功能描述**

执行一次谐波失真测量并返回指定谐波的幅度值。

**参数**

名称	类型	范围	默认值
<n>	整型	1 至 10	---

**说明**

返回值中幅度单位与当前 Y 轴单位一致。

所读取的谐波没有数据时，则返回---

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

**返回格式**

查询以科学计数形式返回指定谐波的幅度值。

例如：-1.692102E+01。

## :READ:HARMonics[:DISToRTion]?

### 命令格式

:READ:HARMonics[:DISToRTion]?

### 功能描述

执行一次谐波失真测量并返回总谐波失真的百分数。

### 说明

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### 返回格式

查询以科学计数形式返回总谐波失真的百分数。例如：2.490393E+02。

## :READ:HARMonics:FREQuency:ALL?

### 命令格式

:READ:HARMonics:FREQuency:ALL?

### 功能描述

执行一次谐波失真测量并返回前 10 个谐波的频率值。

### 说明

如果测量的谐波少于 10 个，则未测量的谐波无返回值。  
该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### 返回格式

查询以整数形式返回频率值，单位为 Hz，以逗号间隔。  
例如：45500000,91000000,136500000,182000000,  
227500000,273000000,318500000,364000000,409500000,455000000。

## :READ:HARMonics:FREQuency? <n>

### 命令格式

:READ:HARMonics:FREQuency? <n>

### 功能描述

执行一次谐波失真测量并返回指定谐波的频率值。

### 参数

名称	类型	范围	默认值
<n>	整型	1 至 10	---

### 说明

所读取的谐波没有数据时，查询返回---。  
该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### 返回格式

查询以整数形式返回指定谐波的频率值，单位 Hz。例如：45500000。

## **:READ:HARMonics:FUNDamental?**

### **命令格式**

:READ:HARMonics:FUNDamental?

### **功能描述**

执行一次谐波失真测量并返回基波的频率值。

### **说明**

该命令等同于:READ:HARMonics:FREQuency? 1。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以整数形式返回基波的频率值，单位为 Hz。例如：45500000。

## **:READ:OBWidth?**

### **命令格式**

:READ:OBWidth?

### **功能描述**

执行一次占用带宽测量并返回测量结果。

### **说明**

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以整数形式返回占用带宽（Hz）和传输频率误差（Hz），以逗号间隔。

例如：1860000,20000。

## **:READ:OBWidth:OBWidth?**

### **命令格式**

:READ:OBWidth:OBWidth?

### **功能描述**

执行一次占用带宽测量并返回占用带宽。

### **说明**

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以整数形式返回占用带宽，单位为 Hz。例如：1860000。



## **:READ:OBWidth:OBWidth:FERRor?**

### **命令格式**

:READ:OBWidth:OBWidth:FERRor?

### **功能描述**

执行一次占用带宽测量并返回传输频率误差。

### **说明**

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以整数形式返回传输频率误差，单位为 Hz。例如：20000。

## **:READ:TOIntercept?**

### **命令格式**

:READ:TOIntercept?

### **功能描述**

执行一次三阶互调失真测量并返回测量结果。

### **说明**

返回值中幅度单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以如下形式返回三阶互调失真测量结果：

低基频信号 (Base Lower) 的频率 (Hz)，幅度，高基频信号 (Base Upper) 的频率 (Hz)，幅度，低频 TOI (3rd Order Lower) 的频率 (Hz)，幅度，三阶互调截止点 (Intercept)，高频 TOI (3rd Order Upper) 的频率 (Hz)，幅度，三阶互调截止点 (Intercept)。

例如：

1500450000,-8.131735E+01,1500450000,-8.131735E+01,1500450000,-8.131735E+01,-8.131735E+01,1500450000,-8.131735E+01,-8.131735E+01。

## **:READ:TOIntercept:IP3?**

### **命令格式**

:READ:TOIntercept:IP3?

### **功能描述**

执行一次三阶互调失真测量并返回低频 TOI (3rd Order Lower) 和 高频 TOI (3rd Order Upper) 的三阶互调截止点 (Intercept) 中的较小值。

### **说明**

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回较小值。例如：-8.131735E+01。

## **:READ:TPOWer?**

### **命令格式**

:READ:TPOWer?

### **功能描述**

执行一次时域功率测量并返回时域功率测量结果。

### **说明**

返回值中功率单位与当前 Y 轴单位一致。

该命令将从最近的测量结果中选择数据传至输出缓冲器中。

### **返回格式**

查询以科学计数形式返回时域功率测量结果。例如：-1.658941E+01。

## [[:SENSe]]命令子系统

### 命令列表:

- ◆ [\[:SENSe\]:ACPower:AVERage:COUNT\\*](#)
- ◆ [\[:SENSe\]:ACPower:AVERage\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:ACPower:AVERage:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:ACPower:BANDwidth:ACHannel\\*](#)
- ◆ [\[:SENSe\]:ACPower:BANDwidth:INTEgration\\*](#)
- ◆ [\[:SENSe\]:ACPower:CSPacing\\*](#)
- ◆ [\[:SENSe\]:BANDwidth:EMIFilter:STATe](#)
- ◆ [\[:SENSe\]:BANDwidth\[:RESolution\]](#)
- ◆ [\[:SENSe\]:BANDwidth\[:RESolution\]:AUTO](#)
- ◆ [\[:SENSe\]:BANDwidth:VIDeo](#)
- ◆ [\[:SENSe\]:BANDwidth:VIDeo:AUTO](#)
- ◆ [\[:SENSe\]:BANDwidth:VIDeo:RATio](#)
- ◆ [\[:SENSe\]:CHPower:AVERage:COUNT\\*](#)
- ◆ [\[:SENSe\]:CHPower:AVERage\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:CHPower:AVERage:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:CHPower:BANDwidth:INTEgration\\*](#)
- ◆ [\[:SENSe\]:CHPower:FREQuency:SPAN\\*](#)
- ◆ [\[:SENSe\]:CNRatio:AVERage:COUNT\\*](#)
- ◆ [\[:SENSe\]:CNRatio:AVERage\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:CNRatio:AVERage:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:CNRatio:BANDwidth:INTEgration\\*](#)
- ◆ [\[:SENSe\]:CNRatio:BANDwidth:NOISe\\*](#)
- ◆ [\[:SENSe\]:CNRatio:OFFSet\\*](#)
- ◆ [\[:SENSe\]:CORRection:CSET:ALL:DELeTe](#)
- ◆ [\[:SENSe\]:CORRection:CSET:ALL\[:STATe\]](#)
- ◆ [\[:SENSe\]:CORRection:CSET<n>:DATA](#)
- ◆ [\[:SENSe\]:CORRection:CSET<n>:DATA:MERGe](#)
- ◆ [\[:SENSe\]:CORRection:CSET<n>:DELeTe](#)
- ◆ [\[:SENSe\]:CORRection:CSET<n>\[:STATe\]](#)
- ◆ [\[:SENSe\]:CORRection:CSET<n>:X:SPACing](#)
- ◆ [\[:SENSe\]:CORRection:CSET:TABLE:STATe](#)
- ◆ [\[:SENSe\]:DEMod](#)
- ◆ [\[:SENSe\]:DEMod:GAIN:AUTO](#)
- ◆ [\[:SENSe\]:DEMod:GAIN:INCRement](#)
- ◆ [\[:SENSe\]:DEMod:STATe](#)
- ◆ [\[:SENSe\]:DEMod:TIME](#)

- ◆ [\[:SENSe\]:DETECTOR\[:FUNCTION\]](#)
- ◆ [\[:SENSe\]:EBWidth:AVERage:COUNT\\*](#)
- ◆ [\[:SENSe\]:EBWidth:AVERage\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:EBWidth:AVERage:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:EBWidth:FREQuency:SPAN\\*](#)
- ◆ [\[:SENSe\]:EBWidth:MAXHold:STATe\\*](#)
- ◆ [\[:SENSe\]:EBWidth:XDB\\*](#)
- ◆ [\[:SENSe\]:EXTRef\[:STATe\]?](#)
- ◆ [\[:SENSe\]:FREQuency:CENTer](#)
- ◆ [\[:SENSe\]:FREQuency:CENTer:DOWN](#)
- ◆ [\[:SENSe\]:FREQuency:CENTer:SET:STEP](#)
- ◆ [\[:SENSe\]:FREQuency:CENTer:STEP:AUTO](#)
- ◆ [\[:SENSe\]:FREQuency:CENTer:STEP\[:INCRement\]](#)
- ◆ [\[:SENSe\]:FREQuency:CENTer:UP](#)
- ◆ [\[:SENSe\]:FREQuency:OFFSet](#)
- ◆ [\[:SENSe\]:FREQuency:SPAN](#)
- ◆ [\[:SENSe\]:FREQuency:SPAN:FULL](#)
- ◆ [\[:SENSe\]:FREQuency:SPAN:PREVious](#)
- ◆ [\[:SENSe\]:FREQuency:SPAN:ZIN](#)
- ◆ [\[:SENSe\]:FREQuency:SPAN:ZOUT](#)
- ◆ [\[:SENSe\]:FREQuency:START](#)
- ◆ [\[:SENSe\]:FREQuency:STOP](#)
- ◆ [\[:SENSe\]:HDISt:AVERage:COUNT\\*](#)
- ◆ [\[:SENSe\]:HDISt:AVERage\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:HDISt:AVERage:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:HDISt:NUMBers\\*](#)
- ◆ [\[:SENSe\]:HDISt:TIME\\*](#)
- ◆ [\[:SENSe\]:HDISt:TIME:AUTO\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:OBWidth:AVERage:COUNT\\*](#)
- ◆ [\[:SENSe\]:OBWidth:AVERage\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:OBWidth:AVERage:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:OBWidth:FREQuency:SPAN\\*](#)
- ◆ [\[:SENSe\]:OBWidth:MAXHold:STATe\\*](#)
- ◆ [\[:SENSe\]:OBWidth:PERCent\\*](#)
- ◆ [\[:SENSe\]:POWer:ARANge](#)
- ◆ [\[:SENSe\]:POWer:ASCale](#)
- ◆ [\[:SENSe\]:POWer:ATUNe](#)
- ◆ [\[:SENSe\]:POWer\[:RF\]:ATTenuation](#)
- ◆ [\[:SENSe\]:POWer\[:RF\]:ATTenuation:AUTO](#)

- ◆ [\[:SENSe\]:POWer\[:RF\]:GAIN\[:STATe\]](#)
- ◆ [\[:SENSe\]:POWer\[:RF\]:MIXer:RANGe\[:UPPer\]](#)
- ◆ [\[:SENSe\]:SIGCapture\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:SIGC\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:MAXHold\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:RESet\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:RESet\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:MAXHold\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:PFSWitch\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:SIGNal\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:AMPUp\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:AMPDown\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:MARK1\[:FREQ\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:MARK1:Switch\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:MARK2\[:FREQ\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SIGCapture:2FSK:MARK2:Switch\[:STATe\]\\*\\*\\*](#)
- ◆ [\[:SENSe\]:SWEep:COUNT](#)
- ◆ [\[:SENSe\]:SWEep:COUNT:CURRent?](#)
- ◆ [\[:SENSe\]:SWEep:TIME](#)
- ◆ [\[:SENSe\]:SWEep:TIME:AUTO](#)
- ◆ [\[:SENSe\]:SWEep:TIME:AUTO:RULEs](#)
- ◆ [\[:SENSe\]:TOI:AVERAge:COUNT\\*](#)
- ◆ [\[:SENSe\]:TOI:AVERAge\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:TOI:AVERAge:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:TOI:FREQuency:SPAN\\*](#)
- ◆ [\[:SENSe\]:TPOWer:AVERAge:COUNT\\*](#)
- ◆ [\[:SENSe\]:TPOWer:AVERAge\[:STATe\]\\*](#)
- ◆ [\[:SENSe\]:TPOWer:AVERAge:TCONtrol\\*](#)
- ◆ [\[:SENSe\]:TPOWer:LLIMit\\*](#)
- ◆ [\[:SENSe\]:TPOWer:MODE\\*](#)
- ◆ [\[:SENSe\]:TPOWer:RLIMit\\*](#)

**说明:**

带\*的命令仅适用于已安装高级测量套件选件的 DSA700；带\*\*\*的命令仅适用于已安装信号无缝捕获套件选件的 DSA700。

## [:SENSe]:ACPower:AVERage:COUNT

### 命令格式

```
[:SENSe]:ACPower:AVERage:COUNT <integer>
[:SENSe]:ACPower:AVERage:COUNT?
```

### 功能描述

设置邻道功率测量的平均次数。  
查询邻道功率测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当打开邻道功率测量时有效。

### 返回格式

查询以整数形式返回平均次数。

### 举例

下面的命令设置平均次数为 100。

```
:SENSe:ACPower:AVERage:COUNT 100
```

下面的查询返回 100。

```
:SENSe:ACPower:AVERage:COUNT?
```

## [:SENSe]:ACPower:AVERage[:STATe]

### 命令格式

```
[:SENSe]:ACPower:AVERage[:STATe] OFF|ON|0|1
[:SENSe]:ACPower:AVERage[:STATe]?
```

### 功能描述

打开或关闭邻道功率测量的平均测量功能。  
查询邻道功率测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当打开邻道功率测量时有效。

平均测量功能的平均次数可通过 [\[:SENSe\]:ACPower:AVERage:COUNT](#) 命令设置。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均测量功能。

```
:SENSe:ACPower:AVERage:STATe ON 或 :SENSe:ACPower:AVERage:STATe 1
```

下面的查询返回 1。

```
:SENSe:ACPower:AVERage:STATe?
```

## [:SENSe]:ACPower:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:ACPower:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:ACPower:AVERage:TCONtrol?
```

### 功能描述

选择邻道功率测量时的平均模式。  
查询邻道功率测量时的平均模式。

### 参数

名称	类型	范围	默认值
—	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均

REPeat: 重复平均

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:ACPower:AVERage:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:ACPower:AVERage:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当打开邻道功率测量时有效。

### 返回格式

查询返回 EXP 或 REP。

### 举例

下面的命令选择平均模式为重复平均。

```
[:SENSe]:ACPower:AVERage:TCONtrol REPeat
```

下面的查询返回 REP。

```
[:SENSe]:ACPower:AVERage:TCONtrol?
```

## [:SENSe]:ACPower:BANDwidth:ACHannel

### 命令格式

```
[:SENSe]:ACPower:BANDwidth:ACHannel <freq>
[:SENSe]:ACPower:BANDwidth:ACHannel?
```

### 功能描述

设置邻近信道的带宽。  
查询邻近信道的带宽。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	33 Hz 至 333.33333 MHz	2 MHz

### 说明

该命令仅当打开邻道功率测量时有效。

邻道带宽和主道带宽联动, 其可设置范围为: 主道带宽/20 至主道带宽×20。

**返回格式**

查询以整数形式返回邻近信道的带宽，单位为 Hz。

**举例**

下面的命令设置邻近信道的带宽为 1 MHz。

```
:SENSe:ACPower:BANDwidth:ACHannel 1000000
```

下面的查询返回 1000000。

```
:SENSe:ACPower:BANDwidth:ACHannel?
```

**[:SENSe]:ACPower:BANDwidth:INTegration****命令格式**

```
[:SENSe]:ACPower:BANDwidth:INTegration <freq>
```

```
[:SENSe]:ACPower:BANDwidth:INTegration?
```

**功能描述**

设置主信道的带宽。

查询主信道的带宽。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	33 Hz 至 333.33333 MHz	2 MHz

**说明**

该命令仅当打开邻道功率测量时有效。

**返回格式**

查询以整数形式返回主信道的带宽，单位为 Hz。

**举例**

下面的命令设置主信道的带宽为 1 MHz。

```
:SENSe:ACPower:BANDwidth:INTegration 1000000
```

下面的查询返回 1000000。

```
:SENSe:ACPower:BANDwidth:INTegration?
```



## [:SENSe]:ACPower:CSPacing

### 命令格式

```
[:SENSe]:ACPower:CSPacing <freq>
[:SENSe]:ACPower:CSPacing?
```

### 功能描述

设置主信道与邻近信道的中心频率差（通道间距）。  
查询通道间距。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	33 Hz 至 333.33333 MHz	2 MHz

### 说明

该命令仅当打开邻道功率测量时有效。

### 返回格式

查询以整数形式返回通道间距，单位为 Hz。

### 举例

下面的命令设置通道间距为 1 MHz。

```
:SENSe:ACPower:CSPacing 1000000
```

下面的查询返回 1000000。

```
:SENSe:ACPower:CSPacing?
```

## [:SENSe]:BANDwidth:EMIFilter:STATe

### 命令格式

```
[:SENSe]:BANDwidth:EMIFilter:STATe OFF|ON|0|1
[:SENSe]:BANDwidth:EMIFilter:STATe?
```

### 功能描述

打开或关闭 EMI 滤波器。  
查询 EMI 滤波器的状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

打开：选择 EMI 滤波器（-6 dB 带宽）。

关闭：选择高斯滤波器（-3 dB 带宽）。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令选择滤波器类型为 EMI 滤波器。

```
:SENSe:BANDwidth:EMIFilter:STATe ON 或 :SENSe:BANDwidth:EMIFilter:STATe 1
```

下面的查询返回 1。

```
:SENSe:BANDwidth:EMIFilter:STATe?
```

## [:SENSe]:BANDwidth[:RESolution]

### 命令格式

```
[:SENSe]:BANDwidth[:RESolution] <freq>
[:SENSe]:BANDwidth[:RESolution]?
```

### 功能描述

设置分辨率带宽 (RBW)。  
查询分辨率带宽。

### 参数

名称	类型	范围	默认值
<freq>	离散型	100 Hz 至 1 MHz (以 1-3-10 为步进)	1 MHz

### 说明

若检波类型选择“准峰值” (参考[:SENSe]:DETECTOR[:FUNCTION]命令) 或滤波器类型选择“EMI” (参考[:SENSe]:BANDwidth:EMIFilter:STATE命令) 时, 分辨率带宽只能选择 200 Hz、9 kHz 或 120 kHz。

### 返回格式

查询以整数形式返回分辨率带宽, 单位为 Hz。

### 举例

下面的命令设置 RBW 为 1000 Hz。

```
:SENSe:BANDwidth:RESolution 1000
```

下面的查询返回 1000。

```
:SENSe:BANDwidth:RESolution?
```

## [:SENSe]:BANDwidth[:RESolution]:AUTO

### 命令格式

```
[:SENSe]:BANDwidth[:RESolution]:AUTO OFF|ON|0|1
[:SENSe]:BANDwidth[:RESolution]:AUTO?
```

### 功能描述

打开或关闭 RBW 的自动设置模式。  
查询 RBW 自动设置模式的状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	ON 1

### 说明

自动模式下, 分辨率带宽将跟随扫宽(非零扫宽)的变化而变化。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开 RBW 的自动设置模式。

```
:SENSe:BANDwidth:RESolution:AUTO ON 或 :SENSe:BANDwidth:RESolution:AUTO 1
```

下面的查询返回 1。

```
:SENSe:BANDwidth:RESolution:AUTO?
```

## [:SENSe]:BANDwidth:VIDeo

### 命令格式

```
[:SENSe]:BANDwidth:VIDeo <freq>
[:SENSe]:BANDwidth:VIDeo?
```

### 功能描述

设置视频带宽（VBW）。  
查询视频带宽。

### 参数

名称	类型	范围	默认值
<freq>	离散型	1 Hz 至 3 MHz（以 1-3-10 为步进）	1 MHz

### 返回格式

查询以整数形式返回视频带宽值，单位为 Hz。

### 举例

下面的命令设置视频带宽为 1000 Hz。

```
:SENSe:BANDwidth:VIDeo 1000
```

下面的查询返回 1000。

```
:SENSe:BANDwidth:VIDeo?
```

## [:SENSe]:BANDwidth:VIDeo:AUTO

### 命令格式

```
[:SENSe]:BANDwidth:VIDeo:AUTO OFF|ON|0|1
[:SENSe]:BANDwidth:VIDeo:AUTO?
```

### 功能描述

打开或关闭 VBW 的自动设置模式。  
查询 VBW 自动设置模式的状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开 VBW 的自动设置模式。

```
:SENSe:BANDwidth:VIDeo:AUTO ON 或 :SENSe:BANDwidth:VIDeo:AUTO 1
```

下面的查询返回 1。

```
:SENSe:BANDwidth:VIDeo:AUTO?
```

## [:SENSe]:BANDwidth:VIDeo:RATio

### 命令格式

```
[:SENSe]:BANDwidth:VIDeo:RATio <number>
[:SENSe]:BANDwidth:VIDeo:RATio?
```

### 功能描述

设置视分比，即 VBW 与 RBW 比值。  
查询视分比。

### 参数

名称	类型	范围	默认值
<number>	离散型	0.000001 至 30000（以 1-3-10 为步进）	1

### 返回格式

查询以科学计数形式返回视分比值。

### 举例

下面的命令设置视分比为 0.01。

```
[:SENSe]:BANDwidth:VIDeo:RATio 0.01
```

下面的查询返回 1.000000E-02。

```
[:SENSe]:BANDwidth:VIDeo:RATio?
```

## [:SENSe]:CHPower:AVERage:COUNT

### 命令格式

```
[:SENSe]:CHPower:AVERage:COUNT <integer>
[:SENSe]:CHPower:AVERage:COUNT?
```

### 功能描述

设置通道功率测量的平均次数。  
查询通道功率测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当通道功率测量打开时有效。

### 返回格式

查询以整数形式返回平均次数。

### 举例

下面的命令设置平均次数为 100。

```
[:SENSe]:CHPower:AVERage:COUNT 100
```

下面的查询返回 100。

```
[:SENSe]:CHPower:AVERage:COUNT?
```

## [:SENSe]:CHPower:AVERAge[:STATe]

### 命令格式

```
[:SENSe]:CHPower:AVERAge[:STATe] OFF|ON|0|1
[:SENSe]:CHPower:AVERAge[:STATe]?
```

### 功能描述

打开或关闭通道功率测量的平均测量功能。  
查询通道功率测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当通道功率测量打开时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均测量功能。

```
:SENSe:CHPower:AVERAge:STATe ON 或 :SENSe:CHPower:AVERAge:STATe 1
```

下面的查询返回 1。

```
:SENSe:CHPower:AVERAge:STATe?
```

## [:SENSe]:CHPower:AVERAge:TCONtrol

### 命令格式

```
[:SENSe]:CHPower:AVERAge:TCONtrol EXPOnential|REPeat
[:SENSe]:CHPower:AVERAge:TCONtrol?
```

### 功能描述

设置通道功率测量时的平均模式。  
查询通道功率测量时的平均模式。

### 参数

名称	类型	范围	默认值
—	关键字	EXPOnential REPeat	EXPOnential

### 说明

EXPOnential: 指数平均

REPeat: 重复平均

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:CHPower:AVERAge:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:CHPower:AVERAge:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当通道功率测量打开时有效。

### 返回格式

查询返回 EXP 或 REP。

**举例**

下面的命令设置平均模式为重复平均。

```
:SENSe:CHPower:AVERage:TCONtrol REPeat
```

下面的查询返回 REP。

```
:SENSe:CHPower:AVERage:TCONtrol?
```

**[[:SENSe]:CHPower:BANDwidth:INTegration****命令格式**

```
[[:SENSe]:CHPower:BANDwidth:INTegration <freq>
```

```
[[:SENSe]:CHPower:BANDwidth:INTegration?
```

**功能描述**

设置通道功率测量的积分带宽。

查询积分带宽。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	100 Hz 至 1 GHz	2 MHz

**说明**

该命令仅当通道功率测量打开时有效。

**返回格式**

查询以整数形式返回积分带宽，单位为 Hz。

**举例**

下面的命令设置积分带宽为 1 MHz。

```
:SENSe:CHPower:BANDwidth:INTegration 1000000
```

下面的查询返回 1000000。

```
:SENSe:CHPower:BANDwidth:INTegration?
```

**[[:SENSe]:CHPower:FREQuency:SPAN****命令格式**

```
[[:SENSe]:CHPower:FREQuency:SPAN <freq>
```

```
[[:SENSe]:CHPower:FREQuency:SPAN?
```

**功能描述**

设置通道功率测量的通道扫宽。

查询通道功率测量的通道扫宽。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	100 Hz 至 1 GHz	3 MHz

**说明**

该命令仅当通道功率测量打开时有效。

通道扫宽与积分带宽联动，其可设置范围为：积分带宽 至 积分带宽×20。

**返回格式**

查询以整数形式返回通道扫宽，单位为 Hz。

**举例**

下面的命令设置通道扫宽为 1 MHz。

