

SSA3000X 频谱分析仪

编程手册

PG0703X-C04A

目 录

1.	远程控制概述	3
1.1	如何远程控制	3
1.2	通信协议	5
1.3	远程控制功能	7
2.	SCPI 简介	11
2.1	命令格式	11
2.2	符号说明	11
2.3	参数类型	12
2.4	命令缩写	13
3.	命令系统	14
3.1	IEEE 公用命令子系统	14
3.2	系统命令	16
3.3	仪器模式命令	21
3.4	Initiate 命令子系统	22
3.5	Sense 命令子系统	22
3.6	Calculate 命令系统	40
3.7	Measurement 命令系统	57
3.8	触发	66
3.9	TG	67
3.10	调制解调	69
3.11	校正	70
3.12	存储命令	71
4.	编程示例	72
4.1	使用 VISA 的编程示例	72
4.2	使用 Sockets/Telnet 的编程示例	86

1. 远程控制概述

SSA3000X 频谱分析仪支持通过 USB、LAN、GPIB-USB 接口与计算机进行通信。用户通过这些接口，结合相应的编程语言或 NI-VISA，使用基于 SCPI（Standard Commands for Programmable Instruments）命令集，可对仪器进行远程编程控制，以及和其他支持 SCPI 命令集的可编程仪器进行互操作。本章将介绍如何构建频谱仪与电脑之间的远程通信。

1.1 如何远程控制

频谱分析仪提供 USB 和 LAN 连接，允许您使用控制器计算机设置远程操作环境。控制器计算机可以是个人计算机（PC）或小型计算机，以及一些智能仪器。

1.1.1 使用 USB 接口连接

请参考以下步骤通过 USB 设备完成与 PC 的连接：

1. 在 PC 上安装 NI-VISA 以获得 USB-TMC 驱动。
2. 使用 USB A-B 电缆将频谱分析仪的 USB Device 端口连接到 PC 的 USB Host 口。



3. 打开频谱分析仪。

频谱分析仪将被自动检测为新的 USB 设备。

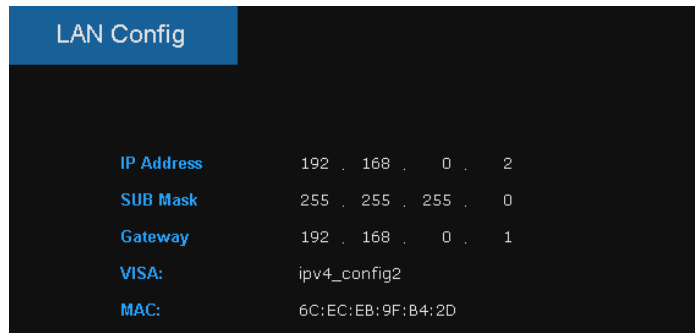
1.1.2 使用 LAN 接口连接

请参考以下步骤通过 LAN 完成与 PC 的连接：

1. 在 PC 上安装 NI-VISA 以获得 VXI 驱动。或者，在没有 NI-VISA 的情况下，使用 PC 操作系统中的 Socket 或 Telnet。
2. 使用网络电缆将频谱分析仪 LAN 端口连接到 PC 的 LAN 口。



3. 打开频谱分析仪。
4. 按下前面板 **System** → Interface → LAN 上的按钮进入 LAN Config 功能菜单。
5. 选择静态或动态的 IP 配置。
 - ◆ 动态: 当前网络中的 DHCP 服务器将为分析仪自动分配网络参数 (IP 地址, 子网掩码, 网关)。
 - ◆ 静态: 您可以手动设置 IP 地址, 子网掩码, 网关方式。 设置后按下应用按钮。



频谱分析仪将被自动或手动检测为新的 LAN 设备。

1.1.3使用 USB-GPIB 适配器连接

请参考以下步骤通过 USB-GPIB 完成与 PC 的连接:

1. 在 PC 上安装 NI-VISA 以获取 GPIB 驱动。
2. 使用 SIGLENT USB-GPIB 适配器将与 PC 的 USB Host 端口连接到 PC 的 GPIB 卡端口。



3. 打开分析仪。

4. 按下前面板 **System** → Interface → GPIB 上的按钮，输入 GPIB 编号。

频谱分析仪将被自动检测为新的 GPIB 点。

1.2 通信协议

1.2.1 通过 VISA 建立通信

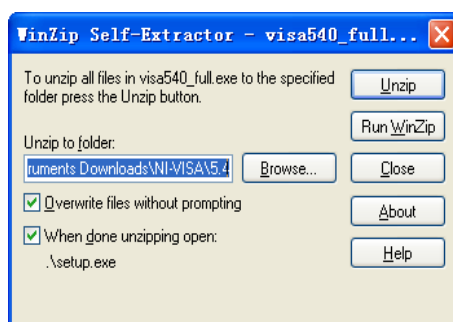
NI-VISA 包含运行引擎版本和完整版本。运行引擎版本提供 NI 设备驱动程序，例如 USB-TMC，VXI，GPIB 等。完整版包括运行引擎和名为 NI MAX 的软件工具，它提供用户界面来控制设备。

您可以从以下网址获得 NI-VISA 完整版本：

<http://www.ni.com/download/>.

下载后，您可以按照以下步骤进行安装：

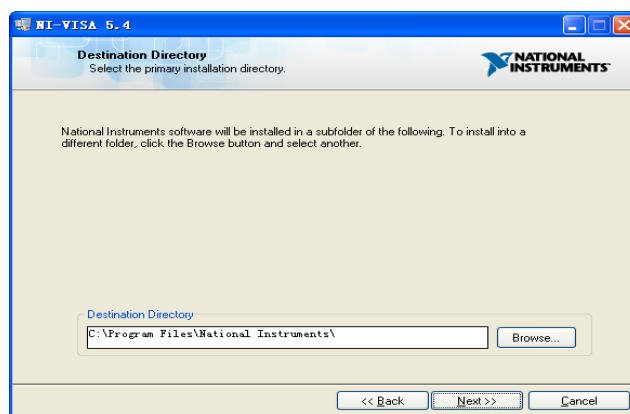
a. 双击 `visa_full.exe`，出现如下对话框：



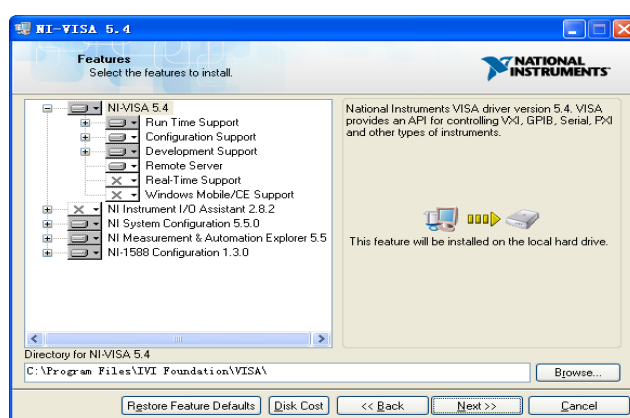
b. 点击 Unzip，解压缩文件后安装过程会自动启动。如果您的计算机需要安装 .NET Framework 4，则其安装过程将自动启动。



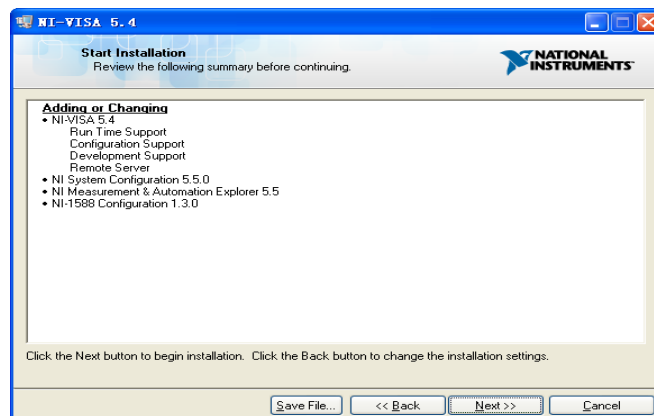
c. 上面显示了 NI-VISA 安装对话框。单击下一步开始安装过程。



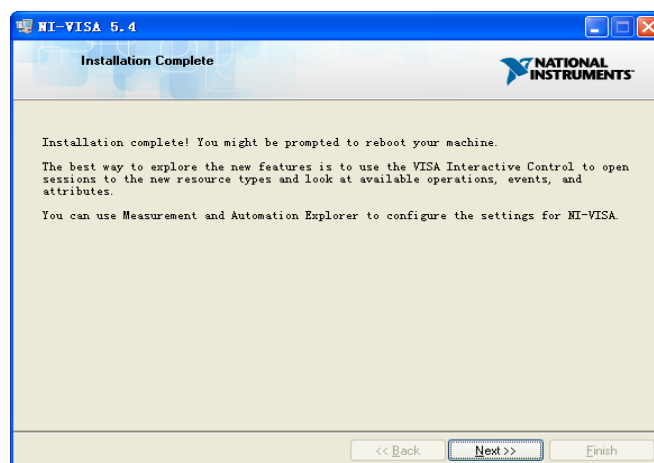
设置安装路径，默认路径是“C: \ Program Files \ National Instruments \”，您可以更改它。点击 Next，对话框如上所示。



d.点击下一步两次，在许可协议对话框中选择“ I accept the above 2 License Agreement(s).”，然后点击下一步，对话框如下图所示：



e.单击下一步运行安装。



现在安装完成，重新启动您的电脑。

1.2.2 通过 Sockets/Telnet 建立通信

通过 LAN 接口，可以使用 VXI-11，Sockets 和 Telnet 协议与频谱分析仪进行通信。VXI-11 在 NI-VISA 中提供，而 Sockets 和 Telnet 通常原本就包含在 PC 的操作系统中。

Socket LAN 是一种用于通过 LAN 接口使用 TCP/IP 与频谱分析仪通信的方法。Sockets 是用于计算机网络的基本技术，允许应用程序使用内置于网络硬件和操作系统中的标准机制进行通信。通过该方法访问频谱分析仪上的端口，从中可以建立与网络计算机的双向通信。

在使用 Socket LAN 之前，您必须选择要使用的分析仪的 Socket 端口号：

- ◆ 标准（Socket）模式：在端口 5025 上可用。使用此端口进行编程。
- ◆ 远程登录（Telnet）模式：Telnet SCPI 服务在端口 5024 上可用。

1.3 远程控制功能

1.3.1 用户自定义编程

用户可以使用 SCPI 命令来编程和控制频谱分析仪。有关详细信息，请参阅“编程示例”中的介绍。

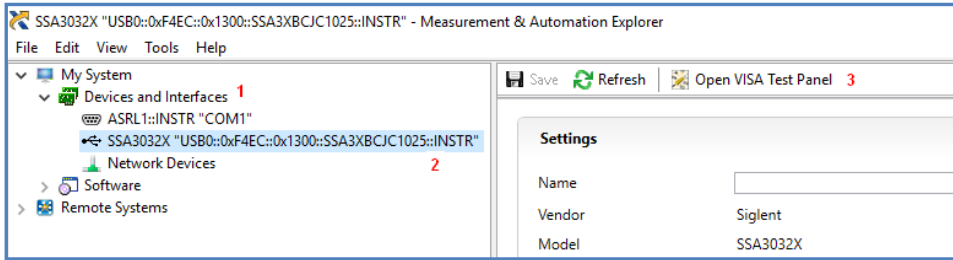
1.3.2 通过 NI MAX 发送 SCPI 命令

用户可以通过 NI-MAX 软件发送 SCPI 命令来远程控制频谱分析仪。

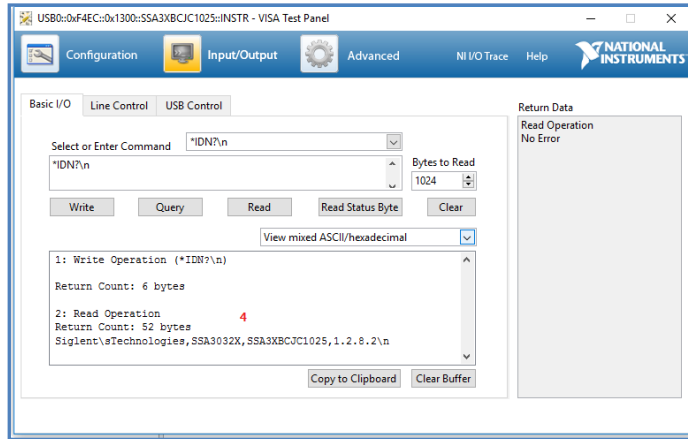
1.3.2.1 使用 USB 接口

运行 NI MAX:

1. 点击软件左上角的“Device and interface”；
2. 找到“USBTMC”设备符号；



3. 点击“Open VISA Test Panel”选项按钮，将出现以下界面。
4. 点击“Input/Output”选项按钮，然后点击“Query”选项按钮以查看操作信息。

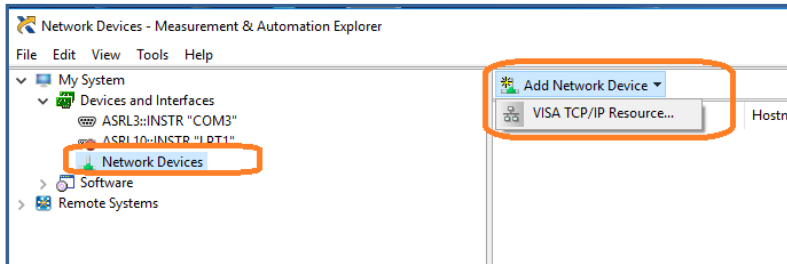


NOTE: * IDN? 命令（称为识别查询）应返回仪器制造商，仪器型号，序列号和其他识别信息。

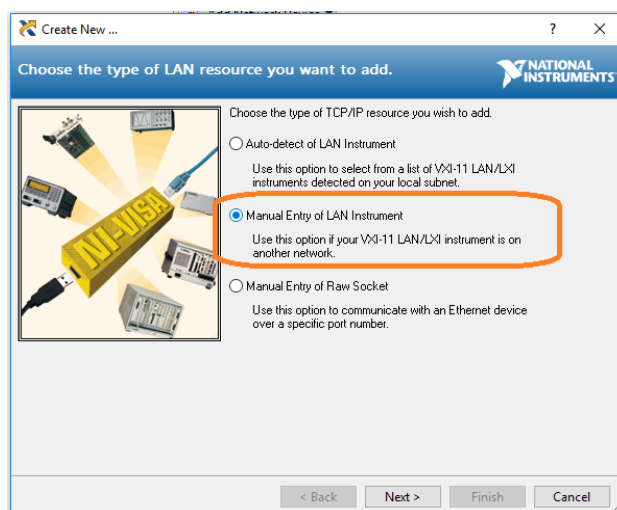
1.3.2.2 使用 LAN 接口

如下所示，选择 Add Network Device，然后选择 VISA TCP / IP Resource: 运行 NI MAX:

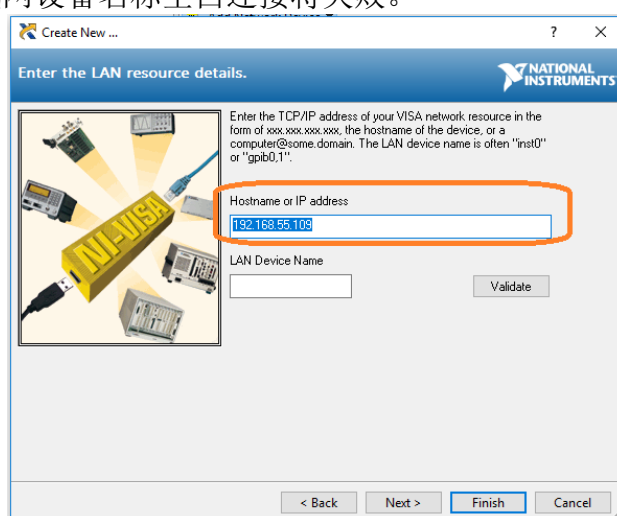
1. 点击软件左上角的：“Device and interface”；
2. 找到“Network Devices”设备符号，点击“Add Network Devices”；



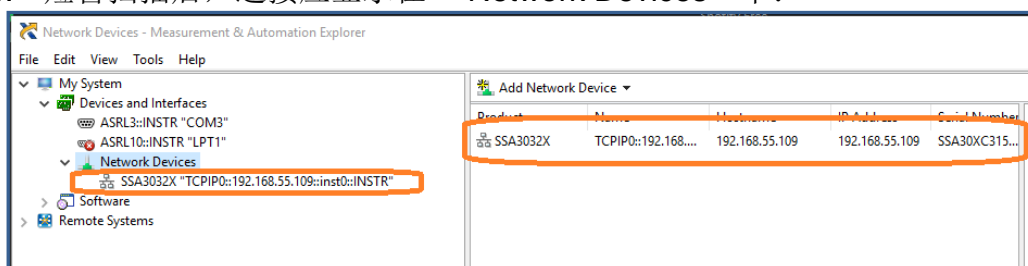
3. 选择 LAN 仪器的手动输入，选择下一步，然后输入 IP 地址，如图所示。点击“Finish”建立连接：



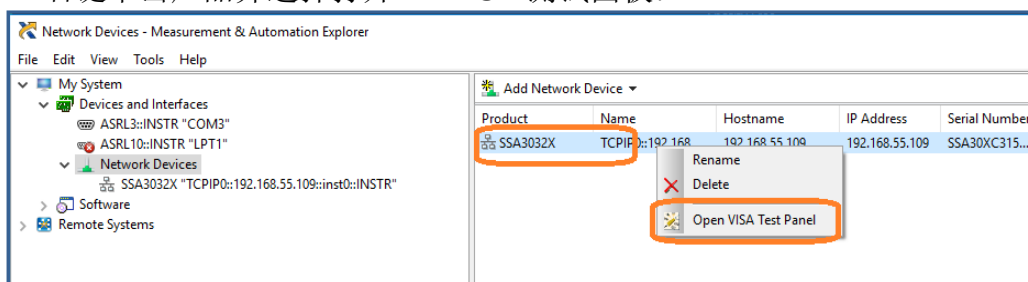
NOTE: 保留局域网设备名称空白连接将失败。



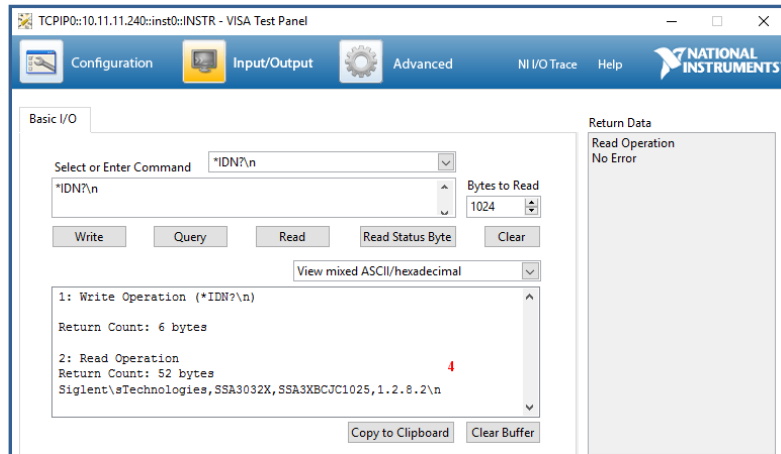
4. 短暂扫描后，连接应显示在” Network Devices”下：



5. 右键单击产品并选择打开 NI-VISA 测试面板：



6. 点击“输入/输出”选项按钮，然后点击“查询”选项按钮。如果一切正常，您将看到如下所示返回的读取操作信息。



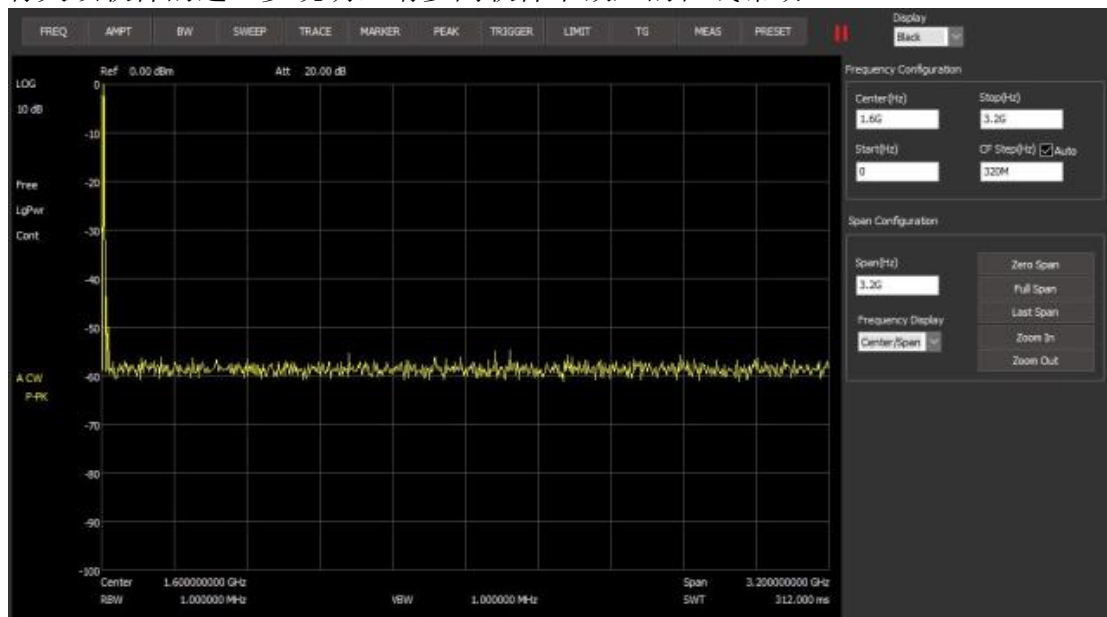
1.3.3 EasySpectrum 软件

用户可以通过 EasySpectrum 远程控制频谱分析仪。PC 软件 EasySpectrum 是针对 Siglent 频谱分析仪使用的基于 PC-Windows 的远程控制工具。您可以从 Siglent 的网站下载它。要通过 USB / LAN 端口将分析仪连接到 PC，您需要先安装 NI VISA。

