

# SSG5000X 系列

## 射频信号发生器

编程手册

CN01A



深圳市鼎阳科技股份有限公司  
SIGLENT TECHNOLOGIES CO.,LTD



## 目录

<b>1</b>	<b>编程概述</b>	<b>1</b>
1.1	建立通信	1
1.1.1	NI-VISA 的安装	1
1.1.2	连接仪器	4
1.2	远程控制的实现	6
1.2.1	用户自定义程序	6
1.2.2	通过 NI-MAX 发送 SCPI 命令	6
1.2.3	通过 Telnet 发送 SCPI 命令	11
1.2.4	通过 Socket 发送 SCPI 命令	13
<b>2</b>	<b>SCPI 语言简介</b>	<b>14</b>
2.1	命令格式	14
2.2	命令符号	14
2.3	参数类型	15
2.4	命令缩写	16
<b>3</b>	<b>命令系统</b>	<b>17</b>
3.1	IEEE 通用命令子系统	17
3.1.1	设备信息查询 (*IDN?)	17
3.1.2	复位 (*RST)	17
3.1.3	清除状态 (*CLS)	17
3.1.4	标准事件状态使能 (*ESE)	18
3.1.5	标准事件状态寄存器询问 (*ESR?)	18
3.1.6	操作完成 (*OPC)	18
3.1.7	允许服务请求 (*SRE)	18
3.1.8	状态字节询问 (*STB?)	19
3.1.9	等待继续 (*WAI)	19
3.1.10	自测试询问 (*TST?)	19
3.2	SYSTEM 命令子系统	20
3.2.1	系统配置	20
3.2.2	系统复位	27
3.3	OUTPUT 命令子系统	30
3.3.1	RF 端口输出 (:OUTPut[:STATe])	30
3.3.2	模拟调制开关状态 (:OUTPut:MODulation[:STATe])	30

3.4	SOURCE 命令子系统.....	31
3.4.1	RF 端口输出 ([:SOURce]:OUTPut) .....	31
3.4.2	软件触发 ([:SOURce]:*TRG) .....	31
3.4.3	频率设置.....	31
3.4.4	幅度设置.....	33
3.4.5	扫描设置.....	41
3.4.6	功率计设置.....	52
3.4.7	模拟调制设置 .....	54
3.4.8	脉冲调制设置 .....	65
3.4.9	LF 源设置.....	76
3.4.10	LF 扫描设置 .....	78
3.4.11	系统复位.....	83
3.4.12	IQ 调制设置.....	84
3.4.13	Custom 调制设置.....	84
3.4.14	ARB 调制设置 .....	92
3.4.15	I/Q 设置.....	119
3.4.16	IoT 调制设置 .....	125
3.4.17	多音调制设置 .....	155
3.4.18	AWGN 调制设置 .....	157
3.5	SENSE 命令子系统 .....	158
3.5.1	功率计设置.....	158
3.6	MEMORY 命令子系统.....	168
3.6.1	ARB 调制设置 .....	168
<b>4</b>	<b>编程示例.....</b>	<b>170</b>
4.1	使用 VISA 的编程示例.....	170
4.1.1	VC++ 示例 .....	170
4.1.2	VB 示例.....	177
4.1.3	MATLAB 示例.....	182
4.1.4	LabVIEW 示例 .....	184
4.2	使用 SOCKET 的编程示例.....	187
4.2.1	Python 示例.....	187

# 1 编程概述

用户可以通过使用 SSG5000X 射频信号发生器的 USB、LAN 或 USB-GPIB 接口，并结合 NI-VISA 和程序语言，远程控制射频信号发生器。基于 LAN 接口，SSG5000X 支持 VXI-11、Sockets 和 Telnet 通信协议。本节介绍了如何建立射频信号发生器和计算机之间的通信，同时介绍了如何远程控制射频信号发生器。

## 1.1 建立通信

### 1.1.1 NI-VISA 的安装

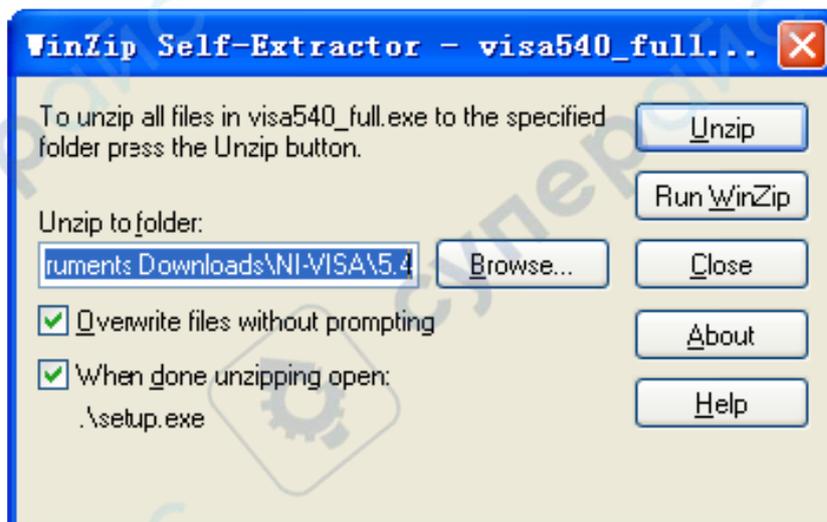
在编程之前，请确保正确安装 NI-VISA 软件的最新版本。

NI-VISA 是用于计算机与设备之间通信的通信库。NI 软件有两种有效 VISA 安装包：完整版和运行引擎版 (Run-Time Engine)。运行引擎版本提供 NI 设备驱动程序，例如 USB-TMC, VXI 和 GPIB 等，它主要用于远程控制。完整版本包括运行引擎和 NI MAX 工具，其中 NI MAX 是用于控制设备的用户界面。

您可以在 NI 官网上下载最新的 NI-VISA 运行引擎或完整版。它们的安装步骤基本相同。

请参阅以下步骤安装 NI-VISA (示例使用 NI-VISA5.4 完整版):

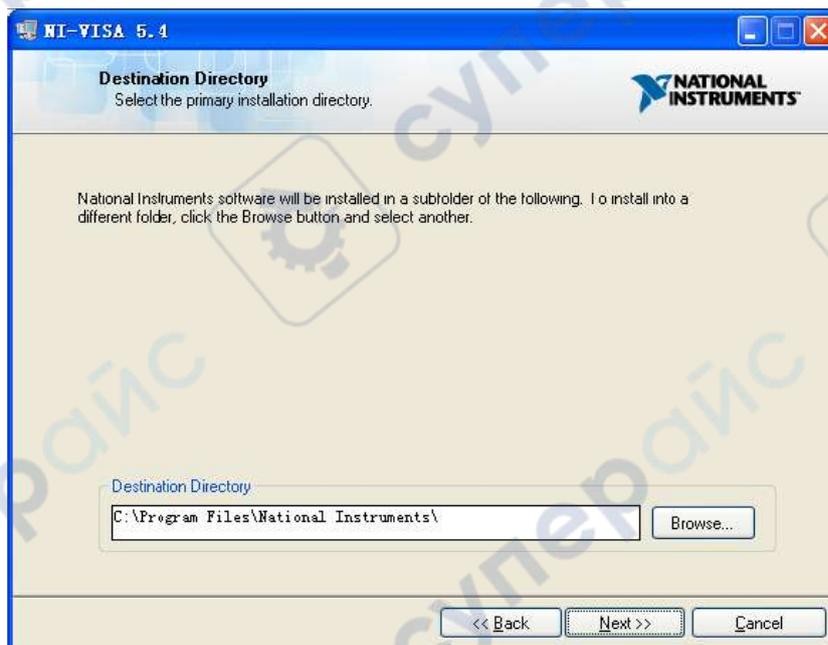
- a. 下载合适版本的 NI-VISA。
- b. 双击 visa540\_full.exe，弹出对话框如下：



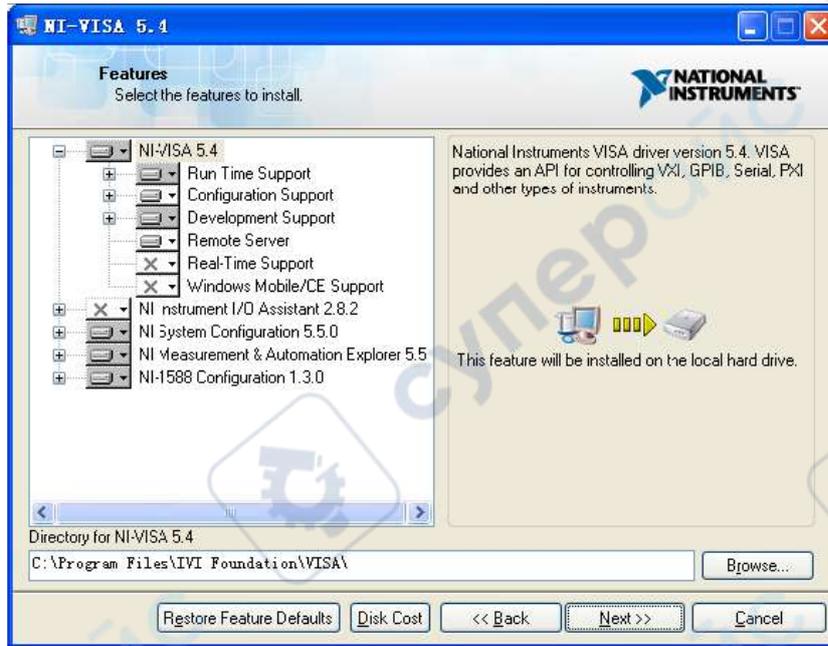
- c. 点击 Unzip 解压文件，当解压完成后，安装程序将自动执行。若您的计算机需要安装 .NET Framework4，则在安装过程会自动安装 .NET Framework4。



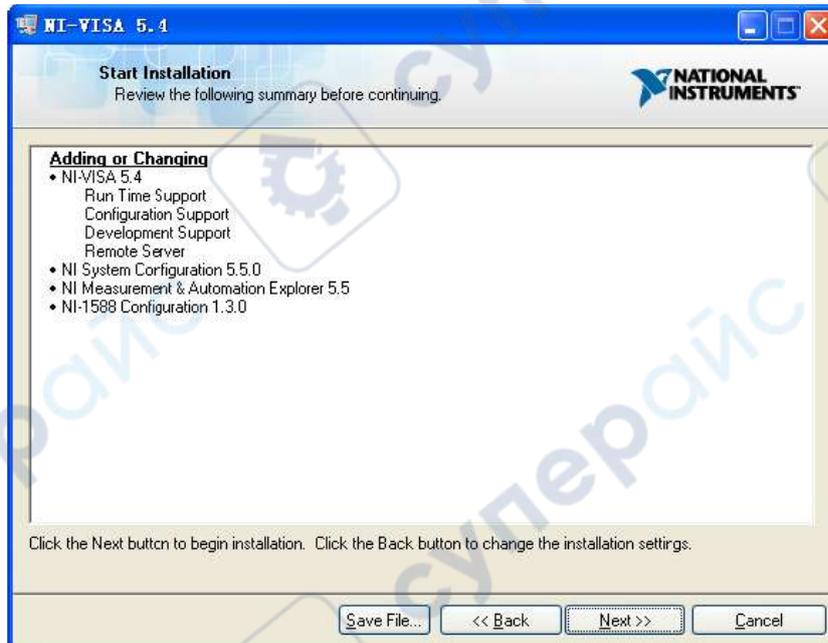
- d. NI-VISA 安装对话框如上图所示，点击 Next 开始安装过程。



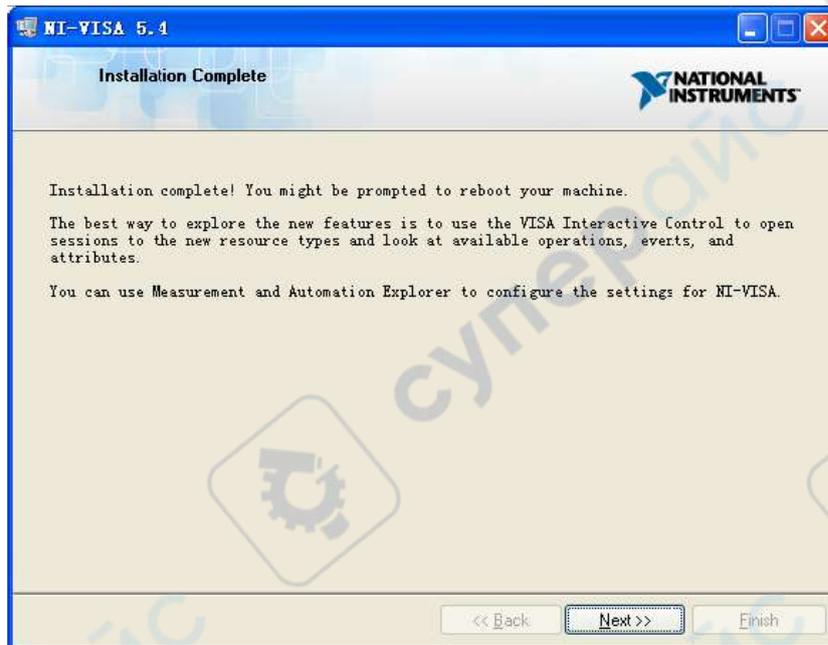
- e. 设置安装路径，默认路径为“C:\Program Files\National Instruments\”。您也可以修改安装路径。点击 Next，对话框如下图所示。



- f. 点击 Next 两次，在许可协议对话框下，选择 “I accept the above 2 License Agreement(s).” 并点击 Next，对话框如下图所示：



- g. 点击 Next 开始安装：



h. 安装完成后，重启电脑。

## 1.1.2 连接仪器

射频信号发生器能够通过 USB 接口、LAN 接口或 USB-GPIB 接口连接计算机。

### 1.1.2.1 使用 USB 接口连接

请参阅以下步骤实现射频信号发生器通过 USB 接口与计算机的连接：

1. 在您的计算机上安装 NI-VISA 以获得 USB-TMC 驱动。
2. 使用 USB A-B 电缆将射频信号发生器的 USB Device 接口连接到计算机的 USB Host 接口。
3. 打开射频信号发生器。

射频信号发生器将被自动检测为新的 USB 设备。

### 1.1.2.2 使用 LAN 接口连接

请参阅以下步骤实现射频信号发生器通过 LAN 接口与计算机的连接：

1. 在计算机上安装 NI-VISA 以获得 VXI 驱动。
2. 使用网络电缆将射频信号发生器连接到计算机或局域网。
3. 打开射频信号发生器。

4. 按下 **UTILITY** → 接口 进入 LAN 设置功能菜单。
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。



射频信号发生器将被自动或手动检测为新的 LAN 设备。

### 1.1.2.3 通过 USB-GPIB 接口连接

请参阅以下步骤实现射频信号发生器通过 USB-GPIB 接口与计算机的连接：

1. 在计算机上安装 NI-VISA 的 GPIB 驱动。
2. 使用 SIGLENT USB-GPIB 适配器将射频信号发生器的 USB Host 接口连接到计算机的 GPIB 接口：



3. 打开射频信号发生器。
4. 按下 **UTILITY** → 接口，在 GPIB 地址中输入 GPIB 编号。

射频信号发生器将被自动检测为新的 GPIB 设备。

## 1.2 远程控制的实现

### 1.2.1 用户自定义程序

用户可通过计算机发送 SCPI 命令实现编程和控制射频信号发生器。相关内容，请参阅“编程示例”章节中的介绍。

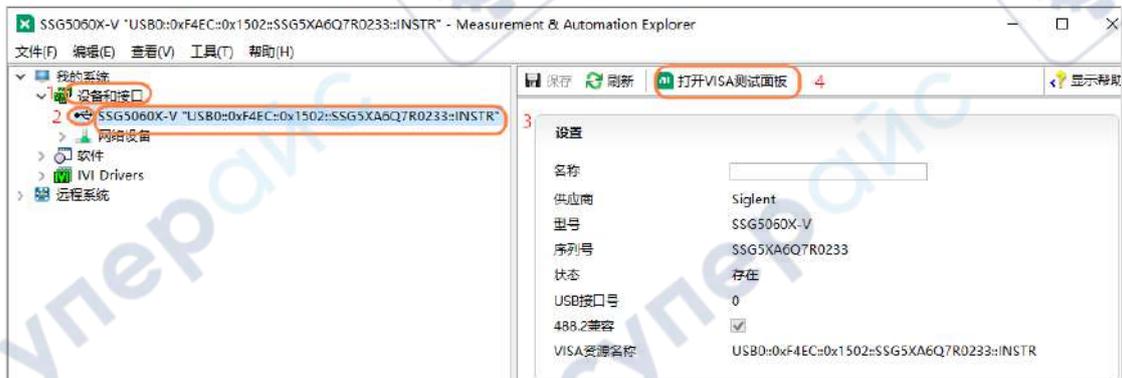
### 1.2.2 通过 NI-MAX 发送 SCPI 命令

NI-MAX 是由 NI 公司创建和维护的程序。它为 VXI、LAN、USB、GPIB 和串行通信提供基础的远程控制接口。用户可以通过 NI-MAX 发送 SCPI 命令远程控制射频信号发生器。

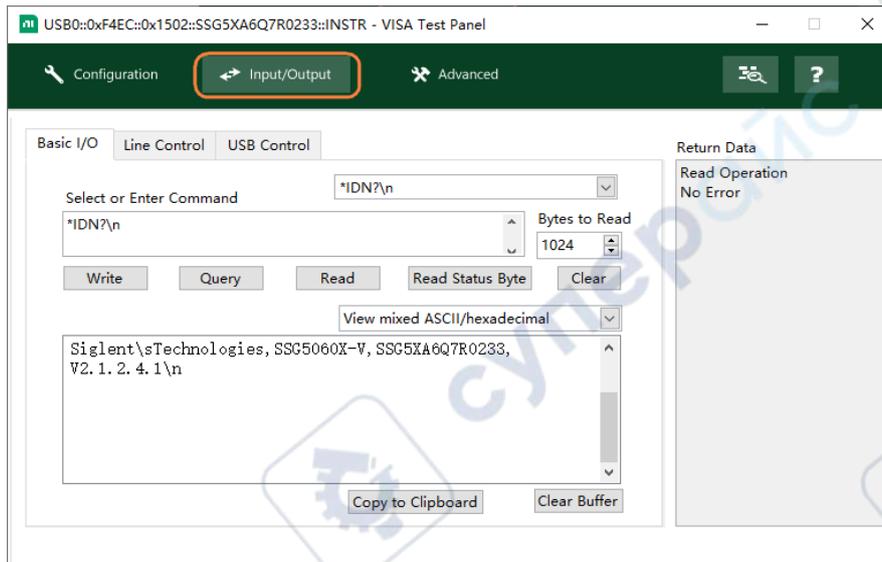
以下介绍在 NI-MAX 中通过 USB、LAN 或 GPIB 接口连接设备并发送向设备发送 SCPI 的步骤。

#### 1.2.2.1 使用 USB 接口

1. 运行 NI-MAX。
2. 点击软件左上角的“设备和接口”。
3. 找到 USBTMC 设备符号  下的射频信号发生器。
4. 选择射频信号发生器设备，并点击“打开 VISA 测试面板”按钮。

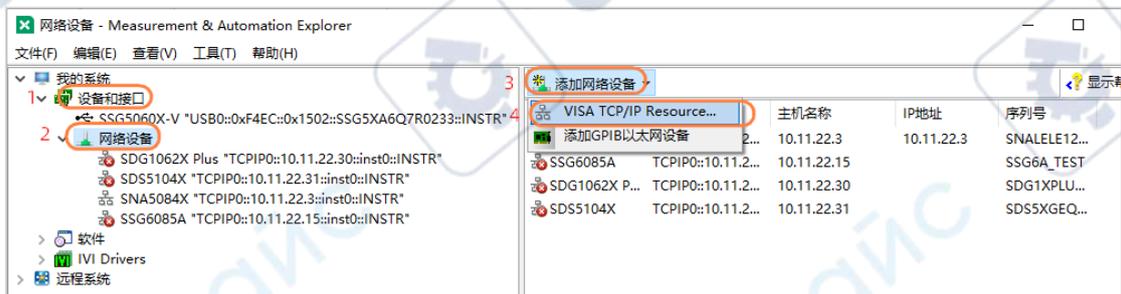


5. 在 VISA 测试面板中选择“Input/Output”页面，此时您可以在输入栏中输入 SCPI 进行写入或查询。如下图点击“Query”按钮查询设备的 IDN：

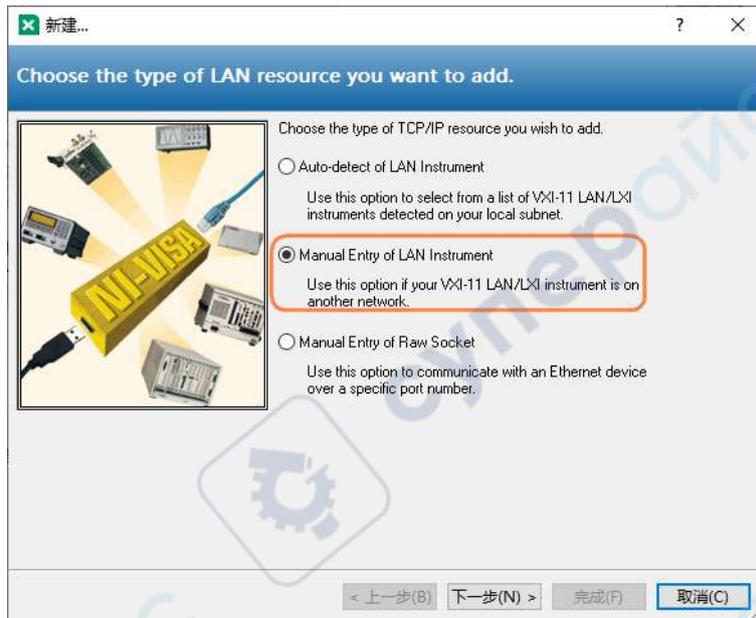


### 1.2.2.2 使用 LAN 接口

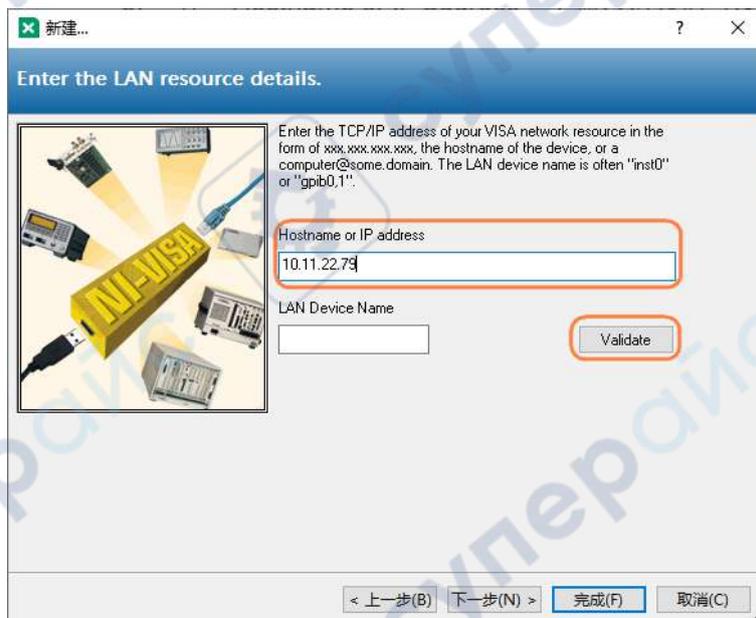
1. 运行 NI-MAX。
2. 点击软件左上角的“设备和接口” → “网络设备”。
3. 点击“添加网络设备”，然后点击“VISA TCP / IP Resource”：



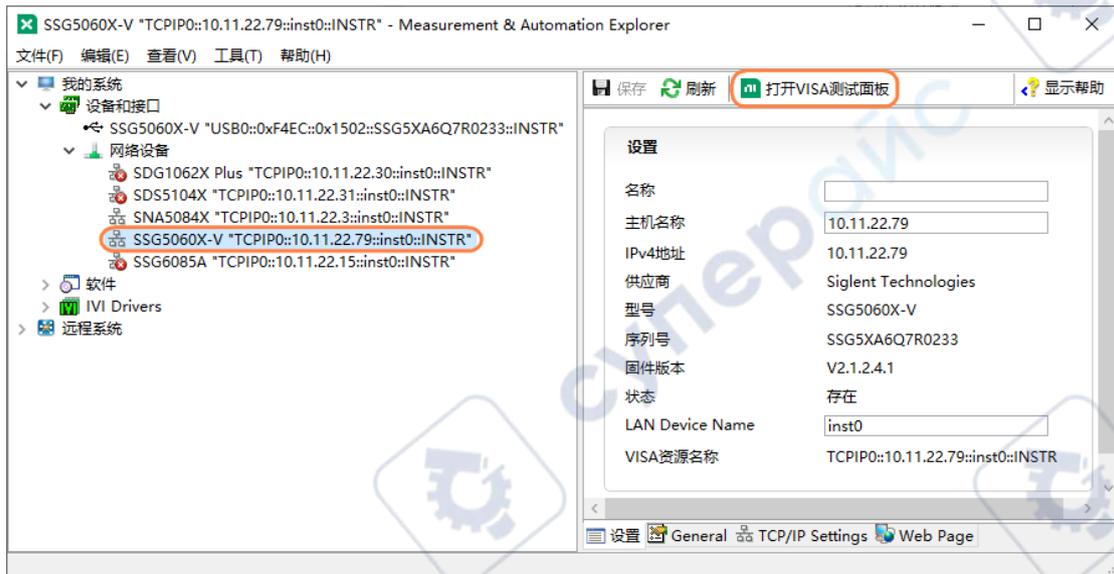
4. 在弹出的“新建...”窗口，选择“Manual Entry of LAN Instrument”，并点击下一步。



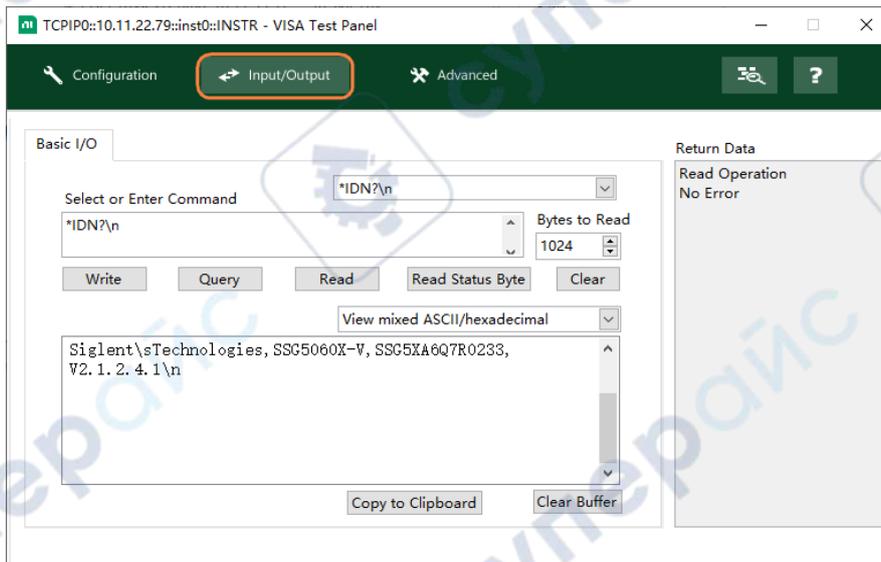
5. 在“Hostname or IP address”中输入射频信号发生器的 IP 地址。您可以点击“Validate”来验证能否通过输入的 IP 来连接设备。



6. 点击“完成”建立连接。
7. 短暂扫描后，射频信号发生器的资源名称应显示在“网络设备”下：

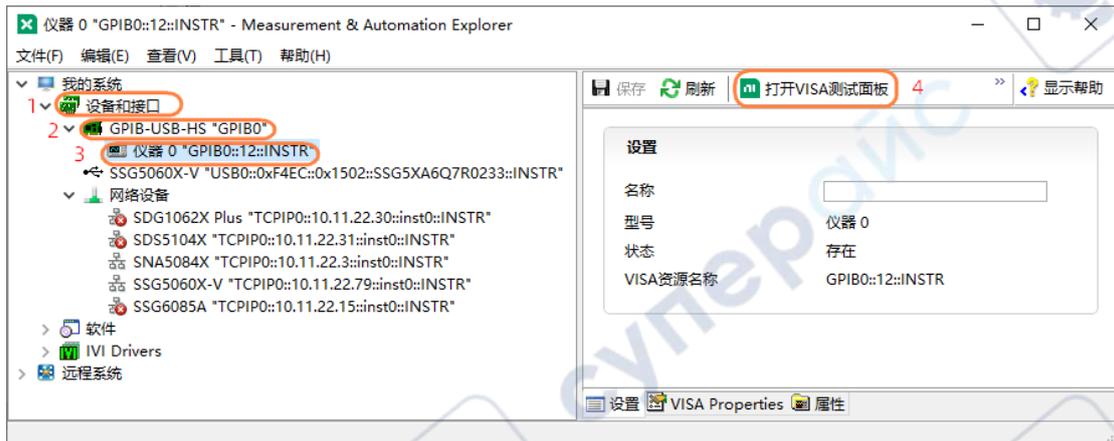


8. 选择射频信号发生器设备，并点击“打开 VISA 测试面板”按钮。
9. 在 VISA 测试面板中选择“Input/Output”页面，此时您可以在输入栏中输入 SCPI 进行写入或查询。如下图点击“Query”按钮查询设备的 IDN：

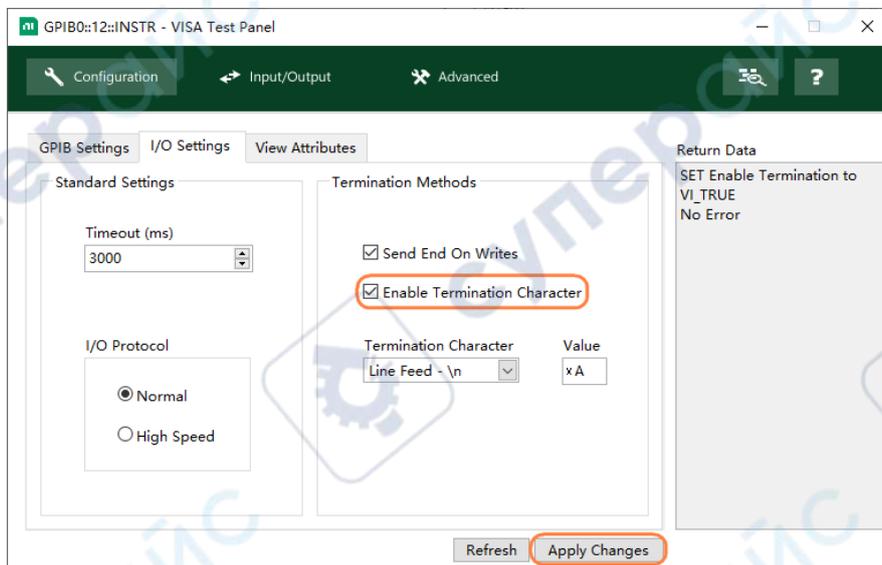


### 1.2.2.3 使用 USB-GPIB 接口

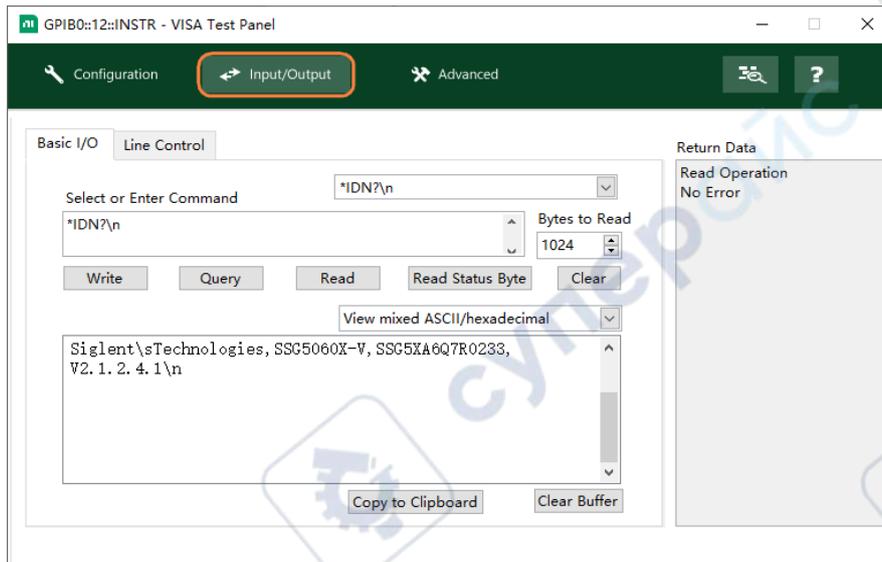
1. 运行 NI-MAX。
2. 点击软件左上角的“设备和接口”。
3. 找到“GPIB-USB-HS”设备符号下的射频信号发生器。
4. 选择射频信号发生器设备，并点击“打开 VISA 测试面板”按钮。