```
[:SENSe:CHPower:FREQuency:SPAN 1000000
```

下面的查询返回 1000000。

```
[:SENSe:CHPower:FREQuency:SPAN?
```

**[:SENSe]:CNRatio:AVERage:COUNT****命令格式**

```
[:SENSe]:CNRatio:AVERage:COUNT <integer>
```

```
[:SENSe]:CNRatio:AVERage:COUNT?
```

**功能描述**

设置载噪比测量的平均次数。

查询载噪比测量的平均次数。

**参数**

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

**说明**

该命令仅当载噪比测量打开时有效。

**返回格式**

查询以整数形式返回平均次数。

**举例**

下面的命令设置平均次数为 100。

```
[:SENSe:CNRatio:AVERage:COUNT 100
```

下面的查询返回 100。

```
[:SENSe:CNRatio:AVERage:COUNT?
```

## [:SENSe]:CNRatio:AVERage[:STATe]

### 命令格式

```
[:SENSe]:CNRatio:AVERage[:STATe] OFF|ON|0|1
[:SENSe]:CNRatio:AVERage[:STATe]?
```

### 功能描述

打开或关闭载噪比测量的平均测量功能。  
查询载噪比测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当载噪比测量打开时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均测量功能。

```
:SENSe:CNRatio:AVERage:STATe 1 或 :SENSe:CNRatio:AVERage:STATe ON
```

下面的查询返回 1。

```
:SENSe:CNRatio:AVERage:STATe?
```

## [:SENSe]:CNRatio:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:CNRatio:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:CNRatio:AVERage:TCONtrol?
```

### 功能描述

设置载噪比测量时的平均模式。  
查询载噪比测量时的平均模式。

### 参数

名称	类型	范围	默认值
—	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均

REPeat: 重复平均

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:CNRatio:AVERage:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:CNRatio:AVERage:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当载噪比测量打开时有效。



### 返回格式

查询返回 EXP 或 REP。

### 举例

下面的命令设置平均模式为重复平均。

```
:SENSe:CNRatio:AVERage:TCONtrol REPeat
```

下面的查询返回 REP。

```
:SENSe:CNRatio:AVERage:TCONtrol?
```

## **[[:SENSe]:CNRatio:BANDwidth:INTegration**

### 命令格式

```
[[:SENSe]:CNRatio:BANDwidth:INTegration <freq>
```

```
[[:SENSe]:CNRatio:BANDwidth:INTegration?
```

### 功能描述

设置载波带宽。

查询载波带宽。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	33 Hz 至 333.33333 MHz	2 MHz

### 说明

该命令仅当载噪比测量打开时有效。

载波带宽与噪声带宽联动，其可设置范围为：噪声带宽/20 至 噪声带宽×20。

### 返回格式

查询以整数形式返回载波带宽，单位为 Hz。

### 举例

下面的命令设置载波带宽为 1 MHz。

```
:SENSe:CNRatio:BANDwidth:INTegration 1000000 或 :SENSe:CNRatio:BANDwidth:INTegration 1MHz
```

下面的查询返回 1000000。

```
:SENSe:CNRatio:BANDwidth:INTegration?
```

## [:SENSe]:CNRatio:BANDwidth:NOISe

### 命令格式

```
[:SENSe]:CNRatio:BANDwidth:NOISe <freq>
[:SENSe]:CNRatio:BANDwidth:NOISe?
```

### 功能描述

设置噪声带宽。  
查询噪声带宽。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	33 Hz 至 333.33333 MHz	2 MHz

### 说明

该命令仅当载噪比测量打开时有效。

### 返回格式

查询以整数形式返回噪声带宽，单位为 Hz。

### 举例

下面的命令设置噪声带宽为 1 MHz。

```
[:SENSe]:CNRatio:BANDwidth:NOISe 1000000
```

下面的查询返回 1000000。

```
[:SENSe]:CNRatio:BANDwidth:NOISe?
```

## [:SENSe]:CNRatio:OFFSet

### 命令格式

```
[:SENSe]:CNRatio:OFFSet <freq>
[:SENSe]:CNRatio:OFFSet?
```

### 功能描述

设置载波中心频率与噪声中心频率的差值（偏移频率）。  
查询偏移频率。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	33 Hz 至 333.33333 MHz	2 MHz

### 说明

该命令仅当载噪比测量打开时有效。

### 返回格式

查询以整数形式返回偏移频率，单位为 Hz。

### 举例

下面的命令设置偏移频率为 1 MHz。

```
[:SENSe]:CNRatio:OFFSet 1000000
```

下面的查询返回 1000000。

```
[:SENSe]:CNRatio:OFFSet?
```

**[[:SENSe]:CORRection:CSET:ALL:DELeTe****命令格式**

[:SENSe]:CORRection:CSET:ALL:DELeTe

**功能描述**

删除所有的幅度校正数据。

**[[:SENSe]:CORRection:CSET:ALL[:STATe]****命令格式**

[:SENSe]:CORRection:CSET:ALL[:STATe] OFF|ON|0|1

[:SENSe]:CORRection:CSET:ALL[:STATe]?

**功能描述**

打开或关闭所有幅度校正功能。

查询幅度校正功能的状态。

**参数**

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**说明**

选择打开时，所有的校正设置被打开，默认启用“天线”校正。

选择关闭时，所有的校正设置被禁用。

打开或关闭单个校正设置，请参考[:SENSe]:CORRection:CSET&lt;n&gt;[:STATe]命令。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开幅度校正功能。

[:SENSe]:CORRection:CSET:ALL:STATe ON 或 :SENSe:CORRection:CSET:ALL:STATe 1

下面的查询返回 1。

[:SENSe]:CORRection:CSET:ALL:STATe?

## [:SENSe]:CORRection:CSET<n>:DATA

### 命令格式

```
[:SENSe]:CORRection:CSET<n>:DATA <freq>,<rel_ampl>{,<freq>,<rel_ampl>}
[:SENSe]:CORRection:CSET<n>:DATA?
```

### 功能描述

使用指定数据创建一条幅度校正曲线。  
查询指定的幅度校正数据。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
<freq>	连续实型	0 Hz 至 1 GHz	——
<rel_ampl>	连续实型	-120 dB 至 100 dB	——

### 说明

<n>: 1 表示天线、2 表示电缆、3 表示其它、4 表示用户。  
可编辑点数范围为 1 至 200。

### 返回格式

查询以<freq>,<rel\_ampl>{,<freq>,<rel\_ampl>}格式返回指定的幅度校正数据。

### 举例

下面的命令创建一条幅度校正曲线。

```
:SENSe:CORRection:CSET1:DATA 900E6,0.3,1.0E9,0.35,1.3E9,0.2
```

下面的查询返回 900000000,0.300000,1000000000,0.350000,1300000000,0.200000。

```
:SENSe:CORRection:CSET1:DATA?
```

## [:SENSe]:CORRection:CSET<n>:DATA:MERGe

### 命令格式

```
[:SENSe]:CORRection:CSET<n>:DATA:MERGe <freq>,<rel_ampl>{,<freq>,<rel_ampl>}
```

### 功能描述

在当前校正曲线上添加校正数据。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
<freq>	连续实型	0 Hz 至 1 GHz	——
<rel_ampl>	连续实型	-120 dB 至 100 dB	——

### 说明

<n>: 1 表示天线、2 表示电缆、3 表示其它、4 表示用户。  
可编辑点数范围为 1 至 200。

### 举例

下面的命令在当前校正曲线上添加两个点。

```
:SENSe:CORRection:CSET1:DATA:MERGe 950E6,0.32,1.4E9,0.1
```

## [[:SENSe]:CORRection:CSET<n>:DELeTe

### 命令格式

```
[[:SENSe]:CORRection:CSET<n>:DELeTe
```

### 功能描述

删除指定的幅度校正设置。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—

### 说明

<n>: 1 表示天线、2 表示电缆、3 表示其它、4 表示用户。

### 举例

下面的命令删除天线校正设置。

```
:SENSe:CORRection:CSET1:DELeTe
```

## [[:SENSe]:CORRection:CSET<n>[:STATe]

### 命令格式

```
[[:SENSe]:CORRection:CSET<n>[:STATe] OFF|ON|0|1  
[:SENSe]:CORRection:CSET<n>[:STATe]?
```

### 功能描述

打开或关闭指定的幅度校正功能。

查询指定幅度校正功能的状态。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	—
—	布尔型	OFF ON 0 1	OFF 0

### 说明

<n>: 1 表示天线、2 表示电缆、3 表示其它、4 表示用户。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令关闭“天线”幅度校正功能。

```
:SENSe:CORRection:CSET1:STATe 0 或 :SENSe:CORRection:CSET1:STATe OFF
```

下面的查询返回 0。

```
:SENSe:CORRection:CSET1:STATe?
```

## [:SENSe]:CORRection:CSET<n>:X:SPACing

### 命令格式

```
[:SENSe]:CORRection:CSET<n>:X:SPACing LINear|LOGarithmic
[:SENSe]:CORRection:CSET<n>:X:SPACing?
```

### 功能描述

选择幅度校正时频率插补的方式。  
查询幅度校正时频率插补的方式。

### 参数

名称	类型	范围	默认值
<n>	离散型	1 2 3 4	——
——	关键字	LINear LOGarithmic	LINear

### 说明

<n>: 1 表示天线、2 表示电缆、3 表示其它、4 表示用户。  
线性模式 (LINear) 时, 频率以线性单位、幅度以对数单位做插补计算。  
对数模式 (LOGarithmic) 时, 频率和幅度都以对数做插补计算。

### 返回格式

查询返回 LIN 或 LOG。

### 举例

下面的命令设置频率插补方式为线性。

```
:SENSe:CORRection:CSET1:X:SPACing LINear
```

下面的查询返回 LIN。

```
:SENSe:CORRection:CSET1:X:SPACing?
```

## [:SENSe]:CORRection:CSET:TABLE:STATe

### 命令格式

```
[:SENSe]:CORRection:CSET:TABLE:STATe OFF|ON|0|1
[:SENSe]:CORRection:CSET:TABLE:STATe?
```

### 功能描述

打开或关闭校正表。  
查询校正表的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开校正表。

```
:SENSe:CORRection:CSET:TABLE:STATe 1 或 :SENSe:CORRection:CSET:TABLE:STATe ON
```

下面的查询返回 1。

```
:SENSe:CORRection:CSET:TABLE:STATe?
```

## [:SENSe]:DEMod

### 命令格式

```
[:SENSe]:DEMod AM|FM|OFF
[:SENSe]:DEMod?
```

### 功能描述

设置解调类型或关闭解调。  
查询解调类型。

### 参数

名称	类型	范围	默认值
——	关键字	AM FM OFF	OFF

### 说明

AM: 调幅。

FM: 调频。

OFF: 关闭。

选择 AM 或 FM 解调类型时，频谱仪自动打开解调功能，此时，使用 [\[:SENSe\]:DEMod:STATe?](#) 命令查询返回 1。

选择 OFF 时，功能同 [\[:SENSe\]:DEMod:STATe OFF|0](#) 命令，频谱仪关闭解调功能，此时，使用 [\[:SENSe\]:DEMod:STATe?](#) 命令查询返回 0。

### 返回格式

查询返回 AM、FM 或 OFF。

### 举例

下面的命令设置解调类型为调幅。

```
:SENSe:DEMod AM
```

下面的查询返回 AM。

```
:SENSe:DEMod?
```

## [:SENSe]:DEMod:GAIN:AUTO

### 命令格式

```
[:SENSe]:DEMod:GAIN:AUTO OFF|ON|0|1
[:SENSe]:DEMod:GAIN:AUTO?
```

### 功能描述

打开或关闭信号增益的自动设置模式。  
查询信号增益自动设置模式的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 说明

仅当频谱仪打开解调功能（即选择 AM 或 FM 解调类型）时，此命令有效。

### 返回格式

查询返回 0 或 1。

**举例**

下面的命令打开信号增益的自动设置模式。

```
:SENSe:DEMod:GAIN:AUTO ON 或 :SENSe:DEMod:GAIN:AUTO 1
```

下面的查询返回 1。

```
:SENSe:DEMod:GAIN:AUTO?
```

**[:SENSe]:DEMod:GAIN:INCRement****命令格式**

```
[:SENSe]:DEMod:GAIN:INCRement <integer>
```

```
[:SENSe]:DEMod:GAIN:INCRement?
```

**功能描述**

设置信号增益。

查询信号增益。

**参数**

名称	类型	范围	默认值
<integer>	整型	1 至 7	7

**说明**

仅当频谱仪打开解调功能（即选择 AM 或 FM 解调类型）时，此命令有效。

**返回格式**

查询以整数形式返回信号增益。

**举例**

下面的命令设置信号增益为 4。

```
:SENSe:DEMod:GAIN:INCRement 4
```

下面的查询返回 4。

```
:SENSe:DEMod:GAIN:INCRement?
```



## [:SENSe]:DEMod:STATe

### 命令格式

```
[:SENSe]:DEMod:STATe OFF|ON|0|1
[:SENSe]:DEMod:STATe?
```

### 功能描述

打开或关闭解调功能。  
查询解调功能的状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

打开解调功能时，频谱仪默认选择 AM 解调类型。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开解调功能。

```
:SENSe:DEMod:STATe 1 或 :SENSe:DEMod:STATe ON
```

下面的查询返回 1。

```
:SENSe:DEMod:STATe?
```

## [:SENSe]:DEMod:TIME

### 命令格式

```
[:SENSe]:DEMod:TIME <time>
[:SENSe]:DEMod:TIME?
```

### 功能描述

设置解调驻留时间。  
查询解调驻留时间。

### 参数

名称	类型	范围	默认值
<time>	连续实型	5 ms 至 1000 s	100 ms

### 返回格式

查询以科学计数形式返回驻留时间，单位为 s。

### 举例

下面的命令设置驻留时间为 500 ms。

```
:SENSe:DEMod:TIME 0.5 或 :SENSe:DEMod:TIME 500ms
```

下面的查询返回 5.000000E-01。

```
:SENSe:DEMod:TIME?
```

## [:SENSe]:DETEctor[:FUNCTion]

### 命令格式

```
[:SENSe]:DETEctor[:FUNCTion] NEGative|NORMal|POSitive|RMS|SAMPle|VAVerage|QPEak
[:SENSe]:DETEctor[:FUNCTion]?
```

### 功能描述

设置检波类型。  
查询检波类型。

### 参数

名称	类型	范围	默认值
—	关键字	NEGative NORMal POSitive RMS SAMPle VAVerage QPEak	POSitive

### 说明

NEGative: 负峰值。  
NORMal: 标准检波。  
POSitive: 正峰值。  
RMS: 有效值平均。  
SAMPle: 抽样检波。  
VAVerage: 电压平均。  
QPEak: 准峰值。

### 返回格式

查询返回 NEG、NORM、POS、RMS、SAMP、VAV 或 QPEAK。

### 举例

下面的命令设置检波类型为峰值。  
[:SENSe]:DETEctor:FUNCTion QPEak

下面的查询返回 QPEAK。

```
[:SENSe]:DETEctor:FUNCTion?
```

## [:SENSe]:EBWidth:AVERage:COUNT

### 命令格式

```
[:SENSe]:EBWidth:AVERage:COUNT <integer>
[:SENSe]:EBWidth:AVERage:COUNT?
```

### 功能描述

设置发射带宽测量的平均次数。  
查询发射带宽测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当发射带宽测量打开时有效。

### 返回格式

查询以整数形式返回平均次数。

**举例**

下面的命令设置平均次数为 100。

```
:SENSe:EBWidth:AVERage:COUNT 100
```

下面的查询返回 100。

```
:SENSe:EBWidth:AVERage:COUNT?
```

**[[:SENSe]:EBWidth:AVERage[:STATe]]****命令格式**

```
[[:SENSe]:EBWidth:AVERage[:STATe] OFF|ON|0|1
```

```
[[:SENSe]:EBWidth:AVERage[:STATe]?
```

**功能描述**

打开或关闭发射带宽测量的平均测量功能。

查询发射带宽测量的平均测量功能状态。

**参数**

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**说明**

该命令仅当发射带宽测量打开时有效。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开平均 测量功能。

```
:SENSe:EBWidth:AVERage:STATe 1 或 :SENSe:EBWidth:AVERage:STATe ON
```

下面的查询返回 1。

```
:SENSe:EBWidth:AVERage:STATe?
```

## [:SENSe]:EBWidth:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:EBWidth:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:EBWidth:AVERage:TCONtrol?
```

### 功能描述

设置发射带宽测量时的平均模式。  
查询发射带宽测量时的平均模式。

### 参数

名称	类型	范围	默认值
——	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均。

REPeat: 重复平均。

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:EBWidth:AVERage:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:EBWidth:AVERage:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当发射带宽测量打开时有效。

### 返回格式

查询返回 EXP 或 REP。

### 举例

下面的命令设置平均模式为重复平均。

```
[:SENSe]:EBWidth:AVERage:TCONtrol REPeat
```

下面的查询返回 REP。

```
[:SENSe]:EBWidth:AVERage:TCONtrol?
```

## [:SENSe]:EBWidth:FREQuency:SPAN

### 命令格式

```
[:SENSe]:EBWidth:FREQuency:SPAN <freq>
[:SENSe]:EBWidth:FREQuency:SPAN?
```

### 功能描述

设置发射带宽测量的扫宽。  
查询发射带宽测量的扫宽。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	100 Hz 至 1 GHz	2 MHz

### 说明

该命令仅当发射带宽测量打开时有效。

该设置更改频谱仪的扫宽。

### 返回格式

查询以整数形式返回扫宽，单位为 Hz。

### 举例

下面的命令设置扫宽为 10 MHz。

```
:SENSe:EBWidth:FREQuency:SPAN 10000000
```

下面的查询返回 10000000。

```
:SENSe:EBWidth:FREQuency:SPAN?
```

## [[:SENSe]:EBWidth:MAXHold:STATe

### 命令格式

```
[[:SENSe]:EBWidth:MAXHold:STATe OFF|ON|0|1
```

```
:SENSe]:EBWidth:MAXHold:STATe?
```

### 功能描述

打开或关闭最大保持。

查询最大保持的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当发射带宽测量打开时有效。

打开最大保持，则将每次测量结果与上次结果比较，取最大值显示。

关闭最大保持，则显示当前测量结果。

最大保持与平均测量互斥，打开最大保持则自动关闭平均测量。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开最大保持。

```
:SENSe:EBWidth:MAXHold:STATe 1 或 :SENSe:EBWidth:MAXHold:STATe ON
```

下面的查询返回 1。

```
:SENSe:EBWidth:MAXHold:STATe?
```

**[[:SENSe]:EBWidth:XDB****命令格式**

```
[[:SENSe]:EBWidth:XDB <real_amp>
[:SENSe]:EBWidth:XDB?
```

**功能描述**

设置 EBW X dB 值。  
查询 EBW X dB 值。

**参数**

名称	类型	范围	默认值
<real_amp>	连续实型	-100 dB 至 -0.1 dB	-10 dB

**说明**

该命令仅当发射带宽测量打开时有效。

**返回格式**

查询以科学计数形式返回 X dB 值。

**举例**

下面的命令设置 X dB 值为 -20 dB。

```
[:SENSe:EBWidth:XDB -20
```

下面的查询返回 -2.000000E+01。

```
[:SENSe:EBWidth:XDB?
```

**[[:SENSe]:EXTRef[:STATe]?****命令格式**

```
[[:SENSe]:EXTRef[:STATe]?
```

**功能描述**

查询仪器当前使用的参考状态。

**说明**

返回 1: 仪器使用外部参考晶振;

返回 0: 仪器使用内部参考晶振。

**返回格式**

查询返回 1 或 0。

## **[[:SENSe]:FREQuency:CENTer**

### 命令格式

[[:SENSe]:FREQuency:CENTer <freq>  
[:SENSe]:FREQuency:CENTer?

### 功能描述

设置中心频率。  
查询中心频率。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	0 Hz 至 1 GHz	500 MHz

### 返回格式

查询以整数形式返回中心频率，单位为 Hz。

### 举例

下面的命令设置中心频率为 1 MHz。

```
[:SENSe:FREQuency:CENTer 1000000
```

下面的查询返回 1000000。

```
[:SENSe:FREQuency:CENTer?
```

## **[[:SENSe]:FREQuency:CENTer:DOWN**

### 命令格式

[[:SENSe]:FREQuency:CENTer:DOWN

### 功能描述

以当前中频步长值为步进减小中心频率。

## **[[:SENSe]:FREQuency:CENTer:SET:STEP**

### 命令格式

[[:SENSe]:FREQuency:CENTer:SET:STEP

### 功能描述

将当前的中心频率设置为中频步长。

## [:SENSe]:FREQuency:CENTer:STEP:AUTO

### 命令格式

```
[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF|ON|0|1
[:SENSe]:FREQuency:CENTer:STEP:AUTO?
```

### 功能描述

打开或关闭中频步长的自动设置模式。  
查询中频步长的自动设置模式状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开自动设置模式。

```
:SENSe:FREQuency:CENTer:STEP:AUTO 1 或 :SENSe:FREQuency:CENTer:STEP:AUTO ON
```

下面的查询返回 1。

```
:SENSe:FREQuency:CENTer:STEP:AUTO?
```

## [:SENSe]:FREQuency:CENTer:STEP[:INCRement]

### 命令格式

```
[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>
[:SENSe]:FREQuency:CENTer:STEP[:INCRement]?
```

### 功能描述

设置中频步长。  
查询中频步长。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	1 Hz 至 1 GHz	100 MHz

### 返回格式

查询以整数形式返回中频步长，单位为 Hz。

### 举例

下面的命令设置中频步长为 100 kHz。

```
:SENSe:FREQuency:CENTer:STEP:INCRement 100000 或 :SENSe:FREQuency:CENTer:STEP:INCRement 100 kHz
```

下面的查询返回 100000。

```
:SENSe:FREQuency:CENTer:STEP:INCRement?
```



## [:SENSe]:FREQuency:CENTer:UP

### 命令格式

```
[:SENSe]:FREQuency:CENTer:UP
```

### 功能描述

以当前中频步长值为步进增大中心频率。

## [:SENSe]:FREQuency:OFFSet

### 命令格式

```
[:SENSe]:FREQuency:OFFSet <freq>
[:SENSe]:FREQuency:OFFSet?
```

### 功能描述

设置频率偏移。  
查询频率偏移。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	-100 GHz 至 100 GHz	0 Hz

### 返回格式

查询以整数形式返回频率偏移，单位为 Hz。  
频率偏移不影响频谱仪的任何硬件设置，仅改变中心频率、起始频率和终止频率的显示值。

### 举例

下面的命令设置频率偏移为 1 MHz。

```
:SENSe:FREQuency:OFFSet 1000000
```

下面的查询返回 1000000。

```
:SENSe:FREQuency:OFFSet?
```

## [:SENSe]:FREQuency:SPAN

### 命令格式

```
[:SENSe]:FREQuency:SPAN <freq>
[:SENSe]:FREQuency:SPAN?
```

### 功能描述

设置扫宽。  
查询扫宽。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	0 Hz 至 1 GHz	1 GHz

### 说明

当扫宽设为 0 时，频谱仪为零扫宽模式，横轴由频率变为时间，频谱仪只显示频率等于中心频率的信号。

### 返回格式

查询以整数形式返回扫宽值，单位为 Hz。

### 举例

下面的命令设置扫宽为 20 MHz。

```
:SENSe:FREQuency:SPAN 20000000
```

下面的查询返回 20000000。

```
:SENSe:FREQuency:SPAN?
```

## **[[:SENSe]:FREQuency:SPAN:FULL**

### 命令格式

```
[[:SENSe]:FREQuency:SPAN:FULL
```

### 功能描述

将扫宽设置为最大值（全扫宽）。

## **[[:SENSe]:FREQuency:SPAN:PREVious**

### 命令格式

```
[[:SENSe]:FREQuency:SPAN:PREVious
```

### 功能描述

将扫宽设置为上次设置的扫宽值。

## **[[:SENSe]:FREQuency:SPAN:ZIN**

### 命令格式

```
[[:SENSe]:FREQuency:SPAN:ZIN
```

### 功能描述

设置扫宽为当前值的 1/2。

## **[[:SENSe]:FREQuency:SPAN:ZOUT**

### 命令格式

```
[[:SENSe]:FREQuency:SPAN:ZOUT
```

### 功能描述

设置扫宽为当前值的 2 倍。

## [[:SENSe]:FREQuency:START

### 命令格式

```
[[:SENSe]:FREQuency:START <freq>
[:SENSe]:FREQuency:START?
```

### 功能描述

设置起始频率。  
查询起始频率。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	0 Hz 至 1 GHz	0 Hz

### 返回格式

查询以整数形式返回起始频率，单位为 Hz。

### 举例

下面的命令设置起始频率为 10 MHz。