EasySpectrum 有如下功能：

- ◆ 频谱监视器，用于与分析仪同时显示和控制跟踪扫描。
- ◆ 一个文件制作器，用于获取用户定义的极限/校正文件，并将它们从 PC 载入分析仪；
- ◆ 一个 EMI 接收器，用于执行 EMI 预先一致性测试，包括预扫描，峰值搜索，最终扫描和报告生成。

有关该软件的进一步说明，请参阅软件中嵌入的在线帮助。



2. SCPI 简介

2.1 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，命令和参数以“空格”分开，多个参数的，参数之间用逗号“,”分隔。命令行后面添加问号“?”，表示对此功能进行查询。例如：

```
:SENSe:FREQuency:CENTer <freq>
```

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

SENSe 是命令的根关键字，FREQuency 和 CENTer 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，<freq>表示可设置的参数；问号“?”表示查询；命令: :SENSe:FREQuency:CENTer 和参数<freq>之间用“空格”分开。

2.2 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。

1、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：

```
:CALCulate:LLINe[1]:2:DATA <x-axis>,<ampl>{,<x-axis>,<ampl>}命令中，后面大括号中的{,<x-axis>,<ampl>}可以省略，也可以设置一对或多对频率、幅度、连接状态参数。
```

2、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：

```
[[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF|ON|0|1]命令中，可选择的命令参数为“OFF”、“ON”、“0”或“1”。
```

3、方括号 []