5. 在 VISA 测试面板中选择 “Configuration” > “I/O Settings”。勾选启用终止字符选项，然后单击应用更改按钮。



6. 在 VISA 测试面板中选择 “Input/Output” 页面，此时您可以在输入栏中输入 SCPI 进行写入或查询。如下图点击 “Query” 按钮查询设备的 IDN:



### 1.2.3 通过 Telnet 发送 SCPI 命令

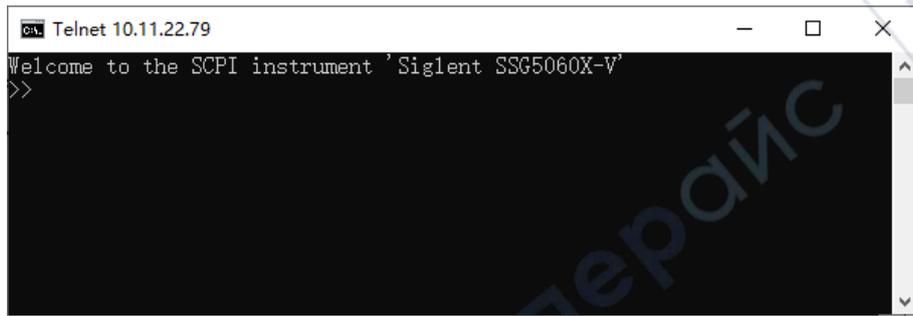
Telnet 提供一种通过 LAN 接口与射频信号发生器通信的方式。Telnet 协议支持从计算机向射频信号发生器发送 SCPI 命令，该方式类似于通过 USB 与射频信号发生器通信。发送和接受信息是交互的，一次只能发送一个命令。Windows 操作系统使用命令提示符样式接口作为 Telnet 客户端。

步骤如下：

1. 在计算机桌面，点击开始，然后右键选择运行。在运行窗口中输入 cmd 并点击确定，此时打开了命令提示符窗口。



2. 在命令提示符窗口，输入 telnet <ip address> 5024 并回车，将弹出可与仪器会话的 Telnet 窗口：



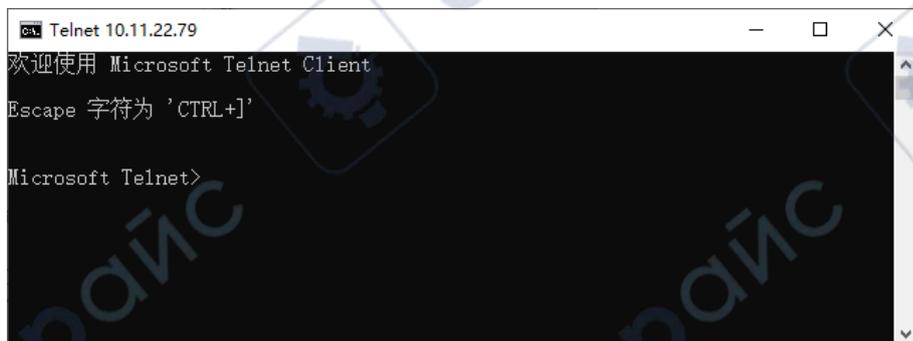
```
Telnet 10.11.22.79
Welcome to the SCPI instrument 'Siglent SSG5060X-V'
>>
```

- 在“>>”提示符后，您可以输入 SCPI 命令远程控制射频信号源。例如输入\*IDN?，此命令将返回公司名、机器型号、序列号和固件版本号。



```
Telnet 10.11.22.79
Welcome to the SCPI instrument 'Siglent SSG5060X-V'
>>*IDN?
Siglent Technologies, SSG5060X-V, SSG5XA6Q7R0233, V2.1.2.4.1
>>
```

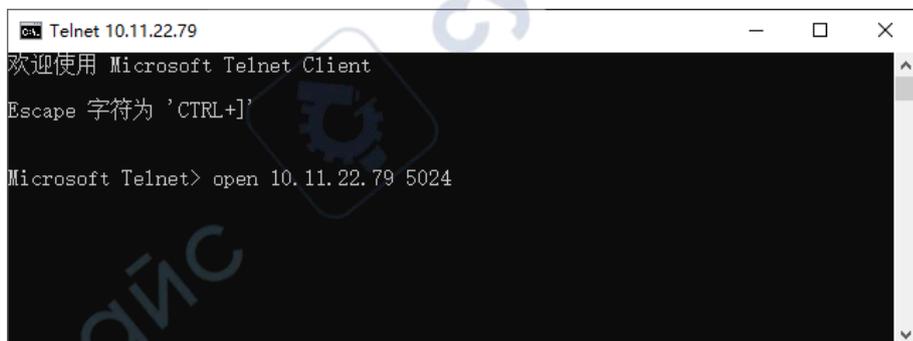
- 同时按下 Ctrl+] 键将退出与仪器的 SCPI 会话：



```
Telnet 10.11.22.79
欢迎使用 Microsoft Telnet Client
Escape 字符为 'CTRL+]'

Microsoft Telnet>
```

- 想要再次进入与仪器的 SCPI 会话，您可以输入 open <ip Address> 5024 并按回车：



```
Telnet 10.11.22.79
欢迎使用 Microsoft Telnet Client
Escape 字符为 'CTRL+]'

Microsoft Telnet> open 10.11.22.79 5024
```

- 想要关闭 Telnet 窗口，可在退出仪器的远程会话后，输入 Quit 并按回车：

```

管理员: C:\Windows\system32\cmd.exe
欢迎使用 Microsoft Telnet Client

Escape 字符为 'CTRL+]'

Microsoft Telnet> open 10.11.22.79 5024
正在连接10.11.22.79...
Microsoft Telnet> open 10.11.22.79 5024
正在连接10.11.22.79...
Microsoft Telnet> Quit

C:\Users\Administrator>

```

#### 1.2.4 通过 Socket 发送 SCPI 命令

Socket 接口可以在不安装其他库的情况下通过 LAN 接口控制射频信号发生器，它可以减少编程的复杂度。详细信息请查阅“编程示例”的“使用 Socket 的编程示例”章节。

SOCKET 地址	IP 地址+接口号
IP 地址	SSG IP 地址
接口号	5025

## 2 SCPI 语言简介

### 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：

```
:SOURce:FREQuency <freq>  
:SOURce:FREQuency?
```

SOURce 是命令的根关键字，FREQuency 是第二级关键字。命令行以冒号 “:” 开始，同时冒号将各级关键字分开。<freq>表示可设置的参数，命令:SOURce:FREQuency 和参数<freq>之间用 “空格” 分开。问号 “?” 表示查询，仪器收到查询命令后将返回一个响应字符串。

### 2.2 命令符号

下面是用在命令中的符号：

#### 1. 三角括号 < >

三角括号< >中的内容是命令参数，必须用一个有效值来替换。例如：

POWER:SPC:TARGet <power>命令，您可以发送为 POWER:SPC:TARGet 0。

#### 2. 方括号 [ ]

方括号中的内容（命令关键字或默认参数）是可省略的。如果省略方括号中的关键字，此命令仍可以产生相同的效果。如果省略方括号中的默认参数，该默认参数仍生效。

#### 3. 竖线 |

竖线用于分隔多个参数枚举值，发送命令时必须选择其中一个参数。例如：

在[:SOURce]:AM:STATe OFF|ON|0|1 命令中，可选择的命令参数为 “OFF”、“ON”、“0” 或 “1”。

#### 4. 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>}，大括号中的 {,<x-axis>,<ampl>}可以省略，也可以设置一对或多对频率、幅度参数。

## 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、浮点型、ASCII 字符串和离散型。

### 1. 布尔型

参数取值为“OFF”、“ON”、“0”或“1”。例如：

```
[:SOURce]:FM:STATe OFF|ON|0|1
```

### 2. 枚举

参数取值为所列举的值。例如：

```
[:SOURce]:SWEep:STATe OFF|FREQuency|LEVel|LEV_FREQ
```

参数为“OFF”、“FREQuency”、“LEVel”或“LEV\_FREQ”。

### 3. 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：

```
[:SOURce]:SWEep:STEP:POINts <value>
```

参数<value>可取 2 到 65535 范围内的任一整数。

### 4. 浮点型

参数可根据精度要求在有效范围内取任意值（通常默认精度为小数点后九位）。例如：

```
[:SOURce]:POWer:OFFSet <value>
```

参数<value>可取-100 到 100 之间的实数。

### 5. 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 IP 地址格式字符串。

### 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 SCPI 命令都对大小写不敏感。您可以以全大写或全小写的形式输入完整的命令。您还可以使用缩写，在这种情况下，缩写命令必须包含命令格式中的所有大写字母。

例如：

```
:CORRection:FLATness:COUNT?
```

您可以通过以下任意一种书写方式发送：

```
:CORRection:FLATness:COUNT?
```

```
:CORRECTION:FLATNESS:COUNT?
```

```
:correction:flatness:count?
```

您也可以缩写为：

```
:CORR:FLAT:COUN?
```

## 3 命令系统

本章主要介绍鼎阳 SSG5000X 系列射频信号发生器的 SCPI 命令系统。

### 3.1 IEEE 通用命令子系统

IEEE 标准定义的通用命令适用于查询设备的基本信息和执行基本操作。这些命令通常以 “\*” 开头以及命令的关键字长度为 3 个字符。

#### 3.1.1 设备信息查询 (\*IDN?)

命令格式	*IDN?
功能描述	用于获取设备信息字符串。字符串内容包括厂商，设备型号，设备序列号和软件版本号。
对应菜单	无
示例	<i>*IDN?</i> 返回值： <i>Siglent Technologies,SSG5060X-V,SSG5XA6Q7R0233, V2.1.2.4.1</i>

#### 3.1.2 复位 (\*RST)

命令格式	*RST
功能描述	重新恢复设备状态为初始状态。*RST 相当于执行两个命令：“:SOURce:PRESet” 和 “*CLS”。
对应菜单	无
示例	<i>*RST</i>

#### 3.1.3 清除状态 (\*CLS)

命令格式	*CLS
功能描述	将所有事件寄存器的值清零，同时清空错误列表。
对应菜单	无
示例	<i>*CLS</i>

### 3.1.4 标准事件状态使能 (\*ESE)

命令格式	*ESE <number> *ESE?
功能描述	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。
参数类型	整型
参数范围	0 ~ 255
对应菜单	无
示例	*ESE 16 *ESE? 返回值: 16\n

### 3.1.5 标准事件状态寄存器询问 (\*ESR?)

命令格式	*ESR?
功能描述	询问及清除标准事件状态寄存器的事件值。
对应菜单	无
示例	*ESR? 返回值: 0\n

### 3.1.6 操作完成 (\*OPC)

命令格式	*OPC *OPC?
功能描述	所有操作结束后，在标准事件状态寄存器中设置比特 0 为 1。 查询当前操作是否完成。
对应菜单	无
示例	*OPC *OPC? 返回值: 1\n

### 3.1.7 允许服务请求 (\*SRE)

命令格式	*SRE <integer> *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。
参数类型	整型
参数范围	0 ~ 255
对应菜单	无
示例	<i>*SRE 24</i> <i>*SRE?</i> 返回值: <i>24\n</i>

### 3.1.8 状态字节询问 (\*STB?)

命令格式	<i>*STB?</i>
功能描述	查询状态字节寄存器的事件值。
对应菜单	无
示例	<i>*STB?</i> 返回值: <i>72\n</i>

### 3.1.9 等待继续 (\*WAI)

命令格式	<i>*WAI</i>
功能描述	等待所有未完成操作完成之后，再执行任何其他命令。
对应菜单	无
示例	<i>*WAI</i>

### 3.1.10 自测试询问 (\*TST?)

命令格式	<i>*TST?</i>
功能描述	查询仪器自检结果。
对应菜单	无
示例	<i>*TST?</i> 返回值: <i>0\n</i>

## 3.2 SYSTem 命令子系统

### 3.2.1 系统配置

#### 3.2.1.1 系统时间 (:SYSTem:TIME)

命令格式	:SYSTem:TIME <hhmmss> :SYSTem:TIME?
功能描述	设置系统时间。 查询系统时间。
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
对应菜单	<b>UTILITY</b> > 设置 > 时间设置
示例	:SYSTem:TIME 182559 :SYSTem:TIME? 返回值: 182613\n

#### 3.2.1.2 系统日期 (:SYSTem:DATE)

命令格式	:SYSTem:DATE <yyyymmdd> :SYSTem:DATE?
功能描述	设置系统日期。 查询系统日期。
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
对应菜单	<b>UTILITY</b> > 设置 > 时间设置
示例	:SYSTem:DATE 20050101 :SYSTem:DATE? 返回值: 20050101\n

### 3.2.1.3 IP 地址 (:SYSTem:COMMunicate:LAN:IPADdress)

命令格式	:SYSTem:COMMunicate:LAN:IPADdress <"xxx.xxx.xxx.xxx"> :SYSTem:COMMunicate:LAN:IPADdress?
功能描述	当 IP 分配类型为静态时, 设置仪器的 IP 地址。 查询仪器的 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	字符串
默认值	无
对应菜单	<b>UTILITY</b> > 接口 > LAN 设置 > IP 地址
示例	:SYSTem:COMMunicate:LAN:IPADdress "192.168.1.12" :SYSTem:COMMunicate:LAN:IPADdress? 返回值: "192.168.1.12" "

### 3.2.1.4 网关 (:SYSTem:COMMunicate:LAN:GATeway)

命令格式	:SYSTem:COMMunicate:LAN:GATeway <"xxx.xxx.xxx.xxx"> :SYSTem:COMMunicate:LAN:GATeway?
功能描述	设置仪器的网关。 查询仪器的网关。
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	字符串
默认值	无
对应菜单	<b>UTILITY</b> > 接口 > LAN 设置 > 网关
示例	:SYSTem:COMMunicate:LAN:GATeway "192.168.1.1" :SYSTem:COMMunicate:LAN:GATeway? 返回值: "192.168.1.1" "

### 3.2.1.5 子网掩码 (:SYSTem:COMMunicate:LAN:SMASK)

命令格式	:SYSTem:COMMunicate:LAN:SMASK <"xxx.xxx.xxx.xxx"> :SYSTem:COMMunicate:LAN:SMASK?
功能描述	设置仪器的子网掩码。 查询仪器的子网掩码。

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	字符串
默认值	无
对应菜单	<b>UTILITY</b> > 接口 > LAN 设置 > 子网掩码
示例	:SYSTem:COMMunicate:LAN:SMASK "255.255.255.0" :SYSTem:COMMunicate:LAN:SMASK? 返回值: "255.255.255.0"␣

### 3.2.1.6 IP 分配类型 (:SYSTem:COMMunicate:LAN:TYPE)

命令格式	:SYSTem:COMMunicate:LAN:TYPE STATIC DHCP :SYSTem:COMMunicate:LAN:TYPE?
功能描述	设置仪器的 IP 分配类型。 查询仪器的 IP 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
对应菜单	<b>UTILITY</b> > 接口 > LAN 设置 > DHCP 状态
示例	:SYSTem:COMMunicate:LAN:TYPE STATIC :SYSTem:COMMunicate:LAN:TYPE? 返回值: STATIC␣

### 3.2.1.7 系统语言 (:SYSTem:LANGuage)

命令格式	:SYSTem:LANGuage CHINese ENGLish :SYSTem:LANGuage?
功能描述	设置系统语言。 查询系统语言。
参数类型	枚举
参数范围	CHINese: 中文, ENGLish: 英文。

返回值	枚举
默认值	无
对应菜单	<b>UTILITY</b> > 设置 > Language
示例	<code>:SYSTem:LANGUage CHINese</code> <code>:SYSTem:LANGUage?</code> 返回值: <code>CHINeseIn</code>

### 3.2.1.8 屏保 (:SYSTem:SCReen:SAVer)

命令格式	<code>:SYSTem:SCReen:SAVer</code> <code>OFF 10S 1MIN 5MIN 15MIN 30MIN 1HOUR 2HOUR</code> <code>:SYSTem:SCReen:SAVer?</code>
功能描述	设置系统屏保类型。 查询系统屏保类型。
参数类型	枚举
参数范围	OFF: 关闭屏保, 10S 1MIN 5MIN 15MIN 30MIN 1HOUR 2HOUR: 设置不同屏保时间。
返回值	枚举
默认值	OFF
对应菜单	<b>UTILITY</b> > 设置 > 屏保
示例	<code>:SYSTem:SCReen:SAVer 30MIN</code> <code>:SYSTem:SCReen:SAVer?</code> 返回值: <code>30MINIn</code>

### 3.2.1.9 蜂鸣器 (:SYSTem:ALARm)

命令格式	<code>:SYSTem:ALARm ON OFF 1 0</code> <code>:SYSTem:ALARm?</code>
功能描述	设置系统蜂鸣器开关状态。 查询系统蜂鸣器开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	无
对应菜单	<b>UTILITY</b> > 设置 > 蜂鸣器

示例	<code>:SYSTem:ALARm ON</code> <code>:SYSTem:ALARm?</code> 返回值: <code>1\n</code>
----	--

### 3.2.1.10 启动设置 (:SYSTem:PON:TYPE)

命令格式	<code>:SYSTem:PON:TYPE DFT LAST</code> <code>:SYSTem:PON:TYPE?</code>
功能描述	设置系统的启动类型。 查询系统的启动类型。
参数类型	枚举
参数范围	DFT: 默认, LAST: 上次。
返回值	枚举
默认值	无
对应菜单	<b>UTILITY</b> > 设置 > 启动设置
示例	<code>:SYSTem:PON:TYPE LAST</code> <code>:SYSTem:PON:TYPE?</code> 返回值: <code>LAST\n</code>

### 3.2.1.11 上电开机 (:SYSTem:POWeron:TYPE)

命令格式	<code>:SYSTem:POWeron:TYPE ON OFF 1 0</code> <code>:SYSTem:POWeron:TYPE?</code>
功能描述	设置系统是否上电开机。 查询系统上电开机状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>UTILITY</b> > 设置 > 上电开机
示例	<code>:SYSTem:POWeron:TYPE ON</code> <code>:SYSTem:POWeron:TYPE?</code> 返回值: <code>1\n</code>

## 3.2.1.12 参考校正 (:SYSTem:REF:DAC:STAT)

命令格式	:SYSTem:REF:DAC:STAT ON OFF 1 0 :SYSTem:REF:DAC:STAT?
功能描述	设置系统的参考校正状态。 查询系统的参考校正状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>UTILITY</b> > 设置 > 参考校正
示例	:SYSTem:REF:DAC:STAT ON :SYSTem:REF:DAC:STAT? 返回值: 1\n

## 3.2.1.13 参考振荡器码字 (:SYSTem:REF:DAC)

命令格式	:SYSTem:REF:DAC <value> :SYSTem:REF:DAC?
功能描述	设置系统的参考振荡器码字。 查询系统的参考振荡器码字。
参数类型	整型
参数范围	0 ~ 65535
返回值	整型
默认值	无
对应菜单	<b>UTILITY</b> > 设置 > 参考校正 > 参考振荡器码字
示例	:SYSTem:REF:DAC 43000 :SYSTem:REF:DAC? 返回值: 43000\n

## 3.2.1.14 保存参考振荡器设置 (:SYSTem:REF:DAC:SAVE)

命令格式	:SYSTem:REF:DAC:SAVE <"file_name">
功能描述	把系统的参考振荡器码字保存进 DAC 文件。
参数类型	字符串

参数范围	无
对应菜单	<b>UTILITY</b> > 设置 > 参考校正 > 保存参考振荡器设置
示例	<code>:SYSTem:REF:DAC:SAVE "U-disk3/test.dac"</code>

### 3.2.1.15 加载参考振荡器设置 (:SYSTem:REF:DAC:LOAD)

命令格式	<code>:SYSTem:REF:DAC:LOAD &lt;"file_name"&gt;</code>
功能描述	从 DAC 文件中加载参考振荡器码字。
参数类型	字符串
参数范围	无
对应菜单	<b>UTILITY</b> > 设置 > 参考校正 > 加载参考振荡器设置
示例	<code>:SYSTem:REF:DAC:LOAD "U-disk3/test.dac"</code>

### 3.2.1.16 重置参考振荡器码字 (:SYSTem:REF:DAC:DEfault)

命令格式	<code>:SYSTem:REF:DAC:DEfault</code>
功能描述	恢复参考振荡器码字为默认值。
对应菜单	<b>UTILITY</b> > 设置 > 参考校正 > 重置为默认值
示例	<code>:SYSTem:REF:DAC:DEfault</code>

### 3.2.1.17 GPIB 地址 (:SYSTem:GPIB)

命令格式	<code>:SYSTem:GPIB &lt;value&gt;</code> <code>:SYSTem:GPIB?</code>
功能描述	设置系统的 GPIB 地址。 查询系统的 GPIB 地址。
参数类型	整型
参数范围	1 ~ 30
返回值	整型
默认值	18
对应菜单	<b>UTILITY</b> > 接口 > GPIB 设置 > GPIB 地址
示例	<code>:SYSTem:GPIB 10</code> <code>:SYSTem:GPIB?</code> 返回值: <code>10\n</code>

### 3.2.1.18 退出远程模式 (:SYSTem:REMOte 0)

命令格式	:SYSTem:REMOte 0
功能描述	退出系统的远程模式。
对应菜单	<input type="button" value="ESC"/>
示例	:SYSTem:REMOte 0

## 3.2.2 系统复位

### 3.2.2.1 系统复位 (:SYSTem:PRESet)

命令格式	:SYSTem:PRESet
功能描述	根据复位类型，复位仪器的参数配置。
对应菜单	<input type="button" value="UTILITY"/> > 复位，或 <input type="button" value="PRESET"/>
示例	<p>复位仪器为默认配置：</p> <pre>:SYSTem:PRESet:TYPE DFT :SYSTem:PRESet</pre> <p>复位仪器为当前配置：</p> <pre>:SYSTem:PRESet:TYPE USER :SYSTem:PRESet:SAVE :SYSTem:PRESet</pre> <p>复位仪器为 XML 文件中的配置：</p> <pre>:SYSTem:PRESet:TYPE USER :SYSTem:PRESet:PATH "Local/test.xml" :SYSTem:PRESet</pre>

### 3.2.2.2 保存系统配置 (:SYSTem:PRESet:SAVE)

命令格式	:SYSTem:PRESet:SAVE
功能描述	保存当前的系统配置。
对应菜单	无
示例	<p>复位仪器为当前配置：</p> <pre>:SYSTem:PRESet:TYPE USER :SYSTem:PRESet:SAVE :SYSTem:PRESet</pre>

## 3.2.2.3 保存系统配置 (:SYSTem:PRESet:PATH)

命令格式	:SYSTem:PRESet:PATH <"file_name">
功能描述	把当前的系统配置保存进 XML 文件中。
参数类型	字符串
参数范围	无
对应菜单	<b>UTILITY</b> > 设置 > 复位类型 (用户) > 保存
示例	:SYSTem:PRESet:PATH "Local/test.xml" :SYSTem:PRESet:PATH "U-disk1/test.xml"

## 3.2.2.4 复位类型 (:SYSTem:PRESet:TYPE)

命令格式	:SYSTem:PRESet:TYPE DFT USER :SYSTem:PRESet:TYPE?
功能描述	设置系统的复位类型。 查询系统的复位类型。
参数类型	枚举
参数范围	DFT: 默认, USER: 用户。
返回值	枚举
默认值	DFT
对应菜单	<b>UTILITY</b> > 设置 > 复位类型
示例	:SYSTem:PRESet:TYPE DFT :SYSTem:PRESet:TYPE? 返回值: DFTn

## 3.2.2.5 恢复出厂设置 (:SYSTem:FDEFault)

命令格式	:SYSTem:FDEFault
功能描述	将仪器状态恢复为出厂设置。
对应菜单	<b>UTILITY</b> > 设置 > 恢复出厂设置
示例	:SYSTem:FDEFault

### 3.2.2.6 复位清除 (:SYSTem:RESet:CLEar)

命令格式	:SYSTem:RESet:CLEar
功能描述	将仪器状态恢复为出厂设置，同时清空 Local 文件夹下的用户文件。
对应菜单	<span style="border: 1px solid black; padding: 2px;">UTILITY</span> > 设置 > 复位清除
示例	:SYSTem:RESet:CLEar

### 3.3 OUTPut 命令子系统

#### 3.3.1 RF 端口输出 (:OUTPut[:STATe])

命令格式	:OUTPut[:STATe] ON OFF 1 0 :OUTPut[:STATe]?
功能描述	设置 RF 端口的输出状态。 查询 RF 端口的输出状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
等效命令	[:SOURce]:OUTPut ON OFF 1 0
对应菜单	<b>RF ON/OFF</b> 或 <b>FREQ</b> > RF 开关
示例	:OUTPut ON :OUTPut? 返回值: 1\n

#### 3.3.2 模拟调制开关状态 (:OUTPut:MODulation[:STATe])

命令格式	:OUTPut:MODulation[:STATe] ON OFF 1 0 :OUTPut:MODulation[:STATe]?
功能描述	设置模拟调制的开关状态。 查询模拟调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
等效命令	[:SOURce]:MODulation ON OFF 1 0 [:SOURce]:MODulation?
对应菜单	ANALOG MOD > On
示例	:OUTPut:MODulation ON :OUTPut:MODulation? 返回值: 1\n

## 3.4 SOURce 命令子系统

### 3.4.1 RF 端口输出 ([:SOURce]:OUTPut)

命令格式	[:SOURce]:OUTPut ON OFF 1 0
功能描述	设置 RF 端口的输出状态。
参数类型	布尔型
参数范围	ON OFF 1 0
等效命令	:OUTPut[:STATe] ON OFF 1 0
对应菜单	<b>RF ON/OFF</b> 或 <b>FREQ</b> > RF 开关
示例	<i>:SOURce:OUTPut ON</i>

### 3.4.2 软件触发 ([:SOURce]:\*TRG)

命令格式	[:SOURce]:*TRG
功能描述	当触发源是总线（涉及的触发功能有扫描触发、扫描点触发、LF 扫描触发、脉冲调制触发、ARB 触发和 IoT 触发等）时，执行软件触发。
对应菜单	无
示例	<i>*TRG</i>

### 3.4.3 频率设置

#### 3.4.3.1 显示频率 ([:SOURce]:FREQuency:DISPlay)

命令格式	[:SOURce]:FREQuency:DISPlay <freq> [:SOURce]:FREQuency:DISPlay?
功能描述	设置屏幕上方参数栏的频率显示值。 查询屏幕上方参数栏的频率显示值。
参数类型	浮点型， 单位：Hz, kHz, MHz 或 GHz, 默认为 Hz
返回值	浮点型，单位：Hz
默认值	最大频率值
对应菜单	频率
示例	<i>:FREQuency:DISPlay 2 MHz</i> <i>:FREQuency:DISPlay?</i> 返回值： <i>2000000\n</i>

## 3.4.3.2 频率 ([:SOURce]:FREQuency)

命令格式	<code>[:SOURce]:FREQuency &lt;freq&gt;</code> <code>[:SOURce]:FREQuency?</code>
功能描述	设置 RF 输出信号的频率。 查询 RF 输出信号的频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
返回值	浮点型, 单位: Hz
默认值	最大频率值
对应菜单	<b>FREQ</b> > 频率
示例	<code>:FREQuency 2 MHz</code> <code>:FREQuency?</code> 返回值: <code>2000000\n</code>

## 3.4.3.3 频率偏移 ([:SOURce]:FREQuency:OFFSet)

命令格式	<code>[:SOURce]:FREQuency:OFFSet &lt;freq&gt;</code> <code>[:SOURce]:FREQuency:OFFSet?</code>
功能描述	设置 RF 输出的频率偏移。 查询 RF 输出的频率偏移。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	-200 GHz ~ 200 GHz
返回值	浮点型, 单位: Hz
默认值	0 Hz
对应菜单	<b>FREQ</b> > 频率偏移
示例	<code>:FREQuency:OFFSet -2 MHz</code> <code>:FREQuency:OFFSet?</code> 返回值: <code>-2000000\n</code>

## 3.4.3.4 相位偏移 ([:SOURce]:PHASe)

命令格式	<code>[:SOURce]:PHASe &lt;phase&gt;</code> <code>[:SOURce]:PHASe?</code>
功能描述	设置 RF 信号的相位。 查询 RF 信号的相位。

参数类型	浮点型, 单位: 度 (°)
参数范围	-360 ~ 360
返回值	浮点型, 单位: 度 (°)
默认值	0
对应菜单	<b>FREQ</b> > 相位偏移
示例	<i>PHASe 20</i> <i>PHASe?</i> 返回值: <i>20\n</i>

#### 3.4.3.5 重置相位 ([:SOURce]:PHASe:RESet)

命令格式	[:SOURce]:PHASe:RESet
功能描述	将 RF 信号的相位偏移显示值重置为 0。
等效命令	[:SOURce]:PHASe:REF
对应菜单	<b>FREQ</b> > 重置相位差显示值
示例	<i>:PHASe:RESet</i>

#### 3.4.3.6 重置相位 ([:SOURce]:PHASe:REF)

命令格式	[:SOURce]:PHASe:REF
功能描述	将 RF 信号的相位偏移显示值重置为 0。
等效命令	[:SOURce]:PHASe:RESet
对应菜单	<b>FREQ</b> > 重置相位差显示值
示例	<i>:PHASe:REF</i>

### 3.4.4 幅度设置

#### 3.4.4.1 显示幅度 ([:SOURce]:POWer:POWer)

命令格式	[:SOURce]:POWer:POWer <power> [:SOURce]:POWer:POWer?
功能描述	设置屏幕上方参数栏的幅度显示值。 查询屏幕上方参数栏的幅度显示值。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm

返回值	浮点型, 单位: dBm
默认值	-130 dBm
对应菜单	幅度
示例	:POWer:POWer -2 :POWer:POWer? 返回值: -2n

#### 3.4.4.2 幅度 ([:SOURce]:POWer)

命令格式	[:SOURce]:POWer <power> [:SOURce]:POWer?
功能描述	设置 RF 输出信号的幅度。 查询 RF 输出信号的幅度。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	请参考数据手册
返回值	浮点型, 单位: dBm
默认值	-130 dBm
对应菜单	<input type="text" value="LEVEL"/> > 幅度
示例	:POWer 2 :POWer? 返回值: 2n

#### 3.4.4.3 幅度 ([:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude])

命令格式	[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude] <power> [:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]?
功能描述	设置 RF 输出信号的幅度。 查询 RF 输出信号的幅度。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	请参考数据手册
返回值	浮点型, 单位: dBm
默认值	-130 dBm
对应菜单	<input type="text" value="LEVEL"/> > 幅度

示例            :*POWer:LEVel -5*  
                   :*POWer:LEVel?*  
 返回值:  
                   -5*\n*

#### 3.4.4.4 幅度偏移 ([:SOURce]:POWer:OFFSet)

命令格式	[:SOURce]:POWer:OFFSet <power> [:SOURce]:POWer:OFFSet?
功能描述	设置 RF 输出的幅度偏移。 查询 RF 输出的幅度偏移。
参数类型	浮点型, 单位: dB
参数范围	-100 ~ 100
返回值	浮点型, 单位: dB
默认值	0
对应菜单	<input type="text" value="LEVEL"/> > 幅度偏移
示例	: <i>POWer:OFFSet 2</i> : <i>POWer:OFFSet?</i> 返回值: 2 <i>\n</i>