```
:SENSe:FREQuency:START 10000000
```

下面的查询返回 10000000。

```
:SENSe:FREQuency:START?
```

## [[:SENSe]:FREQuency:STOP

### 命令格式

```
[[:SENSe]:FREQuency:STOP <freq>
[:SENSe]:FREQuency:STOP?
```

### 功能描述

设置终止频率。  
查询终止频率。

### 参数

名称	类型	范围	默认值
<freq>	连续实型	0 Hz 至 1 GHz	1 GHz

### 返回格式

查询以整数形式返回终止频率，单位为 Hz。

### 举例

下面的命令设置终止频率为 10 MHz。

```
:SENSe:FREQuency:STOP 10000000
```

下面的查询返回 10000000。

```
:SENSe:FREQuency:STOP?
```

## [[:SENSe]:HDISt:AVERage:COUNT

### 命令格式

```
[[:SENSe]:HDISt:AVERage:COUNT <integer>
[:SENSe]:HDISt:AVERage:COUNT?
```

### 功能描述

设置谐波失真测量的平均次数。  
查询谐波失真测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当谐波失真测量打开时有效。

### 返回格式

查询以整数形式返回平均次数。

### 举例

下面的命令设置平均次数为 100。

```
:SENSe:HDISt:AVERage:COUNT 100
```

下面的查询返回 100。

```
:SENSe:HDISt:AVERage:COUNT?
```

## [[:SENSe]:HDISt:AVERage[:STATe]

### 命令格式

```
[[:SENSe]:HDISt:AVERage[:STATe] OFF|ON|0|1
[:SENSe]:HDISt:AVERage[:STATe]?
```

### 功能描述

打开或关闭谐波失真测量的平均测量功能。  
查询谐波失真测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当谐波失真测量打开时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均 测量功能。

```
:SENSe:HDISt:AVERage:STATe 1 或 :SENSe:HDISt:AVERage:STATe ON
```

下面的查询返回 1。

```
:SENSe:HDISt:AVERage:STATe?
```

## [:SENSe]:HDISt:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:HDISt:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:HDISt:AVERage:TCONtrol?
```

### 功能描述

设置谐波失真测量时的平均模式。  
查询谐波失真测量时的平均模式。

### 参数

名称	类型	范围	默认值
—	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均。

REPeat: 重复平均。

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:HDISt:AVERage:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:HDISt:AVERage:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当谐波失真测量打开时有效。

### 返回格式

查询返回 EXP 或 REP。

### 举例

下面的命令设置平均模式为重复平均。

```
[:SENSe]:HDISt:AVERage:TCONtrol REPeat
```

下面的查询返回 REP。

```
[:SENSe]:HDISt:AVERage:TCONtrol?
```

## [:SENSe]:HDISt:NUMBers

### 命令格式

```
[:SENSe]:HDISt:NUMBers <integer>
[:SENSe]:HDISt:NUMBers?
```

### 功能描述

设置待测量的谐波数量。  
查询待测量的谐波数量。

### 参数

名称	类型	范围	默认值
<integer>	整型	2 至 10	10

### 说明

该命令仅当谐波失真测量打开时有效。

**返回格式**

查询以整数形式返回谐波数量。

**举例**

下面的命令设置谐波数量为 5。

```
:SENSe:HDISt:NUMBers 5
```

下面的查询返回 5。

```
:SENSe:HDISt:NUMBers?
```

**[:SENSe]:HDISt:TIME****命令格式**

```
[:SENSe]:HDISt:TIME <time>
```

```
[:SENSe]:HDISt:TIME?
```

**功能描述**

设置谐波失真测量时的扫描时间。

查询谐波失真测量时的扫描时间。

**参数**

名称	类型	范围	默认值
<time>	连续实型	20 us 至 1 ks	33.333 ms

**说明**

该命令仅当谐波失真测量打开时有效。

**返回格式**

查询以科学计数形式返回扫描时间，单位为 s。

**举例**

下面的命令设置扫描时间为 100 ms。

```
:SENSe:HDISt:TIME 0.1
```

下面的查询返回 1.000000E-01。

```
:SENSe:HDISt:TIME?
```

## [:SENSe]:HDISt:TIME:AUTO[:STATe]

### 命令格式

```
[:SENSe]:HDISt:TIME:AUTO[:STATe] OFF|ON|0|1
[:SENSe]:HDISt:TIME:AUTO[:STATe]?
```

### 功能描述

打开或关闭谐波失真测量时扫描时间的自动设置模式。  
查询谐波失真测量时扫描时间的自动设置模式状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 说明

该命令仅当谐波失真测量打开时有效。  
该设置更改频谱仪的扫描时间。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开扫描时间的自动设置模式。

```
:SENSe:HDISt:TIME:AUTO:STATe 1 或 :SENSe:HDISt:TIME:AUTO:STATe ON
```

下面的查询返回 1。

```
:SENSe:HDISt:TIME:AUTO:STATe?
```

## [:SENSe]:OBWidth:AVERAge:COUNT

### 命令格式

```
[:SENSe]:OBWidth:AVERAge:COUNT <integer>
[:SENSe]:OBWidth:AVERAge:COUNT?
```

### 功能描述

设置占用带宽测量的平均次数。  
查询占用带宽测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当占用带宽测量打开时有效。

### 返回格式

查询以整数形式返回平均次数。

### 举例

下面的命令设置平均次数为 100。

```
:SENSe:OBWidth:AVERAge:COUNT 100
```

下面的查询返回 100。

```
:SENSe:OBWidth:AVERAge:COUNT?
```

## [:SENSe]:OBWidth:AVERage[:STATe]

### 命令格式

```
[:SENSe]:OBWidth:AVERage[:STATe] OFF|ON|0|1
[:SENSe]:OBWidth:AVERage[:STATe]?
```

### 功能描述

打开或关闭占用带宽测量的平均测量功能。  
查询占用带宽测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当占用带宽测量打开时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均测量功能。

```
:SENSe:OBWidth:AVERage:STATe 1 或 :SENSe:OBWidth:AVERage:STATe ON
```

下面的查询返回 1。

```
:SENSe:OBWidth:AVERage:STATe?
```

## [:SENSe]:OBWidth:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:OBWidth:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:OBWidth:AVERage:TCONtrol?
```

### 功能描述

设置占用带宽测量时的平均模式。  
查询占用带宽测量时的平均模式。

### 参数

名称	类型	范围	默认值
—	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均。

REPeat: 重复平均。

指数平均时, 计算结果为对最近的 N 次 (由 [\[:SENSe\]:OBWidth:AVERage:COUNT](#) 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由 [\[:SENSe\]:OBWidth:AVERage:COUNT](#) 命令指定) 测量结果做算术平均所得的值。

该命令仅当占用带宽测量打开时有效。

### 返回格式

查询返回 EXP 或 REP。



**举例**

下面的命令设置平均模式为重复平均。

```
:SENSe:OBWidth:AVERAge:TCONtrol REPeat
```

下面的查询返回 REP。

```
:SENSe:OBWidth:AVERAge:TCONtrol?
```

**[[:SENSe]:OBWidth:FREQuency:SPAN****命令格式**

```
[[:SENSe]:OBWidth:FREQuency:SPAN <freq>
```

```
[[:SENSe]:OBWidth:FREQuency:SPAN?
```

**功能描述**

设置占用带宽测量的扫宽。

查询占用带宽测量的扫宽。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	100 Hz 至 1 GHz	2 MHz

**说明**

该命令仅当占用带宽测量打开时有效。

该设置更改频谱仪的扫宽。

**返回格式**

查询以整数形式返回扫宽值，单位为 Hz。

**举例**

下面的命令设置扫宽为 1 MHz。

```
:SENSe:OBWidth:FREQuency:SPAN 1000000
```

下面的查询返回 1000000。

```
:SENSe:OBWidth:FREQuency:SPAN?
```

## [:SENSe]:OBWidth:MAXHold:STATe

### 命令格式

```
[:SENSe]:OBWidth:MAXHold:STATe OFF|ON|0|1
[:SENSe]:OBWidth:MAXHold:STATe?
```

### 功能描述

打开或关闭占用带宽测量的最大保持。  
查询占用带宽测量的最大保持状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当占用带宽测量打开时有效。  
打开最大保持，则将每次测量结果与上次结果比较，取最大值显示。  
关闭最大保持，则显示当前测量结果。  
最大保持与平均测量模式互斥，打开最大保持则自动关闭平均测量模式。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开最大保持功能。

```
:SENSe:OBWidth:MAXHold:STATe 1 或 :SENSe:OBWidth:MAXHold:STATe ON
```

下面的查询返回 1。

```
:SENSe:OBWidth:MAXHold:STATe?
```

## [:SENSe]:OBWidth:PERCent

### 命令格式

```
[:SENSe]:OBWidth:PERCent <real>
[:SENSe]:OBWidth:PERCent?
```

### 功能描述

设置信号功率占整个扫宽功率的百分数（功率比）。  
查询占用带宽测量的功率比。

### 参数

名称	类型	范围	默认值
<real>	连续实型	1 至 99.99	99

### 说明

该命令仅当占用带宽测量打开时有效。  
<real>的取值范围 1 至 99.99 对应频谱仪设置的 1%至 99.99%。

### 返回格式

查询以科学计数形式返回百分数。

**举例**

下面的命令设置功率比为 90%。

```
[:SENSe:OBWidth:PERCent 90
```

下面的查询返回 9.000000E+01。

```
[:SENSe:OBWidth:PERCent?
```

**[:SENSe]:POWer:ARANge****命令格式**

```
[:SENSe]:POWer:ARANge
```

**功能描述**

执行自动量程操作，即在当前扫宽内调节幅度相关参数，使信号易于观察。

**[:SENSe]:POWer:ASCale****命令格式**

```
[:SENSe]:POWer:ASCale
```

**功能描述**

执行自动定标操作，即自动调整参考电平和刻度大小，尽量将信号的峰值放在最上面的网格内，以便于观察迹线。

**[:SENSe]:POWer:ATUNE****命令格式**

```
[:SENSe]:POWer:ATUNE
```

**功能描述**

在全频段内搜索信号，并将频率和幅度参数调整到最佳状态。

**[:SENSe]:POWer[:RF]:ATTenuation****命令格式**

```
[:SENSe]:POWer[:RF]:ATTenuation <rel_ampl>
```

```
[:SENSe]:POWer[:RF]:ATTenuation?
```

**功能描述**

设置射频前端衰减器的衰减值。

查询射频前端衰减器的衰减值。

**参数**

名称	类型	范围	默认值
<rel_ampl>	整型	0 dB 至 30 dB	10 dB

**返回格式**

查询以整数形式返回衰减值，单位为 dB。

**举例**

下面的命令设置衰减值为 20 dB。

```
:SENSe:POWer:RF:ATTenuation 20
```

下面的查询返回 20。

```
:SENSe:POWer:RF:ATTenuation?
```

**[:SENSe]:POWer[:RF]:ATTenuation:AUTO****命令格式**

```
[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF|ON|0|1
```

```
[:SENSe]:POWer[:RF]:ATTenuation:AUTO?
```

**功能描述**

打开或关闭输入衰减的自动设置方式。

查询输入衰减的自动设置方式状态。

**参数**

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	ON 1

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令关闭输入衰减的自动设置方式。

```
:SENSe:POWer:RF:ATTenuation:AUTO 0 或 :SENSe:POWer:RF:ATTenuation:AUTO OFF
```

下面的查询返回 0。

```
:SENSe:POWer:RF:ATTenuation:AUTO?
```

## [[:SENSe]:POWer[:RF]:GAIN[:STATe]

### 命令格式

```
[[:SENSe]:POWer[:RF]:GAIN[:STATe] OFF|ON|0|1
[:SENSe]:POWer[:RF]:GAIN[:STATe]?
```

### 功能描述

打开或关闭前置放大器。  
查询前置放大器的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开前置放大器。

```
:SENSe:POWer:RF:GAIN:STATe 1 或 :SENSe:POWer:RF:GAIN:STATe ON
```

下面的查询返回 1。

```
:SENSe:POWer:RF:GAIN:STATe?
```

## [[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer]

### 命令格式

```
[[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer] <ampl>
[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer]?
```

### 功能描述

设置输入混频器的最大功率。  
查询输入混频器的最大功率。

### 参数

名称	类型	范围	默认值
<ampl>	整型	-30 dBm 至 0 dBm	-10 dBm

### 返回格式

查询以科学计数形式返回输入混频器的最大功率，单位为 dBm。

### 举例

下面的命令设置输入混频器的最大功率为-20 dBm。

```
:SENSe:POWer:RF:MIXer:RANGe:UPPer -20
```

下面的查询返回-2.000000E+01。

```
:SENSe:POWer:RF:MIXer:RANGe:UPPer?
```

## [:SENSe]:SIGCapture[:STATe]

### 命令格式

```
[:SENSe]:SIGCapture[:STATe] OFF|ON|0|1
[:SENSe]:SIGCapture[:STATe]?
```

### 功能描述

打开或关闭 Sig Capture 功能。  
查询 Sig Capture 功能的开关状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开 Sig Capture 功能。

```
:SENSe:SIGCapture:STATe ON 或 :SENSe:SIGCapture:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:STATe?
```

## [:SENSe]:SIGCapture:SIGC[:STATe]

### 命令格式

```
[:SENSe]:SIGCapture:SIGC[:STATe] OFF|ON|0|1
[:SENSe]:SIGCapture:SIGC[:STATe]?
```

### 功能描述

打开或关闭实时迹线。  
查询实时迹线的开关状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

实时迹线与最大保持和 2FSK 互斥。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开实时迹线。

```
:SENSe:SIGCapture:SIGC:STATe ON 或 :SENSe:SIGCapture:SIGC:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:SIGC:STATe?
```

## **[[:SENSe]:SIGCapture:MAXHold[:STATe]**

### 命令格式

```
[[:SENSe]:SIGCapture:MAXHold[:STATe] OFF|ON|0|1
[:SENSe]:SIGCapture:MAXHold[:STATe]?
```

### 功能描述

打开或关闭最大保持。  
查询最大保持的开关状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

最大保持与实时迹线和 2FSK 互斥。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开最大保持。

```
:SENSe:SIGCapture:MAXHold:STATe ON 或 :SENSe:SIGCapture:MAXHold:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:MAXHold:STATe?
```

## **[[:SENSe]:SIGCapture:RESet**

### 命令格式

```
[[:SENSe]:SIGCapture:RESet
```

### 功能描述

执行最大保持重置操作。

### 说明

该命令仅当打开 Sig Capture 功能时有效。

## **[[:SENSe]:SIGCapture:2FSK[:STATe]**

### 命令格式

```
[[:SENSe]:SIGCapture:2FSK[:STATe] OFF|ON|0|1
[:SENSe]:SIGCapture:2FSK[:STATe]?
```

### 功能描述

打开或关闭 2FSK。  
查询 2FSK 的开关状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

**说明**

2FSK 与实时迹线和最大保持互斥。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开最大保持。

```
:SENSe:SIGCapture:2FSK:STATe ON 或 :SENSe:SIGCapture:2FSK:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:2FSK:STATe?
```

**[:SENSe]:SIGCapture:2FSK:RESet****命令格式**

```
[:SENSe]:SIGCapture:2FSK:RESet
```

**功能描述**

执行 2FSK 最大保持重置操作。

**说明**

该命令仅当打开 2FSK 功能时有效。

**[:SENSe]:SIGCapture:2FSK:MAXHold[:STATe]****命令格式**

```
[:SENSe]:SIGCapture:2FSK:MAXHold[:STATe] OFF|ON|0|1  
[:SENSe]:SIGCapture:2FSK:MAXHold[:STATe]?
```

**功能描述**

打开或关闭 2FSK 的最大保持。

查询 2FSK 最大保持的开关状态。

**参数**

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开 2FSK 最大保持。

```
:SENSe:SIGCapture:2FSK:MAXHold:STATe ON 或 :SENSe:SIGCapture:2FSK:MAXHold:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:2FSK:MAXHold:STATe?
```



## [:SENSe]:SIGCapture:2FSK:PFSWitch

### 命令格式

```
[:SENSe]:SIGCapture:2FSK:PFSWitch OFF|ON|0|1
[:SENSe]:SIGCapture:2FSK:PFSWitch?
```

### 功能描述

打开或关闭 2FSK 通过/失败功能。  
查询 2FSK 通过/失败的开关状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当打开 2FSK 功能时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开 2FSK 通过/失败功能。

```
:SENSe:SIGCapture:2FSK:PFSWitch ON 或 :SENSe:SIGCapture:2FSK:PFSWitch 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:2FSK:PFSWitch?
```

## [:SENSe]:SIGCapture:2FSK:SIGNal

### 命令格式

```
[:SENSe]:SIGCapture:2FSK:SIGNal 0|1|2
[:SENSe]:SIGCapture:2FSK:SIGNal?
```

### 功能描述

选择通过/失败测试中 2FSK 当前需要修改的信号。  
查询通过/失败测试中 2FSK 当前需要修改的信号。

### 参数

名称	类型	范围	默认值
—	离散型	0 1 2	0

### 说明

0: 选择修改 2FSK 的信号 1, 对应于峰值 1 和峰值 2。

1: 选择修改 2FSK 的信号 2, 对应于峰值 3 和峰值 4。

2: 选择修改 2FSK 的信号 3, 对应于峰值 5 和峰值 6。

该命令仅当打开 2FSK 功能时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令选择修改 2FSK 的信号 1。

```
:SENSe:SIGCapture:2FSK:SIGNaI 0
```

下面的查询返回 0。

```
:SENSe:SIGCapture:2FSK:SIGNaI?
```

## [[:SENSe]:SIGCapture:2FSK:AMPUp

### 命令格式

```
[[:SENSe]:SIGCapture:2FSK:AMPUp <ample>
```

```
[[:SENSe]:SIGCapture:2FSK:AMPUp?
```

### 功能描述

设置指定 2FSK 信号的幅度上限值。

查询指定 2FSK 信号的幅度上限值。

### 参数

名称	类型	范围	默认值
<ample>	连续实型	幅度下限值至 320 dBm	-10 dBm

### 说明

该命令仅当打开 2FSK 功能时有效。

### 返回格式

查询以科学计数形式返回幅度上限值。

### 举例

下面的命令设置幅度上限值为 -20 dBm。

```
:SENSe:SIGCapture:2FSK:AMPUp -20
```

下面的查询返回 -2.000000E+01。

```
:SENSe:SIGCapture:2FSK:AMPUp?
```

## [[:SENSe]:SIGCapture:2FSK:AMPDown

### 命令格式

```
[[:SENSe]:SIGCapture:2FSK:AMPDown <ample>
```

```
[[:SENSe]:SIGCapture:2FSK:AMPDown?
```

### 功能描述

设置指定 2FSK 信号的幅度下限值。

查询指定 2FSK 信号的幅度下限值。

### 参数

名称	类型	范围	默认值
<ample>	连续实型	-400 dBm 至幅度上限值	-100 dBm

### 说明

该命令仅当打开 2FSK 功能时有效。

### 返回格式

查询以科学计数形式返回幅度下限值。

**举例**

下面的命令设置幅度下限值为 -20 dBm。

```
:SENSe:SIGCapture:2FSK:AMPDown -20
```

下面的查询返回 -2.000000E+01。

```
:SENSe:SIGCapture:2FSK:AMPDown?
```

**[[:SENSe]:SIGCapture:2FSK:MARK1[:FREQ]]****命令格式**

```
[[:SENSe]:SIGCapture:2FSK:MARK1[:FREQ] <freq>
```

```
[[:SENSe]:SIGCapture:2FSK:MARK1[:FREQ]?
```

**功能描述**

设置游标 1 处的频率值。

查询游标 1 处的频率值。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	(中心频率-Span/2) 至 (中心频率+ Span/2) 其中, Span 固定为 1.5 MHz	中心频率+ Span/2

**说明**

该命令仅当打开 2FSK 功能时有效。

**返回格式**

查询以整数形式返回游标 1 处的频率值, 单位为 Hz。

**举例**

下面的命令设置游标 1 处的频率值为 750 MHz。

```
:SENSe:SIGCapture:2FSK:MARK1:FREQ 750000000
```

下面的查询返回 1000000。

```
:SENSe:SIGCapture:2FSK:MARK1:FREQ?
```

**[[:SENSe]:SIGCapture:2FSK:MARK1:Switch[:STATe]]****命令格式**

```
[[:SENSe]:SIGCapture:2FSK:MARK1:Switch[:STATe] OFF|ON|0|1
```

```
[[:SENSe]:SIGCapture:2FSK:MARK1:Switch[:STATe]?
```

**功能描述**

打开或关闭游标 1。

查询游标 1 的开关状态。

**参数**

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**说明**

该命令仅当打开 2FSK 功能时有效。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开游标 1。

```
:SENSe:SIGCapture:2FSK:MARK1:Switch:STATe ON 或 :SENSe:SIGCapture:2FSK:MARK1:Switch:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:2FSK:MARK1:Switch:STATe?
```

**[[:SENSe]:SIGCapture:2FSK:MARK2[:FREQ]****命令格式**

```
[[:SENSe]:SIGCapture:2FSK:MARK2[:FREQ] <freq>
```

```
[[:SENSe]:SIGCapture:2FSK:MARK2[:FREQ]?
```

**功能描述**

设置游标 2 处的频率值。

查询游标 2 处的频率值。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	(中心频率-Span/2) 至 (中心频率+ Span/2) 其中, Span 固定为 1.5 MHz	中心频率+ Span/2

**说明**

该命令仅当打开 2FSK 功能时有效。

**返回格式**

查询以整数形式返回游标 2 处的频率值, 单位为 Hz。

**举例**

下面的命令设置游标 2 处的频率值为 750 MHz。

```
:SENSe:SIGCapture:2FSK:MARK2:FREQ 750000000
```

下面的查询返回 1000000。

```
:SENSe:SIGCapture:2FSK:MARK2:FREQ?
```

**[[:SENSe]:SIGCapture:2FSK:MARK2:Switch[:STATe]****命令格式**

```
[[:SENSe]:SIGCapture:2FSK:MARK2:Switch[:STATe] OFF|ON|0|1
```

```
[[:SENSe]:SIGCapture:2FSK:MARK2:Switch[:STATe]?
```

**功能描述**

打开或关闭游标 2。

查询游标 2 的开关状态。

**参数**

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

**说明**

该命令仅当打开 2FSK 功能时有效。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开游标 2。

```
:SENSe:SIGCapture:2FSK:MARK2:Switch:STATe ON 或 :SENSe:SIGCapture:2FSK:MARK2:Switch:STATe 1
```

下面的查询返回 1。

```
:SENSe:SIGCapture:2FSK:MARK2:Switch:STATe?
```

**[[:SENSe]:SWEep:COUNT****命令格式**

```
[[:SENSe]:SWEep:COUNT <integer>
[:SENSe]:SWEep:COUNT?
```

**功能描述**

设置单次扫描时的扫描次数。

查询单次扫描时的扫描次数。

**参数**

名称	类型	范围	默认值
<integer>	整型	1 至 9999	1

**返回格式**

查询以整数形式返回扫描次数。

**举例**

下面的命令设置扫描次数为 10。

```
:SENSe:SWEep:COUNT 10
```

下面的查询返回 10。

```
:SENSe:SWEep:COUNT?
```

**[[:SENSe]:SWEep:COUNT:CURRent?****命令格式**

```
[[:SENSe]:SWEep:COUNT:CURRent?
```

**功能描述**

查询单次扫描时当前已扫描的次数。

**说明**

[\[:SENSe\]:SWEep:COUNT?](#) 命令查询单次扫描时需要执行的扫描次数；该命令查询单次扫描时已执行的扫描次数。

**返回格式**

查询以整数形式返回次数值。

## [:SENSe]:SWEep:TIME

### 命令格式

```
[:SENSe]:SWEep:TIME <time>
[:SENSe]:SWEep:TIME?
```

### 功能描述

设置扫描时间。  
查询扫描时间。

### 参数

名称	类型	范围	默认值
<time>	连续实型	20 us 至 1000 s	33.333 ms

### 返回格式

查询以科学计数形式返回扫描时间，单位为 s。

### 举例

下面的命令设置扫描时间为 0.1 s。

```
:SENSe:SWEep:TIME 0.1
```

下面的查询返回 1.000000E-01。

```
:SENSe:SWEep:TIME?
```

## [:SENSe]:SWEep:TIME:AUTO

### 命令格式

```
[:SENSe]:SWEep:TIME:AUTO OFF|ON|0|1
[:SENSe]:SWEep:TIME:AUTO?
```

### 功能描述

打开或关闭自动扫描时间。  
查询自动扫描时间的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开自动扫描时间。