方括号中的内容（命令关键字）是可省略的。如果省略参数，仪器将该参数设置为默认值。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?
```

4、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:DEMod:VOLume 5 的形式发送:DEMod:VOLume <value>命令。

2.3 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、枚举、整型、连续实型、离散型、ASCII 字符串。

1、布尔型

参数取值为“OFF”、“ON”、“0”或“1”。例如：

`[[:SENSe]:FREQUency:CENTer:STEP:AUTO OFF|ON|0|1`

2、枚举

参数取值为所列举的值。例如：

`[[:SENSe]:AVERage:TYPE LOGPower|POWER|VOLTage`

参数为“LOGPower”、“POWER”或“VOLTage”。

3、字符串

参数取值为 ASCII 字符的组合。例如：

`:SYSTem:COMMunicate:LAN:IPADdress <“xxx.xxx.xxx.xxx”>`

参数为设定的日期格式字符串。

4、整型

除非另有说明，参数在有效值范围内可以取任意整数值。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：

`[[:SENSe]:DEMod:VOLume <value>`

参数< value >可取 0 到 10 范围内的任一整数。

5、浮点型

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九位有效值），可以任意进行取值。例如：

`:CALCulate:BANDwidth:NDB <value>`

参数< value >可取-100 到 100 之间的实数。

6、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：

`[[:SENSe]:BWIDth:VIDeo:RATio <number>`

参数<number>只能取值为 0.001、0.003、0.01、0.03、0.1、0.3、1.0、3.0、10.0、30.0、100.0、300.0、1000.0。

2.4 命令缩写

所有命令对大小写不敏感，可完整输入命令，包含所有大写或小写，也可以使用缩写，但是如果缩写，必须完整且仅仅输入命令格式中的大写字母，例如：
:DISPlay:WINDow:TRACe:Y:DLINe:STATe?

可缩写成：

:DISP:WIND:TRAC:Y:DLIN:STAT?

3. 命令系统

本章主要介绍鼎阳 SSA3032X 命令系统，主要包括：

IEEE 公用命令	3.1
系统命令	3.2
仪器模式命令	3.3
Initiate 命令子系统	3.4
Sense 命令子系统	3.5
Calculate 命令系统	3.6
Measurement 命令系统	3.7
触发	3.8
TG	3.9
调制解调	3.10
校正	3.11

3.1 IEEE 公用命令子系统

3.1.1 设备信息查询 (*IDN)

命令格式	*IDN?
说明	这里会返回一个包含设备信息的字符串。串内容包括：厂商，设备型号，设备串号，软件版本号，FPGA 版本号，CPLD 版本号
菜单项	无
举例	*IDN? Return: Siglent Technologies,SSA3032,1234567890,100.01.02.06.01

3.1.2 复位 (*RST)

命令格式	*RST
说明	重新恢复设备的状态为初始状态
菜单项	无
举例	*RST

3.1.3 清除状态 (*CLS)

命令格式	*CLS
说明	将所有事件寄存器的值清零，同时清空错误列表

菜单项	无
举例	*CLS

3.1.4 标准事件状态使能 (*ESE)

命令格式	*ESE <number> *ESE?
说明	设定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
菜单项	无
举例	*ESE 16

3.1.5 标准事件状态寄存器询问 (*ESR)

命令格式	*ESR?
说明	询问及清除标准事件状态寄存器的事件值
菜单项	无
举例	*ESR?

3.1.6 操作完成询问 (*OPC)

命令格式	*OPC *OPC?
说明	所有操作结束后，在标准事件状态寄存器中设置比特 0 置 1 查询当前操作是否完成
菜单项	无
举例	*OPC?

3.1.7 允许服务请求 (*SRE)

命令格式	*SRE <integer> *SRE?
说明	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
菜单项	无
举例	*SRE 1

3.1.8 状态字节询问 (*STB)

命令格式	*STB
说明	查询状态字节寄存器的事件值
菜单项	无
举例	*STB

3.1.9 等待继续 (*WAI)

命令格式	*WAI
------	------

说 明	等待操作完成
菜单项	无
举例	*WAI

3.1.10 自测试询问 (*TST)

命令格式	*TST?
说 明	仪器自测试
菜单项	无
举例	*TRG

3.2 系统命令

3.2.1 系统时间 (:SYSTem:TIME)

命令格式	:SYSTem:TIME <hhmmss> :SYSTem:TIME?
说 明	设置系统时间 获取系统时间
参数类型	字符串
参数范围	小时 (0~23), 分 (0~59), 秒 (0~59)
返回值	字符串
默认值	无
菜单项	系统 > 日期&时间
举例	设置系统时间 :SYSTem:TIME 182559 查询 :SYSTem:TIME?

3.2.2 系统日期 (:SYSTem:DATE)

命令格式	:SYSTem:DATE <yyyymmdd> :SYSTem:DATE?
说 明	设置系统日期 获取系统日期
参数类型	字符串
参数范围	年 (四位长度), 月 (1~12), 日 (1~31)
返回值	字符串
默认值	无
菜单项	系统 > 日期&时间
举例	设置系统日期 :SYSTem:DATE 20050101 查询 :SYSTem:DATE?

3.2.3 IP 地址

(:SYSTem:COMMunicate:LAN:IPADdress)

命令格式	:SYSTem:COMMunicate:LAN:IPADdress <“xxx.xxx.xxx.xxx”> :SYSTem:COMMunicate:LAN:IPADdress?
说 明	设置 IP 地址 获取 IP 地址
参数类型	字符串
参数范围	必须符合 IP 设置规范 (0-255:0-255:0-255:0-255)
返回值	IP 地址字符串
默认值	无
菜单项	系统 > 接口设置 > LAN > IP 地址
举例	:SYSTem:COMMunicate:LAN:IPADdress “192.168.1.12” :SYSTem:COMMunicate:LAN:IPADdress?

3.2.4 网关 (:SYSTem:COMMunicate:LAN:GATeway)

命令格式	:SYSTem:COMMunicate:LAN:GATeway <“xxx.xxx.xxx.xxx”> :SYSTem:COMMunicate:LAN:GATeway?
说 明	设置网关 获取网关
参数类型	字符串
参数范围	必须符合 IP 地址的网卡规范 (0-255:0-255:0-255:0-255)
返回值	网关字符串
默认值	无
菜单项	系统 > 接口设置 > LAN > 网关
举例	:SYSTem:COMMunicate:LAN:GATeway “192.168.1.1” :SYSTem:COMMunicate:LAN:GATeway?

3.2.5 子网掩码 (:SYSTem:COMMunicate:LAN:SMASk)

命令格式	:SYSTem:COMMunicate:LAN:SMASk <“xxx.xxx.xxx.xxx”> :SYSTem:COMMunicate:LAN:SMASk?
说 明	根据电脑网络设置设置子网掩码 获取子网掩码
参数类型	字符串
参数范围	必须符合 IP 地址的网卡规范 (0-255:0-255:0-255:0-255)
返回值	子网掩码字符串
默认值	无
菜单项	系统 > 接口设置 > LAN > 子网掩码
举例	:SYSTem:COMMunicate:LAN:SMASk?

3.2.6 IP 分配类型 (:SYSTem:COMMunicate:LAN:TYPE)

命令格式	:SYSTem:COMMunicate:LAN:TYPE STATIC DHCP :SYSTem:COMMunicate:LAN:TYPE?
------	---

说明	设置 IP 的分配类型 读取 IP 的分配类型
参数类型	枚举
参数范围	STATIC: 静态分配 DHCP: 动态分配
返回值	枚举
默认值	无
菜单项	系统 > 接口设置 > LAN > IP 配置
举例	:SYSTem:COMMunicate:LAN:TYPE DHCP :SYSTem:COMMunicate:LAN:TYPE?

3.2.7 多国语言 (:SYSTem:LANGuage)

命令格式	:SYSTem:LANGuage SCHINESE ENGLISH :SYSTem:LANGuage?
说明	设置多国语言 获取多国语言
参数类型	枚举
参数范围	SCHINESE: 中文 ENGLISH: 英文
返回值	枚举
默认值	无
菜单项	系统 > 语言
举例	设置多国语言 :SYSTem:LANGuage SCHINESE 查询 :SYSTem:LANGuage?

3.2.8 开机类型 (:SYSTem:PON:TYPE)

命令格式	:SYSTem:PON:TYPE DFT LAST USER :SYSTem:PON:TYPE?
说明	设置开机加载配置类型 获取开机加载配置类型
参数类型	枚举
参数范围	DFT: 默认 LAST: 上次 USER: 用户
返回值	枚举
默认值	DFT
菜单项	系统 > 上电/复位 > 上电
举例	SYSTem:PON:TYPE DFT

3.2.9 系统复位 (:SYSTem:PRESet)

命令格式	:SYSTem:PRESet
说明	根据复位类型，复位机器的参数配置
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	无
举例	:SYSTem:PRESet

3.2.10 系统重启 (:SYSTem:REStart)

命令格式	:SYSTem:REStart
说明	系统重启（部分机器可能不支持重启，关机后需要手动开启）
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	无
举例	:SYSTem:REStart

3.2.11 复位类型 (:SYSTem:PRESet:TYPE)

命令格式	:SYSTem:PRESet:TYPE DFT LAST USER :SYSTem:PRESet:TYPE?
说明	设置复位配置类型 获取复位配置类型
参数类型	枚举
参数范围	DFT: 默认 LAST: 上次 USER: 用户
返回值	枚举
默认值	DFT
菜单项	系统 > 上电/复位 > 复位
举例	:SYSTem:PRESet:TYPE DFT

3.2.12 恢复出厂设置 (:SYSTem:FDEFault)

命令格式	:SYSTem:FDEFault
说明	恢复出厂默认设置
参数类型	无
参数范围	无
返回值	无

默认值	无
菜单项	系统 > 上电/复位 > 恢复出厂设置
举例	:SYSTem:FDEFault

3.2.13 启用选项（:SYSTem:LKEY）

命令格式	:SYSTem:LKEY <"option">,<"license key">
说明	用注册码启用指定选项，需要重启生效
参数类型	枚举、字符串
参数范围	"option": Meas EMI CAT TG "license key": 鼎阳科技提供授权码，16 位字符串
返回值	无
默认值	无
菜单项	系统 > 系统信息 > 选件
举例	:SYSTem:LKEY EMI,fjbdajffnklmgwno

3.2.14 安装选项询问（:SYSTem:OPTions?）

命令格式	:SYSTem:OPTions?
说明	返回已安装选项目录
参数类型	无
参数范围	无
返回值	"option": Meas EMI CAT TG
默认值	无
菜单项	系统 > 系统信息
举例	:SYSTem:OPTions?

3.2.15 关机（:SYSTem:POWer:OFF）

命令格式	:SYSTem:POWer:OFF
说明	关闭设备
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	无
举例	:SYSTem:POWer:OFF

3.2.16 系统信息（:SYSTem:CONFigure:SYSTem?）

命令格式	:SYSTem:CONFigure:SYSTem?
说明	询问设备系统信息
参数类型	无
参数范围	无
返回值	字符串

默认值	无
菜单项	系统 > 系统信息
举例	:SYSTem:CONFigure:SYSTem?

3.2.17 获取屏幕显示数据 (:HCOPY:SDUMp:DATA?)

命令格式	:HCOPY:SDUMp:DATA?
说明	询问屏幕显示数据。获取到的数据为.bmp 文件数据，用户可以自行保存为.bmp 图片。
参数类型	无
参数范围	无
返回值	字符串
默认值	无
菜单项	无
举例	:HCOPY:SDUMp:DATA?

3.3 仪器模式命令

3.3.1 模式选择 (:INSTrument[:SElect])

命令格式	:INSTrument[:SElect] SA CAT :INSTrument[:SElect]?
说明	选择频谱分析仪工作模式
参数类型	枚举
参数范围	SA: 频谱分析模式 CAT: 反射测量
返回值	枚举
默认值	SA
菜单项	模式
举例	:INSTrument CAT

3.3.2 测量模式选择 (:INSTrument:MEASure)

命令格式	:INSTrument:MEASure OFF ACPR CHPower OBW TPOWer SPECTrogram TOI :INSTrument:MEASure?
说明	设置测量模式 获取测量模式
参数类型	枚举
参数范围	OFF: 测量关 ACPR: 邻道功率比 CHPower: 信道功率 OBW: 占用带宽 TPOWer: 时域功率

	SPECtrogram: 频谱监测 TOI: 三阶交调
返回值	枚举
默认值	OFF
菜单项	测量
举例	:INSTrument:MEASure ACPR

3.4 Initiate 命令子系统

3.4.1 单次扫描 (:INITiate[:IMMediate])

命令格式	:INITiate[:IMMediate]
说明	执行一次 single sweep
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	扫描 > 单次
举例	:INITiate:IMMediate

3.4.2 连续扫描开关 (:INITiate:CONTInuous)

命令格式	:INITiate:CONTInuous OFF ON 0 1 :INITiate:CONTInuous?
说明	开关连续扫描模式 获取连续扫描开关模式
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	扫描 > 扫描模式
举例	:INITiate:CONTInuous OFF

3.5 Sense 命令子系统

3.5.1 频率控制

3.5.1.1 中心频率 ([:SENSe]:FREQUency:CENTer)

命令格式	[:SENSe]:FREQUency:CENTer <freq> [:SENSe]:FREQUency:CENTer?
说明	设置中心频率 按中心频率的步进递增或递减

	查询当前中心频率
参数类型	浮点型, 单位 Hz、KHz、MHz、GHz
参数范围	50 Hz~3.199999950 GHz (2.999999950 GHz、2.099999950 GHz、1.799999950 GHz、1.499999950 GHz、0.999999950 GHz) 零扫宽情况下为 0~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型, 单位 Hz
默认值	1.6 GHz (1.5 GHz、1.05 GHz、0.9 GHz、0.75 GHz、0.5 GHz)
菜单项	频率 > 中心频率
举例	:FREQUency:CENTer 0.2 GHz

3.5.1.2 起始频率 ([:SENSe]:FREQUency:STARt)

命令格式	[:SENSe]:FREQUency:STARt <freq> [:SENSe]:FREQUency:STARt?
说明	设置起始频率 查询当前起始频率
参数类型	浮点型, 单位 Hz、KHz、MHz、GHz
参数范围	0 Hz~3.199999900 GHz (2.999999900 GHz、2.099999900 GHz、1.799999900 GHz、1.499999900 GHz、0.999999900 GHz) 零扫宽情况下为 0~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型, 单位 Hz
默认值	0 Hz
菜单项	频率 > 起始频率
举例	:FREQUency:STARt 100 Hz

3.5.1.3 终止频率 ([:SENSe]:FREQUency:STOP)

命令格式	[:SENSe]:FREQUency:STOP <freq> [:SENSe]:FREQUency:STOP?
说明	设置终止频率 获取当前终止频率
参数类型	浮点型, 单位 Hz、KHz、MHz、GHz
参数范围	100 Hz~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz) 零扫宽情况下为 0~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型, 单位 Hz
默认值	3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
菜单项	频率 > 终止频率
举例	:FREQUency:STOP 1.0 GHz

3.5.1.4 中心频率步进

([:SENSe]:FREQUency:CENTer:STEP[:INCRement])

命令格式	[:SENSe]:FREQUency:CENTer:STEP[:INCRement] <freq> [:SENSe]:FREQUency:CENTer:STEP[:INCRement]?
说明	设置中心频率步进 获取中心频率步进
参数类型	浮点型，单位 Hz、KHz、MHz、GHz
参数范围	1 Hz~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型，单位 Hz
默认值	320 MHz (300 MHz、210 MHz、180 MHz、150 MHz、100 MHz)
菜单项	频率 > 中频步长
举例	:FREQUency:CENTer:STEP 2 MHz

3.5.1.5 中心频率步进工作模式

([:SENSe]:FREQUency:CENTer:STEP:AUTO)

命令格式	[:SENSe]:FREQUency:CENTer:STEP:AUTO OFF ON 0 1 [:SENSe]:FREQUency:CENTer:STEP:AUTO?
说明	开关中心频率步进自动模式 获取中心频率步进自动模式状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	频率 > 中频步长
举例	:FREQUency:CENTer:STEP:AUTO OFF

3.5.1.6 设置步进为中心频率值

([:SENSe]:FREQUency:CENTer:SET:STEP)

命令格式	[:SENSe]:FREQUency:CENTer:SET:STEP
说明	将频率步进值设置与中心频率相同
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	频率 > 中频→步进
举例	:FREQUency:CENTer:SET:STEP

3.5.1.7 扫宽 ([:SENSe]:FREQUency:SPAN)

命令格式	[:SENSe]:FREQUency:SPAN <freq> [:SENSe]:FREQUency:SPAN?
------	--

说 明	设置扫宽大小 获取扫宽大小
参数类型	浮点型，单位 Hz、KHz、MHz、GHz
参数范围	0 Hz, 100 Hz ~ 3.2GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型，单位 Hz
默认值	3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
菜单项	扫宽设置 > 扫宽
举例	:FREQUency:SPAN 1 GHz

3.5.1.8 全扫宽 ([:SENSE]:FREQUency:SPAN:FULL)

命令格式	[:SENSE]:FREQUency:SPAN:FULL
说 明	设置全扫宽工作模式
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	扫宽设置 > 全扫宽
举例	:FREQUency:SPAN:FULL

3.5.1.9 零扫宽 ([:SENSE]:FREQUency:SPAN:ZERO)

命令格式	[:SENSE]:FREQUency:SPAN:ZERO
说 明	设置零扫宽工作模式
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	扫宽设置 > 零扫宽
举例	:FREQUency:SPAN:ZERO

3.5.1.10 上次扫宽 ([:SENSE]:FREQUency:SPAN:PREVIOUS)

命令格式	[:SENSE]:FREQUency:SPAN:PREVIOUS
说 明	设置扫宽大小为上次扫宽
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	扫宽设置 > 上次扫宽
举例	:FREQUency:SPAN:PREVIOUS

3.5.1.11 扫宽减半 ([:SENSE]:FREQUency:SPAN:DOUBLE)

命令格式	[:SENSE]:FREQUency:SPAN:DOUBLE
------	--------------------------------

说明	设置扫宽为当前扫宽的一半
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	扫宽设置 > 放大
举例	:FREQUency:SPAN:DOUBle

3.5.1.12 扫宽加倍 ([:SENSe]:FREQUency:SPAN:HALF)

命令格式	[:SENSe]:FREQUency:SPAN:HALF
说明	设置扫宽为当前扫宽的两倍
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	扫宽设置 > 缩小
举例	:FREQUency:SPAN:HALF

3.5.2 自动调谐

3.5.2.1 自动调谐 ([:SENSe]:FREQUency:TUNE:IMMediate)

命令格式	[:SENSe]:FREQUency:TUNE:IMMediate
说明	自动调谐
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	自动调谐
举例	:FREQUency:TUNE:IMMediate

3.5.3 幅度控制

3.5.3.1 参考电平

(:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel)

命令格式	:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel <value> :DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel?
说明	设置参考电平 获取参考电平
参数类型	浮点型, 单位 dBm、dBmV、dBuV、V、W
参数范围	单位为 dBm 时: -100 dBm ~ 30 dBm 单位为 dBmV 时: -53.01 dBmV ~ 76.99 dBmV、 单位为 dBuV 时: 6.99 dBuV ~ 136.99 dBuV、

	单位为 Volts 时：2.24 uV ~ 7.07 V 单位为 Watts 时：100 fW ~ 1 W
返回值	浮点型，单位 dBm
默认值	0 dBm
菜单项	幅度 > 参考电平
举例	:DISPlay:WINDow:TRACe:Y:RLEVel 20 DBM

3.5.3.2 输入衰减 ([:SENSe]:POWER[:RF]:ATTenuation)

命令格式	[:SENSe]:POWER[:RF]:ATTenuation <value> [:SENSe]:POWER[:RF]:ATTenuation?
说 明	设置衰减值 获取衰减值
参数类型	整型
参数范围	0 dB ~ 50 dB
返回值	整型数据，单位 dB
默认值	20 dB
菜单项	幅度 > 衰减
举例	:POWER:ATTenuation 10

3.5.3.3 输入衰减自动模式

([:SENSe]:POWER[:RF]:ATTenuation:AUTO)

命令格式	[:SENSe]:POWER[:RF]:ATTenuation:AUTO OFF ON 0 1 [:SENSe]:POWER[:RF]:ATTenuation:AUTO?
说 明	开关衰减自动配置模式 获取衰减自动配置模式状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	幅度 > 衰减
举例	:POWER:ATTenuation:AUTO?

3.5.3.4 预置放大开关 ([:SENSe]:POWER[:RF]:GAIN[:STATe])

命令格式	[:SENSe]:POWER[:RF]:GAIN[:STATe] OFF ON 0 1 [:SENSe]:POWER[:RF]:GAIN[:STATe]?
说 明	开关频谱分析仪内部的预置放大器 获取预置放大器开关状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	幅度 > 预放
举例	:POWER:GAIN ON

3.5.3.5 幅度偏移

(:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet)

命令格式	:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet <value> :DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet?
说 明	设置频率偏移量 获取频率偏移量
参数类型	浮点型
参数范围	-300dB 到 300dB
返回值	浮点型, 单位 dB
默认值	0dB
菜单项	幅度 > 幅度偏移
举例	:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet 2

3.5.3.6 刻度单位 (:UNIT:POWer)

命令格式	:UNIT:POWer DBM DBMV DBUV V W :UNIT:POWer?
说 明	设置幅度的显示单位 获取幅度的显示单位
参数类型	枚举
参数范围	DBM DBMV DBUV V W,
返回值	枚举
默认值	DBM
菜单项	幅度 > 单位
举例	:UNIT:POWer DBMV