#### 3.4.4.5 ALC 状态 ([:SOURce]:POWer:ALC)

命令格式	[:SOURce]:POWer:ALC ON OFF AUTO [:SOURce]:POWer:ALC?
功能描述	设置 ALC 的状态。 查询 ALC 的状态。
参数类型	枚举
参数范围	ON OFF AUTO
返回值	ON OFF AUTO
默认值	AUTO
对应菜单	<input type="text" value="LEVEL"/> > ALC 状态
示例	: <i>POWer:ALC ON</i> : <i>POWer:ALC?</i> 返回值: ON <i>\n</i>

## 3.4.4.6 平坦度状态 ([:SOURce]:CORRection[:FLATness])

命令格式	[:SOURce]:CORRection[:FLATness] ON OFF 1 0 [:SOURce]:CORRection[:FLATness]?
功能描述	设置平坦度校正的开启状态。 查询平坦度校正的开启状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>LEVEL</b> > 平坦度
示例	<code>:CORRection:FLATness ON</code> <code>:CORRection?</code> 返回值: <code>1</code>

## 3.4.4.7 平坦度列表添加行 ([:SOURce]:CORRection:FLATness:PAIR)

命令格式	[:SOURce]:CORRection:FLATness:PAIR <freq>,<power>
功能描述	在平坦度列表中添加一行。
参数类型	频率：浮点型，单位：Hz，kHz，MHz 或 GHz，默认为 Hz， 幅度：浮点型，单位：dB
参数范围	频率：全频率范围， 幅度：-100 ~ 100
对应菜单	<b>LEVEL</b> > 平坦度 > 添加
示例	<code>CORRection:FLATness:PAIR 3 MHz,-3</code>

## 3.4.4.8 平坦度列表删除行 ([:SOURce]:CORRection:FLATness:DELeTe)

命令格式	[:SOURce]:CORRection:FLATness:DELeTe <row>
功能描述	从平坦度列表中删除指定的行。
参数类型	整型
参数范围	1 ~ 平坦度列表的总行数
对应菜单	<b>LEVEL</b> > 平坦度 > 删除
示例	<code>:CORRection:FLATness:DELeTe 1</code>

#### 3.4.4.9 平坦度列表行数 ([:SOURCE]:CORREction:FLATness:COUNT?)

命令格式	[:SOURCE]:CORREction:FLATness:COUNT?
功能描述	查询平坦度列表的总行数。
返回值	整型
默认值	0
对应菜单	<input type="text" value="LEVEL"/> > 平坦度
示例	:CORREction:FLATness:COUNT? 返回值: 5\n

#### 3.4.4.10 保存平坦度列表 ([:SOURCE]:CORREction:STORE)

命令格式	[:SOURCE]:CORREction:STORE <"file_name">
功能描述	将平整度列表保存到 UFLT 文件。
参数类型	字符串
参数范围	无
对应菜单	<input type="text" value="LEVEL"/> > 平坦度 > 保存
示例	:CORREction:STORE "U-disk3/test.uflt"

#### 3.4.4.11 加载平坦度列表 ([:SOURCE]:CORREction:LOAD)

命令格式	[:SOURCE]:CORREction:LOAD <"file_name">
功能描述	从 UFLT 文件中加载平整度列表。
参数类型	字符串
参数范围	无
对应菜单	<input type="text" value="LEVEL"/> > 平坦度 > 加载
示例	:CORREction:LOAD "U-disk3/test.uflt"

#### 3.4.4.12 清空平坦度列表 ([:SOURCE]:CORREction:FLATness:PRESet)

命令格式	[:SOURCE]:CORREction:FLATness:PRESet
功能描述	清空平坦度列表。
对应菜单	<input type="text" value="LEVEL"/> > 平坦度 > 清空
示例	:CORREction:FLATness:PRESet

## 3.4.4.13 平坦度列表填充类型 ([:SOURce]:CORRection:FLATness:FILL:TYPE)

命令格式	[:SOURce]:CORRection:FLATness:FILL:TYPE FLATness MANUal SWEEPlist [:SOURce]:CORRection:FLATness:FILL:TYPE?
功能描述	设置平坦度列表的填充类型。 查询平坦度列表的填充类型。
参数类型	枚举
参数范围	FLATness MANUal SWEEPlist
返回值	枚举
默认值	FLATness
对应菜单	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式
示例	<i>:CORRection:FLATness:FILL:TYPE FLATness</i> <i>:CORRection:FLATness:FILL:TYPE?</i> 返回值: <i>FLATness\n</i>

## 3.4.4.14 步进填充开始频率 ([:SOURce]:CORRection:FLATness:STARTfreq)

命令格式	[:SOURce]:CORRection:FLATness:STARTfreq <freq> [:SOURce]:CORRection:FLATness:STARTfreq?
功能描述	当需要用功率计填充平整度列表且填充类型为“步进填充”时，设置步进填充的起始频率。 查询步进填充的起始频率。
参数类型	浮点型， 单位：Hz，kHz，MHz 或 GHz，默认为 Hz
参数范围	全频率范围
返回值	浮点型，单位：Hz
默认值	最大频率值
对应菜单	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式 > 步进填充 > 开始频率
示例	<i>:CORRection:FLATness:STARTfreq 200 MHz</i> <i>:CORRection:FLATness:STARTfreq?</i> 返回值: <i>200000000\n</i>

## 3.4.4.15 步进填充结束频率 ([:SOURce]:CORRection:FLATness:STOPfreq)

命令格式	[:SOURce]:CORRection:FLATness:STOPfreq <freq> [:SOURce]:CORRection:FLATness:STOPfreq?
------	--

<b>功能描述</b>	当需要用功率计填充平整度列表且填充类型为“步进填充”时，设置步进填充的结束频率。 查询步进填充的结束频率。
<b>参数类型</b>	浮点型， 单位：Hz，kHz，MHz 或 GHz，默认为 Hz
<b>参数范围</b>	全频率范围
<b>返回值</b>	浮点型，单位：Hz
<b>默认值</b>	最大频率值
<b>对应菜单</b>	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式 > 步进填充 > 结束频率
<b>示例</b>	<code>:CORRection:FLATness:STOPfreq 500 MHz</code> <code>:CORRection:FLATness:STOPfreq?</code> 返回值： <code>500000000\n</code>

#### 3.4.4.16 步进填充频率步进方式 ([:SOURce]:CORRection:FLATness:SPACE)

<b>命令格式</b>	<code>[:SOURce]:CORRection:FLATness:SPACE LINear LOGarithmic</code> <code>[:SOURce]:CORRection:FLATness:SPACE?</code>
<b>功能描述</b>	当需要用功率计填充平整度列表且填充类型为“步进填充”时，设置步进填充的频率步进方式。 查询步进填充的频率步进方式。
<b>参数类型</b>	枚举
<b>参数范围</b>	LINear LOGarithmic
<b>返回值</b>	枚举
<b>默认值</b>	LINear
<b>对应菜单</b>	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式 > 步进填充 > 频率步进
<b>示例</b>	<code>:CORRection:FLATness:SPACE LINear</code> <code>:CORRection:FLATness:SPACE?</code> 返回值： <code>LINear\n</code>

#### 3.4.4.17 步进填充线性频率步进 ([:SOURce]:CORRection:FLATness:LINStep)

<b>命令格式</b>	<code>[:SOURce]:CORRection:FLATness:LINStep &lt;freq&gt;</code> <code>[:SOURce]:CORRection:FLATness:LINStep?</code>
<b>功能描述</b>	设置步进填充的线性频率步进。 查询步进填充的线性频率步进。

参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
返回值	浮点型, 单位: Hz
默认值	0
对应菜单	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式 > 步进填充 > 线性步进
示例	<code>:CORRection:FLATness:LINStep 200 MHz</code> <code>:CORRection:FLATness:LINStep?</code> 返回值: <code>200000000\n</code>

#### 3.4.4.18 步进填充对数频率步进 ([:SOURce]:CORRection:FLATness:LOGStep)

命令格式	<code>[:SOURce]:CORRection:FLATness:LOGStep &lt;value&gt;</code> <code>[:SOURce]:CORRection:FLATness:LOGStep?</code>
功能描述	设置步进填充的对数频率步进。 查询步进填充的对数频率步进。
参数类型	浮点型, 单位: %
返回值	浮点型, 单位: %
默认值	0
对应菜单	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式 > 步进填充 > 对数步进
示例	<code>:CORRection:FLATness:LOGStep 20</code> <code>:CORRection:FLATness:LOGStep?</code> 返回值: <code>20\n</code>

#### 3.4.4.19 步进填充点数 ([:SOURce]:CORRection:FLATness:POINT)

命令格式	<code>[:SOURce]:CORRection:FLATness:POINT &lt;points&gt;</code> <code>[:SOURce]:CORRection:FLATness:POINT?</code>
功能描述	设置步进填充的扫描点数。 查询步进填充的扫描点数。
参数类型	整型
参数范围	2 ~ 500
返回值	整型
默认值	2
对应菜单	<b>LEVEL</b> > 平坦度 > 设置 > 填充方式 > 步进填充 > 点数
示例	<code>:CORRection:FLATness:POINT 5</code> <code>:CORRection:FLATness:POINT?</code>

返回值:

5n

#### 3.4.4.20 用功率计填充平坦度 ([:SOURce]:CORRection:CSET:DATA[:SENSor][:POWER]:SONCe)

命令格式	[:SOURce]:CORRection:CSET:DATA[:SENSor][:POWER]:SONCe
功能描述	用功率计填充平坦度列表的幅度修正值。
对应菜单	<b>LEVEL</b> > 平坦度 > 设置 > 幅度校准
示例	:CORRection:CSET:DATA:SONCe

### 3.4.5 扫描设置

#### 3.4.5.1 扫描状态 ([:SOURce]:SWEep:STATe)

命令格式	[:SOURce]:SWEep:STATe OFF FREQuency LEVe LEV_FREQ [:SOURce]:SWEep:STATe?
功能描述	设置扫描状态。 查询扫描状态。
参数类型	枚举
参数范围	OFF FREQuency LEVe LEV_FREQ
返回值	枚举
默认值	OFF
对应菜单	<b>SWEEP</b> > 扫描状态
示例	:SWEep:STATe LEV_FREQ :SWEep:STATe? 返回值: LEV_FREQn

#### 3.4.5.2 扫描类型 ([:SOURce]:SWEep:TYPE)

命令格式	[:SOURce]:SWEep:TYPE LIST STEP [:SOURce]:SWEep:TYPE?
功能描述	设置扫描类型为步进扫描或列表扫描。 查询扫描类型。
参数类型	枚举
参数范围	LIST STEP
返回值	枚举

默认值	STEP
对应菜单	<b>SWEEP</b> > 步进扫描 或 列表扫描
示例	:SWEep:TYPE STEP :SWEep:TYPE? 返回值: STEP\n

#### 3.4.5.3 步进扫描开始频率 ([:SOURce]:SWEep:STEP:START:FREQuency)

命令格式	[:SOURce]:SWEep:STEP:START:FREQuency <freq> [:SOURce]:SWEep:STEP:START:FREQuency?
功能描述	设置步进扫描的起始频率。 查询步进扫描的起始频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	全频率范围
返回值	浮点型, 单位: Hz
默认值	最大频率值
对应菜单	<b>SWEEP</b> > 步进扫描 > 开始频率
示例	:SWEep:STEP:START:FREQuency 1 GHz :SWEep:STEP:START:FREQuency? 返回值: 1000000000\n

#### 3.4.5.4 步进扫描结束频率 ([:SOURce]:SWEep:STEP:STOP:FREQuency)

命令格式	[:SOURce]:SWEep:STEP:STOP:FREQuency <freq> [:SOURce]:SWEep:STEP:STOP:FREQuency?
功能描述	设置步进扫描的结束频率。 查询步进扫描的结束频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	全频率范围
返回值	浮点型, 单位: Hz
默认值	最大频率值
对应菜单	<b>SWEEP</b> > 步进扫描 > 结束频率
示例	:SWEep:STEP:STOP:FREQuency 1 GHz

```
:SWEep:STEP:STOP:FREQuency?
```

返回值:

```
1000000000\n
```

### 3.4.5.5 步进扫描开始幅度 ([:SOURce]:SWEep:STEP:START:LEVel)

命令格式	[:SOURce]:SWEep:STEP:START:LEVel <level> [:SOURce]:SWEep:STEP:START:LEVel?
功能描述	设置步进扫描的起始幅度。 查询步进扫描的起始幅度。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	全幅度范围
返回值	浮点型, 单位: dBm
默认值	-130 dBm
对应菜单	<b>SWEEP</b> > 步进扫描 > 开始幅度
示例	:SWEep:STEP:START:LEVel 0 dBm :SWEep:STEP:START:LEVel? 返回值: 0\n

### 3.4.5.6 步进扫描结束幅度 ([:SOURce]:SWEep:STEP:STOP:LEVel)

命令格式	[:SOURce]:SWEep:STEP:STOP:LEVel <level> [:SOURce]:SWEep:STEP:STOP:LEVel?
功能描述	设置步进扫描的结束幅度。 查询步进扫描的结束幅度。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	全幅度范围
返回值	浮点型, 单位: dBm
默认值	-130 dBm
对应菜单	<b>SWEEP</b> > 步进扫描 > 结束幅度
示例	:SWEep:STEP:STOP:LEVel 0 dBm :SWEep:STEP:STOP:LEVel? 返回值: 0\n

## 3.4.5.7 步进扫描驻留时间 ([:SOURce]:SWEep:STEP:DWELI)

命令格式	[:SOURce]:SWEep:STEP:DWELI <time> [:SOURce]:SWEep:STEP:DWELI?
功能描述	设置步进扫描的驻留时间。 查询步进扫描的驻留时间。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	10 ms ~ 100 s
返回值	浮点型, 单位: s
默认值	30 ms
对应菜单	<b>SWEEP</b> > 步进扫描 > 驻留时间
示例	:SWEep:STEP:DWELI 20 ms :SWEep:STEP:DWELI? 返回值: 0.02\n

## 3.4.5.8 步进扫描点数 ([:SOURce]:SWEep:STEP:POINts)

命令格式	[:SOURce]:SWEep:STEP:POINts <points> [:SOURce]:SWEep:STEP:POINts?
功能描述	设置步进扫描的扫描点数。 查询步进扫描的扫描点数。
参数类型	整型
参数范围	2 ~ 65535
返回值	整型
默认值	11
对应菜单	<b>SWEEP</b> > 步进扫描 > 扫描点数
示例	:SWEep:STEP:POINts 2 :SWEep:STEP:POINts? 返回值: 2\n

## 3.4.5.9 步进扫描形状 ([:SOURce]:SWEep:STEP:SHAPE)

命令格式	[:SOURce]:SWEep:STEP:SHAPE TR angle SAWtooth [:SOURce]:SWEep:STEP:SHAPE?
功能描述	设置步进扫描的扫描形状。

	查询步进扫描的扫描形状。
参数类型	枚举
参数范围	TRlangle SAWtooth
返回值	枚举
默认值	SAWtooth
对应菜单	<b>[SWEEP]</b> > 步进扫描 > 扫描形状
示例	<code>:SWEep:STEP:SHApe TRlangle</code> <code>:SWEep:STEP:SHApe?</code> 返回值: <code>TRlangle\n</code>

#### 3.4.5.10 步进扫描频率步进方式 ([:SOURce]:SWEep:STEP:SPACe)

命令格式	<code>[:SOURce]:SWEep:STEP:SPACe LINear LOGarithmic</code> <code>[:SOURce]:SWEep:STEP:SPACe?</code>
功能描述	设置步进扫描的频率步进方式。 查询步进扫描的频率步进方式。
参数类型	枚举
参数范围	LINear LOGarithmic
返回值	枚举
默认值	LINear
对应菜单	<b>[SWEEP]</b> > 步进扫描 > 扫描步进
示例	<code>:SWEep:STEP:SPACe LOGarithmic</code> <code>:SWEep:STEP:SPACe?</code> 返回值: <code>LOGarithmic\n</code>

#### 3.4.5.11 步进扫描线性频率步进 ([:SOURce]:SWEep[:FREQuency]:STEP[:LINear])

命令格式	<code>[:SOURce]:SWEep[:FREQuency]:STEP[:LINear] &lt;freq&gt;</code> <code>[:SOURce]:SWEep[:FREQuency]:STEP[:LINear]?</code>
功能描述	设置步进扫描的线性频率步进。 查询步进扫描的线性频率步进。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
返回值	浮点型, 单位: Hz
默认值	0

对应菜单 **SWEEP** > 步进扫描 > 频率线性步进

示例 `:SWEep:STEP 200 MHz`  
`:SWEep:STEP?`

返回值:

`200000000\n`

### 3.4.5.12 步进扫描对数频率步进 ([:SOURce]:SWEep[:FREQUency]:STEP:LOGarithmic)

命令格式 `[:SOURce]:SWEep[:FREQUency]:STEP:LOGarithmic <value>`  
`[:SOURce]:SWEep[:FREQUency]:STEP:LOGarithmic?`

功能描述 设置步进扫描的对数频率步进。  
查询步进扫描的对数频率步进。

参数类型 浮点型, 单位: %

返回值 浮点型, 单位: %

默认值 0

对应菜单 **SWEEP** > 步进扫描 > 频率对数步进

示例 `:SWEep:STEP:LOGarithmic 20`  
`:SWEep:STEP:LOGarithmic?`

返回值:

`20\n`

### 3.4.5.13 扫描列表添加行 ([:SOURce]:SWEep:LIST:ADDList)

命令格式 `[:SOURce]:SWEep:LIST:ADDList <freq>,<level>,<time>`

功能描述 在扫描列表中添加一行。

参数类型 频率: 浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz,  
幅度: 浮点型, 单位: dB,  
时间: 浮点型, 单位: ns, us, ms 或 s, 默认为 s

参数范围 频率: 全频率范围,  
幅度: 全幅度范围,  
时间: 10.0 ms ~ 100.0 s

对应菜单 **SWEEP** > 列表扫描 > 添加

示例 `:SWEep:LIST:ADDList 1 GHz,0 dBm,1 s`

### 3.4.5.14 扫描列表删除行 ([:SOURce]:SWEep:LIST:DELeTe)

命令格式 `[:SOURce]:SWEep:LIST:DELeTe <row>`

功能描述	从扫描列表中删除指定的行。
参数类型	整型
参数范围	1 ~ 扫描列表的总行数
对应菜单	<b>SWEEP</b> > 列表扫描 > 删除
示例	<code>:SWEep:LIST:DELeTe 1</code>

#### 3.4.5.15 编辑扫描列表 ([:SOURce]:SWEep:LIST:CHANGe)

命令格式	<code>[:SOURce]:SWEep:LIST:CHANGe &lt;row&gt;,&lt;freq&gt;,&lt;power&gt;,&lt;time&gt;</code>
功能描述	编辑扫描列表中的指定行。
参数类型	行号: 整型, 频率: 浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz, 幅度: 浮点型, 单位: dB, 时间: 浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	行号: 1 ~ 扫描列表的总行数, 频率: 全频率范围, 幅度: 全幅度范围, 时间: 10.0 ms ~ 100.0 s
对应菜单	<b>SWEEP</b> > 列表扫描
示例	<code>:SWEep:LIST:CHANGe 1,1 GHz,1 dBm, 1 s</code>

#### 3.4.5.16 查询扫描列表行数 ([:SOURce]:SWEep:LIST:CPOint?)

命令格式	<code>[:SOURce]:SWEep:LIST:CPOint?</code>
功能描述	查询扫描列表的行数。
返回值	整型
默认值	1
对应菜单	<b>SWEEP</b> > 列表扫描
示例	<code>:SWEep:LIST:CPOint?</code> 返回值: <code>5\n</code>

#### 3.4.5.17 查询扫描列表数据 ([:SOURce]:SWEep:LIST:LIST?)

命令格式	<code>[:SOURce]:SWEep:LIST:LIST? &lt;begin_row&gt;,&lt;end_row&gt;</code>
------	---

功能描述	查询扫描列表中 begin_row 到 end_row 的数据。
参数类型	整型, 整型
参数范围	1 ~ 扫描列表的总行数, 开始行 ~ 扫描列表的总行数
返回值	字符串
对应菜单	<b>SWEEP</b> > 列表扫描
示例	: <i>SWEEp:LIST:LIST? 1,3</i> 返回值: <i>1000000000,0,0.03\s2000000000,-2.4,0.03\s3000000000,-4.8,0.03\n</i>

#### 3.4.5.18 清空扫描列表 ([:SOURce]:SWEep:LIST:INITialize:PRESet)

命令格式	[:SOURce]:SWEep:LIST:INITialize:PRESet
功能描述	清空扫描列表。
对应菜单	<b>SWEEP</b> > 列表扫描 > 清空
示例	: <i>SWEEp:LIST:INITialize:PRESet</i>

#### 3.4.5.19 从步进扫描导入扫描列表 ([:SOURce]:SWEep:LIST:INITialize:FSTep)

命令格式	[:SOURce]:SWEep:LIST:INITialize:FSTep
功能描述	从步进扫描导入扫描列表。
对应菜单	<b>SWEEP</b> > 列表扫描 > 导入
示例	: <i>SWEEp:LIST:INITialize:FSTep</i>

#### 3.4.5.20 加载扫描列表 ([:SOURce]:SWEep:LOAD)

命令格式	[:SOURce]:SWEep:LOAD <"file_name">
功能描述	从 LSW 文件中加载扫描列表。
参数类型	字符串
参数范围	无
对应菜单	<b>SWEEP</b> > 列表扫描 > 加载
示例	: <i>SWEEp:LOAD "U-disk3/test.lsw"</i>

## 3.4.5.21 保存扫描列表 ([:SOURce]:SWEep:STORe)

命令格式	[:SOURce]:SWEep:STORe <"file_name">
功能描述	将扫描列表保存到 LSW 文件。
参数类型	字符串
参数范围	无
对应菜单	<b>[SWEEP]</b> > 列表扫描 > 保存
示例	<i>:SWEep:STORe "U-disk3/test.lsw"</i>

## 3.4.5.22 扫描方向 ([:SOURce]:SWEep:DIRect)

命令格式	[:SOURce]:SWEep:DIRect FWD REV [:SOURce]:SWEep:DIRect?
功能描述	设置扫描方向。 查询扫描方向。
参数类型	枚举
参数范围	FWD REV
返回值	枚举
默认值	FWD
对应菜单	<b>[SWEEP]</b> > 扫描方向
示例	<i>:SWEep:DIRect REV</i> <i>:SWEep:DIRect?</i> 返回值: <i>REVn</i>

## 3.4.5.23 扫描模式 ([:SOURce]:SWEep:MODE)

命令格式	[:SOURce]:SWEep:MODE CONTInue SINGle [:SOURce]:SWEep:MODE?
功能描述	设置扫描模式为连续或单次。 查询扫描模式。
参数类型	枚举
参数范围	CONTInue SINGle
返回值	枚举
默认值	CONTInue
对应菜单	<b>[SWEEP]</b> > 扫描模式

示例	<code>:SWEep:MODE SINGLE</code> <code>:SWEep:MODE?</code> 返回值: <code>SINGLE</code>
----	---

#### 3.4.5.24 执行单次扫描 ([:SOURce]:SWEep:EXECute)

命令格式	<code>[:SOURce]:SWEep:EXECute</code>
功能描述	当扫描模式是单次时，执行单次扫描。
对应菜单	<b>SWEEP</b> > 执行单次扫描
示例	<code>:SWEep:EXECute</code>

#### 3.4.5.25 扫描触发方式 ([:SOURce]:SWEep:SWEEp:TRIGger:TYPE)

命令格式	<code>[:SOURce]:SWEep:SWEEp:TRIGger:TYPE AUTO KEY BUS EXT</code> <code>[:SOURce]:SWEep:SWEEp:TRIGger:TYPE?</code>
功能描述	设置扫描触发方式。 查询扫描触发方式。
参数类型	枚举
参数范围	AUTO KEY BUS EXT
返回值	枚举
默认值	AUTO
对应菜单	<b>SWEEP</b> > 触发方式
示例	<code>:SWEep:SWEEp:TRIGger:TYPE KEY</code> <code>:SWEep:SWEEp:TRIGger:TYPE?</code> 返回值: <code>KEY</code>

#### 3.4.5.26 扫描点触发方式 ([:SOURce]:SWEep:POINt:TRIGger:TYPE)

命令格式	<code>[:SOURce]:SWEep:POINt:TRIGger:TYPE AUTO KEY BUS EXT</code> <code>[:SOURce]:SWEep:POINt:TRIGger:TYPE?</code>
功能描述	设置扫描的点触发方式。 查询扫描的点触发方式。
参数类型	枚举
参数范围	AUTO KEY BUS EXT
返回值	枚举
默认值	AUTO

对应菜单	<b>SWEEP</b> > 点触发方式
示例	<code>:SWEEp:POINt:TRIGger:TYPE KEY</code> <code>:SWEEp:POINt:TRIGger:TYPE?</code> 返回值: <code>KEYn</code>

#### 3.4.5.27 扫描触发沿 ([:SOURce]:INPut:TRIGger:SLOPe)

命令格式	<code>[:SOURce]:INPut:TRIGger:SLOPe POSitive NEGative</code> <code>[:SOURce]:INPut:TRIGger:SLOPe?</code>
功能描述	设置扫描功能的外部触发信号的触发沿。 查询扫描功能的外部触发信号的触发沿。
参数类型	枚举
参数范围	POSitive NEGative
返回值	枚举
默认值	POSitive
对应菜单	<b>SWEEP</b> > 触发沿
示例	<code>:INPut:TRIGger:SLOPe NEGative</code> <code>:INPut:TRIGger:SLOPe?</code> 返回值: <code>NEGativen</code>

#### 3.4.5.28 查询当前的扫描点 ([:SOURce]:SWEep:CURRent:DATA?)

命令格式	<code>[:SOURce]:SWEep:CURRent:DATA?</code>
功能描述	查询当前的扫描点。
返回值	字符串 格式为: <code>index,{freq,level,time}</code> Index: 整型, 扫描点索引, Freq: 浮点型, 扫描点频率, 单位: Hz, Level: 浮点型, 扫描点幅度, 单位: dBm, Time: 浮点型, 扫描点驻留时间, 单位: s。
默认值	无
对应菜单	屏幕上方的显示频率和显示幅度
示例	<code>:SWEep:CURRent:DATA?</code> 返回值: <code>1,{1000000000,0,0.03}n</code>

## 3.4.5.29 查询当前扫描点的频率 ([:SOURce]:SWEep:CURRent:FREQuency?)

命令格式	[:SOURce]:SWEep:CURRent:FREQuency?
功能描述	查询当前扫描点的频率。
返回值	浮点型, 单位: Hz
默认值	无
对应菜单	屏幕上方的显示频率
示例	:SWEep:CURRent:FREQuency? 返回值: 1000000000.000000n

## 3.4.5.30 查询当前扫描点的幅度 ([:SOURce]:SWEep:CURRent:LEVel?)

命令格式	[:SOURce]:SWEep:CURRent:LEVel?
功能描述	查询当前扫描点的幅度。
返回值	浮点型, 单位: dBm
默认值	无
对应菜单	屏幕上方的显示幅度
示例	:SWEep:CURRent:LEVel? 返回值: 0.000000n

## 3.4.6 功率计设置

## 3.4.6.1 功率计功率控制状态 ([:SOURce]:POWER:SPC:STATe)

命令格式	[:SOURce]:POWER:SPC:STATe ON OFF 1 0 [:SOURce]:POWER:SPC:STATe?
功能描述	设置功率计测量的功率控制状态。 查询功率计测量的功率控制状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
等效命令	:SENSe[:POWER]:LEV:CTL:STATe ON OFF 1 0 :SENSe[:POWER]:LEV:CTL:STATe?
对应菜单	<span style="border: 1px solid black; padding: 2px;">HOME</span> > POWER SENSOR > 功率控制

示例	<code>:POWER:SPC:STATe ON</code> <code>:POWER:SPC:STATe?</code>
返回值:	<code>1\n</code>

### 3.4.6.2 功率控制目标功率 ([:SOURce]:POWER:SPC:TARGet)

命令格式	<code>[:SOURce]:POWER:SPC:TARGet &lt;power&gt;</code> <code>[:SOURce]:POWER:SPC:TARGet?</code>
功能描述	设置功率计功率控制的目标功率。 查询功率计功率控制的目标功率。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	-120 dBm ~ 20 dBm
返回值	浮点型, 单位: dBm
默认值	0 dBm
等效命令	<code>:SENSe[:POWER]:SPC:TARGet &lt;power&gt;</code> <code>:SENSe[:POWER]:SPC:TARGet?</code>
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制 > 目标功率
示例	<code>:POWER:SPC:TARGet 0</code> <code>:POWER:SPC:TARGet?</code>
返回值:	<code>0\n</code>

### 3.4.6.3 功率控制限制功率 ([:SOURce]:POWER:LIMit)

命令格式	<code>[:SOURce]:POWER:LIMit &lt;power&gt;</code> <code>[:SOURce]:POWER:LIMit?</code>
功能描述	设置功率计功率控制的限制功率。 查询功率计功率控制的限制功率。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	-120 dBm ~ 20 dBm
返回值	浮点型, 单位: dBm
默认值	0 dBm
等效命令	<code>:SENSe[:POWER]:LIMit &lt;power&gt;</code> <code>:SENSe[:POWER]:LIMit?</code>
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制 > 限制功率

示例	<code>POWer:LiMit 1</code> <code>POWer:LiMit?</code> 返回值: <code>1\n</code>
----	---

#### 3.4.6.4 功率控制捕获范围 ([:SOURce]:POWer:SPC:CRANge)

命令格式	<code>[SOURce]:POWer:SPC:CRANge &lt;power&gt;</code> <code>[SOURce]:POWer:SPC:CRANge?</code>
功能描述	设置功率计功率控制的捕获范围。 查询功率计功率控制的捕获范围。
参数类型	浮点型, 单位: dB
参数范围	0 dB ~ 50 dB
返回值	浮点型, 单位: dB
默认值	0 dB
等效命令	<code>:SENSe[:POWer]:SPC:CRANge &lt;power&gt;</code> <code>:SENSe[:POWer]:SPC:CRANge?</code>
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制 > 捕获范围
示例	<code>:POWer:SPC:CRANge 5</code> <code>:POWer:SPC:CRANge?</code> 返回值: <code>5\n</code>

### 3.4.7 模拟调制设置

#### 3.4.7.1 模拟调制开关状态 ([:SOURce]:MODulation)

命令格式	<code>[[:SOURce]:MODulation ON OFF 1 0</code> <code>[[:SOURce]:MODulation?</code>
功能描述	设置模拟调制的开关状态。 查询模拟调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
等效命令	<code>:OUTPut:MODulation[:STATe] ON OFF 1 0</code> <code>:OUTPut:MODulation[:STATe]?</code>
对应菜单	ANALOG MOD > On

示例 `:MODulation ON`  
`:MODulation?`  
 返回值:  
`1\n`

### 3.4.7.2 幅度调制设置

#### 3.4.7.2.1 幅度调制开关状态 ([:SOURce]:AM:STATe)

命令格式	<code>[:SOURce]:AM:STATe ON OFF 1 0</code> <code>[:SOURce]:AM:STATe?</code>
功能描述	设置幅度调制的调制开关状态。 查询幅度调制的调制开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>MOD</b> > 调幅 > AM 状态
示例	<code>:AM:STATe ON</code> <code>:AM:STATe?</code> 返回值: <code>1\n</code>

#### 3.4.7.2.2 幅度调制波形 ([:SOURce]:AM:WAVEform)

命令格式	<code>[:SOURce]:AM:WAVEform SINE SQUAre</code> <code>[:SOURce]:AM:WAVEform?</code>
功能描述	设置幅度调制的调制波形。 查询幅度调制的调制波形。
参数类型	枚举
参数范围	SINE SQUAre
返回值	枚举
默认值	SINE
对应菜单	<b>MOD</b> > 调幅 > AM 波形
示例	<code>:AM:WAVEform SQUAre</code> <code>:AM:WAVEform?</code> 返回值: <code>SQUAre\n</code>

## 3.4.7.2.3 幅度调制源 ([:SOURce]:AM:SOURce)

命令格式	<code>[:SOURce]:AM:SOURce INTernal EXTernal INT,EXT</code> <code>[:SOURce]:AM:SOURce?</code>
功能描述	设置幅度调制的调制源。 查询幅度调制的调制源。
参数类型	枚举
参数范围	INTernal EXTernal INT,EXT
返回值	枚举
默认值	INTernal
对应菜单	<b>MOD</b> > 调幅 > AM 源
示例	<code>:AM:SOURce EXTernal</code> <code>:AM:SOURce?</code> 返回值: <code>EXTernal</code>