```
:SENSe:SWEep:TIME:AUTO 1 或 :SENSe:SWEep:TIME:AUTO ON
```

下面的查询返回 1。

```
:SENSe:SWEep:TIME:AUTO?
```

## [:SENSe]:SWEep:TIME:AUTO:RULEs

### 命令格式

```
[:SENSe]:SWEep:TIME:AUTO:RULEs NORMal|ACCuracy
[:SENSe]:SWEep:TIME:AUTO:RULEs?
```

### 功能描述

选择自动扫描时间的设置方法。  
查询自动扫描时间的设置方法。

### 参数

名称	类型	范围	默认值
——	关键字	NORMal ACCuracy	NORMal

### 返回格式

查询返回 NORM 或 ACC。

### 举例

下面的命令选择自动扫描时间的设置方法为精确。

```
:SENSe:SWEep:TIME:AUTO:RULEs ACCuracy
```

下面的查询返回 ACC。

```
:SENSe:SWEep:TIME:AUTO:RULEs?
```

## [:SENSe]:TOI:AVERAge:COUNT

### 命令格式

```
[:SENSe]:TOI:AVERAge:COUNT <integer>
[:SENSe]:TOI:AVERAge:COUNT?
```

### 功能描述

设置三阶互调失真测量的平均次数。  
查询三阶互调失真测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当三阶互调失真测量打开时有效。

### 返回格式

查询以整数形式返回平均次数。

### 举例

下面的命令设置平均次数为 100。

```
:SENSe:TOI:AVERAge:COUNT 100
```

下面的查询返回 100。

```
:SENSe:TOI:AVERAge:COUNT?
```

## [:SENSe]:TOI:AVERage[:STATe]

### 命令格式

```
[:SENSe]:TOI:AVERage[:STATe] OFF|ON|0|1
[:SENSe]:TOI:AVERage[:STATe]?
```

### 功能描述

打开或关闭三阶互调失真测量的平均测量功能。  
查询三阶互调失真测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当三阶互调失真测量打开时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均测量。

```
:SENSe:TOI:AVERage:STATe ON 或 :SENSe:TOI:AVERage:STATe 1
```

下面的查询返回 1。

```
:SENSe:TOI:AVERage:STATe?
```

## [:SENSe]:TOI:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:TOI:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:TOI:AVERage:TCONtrol?
```

### 功能描述

设置三阶互调失真测量时的平均模式。  
查询三阶互调失真测量时的平均模式。

### 参数

名称	类型	范围	默认值
——	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均。

REPeat: 重复平均。

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:TOI:AVERage:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:TOI:AVERage:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当三阶互调失真测量打开时有效。

### 返回格式

查询返回 EXP 或 REP。



**举例**

下面的命令设置平均模式为重复平均。

```
:SENSe:TOI:AVERAge:TCONtrol REPeat
```

下面的查询返回 REP。

```
:SENSe:TOI:AVERAge:TCONtrol?
```

**[[:SENSe]:TOI:FREQuency:SPAN****命令格式**

```
[[:SENSe]:TOI:FREQuency:SPAN <freq>
```

```
[[:SENSe]:TOI:FREQuency:SPAN?
```

**功能描述**

设置三阶互调失真测量的扫宽。

查询三阶互调失真测量的扫宽。

**参数**

名称	类型	范围	默认值
<freq>	连续实型	100 Hz 至 1 GHz	2 MHz

**说明**

该命令仅当三阶互调失真测量打开时有效。

该设置将更改频谱仪的扫宽。

**返回格式**

查询以整数形式返回扫宽，单位为 Hz。

**举例**

下面的命令设置扫宽为 1 MHz。

```
:SENSe:TOI:FREQuency:SPAN 1000000
```

下面的查询返回 1000000。

```
:SENSe:TOI:FREQuency:SPAN?
```

## [:SENSe]:TPOWer:AVERAge:COUNT

### 命令格式

```
[:SENSe]:TPOWer:AVERAge:COUNT <integer>
[:SENSe]:TPOWer:AVERAge:COUNT?
```

### 功能描述

设置时域功率测量的平均次数。  
查询时域功率测量的平均次数。

### 参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	10

### 说明

该命令仅当时域功率测量打开时有效。

### 返回格式

查询以整数形式返回平均次数。

### 举例

下面的命令设置平均次数为 100。

```
:SENSe:TPOWer:AVERAge:COUNT 100
```

下面的查询返回 100。

```
:SENSe:TPOWer:AVERAge:COUNT?
```

## [:SENSe]:TPOWer:AVERAge[:STATe]

### 命令格式

```
[:SENSe]:TPOWer:AVERAge[:STATe] OFF|ON|0|1
[:SENSe]:TPOWer:AVERAge[:STATe]?
```

### 功能描述

打开或关闭时域功率测量的平均测量功能。  
查询时域功率测量的平均测量功能状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅当时域功率测量打开时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开平均测量。

```
:SENSe:TPOWer:AVERAge:STATe ON 或 :SENSe:TPOWer:AVERAge:STATe 1
```

下面的查询返回 1。

```
:SENSe:TPOWer:AVERAge:STATe?
```

## [:SENSe]:TPOWer:AVERage:TCONtrol

### 命令格式

```
[:SENSe]:TPOWer:AVERage:TCONtrol EXPonential|REPeat
[:SENSe]:TPOWer:AVERage:TCONtrol?
```

### 功能描述

设置时域功率测量时的平均模式。  
查询时域功率测量时的平均模式。

### 参数

名称	类型	范围	默认值
—	关键字	EXPonential REPeat	EXPonential

### 说明

EXPonential: 指数平均。

REPeat: 重复平均。

指数平均时, 计算结果为对最近的 N 次 (由[:SENSe]:TPOWer:AVERage:COUNT 命令指定) 测量结果做指数平均所得的值。

重复平均时, 计算结果为对最近的 N 次 (由[:SENSe]:TPOWer:AVERage:COUNT 命令指定) 测量结果做算术平均所得的值。

该命令仅当时域功率测量打开时有效。

### 返回格式

查询返回 EXP 或 REP。

### 举例

下面的命令设置平均模式为重复平均。

```
[:SENSe]:TPOWer:AVERage:TCONtrol REPeat
```

下面的查询返回 REP。

```
[:SENSe]:TPOWer:AVERage:TCONtrol?
```

## [:SENSe]:TPOWer:LLIMit

### 命令格式

```
[:SENSe]:TPOWer:LLIMit <time>
[:SENSe]:TPOWer:LLIMit?
```

### 功能描述

设置时域功率测量的起始线。  
查询时域功率测量的起始线。

### 参数

名称	类型	范围	默认值
<time>	连续实型	0 us 至 终止线当前值	0 us

### 说明

该命令仅当时域功率测量打开时有效。

### 返回格式

查询以科学计数形式返回起始线, 单位为 s。

**举例**

下面的命令设置起始线为 5 ms。

```
:SENSe:TPOWer:LLIMit 0.005
```

下面的查询返回 5.000000E-03。

```
:SENSe:TPOWer:LLIMit?
```

**[[:SENSe]:TPOWer:MODE****命令格式**

```
[[:SENSe]:TPOWer:MODE AVERage|PEAK|RMS
```

```
[[:SENSe]:TPOWer:MODE?
```

**功能描述**

设置时域功率测量的功率类型。

查询时域功率测量的功率类型。

**参数**

名称	类型	范围	默认值
—	关键字	AVERage PEAK RMS	PEAK

**说明**

AVERage: 平均功率。

PEAK: 峰值功率。

RMS: 有效值功率。

该命令仅当时域功率测量打开时有效。

**返回格式**

查询返回 AVER、PEAK 或 RMS。

**举例**

下面的命令设置功率类型为平均功率。

```
:SENSe:TPOWer:MODE AVERage
```

下面的查询返回 AVER。

```
:SENSe:TPOWer:MODE?
```

## [[:SENSe]:TPOWer:RLIMit

### 命令格式

[[:SENSe]:TPOWer:RLIMit <time>

[[:SENSe]:TPOWer:RLIMit?

### 功能描述

设置时域功率测量的终止线。

查询时域功率测量的终止线。

### 参数

名称	类型	范围	默认值
<time>	连续实型	起始线当前值 至 扫描时间当前值	50 ms

### 说明

该命令仅当时域功率测量打开时有效。

### 返回格式

查询以科学计数形式返回终止线值，单位为 s。

### 举例

下面的命令设置终止线为 10 ms。

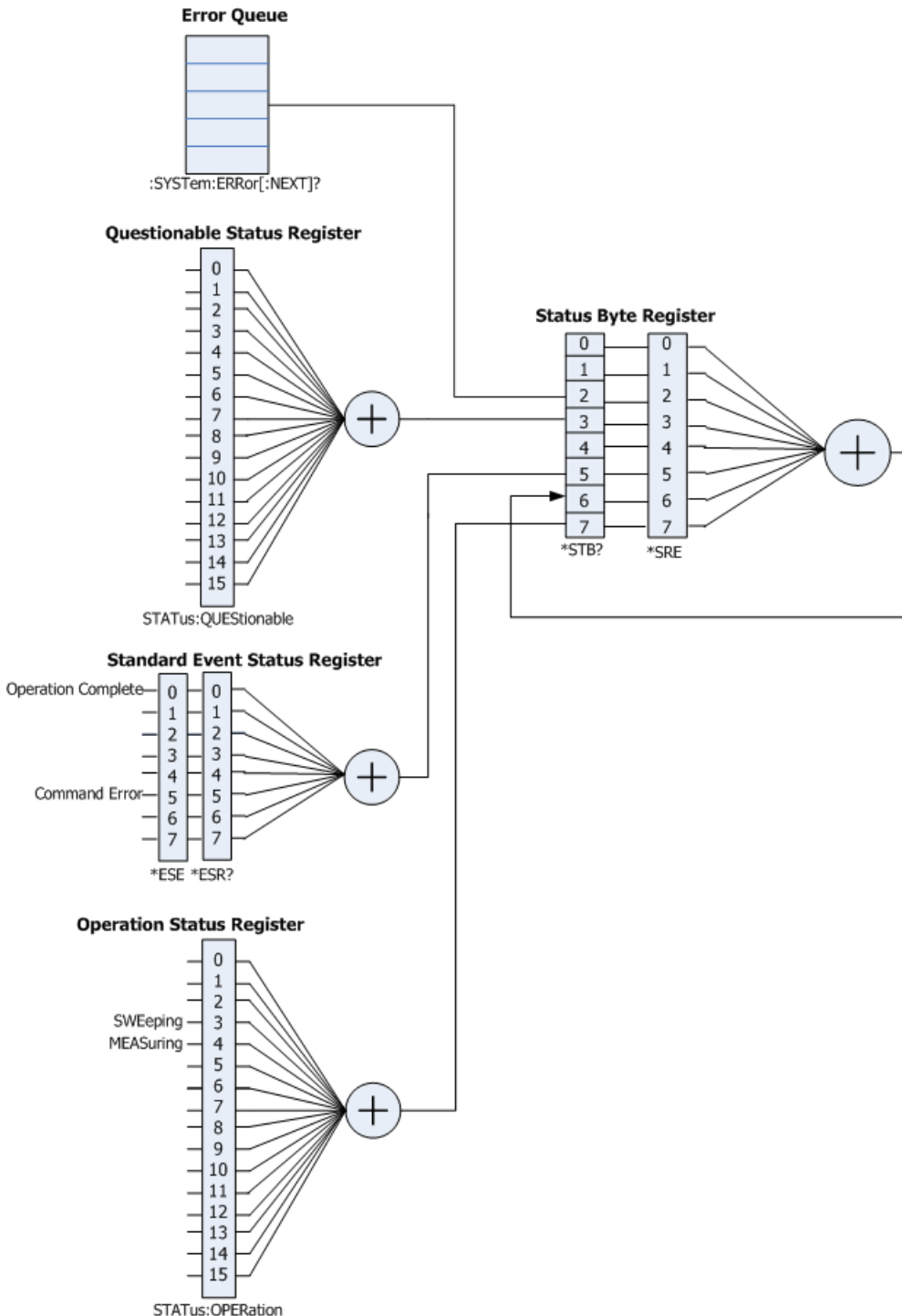
```
[[:SENSe]:TPOWer:RLIMit 0.01
```

下面的查询返回 1.000000E-02。

```
[[:SENSe]:TPOWer:RLIMit?
```

## :STATus命令子系统

:STATus 命令子系统与 [IEEE 488.2 公用命令](#) 主要用于操作或查询状态寄存器。状态寄存器的结构如下图所示，包括可疑状态寄存器（Questionable Status Register）、操作状态寄存器（Operation Status Register）、标准事件状态寄存器（Standard Event Status Register）和状态字节寄存器（Status Byte Register）以及错误队列（Error Queue）。STATus 子系统的命令用于设置、查询可疑状态寄存器和操作状态寄存器；IEEE488.2 公用命令可对标准事件状态寄存器、状态字节寄存器进行相关的操作；[:SYSTem:ERRor\[:NEXT\]?](#) 命令用于查询并删除错误队列中的错误信息。



## 命令列表:

- ◆ [:STATus:OPERation:CONDition?](#)
- ◆ [:STATus:OPERation:ENABLE](#)
- ◆ [:STATus:OPERation\[:EVENT\]?](#)
- ◆ [:STATus:PRESet](#)
- ◆ [:STATus:QUEStionable:CONDition?](#)
- ◆ [:STATus:QUEStionable:ENABLE](#)
- ◆ [:STATus:QUEStionable\[:EVENT\]?](#)

**:STATus:OPERation:CONDition?**

## 命令格式

:STATus:OPERation:CONDition?

## 功能描述

查询操作状态寄存器的条件寄存器值。

## 返回格式

查询以整数形式返回条件寄存器值。例如：24。

**:STATus:OPERation:ENABLE**

## 命令格式

:STATus:OPERation:ENABLE &lt;integer&gt;

:STATus:OPERation:ENABLE?

## 功能描述

设置操作状态寄存器的使能寄存器值。

查询操作状态寄存器的使能寄存器值。

## 参数

名称	类型	范围	默认值
<integer>	整型	参考“说明”	0

## 说明

操作状态寄存器的定义如下表所示，其中，位 0 至位 2、位 5 至位 7、位 13 和位 14 为保留位，可以设置数值但不影响系统，位 15、位 12 至位 8 未使用，始终视为 0，因此<integer>的取值范围为 0000000000000000（十进制 0）和 1111111111111111（十进制 32767）之间位 15、位 12 至位 8 为 0 的二进制数对应的十进制数。

位	值	定义
0	1	Reserved
1	2	Reserved
2	4	Reserved
3	8	SWEeping
4	16	MEASuring
5	32	Reserved
6	64	Reserved

7	128	Reserved
8	0	Not Used
9	0	Not Used
10	0	Not Used
11	0	Not Used
12	0	Not Used
13	8192	Reserved
14	16384	Reserved
15	0	Not Used

**返回格式**

查询以整数形式返回操作状态寄存器中使能寄存器的值。

**举例**

下面的命令设置操作状态寄存器的使能寄存器值为 100。

```
:STATus:OPERation:ENABle 100
```

下面的查询返回 100。

```
:STATus:OPERation:ENABle?
```

**:STATus:OPERation[:EVENT]?****命令格式**

```
:STATus:OPERation[:EVENT]?
```

**功能描述**

查询操作状态寄存器的事件寄存器值。

**返回格式**

查询以整数形式返回事件寄存器值。例如：24。

**:STATus:PRESet****命令格式**

```
:STATus:PRESet
```

**功能描述**

将操作状态寄存器和可疑状态寄存器的使能寄存器值清零。

**:STATus:QUEStionable:CONDition?****命令格式**

```
:STATus:QUEStionable:CONDition?
```

**功能描述**

查询可疑状态寄存器的条件寄存器值。

**返回格式**

查询以整数形式返回可疑状态寄存器 的条件寄存器值。例如：0。



## :STATus:QUEStionable:ENABle

### 命令格式

```
:STATus:QUEStionable:ENABle <integer>
:STATus:QUEStionable:ENABle?
```

### 功能描述

设置可疑状态寄存器的使能寄存器值。  
 查询可疑状态寄存器的使能寄存器值。

### 参数

名称	类型	范围	默认值
<integer>	整型	参考“说明”	0

### 说明

可疑状态寄存器的定义如下表所示，其中，位 0 至位 8、位 13 和位 14 为保留位，可以设置数值但不影响系统，位 15、位 12 至位 9 未使用，始终视为 0，因此<integer>的取值范围为 0000000000000000（十进制 0）和 1111111111111111（十进制 32767）之间位 15、位 12 至位 9 为 0 的二进制数对应的十进制数。

位	值	定义
0	1	Reserved
1	2	Reserved
2	4	Reserved
3	8	Reserved
4	16	Reserved
5	32	Reserved
6	64	Reserved
7	128	Reserved
8	256	Reserved
9	0	Not Used
10	0	Not Used
11	0	Not Used
12	0	Not Used
13	8192	Reserved
14	16384	Reserved
15	0	Not Used

### 返回格式

查询以整数形式返回可疑状态寄存器的使能寄存器值。

## :STATus:QUEStionable[:EVENT]?

### 命令格式

```
:STATus:QUEStionable[:EVENT]?
```

### 功能描述

查询可疑状态寄存器的事件寄存器值。

### 返回格式

查询以整数形式返回可疑状态寄存器 的事件寄存器值。例如：0。

## :SYSTem命令子系统

命令列表:

- ◆ [:SYSTem:BEEPer:STATe](#)
- ◆ [:SYSTem:CLear](#)
- ◆ [:SYSTem:COMMunicate:APORt](#)
- ◆ [:SYSTem:COMMunicate:BRMT](#)
- ◆ [:SYSTem:COMMunicate:GPIB\[:SELF\]:ADDRess](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:AUToip:STATe](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:DHCP:STATe](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:IP:ADDRess](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:IP:DNSServer](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:IP:GATeway](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:IP:SUBMask](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:MANuip:STATe](#)
- ◆ [:SYSTem:COMMunicate:LAN\[:SELF\]:RESet](#)
- ◆ [:SYSTem:COMMunicate:USB\[:SELF\]:ADDRess?](#)
- ◆ [:SYSTem:COMMunicate:USB\[:SELF\]:CLASs](#)
- ◆ [:SYSTem:CONFigure:INFormation?](#)
- ◆ [:SYSTem:CONFigure:MESSAge?](#)
- ◆ [:SYSTem:DATE](#)
- ◆ [:SYSTem:ERRor\[:NEXT\]?](#)
- ◆ [:SYSTem:FSWItch\[:STATe\]](#)
- ◆ [:SYSTem:KLOCK](#)
- ◆ [:SYSTem:LANGuage](#)
- ◆ [:SYSTem:LINemod:STATe?](#)
- ◆ [:SYSTem:LINemod:TYPe](#)
- ◆ [:SYSTem:LKEY](#)
- ◆ [:SYSTem:OPTions?](#)
- ◆ [:SYSTem:PON:TYPE](#)
- ◆ [:SYSTem:PRESet](#)
- ◆ [:SYSTem:PRESet:SAVE](#)
- ◆ [:SYSTem:PRESet:TYPE](#)
- ◆ [:SYSTem:SPEaker\[:STATe\]](#)
- ◆ [:SYSTem:SPEaker:VOLume](#)
- ◆ [:SYSTem:TIME](#)
- ◆ [:SYSTem:TX:STATe?\\*](#)
- ◆ [:SYSTem:TX:SWset\\*\\*](#)
- ◆ [:SYSTem:TX:SWSTa?\\*](#)

- ◆ [:SYSTem:USERkey:CONFirm](#)
- ◆ [:SYSTem:USERkey:KEYCmd](#)
- ◆ [:SYSTem:USERkey:STATe](#)
- ◆ [:SYSTem:VERSiOn?](#)

**说明:**

带\*\*的命令仅适用于已安装射频演示套件选件的 DSA700。

**:SYSTem:BEEPer:STATe****命令格式**

```
:SYSTem:BEEPer:STATe OFF|ON|0|1
:SYSTem:BEEPer:STATe?
```

**功能描述**

打开或关闭通过/失败测试时的蜂鸣器开关。  
查询蜂鸣器的状态。

**参数**

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开蜂鸣器开关。

```
:SYSTem:BEEPer:STATe ON 或 :SYSTem:BEEPer:STATe 1
```

下面的查询返回 1。

```
:SYSTem:BEEPer:STATe?
```

**:SYSTem:CLEar****命令格式**

```
:SYSTem:CLEar
```

**功能描述**

安全清除用户设置的所有数据。

**说明**

清除数据后，恢复出厂状态，包括：

- NVRAM、NorFlash 中保存的用户数据等恢复出厂设置；
- LXI 中的 HOST NAME，IP 地址，密码等恢复出厂设置。

## :SYSTem:COMMunicate:APORt

### 命令格式

```
:SYSTem:COMMunicate:APORt GPIB|LAN|USB|OFF
:SYSTem:COMMunicate:APORt?
```

### 功能描述

选择当前通讯端口或关闭所有通讯端口。  
查询当前所选的通讯端口。

### 参数

名称	类型	范围	默认值
——	关键字	GPIB LAN USB OFF	OFF

### 返回格式

查询返回 GPIB、LAN、USB 或 OFF。

### 举例

下面的命令选择通讯端口为 LAN。

```
:SYSTem:COMMunicate:APORt LAN
```

下面的查询返回 LAN。

```
:SYSTem:COMMunicate:APORt?
```

## :SYSTem:COMMunicate:BRMT

### 命令格式

```
:SYSTem:COMMunicate:BRMT OFF|ON|0|1
:SYSTem:COMMunicate:BRMT?
```

### 功能描述

切换仪器状态至本地模式或远程模式。  
查询仪器当前的工作状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	——

### 说明

OFF|0: 表示切换到本地模式。

ON|1: 表示切换到远程模式。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令将仪器切换至本地模式。

```
:SYSTem:COMMunicate:BRMT OFF 或 :SYSTem:COMMunicate:BRMT 0
```

下面的查询返回 0。

```
:SYSTem:COMMunicate:BRMT?
```

**:SYSTem:COMMunicate:GPIB[:SELF]:ADDRess****命令格式**

```
:SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <integer>
:SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?
```

**功能描述**

设置 GPIB 地址。  
查询 GPIB 地址。

**参数**

名称	类型	范围	默认值
<integer>	整型	0 至 30	18

**返回格式**

查询以整数形式返回 GPIB 地址。

**举例**

下面的命令设置 GPIB 地址为 16。

```
:SYSTem:COMMunicate:GPIB:SELF:ADDRess 16
```

下面的查询返回 16。

```
:SYSTem:COMMunicate:GPIB:SELF:ADDRess?
```

**:SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe****命令格式**

```
:SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe OFF|ON|0|1
:SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe?
```

**功能描述**

打开或关闭自动 IP 设置模式。  
查询自动 IP 设置模式的状态。

**参数**

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	ON 1

**说明**

频谱仪总是按 DHCP、自动 IP、手动 IP 的顺序尝试获取本机的 IP 地址配置，并且三者不能同时关闭。

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令关闭自动 IP 设置模式。

```
:SYSTem:COMMunicate:LAN:SELF:AUTOip:STATe OFF
或 :SYSTem:COMMunicate:LAN:SELF:AUTOip:STATe 0
```

下面的查询返回 0。

```
:SYSTem:COMMunicate:LAN:SELF:AUTOip:STATe?
```

## :SYSTem:COMMunicate:LAN[:SELf]:DHCP:STATe

### 命令格式

```
:SYSTem:COMMunicate:LAN[:SELf]:DHCP:STATe OFF|ON|0|1
:SYSTem:COMMunicate:LAN[:SELf]:DHCP:STATe?
```

### 功能描述

打开或关闭 DHCP 模式。  
查询 DHCP 模式的状态。

### 参数

名称	类型	范围	默认值
---	布尔型	OFF ON 0 1	ON 1

### 说明

频谱仪总是按 DHCP、自动 IP、手动 IP 的顺序尝试获取本机的 IP 地址配置，并且三者不能同时关闭。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令关闭 DHCP 模式。

```
:SYSTem:COMMunicate:LAN:SELf:DHCP:STATe OFF 或 :SYSTem:COMMunicate:LAN:SELf:DHCP:STATe 0
```

下面的查询返回 0。

```
:SYSTem:COMMunicate:LAN:SELf:DHCP:STATe?
```

## :SYSTem:COMMunicate:LAN[:SELf]:IP:ADDress

### 命令格式

```
:SYSTem:COMMunicate:LAN[:SELf]:IP:ADDress <ip_address>
:SYSTem:COMMunicate:LAN[:SELf]:IP:ADDress?
```

### 功能描述

设置 IP 地址。  
查询 IP 地址。

### 参数

名称	类型	范围	默认值
<ip_address>	ASCII 字符串	参考“说明”	---

### 说明

<ip\_address>的格式为 nnn.nnn.nnn.nnn，第一个 nnn 的范围为 1 至 223（127 除外），其他三个 nnn 的范围为 0 至 255。