3.5.3.7 刻度类型

(:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing)

命令格式	:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing LINear LOGarithmic :DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing?
说 明	设置刻度显示类型 获取刻度显示类型
参数类型	枚举
参数范围	LINear: 线显 LOGarithmic: 对数
返回值	枚举
默认值	LOGarithmic
菜单项	幅度 > 刻度类型
举例	:DISPlay:WINDow:TRACe:Y:SPACing LINear

3.5.3.8 刻度/每格

(:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision)

命令格式	:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <integer> :DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?
说明	设置迹线对数显示时的刻度 获取迹线对数显示时的刻度
参数类型	浮点型
参数范围	1 dB ~ 10 dB
返回值	浮点数, 单位 dB
默认值	10 dB
菜单项	幅度 > 刻度/格
举例	:DISPlay:WINDow:TRACe:Y:PDIVision 10 dB

3.5.3.9 修正关 ([:SENSe]:CORRection:OFF)

命令格式	[:SENSe]:CORRection:OFF
说明	关闭修正功能 注意: 关闭修正功能, 会关闭修正 1-4
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	无
举例	:SENSe:CORRection:OFF

3.5.3.10 修正应用状态

([:SENSe]:CORRection:CSET:ALL[:STATe])

命令格式	[:SENSe]:CORRection:CSET:ALL[:STATe] OFF ON 0 1 [:SENSe]:CORRection:CSET:ALL[:STATe]?
说明	打开或关闭所有修正应用 获取修正应用打开状态 注意: 此开关不影响修正 1-4 的开关状态。如果修正 1-4 都关闭, 状态设为 ON 无效
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	幅度 > 修正 > 应用修正
举例	:SENSe:CORRection:CSET:ALL:STATe OFF

3.5.3.11 关闭修正 X

([:SENSe]:CORRection:CSET[1]|2|3|4:OFF)

命令格式	[:SENSe]:CORRection:CSET[1] 2 3 4:OFF [:SENSe]:CORRection:CSET[1] 2 3 4[:STATe]?
说 明	关闭修正 X 获取修正 X 开关状态
参数类型	无
参数范围	无
返回值	0 1
默认值	OFF
菜单项	幅度 > 修正 > 修正 X
举例	:CORRection:CSET2:OFF

3.5.3.12 修正 X 数据

([:SENSe]:CORRection:CSET[1]|2|3|4:DATA)

命令格式	[:SENSe]:CORRection:CSET[1] 2 3 4:DATA <x1,y1,x2,y2;...> [:SENSe]:CORRection:CSET[1] 2 3 4:DATA?
说 明	设置修正数据 查询修正数据
参数类型	修正数据字符串
参数范围	无
返回值	修正数据字符串
默认值	无
菜单项	无
举例	:CORRection:CSET2:DATA?

3.5.3.13 选择修正选项 ([:SENSe]:CORRection:SElect)

命令格式	[:SENSe]:CORRection:SElect 1 2 3 4 [:SENSe]:CORRection:SElect?
说 明	设置修正选项 查询修正选项
参数类型	枚举
参数范围	1 2 3 4
返回值	修正数据字符串
默认值	1
菜单项	幅度 > 修正 > 修正 1 2 3 4
举例	:CORRection:SElect 1

3.5.3.14 加载修正

(:MMEMory:LOAD:CORRection:CSET[1]|2|3|4)

命令格式	:MMEMory:LOAD:CORRection:CSET[1] 2 3 4 <name.COR>
说 明	加载修正文件

	注意：要事先修正文件放在频谱分析仪主目录
参数类型	字符串（文件名）
参数范围	无
返回值	无
默认值	无
菜单项	幅度 > 修正 > 修正 1 2 3 4 > 加载数据
举例	:MMEMory:LOAD:CORRection:CSET1 "oldname.COR"

3.5.3.15 输入阻抗

([:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude])

命令格式	[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] OHM50 OHM75 [:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]?
说 明	设置输入阻抗 查询输入阻抗
参数类型	枚举
参数范围	OHM50 OHM75
返回值	OHM50 OHM75
默认值	无
菜单项	幅度 > 修正
举例	CORRection:IMPedance?

3.5.4 带宽控制

3.5.4.1 分辨率带宽 ([:SENSe]:BWIDth[:RESolution])

命令格式	[:SENSe]:BWIDth[:RESolution] <freq> [:SENSe]:BWIDth[:RESolution]?
说 明	设置分辨率带宽 获取分辨率带宽
参数类型	离散型
参数范围	10 Hz、30 Hz、100 Hz、300 Hz、1 KHz、3 KHz、10 KHz、30 KHz、100 KHz、300 KHz、1 MHz
返回值	浮点型，单位 Hz
默认值	1 MHz
菜单项	BW > 分辨率带宽
举例	:BWIDth 1 KHz

3.5.4.2 分辨率带宽模式

([:SENSe]:BWIDth[:RESolution]:AUTO)

命令格式	[:SENSe]:BWIDth[:RESolution]:AUTO OFF ON 0 1 [:SENSe]:BWIDth[:RESolution]:AUTO?
说 明	设置分辨率带宽的自动模式开关 获取分辨率带宽的自动模式开关状态

参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	BW > 分辨率带宽
举例	:BWID:AUTO ON

3.5.4.3 视频带宽（[:SENSe]:BWIDth:VIDeo）

命令格式	[:SENSe]:BWIDth:VIDeo <freq> [:SENSe]:BWIDth:VIDeo?
说明	设置视频带宽 获取视频带宽
参数类型	离散型
参数范围	1 Hz、3 Hz、10 Hz、30 Hz、100 Hz、300 Hz、1 KHz、3 KHz、10 KHz、30 KHz、100 KHz、300 KHz、1 MHz
返回值	浮点型，单位 Hz
默认值	1 MHz
菜单项	BW > 视屏带宽
举例	:BWIDth:VIDeo 10 KHZ

3.5.4.4 视频带宽模式（[:SENSe]:BWIDth:VIDeo:AUTO）

命令格式	[:SENSe]:BWIDth:VIDeo:AUTO OFF ON 0 1 [:SENSe]:BWIDth:VIDeo:AUTO?
说明	设置视频带宽的自动模式开关 获取视频带宽的自动模式开关状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	BW > 视屏带宽
举例	BWIDth:VIDeo:AUTO OFF

3.5.4.5 视分比（[:SENSe]:BWIDth:VIDeo:RATio）

命令格式	[:SENSe]:BWIDth:VIDeo:RATio <number> [:SENSe]:BWIDth:VIDeo:RATio?
说明	设置视分比 获取视分比
参数类型	离散型、浮点型
参数范围	0.001、0.003、0.01、0.03、0.1、0.3、1.0、3.0、10.0、30.0、100.0、300.0、1000.0
返回值	浮点型
默认值	1.0
菜单项	BW > 视分比

举例	:BWIDth:VIDeo:RATio 30
----	------------------------

3.5.4.6 视分比状态

([:SENSe]:BWIDth:VIDeo:RATio:CONfig?)

命令格式	[:SENSe]:BWIDth:VIDeo:RATio:CONfig?
说明	获取设置视分比状态
参数类型	无
参数范围	无
返回值	0 1
默认值	1
菜单项	无
举例	:BWIDth:VIDeo:RATio:CONfig?

3.5.4.7 设置滤波器 ([:SENSe]:FILTer:TYPE)

命令格式	[:SENSe]:FILTer:TYPE EMI GAUSS [:SENSe]:FILTer:TYPE?
说明	设置滤波器类型 获取滤波器类型
参数类型	枚举
参数范围	EMI: EMI 滤波器 GAUSS: 高斯
返回值	枚举
默认值	GAUSS
菜单项	BW > 滤波器
举例	:FILTer:TYPE EMI

3.5.5 迹线控制

3.5.5.1 迹线显示模式 (:TRACe[1]|2|3|4:MODE)

命令格式	:TRACe[1] 2 3 4:MODE WRITe MAXHold MINHold VIEW BLANk AVERAge :TRACe[1] 2 3 4:MODE?
说明	设置迹线的显示模式 获取迹线的显示模式
参数类型	枚举
参数范围	WRITe: 迹线为正常模式, 更新数据 MAXHold: 显示迹线的最大值 MINHold: 显示迹线的最小值 VIEW: 停止更新迹线, 显示当前的迹线数据 BLANk: 清空迹线数据 AVERAge: 平均
返回值	枚举
默认值	迹线 1 为 WRITe, 迹线 2、3、4 为 BLANk

菜单项	迹线
举例	:TRAC1:MODE VIEW

3.5.5.2 查询迹线数据 (:TRACe[:DATA]?)

命令格式	:TRACe[:DATA]? 1 2 3 4
说明	获取迹线的数据
参数类型	枚举
参数范围	1 2 3 4
返回值	字符串
默认值	1
菜单项	无
举例	:TRACe:DATA? 1

3.5.5.3 查询迹线扫描完成状态 (:TRACe:SWEEp:STATe?)

命令格式	:TRACe:SWEEp:STATe?
说明	查询迹线扫描完成状态
参数类型	无
参数范围	无
返回值	布尔，完成返回 1，扫描中返回 0
默认值	无
菜单项	无
举例	TRACe:SWEEp:STATe?

3.5.5.4 迹线数据格式 (:FORMAt[:TRACe][:DATA])

命令格式	:FORMAt[:TRACe][:DATA] ASCii REAL :FORMAt[:TRACe][:DATA]?
说明	设置读取的迹线数据格式 询问读取的迹线数据格式
参数类型	枚举
参数范围	ASCii REAL
返回值	字符串
默认值	REAL
菜单项	无
举例	:FORMAt ASCii

3.5.5.5 迹线计算类型 (:TRACe:MATH:TYPE)

命令格式	:TRACe:MATH:TYPE Off X-Y+Ref->Z Y-X+Ref->Z X+Y-Ref->Z X+Const->Z X-Const->Z :TRACe:MATH:TYPE?
说明	设置迹线计算类型 询问迹线计算类型 此处命令参数小写不可忽略，即 X-Y+Ref->Z 输入时不可写为 X-Y+R->Z

参数类型	枚举
参数范围	Off: 迹线数学关 X-Y+Ref->Z: 迹线变量 X 减去迹线变量 Y 加上参考电平输出到输出迹线上 Y-X+Ref->Z: 迹线变量 Y 减去迹线变量 X 加上参考电平输出到输出迹线上 X+Y-Ref->Z: 迹线变量 X 加上迹线变量 Y 减去参考电平输出到输出迹线上 X+Const->Z: 迹线变量 X 加上常量输出到输出迹线上 X-Const->Z: 迹线变量 X 减去常量输出到输出迹线上
返回值	枚举
默认值	Off
菜单项	迹线 > 计算类型
举例	:TRACe:MATH:TYPE X-Y+Ref->Z

3.5.5.6 迹线数学变量 X (:TRACe:MATH:X)

命令格式	:TRACe:MATH:X A B C :TRACe:MATH:X?
说明	设置变量 X 迹线 询问变量 X 迹线
参数类型	枚举
参数范围	A B C
返回值	枚举
默认值	A
菜单项	迹线 > 数学 > 变量 X
举例	:TRACe:MATH:X A

3.5.5.7 迹线数学变量 Y (:TRACe:MATH:Y)

命令格式	:TRACe:MATH:Y A B C :TRACe:MATH:Y?
说明	设置变量 Y 迹线 询问变量 Y 迹线
参数类型	枚举
参数范围	A B C
返回值	枚举
默认值	B
菜单项	迹线 > 数学 > 变量 Y
举例	:TRACe:MATH:Y A

3.5.5.8 迹线数学输出 (:TRACe:MATH:Z)

命令格式	:TRACe:MATH:Z A B C :TRACe:MATH:Z?
说明	设置输出迹线

	询问输出迹线
参数类型	枚举
参数范围	A B C
返回值	枚举
默认值	C
菜单项	迹线 > 数学 > 输出
举例	:TRACe:MATH:Z A

3.5.5.9 迹线数学常量 (:TRACe:MATH:CONST)

命令格式	:TRACe:MATH:CONST <const> :TRACe:MATH:CONST?
说明	设置迹线数学常量 询问迹线数学常量
参数类型	枚举
参数范围	-300dB ~300 dB
返回值	浮点型
默认值	0.00dB
菜单项	迹线 > 数学 > 常量
举例	:TRACe:MATH:CONST 7

3.5.6 检波控制

3.5.6.1 检波类型

([:SENSe]:DETector:TRACe[1]|2|3|4[:FUNction])

命令格式	[:SENSe]:DETector:TRACe[1] 2 3 4[:FUNction] NEGative POSitive SAMPlE AVERAge NORMAL QUASi [:SENSe]:DETector:TRACe[1] 2 3 4[:FUNction]?
说明	设置迹线检波类型 获取迹线检波类型
参数类型	枚举
参数范围	NEGative: 负峰值 POSitive: 正峰值 SAMPlE: 采样 AVERAge: 平均 NORMAL: 标准 QUASi: 准峰值, 只能在 EMI 打开的情况下打开
返回值	枚举
默认值	POSitive
菜单项	检波
举例	:DETector:TRAC1 AVERAge

3.5.7 平均控制

3.5.7.1 平均类型（[:SENSe]:AVERAge:TYPE）

命令格式	[:SENSe]:AVERAge:TYPE LOGPower POWER VOLTage [:SENSe]:AVERAge:TYPE?
说明	设置平均类型 获取平均类型
参数类型	枚举
参数范围	LOGPower: 对数平均 POWER: 功率平均 VOLTage: 电压平均
返回值	枚举
默认值	LOGPower
菜单项	BW > 平均类型
举例	AVERAge:TYPE VOLTage

3.5.7.2 平均次数

（[:SENSe]:AVERAge:TRACe[1]|2|3|4:COUNT）

命令格式	[:SENSe]:AVERAge:TRACe[1] 2 3 4:COUNT <integer> [:SENSe]:AVERAge:TRACe[1] 2 3 4:COUNT?
说明	设置迹线的平均次数 获取迹线的平均次数
参数类型	整型
参数范围	1 ~ 999
返回值	整型
默认值	1
菜单项	迹线 > 平均次数
举例	:AVERAge:TRACe1:COUNT 10

3.5.7.3 重新平均

（[:SENSe]:AVERAge:TRACe[1]|2|3|4:CLEAr）

命令格式	[:SENSe]:AVERAge:TRACe[1] 2 3 4:CLEAr
说明	重新开始平均
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	无
举例	:AVERAge:TRAC1:CLEAr

3.5.8 扫描控制

3.5.8.1 扫描模式（[:SENSe]:SWEep:MODE）

命令格式	[:SENSe]:SWEep:MODE AUTO FFT SWEep [:SENSe]:SWEep:MODE?
说明	设置扫描模式 获取扫描模式
参数类型	枚举
参数范围	AUTO: 自动切换 FFT: FFT SWEep: 扫描
返回值	枚举
默认值	SWEep
菜单项	扫描
举例	:SWEep:MODE SWEep

3.5.8.2 扫描时间（[:SENSe]:SWEep:TIME）

命令格式	[:SENSe]:SWEep:TIME <time> [:SENSe]:SWEep:TIME?
说明	设置扫描时间 获取扫描时间
参数类型	浮点型，单位 ks、s、ms、us
参数范围	900us ~ 3ks（准峰值检波时：900us ~ 30ks）
返回值	浮点型，单位 s
默认值	312.416ms（216.288ms，192.256ms，168.224ms， 120.160ms）
菜单项	扫描 > 扫描时间
举例	:SWEep:TIME 5s

3.5.8.3 扫描时间模式（[:SENSe]:SWEep:TIME:AUTO）

命令格式	[:SENSe]:SWEep:TIME:AUTO OFF ON 0 1 [:SENSe]:SWEep:TIME:AUTO?
说明	设置扫描时间自动模式 获取扫描时间自动模式
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	扫描 > 扫描时间
举例	:SWEep:TIME:AUTO ON

3.5.8.4 扫描时间规则（[:SENSe]:SWEep:SPEed）

命令格式	[:SENSe]:SWEep:SPEed NORMAl ACCURacy
------	--------------------------------------

	<code>[[:SENSe]:SWEep:SPEed?</code>
说明	设置扫描时间规则 获取扫描时间规则
参数类型	枚举
参数范围	ACCUracy: 精确 NORMal: 快速
返回值	枚举
默认值	NORMal
菜单项	扫描 > 扫描时间规则
举例	<code>:SWEep:SPEed NORMal</code>

3.5.8.5 扫描次数 (`[[:SENSe]:SWEep:COUNT)`

命令格式	<code>[[:SENSe]:SWEep:COUNT <integer></code> <code>[[:SENSe]:SWEep:COUNT?</code>
说明	设置单次扫描时的扫描次数 读取单次扫描时的扫描次数
参数类型	整型
参数范围	1 ~ 99999
返回值	整型
默认值	1
菜单项	扫描 > 次数
举例	<code>:SWEep:COUNT 10</code>

3.5.8.6 QPD 时间 (`[[:SENSe]:QPD:DWELI:TIME)`

命令格式	<code>[[:SENSe]:QPD:DWELI:TIME <time ></code> <code>[[:SENSe]:QPD:DWELI:TIME?</code>
说明	设置 QPD 时间 获取 QPD 时间
参数类型	浮点型, 单位 s、ms、us
参数范围	0us ~ 10s (准峰值检波时: 900us ~ 30ks)
返回值	浮点型, 单位 s
默认值	500ms
菜单项	扫描 > 驻留时间
举例	<code>:QPD:DWELI:TIME 10 s</code>

3.5.9 显示控制

3.5.9.1 网格亮度

(`:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness)`

命令格式	<code>:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness <value></code> <code>:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness?</code>
说明	设置波形区域网格亮度 获取波形区域网格亮度

参数类型	整型
参数范围	0 ~ 100
返回值	浮点型
默认值	30%
菜单项	显示 > 网格亮度
举例	:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGHtness 50

3.5.9.2 显示线开关

(: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?
说明	开关显示线 获取显示线开关状态
参数类型	枚举
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	显示 > 显示线
举例	:DISPlay:WINDow:TRACe:Y:DLINe:STATe ON

3.5.9.3 显示线 (:DISPlay:WINDow:TRACe:Y:DLINe)

命令格式	:DISPlay:WINDow:TRACe:Y:DLINe <value> :DISPlay:WINDow:TRACe:Y:DLINe?
说明	设置显示线幅度 获取显示线幅度
参数类型	浮点型, 单位 dBm
参数范围	参考电平~参考电平-100 dBm
返回值	浮点型, 单位 dBm
默认值	0 dBm
菜单项	显示 > 显示线
举例	:DISPlay:WINDow:TRACe:Y:DLINe -10

3.6 Calculate 命令系统

3.6.1 Marker

3.6.1.1 开关光标 (:CALCulate:MARKer[1]|2|3|4:STATe)

命令格式	:CALCulate:MARKer[1] 2 3 4:STATe OFF ON 0 1 :CALCulate:MARKer[1] 2 3 4:STATe?
说明	设置光标开关状态 获取光标开关状态
参数类型	布尔型

参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	光标
举例	:CALCulate:MARK1:STATe ON

3.6.1.2 光标全关（:CALCulate:MARKer:AOff）

命令格式	:CALCulate:MARKer:AOff
说明	关闭所有光标
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	无
举例	:CALCulate:MARKer:AOff

3.6.1.3 光标模式（:CALCulate:MARKer[1]|2|3|4:MODE）

命令格式	:CALCulate:MARKer[1] 2 3 4:MODE POSition DELTA BAND OFF :CALCulate:MARKer[1] 2 3 4:MODE?
说明	设置光标模式 获取光标模式
参数类型	枚举
参数范围	POSition: 正常 DELTA: 差值 BAND: 差值对 OFF: 关闭
返回值	枚举
默认值	OFF
菜单项	光标
举例	:CALCulate:MARK1:MODE POSition

3.6.1.4 光标标记迹线（:CALCulate:MARKer[1]|2|3|4:TRACe）

命令格式	:CALCulate:MARKer[1] 2 3 4:TRACe 1 2 3 4 :CALCulate:MARKer[1] 2 3 4:TRACe?
说明	设置光标标记迹线 读取光标标记迹线
参数类型	枚举
参数范围	1 2 3 4
返回值	枚举
默认值	1
菜单项	光标 > 标记迹线
举例	CALCulate:MARK:TRAC 1

3.6.1.5 光标相对于

(:CALCulate:MARKer[1]|2|3|4:RELative:TO)

命令格式	:CALCulate:MARKer[1] 2 3 4:RELative:TO:MARKer 1 2 3 4 :CALCulate:MARKer[1] 2 3 4:RELative:TO:MARKer?
说明	设置光标相对于 获取光标相对于
参数类型	枚举
参数范围	1 2 3 4
返回值	枚举
默认值	1
菜单项	Marker > Relative To
举例	:CALCulate:MARKer1:RELative:TO:MARK 3

3.6.1.6 光标点 X 轴的值 (:CALCulate:MARKer[1]|2|3|4:X)

命令格式	:CALCulate:MARKer[1] 2 3 4:X <para> :CALCulate:MARKer[1] 2 3 4:X?
说明	设置光标点 X 轴的值 读取光标点 X 轴的值 此命令只在光标模式非 OFF 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:STATE :CALCulate:MARKer[1] 2 3 4:MODE 在光标读出类型为频率时，参数取值为频率； 在光标读出类型为时间时，参数取值为时间；参考命令： :CALCulate:MARKer[1] 2 3 4:X:READout
参数类型	频率，浮点型，单位 Hz、KHz、MHz、GHz，默认 Hz 或 时间，浮点型，单位 us、ms、s、ks，默认 s
参数范围	0 Hz ~ 3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz) 或 10 ms ~ 1000 s
返回值	在光标读出类型为频率时，读数为频率，浮点型，单位 Hz； 在光标读出类型为时间时，读数为时间，浮点型，单位 s； 在光标读出类型为周期时，读数为周期，浮点型，单位 s；
默认值	1.6 GHz (1.5 GHz、1.05 GHz、0.9 GHz、0.75 GHz、0.5 GHz) 或 312.64 ms
菜单项	光标 > 正常
举例	:CALCulate:MARKer4:X 0.4 GHz :CALCulate:MARKer4:X 200 ms :CALCulate:MARKer4:X?

3.6.1.7 参考点 X 轴的值

(:CALCulate:MARKer[1]|2|3|4:X:REFerence)

命令格式	:CALCulate:MARKer[1] 2 3 4:X:REFerence <para> :CALCulate:MARKer[1] 2 3 4:X:REFerenc?
------	---

说明	设置参考点光标 X 轴坐标值 读取参考点光标 X 轴坐标值 此命令只在光标模式为 DELTaN BAND 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:MODE 在光标读出类型为频率时，参数取值为频率； 在光标读出类型为时间时，参数取值为时间；参考命令： :CALCulate:MARKer[1] 2 3 4:X:READout
参数类型	频率，浮点型，单位 Hz、KHz、MHz、GHz，默认 Hz 或 时间，浮点型，单位 us、ms、s、ks，默认 s
参数范围	0 Hz ~ 3.2 GHz（3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz）或 10 ms ~ 1000 s
返回值	在光标读出类型为频率时，读数为频率，浮点型，单位 Hz； 在光标读出类型为时间时，读数为时间，浮点型，单位 s； 在光标读出类型为周期时，读数为周期，浮点型，单位 s；