## 3.4.7.2.4 幅度调制深度 ([:SOURce]:AM:DEPTH)

命令格式	<code>[:SOURce]:AM:DEPTH &lt;value&gt;</code> <code>[:SOURce]:AM:DEPTH?</code>
功能描述	设置幅度调制的调制深度。 查询幅度调制的调制深度。
参数类型	浮点型
参数范围	0 ~ 1
返回值	浮点型
默认值	0.5
对应菜单	<b>MOD</b> > 调幅 > 调制深度
示例	<code>:AM:DEPTH 0.2</code> <code>:AM:DEPTH?</code> 返回值: <code>0.2</code>

## 3.4.7.2.5 幅度调制频率 ([:SOURce]:AM:FREQUENCY)

命令格式	<code>[:SOURce]:AM:FREQUENCY &lt;value&gt;</code> <code>[:SOURce]:AM:FREQUENCY?</code>
功能描述	设置幅度调制的调制频率。 查询幅度调制的调制频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz

参数范围	调制波为正弦波时：0.01 Hz ~ 100 kHz， 调制波为方波时：0.01 Hz ~ 20 kHz。
返回值	浮点型，单位：Hz
默认值	1 kHz
对应菜单	<b>MOD</b> > 调幅 > AM 调制频率
示例	<i>:AM:FREQuency 10 kHz</i> <i>:AM:FREQuency?</i> 返回值： <i>10000\n</i>

#### 3.4.7.2.6 幅度调制外调制灵敏度 ([:SOURce]:AM:SENSitivity?)

命令格式	<i>[:SOURce]:AM:SENSitivity?</i>
功能描述	查询幅度调制的外调制灵敏度。
返回值	浮点型，单位：V
默认值	0.125V
对应菜单	<b>MOD</b> > 调幅 > 外调制灵敏度
示例	<i>AM:SENSitivity?</i> 返回值： <i>0.125\n</i>

#### 3.4.7.3 频率调制设置

##### 3.4.7.3.1 频率调制开关状态 ([:SOURce]:FM:STATe)

命令格式	<i>[:SOURce]:FM:STATe ON OFF 1 0</i> <i>[:SOURce]:FM:STATe?</i>
功能描述	设置频率调制的调制开关状态。 查询频率调制的调制开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>MOD</b> > 调频 > FM 状态
示例	<i>:FM:STATe ON</i> <i>:FM:STATe?</i> 返回值： <i>1\n</i>

## 3.4.7.3.2 频率调制源 ([:SOURce]:FM:SOURce)

命令格式	[:SOURce]:FM:SOURce INT1 INT2 INT1,INT2 EXternal INT1,EXT DUAL [:SOURce]:FM:SOURce?
功能描述	设置频率调制的调制源。 查询频率调制的调制源。
参数类型	枚举
参数范围	INT1 INT2 INT1,INT2 EXternal INT1,EXT DUAL
返回值	枚举
默认值	INT1
对应菜单	<b>MOD</b> > 调频 > FM 源
示例	:FM:SOURce EXT :FM:SOURce? 返回值: EXternal\n

## 3.4.7.3.3 频率调制波形 1 ([:SOURce]:FM1:WAVEform)

命令格式	[:SOURce]:FM1:WAVEform SINE SQUAre SAWTooth TRIangle [:SOURce]:FM1:WAVEform?
功能描述	设置频率调制的内部源 1 的调制波形。 查询频率调制的内部源 1 的调制波形。
参数类型	枚举
参数范围	SINE SQUAre SAWTooth TRIangle
返回值	枚举
默认值	SINE
对应菜单	<b>MOD</b> > 调频 > FM 波形 1
示例	:FM1:WAVEform SQUAre :FM1:WAVEform? 返回值: SQUAre\n

## 3.4.7.3.4 频率调制最大频偏 1 ([:SOURce]:FM1:DEViation)

命令格式	[:SOURce]:FM1:DEViation <value> [:SOURce]:FM1:DEViation?
功能描述	设置频率调制的内部源 1 的最大频偏。 查询频率调制的内部源 1 的最大频偏。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz

参数范围	1 Hz ~ N*1 MHz, N 为频率分段系数, 请查阅数据手册
返回值	浮点型, 单位: Hz
默认值	100 kHz
对应菜单	<b>MOD</b> > 调频 > 最大频偏 1
示例	:FM1:DEVIation 500 kHz :FM1:DEVIation? 返回值: 500000\n

#### 3.4.7.3.5 调制频率 1 ([:SOURce]:FM1:FREQuency)

命令格式	[:SOURce]:FM1:FREQuency <value> [:SOURce]:FM1:FREQuency?
功能描述	设置频率调制的内部源 1 的调制频率。 查询频率调制的内部源 1 的调制频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	调制波为正弦波时: 0.01 Hz ~ 100 kHz 调制波为方波/锯齿波/三角波时: 0.01 Hz ~ 20 kHz
返回值	浮点型, 单位: Hz
默认值	10 kHz
对应菜单	<b>MOD</b> > 调频 > FM 调制频率 1
示例	:FM1:FREQuency 40 kHz :FM1:FREQuency? 返回值: 40000\n

#### 3.4.7.3.6 频率调制初相 1 ([:SOURce]:FM1:PHASe)

命令格式	[:SOURce]:FM1:PHASe <value> [:SOURce]:FM1:PHASe?
功能描述	当频率调制源为内部源 1+内部源 2 或双波形时, 设置/查询内部源 1 的初相。
参数类型	浮点型, 单位: degree (°)
参数范围	-360 ~ +360
返回值	浮点型, 单位: degree (°)
默认值	0
对应菜单	<b>MOD</b> > 调频 > FM 初相 1

示例 `:FM1:PHASe -30`  
`:FM1:PHASe?`  
 返回值:  
`-30\n`

#### 3.4.7.3.7 频率调制波形 2 ([:SOURce]:FM2:WAVEform)

命令格式	<code>[:SOURce]:FM2:WAVEform SINE SQUAre SAWTooth TRlangle</code> <code>[:SOURce]:FM2:WAVEform?</code>
功能描述	设置频率调制的内部源 2 的调制波形。 查询频率调制的内部源 2 的调制波形。
参数类型	枚举
参数范围	SINE SQUAre SAWTooth TRlangle
返回值	枚举
默认值	SINE
对应菜单	<b>MOD</b> > 调频 > FM 波形 2
示例	<code>:FM2:WAVEform SQUAre</code> <code>:FM2:WAVEform?</code> 返回值: <code>SQUAre\n</code>

#### 3.4.7.3.8 频率调制最大频偏 2 ([:SOURce]:FM2:DEVIation)

命令格式	<code>[:SOURce]:FM2:DEVIation &lt;value&gt;</code> <code>[:SOURce]:FM2:DEVIation?</code>
功能描述	设置频率调制的内部源 2 的最大频偏。 查询频率调制的内部源 2 的最大频偏。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	1 Hz ~ N*1 MHz N 为频率分段系数, 请查阅数据手册
返回值	浮点型, 单位: Hz
默认值	100 kHz
对应菜单	<b>MOD</b> > 调频 > 最大频偏 2
示例	<code>:FM2:DEVIation 500 kHz</code> <code>:FM2:DEVIation?</code> 返回值: <code>500000\n</code>

## 3.4.7.3.9 调制频率 2 ([:SOURce]:FM2:FREQuency)

命令格式	[:SOURce]:FM2:FREQuency <value> [:SOURce]:FM2:FREQuency?
功能描述	设置频率调制的内部源 2 的调制频率。 查询频率调制的内部源 2 的调制频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	调制波为正弦波时: 0.01 Hz ~ 100 kHz 调制波为方波/锯齿波/三角波时: 0.01 Hz ~ 20 kHz
返回值	浮点型, 单位: Hz
默认值	10 kHz
对应菜单	<b>MOD</b> > 调频 > FM 调制频率 2
示例	:FM2:FREQuency 40 kHz :FM2:FREQuency? 返回值: 40000\n

## 3.4.7.3.10 频率调制初相 2 ([:SOURce]:FM2:PHASe)

命令格式	[:SOURce]:FM2:PHASe <value> [:SOURce]:FM2:PHASe?
功能描述	当频率调制源为内部源 1+内部源 2 或双波形时, 设置/查询内部源 2 的初相。
参数类型	浮点型, 单位: deg 或 rad, 默认为 deg
参数范围	-360° ~ +360°
返回值	浮点型, 单位: deg
默认值	0
对应菜单	<b>MOD</b> > 调频 > FM 初相 2
示例	:FM2:PHASe -30 deg :FM2:PHASe? 返回值: -30\n

## 3.4.7.3.11 内部源 1 幅度占比 ([:SOURce]:FM1:PROPortion)

命令格式	[:SOURce]:FM1:PROPortion <value> [:SOURce]:FM1:PROPortion?
功能描述	当频率调制源为双波形时, 设置/查询内部源 1 的幅度占比。
参数类型	浮点型
参数范围	0 ~ 1

返回值	浮点型
默认值	0.5
对应菜单	<b>MOD</b> > 调频 > 内部源 1 幅度占比
示例	<code>:FM1:PROPortion 0.6</code> <code>:FM1:PROPortion?</code> 返回值: <code>0.6\n</code>

#### 3.4.7.3.12 外调制灵敏度 ([:SOURce]:FM1:SENSitivity?)

命令格式	<code>[:SOURce]:FM1:SENSitivity?</code>
功能描述	当频率调制源包含外部源时，查询外部调制源的灵敏度。
返回值	浮点型，单位：Hz/V
默认值	无
对应菜单	<b>MOD</b> > 调频 > 外调制灵敏度
示例	<code>FM1:SENSitivity?</code> 返回值: <code>125000\n</code>

### 3.4.7.4 相位调制设置

#### 3.4.7.4.1 相位调制开关状态 ([:SOURce]:PM:STATe)

命令格式	<code>[:SOURce]:PM:STATe ON OFF 1 0</code> <code>[:SOURce]:PM:STATe?</code>
功能描述	设置相位调制的调制开关状态。 查询相位调制的调制开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>MOD</b> > 调相 > PM 状态
示例	<code>:PM:STATe ON</code> <code>:PM:STATe?</code> 返回值: <code>1\n</code>

#### 3.4.7.4.2 相位调制波形 ([:SOURce]:PM:WAVEform)

命令格式	<code>[:SOURce]:PM:WAVEform SINE SQUAre</code> <code>[:SOURce]:PM:WAVEform?</code>
------	---

功能描述	设置相位调制的调制波形。 查询相位调制的调制波形。
参数类型	枚举
参数范围	SINE SQUAre
返回值	枚举
默认值	SINE
对应菜单	<b>MOD</b> > 调相 > PM 波形
示例	:PM:WAVEform SQUAre :PM:WAVEform? 返回值: SQUAre\n

#### 3.4.7.4.3 相位调制源 ([:SOURce]:PM:SOURce)

命令格式	[:SOURce]:PM:SOURce INTernal EXTernal INT,EXT [:SOURce]:PM:SOURce?
功能描述	设置相位调制的调制源。 查询相位调制的调制源。
参数类型	枚举
参数范围	INTernal EXTernal INT,EXT
返回值	枚举
默认值	INTernal
对应菜单	<b>MOD</b> > 调相 > PM 源
示例	:PM:SOURce EXTernal :PM:SOURce? 返回值: EXTernal\n

#### 3.4.7.4.4 最大相位偏移 ([:SOURce]:PM:DEViation)

命令格式	[:SOURce]:PM:DEViation <value> [:SOURce]:PM:DEViation?
功能描述	设置相位调制的最大相位偏移。 查询相位调制的最大相位偏移。
参数类型	浮点型, 单位: rad
参数范围	0.01 rad ~ N*5 rad, N 为频率分段系数, 请查阅数据手册
返回值	浮点型, 单位: rad
默认值	1

对应菜单 **MOD** > 调相 > 最大相位偏移

示例 `:PM:DEVIation 2`  
`:PM:DEVIation?`

返回值:  
`2\n`

#### 3.4.7.4.5 相位调制频率 ([:SOURce]:PM:FREQuency)

命令格式 `[:SOURce]:PM:FREQuency <value>`  
`[:SOURce]:PM:FREQuency?`

功能描述 设置相位调制的调制频率。  
查询相位调制的调制频率。

参数类型 浮点型,  
单位: Hz, kHz, MHz 或 GHz, 默认为 Hz

参数范围 调制波为正弦波时: 0.01 Hz ~ 100 kHz,  
调制波为方波时: 0.01 Hz ~ 20 kHz。

返回值 浮点型, 单位: Hz

默认值 10 kHz

对应菜单 **MOD** > 调相 > PM 调制频率

示例 `:PM:FREQuency 1 kHz`  
`:PM:FREQuency?`

返回值:  
`1000\n`

#### 3.4.7.4.6 外调制灵敏度 ([:SOURce]:PM:SENSitivity?)

命令格式 `[:SOURce]:PM:SENSitivity?`

功能描述 查询相位调制的外调制灵敏度。

返回值 浮点型, 单位: rad/V

默认值 无

对应菜单 **MOD** > 调相 > 外调制灵敏度

示例 `PM:SENSitivity?`

返回值:  
`0.25\n`

### 3.4.8 脉冲调制设置

#### 3.4.8.1 脉冲调制开关状态 ([:SOURce]:PULM:STATe)

命令格式	[:SOURce]:PULM:STATe ON OFF 1 0 [:SOURce]:PULM:STATe?
功能描述	设置脉冲调制的调制开关状态。 查询脉冲调制的调制开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<span style="border: 1px solid black; padding: 2px;">MOD</span> > 脉冲 > 脉冲状态
示例	<i>:PULM:STATe ON</i> <i>:PULM:STATe?</i> 返回值: <i>1\n</i>

#### 3.4.8.2 脉冲调制源 ([:SOURce]:PULM:SOURce)

命令格式	[:SOURce]:PULM:SOURce INTernal EXTernal [:SOURce]:PULM:SOURce?
功能描述	设置脉冲调制源为内部或外部。 查询脉冲调制源。
参数类型	枚举
参数范围	INTernal EXTernal
返回值	枚举
默认值	INTernal
等效命令	[:SOURce]:PULM:SOURce:INT INTernal EXTernal [:SOURce]:PULM:SOURce:INT?
对应菜单	<span style="border: 1px solid black; padding: 2px;">MOD</span> > 脉冲 > 脉冲源
示例	<i>:PULM:SOURce INTernal</i> <i>:PULM:SOURce?</i> 返回值: <i>INTernal\n</i>

## 3.4.8.3 脉冲调制源 ([:SOURce]:PULM:SOURce:INT)

命令格式	[:SOURce]:PULM:SOURce:INT INTernal EXTernal [:SOURce]:PULM:SOURce:INT?
功能描述	设置脉冲调制源为内部或外部。 查询脉冲调制源。
参数类型	枚举
参数范围	INTernal EXTernal
返回值	枚举
默认值	INTernal
等效命令	[:SOURce]:PULM:SOURce INTernal EXTernal [:SOURce]:PULM:SOURce?
对应菜单	<b>MOD</b> > 脉冲 > 脉冲源
示例	:PULM:SOURce:INT EXTernal :PULM:SOURce:INT? 返回值: EXTernal\n

## 3.4.8.4 脉冲输出状态 ([:SOURce]:PULM:STATe)

命令格式	[:SOURce]:PULM:OUT:STATe ON OFF 1 0 [:SOURce]:PULM:OUT:STATe?
功能描述	设置脉冲调制的脉冲输出状态。 查询脉冲调制的脉冲输出状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>MOD</b> > 脉冲 > 脉冲输出
示例	:PULM:OUT:STATe ON :PULM:OUT:STATe? 返回值: 1\n

## 3.4.8.5 脉冲输出极性 ([:SOURce]:PULM:POLarity)

命令格式	[:SOURce]:PULM:POLarity NORMal INVerted [:SOURce]:PULM:POLarity?
------	---

功能描述	设置脉冲输出极性。 查询脉冲输出极性。
参数类型	枚举
参数范围	NORMal INVerted
返回值	枚举
默认值	NORMal
对应菜单	<b>MOD</b> > 脉冲 > 脉冲输出极性
示例	<i>:PULM:POL INV</i> <i>:PULM:POLarity?</i> 返回值: <i>INVerted</i>

#### 3.4.8.6 脉冲类型 ([:SOURce]:PULM:MODE)

命令格式	<i>[:SOURce]:PULM:MODE SINGle DOUBle PTRain</i> <i>[:SOURce]:PULM:MODE?</i>
功能描述	设置脉冲类型为单脉冲、双脉冲或脉冲序列。 查询脉冲类型。
参数类型	枚举
参数范围	SINGle DOUBle PTRain
返回值	枚举
默认值	SINGle
对应菜单	<b>MOD</b> > 脉冲 > 脉冲类型
示例	<i>PULM:MODE DOUB</i> <i>PULM:MODE?</i> 返回值: <i>DOUBle</i>

#### 3.4.8.7 脉冲周期 ([:SOURce]:PULM:PERiod)

命令格式	<i>[:SOURce]:PULM:PERiod &lt;value&gt;</i> <i>[:SOURce]:PULM:PERiod?</i>
功能描述	当脉冲类型是单脉冲或双脉冲时，设置脉冲周期。 当脉冲类型是单脉冲或双脉冲时，查询脉冲周期。
参数类型	浮点型， 单位：ns，us，ms 或 s，默认为 s
参数范围	40 ns ~ 300 s

返回值	浮点型, 单位: s
默认值	10 ms
等效命令	<code>[:SOURce]:PULM:INT[1]:PERiod &lt;value&gt;</code> <code>[:SOURce]:PULM:INT[1]:PERiod?</code>
对应菜单	<b>MOD</b> > 脉冲 > 脉冲周期
示例	<code>PULM:PER 220 us</code> <code>PULM:PER?</code> 返回值: <code>0.00022\n</code>

#### 3.4.8.8 脉冲周期 ([:SOURce]:PULM:INT[1]:PERiod)

命令格式	<code>[:SOURce]:PULM:INT[1]:PERiod &lt;value&gt;</code> <code>[:SOURce]:PULM:INT[1]:PERiod?</code>
功能描述	当脉冲类型是单脉冲或双脉冲时, 设置脉冲周期。 当脉冲类型是单脉冲或双脉冲时, 查询脉冲周期。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	40 ns ~ 300 s
返回值	浮点型, 单位: s
默认值	10 ms
等效命令	<code>[:SOURce]:PULM:PERiod &lt;value&gt;</code> <code>[:SOURce]:PULM:PERiod?</code>
对应菜单	<b>MOD</b> > 脉冲 > 脉冲周期
示例	<code>:PULM:INT1:PERiod 900 ns</code> <code>:PULM:INT1:PERiod?</code> 返回值: <code>9e-07\n</code>

#### 3.4.8.9 脉冲宽度 ([:SOURce]:PULM:WIDTH)

命令格式	<code>[:SOURce]:PULM:WIDTH &lt;value&gt;</code> <code>[:SOURce]:PULM:WIDTH?</code>
功能描述	当脉冲类型是单脉冲时, 设置脉冲宽度; 当脉冲类型是双脉冲时, 设置第一个脉冲的脉冲宽度。 查询脉冲宽度。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s

参数范围	20 ns ~ 300 s
返回值	浮点型, 单位: s
默认值	2 ms
等效命令	[[:SOURce]:PULM:INT[1]:PWIDth <value> [:SOURce]:PULM:INT[1]:PWIDth?
对应菜单	<b>MOD</b> > 脉冲 > 脉冲宽度
示例	<i>PULM:WIDT 33 us</i> <i>PULM:WIDT?</i> 返回值: <i>3.3e-05\n</i>

#### 3.4.8.10 脉冲宽度 ([:SOURce]:PULM:INT[1]:PWIDth)

命令格式	[[:SOURce]:PULM:INT[1]:PWIDth <value> [:SOURce]:PULM:INT[1]:PWIDth?
功能描述	当脉冲类型是单脉冲时, 设置脉冲宽度; 当脉冲类型是双脉冲时, 设置第一个脉冲的脉冲宽度。 查询脉冲宽度。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	20 ns ~ 300 s
返回值	浮点型, 单位: s
默认值	2 ms
等效命令	[[:SOURce]:PULM:WIDTH <value> [:SOURce]:PULM:WIDTH?
对应菜单	<b>MOD</b> > 脉冲 > 脉冲宽度
示例	<i>:PULM:INT:PWIDth 400 ns</i> <i>:PULM:INT:PWIDth?</i> 返回值: <i>4e-07\n</i>

#### 3.4.8.11 双脉冲延迟 ([:SOURce]:PULM:DOUBLE:DELay)

命令格式	[[:SOURce]:PULM:DOUBLE:DELay <value> [:SOURce]:PULM:DOUBLE:DELay?
功能描述	当脉冲类型是双脉冲时, 设置双脉冲延迟。 当脉冲类型是双脉冲时, 查询双脉冲延迟。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s

参数范围	20 ns ~ 300 s
返回值	浮点型, 单位: s
默认值	4 ms
对应菜单	<b>MOD</b> > 脉冲 > 双脉冲延迟
示例	<i>:PULM:DOUBle:DELay 2 ms</i> <i>:PULM:DOUBle:DELay?</i> 返回值: <i>0.002\n</i>

#### 3.4.8.12 #2 脉冲宽度 ([:SOURce]:PULM:DOUBle:WIDTh)

命令格式	<i>[:SOURce]:PULM:DOUBle:WIDTh &lt;time&gt;</i> <i>[:SOURce]:PULM:DOUBle:WIDTh?</i>
功能描述	当脉冲类型是双脉冲时, 设置第二个脉冲的脉冲宽度。 当脉冲类型是双脉冲时, 查询第二个脉冲的脉冲宽度。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	20 ns ~ 300 s
返回值	浮点型, 单位: s
默认值	2 ms
对应菜单	<b>MOD</b> > 脉冲 > #2 脉冲宽度
示例	<i>PULM:DOUBle:WIDTh 5 ms</i> <i>PULM:DOUBle:WIDTh?</i> 返回值: <i>0.005\n</i>

#### 3.4.8.13 脉冲序列添加行 ([:SOURce]:PULM:TRAIIn:PAIR)

命令格式	<i>[:SOURce]:PULM:TRAIIn:PAIR &lt;row&gt;</i>
功能描述	复制指定行并粘贴到指定行的前面。
参数类型	整型
参数范围	1 ~ 当前脉冲序列的总行数
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列 > 添加
示例	<i>PULM:TRAIIn:PAIR 1</i>

## 3.4.8.14 脉冲序列删除行 ([:SOURce]:PULM:TRAI:n:DELeTe)

命令格式	[:SOURce]:PULM:TRAI:n:DELeTe <row>
功能描述	从脉冲序列中删除指定的行。
参数类型	整型
参数范围	1 ~ 当前脉冲序列的总行数
对应菜单	<input type="text" value="MOD"/> > 脉冲 > 脉冲序列 > 删除
示例	<i>PULM:TRAI:n:DELeTe 5</i>

## 3.4.8.15 编辑脉冲序列的正脉宽 ([:SOURce]:PULM:TRAI:n:DATA:ONTIme)

命令格式	[:SOURce]:PULM:TRAI:n:DATA:ONTIme <raw>,<on_time>
功能描述	编辑脉冲序列中指定行的正脉宽。
参数类型	行号：整型， 正脉宽：浮点型，单位：ns, us, ms 或 s，默认为 s
参数范围	行号：1 ~ 当前脉冲序列的总行数， 正脉宽：20 ns ~ 300 s
对应菜单	<input type="text" value="MOD"/> > 脉冲 > 脉冲序列
示例	<i>:PULM:TRAI:n:DATA:ONTIme 1,10 ms</i>

## 3.4.8.16 编辑脉冲序列的负脉宽 ([:SOURce]:PULM:TRAI:n:DATA:OFFTime)

命令格式	[:SOURce]:PULM:TRAI:n:DATA:OFFTime <raw>,<off_time>
功能描述	编辑脉冲序列中指定行的负脉宽。
参数类型	行号：整型， 负脉宽：浮点型，单位：ns, us, ms 或 s，默认为 s
参数范围	行号：1 ~ 当前脉冲序列的总行数， 负脉宽：20 ns ~ 300 s
对应菜单	<input type="text" value="MOD"/> > 脉冲 > 脉冲序列
示例	<i>:PULM:TRAI:n:DATA:OFFTime 1, 20 ms</i>

## 3.4.8.17 编辑脉冲序列的重复次数 ([:SOURce]:PULM:TRAI:n:DATA:COUNT)

命令格式	[:SOURce]:PULM:TRAI:n:DATA:COUNT <raw>,<count>
功能描述	编辑脉冲序列指定行的重复次数。

参数类型	行号：整型， 重复次数：整型
参数范围	行号：1 ~ 当前脉冲序列的总行数， 重复次数：1 ~ 65535
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列
示例	<code>:PULM:TRAI:DATA:COUNt 1,3</code>

#### 3.4.8.18 编辑脉冲序列 ([:SOURce]:PULM:TRAI:CHANGe)

命令格式	<code>[:SOURce]:PULM:TRAI:CHANGe &lt;row&gt;,&lt;on_time&gt;,&lt;off_time&gt;,&lt;count&gt;</code>
功能描述	编辑脉冲序列中的指定行。
参数类型	行号：整型， 正脉宽：浮点型，单位：ns, us, ms 或 s，默认为 s， 负脉宽：浮点型，单位：ns, us, ms 或 s，默认为 s， 重复次数：整型
参数范围	行号：1 ~ 扫描列表的总行数， 正脉宽：20 ns ~ 300 s， 负脉宽：20 ns ~ 300 s， 重复次数：1 ~ 65535
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列
示例	<code>:PULM:TRAI:CHANGe 2,3 ms,500 ns,4</code>

#### 3.4.8.19 查询脉冲序列数据 ([:SOURce]:PULM:TRAI:LIST?)

命令格式	<code>[:SOURce]:PULM:TRAI:LIST? &lt;begin_row&gt;,&lt;end_row&gt;</code>
功能描述	查询脉冲序列中 begin_row 到 end_row 的数据。
参数类型	整型，整型
参数范围	1 ~ 脉冲序列的总行数， 开始行 ~ 脉冲序列的总行数
返回值	字符串
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列
示例	<code>:PULM:TRAI:LIST? 1,3</code> 返回值： <code>0.001,0.005,1\s0.003,0.003,2\s0.004,0.002,1\n</code>

## 3.4.8.20 查询脉冲序列行数 ([:SOURce]:PULM:TRAI:n:COUNT?)

命令格式	[:SOURce]:PULM:TRAI:n:COUNT?
功能描述	查询脉冲序列的行数。
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列
示例	:PULM:TRAI:n:COUNT? 返回值: 5\n

## 3.4.8.21 清空脉冲序列 ([:SOURce]:PULM:TRAI:n:CLEAR)

命令格式	[:SOURce]:PULM:TRAI:n:CLEAR
功能描述	清空脉冲序列，恢复默认值。
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列 > 清空
示例	PULM:TRAI:n:CLEAR

## 3.4.8.22 加载脉冲序列 ([:SOURce]:PULM:TRAI:n:LOAD)

命令格式	[:SOURce]:PULM:TRAI:n:LOAD <"file_name">
功能描述	从 PULSTRN 文件中加载脉冲序列。
参数类型	字符串
参数范围	无
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列 > 加载
示例	PULM:TRAI:n:LOAD "U-disk3/test.pulstrn"

## 3.4.8.23 保存脉冲序列 ([:SOURce]:PULM:TRAI:n:STORE)

命令格式	[:SOURce]:PULM:TRAI:n:STORE <"file_name">
功能描述	将脉冲序列保存到 PULSTRN 文件。
参数类型	字符串
参数范围	无
对应菜单	<b>MOD</b> > 脉冲 > 脉冲序列 > 保存
示例	PULM:TRAI:n:STORE "test.pulstrn" PULM:TRAI:n:STORE "U-disk1/test.pulstrn"

## 3.4.8.24 脉冲触发输出状态 ([:SOURce]:PULM:TRIGger:STATe)

命令格式	<code>[:SOURce]:PULM:TRIGger:STATe ON OFF 1 0</code> <code>[:SOURce]:PULM:TRIGger:STATe?</code>
功能描述	设置脉冲触发输出状态。 查询脉冲触发输出状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	1
对应菜单	<b>MOD</b> > 脉冲 > 触发输出
示例	<code>:PULM:TRIGger:STATe OFF</code> <code>:PULM:TRIGger:STATe?</code> 返回值: <code>0\n</code>

## 3.4.8.25 脉冲触发方式 ([:SOURce]:PULM:TRIGger:MODE)

命令格式	<code>[:SOURce]:PULM:TRIGger:MODE AUTO KEY EXTErnal EGATe</code> <code>[:SOURce]:PULM:TRIGger:MODE?</code>
功能描述	设置脉冲触发方式为自动、按键、总线、外部触发或外部门控。 查询脉冲触发方式。
参数类型	枚举
参数范围	AUTO KEY EXTErnal EGATe
返回值	枚举
默认值	AUTO
对应菜单	<b>MOD</b> > 脉冲 > 触发方式
示例	<code>:PULM:TRIG:MODE EXTErnal</code> <code>:PULM:TRIGger:MODE?</code> 返回值: <code>EXTErnal\n</code>

## 3.4.8.26 脉冲外触发延迟 ([:SOURce]:PULM:DELAy)

命令格式	<code>[:SOURce]:PULM:DELAy &lt;value&gt;</code> <code>[:SOURce]:PULM:DELAy?</code>
功能描述	设置脉冲外部触发信号的触发延迟时间。 查询脉冲外部触发信号的触发延迟时间。

参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	140 ns ~ 300 s
返回值	浮点型, 单位: s
默认值	140 ns
对应菜单	<b>MOD</b> > 脉冲 > 触发延迟
示例	<i>:PULM:DEL 30 ms</i> <i>:PULM:DELay?</i> 返回值: <i>0.03</i>

#### 3.4.8.27 脉冲外触发沿 ([:SOURce]:PULM:TRIGger:EXTernal:SLOPe)

命令格式	<i>[:SOURce]:PULM:TRIGger:EXTernal:SLOPe NEGative POSitive</i> <i>[:SOURce]:PULM:TRIGger:EXTernal:SLOPe?</i>
功能描述	设置脉冲外部触发信号的触发沿。 查询脉冲外部触发信号的触发沿。
参数类型	枚举
参数范围	NEGative POSitive
返回值	枚举
默认值	POSitive
对应菜单	<b>MOD</b> > 脉冲 > 触发沿
示例	<i>PULM:TRIG:EXT:SLOP NEG</i> <i>PULM:TRIG:EXT:SLOP?</i> 返回值: <i>NEGative</i>

#### 3.4.8.28 脉冲外部门控触发极性 ([:SOURce]:PULM:TRIGger:EXTernal:GATE:POLarity)

命令格式	<i>[:SOURce]:PULM:TRIGger:EXTernal:GATE:POLarity NORMal INVerted</i> <i>[:SOURce]:PULM:TRIGger:EXTernal:GATE:POLarity?</i>
功能描述	设置脉冲外部门控信号的触发极性。 查询脉冲外部门控信号的触发极性。
参数类型	枚举
参数范围	NORMal INVerted
返回值	枚举
默认值	NORMal

对应菜单	<b>MOD</b> > 脉冲 > 触发极性
示例	<i>:PULM:TRIG:EXT:GATE:POL INVerted</i> <i>:PULM:TRIGger:EXTernal:GATE:POLarity?</i> 返回值: <i>INVerted\n</i>

### 3.4.9 LF 源设置

#### 3.4.9.1 LF 输出状态 ([:SOURce]:LFOutput)

命令格式	<i>[:SOURce]:LFOutput ON OFF 1 0</i> <i>[:SOURce]:LFOutput?</i>
功能描述	设置 LF 的输出状态。 查询 LF 的输出状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>LF</b> > LF 源 > LF 状态
示例	<i>LFOutput ON</i> <i>LFOutput?</i> 返回值: <i>1\n</i>