### 返回格式

查询返回当前 IP 地址，格式为 nnn.nnn.nnn.nnn。

### 举例

下面的命令设置 IP 地址为 172.16.3.199。

```
:SYSTem:COMMunicate:LAN:SELf:IP:ADDress 172.16.3.199
```

下面的查询返回 172.16.3.199。

```
:SYSTem:COMMunicate:LAN:SELf:IP:ADDress?
```

## :SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer

### 命令格式

```
:SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer <ip_address>
:SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer?
```

### 功能描述

设置 DNS 地址。  
查询 DNS 地址。

### 参数

名称	类型	范围	默认值
<ip_address>	ASCII 字符串	参考“说明”	——

### 说明

<ip\_address>的格式为 nnn.nnn.nnn.nnn，第一个 nnn 的范围为 1 至 223（127 除外），其他三个 nnn 的范围为 0 至 255。

### 返回格式

查询返回当前 DNS 地址，格式为 nnn.nnn.nnn.nnn。

### 举例

下面的命令设置 DNS 地址为 172.16.2.2。

```
:SYSTem:COMMunicate:LAN:SELF:IP:DNSServer 172.16.2.2
```

下面的查询返回 172.16.2.2。

```
:SYSTem:COMMunicate:LAN:SELF:IP:DNSServer?
```

## :SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway

### 命令格式

```
:SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway <ip_address>
:SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway?
```

### 功能描述

设置默认网关。  
查询默认网关。

### 参数

名称	类型	范围	默认值
<ip_address>	ASCII 字符串	参考“说明”	——

### 说明

<ip\_address>的格式为 nnn.nnn.nnn.nnn，第一个 nnn 的范围为 1 至 223（127 除外），其他三个 nnn 的范围为 0 至 255。

### 返回格式

查询返回当前默认网关，格式为 nnn.nnn.nnn.nnn。

**举例**

下面的命令设置默认网关为 172.16.3.1。

```
:SYSTem:COMMunicate:LAN:SELF:IP:GATeway 172.16.3.1
```

下面的查询返回 172.16.3.1。

```
:SYSTem:COMMunicate:LAN:SELF:IP:GATeway?
```

**:SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask****命令格式**

```
:SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask<ip_address>
```

```
:SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask?
```

**功能描述**

设置子网掩码。

查询子网掩码。

**参数**

名称	类型	范围	默认值
<ip_address>	ASCII 字符串	参考“说明”	—

**说明**

<ip\_address>的格式为 nnn.nnn.nnn.nnn，其中 nnn 的范围为 0 至 255。

**返回格式**

返回当前的子网掩码，格式为 nnn.nnn.nnn.nnn。

**举例**

下面的命令设置子网掩码为 255.255.255.0。

```
:SYSTem:COMMunicate:LAN:SELF:IP:SUBMask 255.255.255.0
```

下面的查询返回 255.255.255.0。

```
:SYSTem:COMMunicate:LAN:SELF:IP:SUBMask?
```



## :SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe

### 命令格式

```
:SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe OFF|ON|0|1
:SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe?
```

### 功能描述

打开或关闭手动 IP 设置模式。  
查询手动 IP 设置模式的状态。

### 参数

名称	类型	范围	默认值
---	布尔型	OFF ON 0 1	OFF 0

### 说明

频谱仪总是按 DHCP、自动 IP、手动 IP 的顺序尝试获取本机的 IP 地址配置，并且三者不能同时关闭。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开手动 IP 设置模式。

```
:SYSTem:COMMunicate:LAN:SELF:MANuip:STATe ON
或 :SYSTem:COMMunicate:LAN:SELF:MANuip:STATe 1
```

下面的查询返回 1。

```
:SYSTem:COMMunicate:LAN:SELF:MANuip:STATe?
```

## :SYSTem:COMMunicate:LAN[:SELF]:RESet

### 命令格式

```
:SYSTem:COMMunicate:LAN[:SELF]:RESet
```

### 功能描述

复位 LAN 的设置：打开 DHCP 和自动 IP，关闭手动 IP。

## :SYSTem:COMMunicate:USB[:SELF]:ADDRess?

### 命令格式

```
:SYSTem:COMMunicate:USB[:SELF]:ADDRess?
```

### 功能描述

查询 USB 设备地址。

### 返回格式

查询返回 USB 设备地址，如 1。

## :SYSTem:COMMunicate:USB[:SELF]:CLASs

### 命令格式

```
:SYSTem:COMMunicate:USB[:SELF]:CLASs TMC|PRINter|AUTO
:SYSTem:COMMunicate:USB[:SELF]:CLASs?
```

### 功能描述

设置 USB 设备类型。  
查询 USB 设备类型。

### 参数

名称	类型	范围	默认值
—	关键字	TMC PRINter AUTO	TMC

### 说明

选择 AUTO 时，具体设备类型由 USB 主设备确定。  
选择 TMC 时，将频谱仪作为测试及测量设备类（Test & Measurement Class）使用。  
选择 PRINter 时，将频谱仪作为打印机设备类（Printer Class）使用。此时，仪器与 PC 的连接被断开。若要继续远程控制仪器，您需重新连接。

### 返回格式

查询返回 TMC 或 PRIN。

### 举例

下面的命令设置 USB 设备类型为打印机。

```
:SYSTem:COMMunicate:USB:SELF:CLASs PRINter
```

下面的查询返回 PRIN。

```
:SYSTem:COMMunicate:USB:SELF:CLASs?
```

## :SYSTem:CONFIgure:INFORMATION?

### 命令格式

```
:SYSTem:CONFIgure:INFORMATION?
```

### 功能描述

查询频谱仪的系统信息。

### 说明

系统信息包括：型号、序列号、软硬件版本号。

### 返回格式

查询返回系统信息（语言类型取决于当前系统的语言），如：

型号:DSA710

序列号:DSA8A134400008

主板版本号:00.07

射频 FPGA 版本号:00.05

数字 FPGA 版本号:00.05

嵌入软件版本号:00.01.16

Boot 版本号:00.01.04

## :SYSTem:CONFigure:MESSage?

### 命令格式

:SYSTem:CONFigure:MESSage?

### 功能描述

查询最近出现的系统消息。

### 说明

最多可显示 71 条历史消息。

### 返回格式

查询返回最近出现的系统消息（语言类型取决于当前系统的语言），如：

470|校准数据丢失。|2016-05-16 09:38:36

10|U 盘连接。|2016-05-16 09:49:48

## :SYSTem:DATE

### 命令格式

:SYSTem:DATE <year>,<month>,<day>

:SYSTem:DATE?

### 功能描述

设置仪器的日期。

查询仪器的日期。

### 参数

名称	类型	范围	默认值
<year>	ASCII 字符串	2000 至 2099	——
<month>	ASCII 字符串	01 至 12	——
<day>	ASCII 字符串	01 至 31	——

### 返回格式

查询以“YYYY,MM,DD”格式返回当前的日期。

### 举例

下面的命令设置仪器的日期为 2016 年 5 月 16 日。

```
:SYSTem:DATE 2016,05,16
```

下面的查询返回 2016,05,16。

```
:SYSTem:DATE?
```

## :SYSTem:ERRor[:NEXT]?

### 命令格式

:SYSTem:ERRor[:NEXT]?

### 功能描述

查询最新进入错误队列的信息并将其删除。

### 说明

如果没有错误，则返回：0,"No error"。

使用 [\\*CLS](#) 命令可以清除错误队列中的所有信息。

### 返回格式

查询以“错误编号,"错误信息"”格式返回错误信息。例如：-110,"Command header error"。

## :SYSTem:FSWItch[:STATe]

### 命令格式

:SYSTem:FSWItch[:STATe] OFF|ON|0|1  
:SYSTem:FSWItch[:STATe]?

### 功能描述

设置前面板电源开关是否有效。

查询前面板电源开关是否有效。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	ON 1

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令设置前面板电源开关无效。

```
:SYSTem:FSWItch:STATe OFF 或 :SYSTem:FSWItch:STATe 0
```

下面的查询返回 0。

```
:SYSTem:FSWItch:STATe?
```

## :SYSTem:KLOCK

### 命令格式

```
:SYSTem:KLOCK OFF|ON|0|1,<key>
:SYSTem:KLOCK? <key>
```

### 功能描述

锁定或解锁指定的功能按键。  
查询指定的按键是否被锁定。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0
<key>	关键字	参考“说明”	——

### 说明

参数<key>用于指定按键，取值范围如下：

FREQ SPAN AMP	/*FREQ, SPAN, AMPT 键*/
BW SWEEP TRACE	/*BW/Det, Sweep/Trig, Trace/P/F 键*/
MARK MARKFUNC MARKTO PEAK	/*Marker, Marker Fctn, Marker->, Peak 键*/
TUNE	/*Auto 键*/
MEAS MEASSET DEMOD	/*Meas, Meas Setup, Demod 键*/
SYSTEM PRINTSETUP STORAGE	/*System, Print Setup, Storage 键*/
PRESET PRINT	/*Preset, 打印键*/

ON|1：表示锁定。

OFF|0：表示解锁。

同时锁定或解锁多个按键时，按键参数之间请用“，”分隔。

锁定前面板全部按键（不包含电源开关键），请用:SYSTem:KLOCK ON|1,ALL 命令；解锁前面板全部按键，请用:SYSTem:KLOCK OFF|0,ALL 命令。

### 返回格式

查询返回“按键名称 0”或“按键名称 1”。

### 举例

下面的命令锁定 FREQ 按键。

```
:SYSTem:KLOCK ON,FREQ 或 :SYSTem:KLOCK 1,FREQ
```

下面的查询返回 FREQ 1。

```
:SYSTem:KLOCK? FREQ
```

## :SYSTem:LANGuage

### 命令格式

```
:SYSTem:LANGuage ENGLish|CHINese|JAPan|PORTugese|GERMan|POLish|KORea|TCHinese
:SYSTem:LANGuage?
```

### 功能描述

设置仪器的语言类型。  
查询仪器的语言类型。

### 参数

名称	类型	范围	默认值
---	关键字	ENGLish CHINese JAPan PORTugese GERMan POLish KORea TCHinese	ENGLish

### 返回格式

查询返回 ENGL、CHIN、JAP、PORT、GERM、POL、KOR 或 TCHinese。

### 举例

下面的命令设置语言为中文。

```
:SYSTem:LANGuage CHINese
```

下面的查询返回 CHIN。

```
:SYSTem:LANGuage?
```

## :SYSTem:LINemod:STATe?

### 命令格式

```
:SYSTem:LINemod:STATe?
```

### 功能描述

查询产线模式的状态。

### 返回格式

查询返回 0（非产线模式）或 1（产线模式）。

### 举例

下面的查询返回 1。

```
:SYSTem:LINemod:STATe?
```

## :SYSTem:LINemod:TYPe

### 命令格式

```
:SYSTem:LINemod:TYPe FACTory|USER1|USER2|USER3|USER4|USER5|USER6|OFF
```

### 功能描述

设置产线模式加载的预设置或退出产线模式。

### 参数

名称	类型	范围	默认值
——	关键字	FACTory USER1 USER2 USER3 USER4 USER5 USER6 OFF	OFF

### 举例

下面的命令设置产线模式加载的预设置为用户设置 4。

```
:SYSTem:LINemod:TYPe USER4
```

## :SYSTem:LKEY

### 命令格式

```
:SYSTem:LKEY <license key>
```

```
:SYSTem:LKEY? <option>
```

### 功能描述

安装并激活指定的选件。

查询指定选件的序列号。

### 参数

名称	类型	范围	默认值
<license key>	ASCII 字符串	——	——
<option>	整型	1 至 2	——

### 说明

参数<option>表示选件编号。选件编号与选件订货号的对应关系如下所示：

选件编号	选件订货号
1	AMK-DSA800
2	EMI-DSA800

### 返回格式

查询返回序列号。

### 举例

下面的命令安装序列号为 UADEYSACMA6RJACQTAWJZLLPBCVA 的选件（对应的选件编号为 1）。

```
:SYSTem:LKEY UADEYSACMA6RJACQTAWJZLLPBCVA
```

下面的查询返回 UADEYSACMA6RJACQTAWJZLLPBCVA。

```
:SYSTem:LKEY? 1
```

## :SYSTem:OPTions?

### 命令格式

:SYSTem:OPTions?

### 功能描述

查询频谱仪的选件状态，包括选件编号、订货号、类型和激活状态。

### 返回格式

查询返回频谱仪的选件状态，包括选件编号、订货号、类型和激活状态，如：

```
Serial NO.|Option |Option Type |Active
1| AMK-DSA800|正式版|Y
2| EMI-DSA800|未安装|N
```

## :SYSTem:PON:TYPE

### 命令格式

:SYSTem:PON:TYPE PRESet|LAST  
:SYSTem:PON:TYPE?

### 功能描述

设置开机后系统调用的是上次设置还是预置设置。  
查询开机后系统调用的设置类型。

### 参数

名称	类型	范围	默认值
——	关键字	PRESet LAST	PRESet

### 说明

PRESet：预置设置，包括出厂设置和六种用户设置，使用[:SYSTem:PRESet:TYPE](#) 命令选择。

LAST：上次设置。

### 返回格式

查询返回 PRES 或 LAST。

### 举例

下面的命令设置调用上次设置。

```
:SYSTem:PON:TYPE LAST
```

下面的查询返回 LAST。

```
:SYSTem:PON:TYPE?
```



## :SYSTem:PRESet

### 命令格式

```
:SYSTem:PRESet
```

### 功能描述

调用系统的预置设置，将系统恢复到[:SYSTem:PRESet:TYPE](#) 命令指定的状态。

## :SYSTem:PRESet:SAVE

### 命令格式

```
:SYSTem:PRESet:SAVE <type>,<name>
```

### 功能描述

保存用户的设置。

### 参数

名称	类型	范围	默认值
<type>	关键字	USER1 USER2 USER3 USER4 USER5 USER6	——
<name>	ASCII 字符串	需要保存的文件名（不包括路径和后缀）	——

### 说明

该命令仅当[:SYSTem:PRESet:TYPE](#) 命令选择 USER1、USER2、USER3、USER4、USER5 或 USER6 时可用。

### 举例

下面的命令将当前仪器设置以 def2.sta 为文件名保存到 USER2。

```
:SYSTem:PRESet:SAVE USER2,def2
```

## :SYSTem:PRESet:TYPE

### 命令格式

```
:SYSTem:PRESet:TYPe FACTory|USER1|USER2|USER3|USER4|USER5|USER6  
:SYSTem:PRESet:TYPe?
```

### 功能描述

选择系统的预置类型为出厂设置或用户设置 1 至用户设置 6 中的一种。

查询系统的预置类型。

### 参数

名称	类型	范围	默认值
——	关键字	FACTory USER1 USER2 USER3 USER4 USER5 USER6	FACTory

### 返回格式

查询返回 FACT、USER1、USER2、USER3、USER4、USER5 或 USER6。

### 举例

下面的命令设置系统的预置类型为用户设置 5。

```
:SYSTem:PRESet:TYPe USER5
```

下面的查询返回 USER5。

```
:SYSTem:PRESet:TYPe?
```

## :SYSTem:SPEaker[:STATe]

### 命令格式

```
:SYSTem:SPEaker[:STATe] OFF|ON|0|1
:SYSTem:SPEaker[:STATe]?
```

### 功能描述

进行解调设置时，打开或关闭耳机。  
查询解调设置中的耳机状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 说明

该命令仅在打开 AM 或 FM 解调功能时有效。

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开耳机。

```
:SYSTem:SPEaker:STATe ON 或 :SYSTem:SPEaker:STATe 1
```

下面的查询返回 1。

```
:SYSTem:SPEaker:STATe?
```

## :SYSTem:SPEaker:VOLume

### 命令格式

```
:SYSTem:SPEaker:VOLume <integer>
:SYSTem:SPEaker:VOLume?
```

### 功能描述

进行解调设置时，设置耳机的音量。  
查询解调设置时耳机的音量。

### 参数

名称	类型	范围	默认值
<integer>	整型	0 至 255	100

### 说明

该命令仅在打开 AM 或 FM 解调功能时有效。

### 返回格式

查询以整数形式返回耳机音量。

### 举例

下面的命令设置耳机音量为 100。

```
:SYSTem:SPEaker:VOLume 100
```

下面的查询返回 100。

```
:SYSTem:SPEaker:VOLume?
```

## :SYSTem:TIME

### 命令格式

:SYSTem:TIME <hour>,<minute>,<second>  
:SYSTem:TIME?

### 功能描述

设置仪器的时间。  
查询仪器的时间。

### 参数

名称	类型	范围	默认值
<hour>	ASCII 字符串	00 至 23	——
<minute>	ASCII 字符串	00 至 59	——
<second>	ASCII 字符串	00 至 59	——

### 返回格式

查询以“HH,MM,SS”格式返回当前的时间。

### 举例

下面的命令设置时间为 15 时 10 分 30 秒。

```
:SYSTem:TIME 15,10,30
```

下面的查询返回 15,10,30。

```
:SYSTem:TIME?
```

## :SYSTem:TX:STATe?

### 命令格式

:SYSTem:TX:STATe?

### 功能描述

查询 TX1000 的连接状态。

### 返回格式

查询返回 0 或 1。

## :SYSTem:TX:SWset

### 命令格式

:SYSTem:TX:SWset <SW1|SW2|SW3|SW4|SW5>,OFF|ON|0|1

### 功能描述

设置 TX1000 的开关状态。

### 参数

名称	类型	范围	默认值
——	关键字	SW1 SW2 SW3 SW4 SW5	——
——	布尔型	OFF ON 0 1	OFF 0

### 说明

SW1 至 SW5 分别代表 TX1000 的 5 个开关。  
该命令仅当已连接 TX1000 时有效。

### 举例

下面的命令设置开关 1 为开。

:SYSTem:TX:SWset SW1,ON 或 :SYSTem:TX:SWset SW1,1

## :SYSTem:TX:SWSTa?

### 命令格式

:SYSTem:TX:SWSTa? <SW1|SW2|SW3|SW4|SW5>

### 功能描述

查询 TX1000 的开关状态。

### 参数

名称	类型	范围	默认值
——	关键字	SW1 SW2 SW3 SW4 SW5	——

### 说明

SW1 至 SW5 分别代表 TX1000 的 5 个开关。  
该命令仅当频谱仪连接 TX1000 选件之后有效。

### 返回格式

查询返回 0 或 1。

## :SYSTem:USERkey:CONFirm

### 命令格式

```
:SYSTem:USERkey:CONFirm
```

### 功能描述

在定义 UserKey 按键关联功能时，用于确认以使定义生效。

### 说明

只有在 UserKey 设置为打开时（参考:[:SYSTem:USERkey:STATe](#) 命令）发送定义命令（参考:[:SYSTem:USERkey:KEYCmd](#) 命令），然后再发送该命令，UserKey 按键的定义才会生效。

## :SYSTem:USERkey:KEYCmd

### 命令格式

```
:SYSTem:USERkey:KEYCmd <key_value>{,<key_value>}
:SYSTem:USERkey:KEYCmd?
```

### 功能描述

为 UserKey 按键定义一个关联功能。  
查询当前 UserKey 按键关联的功能。

### 参数

名称	类型	范围	默认值
<key_value>	关键字	FREQ SPAN AMP BW SWEep TUNE DEMod  TRACe MEAS MEASset MARK MARKfunc  MARKto PEAK PRESet SYSTem STORage PRINt PRINtsetup  HELP F1 F2 F3 F4 F5 F6 F7 RETUrn PAGEdown	SYSTem

### 说明

如下参数对应前面板按键：

```
FREQ|SPAN|AMP|BW|SWEep|TUNE|DEMod|TRACe|MEAS|MEASset|MARK|MARKfunc|MARKto|PEAK|PRESet|SYSTem|STORage|PRINt|PRINtsetup|HELP
```

如下参数对应前面板菜单软键和菜单控制键：

```
F1|F2|F3|F4|F5|F6|F7|RETUrn|PAGEdown
```

只有在 UserKey 设置为打开时（参考:[:SYSTem:USERkey:STATe](#) 命令）发送该命令，然后再发送:[:SYSTem:USERkey:CONFirm](#) 命令，UserKey 按键的定义才会生效。

### 举例

下面的命令定义 UserKey 按键的关联功能为 System→自检→键盘测试。查询命令的返回值为：System,自检,键盘测试，

```
:SYSTem:USERkey:STATe ON
:SYSTem:USERkey:KEYCmd SYSTem,PAGEdown,F2,F2
:SYSTem:USERkey:CONFirm
:SYSTem:USERkey:KEYCmd?
```

## :SYSTem:USERkey:STATe

### 命令格式

```
:SYSTem:USERkey:STATe OFF|ON|0|1  
:SYSTem:USERkey:STATe?
```

### 功能描述

打开或关闭 UserKey 设置。  
查询 UserKey 设置的状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 举例

下面的命令打开 UserKey 设置。

```
:SYSTem:USERkey:STATe ON 或 :SYSTem:USERkey:STATe 1
```

下面的查询返回 1。

```
:SYSTem:USERkey:STATe?
```

## :SYSTem:VERSion?

### 命令格式

```
:SYSTem:VERSion?
```

### 功能描述

查询仪器遵循的 SCPI 版本号。

### 返回格式

返回 SCPI 版本号。例如：1999.0。

## :TRACe命令子系统

命令列表:

- ◆ [:TRACe:AVERage:CLEar](#)
- ◆ [:TRACe:AVERage:COUNT](#)
- ◆ [:TRACe:AVERage:COUNT:CURRent?](#)
- ◆ [:TRACe:AVERage:RESet](#)
- ◆ [:TRACe:CLEar:ALL](#)
- ◆ [:TRACe\[:DATA\]](#)
- ◆ [:TRACe:MATH:A](#)
- ◆ [:TRACe:MATH:B](#)
- ◆ [:TRACe:MATH:CONSt](#)
- ◆ [:TRACe:MATH:PEAK\[:DATA\]?](#)
- ◆ [:TRACe:MATH:PEAK:POINts?](#)
- ◆ [:TRACe:MATH:PEAK:SORT](#)
- ◆ [:TRACe:MATH:PEAK:TABLE:STATe](#)
- ◆ [:TRACe:MATH:PEAK:THReshold](#)
- ◆ [:TRACe:MATH:STATe](#)
- ◆ [:TRACe:MATH:TYPE](#)
- ◆ [:TRACe<n>:AVERage:TYPE](#)
- ◆ [:TRACe<n>:MODE](#)

### :TRACe:AVERage:CLEar

命令格式

```
:TRACe:AVERage:CLEar
```

功能描述

清除当前已执行的迹线平均次数。

### :TRACe:AVERage:COUNT

命令格式

```
:TRACe:AVERage:COUNT <integer>
:TRACe:AVERage:COUNT?
```

功能描述

设置迹线平均次数。  
查询迹线平均次数。

参数

名称	类型	范围	默认值
<integer>	整型	1 至 1000	100

**返回格式**

查询以整数形式返回迹线平均次数。

**举例**

下面的命令设置平均次数为 200。

```
:TRACe:AVERAge:COUNT 200
```

下面的查询返回 200。

```
:TRACe:AVERAge:COUNT?
```

**:TRACe:AVERAge:COUNT:CURRent?****命令格式**

```
:TRACe:AVERAge:COUNT:CURRent?
```

**功能描述**

查询当前已执行的迹线平均次数。

**返回格式**

查询以整数形式返回当前已执行的迹线平均次数。

**:TRACe:AVERAge:RESet****命令格式**

```
:TRACe:AVERAge:RESet
```

**功能描述**

执行迹线平均复位操作。

**说明**

当迹线类型选择“视频平均”或“功率平均”时，该命令有效。

**:TRACe:CLEAr:ALL****命令格式**

```
:TRACe:CLEAr:ALL
```

**功能描述**

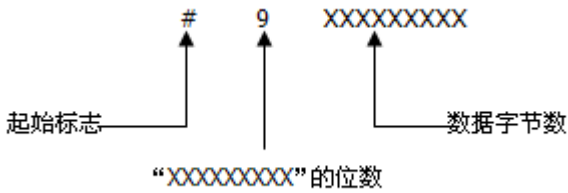
清除所有迹线，即将所有迹线类型都设置为 BLANK（关闭）。





注:

[1] 数据块格式为“数据块报头+数据块”，其中，数据块报头具有如下的格式:



#后面的第一位数字(9)表示数据块报头中剩下的数字位数;数据块报头中剩下的数字表示本次传输的数据字节数(不足9位在前面补0),例如:发送100字节的数据块报头为#9000000100。

[2] 本页面两个例子均发送601个点,查询返回601个点,此处限于篇幅,后续数据省略。另外,上述两个例子的查询返回值均为迹线停止扫描情况下所得。

## :TRACe:MATH:A

### 命令格式

:TRACe:MATH:A T1|T2|T3

:TRACe:MATH:A?