默认值	1.6 GHz（1.5 GHz、1.05 GHz、0.9 GHz、0.75 GHz、0.5 GHz） 或 312.64 ms
菜单项	光标 > 差值对
举例	:CALCulate:MARKer1:X:REFerence 1.6 GHz

3.6.1.8 光标对 X 轴的差值

(:CALCulate:MARKer[1]|2|3|4:X:DELTA)

命令格式	:CALCulate:MARKer[1] 2 3 4:X:DELTA <para> :CALCulate:MARKer[1] 2 3 4:X:DELTA?
说明	设置光标对 X 轴光标点值减去参考点值的差 读取光标对 X 轴光标点值减去参考点值的差 此命令只在光标模式为 DELTaN BAND 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:MODE 在光标读出类型为频率时，参数取值为频率； 在光标读出类型为时间时，参数取值为时间；参考命令： :CALCulate:MARKer[1] 2 3 4:X:READout
参数类型	频率，浮点型，单位 Hz、KHz、MHz、GHz，默认 Hz 或 时间，浮点型，单位 us、ms、s、ks，默认 s
参数范围	0 Hz ~ 3.2 GHz（3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz）或 10 ms ~ 1000 s
返回值	在光标读出类型为频率时，读数为频率，浮点型，单位 Hz； 在光标读出类型为时间时，读数为时间，浮点型，单位 s； 在光标读出类型为周期时，读数为周期，浮点型，单位 s；
默认值	0 Hz 或 0 s
菜单项	光标 > 差值
举例	:CALCulate:MARKer2:X:DELTA 1.6 GHz

3.6.1.9 光标对中心点 X 轴的值 (:CALCulate:MARKer[1]|2|3|4:X:CENTer)

命令格式	:CALCulate:MARKer[1] 2 3 4:X:CENTer <para> :CALCulate:MARKer[1] 2 3 4:X:CENTer?
说明	设置光标对中心点 X 轴的值 读取光标对中心点 X 轴的值 此命令只在光标模式为 DELTA BAND 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:MODE 在光标读出类型为频率时，参数取值为频率； 在光标读出类型为时间时，参数取值为时间；参考命令： :CALCulate:MARKer[1] 2 3 4:X:READout
参数类型	频率，浮点型，单位 Hz、KHz、MHz、GHz，默认 Hz 或 时间，浮点型，单位 us、ms、s、ks，默认 s
参数范围	0 Hz ~ 3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz) 或 10 ms ~ 1000 s
返回值	在光标读出类型为频率时，读数为频率，浮点型，单位 Hz； 在光标读出类型为时间时，读数为时间，浮点型，单位 s； 在光标读出类型为周期时，读数为周期，浮点型，单位 s；
默认值	1.6 GHz (1.5 GHz、1.05 GHz、0.9 GHz、0.75 GHz、0.5 GHz) 或 10 ms ~ 1000 s
菜单项	光标 > 差值对 > 跨度
举例	:CALCulate:MARKer3:X:CENTer 1.6 GHz

3.6.1.10 光标对 X 轴的跨度值 (:CALCulate:MARKer[1]|2|3|4:X:SPAN)

命令格式	:CALCulate:MARKer[1] 2 3 4:X:SPAN <para> :CALCulate:MARKer[1] 2 3 4:X:SPAN?
说明	设置光标对 X 轴的跨度值 读取光标对 X 轴的跨度值 此命令只在光标模式为 DELTA BAND 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:MODE 在光标读出类型为频率时，参数取值为频率； 在光标读出类型为时间时，参数取值为时间；参考命令： :CALCulate:MARKer[1] 2 3 4:X:READout
参数类型	频率，浮点型，单位 Hz、KHz、MHz、GHz，默认 Hz 或 时间，浮点型，单位 us、ms、s、ks，默认 s
参数范围	0 Hz ~ 3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz) 或 10 ms ~ 1000 s
返回值	在光标读出类型为频率时，读数为频率，浮点型，单位 Hz； 在光标读出类型为时间时，读数为时间，浮点型，单位 s； 在光标读出类型为周期时，读数为周期，浮点型，单位 s；

默认值	0 Hz 或 0 s
菜单项	光标 > 差值对 > 跨度
举例	:CALCulate:MARKer4:X:SPAN 2 GHz

3.6.1.11 光标点 Y 轴的值 (:CALCulate:MARKer[1]|2|3|4:Y?)

命令格式	:CALCulate:MARKer[1] 2 3 4:Y?
说明	读取光标点 Y 轴的值，同时可用于读取光标功能中的光标噪声 执行此命令需确保光标已经处于非 OFF 状态，参考命令： :CALCulate:MARKer[1] 2 3 4:STATE :CALCulate:MARKer[1] 2 3 4:MODE
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	光标 > 正常
举例	:CALCulate:MARKer1:Y? Return: -25

3.6.1.12 光标对参考点 Y 轴值

(:CALCulate:MARKer[1]|2|3|4:Y:REFEreence?)

命令格式	:CALCulate:MARKer[1] 2 3 4:Y:REFEreence?
说明	读取光标对参考点 Y 轴的值 此命令只在光标模式为 DELTA BAND 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:MODE
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	光标 > 差值对
举例	:CALCulate:MARKer1:Y:REFEreence? Return: -25

3.6.1.13 光标对 Y 轴差值

(:CALCulate:MARKer[1]|2|3|4:Y:DELTA?)

命令格式	:CALCulate:MARKer[1] 2 3 4:Y:DELTA?
说明	读取光标对 Y 轴的差值 此命令只在光标模式为 DELTA BAND 时才能生效，参考命令： :CALCulate:MARKer[1] 2 3 4:MODE
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	光标 > 差值

举例	:CALCulate:MARKer1:Y:DELTA? Return: -25
----	--

3.6.1.14 打开光标表 (:CALCulate:MARKer:TABLE)

命令格式	:CALCulate:MARKer:TABLE ON OFF 0 1 :CALCulate:MARKer:TABLE?
说明	打开关闭光标表 查询光标表打开状态
参数类型	布尔型
参数范围	ON OFF 0 1
返回值	0 1
默认值	0
菜单项	光标 > 光标表
举例	:CALCulate:MARKer:TABLE ON

3.6.1.15 光标到起始频率

(:CALCulate:MARKer[1]|2|3|4[:SET]:START)

命令格式	:CALCulate:MARKer[1] 2 3 4[:SET]:START
说明	将光标 X 轴的值设置为起始频率 若对应光标未打开，发送此命令会自动打开光标在中心频率处。
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标 > 光标→起始
举例	:CALCulate:MARKer1:START

3.6.1.16 光标到终止频率

(:CALCulate:MARKer[1]|2|3|4[:SET]:STOP)

命令格式	:CALCulate:MARKer[1] 2 3 4[:SET]:STOP
说明	将光标 X 轴的值设置为终止频率 若对应光标未打开，发送此命令会自动打开光标在中心频率处。
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标 > 光标→终止
举例	:CALCulate:MARKer1:STOP

3.6.1.17 光标到中心频率

(:CALCulate:MARKer[1]|2|3|4[:SET]:CENTER)

命令格式	:CALCulate:MARKer[1] 2 3 4[:SET]:CENTER
------	---

说明	将光标 X 轴的值设置为中心频率 若对应光标未打开，发送此命令会自动打开光标在中心频率处。
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标 > 光标→中频
举例	:CALCulate:MARKer1:CENTer

3.6.1.18 光标到中频步进

(:CALCulate:MARKer[1]|2|3|4[:SET]:STEP)

命令格式	:CALCulate:MARKer[1] 2 3 4[:SET]:STEP
说明	将光标 X 轴的值设置为中频步进 若对应光标未打开，发送此命令会自动打开光标在中心频率处。
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标 > 光标→步进
举例	:CALCulate:MARKer1:STEP

3.6.1.19 光标到参考电平

(:CALCulate:MARKer[1]|2|3|4[:SET]:RLEVEL)

命令格式	:CALCulate:MARKer[1] 2 3 4[:SET]:RLEVEL
说明	将光标 Y 轴的值设置为参考电平 若对应光标未打开，发送此命令会自动打开光标在中心频率处。
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标 > 光标→参考
举例	:CALCulate:MARKer2:RLEVEL

3.6.1.20 光标差值到扫宽

(:CALCulate:MARKer[1]|2|3|4:DELTA[:SET]:SPAN)

命令格式	:CALCulate:MARKer[1] 2 3 4:DELTA[:SET]:SPAN
说明	将光标对 X 轴的差值设置为扫宽 此命令只在光标模式为 DELTA BAND 时才能生效，参考命令 :CALCulate:MARKer[1] 2 3 4:MODE
参数类型	无
参数范围	无
返回值	无

默认值	无
菜单项	光标 > △光标→扫宽
举例	:CALCulate:MARKer2:DELTA:SPAN

3.6.1.21 光标差值到中频

(:CALCulate:MARKer[1]|2|3|4:DELTA[:SET]:CENTER)

命令格式	:CALCulate:MARKer[1] 2 3 4:DELTA[:SET]:CENTER
说明	将光标对 X 轴的差值设置为中频 此命令只在光标模式为 DELTA BAND 时才能生效，参考命令:CALCulate:MARKer[1] 2 3 4:MODE
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标 > △光标→中频
举例	:CALCulate:MARKer3:DELTA:CENTER

3.6.1.22 光标峰值类型

(:CALCulate:MARKer:PEAK:SEARch:MODE)

命令格式	:CALCulate:MARKer:PEAK:SEARch:MODE MAXimum MINimum :CALCulate:MARKer:PEAK:SEARch:MODE?
说明	设置峰值搜索的模式 获取峰值搜索的模式
参数类型	枚举
参数范围	MAXimum: 正峰值 MINimum: 负峰值
返回值	枚举
默认值	MAXimum
菜单项	峰值 > 搜索配置 > 峰值类型
举例	:CALCulate:MARKer:PEAK:SEARch:MODE MINimum

3.6.1.23 峰值绝对阈值

(:CALCulate:MARKer:PEAK:THReshold)

命令格式	:CALCulate:MARKer:PEAK:THReshold <value> :CALCulate:MARKer:PEAK:THReshold?
说明	设置峰值搜索条件的绝对阈值 获取峰值搜索条件的绝对阈值
参数类型	浮点型，单位 dBm
参数范围	-200.0 dBm~ 200.0 dBm
返回值	浮点型，单位 dBm
默认值	-160.0 dBm
菜单项	峰值 > 搜索配置 > 峰值阈值

举例	:CALCulate:MARKer:PEAK:THReshold -50
----	--------------------------------------

3.6.1.24 峰值相对偏移

(:CALCulate:MARKer:PEAK:EXCursion)

命令格式	:CALCulate:MARKer:PEAK:EXCursion <value> :CALCulate:MARKer:PEAK:EXCursion?
说明	设置峰值搜索条件的相对门限
参数类型	浮点型, 单位 dB
参数范围	0 ~ 200.0dB
返回值	浮点型, 单位 dB
默认值	0 dB
菜单项	峰值 > 搜索配置 > 峰值偏移
举例	:CALCulate:MARKer:PEAK:EXCursion 10

3.6.1.25 打开峰值表 (:CALCulate:MARKer:PEAK:TABLE)

命令格式	:CALCulate:MARKer:PEAK:TABLE ON OFF 0 1 :CALCulate:MARKer:PEAK:TABLE?
说明	打开关闭峰值表 查询峰值表打开状态
参数类型	布尔型
参数范围	ON OFF 0 1
返回值	0 1
默认值	0
菜单项	峰值 > 峰值表
举例	:CALCulate:MARKer:PEAK:TABLE ON

3.6.1.26 获取峰值表数据 (:CALCulate:PEAK:TABLE)

命令格式	:CALCulate:PEAK:TABLE?
说明	获取峰值表数据
参数类型	无
参数范围	无
返回值	字符串
默认值	无
菜单项	峰值 > 峰值表
举例	:CALCulate:PEAK:TABLE?

3.6.1.27 光标连续峰值

(:CALCulate:MARKer[1]|2|3|4:CPEak[:STATe])

命令格式	:CALCulate:MARKer[1] 2 3 4:CPEak[:STATe] OFF ON 0 1 :CALCulate:MARKer[1] 2 3 4:CPEak[:STATe]?
说明	开关连续峰值搜索功能 获取连续峰值搜索功能开关状态

参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	无
菜单项	峰值 > 连续峰值
举例	:CALCulate:MARKer1:CPEak ON

3.6.1.28 光标峰值搜索

(:CALCulate:MARKer[1]|2|3|4:MAXimum)

命令格式	:CALCulate:MARKer[1] 2 3 4:MAXimum
说明	光标搜索峰值，并用指定的光标标记 (若峰峰值打开则进行峰峰值搜索，反之则进行单峰值搜索，参考命令 :CALCulate:MARKer[1] 2 3 4:PTPeak:STATe 搜索条件包括峰值类型、绝对阈值和相对偏移，参考如下命令 :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	峰值
举例	:CALCulate:MARKer4:MAXimum

3.6.1.29 光标搜索下一峰值

(:CALCulate:MARKer[1]|2|3|4:MAXimum:NEXT)

命令格式	:CALCulate:MARKer[1] 2 3 4:MAXimum:NEXT
说明	光标搜索下一个峰值，并用指定的光标标记 (依据已设定的搜索条件，包括峰值类型、绝对阈值和相对偏移，参考如下命令 :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	峰值 > 下一峰值
举例	:CALCulate:MARKer1:MAXimum:NEXT

3.6.1.30 光标搜索左峰值

(:CALCulate:MARKer[1]|2|3|4:MAXimum:LEFT)

命令格式	:CALCulate:MARKer[1] 2 3 4:MAXimum:LEFT
说明	光标搜索左峰值，并用指定的光标标记 (依据已设定的搜索条件，包括峰值类型、绝对阈值和相对偏移，参考如下命令 :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	峰值 > 左峰值
举例	:CALCulate:MARKer1:MAXimum:LEFT

3.6.1.31 光标搜索右峰值

(:CALCulate:MARKer[1]|2|3|4:MAXimum:RIGHT)

命令格式	:CALCulate:MARKer[1] 2 3 4:MAXimum:RIGHT
说明	光标搜索右峰值，并用指定的光标标记 (依据已设定的搜索条件，包括峰值类型、绝对阈值和相对偏移，参考如下命令 :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	峰值 > 右峰值
举例	:CALCulate:MARKer1:MAXimum:RIGHT

3.6.1.32 峰峰值搜索 (:CALCulate:MARKer[1]|2|3|4:PTPeak)

命令格式	:CALCulate:MARKer[1] 2 3 4:PTPeak
说明	执行一次峰峰搜索，并用指定光标对标记
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	峰值 > 峰峰值
举例	:CALCulate:MARKer1:PTPeak

3.6.1.33 光标功能（:CALCulate:MARKer[1]|2|3|4:FUNCTION）

命令格式	:CALCulate:MARKer[1] 2 3 4:FUNCTION OFF FCOunt NOISe NDB :CALCulate:MARKer[1] 2 3 4:FUNCTION?
说明	设置光标功能 获取光标功能
参数类型	枚举
参数范围	OFF: 正常光标 FCOunt: 频率计数 NOISe: 光标噪声 NDB: N dB 带宽
返回值	枚举
默认值	OFF
菜单项	光标功能
举例	:CALCulate:MARK1:FUNCTION FCOunt

3.6.1.34 查询频率计数器

（:CALCulate:MARKer[1]|2|3|4:FCOunt:X？）

命令格式	:CALCulate:MARKer[1] 2 3 4:FCOunt:X?
说明	读取光标频率计数器读数
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	光标功能 > 频率计数
举例	:CALCulate:MARK:FCOunt:X?

3.6.1.35 N dB 带宽读值

（:CALCulate:MARKer[1]|2|3|4:BANDwidth:RESult？）

命令格式	:CALCulate:MARKer[1] 2 3 4:BANDwidth:RESult?
说明	设置读取 N dB 带宽结果
参数类型	无
参数范围	无
返回值	浮点型
默认值	无
菜单项	光标功能 > N dB 带宽
举例	:CALCulate:MARK1:BANDwidth:RESult?

3.6.1.36 N dB 带宽参考值

（:CALCulate:MARKer[1]|2|3|4:BANDwidth[1]|2|3|4:RESult？）

命令格式	:CALCulate:MARKer[1] 2 3 4:BANDwidth:NDB <value>
------	--

	:CALCulate:MARKer[1] 2 3 4:BANDwidth:NDB?
说明	设置读取 N dB 带宽参考值
参数类型	浮点型
参数范围	-100dB ~ 100dB
返回值	浮点型
默认值	-3 dB
菜单项	光标功能 > N dB 带宽
举例	:CALCulate:MARK1:BANDwidth:NDB 10 DB

3.6.1.37 光标 X 轴读数方式

(:CALCulate:MARKer[1]|2|3|4:X:READout)

命令格式	:CALCulate:MARKer[1] 2 3 4:X:READout FREQuency TIME PERiod :CALCulate:MARKer[1] 2 3 4:X:READout?
说明	设置光标在 X 轴读数方式 读取光标在 X 轴读数方式
参数类型	枚举
参数范围	FREQuency: 频率 TIME: 时间 PERiod: 周期
返回值	枚举
默认值	FREQuency
菜单项	光标功能 > 读数
举例	:CALCulate:MARKer1:X:READout FREQuency

3.6.2 Limit

3.6.2.1 限制测试开始 (:CALCulate:LLINe:TEST:START)

命令格式	:CALCulate:LLINe:TEST:START
说明	限制测试开始
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	限制 > 测试
举例	:CALCulate:LLINe:TEST:START

3.6.2.2 限制测试停止 (:CALCulate:LLINe:TEST:STOP)

命令格式	:CALCulate:LLINe:TEST:STOP
说明	停止限制测试
参数类型	无
参数范围	无
返回值	无

默认值	无
菜单项	限制 > 测试
举例	:CALCulate:LLINe:TEST:STOP

3.6.2.3 读取限制测试状态 (:CALCulate:LLINe:TEST:STATE?)

命令格式	:CALCulate:LLINe:TEST:STATE?
说明	读取限制测试状态
参数类型	无
参数范围	无
返回值	0 1
默认值	OFF
菜单项	限制 > 测试
举例	:CALCulate:LLINe:TEST:STAT?

3.6.2.4 限制线状态 (:CALCulate:LLINe[1]|2:STATE)

命令格式	:CALCulate:LLINe[1] 2:STATE OFF ON 0 1 :CALCulate:LLINe[1] 2:STATE?
说明	设置限制状态 获取限制状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	限制 > 限制
举例	:CALCulate:LLINe1:STATE OFF

3.6.2.5 限制类型 (:CALCulate:LLINe[1]|2:TYPE)

命令格式	:CALCulate:LLINe[1] 2:TYPE UPPer LOWer :CALCulate:LLINe[1] 2:TYPE?
说明	设置限制类型 获取限制类型
参数类型	枚举
参数范围	UPPer LOWer
返回值	枚举
默认值	LINE1 默认 UPPer, LINE2 默认 LOWer
菜单项	限制 > 编辑
举例	:CALCulate:LLINe1:TYPE LOWer

3.6.2.6 限制模式 (:CALCulate:LLINe[1]|2:MODE)

命令格式	:CALCulate:LLINe[1] 2:MODE LINE POINT :CALCulate:LLINe[1] 2:MODE?
说明	设置限制模式 获取限制模式

参数类型	枚举
参数范围	LINE POINT
返回值	枚举
默认值	LINE
菜单项	限制 > 编辑
举例	:CALCulate:LLINe1:MODE POINT

3.6.2.7 限制线幅值 (:CALCulate:LLINe[1]|2:Y)

命令格式	:CALCulate:LLINe[1] 2:Y <value> :CALCulate:LLINe[1] 2:Y?
说明	设置限制线幅值 获取限制线幅值
参数类型	浮点型
参数范围	-400 dBm~330 dBm
返回值	浮点型
默认值	0dBm
菜单项	限制 > 编辑 > 幅度
举例	:CALCulate:LLINe1:Y 5dBm

3.6.2.8 限制规则数据 (:CALCulate:LLINe[1]|2:DATA)

命令格式	:CALCulate:LLINe[1] 2:DATA <x-axis>,<ampl>{,<x-axis>, <ampl>} :CALCulate:LLINe[1] 2:DATA?
说明	设置限制规则数据 获取限制规则数据
参数类型	X 轴: 浮点型 幅度: 浮点型
参数范围	X 轴: 0~3.2GHz 幅度: -400 dBm~330 dBm
返回值	X 轴: 浮点型 幅度: 浮点型
默认值	X 轴: -1Hz 幅度: 0 dBm
菜单项	限制 > 编辑
举例	:CALC:LLINe1:DATA 10000000,-20,20000000,-30

3.6.2.9 增加限制规则数据 (:CALCulate:LLINe[1]|2:DATA)

命令格式	:CALCulate:LLINe[1] 2:ADD <x-axis>,<ampl>
说明	增加限制规则数据
参数类型	X 轴: 浮点型 幅度: 浮点型
参数范围	X 轴: 0~3.2GHz 幅度: 无
返回值	X 轴: 浮点型

	幅度: 浮点型
默认值	X 轴: -1Hz 幅度: 0 dBm
菜单项	限制 > 编辑
举例	:CALCulate:LLINe1:ADD 1000000,-20

3.6.2.10 删除指定限制规则点 (:CALCulate:LLINe[1]|2:DELeTe)

命令格式	:CALCulate:LLINe[1] 2:DELeTe <number>
说 明	删除指定限制规则点
参数类型	整型
参数范围	无
返回值	无
默认值	无
菜单项	限制 > 编辑 > 删除点
举例	:CALCulate:LLINe1:DELeTe 2

3.6.2.11 删除全部限制规则点 (:CALCulate:LLINe:ALL:DELeTe)

命令格式	:CALCulate:LLINe[1] 2:ALL:DELeTe
说 明	删除全部限制规则点
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	限制 > 编辑 > 删除全部
举例	:CALCulate:LLINe2:ALL:DELeTe

3.6.2.12 限制规则水平轴单位 (:CALCulate:LLINe:CONTRol:DOMain)

命令格式	:CALCulate:LLINe:CONTRol:DOMain FREQUency TIME :CALCulate:LLINe:CONTRol:DOMain?
说 明	设置限制规则水平单位 查询限制规则水平单位
参数类型	枚举
参数范围	FREQUency: 频率 TIME: 时间
返回值	枚举
默认值	FREQUency
菜单项	限制 > 设置 > X 轴
举例	:CALCulate:LLINe:CONTRol:DOMain FREQUency

3.6.2.13 限制蜂鸣器（:CALCulate:LLINe:CONTRol:BEEP）

命令格式	:CALCulate:LLINe:CONTRol:BEEP OFF ON 0 1 :CALCulate:LLINe:CONTRol:BEEP?
说明	设置限制蜂鸣器 获取限制蜂鸣器状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	ON
菜单项	限制 > 设置 > 蜂鸣器
举例	:CALCulate:LLINe:CONTRol:BEEP OFF

3.6.2.14 查询限制测试结果（:CALCulate:LLINe:FAIL?）