#### 3.4.9.2 LF 电平 ([:SOURce]:LFOutput:VOLTage)

命令格式	<i>[:SOURce]:LFOutput:VOLTage &lt;voltage&gt;</i> <i>[:SOURce]:LFOutput:VOLTage?</i>
功能描述	设置 LF 输出信号的电平。 查询 LF 输出信号的电平。
参数类型	浮点型, 单位: dBm, uVpp, mVpp, Vpp, nW, uW 或 mW, 默认为 Vpp
参数范围	1 mVpp ~ 3 Vpp
返回值	浮点型, 单位: Vpp
默认值	0.5 Vpp
对应菜单	<b>LF</b> > LF 源 > LF 电平
示例	<i>LFOutput:VOLTage 2 Vpp</i> <i>LFOutput:VOLTage?</i>

返回值:

2\n

### 3.4.9.3 LF 幅度偏移 ([:SOURce]:LFOuTput:OFFSEt)

命令格式	[:SOURce]:LFOuTput:OFFSEt <voltage> [:SOURce]:LFOuTput:OFFSEt?
功能描述	设置 LF 输出信号的幅度偏移。 查询 LF 输出信号的幅度偏移。
参数类型	浮点型, 单位: dBm, uV, mV, V, nW, uW 或 mW, 默认为 V
参数范围	$ LFOffset  \leq \min(2.5V - \frac{1}{2}LEVEL, 2V)$
返回值	浮点型, 单位: V
默认值	0
对应菜单	<b>LF</b> > LF 源 > LF 幅度偏移
示例	<i>LFOutput:OFFSEt 1 V</i> <i>LFOutput:OFFSEt?</i> 返回值: 1\n

### 3.4.9.4 LF 频率 ([:SOURce]:LFOuTput:FREQuency)

命令格式	[:SOURce]:LFOuTput:FREQuency <freq> [:SOURce]:LFOuTput:FREQuency?
功能描述	设置 LF 输出信号的频率。 查询 LF 输出信号的频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	LF 波形为正弦波时: 0.01 Hz ~ 1 MHz LF 波形为方波/锯齿波/三角波时: 0.01 Hz ~ 20 kHz
返回值	浮点型, 单位: Hz
默认值	1 kHz
对应菜单	<b>LF</b> > LF 源 > LF 频率
示例	<i>LFOutput:FREQuency 10 kHz</i> <i>LFOutput:FREQuency?</i> 返回值: 10000\n

## 3.4.9.5 LF 波形 ([:SOURce]:LFOutput:SHAPE)

命令格式	[:SOURce]:LFOutput:SHAPE SINE SQUare TRIangle SAWTooth DC [:SOURce]:LFOutput:SHAPE?
功能描述	设置 LF 输出信号的波形。 查询 LF 输出信号的波形。
参数类型	枚举
参数范围	SINE SQUare TRIangle SAWTooth DC
返回值	枚举
默认值	SINE
对应菜单	<b>LF</b> > LF 源 > LF 波形
示例	<i>:LFOutput:SHAPE TRIangle</i> <i>:LFOutput:SHAPE?</i> 返回值: <i>TRIangle</i>

## 3.4.9.6 LF 相位 ([:SOURce]:LFOutput:PHASe)

命令格式	[:SOURce]:LFOutput:PHASe <deg> [:SOURce]:LFOutput:PHASe?
功能描述	设置 LF 输出信号的相位。 查询 LF 输出信号的相位。
参数类型	浮点型, 单位: 度 (°)
参数范围	-360 ~ 360
返回值	浮点型, 单位: 度 (°)
默认值	0
对应菜单	<b>LF</b> > LF 源 > LF 相位
示例	<i>LFOutput:PHASe 20</i> <i>LFOutput:PHASe?</i> 返回值: <i>20</i>

## 3.4.10 LF 扫描设置

## 3.4.10.1 LF 扫描状态 ([:SOURce]:LFOutput:SWEep)

命令格式	[:SOURce]:LFOutput:SWEep ON OFF 1 0 [:SOURce]:LFOutput:SWEep?
功能描述	设置 LF 的扫描状态。

	查询 LF 的扫描状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>LF</b> > LF 扫描 > 扫描状态
示例	<i>:LFOutput:SWEEP 1</i> <i>:LFOutput:SWEEP?</i> 返回值: <i>1\n</i>

#### 3.4.10.2 LF 扫描方向 ([:SOURce]:LFOutput:SWEEP:DIRect)

命令格式	<i>[:SOURce]:LFOutput:SWEEP:DIRect UP DOWN</i> <i>[:SOURce]:LFOutput:SWEEP:DIRect?</i>
功能描述	设置 LF 扫描的方向。 查询 LF 扫描的方向。
参数类型	枚举
参数范围	UP DOWN
返回值	枚举
默认值	UP
对应菜单	<b>LF</b> > LF 扫描 > 扫描方向
示例	<i>:LFOutput:SWEEP:DIRect DOWN</i> <i>:LFOutput:SWEEP:DIRect?</i> 返回值: <i>DOWN\n</i>

#### 3.4.10.3 LF 扫描开始频率 ([:SOURce]:LFOutput:SWEEP:STARt:FREQuency)

命令格式	<i>[:SOURce]:LFOutput:SWEEP:STARt:FREQuency &lt;freq&gt;</i> <i>[:SOURce]:LFOutput:SWEEP:STARt:FREQuency?</i>
功能描述	设置 LF 扫描的开始频率。 查询 LF 扫描的开始频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	LF 波形为正弦波时: 0.01 Hz ~ 1 MHz LF 波形为方波/锯齿波/三角波时: 0.01 Hz ~ 20 kHz

返回值	浮点型, 单位: Hz
默认值	500 Hz
对应菜单	<b>LF</b> > LF 扫描 > 开始频率
示例	<code>:LFOutput:SWEep:STARt:FREQuency 100</code> <code>:LFOutput:SWEep:STARt:FREQuency?</code> 返回值: <code>100\n</code>

#### 3.4.10.4 LF 扫描结束频率 ([:SOURce]:LFOutput:SWEep:STOP:FREQuency)

命令格式	<code>[:SOURce]:LFOutput:SWEep:STOP:FREQuency &lt;freq&gt;</code> <code>[:SOURce]:LFOutput:SWEep:STOP:FREQuency?</code>
功能描述	设置 LF 扫描的结束频率。 查询 LF 扫描的结束频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	LF 波形为正弦波时: 0.01 Hz ~ 1 MHz LF 波形为方波/锯齿波/三角波时: 0.01 Hz ~ 20 kHz
返回值	浮点型, 单位: Hz
默认值	1.5 kHz
对应菜单	<b>LF</b> > LF 扫描 > 结束频率
示例	<code>:LFOutput:SWEep:STOP:FREQuency 1000</code> <code>:LFOutput:SWEep:STOP:FREQuency?</code> 返回值: <code>1000\n</code>

#### 3.4.10.5 LF 扫描中心频率 ([:SOURce]:LFOutput:SWEep:CENTer:FREQuency)

命令格式	<code>[:SOURce]:LFOutput:SWEep:CENTer:FREQuency &lt;freq&gt;</code> <code>[:SOURce]:LFOutput:SWEep:CENTer:FREQuency?</code>
功能描述	设置 LF 扫描的中心频率。 查询 LF 扫描的中心频率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	LF 波形为正弦波时: 0.01 Hz ~ 1 MHz LF 波形为方波/锯齿波/三角波时: 0.01 Hz ~ 20 kHz
返回值	浮点型, 单位: Hz

默认值	1 kHz
对应菜单	<input type="text" value="LF"/> > LF 扫描 > 中心频率
示例	<code>:LFOutput:SWEep:CENTer:FREQuency 550</code> <code>:LFOutput:SWEep:CENTer:FREQuency?</code> 返回值: <code>550\n</code>

#### 3.4.10.6 LF 扫描宽度 ([:SOURce]:LFOutput:SWEep:SPAN:FREQuency)

命令格式	<code>[:SOURce]:LFOutput:SWEep:SPAN:FREQuency &lt;freq&gt;</code> <code>[:SOURce]:LFOutput:SWEep:SPAN:FREQuency?</code>
功能描述	设置 LF 扫描的频率宽度。 查询 LF 扫描的频率宽度。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	LF 波形为正弦波时: 0.00 Hz ~ 999.99999 kHz LF 波形为方波/锯齿波/三角波时: 0.00 Hz ~ 19.99999 kHz
返回值	浮点型, 单位: Hz
默认值	1 kHz
对应菜单	<input type="text" value="LF"/> > LF 扫描 > 扫描宽度
示例	<code>:LFOutput:SWEep:SPAN:FREQuency 550</code> <code>:LFOutput:SWEep:SPAN:FREQuency?</code> 返回值: <code>550\n</code>

#### 3.4.10.7 LF 扫描时间 ([:SOURce]:LFOutput:SWEep:DWELI)

命令格式	<code>[:SOURce]:LFOutput:SWEep:DWELI &lt;time&gt;</code> <code>[:SOURce]:LFOutput:SWEep:DWELI?</code>
功能描述	设置 LF 扫描的扫描时间。 查询 LF 扫描的扫描时间。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s
参数范围	1 ms ~ 500 s
返回值	浮点型, 单位: s
默认值	1 s
对应菜单	<input type="text" value="LF"/> > LF 扫描 > 扫描时间

示例	<code>:LFOutput:SWEEp:DWELI 2 s</code> <code>:LFOutput:SWEEp:DWELI?</code>
	返回值: <code>2\n</code>

### 3.4.10.8 LF 扫描触发方式 ([:SOURce]:LFOutput:SWEEp:TRIGger:TYPE)

命令格式	<code>[:SOURce]:LFOutput:SWEEp:TRIGger:TYPE AUTO KEY BUS EXT</code> <code>[:SOURce]:LFOutput:SWEEp:TRIGger:TYPE?</code>
功能描述	设置 LF 扫描的触发方式。 查询 LF 扫描的触发方式。
参数类型	枚举
参数范围	AUTO KEY BUS EXT
返回值	枚举
默认值	AUTO
对应菜单	<input type="text" value="LF"/> > LF 扫描 > 触发方式
示例	<code>:LFOutput:SWEEp:TRIGger:TYPE KEY</code> <code>:LFOutput:SWEEp:TRIGger:TYPE?</code>
	返回值: <code>KEY\n</code>

### 3.4.10.9 LF 扫描触发沿 ([:SOURce]:LFOutput:SWEEp:XPOLar)

命令格式	<code>[:SOURce]:LFOutput:SWEEp:XPOLar POS NEG</code> <code>[:SOURce]:LFOutput:SWEEp:XPOLar?</code>
功能描述	设置 LF 扫描的外部触发信号的触发沿。 查询 LF 扫描的外部触发信号的触发沿。
参数类型	枚举
参数范围	POS NEG
返回值	枚举
默认值	POS
对应菜单	<input type="text" value="LF"/> > LF 扫描 > 触发沿
示例	<code>:LFOutput:SWEEp:XPOLar POS</code> <code>:LFOutput:SWEEp:XPOLar?</code>
	返回值: <code>POS\n</code>

### 3.4.10.10 LF 扫描形状 ([:SOURce]:LFOutput:SWEep:SHAPE)

命令格式	<code>[:SOURce]:LFOutput:SWEep:SHAPE TRlangle SAWTooth</code> <code>[:SOURce]:LFOutput:SWEep:SHAPE?</code>
功能描述	设置 LF 扫描的扫描形状。 查询 LF 扫描的扫描形状。
参数类型	枚举
参数范围	TRlangle SAWTooth
返回值	枚举
默认值	SAWTooth
对应菜单	<b>LF</b> > LF 扫描 > 扫描形状
示例	<code>:LFOutput:SWEep:SHAPE TRlangle</code> <code>:LFOutput:SWEep:SHAPE?</code> 返回值: <code>TRlangle\n</code>

### 3.4.10.11 LF 扫描频率步进方式 ([:SOURce]:LFOutput:SWEep:SPACing)

命令格式	<code>[:SOURce]:LFOutput:SWEep:SPACing LINear LOGarithmic</code> <code>[:SOURce]:LFOutput:SWEep:SPACing?</code>
功能描述	设置 LF 扫描的频率步进方式。 查询 LF 扫描的频率步进方式。
参数类型	枚举
参数范围	LINear LOGarithmic
返回值	枚举
默认值	LINear
对应菜单	<b>LF</b> > LF 扫描 > 扫描方式
示例	<code>:LFOutput:SWEep:SPACing LOGarithmic</code> <code>:LFOutput:SWEep:SPACing?</code> 返回值: <code>LOGarithmic\n</code>

## 3.4.11 系统复位

### 3.4.11.1 系统复位 (:SOURce:PRESet)

命令格式	<code>:SOURce:PRESet</code>
功能描述	将仪器状态设置为出厂设置。

对应菜单	无
示例	<code>:SOURce:PRESet</code>

### 3.4.12 IQ 调制设置

#### 3.4.12.1 IQ 调制总开关状态 ([:SOURce]:FUNCTion:DM:STATe)

命令格式	<code>[:SOURce]:FUNCTion:DM:STATe ON OFF 1 0</code> <code>[:SOURce]:FUNCTion:DM:STATe?</code>
功能描述	此命令设置/查询 IQ 调制的总开关状态。 注：开启 IQ 调制功能时，必须打开该开关才能开启调制功能。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>HOME</b> > IQ MOD > On, 或 <b>MOD ON/OFF</b>
示例	<code>:FUNCTion:DM:STATe ON</code> <code>:FUNCTion:DM:STATe?</code> 返回值: <code>1\n</code>

### 3.4.13 Custom 调制设置

#### 3.4.13.1 Custom 调制开关状态 ([:SOURce]:RADio:CUSTom[:STATe])

命令格式	<code>[:SOURce]:RADio:CUSTom[:STATe] ON OFF 1 0</code> <code>[:SOURce]:RADio:CUSTom[:STATe]?</code>
功能描述	设置 Custom 调制的开关状态。 查询 Custom 调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > Custom > Custom 状态
示例	<code>:RADio:CUSTom 1</code> <code>:RADio:CUSTom?</code> 返回值: <code>1\n</code>

## 3.4.13.2 数据源 ([:SOURce]:RADio:CUSTom:DATA)

命令格式	[:SOURce]:RADio:CUSTom:DATA PN7 PN9 PN15 PN23 USER [:SOURce]:RADio:CUSTom:DATA?
功能描述	此命令设置/查询 Custom 调制的数据源。
参数类型	枚举
参数范围	PN7 PN9 PN15 PN23 USER
返回值	枚举
默认值	PN7
对应菜单	<b>IQ</b> > Custom > 数据源 > 数据源
示例	:RADio:CUSTom:DATA PN9 :RADio:CUSTom:DATA? 返回值: PN9\n

## 3.4.13.3 符号率 ([:SOURce]:RADio:CUSTom:SRATe)

命令格式	[:SOURce]:RADio:CUSTom:SRATe <val> [:SOURce]:RADio:CUSTom:SRATe?
功能描述	此命令设置/查询 Custom 调制的符号率。
参数类型	浮点型, 单位: Sps
参数范围	500 Sps ~ 120 MSps
返回值	浮点型, 单位: Sps
默认值	1 MSps
对应菜单	<b>IQ</b> > Custom > 数据源 > 符号率
示例	:RADio:CUSTom:SRATe 2000000 :RADio:CUSTom:SRATe? 返回值: 2000000\n

## 3.4.13.4 符号长度 ([:SOURce]:RADio:CUSTom:SLENgth)

命令格式	[:SOURce]:RADio:CUSTom:SLENgth <val> [:SOURce]:RADio:CUSTom:SLENgth?
功能描述	此命令设置/查询 Custom 调制的符号长度。
参数类型	整型
参数范围	100 ~ 100000
返回值	整型

默认值	512
对应菜单	<input type="text" value="IQ"/> > Custom > 数据源 > 符号长度
示例	<code>:RADio:CUSTom:SLENgth 1024</code> <code>:RADio:CUSTom:SLENgth?</code> 返回值: <code>1024\n</code>

#### 3.4.13.5 查询位/符号 ([:SOURce]:RADio:CUSTom:SBIT?)

命令格式	<code>[:SOURce]:RADio:CUSTom:SBIT?</code>
功能描述	此命令查询 Custom 调制的比特位/符号 (bits per symbol) 值。
返回值	整型
默认值	4
对应菜单	<input type="text" value="IQ"/> > Custom > 数据源 > 位/符号
示例	<code>:RADio:CUSTom:SBIT?</code> 返回值: <code>4\n</code>

#### 3.4.13.6 调制类型 ([:SOURce]:RADio:CUSTom:MODulation[:TYPE])

命令格式	<code>[:SOURce]:RADio:CUSTom:MODulation[:TYPE] ASK2  ASK4  ASK8  ASK16  BPSK  QPSK  PSK8  DBPSK  DQPSK  D8PSK  PI4DQPSK  PI8D8PSK  OQPSK  QAM16  QAM32  QAM64  QAM128  QAM256  QAM512  FSK2  FSK4  FSK8  FSK16  MSK1  USER</code> <code>[:SOURce]:RADio:CUSTom:MODulation[:TYPE]?</code>
功能描述	此命令设置/查询 Custom 调制的调制类型。
参数类型	枚举
参数范围	ASK2  ASK4  ASK8  ASK16  BPSK  QPSK  PSK8  DBPSK  DQPSK  D8PSK  PI4DQPSK  PI8D8PSK  OQPSK  QAM16  QAM32  QAM64  QAM128  QAM256  QAM512  FSK2  FSK4  FSK8  FSK16  MSK1  USER
返回值	枚举
默认值	QAM16
对应菜单	<input type="text" value="IQ"/> > Custom > 调制设置 > 调制类型
示例	<code>:RADio:CUSTom:MODulation ASK2</code> <code>:RADio:CUSTom:MODulation?</code> 返回值: <code>ASK2\n</code>

## 3.4.13.7 格雷码 ([:SOURce]:RADio:CUSTom:MODulation:GRAY)

命令格式	[:SOURce]:RADio:CUSTom:MODulation:GRAY ON OFF 1 0 [:SOURce]:RADio:CUSTom:MODulation:GRAY?
功能描述	此命令设置/查询调制符号是否使用格雷码编码。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > Custom > 调制设置 > 格雷码
示例	:RADio:CUSTom:MODulation:GRAY 1 :RADio:CUSTom:MODulation:GRAY? 返回值: 1\n

## 3.4.13.8 保存自定义 IQ 数据 ([:SOURce]:RADio:CUSTom:MODulation:STORe)

命令格式	[:SOURce]:RADio:CUSTom:MODulation:STORe <"file_name">
功能描述	将自定义 IQ 数据保存到 MAP 文件。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置 > 保存
示例	:RADio:CUSTom:MODulation:STORe "test.map"

## 3.4.13.9 加载自定义 IQ 数据 ([:SOURce]:RADio:CUSTom:MODulation:LOAD)

命令格式	[:SOURce]:RADio:CUSTom:MODulation:LOAD <"file_name">
功能描述	从 MAP 文件中加载自定义 IQ 数据。
参数类型	字符串
参数范围	无
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置 > 加载
示例	:RADio:CUSTom:MODulation:LOAD "test.map"

## 3.4.13.10 查询自定义 IQ 数据 ([:SOURce]:RADio:CUSTom:MODulation:UIQ?)

命令格式	[:SOURce]:RADio:CUSTom:MODulation:UIQ?
------	--

功能描述	查询自定义的 IQ 数据
返回值	字符串, 格式: I 值 Q 值\nI 值 Q 值\n... I 值 Q 值\n\n
默认值	0.632456 0.000000\n1.264911 0.000000\n\n
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置
示例	<code>:RADio:CUSTom:MODulation:UIQ?</code> 返回值: <code>0.632456 0.000000\n0.700000 -0.700000\n-0.700000 0.700000\n1.264910 0.000000\n\n</code>

#### 3.4.13.11 自定义 IQ 数据添加行 ([:SOURce]:RADio:CUSTom:MODulation:INSErt)

命令格式	<code>[:SOURce]:RADio:CUSTom:MODulation:INSErt &lt;symbol&gt;,&lt;i data&gt;,&lt;q data&gt;</code>
功能描述	在自定义 IQ 数据列表中插入一行。
参数类型	符号: 整型, I 值: 浮点型, Q 值: 浮点型
参数范围	符号: 0 ~ (当前符号总数 - 1), I 值: -1 ~ 1, Q 值: -1 ~ 1
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置 > 插入
示例	<code>:RADio:CUSTom:MODulation:INSErt 0,0.5,0.5</code>

#### 3.4.13.12 自定义 IQ 数据编辑行 ([:SOURce]:RADio:CUSTom:MODulation:CHANge)

命令格式	<code>[:SOURce]:RADio:CUSTom:MODulation:CHANge &lt;symbol&gt;,&lt;i data&gt;,&lt;q data&gt;</code>
功能描述	在自定义 IQ 数据列表中编辑指定的行。
参数类型	符号: 整型, I 值: 浮点型, Q 值: 浮点型
参数范围	符号: 0 ~ (当前符号总数 - 1), I 值: -1 ~ 1, Q 值: -1 ~ 1
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置
示例	<code>:RADio:CUSTom:MODulation:CHANge 0,0.5,0.5</code>

## 3.4.13.13 自定义 IQ 数据删除行 ([:SOURce]:RADio:CUSTom:MODulation:DELEte)

命令格式	[:SOURce]:RADio:CUSTom:MODulation:DELEte <symbol>
功能描述	从自定义 IQ 数据列表中删除指定的行。
参数类型	整型
参数范围	0 ~ (当前符号总数 - 1)
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置 > 删除
示例	:RADio:CUSTom:MODulation:DELEte 0

## 3.4.13.14 清空自定义 IQ 数据列表 ([:SOURce]:RADio:CUSTom:MODulation:CLEAR)

命令格式	[:SOURce]:RADio:CUSTom:MODulation:CLEAR
功能描述	该命令重置自定义 IQ 数据列表为默认值。
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (自定义) > 自定义设置 > 清空
示例	:RADio:CUSTom:MODulation:CLEAR

## 3.4.13.15 FSK 频偏 ([:SOURce]:RADio:CUSTom:MODulation:FSK[:DEViation])

命令格式	[:SOURce]:RADio:CUSTom:MODulation:FSK[:DEViation] <val> [:SOURce]:RADio:CUSTom:MODulation:FSK[:DEViation]?
功能描述	此命令设置/查询 FSK 调制的频偏值。
参数类型	浮点型, 单位: Hz
参数范围	0 ~ 0.8*符号率*过采样倍数
返回值	浮点型, 单位: Hz
默认值	600000
对应菜单	<b>IQ</b> > Custom > 调制设置 > 调制类型 (FSK) > FSK 频偏
示例	:RADio:CUSTom:MODulation:FSK:DEViation 450e3 :RADio:CUSTom:MODulation:FSK:DEViation? 返回值: 450000\n

## 3.4.13.16 滤波器类型 ([:SOURce]:RADio:CUSTom:FILTer)

命令格式	[:SOURce]:RADio:CUSTom:FILTer NONE  RAISecosine  ROOTcosine  GAUSSian [:SOURce]:RADio:CUSTom:FILTer?
功能描述	此命令设置/查询 Custom 调制的滤波器类型。

参数类型	枚举
参数范围	NONE RAISecosine ROOTcosine GAUSSian
返回值	枚举
默认值	ROOTcosine
对应菜单	<b>IQ</b> > Custom > 滤波器设置 > 滤波器类型
示例	:RADio:CUSTom:FILTer GAUSSian :RADio:CUSTom:FILTer? 返回值: GAUSSian\n

#### 3.4.13.17 滤波器系数 ([:SOURce]:RADio:CUSTom:ALPHa)

命令格式	[:SOURce]:RADio:CUSTom:ALPHa <val> [:SOURce]:RADio:CUSTom:ALPHa?
功能描述	此命令设置/查询升余弦或根升余弦滤波器的 Alpha 值，或高斯滤波器的 BT 值。
参数类型	浮点型
参数范围	0.010 ~ 1.000
返回值	浮点型
默认值	0.350
对应菜单	<b>IQ</b> > Custom > 滤波器设置 > 滤波器 Alpha/ 滤波器 BT
示例	:RADio:CUSTom:ALPHa 0.22 :RADio:CUSTom:ALPHa? 返回值: 0.22\n

#### 3.4.13.18 滤波器长度 ([:SOURce]:RADio:CUSTom:FILTer:LENGth)

命令格式	[:SOURce]:RADio:CUSTom:FILTer:LENGth <length> [:SOURce]:RADio:CUSTom:FILTer:LENGth?
功能描述	此命令设置/查询 Custom 调制的滤波器长度。
参数类型	整型
参数范围	1 ~ 512
返回值	整型
默认值	128
对应菜单	<b>IQ</b> > Custom > 滤波器设置 > 滤波器长度
示例	:RADio:CUSTom:FILTer:LENGth 64 :RADio:CUSTom:FILTer:LENGth? 返回值: 64\n

## 3.4.13.19 过采样倍数 ([:SOURce]:RADio:CUSTom:FILTer:OSAMple)

命令格式	[:SOURce]:RADio:CUSTom:FILTer:OSAMple <val> [:SOURce]:RADio:CUSTom:FILTer:OSAMple?
功能描述	此命令设置/查询 Custom 调制的过采样倍数。
参数类型	整型
参数范围	2 ~ 32
返回值	整型
默认值	2
对应菜单	<b>IQ</b> > Custom > 滤波器设置 > 过采样倍数
示例	:RADio:CUSTom:FILTer:OSAMple 4 :RADio:CUSTom:FILTer:OSAMple? 返回值: 4\n

## 3.4.13.20 查询比特率 ([:SOURce]:RADio:CUSTom:BRATe?)

命令格式	[:SOURce]:RADio:CUSTom:BRATe?
功能描述	此命令查询 Custom 调制的比特率。
返回值	浮点型, 单位: bps
默认值	4 Mbps
对应菜单	无
示例	:RADio:CUSTom:BRATe? 返回值: 4000000\n

## 3.4.13.21 保存波形 ([:SOURce]:RADio:CUSTom:SAVE)

命令格式	[:SOURce]:RADio:CUSTom:SAVE <"file_name">
功能描述	将 Custom 调制的波形数据保存到 ARB 文件。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
对应菜单	<b>IQ</b> > Custom > 保存波形
示例	:RADio:CUSTom:SAVE "test.arb"

## 3.4.13.22 更新 ([:SOURce]:RADio:CUSTom:DOWNload)

命令格式	[:SOURce]:RADio:CUSTom:DOWNload
功能描述	此命令更新 Custom 调制的设置。
对应菜单	<input type="text" value="IQ"/> > Custom > 更新
示例	:RADio:CUSTom:DOWNload

## 3.4.14 ARB 调制设置

## 3.4.14.1 ARB 调制开关状态 ([:SOURce]:RADio:ARB[:STATe])

命令格式	[:SOURce]:RADio:ARB[:STATe] ON OFF 1 0 [:SOURce]:RADio:ARB[:STATe]?
功能描述	此命令设置/查询 ARB 调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<input type="text" value="IQ"/> > ARB > ARB 状态
示例	:RADio:ARB 1 :RADio:ARB? 返回值: 1\n

## 3.4.14.2 波形选择 ([:SOURce]:RADio:ARB:WAVeform)

命令格式	[:SOURce]:RADio:ARB:WAVeform <"WFM:segment" "SEQ:sequence"> [:SOURce]:RADio:ARB:WAVeform?
功能描述	此命令设置/查询 ARB 当前播放的波形段或波形序列。
参数类型	字符串
参数范围	"WFM:segment": <input type="text" value="IQ"/> > ARB > ARB 波形段列表中的波形名称, "SEQ:sequence": <input type="text" value="IQ"/> > ARB > ARB 波形序列列表中的波形名称。
返回值	字符串
默认值	*NONE
对应菜单	<input type="text" value="IQ"/> > ARB > 波形选择
示例	:RADio:ARB:WAVeform "WFM:SINE_WAVE" :RADio:ARB:WAVeform?