### 功能描述

设置迹线数学运算式中 A 代表迹线 1、迹线 2 或迹线 3。

查询迹线数学 运算式中 A 代表哪一迹线。

### 参数

名称	类型	范围	默认值
—	关键字	T1 T2 T3	T1

### 返回格式

查询返回 T1、T2 或 T3。

### 举例

下面的命令设置 A 为迹线 2。

:TRACe:MATH:A T2

下面的查询返回 T2。

:TRACe:MATH:A?

## :TRACe:MATH:B

### 命令格式

:TRACe:MATH:B T1|T2|T3

:TRACe:MATH:B?

### 功能描述

设置迹线数学运算式中 B 代表迹线 1、迹线 2 或迹线 3。

查询迹线数学运算式中 B 代表哪一迹线。

### 参数

名称	类型	范围	默认值
—	关键字	T1 T2 T3	T2

**返回格式**

查询返回 T1、T2 或 T3。

**举例**

下面的命令设置 B 为迹线 1。

```
:TRACe:MATH:B T1
```

下面的查询返回 T1。

```
:TRACe:MATH:B?
```

**:TRACe:MATH:CONST****命令格式**

```
:TRACe:MATH:CONST <integer>
```

```
:TRACe:MATH:CONST?
```

**功能描述**

设置迹线数学运算式中的常数值。

查询迹线数学运算式中的常数值。

**参数**

名称	类型	范围	默认值
<integer>	连续实型	-300 dB 至 300 dB	0 dB

**返回格式**

查询以科学计数形式返回常数值。

**举例**

下面的命令设置常数值为 50 dB。

```
:TRACe:MATH:CONST 50
```

下面的查询返回 5.000000E+01。

```
:TRACe:MATH:CONST?
```

**:TRACe:MATH:PEAK[:DATA]?****命令格式**

```
:TRACe:MATH:PEAK[:DATA]?
```

**功能描述**

查询峰值表中峰值的频率 (Hz) 和幅度 (与当前 Y 轴单位一致)。

**说明**

该命令只查询迹线 1。

**返回格式**

查询以如下形式返回峰值表中峰值的频率和幅度。最多返回 10 对频率和幅度值。

例如: 43500000,-7.253288E+01,43950000,-7.169086E+01。

## :TRACe:MATH:PEAK:POINts?

### 命令格式

:TRACe:MATH:PEAK:POINts?

### 功能描述

查询峰值表中的峰值个数。

### 返回格式

查询返回 0 至 10 之间的整数。

## :TRACe:MATH:PEAK:SORt

### 命令格式

:TRACe:MATH:PEAK:SORt AMPLitude|FREQuency  
:TRACe:MATH:PEAK:SORt?

### 功能描述

设置峰值表中的排序方式。  
查询峰值表中的排序方式。

### 参数

名称	类型	范围	默认值
—	关键字	AMPLitude FREQuency	FREQuency

### 返回格式

查询返回 AMPL 或 FREQ。

### 举例

下面的命令设置峰值表中的排序方式为幅度。

```
:TRACe:MATH:PEAK:SORt AMPLitude
```

下面的查询返回 AMPL。

```
:TRACe:MATH:PEAK:SORt?
```

## :TRACe:MATH:PEAK:TABLE:STATe

### 命令格式

:TRACe:MATH:PEAK:TABLE:STATe OFF|ON|0|1  
:TRACe:MATH:PEAK:TABLE:STATe?

### 功能描述

打开或关闭峰值表。  
查询峰值表的状态。

### 参数

名称	类型	范围	默认值
—	布尔型	OFF ON 0 1	OFF 0

### 返回格式

查询返回 0 或 1。

### 举例

下面的命令打开峰值表。

```
:TRACe:MATH:PEAK:TABLE:STATe 1 或 :TRACe:MATH:PEAK:TABLE:STATe ON
```

下面的查询返回 1。

```
:TRACe:MATH:PEAK:TABLE:STATe?
```

## :TRACe:MATH:PEAK:THReshold

### 命令格式

```
:TRACe:MATH:PEAK:THReshold NORMal|DLMore|DLLess
```

```
:TRACe:MATH:PEAK:THReshold?
```

### 功能描述

设置峰值表中峰值的显示方式。

查询峰值表中峰值的显示方式。

### 参数

名称	类型	范围	默认值
——	关键字	NORMal DLMore DLLess	NORMal

### 说明

NORMal: 常规, 显示前十个符合搜索参数的峰值。

DLMore: > 显示线, 显示前十个符合搜索参数且幅度大于显示线的峰值。

DLLess: < 显示线, 显示前十个符合搜索参数且幅度小于显示线的峰值。

### 返回格式

查询返回 NORM、DLM 或 DLL。

### 举例

下面的命令设置峰值表中峰值的显示方式为常规。

```
:TRACe:MATH:PEAK:THReshold NORMal
```

下面的查询返回 NORM。

```
:TRACe:MATH:PEAK:THReshold?
```

## :TRACe:MATH:STATe

### 命令格式

```
:TRACe:MATH:STATe OFF|ON|0|1
```

```
:TRACe:MATH:STATe?
```

### 功能描述

打开或关闭迹线的数学运算。

查询迹线数学运算的状态。

### 参数

名称	类型	范围	默认值
——	布尔型	OFF ON 0 1	OFF 0

**返回格式**

查询返回 0 或 1。

**举例**

下面的命令打开迹线的数学运算。

```
:TRACe:MATH:STATe 1 或 :TRACe:MATH:STATe ON
```

下面的查询返回 1。

```
:TRACe:MATH:STATe?
```

**:TRACe:MATH:TYPE****命令格式**

```
:TRACe:MATH:TYPE A-B|A+CONST|A-CONST
:TRACe:MATH:TYPE?
```

**功能描述**

设置迹线的运算类型。

查询迹线的运算类型。

**参数**

名称	类型	范围	默认值
—	关键字	A-B A+CONST A-CONST	A-B

**返回格式**

查询返回 A-B、A+CONST 或 A-CONST。

**举例**

下面的命令设置迹线的运算类型为 A+常量。

```
:TRACe:MATH:TYPE A+CONST
```

下面的查询返回 A+CONST。

```
:TRACe:MATH:TYPE?
```

**:TRACe<n>:AVERAge:TYPE****命令格式**

```
:TRACe<n>:AVERAge:TYPE VIDEo|RMS
:TRACe<n>:AVERAge:TYPE?
```

**功能描述**

设置迹线的平均类型。

查询迹线的平均类型。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3	—
—	关键字	VIDeo RMS	VIDeo

**说明**

VIDeo: 视频平均。

RMS: 功率平均。

通过/失败功能打开时, :TRACe2:AVERAge:TYPE?和:TRACe3:AVERAge:TYPE?均返回 ERR。

**返回格式**

查询返回 VID 或 RMS。

**举例**

下面的命令设置迹线 1 的平均类型为视频平均。

```
:TRACe1:AVERAge:TYPE VIDeo
```

下面的查询返回 VID。

```
:TRACe1:AVERAge:TYPE?
```

**:TRACe<n>:MODE****命令格式**

```
:TRACe<n>:MODE WRITe|MAXHold|MINHold|VIEW|BLANK|VIDeoavg|POWeravg
```

```
:TRACe<n>:MODE?
```

**功能描述**

设置指定迹线的类型。

查询指定迹线的类型。

**参数**

名称	类型	范围	默认值
<n>	离散型	1 2 3	—
—	关键字	WRITe MAXHold MINHold VIEW BLANK VIDeoavg POWeravg	WRITe

**说明**

WRITe: 清除写入。

MAXHold: 最大保持。

MINHold: 最小保持。

VIEW: 查看。

BLANK: 关闭。

VIDeoavg: 视频平均。

POWeravg: 功率平均。

**返回格式**

查询返回 WRIT、MAXH、MINH、VIEW、BLANK、VID 或 POW。

**举例**

下面的命令设置迹线 1 的类型为视频平均。

```
:TRACe1:MODE VIDeoavg
```

下面的查询返回 VID。

```
:TRACe1:MODE?
```

## :TRIGger命令子系统

命令列表:

- ◆ [:TRIGger:SEQuence:EXTernal:READy?](#)
- ◆ [:TRIGger:SEQuence:EXTernal:SLOPe](#)
- ◆ [:TRIGger:SEQuence:SOURce](#)
- ◆ [:TRIGger:SEQuence:VIDeo:LEVel](#)

### :TRIGger:SEQuence:EXTernal:READy?

命令格式

```
:TRIGger:SEQuence:EXTernal:READy?
```

功能描述

查询当前触发是否准备完毕。

说明

仅当选择外部触发时，此命令有效。

返回格式

返回 1—触发准备完毕；

返回 0—触发正在准备中。

### :TRIGger:SEQuence:EXTernal:SLOPe

命令格式

```
:TRIGger:SEQuence:EXTernal:SLOPe POSitive|NEGative
:TRIGger:SEQuence:EXTernal:SLOPe?
```

功能描述

设置外部触发的触发边沿。

查询外部触发的触发边沿。

参数

名称	类型	范围	默认值
——	关键字	POSitive NEGative	POSitive

说明

POSitive: 上升沿。

NEGative: 下降沿。

返回格式

查询返回 POS 或 NEG。

举例

下面的命令设置外部触发的触发沿为上升沿。

```
:TRIGger:SEQuence:EXTernal:SLOPe POSitive
```

下面的查询返回 POS。

```
:TRIGger:SEQuence:EXTernal:SLOPe?
```



## :TRIGger:SEQuence:SOURce

### 命令格式

```
:TRIGger:SEQuence:SOURce IMMEDIATE|VIDeo|EXTernal
:TRIGger:SEQuence:SOURce?
```

### 功能描述

设置触发类型。  
查询触发类型。

### 参数

名称	类型	范围	默认值
—	关键字	IMMEDIATE VIDeo EXTernal	IMMEDIATE

### 说明

IMMEDIATE: 自由触发。

VIDeo: 视频触发。当选择非零扫宽或在零扫宽条件下选择“有效值平均”或“电压平均”检波方式时，该触发方式不可用。

EXTernal: 外部触发。

### 返回格式

查询返回 IMM、VID 或 EXT。

### 举例

下面的命令设置触发类型为外部触发。

```
:TRIGger:SEQuence:SOURce EXTernal
```

下面的查询返回 EXT。

```
:TRIGger:SEQuence:SOURce?
```

## :TRIGger:SEQuence:VIDeo:LEVel

### 命令格式

```
:TRIGger:SEQuence:VIDeo:LEVel <ampl>
:TRIGger:SEQuence:VIDeo:LEVel?
```

### 功能描述

设置视频触发时的触发电平。  
查询视频触发时的触发电平。

### 参数

名称	类型	范围	默认值
<ampl>	连续实型	-300 dBm 至 50 dBm	0 dBm

### 说明

该命令仅在选择视频触发时有效，您可以使用:[TRIGger:SEQuence:SOURce](#) 命令选择视频触发。

注意：当选择非零扫宽或在零扫宽条件下选择“有效值平均”或“电压平均”检波方式时，视频触发方式不可用。

### 返回格式

查询以科学计数形式返回触发电平值。

### 举例

下面的命令设置触发电平为 10 dBm。

```
:TRIGger:SEQuence:VIDeo:LEVel 10
```

下面的查询返回 1.000000E+01。

```
:TRIGger:SEQuence:VIDeo:LEVel?
```

## :UNIT命令子系统

命令列表:

◆ [:UNIT:POWer](#)

### :UNIT:POWer

命令格式

:UNIT:POWer DBM|DBMV|DBUV|V|W

:UNIT:POWer?

功能描述

设置 Y 轴单位。

查询 Y 轴单位。

参数

名称	类型	范围	默认值
—	关键字	DBM DBMV DBUV V W	参考“说明”

说明

对数单位默认为 dBm。

线性单位默认为 V。

返回格式

查询返回 DBM、DBMV、DBUV、V 或 W。

举例

下面的命令设置幅度单位为 Watts。

```
:UNIT:POWer W
```

下面的查询返回 W。

```
:UNIT:POWer?
```



## 第3章 编程实例

本章例举了在 Visual C++ 6.0, Visual Basic 6.0 和 LabVIEW 8.6 开发环境中如何使用命令实现频谱仪常用功能的编程实例，同时给出在 Linux 操作系统下控制频谱仪实现常用功能的编程实例。这些实例都是基于 NI (National Instrument) -VISA (Virtual Instrument Software Architecture) 库编程实现的。

NI-VISA(National Instrument-Virtual Instrument Software Architecture)是美国国家仪器有限公司根据 VISA 标准编写的应用程序接口。您可以使用 NI-VISA 通过 USB 等仪器总线实现频谱仪与 PC 的通信。VISA 定义了一套软件命令，用户无需了解接口总线如何工作，就可以对仪器进行控制。具体细节可参考 NI-VISA 的帮助。

### 本章主要内容：

- ◆ [编程说明](#)
- ◆ [编程准备](#)
- ◆ [Visual C++ 6.0 编程实例](#)
- ◆ [Visual Basic 6.0 编程实例](#)
- ◆ [LabVIEW 8.6 编程实例](#)
- ◆ [Linux 编程实例](#)

## 编程说明

本节介绍了在编程操作过程中可能出现的一些问题及解决方法。当您遇到如下这些问题时，请按照相应的说明进行处理。

1. 当您使用网络搭建的工作环境时，建议单独搭建一个纯净的局域网环境。
2. 如果局域网环境比较复杂（如，有很多设备和广播消息）时，建议在编程过程中增加一些容错处理。详细信息请参考“[Visual C++ 6.0 编程实例](#)”一节中的仪器写/读操作带异常处理函数“[InstrWriteEx\(\)](#)”和“[InstrReadEx\(\)](#)”。
3. 本设备的 Socket 编程端口号是 5555。

## 编程准备

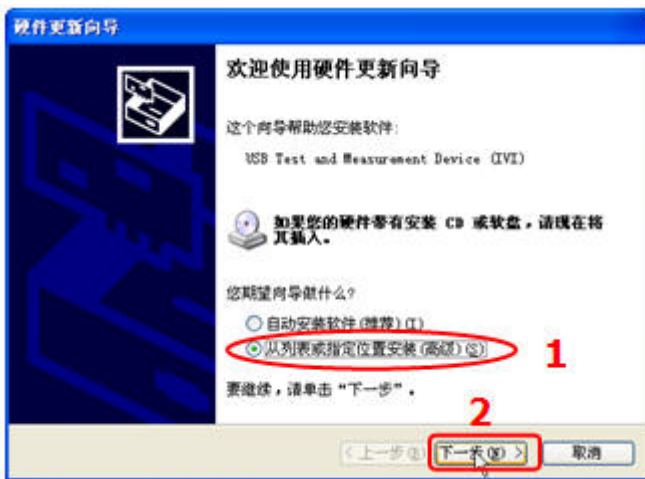
本节所述编程准备工作仅适用于在 Windows 操作系统下使用 Visual C++ 6.0、Visual Basic 6.0 和 LabVIEW 8.6 开发工具进行编程。关于在 Linux 操作系统下的编程准备，请参考“[Linux 编程实例](#)”一节中的“[Linux 编程准备](#)”。

首先确认您的电脑上是否已经安装 NI 的 VISA 库（可到 <http://www.ni.com/visa/> 下载）。本文中默认安装路径为 C:\Program Files\IVI Foundation\VISA。

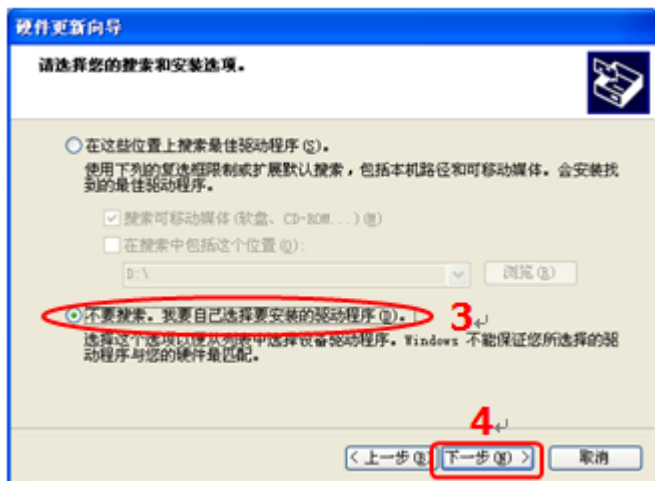
通过频谱仪的 USB 接口与 PC 通信。请使用 USB 数据线将频谱仪后面板的 USB Device 接口与 PC 的 USB 接口相连。

频谱仪与 PC 正确连接后，启动仪器，此时 PC 上将弹出“**硬件更新向导**”对话框，请按照向导的提示安装“**USB Test and Measurement Device (IVI)**”（安装步骤如下）。

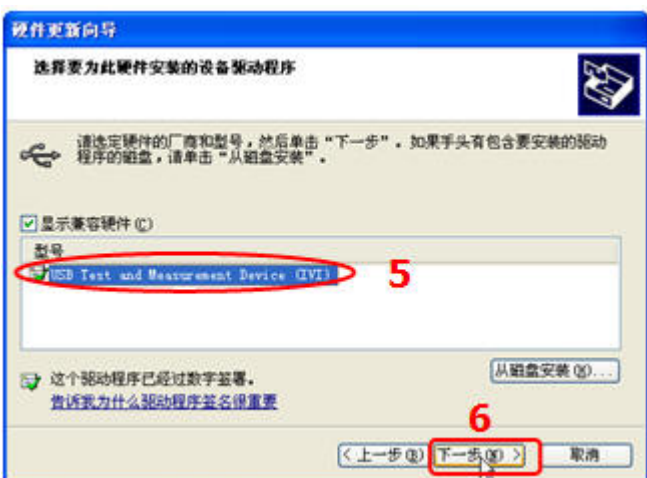
1. 选择“从列表或指定位置安装（高级）”；
2. 点击“下一步”；



3. 选择“不要搜索。我要自己选择要安装的驱动程序”；
4. 点击“下一步”；



- 5. 选择“USB Test and Measurement Device (IVI)”;
- 6. 点击“下一步”;



- 7. 安装过程结束后，点击“完成”。



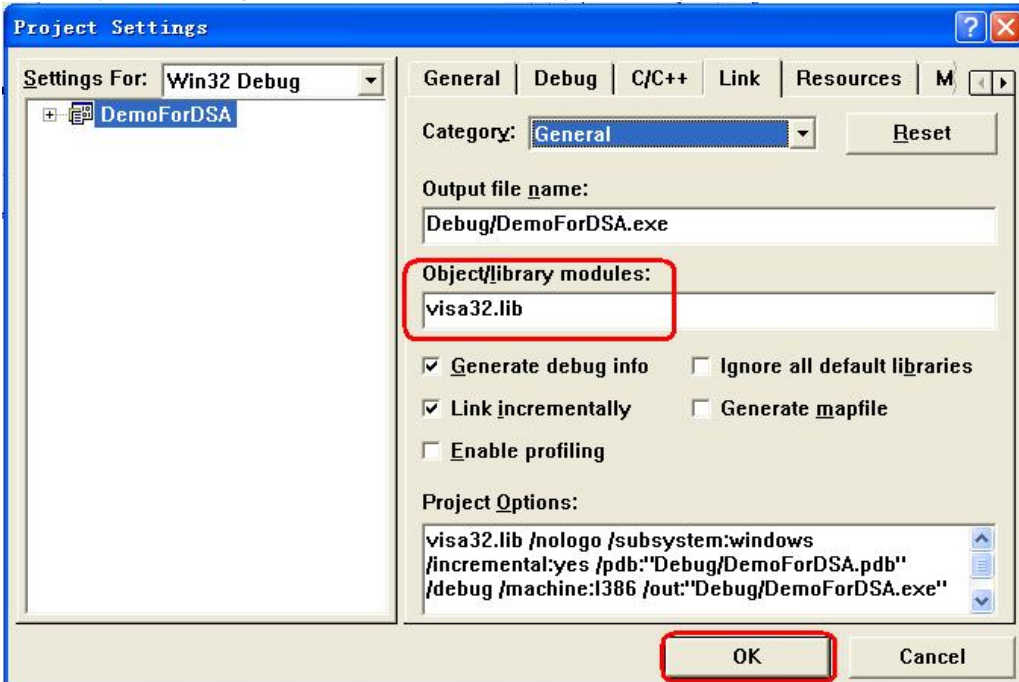


至此，编程准备工作结束，下面将详细介绍在 Visual C++ 6.0、Visual Basic 6.0 和 LabVIEW 8.6 开发环境中的编程实例。

## Visual C++ 6.0 编程实例

进入 Visual C++ 6.0 编程环境，按照下列步骤操作：

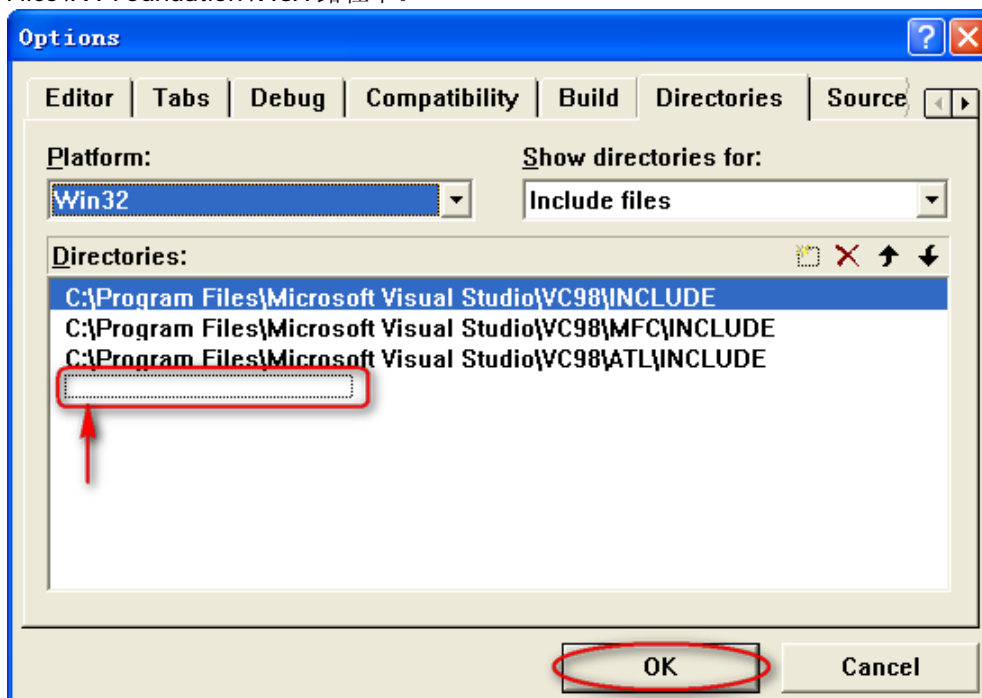
1. 建立一个基于对话框的 MFC 的工程，命名为 DemoForDSA。
2. 打开 Project→Settings 中的 Link 选项卡，在 Object/library modules 中手动添加 visa32.lib。



3. 打开 Tools → Options 中的 Directories 选项卡。  
在 Show directories for 中选择 Include files，双击 Directories 选框中的空白处添加 Include 的路径：  
C:\Program Files\IVI Foundation\VISA\WinNT\include。  
在 Show directories for 中选择 Library files，双击 Directories 选框中的空白处添加 Lib 的路径：C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc。