命令格式	:CALCulate:LLINe:FAIL?
说明	查询限制测试结果
参数类型	无
参数范围	无
返回值	PASS FAIL
默认值	无
菜单项	无
举例	:CALCulate:LLINe:FAIL?

3.6.2.15 限制测试失败即停（:CALCulate:LLINe:FAIL:STOP）

命令格式	:CALCulate:LLINe:FAIL:STOP OFF ON 0 1 :CALCulate:LLINe:FAIL:STOP?
说明	设置限制测试失败即停 查询失败即停状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	限制 > 设置 > 失败即停
举例	:CALCulate:LLINe:FAIL:STOP OFF

3.7 Measurement 命令系统

3.7.1 邻道功率比（ACPR）

3.7.1.1 主信道带宽

（[:SENSe]:ACPRatio:BWIDth:INTegration）

命令格式	[:SENSe]:ACPRatio:BWIDth:INTegration <freq>
------	---

	[[:SENSe]:ACPRatio:BWIDth:INTEgration?
说 明	设置主信道带宽 获取主信道带宽
参数类型	浮点型，单位 Hz、KHz、MHz、GHz
参数范围	100 Hz~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型，单位 Hz
默认值	1MHz
菜单项	测量 > 邻道功率比 > 测量设置 > 主信道
举例	INSTRument:MEASure ACPR :ACPRatio:BWIDth:INTEgration 20 MHz

3.7.1.2 邻道带宽

([:SENSe]:ACPRatio:OFFSet:BWIDth[:INTEgration])

命令格式	[[:SENSe]:ACPRatio:OFFSet:BWIDth[:INTEgration] <freq> [:SENSe]:ACPRatio:OFFSet:BWIDth[:INTEgration]?
说 明	设置邻道带宽 获取邻道带宽
参数类型	浮点型，单位 Hz、KHz、MHz、GHz
参数范围	100 Hz~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型，单位 Hz
默认值	1MHz
菜单项	测量 > 邻道功率比 > 测量设置 > 邻道带宽
举例	:ACPRatio:OFFSet:BWIDth 20 MHz

3.7.1.3 邻道间隔

([:SENSe]:ACPRatio:OFFSet[:FREQuency])

命令格式	[[:SENSe]:ACPRatio:OFFSet[:FREQuency] <freq> [:SENSe]:ACPRatio:OFFSet[:FREQuency]?
说 明	设置邻道间隔 获取邻道间隔
参数类型	浮点型，单位 Hz、KHz、MHz、GHz
参数范围	100 Hz~700 MHz
返回值	浮点型，单位 Hz
默认值	3MHz
菜单项	测量 > 邻道功率比
举例	:ACPRatio:OFFSet 20 MHz

3.7.1.4 读取主信道功率 (:MEASure:ACPRatio:MAIN?)

命令格式	:MEASure:ACPRatio:MAIN?
说 明	获取主信道功率
参数类型	无

参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	测量 > 邻道功率比
举例	:MEASure:ACPRatio:MAIN?

3.7.1.5 读取低频邻道功率

(:MEASure:ACPRatio:LOWer:POWer?)

命令格式	:MEASure:ACPRatio:LOWer:POWer?
说明	获取低频邻道功率
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	测量 > 邻道功率比
举例	:MEASure:ACPRatio:LOWer:POWer?

3.7.1.6 读取低频邻道功率比 (:MEASure:ACPRatio:LOWer?)

命令格式	:MEASure:ACPRatio:LOWer?
说明	获取低频邻道功率比
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	测量 > 邻道功率比
举例	:MEASure:ACPRatio:LOWer?

3.7.1.7 读取高频邻道功率

(:MEASure:ACPRatio:UPPer:POWer?)

命令格式	:MEASure:ACPRatio:UPPer:POWer?
说明	获取高频邻道功率
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	测量 > 邻道功率比
举例	:MEASure:ACPRatio:UPPer:POWer?

3.7.1.8 读取高频邻道功率比 (:MEASure:ACPRatio:UPPer?)

命令格式	:MEASure:ACPRatio:UPPer?
说明	获取高频邻道功率比

参数类型	无
参数范围	无
返回值	浮点型
默认值	无
菜单项	测量 > 邻道功率比
举例	:MEASure:ACPRatio:UPPer?

3.7.2 信道功率 (CHP)

3.7.2.1 积分带宽 ([:SENSe]:CHPower:BWIDth:INTEgration)

命令格式	[:SENSe]:CHPower:BWIDth:INTEgration <freq> [:SENSe]:CHPower:BWIDth:INTEgration?
说明	设置积分带宽 获取积分带宽
参数类型	浮点型, 单位 Hz、KHz、MHz、GHz
参数范围	100 Hz~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz) 零扫宽情况下为 0~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型, 单位 Hz
默认值	2 MHz
菜单项	测量 > 信道功率 > 测量设置 > 积分带宽
举例	:CHPower:BWIDth:INTEgration 1.8 GHz

3.7.2.2 扫宽功率

([:SENSe]:CHPower:FREQuency:SPAN:POWer)

命令格式	[:SENSe]:CHPower:FREQuency:SPAN:POWer
说明	将信道扫宽的值设置为积分带宽
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	测量 > 信道功率 > 测量设置 > 扫宽功率
举例	:CHPower:FREQuency:SPAN:POWer

3.7.2.3 读取信道功率和功率谱密度 (:MEASure:CHPower?)

命令格式	:MEASure:CHPower?
说明	读信道功率和功率谱密度
参数类型	无
参数范围	无
返回值	浮点型, 信道功率单位 dBm 浮点型, 功率谱密度单位 dBm/Hz

默认值	无
菜单项	测量 > 信道功率
举例	:MEASure:CHPower?

3.7.2.4 读取信道功率 (:MEASure:CHPower:CHPower?)

命令格式	:MEASure:CHPower:CHPower?
说明	读信道功率
参数类型	无
参数范围	无
返回值	浮点型, 信道功率单位 dBm
默认值	无
菜单项	测量 > 信道功率
举例	:MEASure:CHPower:CHPower?

3.7.2.5 读取信道功率谱密度 (:MEASure:CHPower:DENSity?)

命令格式	:MEASure:CHPower:DENSity?
说明	读信道功率谱密度
参数类型	无
参数范围	无
返回值	浮点型, 功率谱密度单位 dBm/Hz
默认值	无
菜单项	测量 > 信道功率
举例	:MEASure:CHPower:DENSity?

3.7.3 占用带宽 (OBW)

3.7.3.1 计算方法 ([:SENSE]:OBWidth:METHod)

命令格式	[:SENSE]:OBWidth:METHod PERCent DBC [:SENSE]:OBWidth:METHod?
说明	设置占用带宽的计算方法 获取占用带宽的计算方法
参数类型	枚举
参数范围	PERCent: 百分比 DBC: dBc
返回值	枚举
默认值	PERCent
菜单项	测量 > 占用带宽 > 测量设置 > 方法
举例	:OBW:METHod PERCent

3.7.3.2 占用带宽百分比 ([:SENSE]:OBWidth:PERCent)

命令格式	[:SENSE]:OBWidth:PERCent <para> [:SENSE]:OBWidth:PERCent?
说明	设置占用带宽百分比

	获取占用带宽百分比
参数类型	浮点型
参数范围	10~99.99
返回值	浮点型
默认值	99
菜单项	测量 > 占用带宽 > 测量设置 > %
举例	:OBW:PERCent 50

3.7.3.3 占用带宽 dBc ([:SENSe]:OBWidth:XDB)

命令格式	[:SENSe]:OBWidth:XDB <value> [:SENSe]:OBWidth:XDB?
说明	设置占用带宽 dBc 值 获取占用带宽 dBc 值
参数类型	浮点型
参数范围	0.1~100
返回值	浮点型
默认值	26
菜单项	测量 > 占用带宽 > 测量设置 > dBc
举例	:OBWidth:XDB 3

3.7.3.4 读取占用带宽和带宽中心 (:MEASure:OBWidth?)

命令格式	:MEASure:OBWidth?
说明	读取占用带宽和带宽中心
参数类型	无
参数范围	无
返回值	浮点型，单位 Hz
默认值	无
菜单项	测量 > 占用带宽
举例	:MEASure:OBW?

3.7.3.5 读取占用带宽 (:MEASure:OBWidth:OBWidth?)

命令格式	:MEASure:OBWidth:OBWidth?
说明	读取占用带宽
参数类型	无
参数范围	无
返回值	浮点型，单位 Hz
默认值	无
菜单项	测量 > 占用带宽
举例	:MEASure:OBW:OBW?

3.7.3.6 读取占用带宽中心 (:MEASure:OBWidth:CENTRoid?)

命令格式	:MEASure:OBWidth:CENTRoid?
------	----------------------------

说明	读取占用带宽中心
参数类型	无
参数范围	无
返回值	浮点型，单位 Hz
默认值	无
菜单项	测量 > 占用带宽
举例	:MEASure:OBW:CENTRoid?

3.7.3.7 传输频率误差

(:MEASure:OBWidth:OBWidth:FERRor?)

命令格式	:MEASure:OBWidth:OBWidth:FERRor?
说明	获取传输频率误差
参数类型	无
参数范围	无
返回值	浮点型，单位 Hz
默认值	无
菜单项	测量 > 占用带宽
举例	:MEASure:OBWidth:OBWidth:FERRor?

3.7.4 时域功率 (T-Power)

3.7.4.1 时域功率中心频率

([:SENSe]:TPOWer:FREQUency:CENTer)

命令格式	[:SENSe]:TPOWer:FREQUency:CENTer <freq> [:SENSe]:TPOWer:FREQUency:CENTer?
说明	设置时域功率中心频率 获取时域功率中心频率
参数类型	浮点型，单位 Hz、KHz、MHz、GHz
参数范围	50 Hz~3.199999950 GHz (2.999999950 GHz、2.099999950 GHz、1.799999950 GHz、1.499999950 GHz、0.999999950 GHz) 零扫宽情况下为 0~3.2 GHz (3.0 GHz、2.1 GHz、1.8 GHz、1.5 GHz、1.0 GHz)
返回值	浮点型，单位 Hz
默认值	1.6GHz (1.5 GHz、1.05 GHz、0.9 GHz、0.75 GHz、0.5 GHz)
菜单项	测量 > 时域功率 > 测量设置 > 中心频率
举例	:TPOWer:FREQUency:CENTer 15KHz

3.7.4.2 时域功率起始线 ([:SENSe]:TPOWer:LLIMit)

命令格式	[:SENSe]:TPOWer:LLIMit <time> [:SENSe]:TPOWer:LLIMit?
说明	设置时域功率功测量起始线

	获取时域功率功测量起始线
参数类型	浮点型，单位 ks、s、ms、us
参数范围	0 ~ 1000 s
返回值	浮点型，时间单位：秒（S）
默认值	0
菜单项	测量 > 时域功率 > 测量设置 > 起始线
举例	:TPOWer:LLIMit 0.01

3.7.4.3 时域功率截止线（[:SENSe]:TPOWer:RLIMit）

命令格式	[:SENSe]:TPOWer:RLIMit <time> [:SENSe]:TPOWer:RLIMit?
说明	设置时域功率功测量截止线 获取时域功率功测量截止线
参数类型	浮点型，单位 ks、s、ms、us
参数范围	0 ~ 1000 s
返回值	浮点型，时间单位：秒（S）
默认值	20ms
菜单项	测量 > 时域功率 > 测量设置 > 终止线
举例	:TPOWer:RLIMit 0.02

3.7.4.4 读取时域功率（:MEASure:TPOWer?）

命令格式	:MEASure:TPOWer?
说明	读取时域功率
参数类型	无
参数范围	无
返回值	浮点型，单位 dBm
默认值	无
菜单项	测量 > 时域功率
举例	:MEASure:TPOWer?

3.7.5 频谱监测（SPECtrogram）

3.7.5.1 频谱监测状态（[:SENSe]:SPECtrogram:STATe）

命令格式	[:SENSe]:SPECtrogram:STATe RUN PAUSe [:SENSe]:SPECtrogram:STATe?
说明	设置频谱监测状态 读取频谱监测状态
参数类型	枚举
参数范围	RUN: 开始 PAUSe: 暂停
返回值	枚举
默认值	RUN
菜单项	测量 > 频谱监测 > 测量设置 > 频谱监测

举例	:SPECtrogram:STATe PAUSE
----	--------------------------

3.7.5.2 重新开始频谱监测 ([:SENSE]:SPECtrogram:REStart)

命令格式	[:SENSE]:SPECtrogram:REStart
说明	重新开始频谱监测
参数类型	无
参数范围	无
返回值	无
默认值	无
菜单项	测量 > 频谱监测 > 测量设置 > 重新开始
举例	:SPECtrogram:REStart

3.7.6 三阶交调 (TOI)

3.7.6.1 读取三阶交调测量结果 (:MEASure:TOI?)

命令格式	:MEASure:TOI?
说明	读取三阶互调失真的测量结果 以科学计数形式返回以逗号间隔的下列值： 低基频信号(Lower Tone)的频率(Hz)，幅度，高基频信号(Upper Tone)的频率(Hz)，幅度，低频 TOI(Lower 3rd)的频率(Hz)，幅度，三阶互调截止点(Intercept)，高频 TOI(Upper 3rd)的频率(Hz)，幅度，三阶互调截止点(Intercept)。
参数类型	无
参数范围	无
返回值	浮点型
默认值	无
菜单项	测量 > 三阶交调
举例	:MEASure:TOI?

3.7.6.2 读取三阶交调截止点 (:MEASure:TOI?)

命令格式	:MEASure:TOI:IP3?
说明	读取低频 TOI(Lower 3rd)和 高频 TOI(Upper 3rd)的三阶互调截止点(Intercept)中的较小值
参数类型	无
参数范围	无
返回值	浮点型
默认值	无
菜单项	测量 > 三阶交调
举例	:MEASure:TOI:IP3?

3.8 触发

3.8.1 触发类型 (:TRIGger[:SEQuence]:SOURce)

命令格式	:TRIGger[:SEQuence]:SOURce IMMEDIATE VIDeo EXTernal :TRIGger[:SEQuence]:SOURce?
说明	设置触发类型 获取触发类型
参数类型	枚举
参数范围	IMMEDIATE:自由触发 VIDeo:视屏触发 EXTernal:外部触发
返回值	枚举
默认值	IMMEDIATE
菜单项	触发
举例	:TRIGger:SOURce IMMEDIATE

3.8.2 触发电平 (:TRIGger[:SEQuence]:VIDeo:LEVel)

命令格式	:TRIGger[:SEQuence]:VIDeo:LEVel <value> :TRIGger[:SEQuence]:VIDeo:LEVel?
说明	设置触发电平 获取触发电平
参数类型	浮点型, 单位 dBm、dBmV、dBuV、V、W
参数范围	单位为 dBm 时: -200dBm ~ 50dBm 单位为 dBmV 时: -253.01dBmV ~ 96.99dBmV 单位为 dBuV 时: -193.01dBuV ~ 156.99dBuV 单位为 Volts 时: 223.61aV ~ 70.71V 单位为 Watts 时: 1.00E-33W ~ 100W
返回值	浮点型, 单位 dBm、dBmV、dBuV、V、W
默认值	0 dBm
菜单项	触发 > 触发电平
举例	:TRIGger:VIDeo:LEVel 0.5 dBm

3.8.3 触发沿 (:TRIGger[:SEQuence]:RFBurst:SLOPe)

命令格式	:TRIGger[:SEQuence]:RFBurst:SLOPe POSitive NEGative :TRIGger[:SEQuence]:RFBurst:SLOPe?
说明	设置触发沿 上升沿 下降沿 获取触发沿
参数类型	枚举
参数范围	POSitive:上升沿 NEGative:下降沿
返回值	枚举

默认值	POSitive
菜单项	触发 > 触发沿
举例	:TRIGger:RFBurst:SLOPe POSitive

3.9 TG

3.9.1TG 开关 (:OUTPut[:STATe])

命令格式	:OUTPut[:STATe] OFF ON 0 1 :OUTPut[:STATe]?
说 明	开关 TG 读取 TG 开关状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	0
菜单项	TG > TG
举例	:OUTPut ON

3.9.2TG 输出功率

(:SOURce:POWER[:LEVel][:IMMediate][:AMPLitude])

命令格式	:SOURce:POWER[:LEVel][:IMMediate][:AMPLitude] <value> :SOURce:POWER[:LEVel][:IMMediate][:AMPLitude]?
说 明	设置 TG 输出功率 读取 TG 输出功率
参数类型	浮点型，单位 dBm
参数范围	0 dBm ~ -20 dBm
返回值	浮点型
默认值	0 dBm
菜单项	TG > 信号幅度
举例	:SOURce:POWER -20

3.9.3TG 电平偏移 (:SOURce:CORRection:OFFSet)

命令格式	:SOURce:CORRection:OFFSet <value> :SOURce:CORRection:OFFSet?
说 明	设置 TG 输出电平偏移 读取 TG 输出电平偏移
参数类型	浮点型，单位 dBm
参数范围	200 dBm ~ -200 dBm
返回值	浮点型
默认值	0 dBm
菜单项	TG > 幅度偏移

举例	:SOURce:CORRection:OFFSet 1
----	-----------------------------

3.9.4 TG 归一化开关 (:CALCulate:NTData[:STATe])

命令格式	:CALCulate:NTData[:STATe] OFF ON 0 1 :CALCulate:NTData[:STATe]?
说明	开关 TG 归一化 读取 TG 归一化状态
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	0
菜单项	TG > 归一化 > 归一化
举例	:CALCulate:NTData ON

3.9.5 TG 归一化参考电平

(:DISPlay:WINDow:TRACe:Y[:SCALe]:NRLevel)

命令格式	:DISPlay:WINDow:TRACe:Y[:SCALe]:NRLevel <value> :DISPlay:WINDow:TRACe:Y[:SCALe]:NRLevel?
说明	设置 TG 归一化参考电平 读取 TG 归一化参考电平
参数类型	浮点型, 单位 dB
参数范围	-200 dB ~ 200 dB
返回值	浮点型, 单位 dB
默认值	0 dB
菜单项	TG > 归一化 > 参考电平
举例	:DISPlay:WINDow:TRACe:Y:NRLevel 10

3.9.6 TG 归一化参考位置

(:DISPlay:WINDow:TRACe:Y[:SCALe]:NRPosition)

命令格式	:DISPlay:WINDow:TRACe:Y[:SCALe]:NRPosition <integer> :DISPlay:WINDow:TRACe:Y[:SCALe]:NRPosition?
说明	设置 TG 归一化参考位置 读取 TG 归一化参考位置
参数类型	整型
参数范围	0 ~ 100%
返回值	浮点型
默认值	100%
菜单项	TG > 归一化 > 参考位置
举例	:DISPlay:WINDow:TRACe:Y:NRPosition 10

3.9.7 TG 归一化参考迹线开关 (:DISPlay:WINDow:NTTRace[:STATe])

命令格式	:DISPlay:WINDow:NTTRace[:STATe] OFF ON 0 1 :DISPlay:WINDow:NTTRace[:STATe]?
说明	设置 TG 归一化参考迹线开关
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	0
菜单项	TG > 归一化 > 参考迹线
举例	:DISPlay:WINDow:NTTRace ON

3.10 调制解调

3.10.1 解调模式 (:SENSe]:DEMod)

命令格式	:SENSe]:DEMod AM FM OFF :SENSe]:DEMod?
说明	设置解调模式 获取解调模式
参数类型	枚举
参数范围	AM: 幅度调制 FM: 频率调制 OFF: 关
返回值	枚举
默认值	OFF
菜单项	解调
举例	:DEMod AM

3.10.2 解调时间 (:SENSe]:DEMod:TIME)

命令格式	:SENSe]:DEMod:TIME <time> :SENSe]:DEMod:TIME?
说明	设置解调时间 获取解调时间
参数类型	浮点型, 单位 ms、us、s
参数范围	5 ms ~1000 s
返回值	浮点型, 单位 s
默认值	5 s
菜单项	解调
举例	DEMod:TIME 5 ms

3.10.3 耳机 ([:SENSe]:DEMod:EPHONE)

命令格式	[:SENSe]:DEMod:EPHONE OFF ON 0 1 [:SENSe]:DEMod:EPHONE?
说明	开关耳机
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	解调 > 耳机
举例	:DEMod:EPHONE ON

3.10.4 声音 ([:SENSe]:DEMod:VOLUME)

命令格式	[:SENSe]:DEMod:VOLUME <value> [:SENSe]:DEMod:VOLUME?
说明	调节声音大小 获取声音大小
参数类型	整型
参数范围	0 ~ 10
返回值	整型
默认值	6
菜单项	解调 > 声音
举例	:DEMod:EPHONE ON

3.11 校正

3.11.1 校正开关 ([:SENSe]:CALibration:STATE)

命令格式	[:SENSe]:CALibration:STATE OFF ON 0 1 [:SENSe]:CALibration:STATE?
说明	设置校正开关 获取校正开关
参数类型	布尔型
参数范围	OFF ON 0 1
返回值	0 1
默认值	OFF
菜单项	系统 > 校准 > 自动校准
举例	:CALibration:STATE ON

3.12 存储命令

3.12.1 存储文件 (:MMEMory:STORe)

命令格式	:MMEMory:STORe STA TRC COR CSV LIM JPG BMP PNG,<file>
说明	写入文件
参数类型	字符串
参数范围	无
返回值	无
默认值	无
菜单项	文件>保存
举例	:MMEMory:STORe STA,ABC.sta

3.12.2 加载文件 (:MMEMory:LOAD)

命令格式	:MMEMory:LOAD STA TRC COR LIM,<file>
说明	读取文件
参数类型	字符串
参数范围	无
返回值	无
默认值	无
菜单项	文件>打开/加载
举例	:MMEMory:LOAD STA,ABC.sta

3.12.3 删除文件 (:MMEMory:DELeTe)

命令格式	:MMEMory:DELeTe <file>
说明	删除文件或文件夹
参数类型	字符串
参数范围	无
返回值	无
默认值	无
菜单项	文件>操作>删除
举例	:MMEMory:DELeTe ABC.sta

4. 编程示例

本章给程序员了一些例子，在这些例子中你可以看到如何使用 `ni-visa` 库或 `Sockets` 和本章之前描述的命令来控制我们的设备。通过编程示例，你可以开发更多的功能应用。这个例子是 `Visual Studio` 开发的项目。

4.1 使用 VISA 的编程示例

4.1.1 VC++ 示例

环境: Windows xp system, Visual Studio

示例内容: 使用 `ni-visa` 控制设备和 `Usbtmc TCP / IP` 访问读写。

按以下步骤完成示例:

1、打开 `Visual Studio`, 创建一个新的 `vc++ win32` 项目。

设置项目环境使用 `ni-visa` 库, 有两种方法可以使用 `ni-visa`, 静态方式和自动方式:

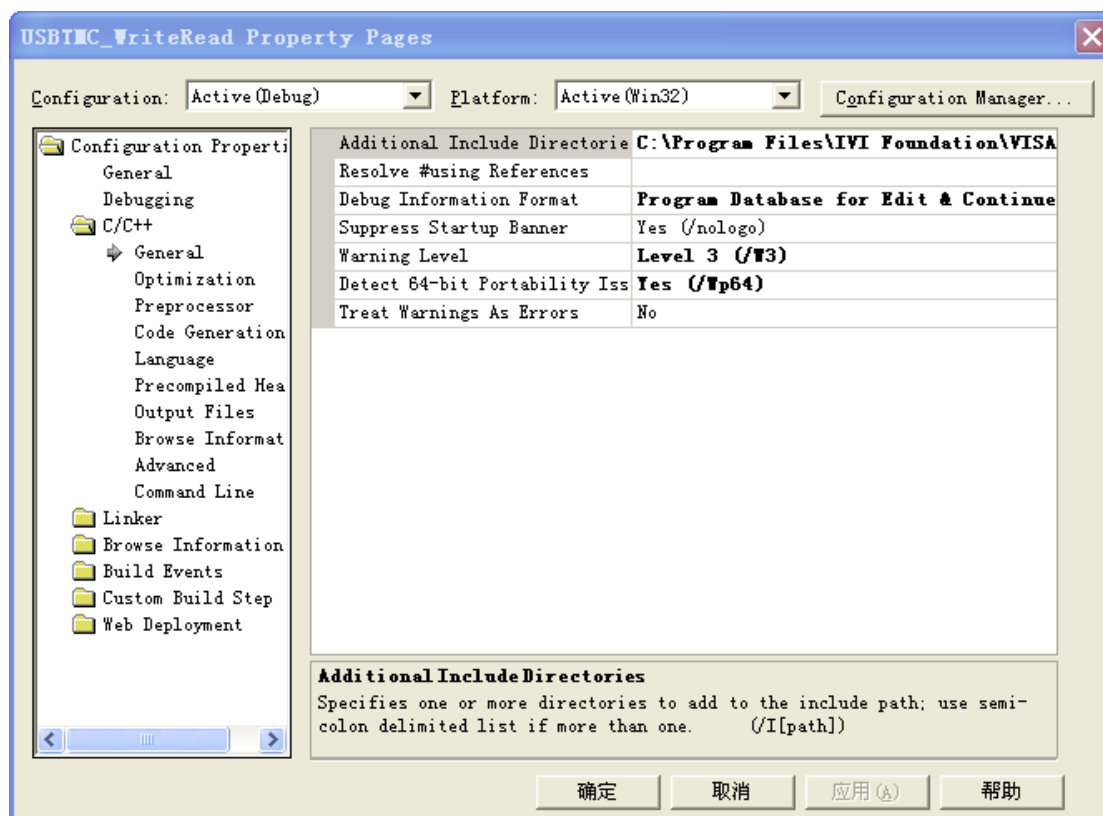
(1) 静态方式:

在 `NI-VISA` 安装路径查找文件: `visa.h`, `visatype.h`, `visa32.lib`。将它们复制到你的项目, 并将它们添加到项目中。在项目 `.cpp` 文件, 添加如下两行

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

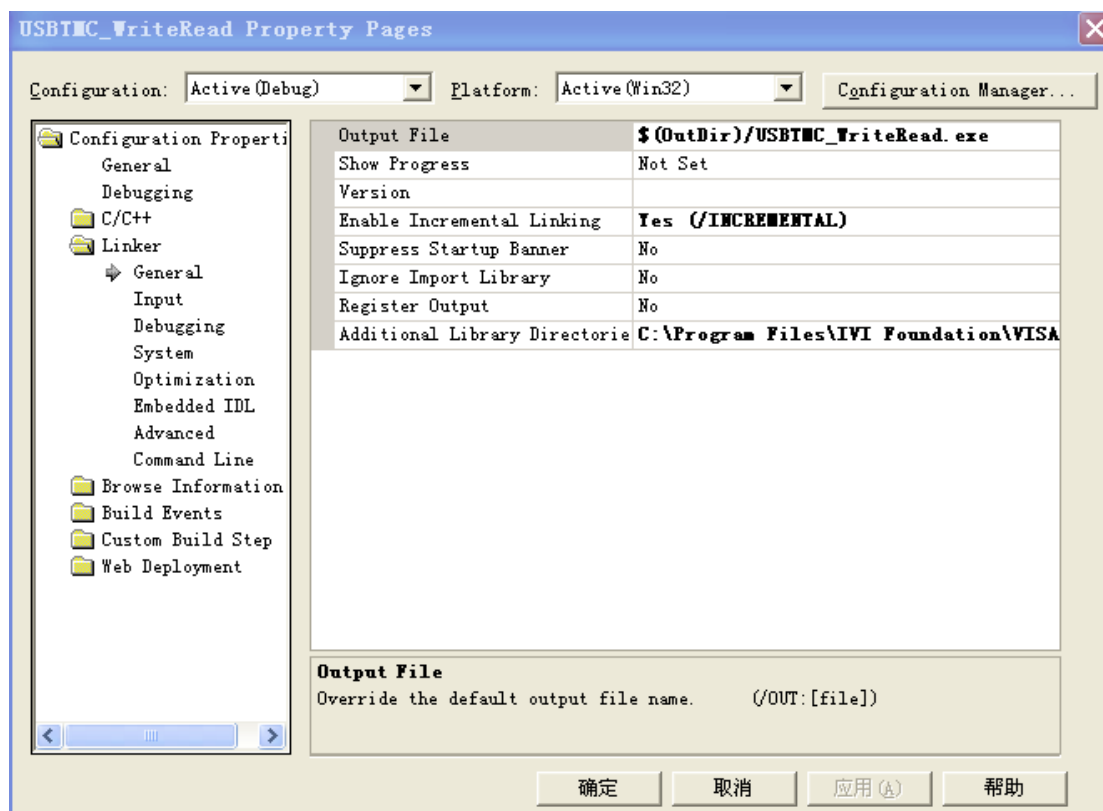
(2) 自动方式:

设置 `.h` 文件包括目录, `ni-visa` 安装路径。在我们的电脑, 我们设置的路径是: `C: \Program Files\IVI Foundation\VISA\WINNT\include`。设置这条路径到项目—属性—`C/C++`—通用—附加包含路径, 如图:

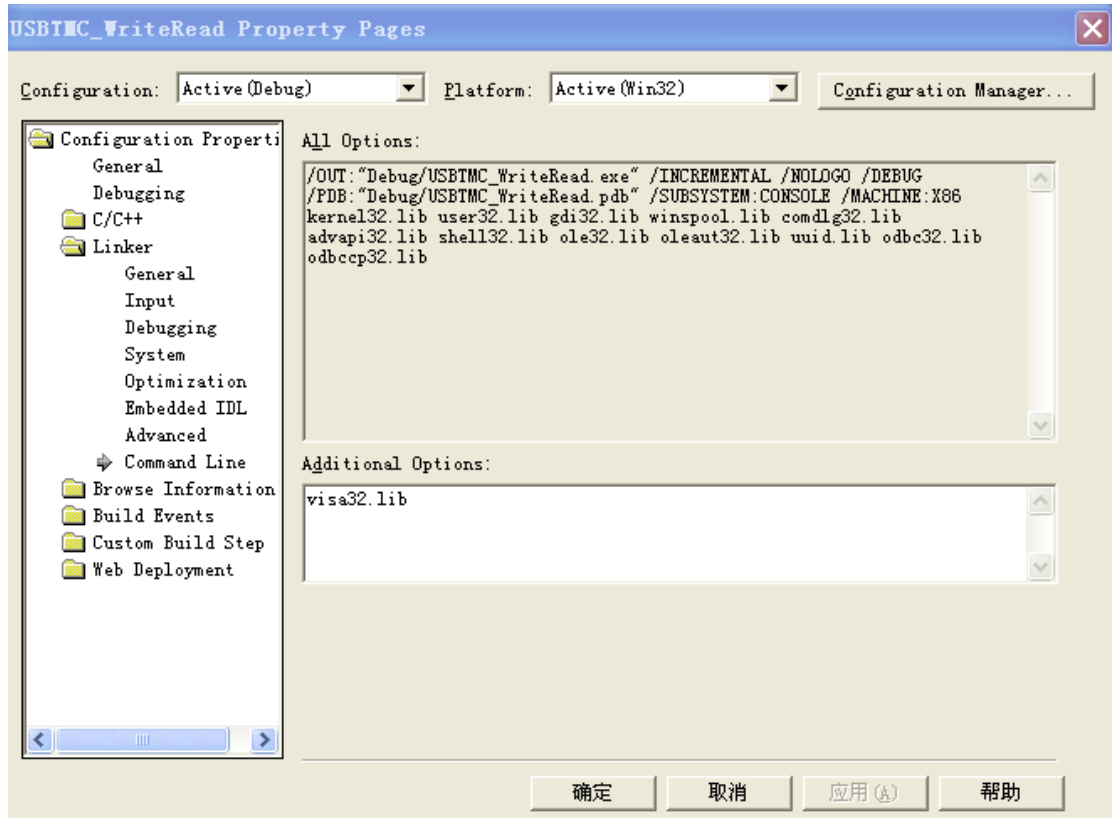


2、设置库路径设置库文件:

设置库路径: 在 ni-visa 安装路径, 在我们的电脑, 我们设置的路径是: C:\Program Files\IVI Foundation\VISA\WINNT\LIBMSC。设置这条路径到项目—性能—连接器—常规—附加库目录, 如图:



设置库文件： project---properties---Linker---Command Line---Additional Options: visa32.lib



包括 visa.h file:在 XXX.cpp 文件里:

```
#include <visa.h>
```

3、增加代码:

(1)USBTMC 存取代码::

写一个 Usbtmc_test 函数。

```
int Usbtmc_test()
{
/*这段代码演示了使用 NI-VISA 发送同步读取和写入命令到 */
/*一个 USB 测试&测量类(USBTMC)仪器。 */
/*这个例子写“ IDN ?\n ”字符串到所有连接到系统的 USBTMC */
/*设备并试图使用读写函数读回结果。 */
/*代码的一般流程是打开资源管理器 */
/*打开 VISA 会话到仪器 */
/*使用 viPrintf 写仪器标志查询 */
/* 尝试随 viScanf 读取一个响应 */
/* 关闭 VISA 会话 */
/*****/
```

```
ViSession defaultRM;
```

```
ViSession instr;
ViUInt32 numInstrs;
ViFindList findList;
ViStatus status;
char instrResourceString[VI_FIND_BUFLEN];
unsigned char buffer[100];
inti;
/*首先, 我们必须调用 viOpenDefaultRM 得到管理器的句柄。 */
/*我们将在 defaultRM 存储此手柄。 */
status=viOpenDefaultRM (&defaultRM);
if (status<VI_SUCCESS)
{
printf ("Could not open a session to the VISA Resource Manager!\n");
return status;
}
/**寻找我们的系统中所有的 USB TMC VISA 资源 */
*和然后将资源的数目存储在系统中的 numInstrs 里。 */
status = viFindRsrc (defaultRM, "USB?*INSTR", &findList, &numInstrs,
instrResourceString);
if (status<VI_SUCCESS)
{
printf ("An error occurred while finding resources.\nPress 'Enter' to continue.");
fflush(stdin);
getchar();
viClose (defaultRM);
return status;
}
/**现在, 我们将对所有 USB TMC 仪器打开 VISA 会话。我们必须
* 从 viOpenDefaultRM 使用句柄, 也必须使用一个字符串指示要
*打开的仪器, 这就是所谓的仪器描述符。该字符串的格式可以在功
*能面板中右键单击参数描述中找到。打开一个会话到设备后, 我们
*将得到一个仪器使用的句柄, 在之后使用 VISA 功能时用到。在这
*个函数的 AccessMode 和超时参数是为将来的功能预留。这两个参
*数被给予值 VI_NULL。 */
for (i=0; i<int(numInstrs); i++)
{
if (i> 0)
{
viFindNext (findList, instrResourceString);
}
status = viOpen (defaultRM, instrResourceString, VI_NULL, VI_NULL, &instr);
if (status<VI_SUCCESS)
{
printf ("Cannot open a session to the device %d.\n", i+1);
continue;
}
}
/* *在这一点上, 我们现在有一个会话打开到 USB TMC 仪器。现在,
```

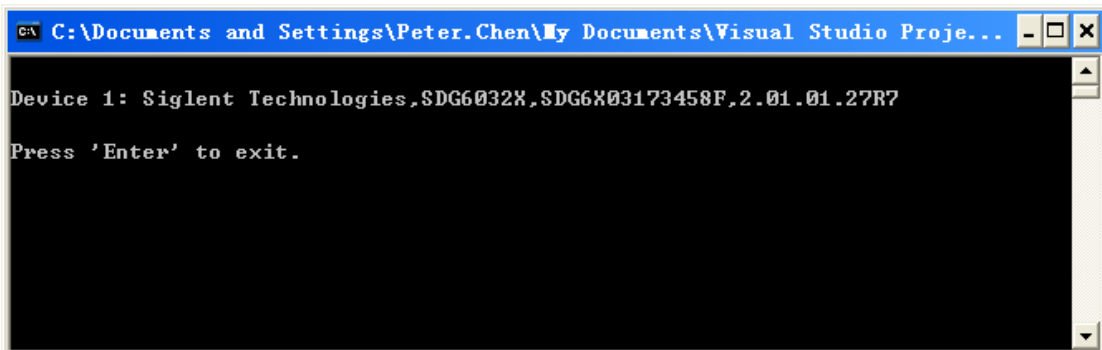
*我们将使用 viPrintf 函数发送字符串"*IDN?\n"到设备，要求设备识别。*/

```
char * cmmand = "*IDN?\n";
status = viPrintf (instr, cmmand);
if (status<VI_SUCCESS)
{
printf ("Error writing to the device %d.\n", i+1);
status = viClose (instr);
continue;
}
/** 现在我们将尝试从设备读回一个设备信息查询的响应。我们将
*使用 viScanf 函数来获取数据。在数据被读出后，响应显示出来*/
status = viScanf(instr, "%t", buffer);
if (status<VI_SUCCESS)
{
printf ("Error reading a response from the device %d.\n", i+1);
}
else
{
printf ("\nDevice %d: %s\n", i+1 , buffer);
}
status = viClose (instr);

/**现在，我们将关闭会话使用 viClose 仪器。此操作释放所有系统资源。*/
status = viClose (defaultRM);
printf("Press 'Enter' to exit.");
fflush(stdin);
getchar();
return 0;
}
```

```
int _tmain(int argc, _TCHAR* argv[])
{
Usbtmc_test();
return 0;
}
```

运行结果:



```
C:\Documents and Settings\Peter.Chen\My Documents\Visual Studio Proje...
Device 1: Siglent Technologies,SDG6032X,SDG6X03173458F,2.01.01.27R7
Press 'Enter' to exit.
```

(2)TCP/IP access code.

写 TCP_IP_Test 函数:

```
int TCP_IP_Test(char *pIP)
{
```

```

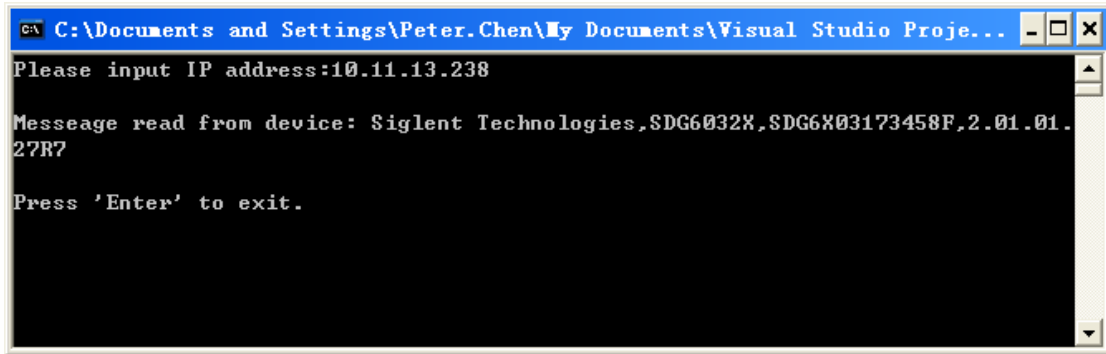
char outputBuffer[VI_FIND_BUFLEN];
ViSession defaultRM, instr;
ViStatus status;
/* 首先，我们需要打开默认的资源管理器。 */
status = viOpenDefaultRM (&defaultRM);
if (status<VI_SUCCESS)
{
printf("Could not open a session to the VISA Resource Manager!\n");
}

/*现在，我们将通过 TCP / IP 设备打开一个会话*/

char head[256] ="TCPIP0::";
char tail[] = "::INSTR";
strcat(head,pIP);
strcat(head,tail);
status = viOpen (defaultRM, head, VI_LOAD_CONFIG, VI_NULL, &instr);
if (status<VI_SUCCESS)
{
printf ("An error occurred opening the session\n");
viClose(defaultRM);
}
status = viPrintf(instr, "*idn?\n");
status = viScanf(instr, "%t", outputBuffer);
if (status<VI_SUCCESS)
{
printf("viRead failed with error code: %x \n",status);
viClose(defaultRM);
}
else
{
printf ("\nMesseage read from device: %*s\n", 0,outputBuffer);
}
status = viClose (instr);
status = viClose (defaultRM);
printf("Press 'Enter' to exit.");
fflush(stdin);
getchar();
return 0;
}
int _tmain(int argc, _TCHAR* argv[])
{
printf("Please input IP address:");
char ip[256];
fflush(stdin);
gets(ip);
TCP_IP_Test(ip);
return 0;
}

```

运行结果:



```
C:\Documents and Settings\Peter.Chen\My Documents\Visual Studio Proje...
Please input IP address:10.11.13.238

Message read from device: Siglent Technologies,SDG6032X,SDG6X03173458F,2.01.01.
27R7

Press 'Enter' to exit.
```

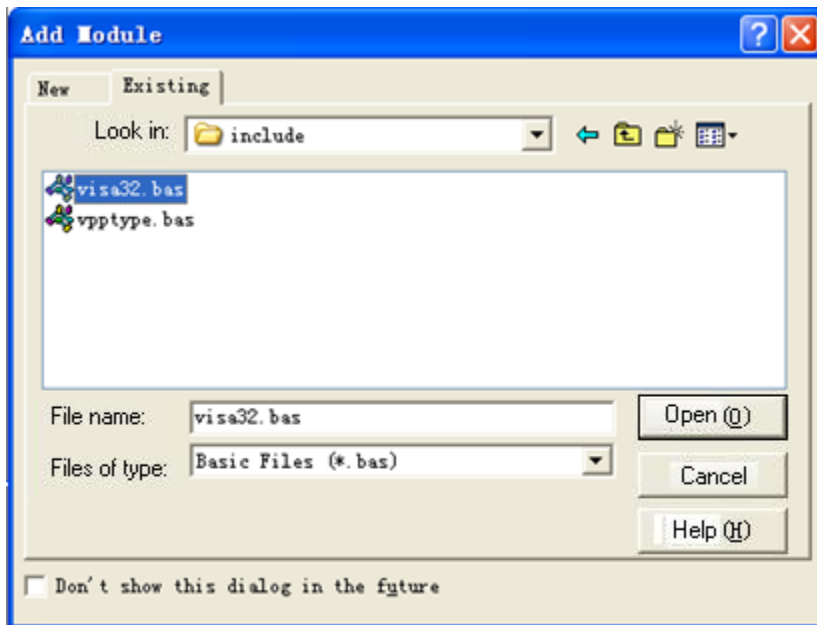
4.1.2 VB 示例

环境: Windows 7 , Microsoft Visual Basic 6.0

示例内容: 使用 NI-VISA, 通过 USBTMC 和 TCP/ IP 访问控制设备去写入和读取。

按照步骤完成的例子:

- 1、打开 Visual Basic, 建立一个标准的应用程序项目 (标准 EXE)。
- 2、设置项目的环境中使用 NI-VISA 库, 单击项目的现有标签>>添加模块。搜索 visa32.bas 中 NI-VISA 安装路径下的 include 文件夹文件, 并添加文件。



这使得 VISA 功能和 VISA 的数据类型在程序中使用。

3、添加代码:

(1)USBTMC 存取代码:

写 Usbtmc_test 函数.

Private Function Usbtmc_test() As Long

'这段代码演示了使用 NI-VISA 发送同步读取和写入命令到

'一个 USB 测试&测量类(USBTMC)仪器。

'这个例子写"* IDN ?\n "字符串到所有连接到系统的 USBTMC

'设备并试图使用读写函数读回结果。

'代码的一般流程是打开资源管理器

'打开 VISA 会话到仪器

'使用 viPrintf 写仪器标志查询

'尝试随 viScanf 读取一个响应

'关闭 VISA 会话

```
Const MAX_CNT = 200

Dim defaultRM As Long
Dim instrsesn As Long
Dim numInstrs As Long
Dim findList As Long
Dim retCount As Long
Dim status As Long
Dim instrResourceString As String * VI_FIND_BUFLEN
Dim Buffer As String * MAX_CNT
Dim i As Integer
```

'首先，我们必须调用 viOpenDefaultRM 得到管理器的句柄。

'我们将在 defaultRM 存储此手柄。

```
status = viOpenDefaultRM(defaultRM)
If (status < VI_SUCCESS) Then
    resultTxt.Text = "Could not open a session to the VISA Resource
Manager!"
    Usbtmc_test = status
    Exit Function
End If
```

'寻找我们的系统中所有的 USB TMC VISA 资源

'和然后将资源的数目存储在系统中的 numInstrs 里。

```
status = viFindRsrc(defaultRM, "USB?*INSTR", findList, numInstrs,
instrResourceString)
If (status < VI_SUCCESS) Then
    resultTxt.Text = "An error occurred while finding resources."
    viClose(defaultRM)
    Usbtmc_test = status
    Exit Function
End If
```

'现在，我们将对所有 USB TMC 仪器打开 VISA 会话。我们必须

'从 viOpenDefaultRM 使用句柄， 也必须使用一个字符串指示要

'打开的仪器，这就是所谓的仪器描述符。该字符串的格式可以在功

'能面板中右键单击参数描述中找到。打开一个会话到设备后，我们

'将得到一个仪器使用的句柄，在之后使用 VISA 功能时用到。在这

个函数的 `AccessMode` 和超时参数是为将来的功能预留。这两个参数被给予值 `VI_NULL`。

```
For i = 0 To numInstrs
    If (i > 0) Then
        status = viFindNext(findList, instrResourceString)
    End If
    status = viOpen(defaultRM, instrResourceString, VI_NULL, VI_NULL,
instrsesn)
    If (status < VI_SUCCESS) Then
        resultTxt.Text = "Cannot open a session to the device " + CStr(i + 1)
        GoTo NextFind
    End If
```

在这一点上，我们现在有一个会话打开到 **USB TMC** 仪器。现在，我们将使用 `viPrintf` 函数发送字符串“*IDN?\n”到设备，要求设备识别。

```
status = viWrite(instrsesn, "*IDN?", 5, retCount)
If (status < VI_SUCCESS) Then
    resultTxt.Text = "Error writing to the device."
    status = viClose(instrsesn)
    GoTo NextFind
End If
```

现在我们将尝试从设备读回一个设备信息查询的响应。我们将使用 `viScanf` 函数来获取数据。在数据被读出后，响应显示出来

```
status = viRead(instrsesn, Buffer, MAX_CNT, retCount)
If (status < VI_SUCCESS) Then
    resultTxt.Text = "Error reading a response from the device." + CStr(i + 1)
Else
    resultTxt.Text = "Read from device: " + CStr(i + 1) + " " + Buffer
End If
status = viClose(instrsesn)
Next i
```

现在，我们将关闭会话使用 `viClose` 仪器。此操作释放所有系统资源。

```
status = viClose(defaultRM)
Usbtmc_test = 0
End Function
```

(2)TCP/IP 访问代码.

写 `TCP_IP_Test` 函数:

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

首先，我们需要打开默认的资源管理器。


```

    status = viOpenDefaultRM(defaultRM)
    If (status < VI_SUCCESS) Then
        resultTxt.Text = "Could not open a session to the VISA
Resource Manager!"
        TCP_IP_Test = status
        Exit Function
    End If

```

' 现在, 我们将通过 TCP / IP 设备打开一个会话

```

    status = viOpen(defaultRM, "TCPIP0::" + ip + "::INSTR",
VI_LOAD_CONFIG, VI_NULL, instrsesn)
    If (status < VI_SUCCESS) Then
        resultTxt.Text = "An error occurred opening the session"
        viClose(defaultRM)
        TCP_IP_Test = status
        Exit Function
    End If

    status = viWrite(instrsesn, "*IDN?", 5, count)
    If (status < VI_SUCCESS) Then
        resultTxt.Text = "Error writing to the device."
    End If
    status = viRead(instrsesn, outputBuffer, VI_FIND_BUFLEN, count)
    If (status < VI_SUCCESS) Then
        resultTxt.Text = "Error reading a response from the device." +
CStr(i + 1)
    Else
        resultTxt.Text = "read from device:" + outputBuffer
    End If
    status = viClose(instrsesn)
    status = viClose(defaultRM)
    TCP_IP_Test = 0
End Function

```

(3) 按键控制代码:

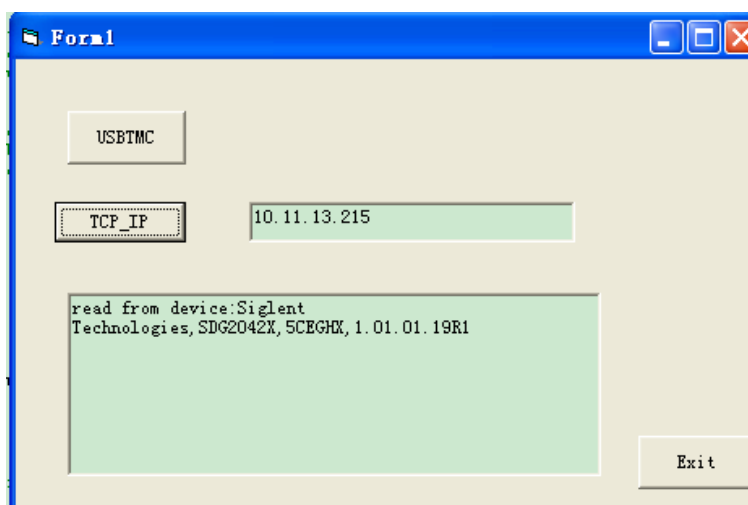
```

Private Sub exitBtn_Click()
    End
End Sub
Private Sub tcpipBtn_Click()
    Dim stat As Long
    stat = TCP_IP_Test(ipTxt.Text)
    If (stat < VI_SUCCESS) Then
        resultTxt.Text = Hex(stat)
    End If
End Sub
Private Sub usbBtn_Click()
    Dim stat As Long
    stat = Usbtmc_test
    If (stat < VI_SUCCESS) Then
        resultTxt.Text = Hex(stat)
    End If
End Sub

```

End If
End Sub

运行结果:



4.1.3 MATLAB 示例

环境: windows 7, MATLAB R2013a

示例内容: 使用 NI-VISA, 通过 USBTMC 和 TCP/ IP 访问控制设备去写入和读取。

按照步骤完成的例子:

1、打开 MATLAB, 修改当前目录。在本演示中, 将当前目录修改为 D: \ USBTMC_TCPIP_Demo。

2、点击文件>>新建>>脚本 (File>>New>>Script) 在 Matlab 界面来创建一个空的 M 文档

3、添加代码:

(1)USBTMC 存取代码:

写入 Usbtmc_test 函数。