返回值:

*WFM:SINE\_WAVE**n*

*:RADio:ARB:WAVeform "SEQ:test\_seq"*

*:RADio:ARB:WAVeform?*

返回值:

*SEQ:test\_seq**n*

#### 3.4.14.3 获取波形段列表 ([:SOURce]:IQ:DUALarb:SEGMent:NAMES?)

命令格式	<code>[:SOURce]:IQ:DUALarb:SEGMent:NAMES?</code>
功能描述	此命令查询 ARB 调制中已加载的波形段列表。
返回值	字符串, 格式: 波形段名称 点数\n 波形段名称 点数\n...波形段名称 点数\n
默认值	RAMP_WAVE 200 SINE_WAVE 200
对应菜单	<input type="checkbox"/> IQ > ARB > 波形段
示例	<i>:IQ:DUALarb:SEGMent:NAMES?</i> 返回值: <i>RAMP_WAVE 200\nSINE_WAVE 200\n</i>

#### 3.4.14.4 加载波形段 ([:SOURce]:IQ:DUALarb:SEGMent:LOAD)

命令格式	<code>[:SOURce]:IQ:DUALarb:SEGMent:LOAD &lt;"file_name"&gt;</code>
功能描述	此命令在 ARB 段列表中加载波形段文件, 可加载文件类型: *.arb, *.wdbin, *.txt。
参数类型	字符串
参数范围	无
等效命令	<code>[:SOURce]:IQ:DUALarb:SEGMent:LOAD:DATA &lt;"file_name"&gt;</code> <code>[:SOURce]:IQ:DUALarb:SEGMent:LOAD:TEXT &lt;"file_name"&gt;</code>
对应菜单	<input type="checkbox"/> IQ > ARB > 波形段 > 加载
示例	<i>:IQ:DUALarb:SEGMent:LOAD "Local/test.arb"</i> <i>:IQ:DUALarb:SEGMent:LOAD "Local/sine.wdbin"</i> <i>:IQ:DUALarb:SEGMent:LOAD "Local/16qam.txt"</i>

#### 3.4.14.5 加载波形段 ([:SOURce]:IQ:DUALarb:SEGMent:LOAD:DATA)

命令格式	<code>[:SOURce]:IQ:DUALarb:SEGMent:LOAD:DATA &lt;"file_name"&gt;</code>
功能描述	此命令在 ARB 调制中加载波形段, 可加载文件类型: *.arb, *.wdbin, *.txt。

参数类型	字符串
参数范围	无
等效命令	<code>[[:SOURce]:IQ:DUALarb:SEGMent:LOAD &lt;"file_name"&gt;</code> <code>[[:SOURce]:IQ:DUALarb:SEGMent:LOAD:TEXT &lt;"file_name"&gt;</code>
对应菜单	<b>IQ</b> > ARB > 波形段 > 加载
示例	<code>:IQ:DUALarb:SEGMent:LOAD:DATA "Local/test.arb"</code> <code>:IQ:DUALarb:SEGMent:LOAD:DATA "Local/sine.wdbin"</code> <code>:IQ:DUALarb:SEGMent:LOAD:DATA "Local/16qam.txt"</code>

#### 3.4.14.6 加载波形段 ([[:SOURce]:IQ:DUALarb:SEGMent:LOAD:TEXT])

命令格式	<code>[[:SOURce]:IQ:DUALarb:SEGMent:LOAD:TEXT &lt;"file_name"&gt;</code>
功能描述	此命令在 ARB 调制中加载波形段，可加载文件类型：*.arb, *.wdbin, *.txt。
参数类型	字符串
参数范围	无
等效命令	<code>[[:SOURce]:IQ:DUALarb:SEGMent:LOAD &lt;"file_name"&gt;</code> <code>[[:SOURce]:IQ:DUALarb:SEGMent:LOAD:DATA &lt;"file_name"&gt;</code>
默认值	无
对应菜单	<b>IQ</b> > ARB > 波形段 > 加载
示例	<code>:IQ:DUALarb:SEGMent:LOAD:TEXT "Local/test.arb"</code> <code>:IQ:DUALarb:SEGMent:LOAD:TEXT "Local/sine.wdbin"</code> <code>:IQ:DUALarb:SEGMent:LOAD:TEXT "Local/16qam.txt"</code>

#### 3.4.14.7 删除波形段 ([[:SOURce]:IQ:DUALarb:SEGMent:DEL])

命令格式	<code>[[:SOURce]:IQ:DUALarb:SEGMent:DEL &lt;"segment"&gt;</code>
功能描述	此命令删除波形段列表中的波形段。
参数类型	字符串
参数范围	<b>IQ</b> > ARB > ARB 波形段列表中的波形名称
对应菜单	<b>IQ</b> > ARB > 波形段 > 删除
示例	<code>:IQ:DUALarb:SEGMent:DEL "sine"</code>

#### 3.4.14.8 重命名波形段 ([[:SOURce]:IQ:DUALarb:SEGMent:REName])

命令格式	<code>[[:SOURce]:IQ:DUALarb:SEGMent:REName &lt;"original_name"&gt;,&lt;"new_name"&gt;</code>
功能描述	此命令重命名 ARB 段列表中的波形段的名称。
参数类型	字符串，字符串

参数范围	original_name: <input type="text" value="IQ"/> > ARB > ARB 波形段列表中的波形名称, new_name: 命名规则请参阅用户手册。
对应菜单	<input type="text" value="IQ"/> > ARB > 波形段 > 重命名
示例	<code>:IQ:DUALarb:SEGMent:RENAmE "test","RENAME_WAVE"</code>
	注: 此 SCPI 命令不包含回车符或换行符。上述命令格式中包含的是空格。

#### 3.4.14.9 清空波形段 ([:SOURce]:IQ:DUALarb:SEGMent:CLEAR)

命令格式	<code>[:SOURce]:IQ:DUALarb:SEGMent:CLEAR</code>
功能描述	此命令清空 ARB 段列表。 注: 正在播放的波形段会保留。
对应菜单	<input type="text" value="IQ"/> > ARB > 波形段 > 清空
示例	<code>:IQ:DUALarb:SEGMent:CLEAR</code>

#### 3.4.14.10 创建波形序列 ([:SOURce]:RADio:ARB:SEQuence)

命令格式	<code>[:SOURce]:RADio:ARB:SEQuence &lt;"file_name"&gt;,&lt;"waveform1"&gt;,&lt;reps&gt;,&lt;marker&gt;,{&lt;"waveform2"&gt;,&lt;reps&gt;,&lt;marker&gt;,...} [:SOURce]:RADio:ARB:SEQuence? &lt;"file_name"&gt;</code>
功能描述	此命令创建一个波形序列, 波形序列由段和其他序列组成。段和序列的播放顺序与命令放入波形序列中的顺序相同。 查询命令返回波形序列的内容。
参数类型	file_name: 字符串, 要创建的波形序列名称, waveform: 字符串, 波形段或序列名称, reps: 整型, 波形重复次数, marker: 枚举, 波形的标识点。
参数范围	file_name: 命名规则请参阅用户手册, waveform: ARB 波形段或波形序列列表中的波形, reps: 1 ~ 65535, marker: NONE  M1  M2  M3  M4  M1M2  M1M3  M1M4  M2M3  M2M4  M3M4  M1M2M3  M1M2M4  M1M3M4  M2M3M4  ALL
返回值	字符串
默认值	无
对应菜单	<input type="text" value="IQ"/> > ARB > 波形序列 > 创建
示例	<code>:RADio:ARB:SEQuence "SEQ:Test_Data","WFM:ramp_wave",25,M1M4,"WFM:sine_wave",100,</code>

```
ALL,"SEQ:seq1",3,NONE
```

```
:RADio:ARB:SEquence? "SEQ:Test_Data"
```

返回值:

```
"WFM:ramp_wave",25,M1M4,"WFM:sine_wave",100,ALL,"SEQ:seq1",3,NONE\n
```

注：此 SCPI 命令不包含回车符或换行符。上述示例中包含的是空格。

#### 3.4.14.11 采样时钟 ([:SOURce]:RADio:ARB:SClock:RATE)

命令格式	[:SOURce]:RADio:ARB:SClock:RATE <rate> [:SOURce]:RADio:ARB:SClock:RATE?
功能描述	此命令设置/查询 ARB 调制的采样时钟。
参数类型	浮点型，单位：Hz，kHz，MHz 或 GHz，默认为 Hz
参数范围	0.002 Hz ~ 240 MHz
返回值	浮点型，单位：Hz
默认值	2 MHz
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 采样时钟
示例	:RADio:ARB:SClock:RATE 4 MHz :RADio:ARB:SClock:RATE? 返回值: 4000000\n

#### 3.4.14.12 调制器衰减类型 ([:SOURce]:RADio:ARB:IQ:MODulation:ATTen:AUTO)

命令格式	[:SOURce]:RADio:ARB:IQ:MODulation:ATTen:AUTO AUTO MANUal [:SOURce]:RADio:ARB:IQ:MODulation:ATTen:AUTO?
功能描述	此命令设置/查询 ARB 调制器的衰减类型。
参数类型	枚举
参数范围	AUTO MANUal
返回值	枚举
默认值	AUTO
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制器衰减类型
示例	:RADio:ARB:IQ:MODulation:ATTen:AUTO AUTO :RADio:ARB:IQ:MODulation:ATTen:AUTO? 返回值: AUTO\n

## 3.4.14.13 调制器衰减值 ([:SOURce]:RADio:ARB:IQ:MODulation:ATTen)

命令格式	[:SOURce]:RADio:ARB:IQ:MODulation:ATTen <val> [:SOURce]:RADio:ARB:IQ:MODulation:ATTen?
功能描述	此命令设置/查询 ARB 调制器的衰减值。
参数类型	浮点型, 单位: dB
参数范围	0 ~ 20
返回值	浮点型, 单位: dB
默认值	3
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制衰减
示例	:RADio:ARB:IQ:MODulation:ATTen 10 :RADio:ARB:IQ:MODulation:ATTen? 返回值: 10\n

## 3.4.14.14 实时 AWGN 状态 ([:SOURce]:RADio:ARB:NOISe[:STATe])

命令格式	[:SOURce]:RADio:ARB:NOISe[:STATe] ON OFF 1 0 [:SOURce]:RADio:ARB:NOISe[:STATe]?
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN
示例	:RADio:ARB:NOISe 1 :RADio:ARB:NOISe? 返回值: 1\n

## 3.4.14.15 实时 AWGN 输出类型 ([:SOURce]:RADio:ARB:NOISe:OUTPut)

命令格式	[:SOURce]:RADio:ARB:NOISe:OUTPut CARRier NOISe CARRier_NOISe [:SOURce]:RADio:ARB:NOISe:OUTPut?
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的输出类型。
参数类型	枚举
参数范围	CARRier NOISe CARRier_NOISe
返回值	枚举

默认值	CARRier_NOISe
对应菜单	<input type="checkbox"/> IQ > ARB > ARB 设置 > 实时 AWGN > 输出选择
示例	<i>:RADio:ARB:NOISe:OUTPut CARRier</i> <i>:RADio:ARB:NOISe:OUTPut?</i> 返回值: <i>CARRier\n</i>

#### 3.4.14.16 实时 AWGN 功率控制模式 ([:SOURce]:RADio:ARB:NOISe:POWer:TYPE)

命令格式	<i>[:SOURce]:RADio:ARB:NOISe:POWer:TYPE CARRier CHNO TONO TOPO</i> <i>[:SOURce]:RADio:ARB:NOISe:POWer:TYPE?</i>
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的功率控制模式。
参数类型	枚举
参数范围	CARRier CHNO TONO TOPO
返回值	枚举
默认值	TOPO
对应菜单	<input type="checkbox"/> IQ > ARB > ARB 设置 > 实时 AWGN > 功率控制模式
示例	<i>:RADio:ARB:NOISe:POWer:TYPE CARRier</i> <i>:RADio:ARB:NOISe:POWer:TYPE?</i> 返回值: <i>CARRier\n</i>

#### 3.4.14.17 实时 AWGN 总功率 ([:SOURce]:RADio:ARB:NOISe:POWer:TOTAL)

命令格式	<i>[:SOURce]:RADio:ARB:NOISe:POWer:TOTAL &lt;power&gt;</i> <i>[:SOURce]:RADio:ARB:NOISe:POWer:TOTAL?</i>
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的总功率值。
参数类型	浮点型, 单位: dBm
参数范围	-140 ~ 10
返回值	浮点型, 单位: dBm
默认值	0
对应菜单	<input type="checkbox"/> IQ > ARB > ARB 设置 > 实时 AWGN > 总功率
示例	<i>:RADio:ARB:NOISe:POWer:TOTAL 0 dBm</i> <i>:RADio:ARB:NOISe:POWer:TOTAL?</i> 返回值: <i>0\n</i>

## 3.4.14.18 实时 AWGN 载波功率 ([:SOURce]:RADio:ARB:NOISe:POWer:CARRier)

命令格式	<code>[:SOURce]:RADio:ARB:NOISe:POWer:CARRier &lt;power&gt;</code> <code>[:SOURce]:RADio:ARB:NOISe:POWer:CARRier?</code>
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的载波功率值。
参数类型	浮点型, 单位: dBm
参数范围	和实时 AWGN 的总功率值相关。
返回值	浮点型, 单位: dBm
默认值	-3.27
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 载波功率
示例	<code>:RADio:ARB:NOISe:POWer:CARRier 0 dBm</code> <code>:RADio:ARB:NOISe:POWer:CARRier?</code> 返回值: <code>0\n</code>

## 3.4.14.19 实时 AWGN 总噪声功率 ([:SOURce]:RADio:ARB:NOISe:POWer:TONoise)

命令格式	<code>[:SOURce]:RADio:ARB:NOISe:POWer:TONoise &lt;power&gt;</code> <code>[:SOURce]:RADio:ARB:NOISe:POWer:TONoise?</code>
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的总噪声功率值。
参数类型	浮点型, 单位: dBm
参数范围	和实时 AWGN 的总功率值相关。
返回值	浮点型, 单位: dBm
默认值	-3.27
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 总噪声功率
示例	<code>:RADio:ARB:NOISe:POWer:TONoise 0 dBm</code> <code>:RADio:ARB:NOISe:POWer:TONoise?</code> 返回值: <code>0\n</code>

## 3.4.14.20 实时 AWGN 通道噪声功率 ([:SOURce]:RADio:ARB:NOISe:POWer:CHNOise)

命令格式	<code>[:SOURce]:RADio:ARB:NOISe:POWer:CHNOise &lt;power&gt;</code> <code>[:SOURce]:RADio:ARB:NOISe:POWer:CHNOise?</code>
功能描述	此命令设置/查询 ARB 实时 AWGN 的通道噪声功率值。
参数类型	浮点型, 单位: dBm
参数范围	和实时 AWGN 的总功率值相关。
返回值	浮点型, 单位: dBm

默认值	-3.27
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 通道噪声功率
示例	<code>:RADio:ARB:NOISe:POWer:CHNOise 0 dBm</code> <code>:RADio:ARB:NOISe:POWer:CHNOise?</code> 返回值: <code>0n</code>

#### 3.4.14.21 实时 AWGN 载波噪声比格式 ([:SOURce]:RADio:ARB:NOISe:CN:FORMat)

命令格式	<code>[:SOURce]:RADio:ARB:NOISe:CN:FORMat CARRier BIT</code> <code>[:SOURce]:RADio:ARB:NOISe:CN:FORMat?</code>
功能描述	此命令设置/查询 ARB 实时 AWGN 的载波噪声比格式。
参数类型	枚举
参数范围	CARRier BIT
返回值	枚举
默认值	CARRier
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 载波噪声比格式
示例	<code>:RADio:ARB:NOISe:CN:FORMat BIT</code> <code>:RADio:ARB:NOISe:CN:FORMat?</code> 返回值: <code>BITn</code>

#### 3.4.14.22 实时 AWGN 载波噪声比 ([:SOURce]:RADio:ARB:NOISe:CN)

命令格式	<code>[:SOURce]:RADio:ARB:NOISe:CN &lt;val&gt;</code> <code>[:SOURce]:RADio:ARB:NOISe:CN?</code>
功能描述	此命令设置/查询 ARB 实时 AWGN 的载波噪声比。
参数类型	浮点型, 单位: dB
参数范围	-100 ~ 100
返回值	浮点型, 单位: dB
默认值	0
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 载波噪声比
示例	<code>:RADio:ARB:NOISe:CN -5</code> <code>:RADio:ARB:NOISe:CN?</code> 返回值: <code>-5n</code>

## 3.4.14.23 实时 AWGN 比特信噪比 ([:SOURce]:RADio:ARB:NOISe:CBNO)

命令格式	[:SOURce]:RADio:ARB:NOISe:CBNO <val> [:SOURce]:RADio:ARB:NOISe:CBNO?
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的比特信噪比。
参数类型	浮点型, 单位: dB
参数范围	和载波信噪比及载波比特率相关。
返回值	浮点型, 单位: dB
默认值	0
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > Eb/No
示例	:RADio:ARB:NOISe:CBNO -5 :RADio:ARB:NOISe:CBNO? 返回值: -5n

## 3.4.14.24 实时 AWGN 载波比特率 ([:SOURce]:RADio:ARB:NOISe:BRATe)

命令格式	[:SOURce]:RADio:ARB:NOISe:BRATe <rate> [:SOURce]:RADio:ARB:NOISe:BRATe?
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的载波比特率。
参数类型	浮点型, 单位: Sps
参数范围	1 ~ 10*载波带宽
返回值	浮点型, 单位: Sps
默认值	1
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 载波比特率
示例	:RADio:ARB:NOISe:BRATe 5 :RADio:ARB:NOISe:BRATe? 返回值: 5n

## 3.4.14.25 实时 AWGN 载波带宽 ([:SOURce]:RADio:ARB:NOISe:CBWidth)

命令格式	[:SOURce]:RADio:ARB:NOISe:CBWidth <bandwidth> [:SOURce]:RADio:ARB:NOISe:CBWidth?
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的载波带宽。
参数类型	浮点型, 单位: Hz
参数范围	1 Hz ~ 120 MHz
返回值	浮点型, 单位: Hz

默认值	1 Hz
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 载波带宽
示例	<code>:RADio:ARB:NOISe:CBWidth 5000000</code> <code>:RADio:ARB:NOISe:CBWidth?</code> 返回值: <code>5000000\n</code>

#### 3.4.14.26 实时 AWGN 白噪声带宽 ([:SOURce]:RADio:ARB:NOISe:NBWidth)

命令格式	<code>[:SOURce]:RADio:ARB:NOISe:NBWidth &lt;bandwidth&gt;</code> <code>[:SOURce]:RADio:ARB:NOISe:NBWidth?</code>
功能描述	此命令设置/查询 ARB 调制实时 AWGN 的白噪声带宽。
参数类型	浮点型, 单位: Hz
参数范围	载波带宽 ~ 120 MHz
返回值	浮点型, 单位: Hz
默认值	1 Hz
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 实时 AWGN > 白噪声带宽
示例	<code>:RADio:ARB:NOISe:NBWidth 5000000</code> <code>:RADio:ARB:NOISe:NBWidth?</code> 返回值: <code>5000000\n</code>

#### 3.4.14.27 ARB 调制滤波器类型 ([:SOURce]:IQ:DUALarb:FILTer:TYPE)

命令格式	<code>[:SOURce]:IQ:DUALarb:FILTer:TYPE &lt;type&gt;</code> <code>[:SOURce]:IQ:DUALarb:FILTer:TYPE?</code>
功能描述	此命令设置/查询 ARB 调制的滤波器类型。
参数类型	枚举
参数范围	NONE RAISecosine ROOTcosine GAUSSian HSINe
返回值	枚举
默认值	NONE
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制滤波器 > 滤波器类型
示例	<code>:IQ:DUALarb:FILTer:TYPE GAUSSian</code> <code>:IQ:DUALarb:FILTer:TYPE?</code> 返回值: <code>GAUSSian\n</code>

## 3.4.14.28 ARB 滤波器系数 ([:SOURce]:IQ:DUALarb:FILTer:ALPHa)

命令格式	[:SOURce]:IQ:DUALarb:FILTer:ALPHa <val> [:SOURce]:IQ:DUALarb:FILTer:ALPHa?
功能描述	此命令设置/查询 ARB 调制的升余弦或根升余弦滤波器的 Alpha 值，或高斯滤波器的 BT 值。
参数类型	浮点型
参数范围	0.010 ~ 1.000
返回值	浮点型
默认值	0.500
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制滤波器 > 滤波器 Alpha/ 滤波器 BT
示例	:IQ:DUALarb:FILTer:ALPHa 0.22 :IQ:DUALarb:FILTer:ALPHa? 返回值: 0.22\n

## 3.4.14.29 ARB 滤波器长度 ([:SOURce]:IQ:DUALarb:FILTer:LENGTh)

命令格式	[:SOURce]:IQ:DUALarb:FILTer:LENGTh <len> [:SOURce]:IQ:DUALarb:FILTer:LENGTh?
功能描述	此命令设置/查询 ARB 调制的滤波器长度。
参数类型	整型
参数范围	1 ~ 512
返回值	整型
默认值	32
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制滤波器 > 滤波器长度
示例	:IQ:DUALarb:FILTer:LENGTh 64 :IQ:DUALarb:FILTer:LENGTh? 返回值: 64\n

## 3.4.14.30 ARB 滤波器过采样倍数 ([:SOURce]:IQ:DUALarb:OSAMple)

命令格式	[:SOURce]:IQ:DUALarb:OSAMple <val> [:SOURce]:IQ:DUALarb:OSAMple?
功能描述	此命令设置/查询 ARB 调制的过采样倍数。
参数类型	整型
参数范围	2 ~ 32

返回值	整型
默认值	2
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制滤波器 > 过采样
示例	<code>:IQ:DUALarb:OSAMple 4</code> <code>:IQ:DUALarb:OSAMple?</code> 返回值: <code>4\n</code>

#### 3.4.14.31 ARB 滤波器更新 ([:SOURce]:IQ:DUALarb:FILTer:UPDAte)

命令格式	<code>[:SOURce]:IQ:DUALarb:FILTer:UPDAte</code>
功能描述	此命令更新 ARB 调制滤波器的设置。
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 调制滤波器 > 更新
示例	<code>:IQ:DUALarb:FILTer:UPDAte</code>

#### 3.4.14.32 基带频率偏置状态 ([:SOURce]:RADio:ARB:OFFSet:STATe)

命令格式	<code>[:SOURce]:RADio:ARB:OFFSet:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ARB:OFFSet:STATe?</code>
功能描述	此命令设置/查询 ARB 调制基带频率偏置的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 基带频率偏置
示例	<code>:RADio:ARB:OFFSet:STATe 1</code> <code>:RADio:ARB:OFFSet:STATe?</code> 返回值: <code>1\n</code>

#### 3.4.14.33 基带偏置频率 ([:SOURce]:RADio:ARB:OFFSet:FREQuency)

命令格式	<code>[:SOURce]:RADio:ARB:OFFSet:FREQuency &lt;freq&gt;</code> <code>[:SOURce]:RADio:ARB:OFFSet:FREQuency?</code>
功能描述	此命令设置/查询 ARB 调制的基带偏置频率。
参数类型	浮点型, 单位: Hz
参数范围	0 Hz ~ 60 MHz

返回值	浮点型, 单位: Hz
默认值	0 Hz
对应菜单	<b>IQ</b> > ARB > ARB 设置 > 偏置频率
示例	<code>:RADio:ARB:OFFSet:FREQuency -1000000</code> <code>:RADio:ARB:OFFSet:FREQuency?</code> 返回值: <code>-1000000n</code>

#### 3.4.14.34 选择活动标识点 ([:SOURce]:IQ:DUALarb:MARKer)

命令格式	<code>[:SOURce]:IQ:DUALarb:MARKer &lt;marker&gt;</code> <code>[:SOURce]:IQ:DUALarb:MARKer?</code>
功能描述	此命令选择/查询要编辑的活动标记。
参数类型	整型
参数范围	1 ~ 4
对应菜单	<b>IQ</b> > ARB > 标识设置 > 标识编号
示例	<code>:IQ:DUALarb:MARKer 2</code> <code>:IQ:DUALarb:MARKer?</code> 返回值: <code>2n</code>

#### 3.4.14.35 标识点设置 ([:SOURce]:RADio:ARB:MARKer:[SET])

命令格式	<code>[:SOURce]:RADio:ARB:MARKer:[SET]</code> <code>&lt;"segment"&gt;,&lt;index&gt;,&lt;first_point&gt;,&lt;last_point&gt;,&lt;skip_count&gt;</code>
功能描述	此命令设置波形段的标识点。
参数类型	segment: 字符串, 波形段名称, index: 整型, 标记段的序号, first_point: 整型, 标记波形段的第一个标识点, last_point: 整型, 标记波形段的最后一个标识点, skip_count: 整型, 标记波形段的标识点间隔。
参数范围	segment: ARB 段列表中的波形段, index: 1 ~ 1024, first_point: 1 ~ 波形段的点数, last_point: <first_point> ~ 波形段的点数, skip_count: 0 ~ (<last_point> - <first_point>)
对应菜单	<b>IQ</b> > ARB > 标识设置 > 标识点设置
示例	<code>:RADio:ARB:MARKer:CLEar:ALL "RAMP_WAVE",2</code> <code>:IQ:DUALarb:MARKer 2</code>

```
:RADio:ARB:MARKer "RAMP_WAVE",1,10,20,5
:RADio:ARB:MARKer "RAMP_WAVE",2,100,150,0
```

此时 RAMP\_WAVE 的标识为：10,20,5\n100,150,0

注：此 SCPI 命令不包含回车符或换行符。上述命令格式中包含的是空格。

#### 3.4.14.36 清空标识点 ([:SOURce]:RADio:ARB:MARKer:CLEAr:ALL)

命令格式	[:SOURce]:RADio:ARB:MARKer:CLEAr:ALL <"segment">,<marker>
功能描述	此命令清空波形段的指定标识点的标识。
参数类型	segment: 字符串, 波形段名称, marker: 整型, 标识点。
参数范围	segment: ARB 段列表中的波形段, marker: 1 ~ 4
对应菜单	 > ARB > 标识设置 > 标识点设置 > 清空
示例	:RADio:ARB:MARKer:CLEAr:ALL "SINE_WAVE",1

#### 3.4.14.37 标识极性 ([:SOURce]:RADio:ARB:MPOLarity:MARKer1|2|3|4)

命令格式	[:SOURce]:RADio:ARB:MPOLarity:MARKer1 2 3 4 NEG POS [:SOURce]:RADio:ARB:MPOLarity:MARKer1 2 3 4?
功能描述	此命令设置/查询指定标识点的极性。
参数类型	枚举
参数范围	NEG POS
返回值	枚举
默认值	NEG
对应菜单	 > ARB > 标识设置 > 标识极性
示例	:RADio:ARB:MPOLarity:MARKer1 NEG :RADio:ARB:MPOLarity:MARKer1? 返回值: NEG\n

#### 3.4.14.38 输出标识 ([:SOURce]:RADio:ARB:MARKer:OUTPut)

命令格式	[:SOURce]:RADio:ARB:MARKer:OUTPut None Marker1 Marker2 Marker3 Marker4 [:SOURce]:RADio:ARB:MARKer:OUTPut?
功能描述	此命令设置/查询波形段的输出标识点。

参数类型	枚举
参数范围	None Marker1 Marker2 Marker3 Marker4
返回值	枚举
默认值	Marker1
对应菜单	<b>IQ</b> > ARB > 标识设置 > 输出标识
示例	:RADio:ARB:MARKer:OUTPut Marker2 :RADio:ARB:MARKer:OUTPut? 返回值: Marker2\n

注：此 SCPI 命令不包含回车符或换行符。上述命令格式中包含的是空格。

#### 3.4.14.39 标识延迟 ([:SOURce]:IQ:DUALarb:MARKer:DELAy)

命令格式	[:SOURce]:IQ:DUALarb:MARKer:DELAy <time> [:SOURce]:IQ:DUALarb:MARKer:DELAy?
功能描述	此命令设置/查询波形段的标识延迟时间。
参数类型	浮点型，单位：ns, us, ms 或 s，默认为 s
参数范围	-4 us ~ 860 us
返回值	浮点型，单位：s
默认值	0 ns
对应菜单	<b>IQ</b> > ARB > 标识设置 > 标识延迟
示例	:IQ:DUALarb:MARKer:DELAy 20 us :IQ:DUALarb:MARKer:DELAy? 返回值: 2e-05\n

#### 3.4.14.40 脉冲/RF 消隐 ([:SOURce]:RADio:ARB:MDEStination:PULSe)

命令格式	[:SOURce]:RADio:ARB:MDEStination:PULSe None Marker1 Marker2 Marker3 Marker4 [:SOURce]:RADio:ARB:MDEStination:PULSe?
功能描述	此命令设置/查询波形段的标记脉冲/RF 消隐功能。
参数类型	枚举
参数范围	None Marker1 Marker2 Marker3 Marker4
返回值	枚举
默认值	None
对应菜单	<b>IQ</b> > ARB > 标识设置 > 脉冲/RF 消隐

**示例** `:RADio:ARB:MDEStination:PULSe Marker2`  
`:RADio:ARB:MDEStination:PULSe?`  
 返回值:  
`Marker2\n`

注：此 SCPI 命令不包含回车符或换行符。上述命令格式中包含的是空格。

#### 3.4.14.41 波形削减 ([:SOURce]:RADio:ARB:CLIPping)

<b>命令格式</b>	<code>[:SOURce]:RADio:ARB:CLIPping &lt;"segment"&gt;,IJQ IORQ,&lt;val&gt;[,&lt;val&gt;]</code>
<b>功能描述</b>	此命令对波形段进行削减。
<b>参数类型</b>	segment: 字符串, 波形段名称, IJQ IORQ: 枚举, 削减类型,  I+jQ 或 I , Q , val: 浮点型, 削减系数。
<b>参数范围</b>	segment: ARB 段列表中的波形段, IJQ IORQ: IJQ 表示 I+jQ , IORQ 表示 I , Q , val: 0.01 ~ 1。
<b>对应菜单</b>	<code>IQ</code> > ARB > 波形设置 > 波形段选择 & 削减类型 /  I+jQ 削减至 /  I 削减至 /  Q 削减至
<b>示例</b>	<code>:RADio:ARB:CLIPping "SINE_WAVE",IJQ,0.75</code> <code>:RADio:ARB:CLIPping "RAMP_WAVE",IORQ,0.75,0.8</code>

#### 3.4.14.42 波形缩放 ([:SOURce]:RADio:ARB:SCaling)

<b>命令格式</b>	<code>[:SOURce]:RADio:ARB:SCaling &lt;"segment"&gt;,&lt;val&gt;</code>
<b>功能描述</b>	此命令对波形段进行缩放。
<b>参数类型</b>	segment: 字符串, 波形段名称, val: 浮点型, 缩放系数。
<b>参数范围</b>	segment: ARB 段列表中的波形段, val: 0.01 ~ 1。
<b>对应菜单</b>	<code>IQ</code> > ARB > 波形设置 > 波形段选择 & 缩放
<b>示例</b>	<code>:RADio:ARB:Scaling "RAMP_WAVE",0.75</code>

#### 3.4.14.43 ARB 触发类型 ([:SOURce]:IQ:DUALarb:TRIGger:TYPE)

<b>命令格式</b>	<code>[:SOURce]:IQ:DUALarb:TRIGger:TYPE CONTInous  SINGLE  SADVance  GATE</code> <code>[:SOURce]:IQ:DUALarb:TRIGger:TYPE?</code>
-------------	---

功能描述	此命令设置/查询 ARB 的触发类型。
参数类型	枚举
参数范围	CONTInous: 连续触发, SINGle: 单次触发, SADVance: 段提前触发, GATE: 门选通触发。
返回值	枚举
默认值	CONTInous
对应菜单	<b>IQ</b> > ARB > 触发 > 触发类型
示例	<i>:IQ:DUALarb:TRIGger:TYPE SINGLE</i> <i>:IQ:DUALarb:TRIGger:TYPE?</i> 返回值: <i>SINGLE</i> \n

#### 3.4.14.44 ARB 连续触发模式 ([:SOURce]:IQ:DUALarb:TRIGger:CONTInous)

命令格式	<i>[:SOURce]:IQ:DUALarb:TRIGger:CONTInous FREErun   RUNIgnored   RUNRestart</i> <i>[:SOURce]:IQ:DUALarb:TRIGger:CONTInous?</i>
功能描述	此命令设置/查询 ARB 连续触发的触发模式。
参数类型	枚举
参数范围	FREErun: 自由播放, RUNIgnored: 触发&播放, RUNRestart: 复位&播放。
返回值	枚举
默认值	FREErun
对应菜单	<b>IQ</b> > ARB > 触发 > 触发类型 (连续) > 连续模式
示例	<i>:IQ:DUALarb:TRIGger:CONTInous RUNIgnored</i> <i>:IQ:DUALarb:TRIGger:CONTInous?</i> 返回值: <i>RUNIgnored</i> \n

#### 3.4.14.45 ARB 单次触发模式 ([:SOURce]:IQ:DUALarb:TRIGger:SINGle)

命令格式	<i>[:SOURce]:IQ:DUALarb:TRIGger:SINGle NOREtrigger   BUFFeredtrig   RESTartontrig</i> <i>[:SOURce]:IQ:DUALarb:TRIGger:SINGle?</i>
功能描述	此命令设置/查询 ARB 单次触发的触发模式。

参数类型	枚举
参数范围	NOREtrigger: 忽略重触发, BUFFeredtrig: 触发缓冲, REStartontrig: 触发重启。
返回值	枚举
默认值	NOREtrigger
对应菜单	<b>IQ</b> > ARB > 触发 > 触发类型 (单次) > 单次模式
示例	<i>:IQ:DUALarb:TRIGger:SINGLE BUFFeredtrig</i> <i>:IQ:DUALarb:TRIGger:SINGLE?</i> 返回值: <i>BUFFeredtrig</i>

#### 3.4.14.46 ARB 段提前触发模式 ([:SOURce]:IQ:DUALarb:TRIGger:SEGMENT)

命令格式	<i>[:SOURce]:IQ:DUALarb:TRIGger:SEGMENT SINGLE   CONTInous</i> <i>[:SOURce]:IQ:DUALarb:TRIGger:SEGMENT?</i>
功能描述	此命令设置/查询 ARB 段提前触发的触发模式。
参数类型	枚举
参数范围	SINGLE   CONTInous
返回值	枚举
默认值	CONTInous
对应菜单	<b>IQ</b> > ARB > 触发 > 触发类型 (段提前) > 段模式
示例	<i>:IQ:DUALarb:TRIGger:SEGMENT SINGLE</i> <i>:IQ:DUALarb:TRIGger:SEGMENT?</i> 返回值: <i>SINGLE</i>

#### 3.4.14.47 ARB 门选通触发模式 ([:SOURce]:IQ:DUALarb:TRIGger:GATE)

命令格式	<i>[:SOURce]:IQ:DUALarb:TRIGger:GATE LOW   HIGH</i> <i>[:SOURce]:IQ:DUALarb:TRIGger:GATE?</i>
功能描述	此命令设置/查询 ARB 门选通触发的触发模式。
参数类型	枚举
参数范围	LOW: 低电平有效, HIGH: 高电平有效。
返回值	枚举
默认值	HIGH

对应菜单	<b>IQ</b> > ARB > 触发 > 触发类型 (门选通) > 门选通模式
示例	<code>:IQ:DUALarb:TRIGger:GATE LOW</code> <code>:IQ:DUALarb:TRIGger:GATE?</code> 返回值: <code>LOWn</code>

#### 3.4.14.48 ARB 触发源 ([:SOURce]:IQ:DUALarb:TRIGger:SOURce)

命令格式	<code>[:SOURce]:IQ:DUALarb:TRIGger:SOURce KEY  BUS  EXT</code> <code>[:SOURce]:IQ:DUALarb:TRIGger:SOURce?</code>
功能描述	此命令设置/查询 ARB 触发的触发源。
参数类型	枚举
参数范围	KEY: 按键触发, BUS: 总线触发, EXT: 外部触发。
返回值	枚举
默认值	KEY
对应菜单	<b>IQ</b> > ARB > 触发 > 触发源
示例	<code>:IQ:DUALarb:TRIGger:SOURce EXT</code> <code>:IQ:DUALarb:TRIGger:SOURce?</code> 返回值: <code>EXTn</code>

#### 3.4.14.49 ARB 总线触发 ([:SOURce]:IQ:DUALarb:TRG)

命令格式	<code>[:SOURce]:IQ:DUALarb:TRG</code>
功能描述	当 ARB 触发源为总线时, 执行此命令将发送一个 ARB 触发信号。
对应菜单	无
示例	<code>:IQ:DUALarb:TRG</code>