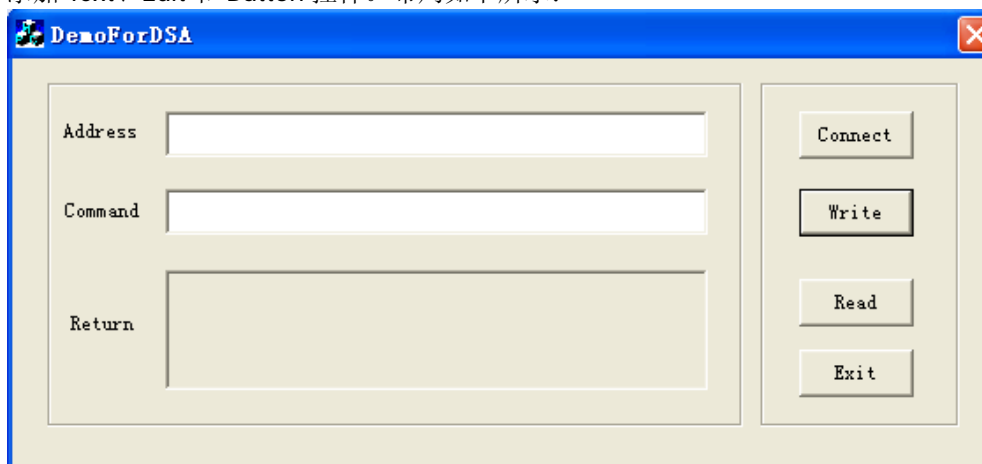


注意: 此处添加的两个路径与您计算机上 NI-VISA 的安装路径相关。此处默认 NI-VISA 安装在 C:\Program Files\IVI Foundation\VISA 路径下。



至此, VISA 库添加完毕。

4. 添加 Text、Edit 和 Button 控件。布局如下所示:



5. 添加控件变量。

打开 View→ClassWizard 的 Member Variables 选项卡, 添加下述三个变量:

仪器地址 CString m\_strInstrAddr

命令 CString m\_strCommand

返回值 CString m\_strResult

6. 封装 VISA 的读和写操作。

- 1) 对 VISA 的写操作进行封装以便于操作。

```
bool CDemoForDSADlg::InstrWrite(CString strAddr, CString strContent) //Write operation
{
    ViSession defaultRM,instr;
    ViStatus status;
    ViUInt32 retCount;
    char * SendBuf = NULL;
```

```

char * SendAddr = NULL;
bool bWriteOK = false;
CString str;

// Change the address's data style from CString to char*
SendAddr = strAddr.GetBuffer(strAddr.GetLength());
strcpy(SendAddr,strAddr);
strAddr.ReleaseBuffer();

// Change the command's data style from CString to char*
SendBuf = strContent.GetBuffer(strContent.GetLength());
strcpy(SendBuf,strContent);
strContent.ReleaseBuffer();

//open a VISA resource
status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
    AfxMessageBox("No VISA resource was opened!");
    return false;
}

status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

//Write command to the instrument
status = viWrite(instr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);

//Close the system
status = viClose(instr);
status = viClose(defaultRM);

return bWriteOK;
}

```

- 2) 对 VISA 的读操作进行封装便于操作。

```

bool CDemoForDSADlg::InstrRead(CString strAddr, CString *pstrResult) //Read operation
{
    ViSession defaultRM,instr;
    ViStatus status;
    ViUInt32 retCount;
    char * SendAddr = NULL;
    unsigned char RecBuf[MAX_REC_SIZE];
    bool bReadOK = false;
    CString str;

    // Change the address's data style from CString to char*
    SendAddr = strAddr.GetBuffer(strAddr.GetLength());
    strcpy(SendAddr,strAddr);
    strAddr.ReleaseBuffer();

    memset(RecBuf,0,MAX_REC_SIZE);

    //Open a VISA resource
    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        // Error Initializing VISA...exiting

```

```

    AfxMessageBox("No VISA resource was opened!");
    return false;
}

//Open the instrument
status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

//Read from the instrument
status = viRead(instr, RecBuf, MAX_REC_SIZE, &retCount);

//close the system
status = viClose(instr);
status = viClose(defaultRM);

(*pstrResult).Format("%s", RecBuf);

return bReadOK;
}

```

- 3) 对 VISA 的读操作带异常处理函数的封装。

```

ViStatus CDemoForDSADlg::OpenVisaDevice(CString strAddr) //Open a VISA device
{
    ViStatus status;
    char * SendAddr = NULL;

    // Change the address's data style from CString to char*
    SendAddr = strAddr.GetBuffer(strAddr.GetLength());
    strcpy(SendAddr, strAddr);
    strAddr.ReleaseBuffer();

    //Open a VISA resource
    status = viOpenDefaultRM(&m_SessRM);

    if (status == 0)
    {
        //Open the device
        status = viOpen(m_SessRM, SendAddr, VI_NULL, VI_NULL, &m_SessInstr);

        //If you fails to open the connection, close the resource
        if (status != 0)
        {
            viClose(m_SessRM);
        }
    }

    return status;
}

```

```

ViStatus CDemoForDSADlg::CloseVisaDevice() //Close a VISA device
{
    ViStatus status;

    //Close the device
    status = viClose(m_SessInstr);

    if (status == 0)

```

```

{
    //close the resource
    status = viClose(m_SessRM);
}

return status;
}

bool CDemoForDSADlg::InstrWriteEx(CString strAddr, CString strContent) //Write operation with
exception handling
{
    ViStatus status;
    ViUInt32 retCount;
    char * SendBuf = NULL;
    bool bWriteOK = true;

    // Change the address's data style from CString to char*
    SendBuf = strContent.GetBuffer(strContent.GetLength());
    strcpy(SendBuf, strContent);
    strContent.ReleaseBuffer();

    do
    {
        //Write command to the instrument
        status = viWrite(m_SessInstr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);

        //If an error occurs, perform error handling
        if (status < 0)
        {
            //If the time exceed the limit value, resend the command after a delay of 1s
            if (VI_ERROR_TMO == status)
            {
                Sleep(1000);
                status = viWrite(m_SessInstr, (unsigned char *)SendBuf, strlen(SendBuf),
                &retCount);
            }
            else
            {
                //If another error occurs, reopen the connection after the connection is closed and
                resend the command
                status = CloseVisaDevice();
                Sleep(1000);
                status = OpenVisaDevice(m_strInstrAddr);
                if (status == 0)
                {
                    status = viWrite(m_SessInstr, (unsigned char *)SendBuf, strlen(SendBuf),
                    &retCount);
                }
            }
        }
    } while (status < 0);

    return bWriteOK;
}

bool CDemoForDSADlg::InstrReadEx(CString strAddr, CString *pstrResult) //Read operation with
exception handling

```

```

{
ViStatus status;
ViUInt32 retCount;
char * SendAddr = NULL;
unsigned char RecBuf[MAX_REC_SIZE];
bool bReadOK = true;

// Change the address's data style from CString to char*
SendAddr = strAddr.GetBuffer(strAddr.GetLength());
strcpy(SendAddr,strAddr);
strAddr.ReleaseBuffer();

memset(RecBuf,0,MAX_REC_SIZE);

do
{
//Read from the instrument
status = viRead(m_SessInstr, RecBuf, MAX_REC_SIZE, &retCount);
if (status < 0)
{
//If the time exceed the limit value, read from the instrument after a delay of 1s
if (VI_ERROR_TMO == status)
{
Sleep(1000);
status = viRead(m_SessInstr, RecBuf, MAX_REC_SIZE, &retCount);
}
else
{
//If another error occurs, reopen the connection after the connection is closed and
reread from instrument
status = CloseVisaDevice();
Sleep(1000);
status = OpenVisaDevice(m_strInstrAddr);
if (status == 0)
{
status = viRead(m_SessInstr, RecBuf, MAX_REC_SIZE, &retCount);
}
}
}
} while (status < 0);

(*pstrResult).Format("%s",RecBuf);

return bReadOK;
}

```

## 7. 增加控件消息响应代码。

### 1) 连接仪器

```

void CDemoForDSADlg::OnBtConnectInstr() // Connect to the instrument
{
// TODO: Add your control notification handler code here
ViStatus status;
ViSession defaultRM;
ViString expr = "?*";
ViPFindList findList = new unsigned long;

```

```

    ViPUInt32 retcnt = new unsigned long;
    ViChar instrDesc[1000];
    CString strSrc = "";
    CString strInstr = "";
    unsigned long i = 0;
    bool bFindDSA = false;

    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        // Error Initializing VISA...exiting
        MessageBox("No VISA instrument was opened ! ");
        return ;
    }

    memset(instrDesc,0,1000);

    // Find resource
    status = viFindRsrc(defaultRM,expr,findList, retcnt, instrDesc);

    for (i = 0;i < (*retcnt);i++)
    {
        // Get instrument name
        strSrc.Format("%s",instrDesc);
        InstrWrite(strSrc,"*IDN?");
        ::Sleep(200);
        InstrRead(strSrc,&strInstr);

        // If the instrument(resource) belongs to the DSA series then jump out //from the loop
        strInstr.MakeUpper();
        if (strInstr.Find("DSA") >= 0)
        {
            bFindDSA = true;
            m_strInstrAddr = strSrc;
            break;
        }

        //Find next instrument
        status = viFindNext(*findList,instrDesc);
    }

    if (bFindDSA == false)
    {
        MessageBox("Didn't find any DSA!");
    }
    UpdateData(false);
}

```

## 2) 写操作

```

void CDemoForDSADlg::OnBtWrite()           //Write operation
{
    // TODO: Add your control notification handler code here
    UpdateData(true);
    if (m_strInstrAddr.IsEmpty())
    {
        MessageBox("Please connect to the instrument first!");
    }
}

```

```
InstrWrite(m_strInstrAddr,m_strCommand);  
m_strResult.Empty();  
UpdateData(false);  
}
```

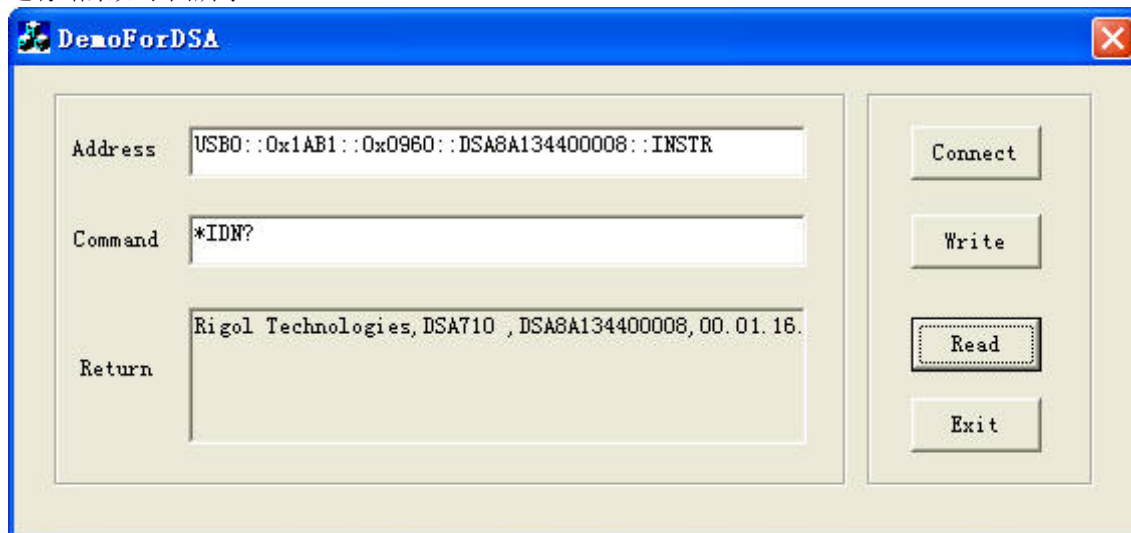
## 3) 读操作

```
void CDemoForDSADlg::OnBtRead()           //Read operation  
{  
    // TODO: Add your control notification handler code here  
    UpdateData(true);  
    InstrRead(m_strInstrAddr,&m_strResult);  
    UpdateData(false);  
}
```

## 8. 运行结果。

- 1) 点击“Connect”寻找频谱仪；
- 2) 在“Command”编辑框中输入“\*IDN?”；
- 3) 点击“Write”将命令写入频谱仪中；
- 4) 点击“Read”读取返回值。

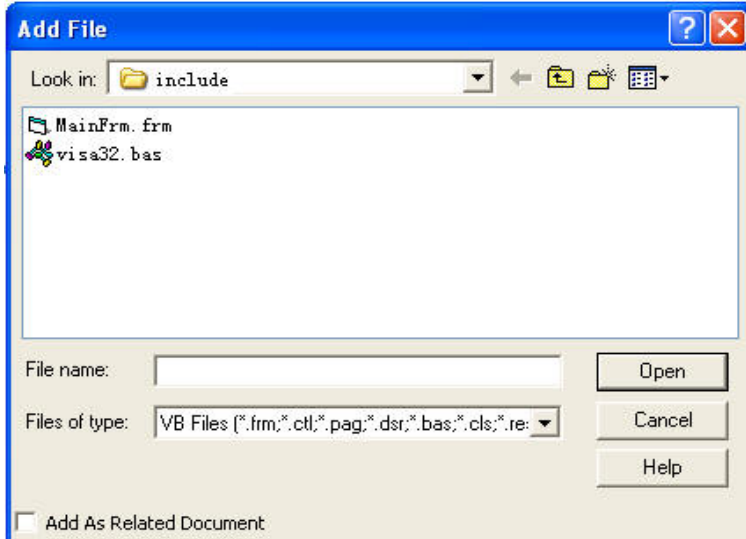
运行结果如下图所示。



## Visual Basic 6.0 编程实例

进入 Visual Basic 6.0 编程环境，按照下列步骤操作：

1. 建立一个标准应用程序工程（Standard EXE），命名为 DemoForDSA。
2. 打开 Project → Add File...，在 NI-VISA 安装路径下的 include 文件夹中查找 visa32.bas 文件并添加到工程中，visa32.bas 模块包含了所有 VISA 库中的函数和常数声明。



随后增加 Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long) 声明语句到 visa32.bas 中，也可以新建模块来声明 Sleep 函数。

3. 添加 Label、Text 和 Button 控件。布局如下所示：



4. 封装 VISA 的读和写操作
  - 1) 对 VISA 的写操作进行封装便于操作。

```

'-----
'Function Name: InstrWrite
'Function: Send command to the instrument
'Input: rsrcName,instrument(resource) name
strCmd,Command
'-----
Public Sub InstrWrite(rsrcName As String, strCmd As String)
    Dim status As Long
    Dim dfltRM As Long
    Dim sesn As Long
    Dim rSize As Long

```



```

'Initialize the system
status = viOpenDefaultRM(dfltRM)
'Failed to initialize the system
If (status < VI_SUCCESS) Then
    MsgBox " No VISA resource was opened! "
    Exit Sub
End If
'Open the VISA instrument
status = viOpen(dfltRM, rsrcName, VI_NULL, VI_NULL, sesn)
'Failed to open the instrument
If (status < VI_SUCCESS) Then
    MsgBox "Failed to open the instrument! "
    Exit Sub
End If

'Write command to the instrument
status = viWrite(sesn, strCmd, Len(strCmd), rSize)
'Failed to write to the instrument
If (status < VI_SUCCESS) Then
    MsgBox " Failed to write to the instrument! "
    Exit Sub
End If

'Close the system
status = viClose(sesn)
status = viClose(dfltRM)

End Sub

```

- 2) 对 VISA 的读操作进行封装便于操作。

```

'-----
'Function Name: InstrRead
'Function: Read the return value from the instrument
'Input: rsrcName,Resource name
'Return: The string gotten from the instrument
'-----
Public Function InstrRead(rsrcName As String) As String
    Dim status As Long
    Dim dfltRM As Long
    Dim sesn As Long
    Dim strTemp0 As String * 256
    Dim strTemp1 As String
    Dim rSize As Long

    'Begin by initializing the system
    status = viOpenDefaultRM(dfltRM)
    'Initial failed
    If (status < VI_SUCCESS) Then
        MsgBox " Failed to open the instrument! "
        Exit Function
    End If
    'Open the instrument
    status = viOpen(dfltRM, rsrcName, VI_NULL, VI_NULL, sesn)
    'Open instrument failed
    If (status < VI_SUCCESS) Then
        MsgBox " Failed to open the instrument! "
        Exit Function
    End If

```

```

End If

' Read from the instrument
status = viRead(sesn, strTemp0, 256, rSize)
' Read failed
If (status < VI_SUCCESS) Then
    MsgBox " Failed to read from the instrument! "
    Exit Function
End If

'Close the system
status = viClose(sesn)
status = viClose(dfItRM)

' Remove the space at the end of the string
strTemp1 = Left(strTemp0, rSize)
InstrRead = strTemp1
End Function

```

## 5. 增加控件事件代码

### 1) 连接仪器

```

' Connect to the instrument
Private Sub CmdConnect_Click()
    Const MAX_CNT = 200
    Dim status As Long
    Dim dfItRM As Long
    Dim sesn As Long
    Dim fList As Long
    Dim buffer As String * MAX_CNT, Desc As String * 256
    Dim nList As Long, retCount As Long
    Dim rsrcName(19) As String * VI_FIND_BUFLEN, instrDesc As String * VI_FIND_BUFLEN
    Dim i, j As Long
    Dim strRet As String
    Dim bFindDSA As Boolean

    ' Initialize the system
    status = viOpenDefaultRM(dfItRM)
    ' Initialize failed
    If (status < VI_SUCCESS) Then
        MsgBox " No VISA resource was opened ! "
        Exit Sub
    End If

    ' Find instrument resource
    Call viFindRsrc(dfItRM, "USB?*INSTR", fList, nList, rsrcName(0))
    ' Get the list of the instrument(resource)
    strRet = ""
    bFindDSA = False
    For i = 0 To nList - 1
        ' Get the instrument name
        InstrWrite rsrcName(i), "*IDN?"
        Sleep 200
        strRet = InstrRead(rsrcName(i))
        ' Continue to switch the resource until find a DSA instrument
        strRet = UCase(strRet)
        j = InStr(strRet, "DSA")
        If (j >= 0) Then

```

```

        bFindDSA = True
    Exit For
End If

    Call viFindNext(fList + i - 1, rsrcName(i))
Next i
'Display
If (bFindDSA = True) Then
    TxtInsAddr.Text = rsrcName(i)
Else
    TxtInsAddr.Text = ""
End If
End Sub

```

## 2) 写操作

'Write the command to the instrument

```

Private Sub CmdWrite_Click()
    If (TxtInsAddr.Text = "") Then
        MsgBox ("Please write the instrument address! ")
    End If

    InstrWrite TxtInsAddr.Text, TxtCommand.Text
End Sub

```

## 3) 读操作

'Read the return value from the instrument

```

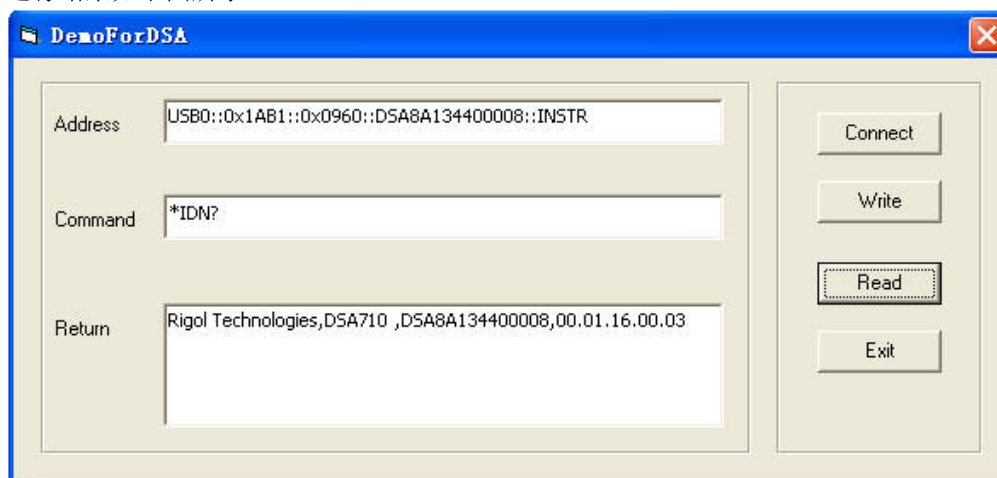
Private Sub CmdRead_Click()
    Dim strTemp As String
    strTemp = InstrRead(TxtInsAddr.Text)
    TxtReturn.Text = strTemp
End Sub

```

## 6. 运行结果

- 1) 点击“Connect”寻找频谱仪；
- 2) 在“Command”编辑框中输入“\*IDN?”；
- 3) 点击“Write”将命令写入频谱仪中；
- 4) 点击“Read”读取返回值。

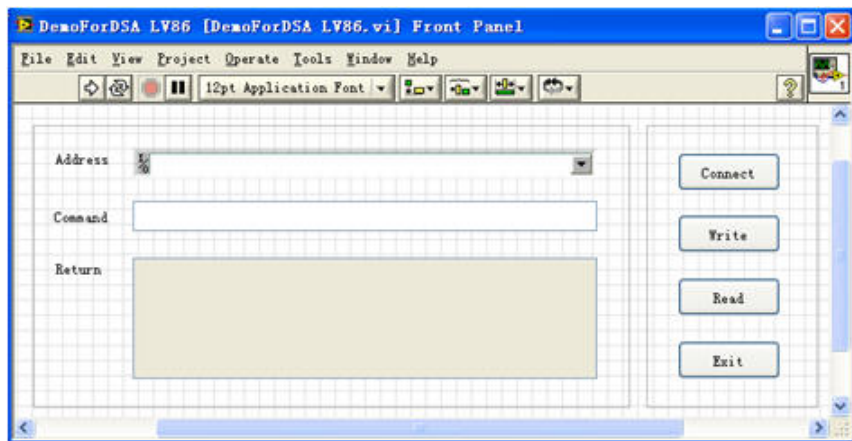
运行结果如下图所示。



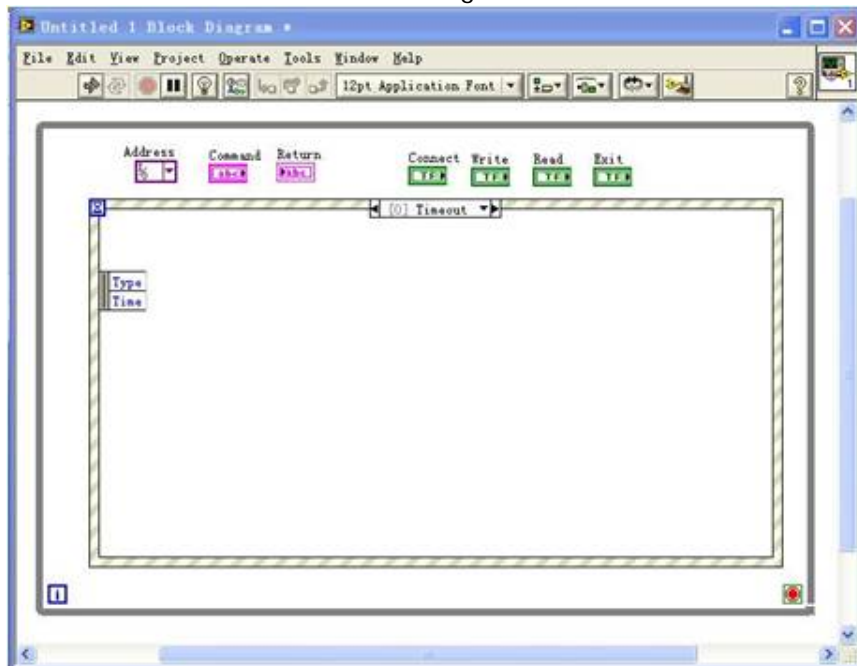
## LabVIEW 8.6 编程实例

进入 Labview 8.6 编程环境，按照下列步骤操作：

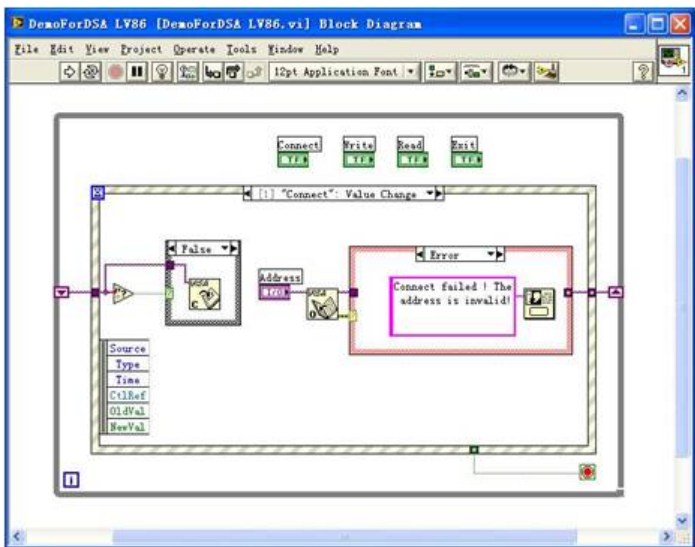
1. 新建一个 VI 文件，命名为 DemoForDSA LV86。
2. 在前面板界面添加控件，包括地址栏、命令栏和返回值栏以及连接、写入、读取和退出按钮。