```
function USBTMC_test()
```

```
%这段代码演示了使用 NI-VISA 发送同步读取和写入命令到
```

```
%一个 USB 测试&测量类(USBTMC)仪器。
```

```
%创建一个 VISA-USB 对象连接到 USB 仪器上
```

```
vu = visa('ni','USB0::0xF4EC::0x1300::0123456789::INSTR');
```

```
%打开创建的 VISA 对象
```

```
fopen(vu);
```

%发送字符串"* IDN?", 查询设备信息。

```
fprintf(vu, '*IDN?');
```

%请求数据

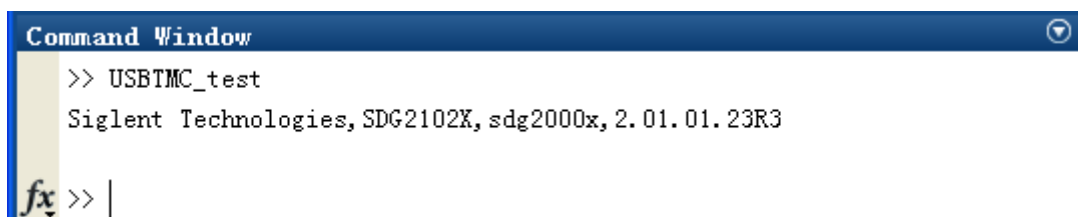
```
outputbuffer = fscanf(vu);
disp(outputbuffer);
```

%关闭 VISA 对象

```
fclose(vu);
delete(vu);
clear vu;
```

```
end
```

运行结果:



```
Command Window
>> USBTMC_test
Siglent Technologies, SDG2102X, sdg2000x, 2.01.01.23R3
fx >> |
```

(2)TCP/IP 存取代码:

写入 TCP_IP_Test 函数。

```
function TCP_IP_test()
```

%这段代码演示了使用 NI-VISA 发送同步读取和写入命令到

%一个 TCP/IP 仪器。

%创建一个 VISA-TCPIP 对象连接到配置了 IP 地址的仪器。

```
vt = visa('ni', ['TCPIP0::', IPstr, '::INSTR']);
```

%打开创建的 VISA 对象

```
fopen(vt);
```

%发送字符串"*IDN?", 查询设备信息

```
fprintf(vt, '*IDN?');
```

%请求数据

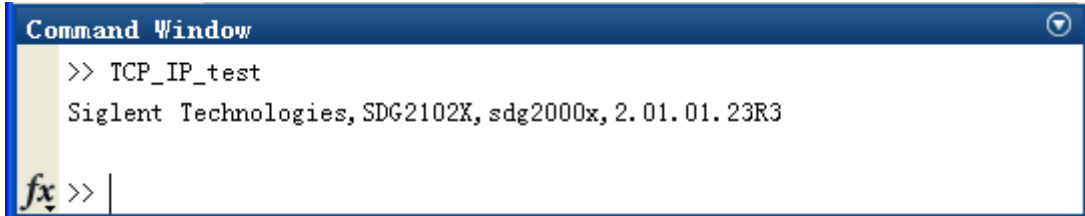
```
outputbuffer = fscanf(vt);
disp(outputbuffer);
```

%关闭 VISA 对象

```
fclose(vt);  
delete(vt);  
clear vt;
```

end

运行结果:



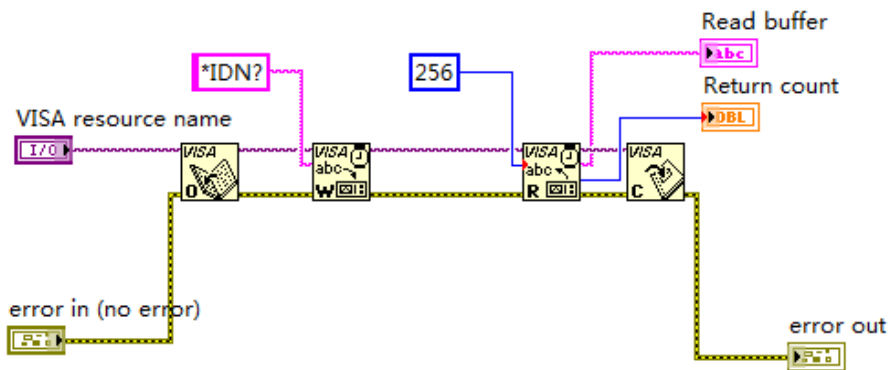
4.1.4 LabVIEW 示例

环境: windows 7 system, LabVIEW 2011

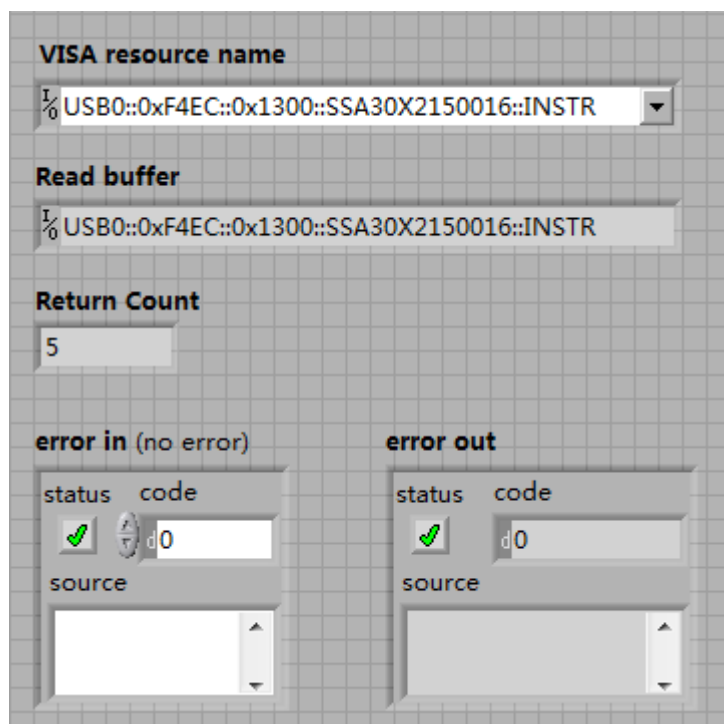
示例内容: 使用 NI-VISA, 通过 USBTMC 和 TCP/IP 访问控制设备去写入和读取。

按照步骤完成的例子:

- 1、打开 LabVIEW, 创建 VI 文件.
- 2、添加控件。右键单击前面板接口, 选择并加入 VISA 资源名称, 错误输入, 错误输出以及控制栏的一些指标
- 3、打开框图接口。在 VISA 资源名称单击鼠标右键, 可以从 VISA 调色板的快捷菜单中选择添加以下功能: VISA 写, VISA 读, VISA 打开和 VISA 关闭。
- 4、把它们连接起来, 如下图所示:



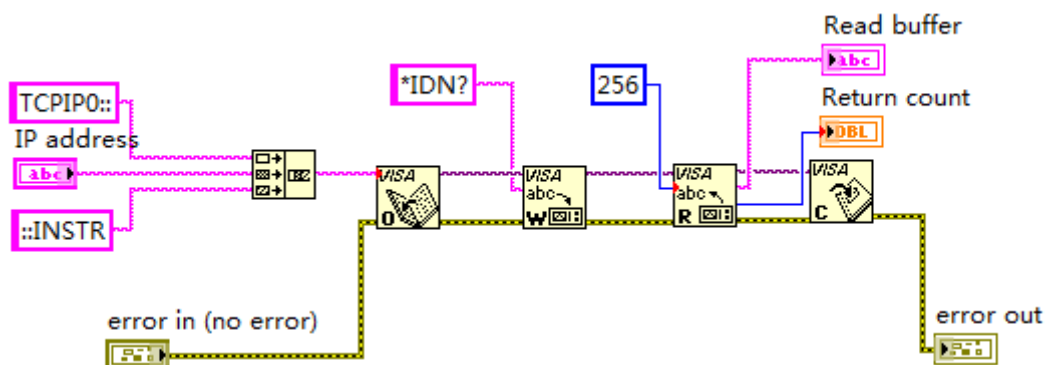
- 5、从 VISA 资源名称列表框中选择设备资源然后运行程序。



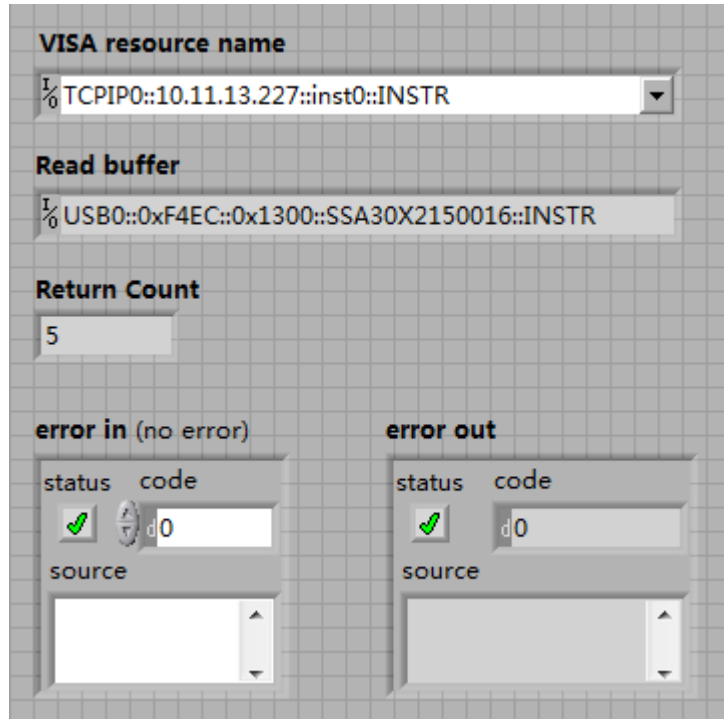
在这个例子中，VI 打开一个 VISA 会话到 USBTMC 设备，写入一个命令到设备，然后读回并响应。此例中，发送的特定的命令是设备 ID 查询。请与您的设备制造商核对设备命令集。在所有通讯完成后，VI 关闭 VISA 会话。

通过 TCP/IP 与以太网仪器通信是类似于 USBTMC 的。但是，你需要改变 VISA 写入和 VISA 读取功能来同步 I/O。LabVIEW 的默认值是异步的 I/O。右键单击该节点，然后从快捷菜单中选择同步 I/O 模式->同步来写入或读取同步数据。

1、把它们连接起来，如下图所示：



2、输入 IP 地址然后运行程序。



4.2 使用 Sockets/Telnet 的编程示例

Windows 操作系统自身支持 socket 通信，这种通信方式也是比较简明的。需要注意的是 SCPI 命令字符串的结尾需要加上“\n”（换行符）。

4.2.1 Python 示例

Python 是一种解释性的编程语言，可让您快速工作并且非常方便。Python 具有一个低级网络模块，可以访问 socket 接口。Python 脚本可以在此基础上进行各种测试和测量任务。

环境: Win7 32bit system, Python v2.7.5

示例内容: 打开 socket，发送询问 SCPI，然后关闭，以上循环十次。

脚本内容:

```
#!/usr/bin/env python
#-*- coding:utf-8 -*-
#-----
# The short script is a example that open a socket, sends a query,
# print the return message and closes the socket.
#-----
import socket # for sockets
import sys # for exit
import time # for sleep
#-----
remote_ip = "10.11.13.32" # should match the instrument's IP address
port = 5024 # the port number of the instrument service
count = 0
```

```
def SocketConnect():
    try:
        #create an AF_INET, STREAM socket (TCP)
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    except socket.error:
        print ('Failed to create socket.')
        sys.exit();
    try:
        #Connect to remote server
        s.connect((remote_ip , port))
        info = s.recv(4096)
        print (info)
    except socket.error:
        print ('failed to connect to ip ' + remote_ip)
    return s

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

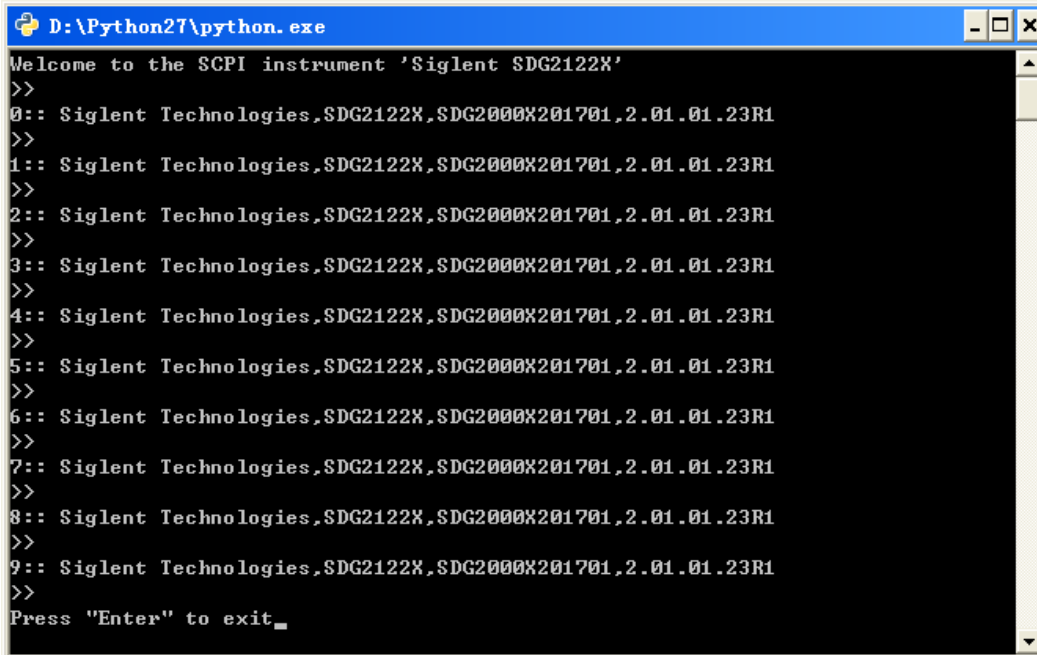
def SocketClose(Sock):
    #close the socket
    Sock.close()
    time.sleep(.300)

def main():
    global remote_ip
    global port
    global count

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

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

运行结果:

A screenshot of a Windows command prompt window titled "D:\Python27\python.exe". The window displays a series of SCPI commands and responses. The first line is "Welcome to the SCPI instrument 'Siglent SDG2122X'". This is followed by ten lines of "0:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1" and "1:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1" through "9:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1". Each line is followed by ">>". The final line is "Press 'Enter' to exit_".

```
D:\Python27\python.exe
Welcome to the SCPI instrument 'Siglent SDG2122X'
>>
0:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
1:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
2:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
3:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
4:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
5:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
6:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
7:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
8:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
9:: Siglent Technologies,SDG2122X,SDG2000X201701,2.01.01.23R1
>>
Press "Enter" to exit_
```

4.2.2 Telnet 示例

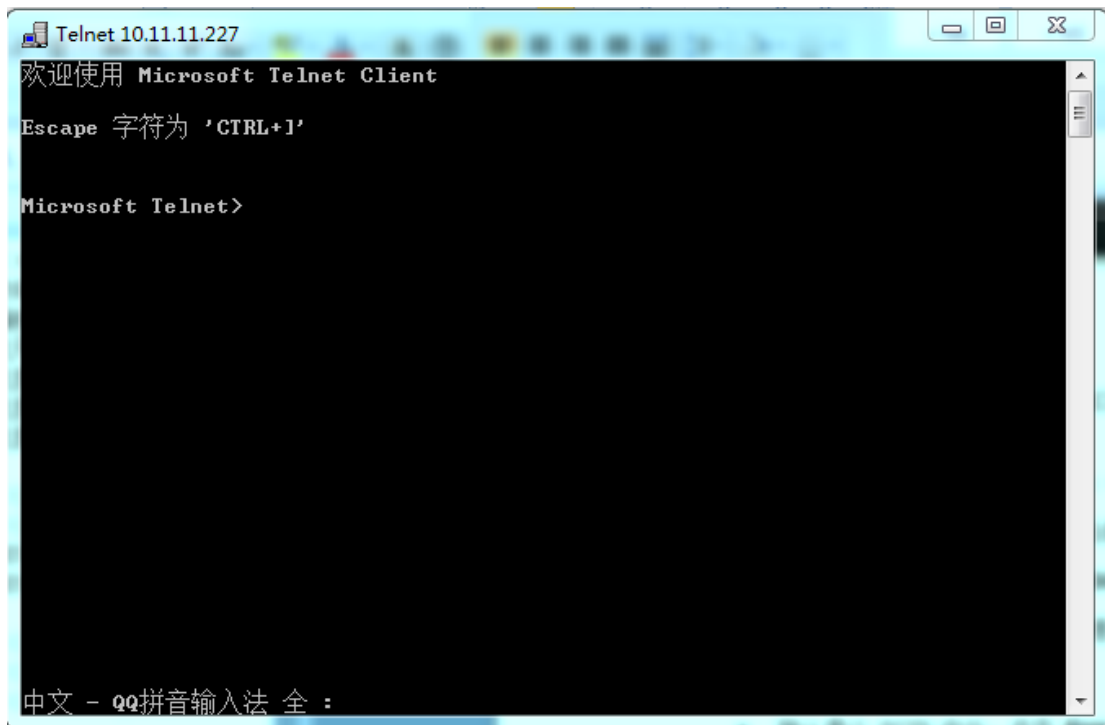
Telnet SCPI: 提供了使用局域网端口号5024将单个SCPI命令从远程PC发送到分析仪的功能。

如何使用Telnet发送单个SCPI命令:

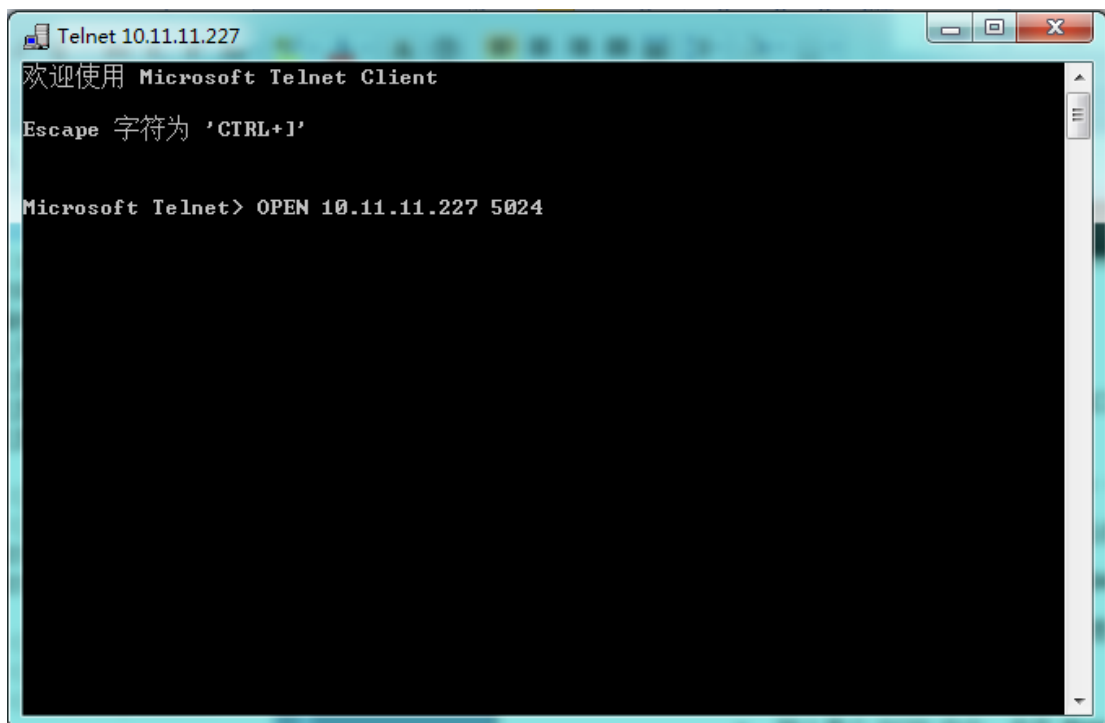
1. 在远程PC上, 点击 **开始**, 然后点击 **运行**
2. 输入: **telnet <ip address> 5024**
3. 远程PC屏幕上会出现带有 **>>** 提示符的Telnet窗口。
4. SCPI 提示符操作:
 - 输入单个 SCPI 命令。点击 **Enter** 发送命令。

如输入 ***IDN?** 点击 **Enter** 会回复设备ID信息:

- 点击右上角的 **X** 推出命令窗口。
- 要获得正常的telnet提示符, 请按 **Ctrl +]** (右括号)。



- 想要重新获得 SCPI 提示符，输入 **open** <ip Address> **5024** 点击 **Enter**。



- 想要关闭Telnet窗口，输入 **Quit** 然后按下 **Enter**。