#### 3.4.14.50 ARB 外触发极性 ([:SOURce]:IQ:DUALarb:TRIGger:POL)

命令格式	<code>[:SOURce]:IQ:DUALarb:TRIGger:POL POS   NEG</code> <code>[:SOURce]:IQ:DUALarb:TRIGger:POL?</code>
功能描述	此命令设置/查询 ARB 外部触发的触发极性。
参数类型	枚举
参数范围	POS: 正相, NEG: 反相。

返回值	枚举
默认值	POS
对应菜单	<b>IQ</b> > ARB > 触发 > 触发源（外部） > 外触发极性
示例	<code>:IQ:DUALarb:TRIGger:POL NEG</code> <code>:IQ:DUALarb:TRIGger:POL?</code> 返回值: <code>NEG\n</code>

#### 3.4.14.51 ARB 外触发延迟类型 ([:SOURce]:IQ:DUALarb:TRIGger:DELAy:TYPE)

命令格式	<code>[:SOURce]:IQ:DUALarb:TRIGger:DELAy:TYPE OFF   TIME   SAMPlE</code> <code>[:SOURce]:IQ:DUALarb:TRIGger:DELAy:TYPE?</code>
功能描述	此命令设置/查询 ARB 外触发的触发延迟类型。
参数类型	枚举
参数范围	OFF: 关闭, TIME: 时间, SAMPlE: 采样点。
返回值	枚举
默认值	OFF
对应菜单	<b>IQ</b> > ARB > 触发 > 触发源（外部） > 延迟类型
示例	<code>:IQ:DUALarb:TRIGger:DELAy:TYPE SAMPlE</code> <code>:IQ:DUALarb:TRIGger:DELAy:TYPE?</code> 返回值: <code>SAMPlE\n</code>

#### 3.4.14.52 ARB 外触发延迟时间 ([:SOURce]:IQ:DUALarb:TRIGger:DELAy:Time)

命令格式	<code>[:SOURce]:IQ:DUALarb:TRIGger:DELAy:Time &lt;value&gt;</code> <code>[:SOURce]:IQ:DUALarb:TRIGger:DELAy:Time?</code>
功能描述	此命令设置/查询 ARB 外触发的触发延迟时间。
参数类型	浮点型，单位：s
参数范围	0 ~ 40s
返回值	浮点型，单位：s
默认值	0
对应菜单	<b>IQ</b> > ARB > 触发 > 触发源（外部） > 延迟类型（时间） > 延迟时间
示例	<code>:IQ:DUALarb:TRIGger:DELAy:Time 2</code> <code>:IQ:DUALarb:TRIGger:DELAy:Time?</code> 返回值: <code>2\n</code>

## 3.4.14.53 ARB 外触发延迟采样点数 ([:SOURce]:IQ:DUALarb:TRIGger:DELAy:SAMPlE)

命令格式	[:SOURce]:IQ:DUALarb:TRIGger:DELAy:SAMPlE <value> [:SOURce]:IQ:DUALarb:TRIGger:DELAy:SAMPlE?
功能描述	此命令设置/查询 ARB 外触发的触发延迟采样点数。
参数类型	整型
参数范围	0 ~ 1000000000
返回值	整型
默认值	0
对应菜单	<input type="checkbox"/> IQ > ARB > 触发 > 触发源 (外部) > 延迟类型 (采样点) > 延迟采样点数
示例	<i>:IQ:DUALarb:TRIGger:DELAy:SAMPlE 500</i> <i>:IQ:DUALarb:TRIGger:DELAy:SAMPlE?</i> 返回值: <i>500\n</i>

## 3.4.14.54 查询波形头文件内容 ([:SOURce]:IQ:DUALarb:HEADer:INFO?)

命令格式	[:SOURce]:IQ:DUALarb:HEADer:INFO?
功能描述	此命令查询波形的头文件内容。在查询头文件内容时，需要先选择播放的波形。
返回值	字符串
默认值	无
对应菜单	<input type="checkbox"/> IQ > ARB > 波形头文件
示例	<i>:RADio:ARB:WAVeform "WFM:SINE_WAVE"</i> <i>:IQ:DUALarb:HEADer:INFO?</i> 返回值: <i>discript= sampling rate=Unspecified marker1 polary=Unspecified marker2 polary=Unspecified marker3 polary=Unspecified marker4 polary=Unspecified rf marker=Unspecified output marker=Unspecified atten type=Unspecified atten value=Unspecified noise state=Unspecified noise output=Unspecified noise power control=Unspecified noise total power=Unspecified noise carrier power=Unspecified noise noise power=Unspecified channel noise power=Unspecified</i>

```

carrier to noise ratio format=Unspecified
carrier to noise ratio=Unspecified
bit to noise ratio=Unspecified
carrier bit ratio=Unspecified
carrier bandwidth=Unspecified
noise bandwidth=Unspecified
baseband offset state=Unspecified
baseband offset freq=Unspecified\n\n

```

注：在此 SCPI 示例中，返回值中的各字段由换行符分隔。

#### 3.4.14.55 清空头文件 ([:SOURce]:IQ:DUALarb:HEADer:CLEAR)

命令格式	[:SOURce]:IQ:DUALarb:HEADer:CLEAR
功能描述	此命令清空波形段或波形序列的头文件内容。
对应菜单	<input type="checkbox"/> IQ > ARB > 波形头文件 > 清空头文件
示例	:IQ:DUALarb:HEADer:CLEAR

#### 3.4.14.56 保存至头文件 ([:SOURce]:IQ:DUALarb:HEADer:STORE)

命令格式	[:SOURce]:IQ:DUALarb:HEADer:STORE
功能描述	此命令保存波形段或波形序列的头文件内容。
对应菜单	<input type="checkbox"/> IQ > ARB > 波形头文件 > 保存至头文件
示例	:IQ:DUALarb:HEADer:STORE

#### 3.4.14.57 头文件描述 ([:SOURce]:IQ:DUALarb:HEADer:DESCript)

命令格式	[:SOURce]:IQ:DUALarb:HEADer:DESCript <"string"> [:SOURce]:IQ:DUALarb:HEADer:DESCript?
功能描述	此命令设置/查询波形段或波形序列的头文件描述。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
返回值	字符串
默认值	无描述
对应菜单	<input type="checkbox"/> IQ > ARB > 波形头文件 > 说明
示例	:IQ:DUALarb:HEADer:DESCript "INFO" :IQ:DUALarb:HEADer:DESCript? 返回值: INFO\n

## 3.4.14.58 多载波波形名称 ([:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:NAME)

命令格式	<code>[:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:NAME &lt;"waveform"&gt;</code> <code>[:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:NAME?</code>
功能描述	此命令设置/查询 ARB 多载波的波形名称。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
返回值	字符串
默认值	MULTICARRIER
对应菜单	<b>IQ</b> > ARB > 多载波 > 波形名称
示例	<code>:RADio:DMODulation:ARB:SETup:MCARrier:NAME "MULTI_TEST"</code> <code>:RADio:DMODulation:ARB:SETup:MCARrier:NAME?</code> 返回值: <code>MULTI_TESTn</code>

## 3.4.14.59 创建并加载多载波 ([:SOURce]:RADio:DMODulation:ARB:SETup)

命令格式	<code>[:SOURce]:RADio:DMODulation:ARB:SETup</code>
功能描述	此命令可以根据当前设置创建多载波，然后将该多载波加载到 ARB 段中并选择播放该多载波。
对应菜单	<b>IQ</b> > ARB > 多载波 > 创建并加载
示例	<code>:RADio:DMODulation:ARB:SETup</code>

## 3.4.14.60 多载波辅助设置 ([:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier)

命令格式	<code>[:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier</code> <code>&lt;"segment"&gt;,&lt;num&gt;,&lt;freq_space&gt;</code> <code>[:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier?</code>
功能描述	此命令使用指定数量的相同子载波和频率间隔构建一个载波表。 查询命令返回多载波的子载波名称、子载波数量和频率间隔。
参数类型	segment: 字符串, num: 整型, 载波数, freq_space: 双精度浮点型, 载波间的频率间隔, 单位: Hz。
参数范围	segment: ARB 波形段列表中的波形段, num: 2 ~ 100, freq_space: 0 ~ 最大采样带宽/(载波数-1)。
返回值	字符串,

	格式: <carrier>,<num carriers>,<freq spacing>
默认值	*NONE,2,1000000\n
对应菜单	<input type="checkbox"/> IQ > ARB > 多载波 > 载波列表 > 辅助设置
示例	:RADio:DMODulation:ARB:SETup:MCARrier "SINE_WAVE",3,2e6 :RADio:DMODulation:ARB:SETup:MCARrier? 返回值: SINE_WAVE,3,2000000\n

注: 此 SCPI 命令不包含回车符或换行符。上述命令格式中包含的是空格。

#### 3.4.14.61 编辑多载波列表 ([:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:TABLE)

命令格式	[:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:TABLE INIT APPend <carrier_num>,<"segment">,<freq_offset>,<power>,<phase> [:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:TABLE? <carrier_num>
功能描述	此命令编辑 ARB 多载波列表的指定行。 INIT: 该选项删除所有的多载波, 然后写入一行指定的载波, APPend: 该选项在多载波列表后新增一行指定的载波, carrier_num: 该选项修改多载波列表的指定行。  查询命令查询 ARB 多载波列表的指定行。
参数类型	INIT APPend: 枚举, carrier_num: 整型, 载波行号, segment: 字符串, ARB 波形段列表中的波形段, freq_offset: 双精度浮点型, 载波的偏置频率, 单位: Hz, power: 双精度浮点型, 载波增益, 单位: dB, phase: 双精度浮点型, 载波相位, 单位: ° (度)。
参数范围	carrier_num: 1 ~ 载波总数, segment: ARB 波形段列表中的波形段, freq_offset: -60 MHz ~ 60 MHz, power: -40 ~ 0, phase: -360 ~ 360。
返回值	字符串, 格式: <carrier>,<freq_offset>,<power>,<phase>,<sample_clock>,<sample_points>
默认值	SINE_WAVE,0,0,0,2e+06,200\n
对应菜单	<input type="checkbox"/> IQ > ARB > 多载波 > 载波列表

**示例**

```
:RADio:DMODulation:ARB:SETup:MCARrier:TABLE
INIT,"RAMP_WAVE",1000000,-10,20

:RADio:DMODulation:ARB:SETup:MCARrier:TABLE
APPend,"RAMP_WAVE",2000000,-5,90

:RADio:DMODulation:ARB:SETup:MCARrier:
TABLE 2,"TEST",-5000000,-2,-30

:RADio:DMODulation:ARB:SETup:MCARrier:TABLE? 1
返回值:
RAMP_WAVE,1e+06,-10,20,2e+06,200Vn
```

注：此 SCPI 命令不包含回车符或换行符。上述命令格式和示例中包含的是空格。

#### 3.4.14.62 保存多载波列表 ([:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:STORE)

<b>命令格式</b>	[:SOURce]:RADio:DMODulation:ARB:SETup:MCARrier:STORE <"file_name">
<b>功能描述</b>	此命令把多载波列表保存到 ML 文件中。
<b>参数类型</b>	字符串
<b>参数范围</b>	命名规则请参阅用户手册。
<b>对应菜单</b>	<input type="checkbox"/> IQ > ARB > 多载波 > 多载波列表 > 保存
<b>示例</b>	:RADio:DMODulation:ARB:SETup:MCARrier:STORE "Multi_Table.ml"

注：此 SCPI 命令不包含回车符或换行符。上述命令格式中包含的是空格。

#### 3.4.14.63 加载多载波列表 ([:SOURce]:IQ:CARRier:LOAD)

<b>命令格式</b>	[:SOURce]:IQ:CARRier:LOAD <"file_name">
<b>功能描述</b>	此命令从 ML 文件中加载多载波列表。
<b>参数类型</b>	字符串
<b>参数范围</b>	无
<b>对应菜单</b>	<input type="checkbox"/> IQ > ARB > 多载波 > 多载波列表 > 加载
<b>示例</b>	:IQ:CARRier:LOAD "Multi_Table.ml"

#### 3.4.14.64 多载波功率参考 ([:SOURce]:IQ:CARRier:POWER:TYPE)

<b>命令格式</b>	[:SOURce]:IQ:CARRier:POWER:TYPE RMS   PEAK [:SOURce]:IQ:CARRier:POWER:TYPE?
<b>功能描述</b>	此命令设置/查询 ARB 多载波的功率参考类型。

参数类型	枚举
参数范围	RMS   PEAK
返回值	枚举
默认值	PEAK
对应菜单	<b>IQ</b> > ARB > 多载波 > 功率参考
示例	:IQ:CARRier:POWER:TYPE RMS :IQ:CARRier:POWER:TYPE? 返回值: RMS\n

#### 3.4.14.65 多载波信号周期模式 ([:SOURce]:IQ:CARRier:PERIod:MODE)

命令格式	[[:SOURce]:IQ:CARRier:PERIod:MODE LONGest   SHORtest   USER   LCM [:SOURce]:IQ:CARRier:PERIod:MODE?
功能描述	此命令设置/查询 ARB 多载波的信号周期模式。
参数类型	枚举
参数范围	LONGest: 最长文件, SHORtest: 最短文件, USER: 自定义, LCM: LCM。
返回值	枚举
默认值	LCM
对应菜单	<b>IQ</b> > ARB > 多载波 > 信号周期模式
示例	:IQ:CARRier:PERIod:MODE LONGest :IQ:CARRier:PERIod:MODE? 返回值: LONGest\n

#### 3.4.14.66 多载波信号周期 ([:SOURce]:IQ:CARRier:PERIod)

命令格式	[[:SOURce]:IQ:CARRier:PERIod <value> [:SOURce]:IQ:CARRier:PERIod?
功能描述	当多载波信号周期模式为自定义时, 此命令可设置多载波的信号周期。 查询命令返回多载波的信号周期。
参数类型	浮点型, 单位: s。
参数范围	200/多载波采样率 ~ min(10e6/多载波采样率, (35e6 - 子载波总采样点数)/(3 * 多载波采样率))
返回值	浮点型, 单位: s。

默认值	无
对应菜单	<b>IQ</b> > ARB > 多载波 > 信号周期
示例	<code>:IQ:CARRier:PERIod 10e-3</code> <code>:IQ:CARRier:PERIod?</code> 返回值: <code>0.01\n</code>

#### 3.4.14.67 查询多载波采样率 ([:SOURce]:IQ:CARRier:SAMPLerate?)

命令格式	<code>[:SOURce]:IQ:CARRier:SAMPLerate?</code>
功能描述	此命令查询 ARB 多载波的采样率。
返回值	浮点型，单位：Hz。
默认值	无
对应菜单	<b>IQ</b> > ARB > 多载波 > 采样率
示例	<code>:IQ:CARRier:SAMPLerate?</code> 返回值: <code>21000000\n</code>

### 3.4.15 I/Q 设置

#### 3.4.15.1 I/Q 调制状态开关 ([:SOURce]:DM:STATe)

命令格式	<code>[:SOURce]:DM:STATe ON OFF 1 0</code> <code>[:SOURce]:DM:STATe?</code>
功能描述	此命令设置/查询 I/Q 调制器的开关状态。 注：打开 Custom 调制、ARB 调制、Stream 调制、IoT 调制、多音或 AWGN 功能时会自动打开 I/Q 调制器。您还需要打开 IQ 调制总开关以打开调制功能，相关命令请查阅“IQ 调制总开关状态 ([:SOURce]:FUNCTion:DM:STATe)”。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 调制状态
示例	<code>:DM:STATe ON</code> <code>:DM:STATe?</code> 返回值: <code>1\n</code>

## 3.4.15.2 I/Q 源 ([:SOURce]:DM:SOURce)

命令格式	[:SOURce]:DM:SOURce EXTERNAL INTERNAL [:SOURce]:DM:SOURce?
功能描述	此命令选择/查询 I/Q 调制器源。
参数类型	枚举
参数范围	EXTERNAL: 外部调制源, INTERNAL: 内部调制源。
返回值	枚举
默认值	INTERNAL
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 源
示例	:DM:SOURce EXTERNAL :DM:SOURce? 返回值: EXTERNAL\n

## 3.4.15.3 补偿通道 ([:SOURce]:DM:BW:CAL:LINK)

命令格式	[:SOURce]:DM:BW:CAL:LINK RF OUTPUT [:SOURce]:DM:BW:CAL:LINK?
功能描述	此命令设置/查询 I/Q 调制的宽带补偿链路。
参数类型	枚举
参数范围	RF OUTPUT
返回值	枚举
默认值	RF
对应菜单	<b>IQ</b> > I/Q 设置 > 补偿通道
示例	:DM:BW:CAL:LINK OUTPUT :DM:BW:CAL:LINK? 返回值: OUTPUT\n

## 3.4.15.4 I/Q 调节开关 ([:SOURce]:DM:IQADjustment[:STATe])

命令格式	[:SOURce]:DM:IQADjustment[:STATe] ON OFF 1 0 [:SOURce]:DM:IQADjustment[:STATe]?
功能描述	此命令设置/查询 I/Q 调节的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0

返回值	布尔型
默认值	1
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 调节
示例	<code>:DM:IQADjustment ON</code> <code>:DM:IQADjustment?</code> 返回值: <code>1\n</code>

#### 3.4.15.5 增益平衡 ([:SOURce]:DM:IQADjustment:GAIN)

命令格式	<code>[:SOURce]:DM:IQADjustment:GAIN &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:GAIN?</code>
功能描述	此命令设置/查询 I 信号相对于 Q 信号的增益。
参数类型	浮点型, 单位: dB。
参数范围	-4 ~ 4
返回值	浮点型, 单位: dB。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 调节 > 增益平衡
示例	<code>:DM:IQADjustment:GAIN -0.5</code> <code>:DM:IQADjustment:GAIN?</code> 返回值: <code>-0.5\n</code>

#### 3.4.15.6 I 偏置 ([:SOURce]:DM:IQADjustment:IOFFset)

命令格式	<code>[:SOURce]:DM:IQADjustment:IOFFset &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:IOFFset?</code>
功能描述	此命令调整/查询 I 通道偏移值。
参数类型	浮点型, 单位: %。
参数范围	-50 ~ 50
返回值	浮点型, 单位: %。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 调节 > I 偏置
示例	<code>:DM:IQADjustment:IOFFset 1.2</code> <code>:DM:IQADjustment:IOFFset?</code> 返回值: <code>1.2\n</code>

## 3.4.15.7 Q 偏置 ([:SOURce]:DM:IQADjustment:QOFFset)

命令格式	<code>[:SOURce]:DM:IQADjustment:QOFFset &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:QOFFset?</code>
功能描述	此命令调整/查询 Q 通道偏移值。
参数类型	浮点型, 单位: %。
参数范围	-50 ~ 50
返回值	浮点型, 单位: %。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 调节 > Q 偏置
示例	<code>:DM:IQADjustment:QOFFset -0.35</code> <code>:DM:IQADjustment:QOFFset?</code> 返回值: <code>-0.35</code>

## 3.4.15.8 正交相位调节 ([:SOURce]:DM:IQADjustment:QSKew)

命令格式	<code>[:SOURce]:DM:IQADjustment:QSKew &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:QSKew?</code>
功能描述	此命令通过增加或减少 Q 相位角来调整 I 和 Q 矢量之间的相位角 (正交偏斜)。它仅影响射频输出路径。正偏斜使角度从 90 度开始增加, 而负偏斜使角度从 90 度开始减小。当正交偏斜为零时, I 和 Q 矢量之间的相位角为 90 度。
参数类型	浮点型, 单位: ° (度)。
参数范围	-10 ~ 10
返回值	浮点型, 单位: ° (度)。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 调节 > 正交相位调节
示例	<code>:DM:IQAD:QSK 1.52</code> <code>:DM:IQAD:QSK?</code> 返回值: <code>1.52</code>

## 3.4.15.9 I/Q 输出状态 ([:SOURce]:DM:IQADjustment:EXTernal[:STATe])

命令格式	<code>[:SOURce]:DM:IQADjustment:EXTernal[:STATe] ON OFF 1 0</code> <code>[:SOURce]:DM:IQADjustment:EXTernal[:STATe]?</code>
功能描述	此命令设置/查询 I/Q 信号的输出状态。
参数类型	布尔型

参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 输出
示例	<code>:DM:IQADjustment:EXternal 1</code> <code>:DM:IQADjustment:EXternal?</code> 返回值: <code>1\n</code>

#### 3.4.15.10 I/Q 输出衰减 ([:SOURce]:DM:IQADjustment:EXternal:IQATten)

命令格式	<code>[:SOURce]:DM:IQADjustment:EXternal:IQATten &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:EXternal:IQATten?</code>
功能描述	此命令设置 I/Q 输出的衰减值。
参数类型	浮点型, 单位: dB。
参数范围	0 ~ 20
返回值	浮点型, 单位: dB。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 输出 > I/Q 输出衰减
示例	<code>:DM:IQADjustment:EXternal:IQATten 2.13</code> <code>:DM:IQADjustment:EXternal:IQATten?</code> 返回值: <code>2.13\n</code>

#### 3.4.15.11 I/Q 输出增益平衡 ([:SOURce]:DM:IQADjustment:EXternal:GAIN)

命令格式	<code>[:SOURce]:DM:IQADjustment:EXternal:GAIN &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:EXternal:GAIN?</code>
功能描述	此命令设置路由到后面板 I 和 Q 输出连接器的信号的 I/Q 增益比。
参数类型	浮点型, 单位: dB。
参数范围	-4 ~ 4
返回值	浮点型, 单位: dB。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 输出 > I/Q 输出增益平衡
示例	<code>:DM:IQADjustment:EXternal:GAIN -1.31</code> <code>:DM:IQADjustment:EXternal:GAIN?</code> 返回值: <code>-1.31\n</code>

## 3.4.15.12 I 输出偏置 ([:SOURce]:DM:IQADjustment:EXTernal:DIOffset)

命令格式	<code>[:SOURce]:DM:IQADjustment:EXTernal:DIOffset &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:EXTernal:DIOffset?</code>
功能描述	此命令设置路由到 I 输出连接器的同相 (I) 信号的差分偏移电压。
参数类型	浮点型, 单位: mV 或 V, 默认为 V。
参数范围	-3 V ~ 3 V
返回值	浮点型, 单位: V。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 输出 > I 输出偏置
示例	<code>:DM:IQADjustment:EXTernal:DIOffset 0.12</code> <code>:DM:IQADjustment:EXTernal:DIOffset?</code> 返回值: <code>0.12/n</code>

## 3.4.15.13 Q 输出偏置 (:DM:IQADjustment:EXTernal:DQOffset)

命令格式	<code>[:SOURce]:DM:IQADjustment:EXTernal:DQOffset &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:EXTernal:DQOffset?</code>
功能描述	此命令设置路由到 Q 输出连接器的正交相 (Q) 信号的差分偏移电压。
参数类型	浮点型, 单位: mV 或 V, 默认为 V。
参数范围	-3 V ~ 3 V
返回值	浮点型, 单位: V。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 输出 > Q 输出偏置
示例	<code>:DM:IQADjustment:EXTernal:DQOffset -0.12</code> <code>:DM:IQADjustment:EXTernal:DQOffset?</code> 返回值: <code>-0.12/n</code>

## 3.4.15.14 I/Q 共模偏置 ([:SOURce]:DM:IQADjustment:EXTernal:COFFset)

命令格式	<code>[:SOURce]:DM:IQADjustment:EXTernal:COFFset &lt;val&gt;</code> <code>[:SOURce]:DM:IQADjustment:EXTernal:COFFset?</code>
功能描述	此命令设置进入后面板 I 和 Q 输出连接器的同相 (I) 和正交相 (Q) 信号的共模偏移电压。
参数类型	浮点型, 单位: mV 或 V, 默认为 V。
参数范围	-3 V ~ 3 V

返回值	浮点型，单位：V。
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 输出 > I/Q 共模偏置
示例	<code>:DM:IQADjustment:EXternal:COFFset 1 mV</code> <code>:DM:IQADjustment:EXternal:COFFset?</code> 返回值： <code>0.001\n</code>

#### 3.4.15.15 I/Q 交换开关状态 ([:SOURce]:IQ:BW:SWAP)

命令格式	<code>[:SOURce]:IQ:BW:SWAP ON OFF 1 0</code> <code>[:SOURce]:IQ:BW:SWAP?</code>
功能描述	此命令设置/查询 I 和 Q 信号的交换状态。启用后，I 路信号保持不变，Q 路信号将会取反，频谱跟正常模式是镜像。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > I/Q 设置 > I/Q 交换
示例	<code>:IQ:BW:SWAP ON</code> <code>:IQ:BW:SWAP?</code> 返回值： <code>1\n</code>

#### 3.4.16 IoT 调制设置

##### 3.4.16.1 协议类型 ([:SOURce]:RADio:IOT:TYPE)

命令格式	<code>[:SOURce]:RADio:IOT:TYPE ZIGBee   ZWAVe</code> <code>[:SOURce]:RADio:IOT:TYPE?</code>
功能描述	此命令设置/查询 IoT 调制的协议类型。
参数类型	枚举
参数范围	ZIGBee   ZWAVe
返回值	枚举
默认值	ZIGBee
对应菜单	<b>IQ</b> > IoT > 协议类型
示例	<code>:RADio:IOT:TYPE ZWAVe</code> <code>:RADio:IOT:TYPE?</code>

返回值：  
ZWAve\n

### 3.4.16.2 ZigBee 设置

#### 3.4.16.2.1 保存波形 ([:SOURce]:RADio:ZIGBee:SAVE)

命令格式	[:SOURce]:RADio:ZIGBee:SAVE <"file_name">
功能描述	将 ZigBee 调制的波形数据保存到文件。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 保存波形
示例	:RADio:ZIGBee:SAVE "test_zigbee.arb"

#### 3.4.16.2.2 ZigBee 状态开关 ([:SOURce]:RADio:ZIGBee:STATe)

命令格式	[:SOURce]:RADio:ZIGBee:STATe ON OFF 1 0 [:SOURce]:RADio:ZIGBee:STATe?
功能描述	此命令设置/查询 ZigBee 调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > Zigbee 状态
示例	:RADio:ZIGBee:STATe ON :RADio:ZIGBee:STATe? 返回值： 1\n

#### 3.4.16.2.3 过采样率 ([:SOURce]:RADio:ZIGBee:OSAMple:RATIo)

命令格式	[:SOURce]:RADio:ZIGBee:OSAMple:RATIo <val> [:SOURce]:RADio:ZIGBee:OSAMple:RATIo?
功能描述	此命令设置/查询 ZigBee 调制波的过采样率。
参数类型	整型
参数范围	2 ~ 64
返回值	整型
默认值	8

对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 基本设置 > 过采样率
示例	<code>:RADio:ZIGBee:OSAMple:RATIo 10</code> <code>:RADio:ZIGBee:OSAMple:RATIo?</code> 返回值: <code>10\n</code>

#### 3.4.16.2.4 帧数 ([:SOURce]:RADio:ZIGBee:FRAME:NUMBer)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FRAME:NUMBer &lt;val&gt;</code> <code>[:SOURce]:RADio:ZIGBee:FRAME:NUMBer?</code>
功能描述	此命令设置和获取 ZigBee 调制波中包含的帧数。
参数类型	整型
参数范围	1 ~ 2000
返回值	整型
默认值	1
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 基本设置 > 帧数
示例	<code>:RADio:ZIGBee:FRAME:NUMBer 2</code> <code>:RADio:ZIGBee:FRAME:NUMBer?</code> 返回值: <code>2\n</code>

#### 3.4.16.2.5 查询总采样点数 ([:SOURce]:RADio:ZIGBee:TOTAL:SAMPLE:POINts?)

命令格式	<code>[:SOURce]:RADio:ZIGBee:TOTAL:SAMPLE:POINts?</code>
功能描述	此命令查询 ZigBee 调制波的总采样点数。
返回值	整型
默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 基本设置 > 总采样点数
示例	<code>:RADio:ZIGBee:TOTAL:SAMPLE:POINts?</code> 返回值: <code>20608\n</code>

#### 3.4.16.2.6 查询波形长度 ([:SOURce]:RADio:ZIGBee:WAVEform:LENGth?)

命令格式	<code>[:SOURce]:RADio:ZIGBee:WAVEform:LENGth?</code>
功能描述	此命令查询 ZigBee 调制波的波形长度。
返回值	浮点型, 单位: s。
默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 基本设置 > 波形长度

**示例** `:RADio:ZIGBee:WAVEform:LENGth?`  
返回值:  
`0.00644\n`

#### 3.4.16.2.7 调制方式 ([:SOURce]:RADio:ZIGBee:PHY:SCHEme)

**命令格式** `[:SOURce]:RADio:ZIGBee:PHY:SCHEme OQPSK| BPSK`  
`[:SOURce]:RADio:ZIGBee:PHY:SCHEme?`

**功能描述** 此命令设置/查询 ZigBee 调制的调制方式。

**参数类型** 枚举

**参数范围** OQPSK| BPSK

**返回值** 枚举

**默认值** OQPSK

**对应菜单** `IQ` > IoT > 协议类型 (ZigBee) > 通用设置 > 调制方式

**示例** `:RADio:ZIGBee:PHY:SCHEme BPSK`  
`:RADio:ZIGBee:PHY:SCHEme?`  
返回值:  
`BPSK\n`

#### 3.4.16.2.8 空闲间隔 ([:SOURce]:RADio:ZIGBee:IDLE)

**命令格式** `[:SOURce]:RADio:ZIGBee:IDLE <val>`  
`[:SOURce]:RADio:ZIGBee:IDLE?`

**功能描述** 此命令设置/查询 ZigBee 调制波所包含的帧之间的空闲间隔 (以秒为单位)。

**参数类型** 浮点型, 单位: ns, us, ms 或 s, 默认为 s。

**参数范围** 0 ~ 200 ms

**返回值** 浮点型, 单位: s

**默认值** 100 us

**对应菜单** `IQ` > IoT > 协议类型 (ZigBee) > 通用设置 > 空闲间隔

**示例** `:RADio:ZIGBee:IDLE 10 ms`  
`:RADio:ZIGBee:IDLE?`  
返回值:  
`0.01\n`

#### 3.4.16.2.9 频带 ([:SOURce]:RADio:ZIGBee:FREQ:BAND)

**命令格式** `[:SOURce]:RADio:ZIGBee:FREQ:BAND F868M | F915M| F2450M`  
`[:SOURce]:RADio:ZIGBee:FREQ:BAND?`

**功能描述** 此命令设置/查询 ZigBee 调制的频段。

**参数类型** 枚举

参数范围	F868M   F915M  F2450M
返回值	枚举
默认值	F868M
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 通用设置 > 频段
示例	<code>:RADio:ZIGBee:FREQ:BAND F915M</code> <code>:RADio:ZIGBee:FREQ:BAND?</code> 返回值: <code>F915M\n</code>

#### 3.4.16.2.10 查询数据速率 ([:SOURce]:RADio:ZIGBee:RATE?)

命令格式	<code>[:SOURce]:RADio:ZIGBee:RATE?</code>
功能描述	此命令查询 ZigBee 调制的数据速率。
返回值	整型, 单位: b/s。
默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 通用设置 > 数据速率
示例	<code>:RADio:ZIGBee:RATE?</code> 返回值: <code>40000\n</code>

#### 3.4.16.2.11 查询前导码 ([:SOURce]:RADio:ZIGBee:PREAmble?)

命令格式	<code>[:SOURce]:RADio:ZIGBee:PREAmble?</code>
功能描述	此命令获取 ZigBee PPDU 的前导码字段 (十六进制)。
返回值	整型
默认值	00000000
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 前导码 (Hex)
示例	<code>:RADio:ZIGBee:PREAmble?</code> 返回值: <code>00000000\n</code>

#### 3.4.16.2.12 查询 SFD ([:SOURce]:RADio:ZIGBee:SFD?)

命令格式	<code>[:SOURce]:RADio:ZIGBee:SFD?</code>
功能描述	此命令获取 ZigBee PPDU 的 SFD 字段 (十六进制)。
返回值	整型
默认值	a7
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > SFD (Hex)

示例	<code>:RADio:ZIGBee:SFD?</code> 返回值: <code>a7\n</code>
----	--

#### 3.4.16.2.13 查询 PHR ([:SOURce]:RADio:ZIGBee:PHR?)

命令格式	<code>[:SOURce]:RADio:ZIGBee:PHR?</code>
功能描述	此命令获取 ZigBee PDU 的 PHR 字段 (十六进制)。
返回值	整型
默认值	21
对应菜单	<code>[IQ]</code> > IoT > 协议类型 (ZigBee) > PDU 设置 > PHR (Hex)
示例	<code>:RADio:ZIGBee:PHR?</code> 返回值: <code>21\n</code>