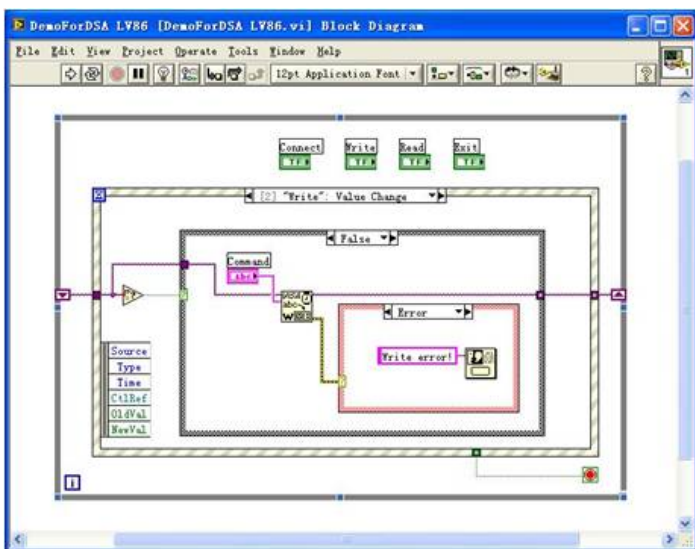
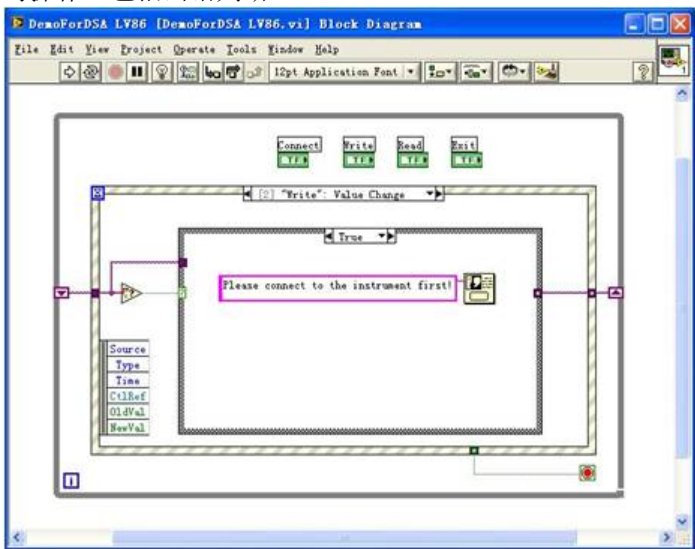
3. 点击 Window 菜单下的 Show Block Diagram，创建事件结构。



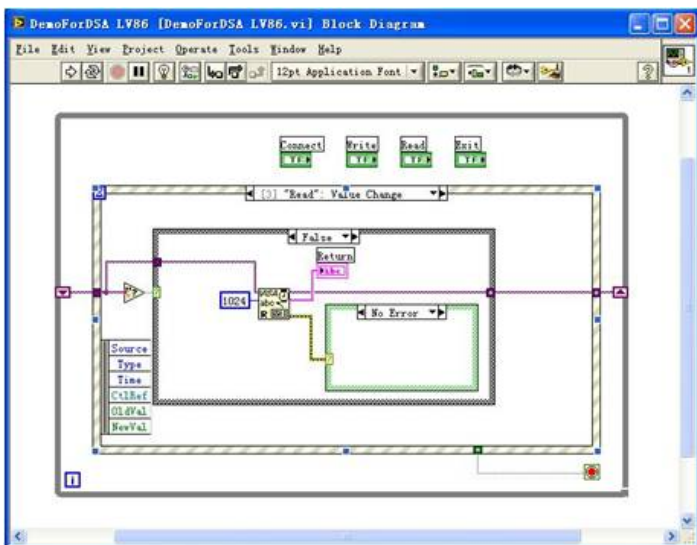
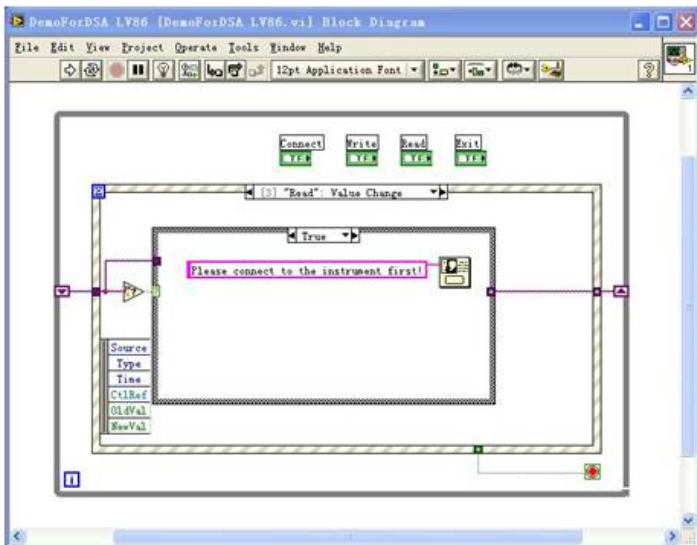
4. 添加事件（包括连接仪器、写操作、读操作和退出）。
  - 1) 连接仪器



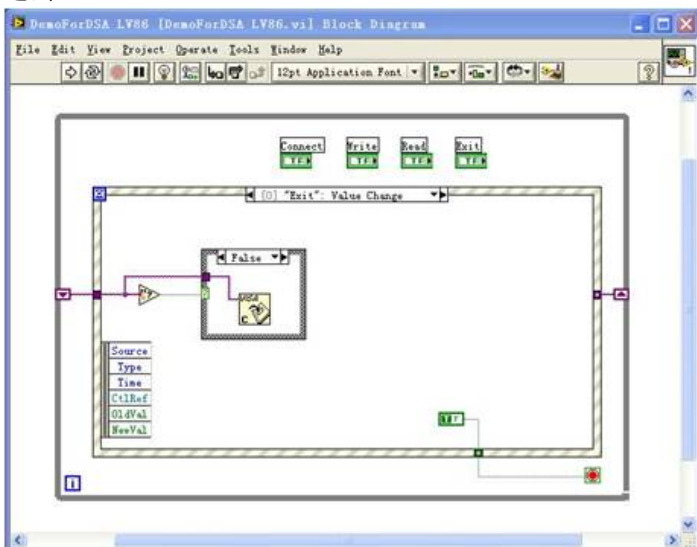
2) 写操作（包括出错判断）



3) 读操作（包括出错处理）

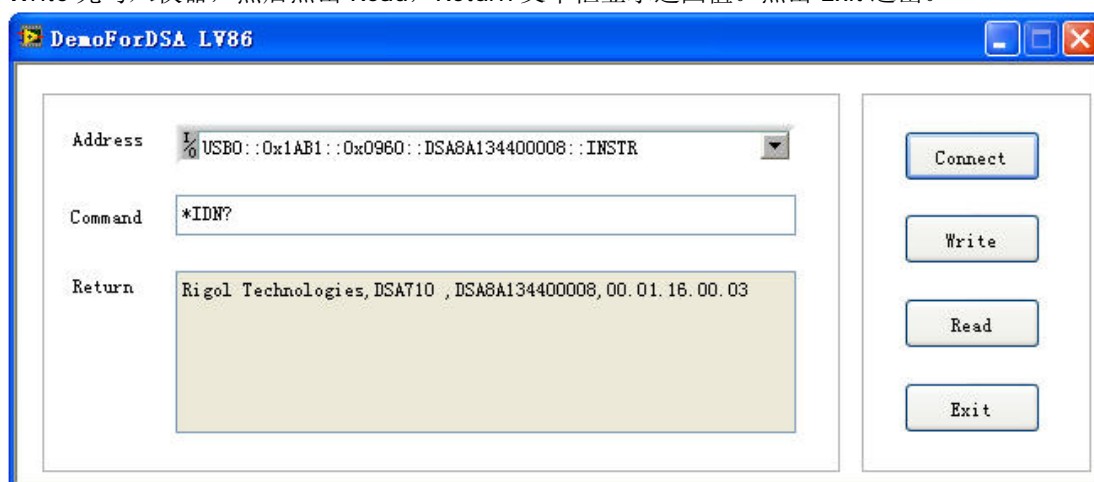


4) 退出



5) 运行程序，出现如下图所示界面。点击 Address 下拉框选择 VISA 资源名称，点击 Connect 连接仪

器，在 Command 文本框中输入命令，点击 Write 写入仪器。若为查询命令（如\*IDN?），需点击 Write 先写入仪器，然后点击 Read，Return 文本框显示返回值。点击 Exit 退出。



## Linux编程实例

本节演示如何在 Linux 操作系统下编程控制频谱仪实现常用功能。

### Linux编程准备

1. 本节所用的编程环境：  
操作系统：Fedora 8 (Linux-2.6.23)  
GCC 版本：gcc-4.1.2
2. 安装 VISA 库：确认您的电脑上是否已经安装 NI 的 VISA 库（可到 NI 网站 <http://www.ni.com/visa/> 下载）。如果没有安装可按照以下步骤安装：  
先从 NI 网站上下载 VISA 库 NI-VISA-4.4.0.ISO。

创建新目录

```
#mkdir NI_VISA
```

挂载 iso 文件

```
#mount -o loop -t iso9660 NI-VISA-4.4.0.iso NI_VISA
```

进入目录 NI\_VISA 下进行安装

```
#cd NI_VISA
```

```
#./INSTALL
```

卸载 iso 文件

```
#umount NI_VISA
```

安装完成后，默认的安装路径为 /usr/local。

3. 本节通过频谱仪的 LAN 口与 PC 通信。请使用网线将频谱仪后面板的 LAN 接口与 PC 的 LAN 接口相连。您也可以使用网线将频谱仪连接至 PC 所在的局域网内。

频谱仪与 PC 正确连接后，配置频谱仪的网络地址，使之与 PC 的网络地址在同一个网络段中。例如，PC 的网络地址与 DNS 设置分别如下图所示。那么，频谱仪网络地址应配置为：

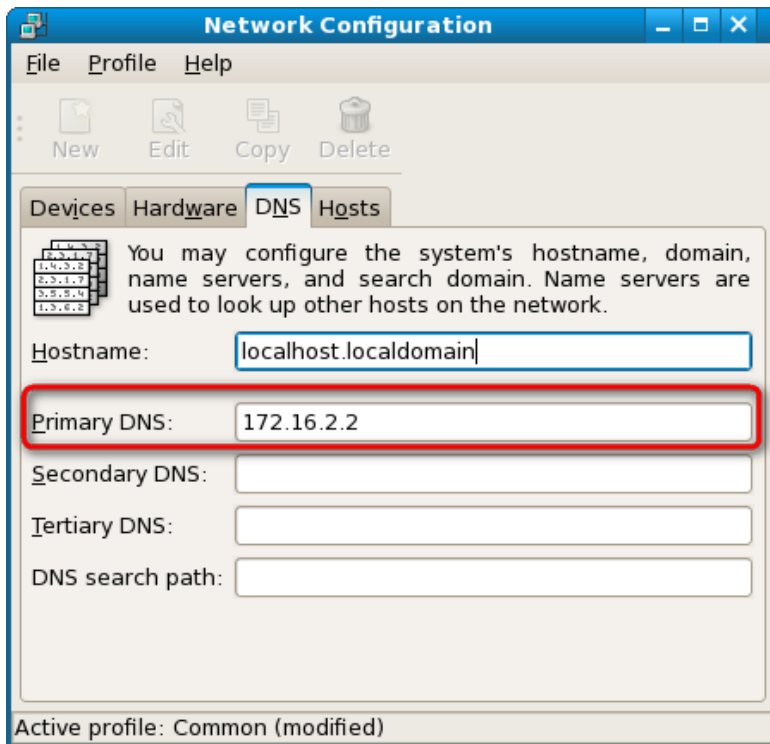
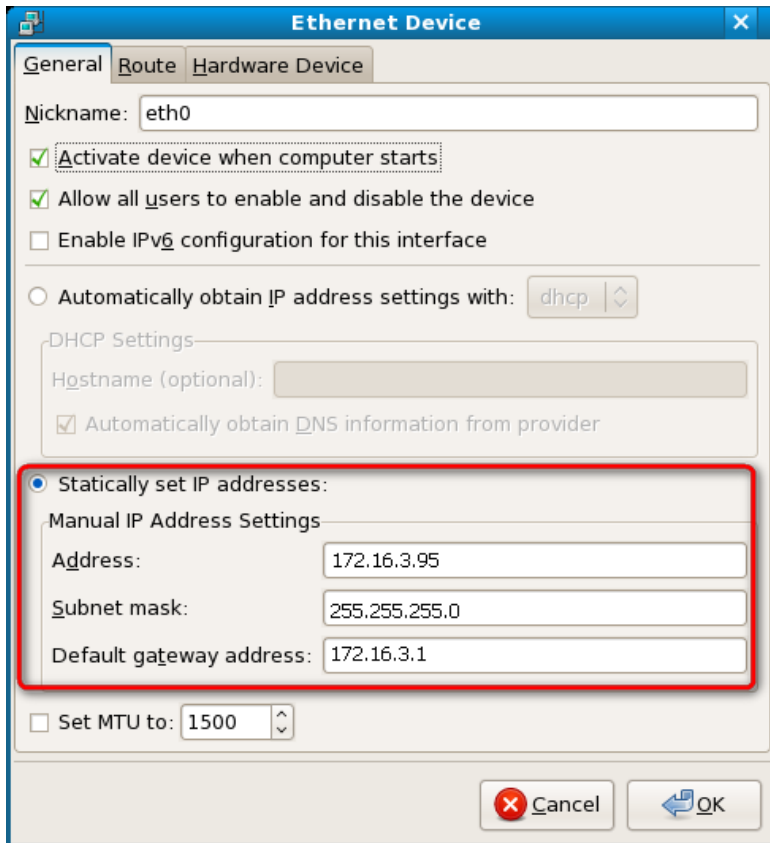
IP 地址：172.16.3.X\*

默认网关：172.16.3.1

子网掩码：255.255.255.0

DNS：172.16.2.2

注\*：X 的取值范围为 2 至 254 中尚未被使用的值。



4. 请使用如下两种方式之一将库的位置添加到库的搜索路径之中，以便程序能够自动加载所安装的库文件。





## Linux编程步骤

1. 编辑 DemoForDSA.h 头文件，声明一个类，以封装对仪器的操作和属性。

```
#ifndef DEMO_FOR_DSA_H
#define DEMO_FOR_DSA_H

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <iostream>
// #include <syswait.h>
using namespace std;

#define MAX_SEND_BUF_SIZE 50
#define MAX_REC_SIZE 300

class DemoForDSA
{
// Construction
public:
DemoForDSA();
bool InstrRead(string strAddr, string & pstrResult);
bool InstrWrite(string strAddr, string strContent);
bool ConnectInstr();

string m_strInstrAddr;
string m_strResult;
string m_strCommand;

};

void makeupper(string & instr);

#endif
```

2. 编辑 DemoForDSA.cpp 文件，实现对仪器的各种操作。

```
#include "visa.h"
#include "DemoForDSA.h"

DemoForDSA::DemoForDSA()
{
m_strInstrAddr = "";
m_strResult = "";
m_strCommand = "";
}

bool DemoForDSA::ConnectInstr()

{
ViUInt32 retCount;
ViStatus status;
ViSession defaultRM;
ViString expr = "?*";
ViPFindList findList = new unsigned long;
ViPUInt32 retcnt = new unsigned long;
string strSrc = "";
string strInstr = "";
```

```
ViChar instrDesc[1000];

unsigned long i = 0;
bool bFindDSA = false;
memset(instrDesc,0,1000);

//打开VISA设备
status = viOpenDefaultRM(&defaultRM);

if (status < VI_SUCCESS)
{
    cout<<"No VISA equipment!"<<endl;
    return false;
}

//寻找资源
status = viFindRsrc(defaultRM,expr,findList, retcnt, instrDesc);

for (i = 0;i < (*retcnt);i++)
{
    //获取仪器的名称
    strSrc=

    InstrWrite(strSrc,"*IDN?");
    usleep(200);
    InstrRead(strSrc,strInstr);

    //如果是DSA系列的,则找到仪器,退出
    makeupper(strInstr);
    if (strInstr.find("DSA",0) > 0)
    {
        bFindDSA = true;
        m_strInstrAddr = strSrc;
        break;
    }

    //获取下一设备
    status = viFindNext(*findList,instrDesc);
}

if (bFindDSA == false)
{
    printf("DSA device not found!\n");
    return false;
}

return true;
}

bool DemoForDSA::InstrWrite(string strAddr, string strContent) //写操作
{
    ViSession defaultRM,instr;
    ViStatus status;
    ViUInt32 retCount;
    char * SendBuf = NULL;
    char * SendAddr = NULL;
    bool bWriteOK = false;
```

```
string str;

    //地址转换, 把string类型转为char*
SendAddr = const_cast<char*>(strAddr.c_str());

    //地址转换, 把string类型转为char*
SendBuf = const_cast<char*>(strContent.c_str());

//打开具体设备
status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
    cout<<"No VISA equipment!"<<endl;
    return false;
}

status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

//往设备写命令
status = viWrite(instr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);

    //关闭设备
status = viClose(instr);
status = viClose(defaultRM);
return bWriteOK;
}

bool DemoForDSA::InstrRead(string strAddr, string & pstrResult) //仪器读
{
    ViSession defaultRM,instr;
    ViStatus status;
    ViUInt32 retCount;
    char* SendAddr = NULL;
    char * result = NULL;
    bool bReadOK = false;
    unsigned char RecBuf[MAX_REC_SIZE];
    string str;
    memset(RecBuf,0,MAX_REC_SIZE);

    result=char*)malloc(MAX_REC_SIZE*sizeof(char));
    memset(result,0,MAX_REC_SIZE);

    //地址转换, 把string类型转为char*
SendAddr=const_cast<char*>(strAddr.c_str());

//打开VISA设备
status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
    // Error Initializing VISA...exiting
    cout<<"No VISA equipment!"<<endl;
    return false;
}

    //打开具体设备
status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);
```

```
//读取设备
status = viRead(instr, RecBuf, MAX_REC_SIZE, &retCount);

//关闭设备
status = viClose(instr);
status = viClose(defaultRM);
sprintf(result, "%s", RecBuf);
pstrResult = result;
free(result);
return bReadOK;
}

void makeupper( string &instr)
{
    string outstr = "";
    if(instr == "")
    {
        exit(0);
    }

    for(int i = 0; i < instr.length(); i++)
    {
        instr[i] = toupper(instr[i]);
    }
}
```

3. 编辑函数文件 mainloop.cpp，完成流程控制。

```
#include "DemoForDSA.h"

void menudisplay()
{
    cout << "\t\t Please operate the instrument:\n read write quit" << endl;
}

int main()
{
    DemoForDSA demo;
    char temp[50];
    if(!demo.ConnectInstr())
    {
        cout << "can not connect the equipment!" << endl;
        return 0;
    }
    else
    {
        cout << "\n connect equipment success!" << endl;
        cout << " the equipment address is : " << demo.m_strInstrAddr << endl;
    }

    while(1)
    {
        menudisplay();
        //cin >> demo.m_strCommand;
        cin.getline(temp, 50);
    }
}
```

```

demo.m_strCommand=
if(demo.m_strCommand[0]='r' && demo.m_strCommand[1]='e'
  && demo.m_strCommand[2]='a' && demo.m_strCommand[3]='d')
{
  //demo.InstrWrite(demo.m_strInstrAddr,"*IDN?");
  //demo.InstrRead(demo.m_strInstrAddr,demo.m_strResult);
  cout<<"read result:"<<demo.m_strResult<<endl;
  demo.m_strResult="";
}

else if (demo.m_strCommand[0]='w' && demo.m_strCommand[1]='r'
  && demo.m_strCommand[2]='i' && demo.m_strCommand[3]='t' &&
  demo.m_strCommand[4]='e')
{
  if (demo.m_strInstrAddr=="")
  {
    cout<<"Please connect the instrument! \n";
  }
  demo.InstrWrite(demo.m_strInstrAddr,demo.m_strCommand.substr(5,40));
  usleep(200);

  //读仪器
  demo.InstrRead(demo.m_strInstrAddr,demo.m_strResult);
}

else if (demo.m_strCommand[0] == 'q' && demo.m_strCommand[1] == 'u'
  && demo.m_strCommand[2] == 'i' && demo.m_strCommand[3] == 't')
{
  break;
}
else if(demo.m_strCommand != "")
{
  cout<<"Bad command!"<<endl;
}
}
return 1;
}

```

#### 4. makefile 文件

```
src = DemoForDSA.cpp mainloop.cpp DemoForDSA.h
```

```
obj = DemoForDSA.o mainloop.o
```

```
INCLUDE= -I/usr/local/vxipnp/linux/include
```

```
LIB= -lvisa -lc -lpthread
```

```
CC=
```

```
demo : $(obj)
```

```
$(CC) $(INCLUDE) $(LIB) -o demo $(obj)
```

```
mainloop.o : mainloop.cpp DemoForDSA.h
```

```
$(CC) -c $< -o $@
```

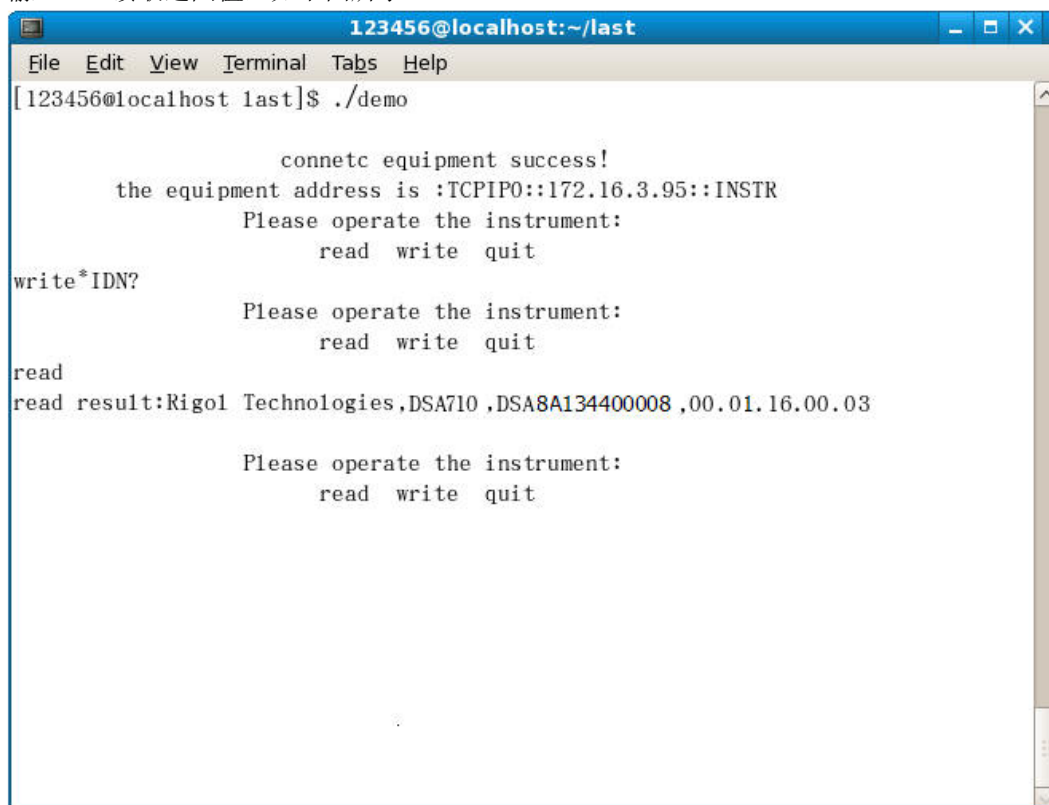
```
DemoForDSA.o: DemoForDSA.cpp DemoForDSA.h
```

```
$(CC) -c $< -o $@
```

```
.PHONY : clean
clean:
    rm demo $(obj)
```

## 5. 运行结果

- 1) #make
- 2) ./demo
- 3) 程序运行，自动连接仪器。如果没有找到仪器，提示 “No VISA equipment!”，并退出。如果找到仪器，并连接成功，出现如图示界面。
- 4) 输入write<命令> 将命令写入频谱仪中，例如：write<\*IDN?>。
- 5) 输入read读取返回值，如下图所示。



```
123456@localhost:~/last
File Edit View Terminal Tabs Help
[123456@localhost last]$ ./demo

          connetc equipment success!
the equipment address is :TCPIP0::172.16.3.95::INSTR
Please operate the instrument:
    read write quit
write*IDN?
Please operate the instrument:
    read write quit
read
read result:Rigol Technologies,DSA710 ,DSA8A134400008 ,00.01.16.00.03

Please operate the instrument:
    read write quit
```