#### 3.4.16.2.14 MAC 帧类型 ([:SOURce]:RADio:ZIGBee:FRAMe:TYPE)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FRAMe:TYPE General   Beacon   Data   ACK   MAC</code> <code>[:SOURce]:RADio:ZIGBee:FRAMe:TYPE?</code>
功能描述	此命令设置/查询 ZigBee 调制的帧类型。
参数类型	枚举
参数范围	General   Beacon   Data   ACK   MAC
返回值	枚举
默认值	General
对应菜单	<code>[IQ]</code> > IoT > 协议类型 (ZigBee) > PDU 设置 > MAC 帧类型
示例	<code>:RADio:ZIGBee:FRAMe:TYPE Data</code> <code>:RADio:ZIGBee:FRAMe:TYPE?</code> 返回值: <code>Data\n</code>

#### 3.4.16.2.15 MAC Header 状态开关 ([:SOURce]:RADio:ZIGBee:MAC:HEADer:STATe)

命令格式	<code>[:SOURce]:RADio:ZIGBee:MAC:HEADer:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZIGBee:MAC:HEADer:STATe?</code>
功能描述	此命令设置/查询 ZigBee PDU 的 MAC Header 字段的使能状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	1

对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC Header
示例	<code>:RADio:ZIGBee:MAC:HEADer:STATe OFF</code> <code>:RADio:ZIGBee:MAC:HEADer:STATe?</code> 返回值: <code>0\n</code>

#### 3.4.16.2.16 通用帧 MAC Header 字段 ([:SOURce]:RADio:ZIGBee:FRAMe:GENERal)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FRAMe:GENERal &lt;"header"&gt;</code> <code>[:SOURce]:RADio:ZIGBee:FRAMe:GENERal?</code>
功能描述	此命令设置/查询 ZigBee 通用帧的 MAC Header 字段。
参数类型	字符串
参数范围	无
返回值	字符串
默认值	<code>8821,01,1234,5678,4321,8765,,, \n</code>
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC 帧类型 (通用帧) > MAC Header
示例	<code>:RADio:ZIGBee:FRAMe:GENERal "8888,01,1234,5678,4321,8765,,,AABB"</code> <code>:RADio:ZIGBee:FRAMe:GENERal?</code> 返回值: <code>8888,01,1234,5678,4321,8765,,,AABB\n</code>

#### 3.4.16.2.17 信标帧 MAC Header 字段 ([:SOURce]:RADio:ZIGBee:FRAMe:BEACon)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FRAMe:BEACon &lt;"header"&gt;</code> <code>[:SOURce]:RADio:ZIGBee:FRAMe:BEACon?</code>
功能描述	此命令设置/查询 ZigBee 信标帧的 MAC Header 字段。
参数类型	字符串
参数范围	无
返回值	字符串
默认值	<code>8820,01,1234,5678,4321,8765,,0000,00,00\n</code>
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC 帧类型 (信标帧) > MAC Header
示例	<code>:RADio:ZIGBee:FRAMe:BEACon</code> <code>"8821,01,1234,5678,4321,8765,,0000,BD,00"</code> <code>:RADio:ZIGBee:FRAMe:BEACon?</code> 返回值: <code>8820,01,1234,5678,4321,8765,,0000,BD,00\n</code>

## 3.4.16.2.18 数据帧 MAC Header 字段 ([:SOURCE]:RADIO:ZIGBEE:FRAME:DATA)

命令格式	<code>[:SOURCE]:RADIO:ZIGBEE:FRAME:DATA &lt;"header"&gt;</code> <code>[:SOURCE]:RADIO:ZIGBEE:FRAME:DATA?</code>
功能描述	此命令设置/查询 ZigBee 数据帧的 MAC Header 字段。
参数类型	字符串
参数范围	无
返回值	字符串
默认值	8821,01,1234,5678,4321,8765,,\n
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC 帧类型 (数据帧) > MAC Header
示例	<code>:RADIO:ZIGBEE:FRAME:DATA "8821,01,1234,5678,4321,8765,,AABB,"</code> <code>:RADIO:ZIGBEE:FRAME:DATA?</code> 返回值: <code>8821,01,1234,5678,4321,8765,,AABB,\n</code>

## 3.4.16.2.19 应答帧 MAC Header 字段 ([:SOURCE]:RADIO:ZIGBEE:FRAME:ACK)

命令格式	<code>[:SOURCE]:RADIO:ZIGBEE:FRAME:ACK &lt;"header"&gt;</code> <code>[:SOURCE]:RADIO:ZIGBEE:FRAME:ACK?</code>
功能描述	此命令设置/查询 ZigBee 应答帧的 MAC Header 字段。
参数类型	字符串
参数范围	无
返回值	字符串
默认值	8822,01\n
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC 帧类型 (应答帧) > MAC Header
示例	<code>:RADIO:ZIGBEE:FRAME:ACK "1212,56"</code> <code>:RADIO:ZIGBEE:FRAME:ACK?</code> 返回值: <code>1212,56\n</code>

## 3.4.16.2.20 命令帧 MAC Header 字段 ([:SOURCE]:RADIO:ZIGBEE:FRAME:MAC)

命令格式	<code>[:SOURCE]:RADIO:ZIGBEE:FRAME:MAC &lt;"header"&gt;</code> <code>[:SOURCE]:RADIO:ZIGBEE:FRAME:MAC?</code>
功能描述	此命令设置/查询 ZigBee 命令帧的 MAC Header 字段。
参数类型	字符串
参数范围	无
返回值	字符串

默认值	8823,01,1234,5678,4321,8765,,,,00\n
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC 帧类型 (命令帧) > MAC Header
示例	<code>:RADio:ZIGBee:FRAMe:MAC "8823,56,1234,5678,4321,8765,, AABB,, 12"</code> <code>:RADio:ZIGBee:FRAMe:MAC?</code> 返回值: <code>8823,01,1234,5678,4321,8765,,,,00\n</code>

#### 3.4.16.2.21 数据类型 ([:SOURce]:RADio:ZIGBee:PNTYpe)

命令格式	<code>[:SOURce]:RADio:ZIGBee:PNTYpe PN9   PN15   USER</code> <code>[:SOURce]:RADio:ZIGBee:PNTYpe?</code>
功能描述	此命令设置/查询 ZigBee PPDU 的数据类型。
参数类型	枚举
参数范围	PN9   PN15   USER
返回值	枚举
默认值	PN9
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据类型
示例	<code>:RADio:ZIGBee:PNTYpe PN15</code> <code>:RADio:ZIGBee:PNTYpe?</code> 返回值: <code>PN15\n</code>

#### 3.4.16.2.22 PN 种子 ([:SOURce]:RADio:ZIGBee:SEED)

命令格式	<code>[:SOURce]:RADio:ZIGBee:SEED &lt;val&gt;</code> <code>[:SOURce]:RADio:ZIGBee:SEED?</code>
功能描述	此命令设置/查询生成 PN 序列的种子。
参数类型	整型
参数范围	1 ~ 32767
返回值	整型
默认值	511
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > PN 种子 (Hex)
示例	<code>:RADio:ZIGBee:SEED 974</code> <code>:RADio:ZIGBee:SEED?</code> 返回值: <code>974\n</code>

## 3.4.16.2.23 设置自定义 PN 数据 ([:SOURce]:RADio:ZIGBee:PN:USER:DATA)

命令格式	[:SOURce]:RADio:ZIGBee:PN:USER:DATA <user_data>
功能描述	此命令为 ZigBee PPDU Payload 字段设置自定义的 PN 数据。
参数类型	二进制字符串
参数范围	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据类型 (自定义) > 自定义数据
示例	<code>:RADio:ZIGBee:PN:USER:DATA 0101001100111</code>

## 3.4.16.2.24 查询 PN 数据 ([:SOURce]:RADio:ZIGBee:PN:DATA?)

命令格式	[:SOURce]:RADio:ZIGBee:PN:DATA? PN9   PN15   USER
功能描述	此命令查询 ZigBee PPDU Payload 字段的 PN 数据。
参数类型	枚举
参数范围	PN9   PN15   USER
返回值	二进制字符串
默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据类型 (自定义) > 自定义数据
示例	<code>:RADio:ZIGBee:PN:DATA? USER</code> 返回值: <code>0101001100111n</code>

## 3.4.16.2.25 保存自定义 PN 数据 ([:SOURce]:RADio:ZIGBee:PN:SAVE)

命令格式	[:SOURce]:RADio:ZIGBee:PN:SAVE <"file_name">
功能描述	此命令将 ZigBee PPDU Payload 字段的自定义 PN 数据保存进 UDATA 文件。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据类型 (自定义) > 自定义数据 > 保存
示例	<code>:RADio:ZIGBee:PN:SAVE "test.udata"</code>

## 3.4.16.2.26 加载自定义 PN 数据 ([:SOURce]:RADio:ZIGBee:PN:LOAD)

命令格式	[:SOURce]:RADio:ZIGBee:PN:LOAD <"file_name">
------	--

功能描述	此命令从 UDATA 文件中加载 ZigBee PPDU Payload 字段的自定义 PN 数据。
参数类型	字符串
参数范围	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据类型 (自定义) > 自定义数据 > 加载
示例	<code>:RADio:ZIGBee:PN:LOAD "test.udata"</code>

#### 3.4.16.2.27 数据长度 ([:SOURce]:RADio:ZIGBee:FRAME:LEN)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FRAME:LEN &lt;val&gt;</code> <code>[:SOURce]:RADio:ZIGBee:FRAME:LEN?</code>
功能描述	此命令设置/查询 ZigBee PPDU Payload 字段的数据长度。
参数类型	整型, 单位: octets
参数范围	0 ~ 114
返回值	整型, 单位: octets
默认值	20
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据长度
示例	<code>:RADio:ZIGBee:FRAME:LEN 10</code> <code>:RADio:ZIGBee:FRAME:LEN?</code> 返回值: <code>10\n</code>

#### 3.4.16.2.28 数据连续模式 ([:SOURce]:RADio:ZIGBee:CONTInuous:STATe)

命令格式	<code>[:SOURce]:RADio:ZIGBee:CONTInuous:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZIGBee:CONTInuous:STATe?</code>
功能描述	该命令启用或禁用数据连续状态。连续模式将使数据位连续分布在多帧中, 截断模式将对所有帧具有相同的有效负载数据位。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > 数据模式
示例	<code>:RADio:ZIGBee:CONTInuous:STATe 1</code> <code>:RADio:ZIGBee:CONTInuous:STATe?</code> 返回值: <code>1\n</code>

## 3.4.16.2.29 MAC FCS 开关 ([:SOURce]:RADio:ZIGBee:FCS:STATe)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FCS:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZIGBee:FCS:STATe?</code>
功能描述	在 PSDU 中启用或禁用 MAC FCS。当关闭时，它可用于模拟无效 FCS 情况，因为 FCS 部分实际上填充了用户数据位。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	1
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > PPDU 设置 > MAC FCS
示例	<code>:RADio:ZIGBee:FCS:STATe 0</code> <code>:RADio:ZIGBee:FCS:STATe?</code> 返回值: <code>0</code>

## 3.4.16.2.30 符号率偏差 ([:SOURce]:RADio:ZIGBee:SYMBle:TIMIng:ERROr)

命令格式	<code>[:SOURce]:RADio:ZIGBee:SYMBle:TIMIng:ERROr &lt;val&gt;</code> <code>[:SOURce]:RADio:ZIGBee:SYMBle:TIMIng:ERROr?</code>
功能描述	此命令设置/查询 ZigBee 调制的符号率偏差。
参数类型	整型，单位：ppm
参数范围	-300 ~ 300
返回值	整型，单位：ppm
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 损伤 > 符号率偏差
示例	<code>:RADio:ZIGBee:SYMBle:TIMIng:ERROr -10</code> <code>:RADio:ZIGBee:SYMBle:TIMIng:ERROr?</code> 返回值: <code>-10</code>

## 3.4.16.2.31 载波频率偏移 ([:SOURce]:RADio:ZIGBee:FREQ:OFFSEt)

命令格式	<code>[:SOURce]:RADio:ZIGBee:FREQ:OFFSEt &lt;val&gt;</code> <code>[:SOURce]:RADio:ZIGBee:FREQ:OFFSEt?</code>
功能描述	此命令设置或获取 ZigBee 调制的标称载波频率的偏移 (以 Hz 为单位)。
参数类型	浮点型，单位：Hz
参数范围	-200000 ~ 200000
返回值	浮点型，单位：Hz
默认值	0

对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 损伤 > 载波频率偏移
示例	<code>:RADio:ZIGBee:FREQ:OFFSET 1000</code> <code>:RADio:ZIGBee:FREQ:OFFSET?</code> 返回值: <code>1000\n</code>

#### 3.4.16.2.32 标识 1 ([:SOURce]:RADio:ZIGBee:MARKer:ONE:SOURce)

命令格式	<code>[:SOURce]:RADio:ZIGBee:MARKer:ONE:SOURce WAVEform  FRAME</code> <code>[:SOURce]:RADio:ZIGBee:MARKer:ONE:SOURce?</code>
功能描述	此命令设置/查询 ZigBee 调制波的标记 1 的源。 波形开始 - 指示波形的开始, 帧开始 - 表示波形中每帧的开始。
参数类型	枚举
参数范围	WAVEform  FRAME
返回值	枚举
默认值	FRAME
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 标识设置 > 标识 1
示例	<code>:RADio:ZIGBee:MARKer:ONE:SOURce WAVEform</code> <code>:RADio:ZIGBee:MARKer:ONE:SOURce?</code> 返回值: <code>WAVEform\n</code>

#### 3.4.16.2.33 标识 2 ([:SOURce]:RADio:ZIGBee:MARKer:TWO:SOURce)

命令格式	<code>[:SOURce]:RADio:ZIGBee:MARKer:TWO:SOURce WAVEform  FRAME</code> <code>[:SOURce]:RADio:ZIGBee:MARKer:TWO:SOURce?</code>
功能描述	此命令设置/查询 ZigBee 调制波的标记 2 的源。 波形开始 - 指示波形的开始, 帧开始 - 表示波形中每帧的开始。
参数类型	枚举
参数范围	WAVEform  FRAME
返回值	枚举
默认值	FRAME
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 标识设置 > 标识 2
示例	<code>:RADio:ZIGBee:MARKer:TWO:SOURce WAVEform</code> <code>:RADio:ZIGBee:MARKer:TWO:SOURce?</code> 返回值: <code>WAVEform\n</code>

## 3.4.16.2.34 ZigBee 触发类型 ([:SOURce]:RADio:ZIGBee:TRIGger:TYPE)

命令格式	[:SOURce]:RADio:ZIGBee:TRIGger:TYPE CONTInous  SINGle  GATE [:SOURce]:RADio:ZIGBee:TRIGger:TYPE?
功能描述	此命令设置/查询 ZigBee 触发的触发类型。
参数类型	枚举
参数范围	CONTInous  SINGle  GATE
返回值	枚举
默认值	CONTInous
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 触发设置 > 触发类型
示例	:RADio:ZIGBee:TRIGger:TYPE SINGle :RADio:ZIGBee:TRIGger:TYPE? 返回值: SINGle\n

## 3.4.16.2.35 ZigBee 连续触发模式 ([:SOURce]:RADio:ZIGBee:TRIGger:CONTInous)

命令格式	[:SOURce]:RADio:ZIGBee:TRIGger:CONTInous FREERun   RUNIgnored [:SOURce]:RADio:ZIGBee:TRIGger:CONTInous?
功能描述	此命令设置/查询 ZigBee 连续触发的触发模式。
参数类型	枚举
参数范围	FREERun   RUNIgnored
返回值	枚举
默认值	FREERun
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 触发设置 > 触发类型 (连续) > 连续模式
示例	:RADio:ZIGBee:TRIGger:CONTInous RUNIgnored :RADio:ZIGBee:TRIGger:CONTInous? 返回值: RUNIgnored\n

## 3.4.16.2.36 ZigBee 门选通触发模式 ([:SOURce]:RADio:ZIGBee:TRIGger:GATE)

命令格式	[:SOURce]:RADio:ZIGBee:TRIGger:GATE LOW   HIGH [:SOURce]:RADio:ZIGBee:TRIGger:GATE?
功能描述	此命令设置/查询 ZigBee 门选通触发的触发模式。
参数类型	枚举
参数范围	LOW   HIGH
返回值	枚举
默认值	LOW

对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 触发设置 > 触发类型 (门选通) > 门选通模式
示例	<code>:RADio:ZIGBee:TRIGger:GATE HIGH</code> <code>:RADio:ZIGBee:TRIGger:GATE?</code> 返回值: <code>HIGHn</code>

#### 3.4.16.2.37 ZigBee 触发源 ([:SOURce]:RADio:ZIGBee:TRIGger:SOURce)

命令格式	<code>[:SOURce]:RADio:ZIGBee:TRIGger:SOURce KEY BUS EXT</code> <code>[:SOURce]:RADio:ZIGBee:TRIGger:SOURce?</code>
功能描述	此命令设置/查询 ZigBee 的触发源。
参数类型	枚举
参数范围	KEY BUS EXT
返回值	枚举
默认值	KEY
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 触发设置 > 触发源
示例	<code>:RADio:ZIGBee:TRIGger:SOURce BUS</code> <code>:RADio:ZIGBee:TRIGger:SOURce?</code> 返回值: <code>BUSn</code>

#### 3.4.16.2.38 ZigBee 外触发极性 ([:SOURce]:RADio:ZIGBee:TRIGger:POL)

命令格式	<code>[:SOURce]:RADio:ZIGBee:TRIGger:POL POS NEG</code> <code>[:SOURce]:RADio:ZIGBee:TRIGger:POL?</code>
功能描述	此命令设置/查询 ZigBee 外触发的触发极性。
参数类型	枚举
参数范围	POS NEG
返回值	枚举
默认值	POS
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 触发设置 > 触发源 (外部) > 外触发极性
示例	<code>:RADio:ZIGBee:TRIGger:POL NEG</code> <code>:RADio:ZIGBee:TRIGger:POL?</code> 返回值: <code>NEGn</code>

#### 3.4.16.2.39 ZigBee 外触发延迟采样点数 ([:SOURce]:RADio:ZIGBee:TRIGger:DELAy:SAMPlE)

命令格式	<code>[:SOURce]:RADio:ZIGBee:TRIGger:DELAy:SAMPlE &lt;samples&gt;</code> <code>[:SOURce]:RADio:ZIGBee:TRIGger:DELAy:SAMPlE?</code>
------	---

功能描述	此命令设置/查询 ZigBee 外触发的延迟触发采样点数。
参数类型	整型
参数范围	0 ~ 100000000
返回值	整型
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (ZigBee) > 触发设置 > 延迟采样点数
示例	<code>:RADio:ZIGBee:TRIGger:DELAy:SAMPLe 1000</code> <code>:RADio:ZIGBee:TRIGger:DELAy:SAMPLe?</code> 返回值: <code>1000\n</code>

#### 3.4.16.2.40 ZigBee 总线触发 ([:SOURce]:RADio:ZIGBee:TRG)

命令格式	<code>[:SOURce]:RADio:ZIGBee:TRG</code>
功能描述	此命令发送 ZigBee 总线触发源的触发信号。
对应菜单	无
示例	<code>:RADio:ZIGBee:TRG</code>

#### 3.4.16.3 Z-Wave 设置

##### 3.4.16.3.1 保存波形 ([:SOURce]:RADio:ZWAVE:SAVE)

命令格式	<code>[:SOURce]:RADio:ZWAVE:SAVE &lt;"file_name"&gt;</code>
功能描述	将 Z-Wave 调制的波形数据保存到文件。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 保存波形
示例	<code>:RADio:ZWAVE:SAVE "test_zwave.arb"</code>

##### 3.4.16.3.2 Z-Wave 状态开关 ([:SOURce]:RADio:ZWAVE:STATe)

命令格式	<code>[:SOURce]:RADio:ZWAVE:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZWAVE:STATe?</code>
功能描述	此命令设置/查询 Z-Wave 调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	0

对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > Z-Wave 状态
示例	<code>:RADio:ZWAVe:STATe ON</code> <code>:RADio:ZWAVe:STATe?</code> 返回值: <code>1\n</code>

#### 3.4.16.3.3 过采样率 ([:SOURce]:RADio:ZWAVe:OSAMple:RATIo)

命令格式	<code>[:SOURce]:RADio:ZWAVe:OSAMple:RATIo &lt;val&gt;</code> <code>[:SOURce]:RADio:ZWAVe:OSAMple:RATIo?</code>
功能描述	此命令设置/查询 Z-Wave 调制波的过采样率。
参数类型	整型
参数范围	2 ~ 64
返回值	整型
默认值	8
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 基本设置 > 过采样率
示例	<code>:RADio:ZWAVe:OSAMple:RATIo 10</code> <code>:RADio:ZWAVe:OSAMple:RATIo?</code> 返回值: <code>10\n</code>

#### 3.4.16.3.4 帧数 ([:SOURce]:RADio:ZWAVe:FRAMe:NUMBer)

命令格式	<code>[:SOURce]:RADio:ZWAVe:FRAMe:NUMBer &lt;val&gt;</code> <code>[:SOURce]:RADio:ZWAVe:FRAMe:NUMBer?</code>
功能描述	此命令设置和获取 Z-Wave 调制波中包含的帧数。
参数类型	整型
参数范围	1 ~ 2000
返回值	整型
默认值	1
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 基本设置 > 帧数
示例	<code>:RADio:ZWAVe:FRAMe:NUMBer 2</code> <code>:RADio:ZWAVe:FRAMe:NUMBer?</code> 返回值: <code>2\n</code>

#### 3.4.16.3.5 查询总采样点数 ([:SOURce]:RADio:ZWAVe:TOTAL:SAMPlE:POINtS?)

命令格式	<code>[:SOURce]:RADio:ZWAVe:TOTAL:SAMPlE:POINtS?</code>
功能描述	此命令查询 Z-Wave 调制波的总采样点数。
返回值	整型

默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 基本设置 > 总采样点数
示例	:RADio:ZWAVe:TOTAL:SAMPle:POINts? 返回值: 17720\n

#### 3.4.16.3.6 查询波形长度 ([:SOURce]:RADio:ZWAVe:WAVEform:LENGth?)

命令格式	[:SOURce]:RADio:ZWAVe:WAVEform:LENGth?
功能描述	此命令查询 Z-Wave 调制波的波形长度。
返回值	浮点型, 单位: s。
默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 基本设置 > 波形长度
示例	:RADio:ZWAVe:WAVEform:LENGth? 返回值: 0.0922916666666667\n

#### 3.4.16.3.7 数据速率 ([:SOURce]:RADio:ZWAVe:RATE:TYPE)

命令格式	[:SOURce]:RADio:ZWAVe:RATE:TYPE R1 R2 R3 [:SOURce]:RADio:ZWAVe:RATE:TYPE?
功能描述	此命令设置/查询 Z-Wave 调制的数据速率。
参数类型	枚举
参数范围	R1: R1 -- 9.6 kbps, R2: R2 -- 40 kbps, R3: R3 -- 100 kbps。
返回值	枚举
默认值	R1
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 通用设置 > 数据速率
示例	:RADio:ZWAVe:RATE:TYPE R2 :RADio:ZWAVe:RATE:TYPE? 返回值: R2\n

#### 3.4.16.3.8 调制方式 ([:SOURce]:RADio:ZWAVe:MODUlation:TYPE)

命令格式	[:SOURce]:RADio:ZWAVe:MODUlation:TYPE FSK   GFSK [:SOURce]:RADio:ZWAVe:MODUlation:TYPE?
功能描述	此命令设置/查询 Z-Wave 调制的调制方式。
参数类型	枚举

参数范围	FSK   GFSK
返回值	枚举
默认值	R1/R2 为 FSK, R3 为 GFSK。
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 通用设置 > 调制方式
示例	<code>:RADio:ZWAVE:MODUlation:TYPE GFSK</code> <code>:RADio:ZWAVE:MODUlation:TYPE?</code> 返回值: <code>GFSK\n</code>

#### 3.4.16.3.9 空闲间隔 ([:SOURce]:RADio:ZWAVE:IDLE)

命令格式	<code>[:SOURce]:RADio:ZWAVE:IDLE &lt;val&gt;</code> <code>[:SOURce]:RADio:ZWAVE:IDLE?</code>
功能描述	此命令设置/查询 Z-Wave 调制波所包含的帧之间的空闲间隔 (以秒为单位)。
参数类型	浮点型, 单位: ns, us, ms 或 s, 默认为 s。
参数范围	0 ~ 200 ms
返回值	浮点型, 单位: s
默认值	100 us
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 通用设置 > 空闲间隔
示例	<code>:RADio:ZWAVE:IDLE 10 ms</code> <code>:RADio:ZWAVE:IDLE?</code> 返回值: <code>0.01\n</code>

#### 3.4.16.3.10 斜坡符号数 ([:SOURce]:RADio:ZWAVE:RAMP:SYMBOL)

命令格式	<code>[:SOURce]:RADio:ZWAVE:RAMP:SYMBOL &lt;val&gt;</code> <code>[:SOURce]:RADio:ZWAVE:RAMP:SYMBOL?</code>
功能描述	此命令设置/查询 Z-Wave 调制的波形斜坡上升和下降的持续符号数。
参数类型	整型
参数范围	1 ~ 10
返回值	整型
默认值	10
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 基本设置 > 斜坡符号数
示例	<code>:RADio:ZWAVE:RAMP:SYMBOL 6</code> <code>:RADio:ZWAVE:RAMP:SYMBOL?</code> 返回值: <code>6\n</code>

## 3.4.16.3.11 斜坡符号类型 ([:SOURce]:RADio:ZWAVE:RAMP:MODE)

命令格式	[:SOURce]:RADio:ZWAVE:RAMP:MODE FIRLast CENTer ONE ZERO [:SOURce]:RADio:ZWAVE:RAMP:MODE?
功能描述	此命令选择/查询 Z-Wave 调制的波形斜坡上升和下降周期的符号类型。
参数类型	枚举
参数范围	FIRLast CENTer ONE ZERO
返回值	枚举
默认值	FIRLast
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 通用设置 > 斜坡符号类型
示例	:RADio:ZWAVE:RAMP:MODE CENTer :RADio:ZWAVE:RAMP:MODE? 返回值: CENTer\n

## 3.4.16.3.12 查询 PPDU 长度 ([:SOURce]:RADio:ZWAVE:PPDU:LENGth?)

命令格式	[:SOURce]:RADio:ZWAVE:PPDU:LENGth?
功能描述	此命令查询 Z-Wave 调制的 PPDU 字节长度。
返回值	整型, 单位: octets。
默认值	54
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > PPDU 长度
示例	:RADio:ZWAVE:PPDU:LENGth? 返回值: 53\n

## 3.4.16.3.13 前导码长度 ([:SOURce]:RADio:ZWAVE:PREAmble:LENGth)

命令格式	[:SOURce]:RADio:ZWAVE:PREAmble:LENGth <val> [:SOURce]:RADio:ZWAVE:PREAmble:LENGth?
功能描述	此命令设置/获取 Z-Wave PPDU 的前导码长度。
参数类型	整型, 单位: octets。
参数范围	10 ~ 100
返回值	整型, 单位: octets。
默认值	10
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 前导码长度
示例	:RADio:ZWAVE:PREAmble:LENGth 12 :RADio:ZWAVE:PREAmble:LENGth? 返回值: 12\n

## 3.4.16.3.14 帧起始 ([:SOURce]:RADio:ZWAVe:SOF)

命令格式	[:SOURce]:RADio:ZWAVe:SOF <val> [:SOURce]:RADio:ZWAVe:SOF?
功能描述	此命令设置/获取 Z-Wave PPDU 的帧起始字段 (十进制)。
参数类型	整型
参数范围	0 ~ 255
返回值	整型
默认值	240
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 帧起始 (Hex)
示例	:RADio:ZWAVe:SOF 100 :RADio:ZWAVe:SOF? 返回值: 100\n

## 3.4.16.3.15 MAC Header 状态开关 ([:SOURce]:RADio:ZWAVe:MAC:HEADer:STATe)

命令格式	[:SOURce]:RADio:ZWAVe:MAC:HEADer:STATe ON OFF 1 0 [:SOURce]:RADio:ZWAVe:MAC:HEADer:STATe?
功能描述	此命令设置/查询 Z-Wave PPDU 的 MAC Header 字段的使能状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	1
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > MAC Header
示例	:RADio:ZWAVe:MAC:HEADer:STATe OFF :RADio:ZWAVe:MAC:HEADer:STATe? 返回值: 0\n

## 3.4.16.3.16 MAC Header 字段 ([:SOURce]:RADio:ZWAVe:MAC:HEADer)

命令格式	[:SOURce]:RADio:ZWAVe:MAC:HEADer <"header"> [:SOURce]:RADio:ZWAVe:MAC:HEADer?
功能描述	此命令设置/查询 Z-Wave PPDU 的 MAC Header 字段。
参数类型	字符串
参数范围	无
返回值	字符串
默认值	12345678,01,3412,2A,01,01

对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > MAC Header
示例	<code>:RADio:ZWAVe:MAC:HEADer "12345678,01,3412,2B,01,AB"</code> <code>:RADio:ZWAVe:MAC:HEADer?</code> 返回值: <code>12345678,01,3412,2A,01,AB\n</code>

### 3.4.16.3.17 Mac Header Sequence Number 开关状态

#### ([:SOURce]:RADio:ZWAVe:MAC:HEADer:SEQUence:STATe)

命令格式	<code>[:SOURce]:RADio:ZWAVe:MAC:HEADer:SEQUence:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZWAVe:MAC:HEADer:SEQUence:STATe?</code>
功能描述	此命令设置/查询 Z-Wave PPDU MAC Header 字段中的 Sequence Number 字段的开启状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > MAC Header > Sequence Number
示例	<code>:RADio:ZWAVe:MAC:HEADer:SEQUence:STATe ON</code> <code>:RADio:ZWAVe:MAC:HEADer:SEQUence:STATe?</code> 返回值: <code>1\n</code>

### 3.4.16.3.18 数据类型 ([:SOURce]:RADio:ZWAVe:DATA:TYPE)

命令格式	<code>[:SOURce]:RADio:ZWAVe:DATA:TYPE PN9   PN15   USER</code> <code>[:SOURce]:RADio:ZWAVe:DATA:TYPE?</code>
功能描述	此命令设置/查询 Z-Wave PPDU 的数据类型。
参数类型	枚举
参数范围	PN9   PN15   USER
返回值	枚举
默认值	PN9
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据类型
示例	<code>:RADio:ZWAVe:DATA:TYPE PN15</code> <code>:RADio:ZWAVe:DATA:TYPE?</code> 返回值: <code>PN15\n</code>

## 3.4.16.3.19 PN 种子 ([:SOURce]:RADio:ZWAVE:PN:SEED)

命令格式	[:SOURce]:RADio:ZWAVE:PN:SEED <val> [:SOURce]:RADio:ZWAVE:PN:SEED?
功能描述	此命令设置/查询生成 PN 序列的种子。
参数类型	整型
参数范围	1 ~ 32767
返回值	整型
默认值	511
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > PN 种子 (Hex)
示例	:RADio:ZWAVE:PN:SEED 974 :RADio:ZWAVE:PN:SEED? 返回值: 974\n

## 3.4.16.3.20 设置自定义 PN 数据 ([:SOURce]:RADio:ZWAVE:PN:USER:DATA)

命令格式	[:SOURce]:RADio:ZWAVE:PN:USER:DATA <user_data>
功能描述	此命令为 Z-Wave PPDU Payload 字段设置自定义的 PN 数据。
参数类型	二进制字符串
参数范围	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据类型 (自定义) > 自定义数据
示例	:RADio:ZWAVE:PN:USER:DATA 0101001100111

## 3.4.16.3.21 查询 PN 数据 ([:SOURce]:RADio:ZWAVE:PN:DATA?)

命令格式	[:SOURce]:RADio:ZWAVE:PN:DATA? PN9   PN15   USER
功能描述	此命令查询 Z-Wave PPDU Payload 字段的 PN 数据。
参数类型	枚举
参数范围	PN9   PN15   USER
返回值	二进制字符串
默认值	无
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据类型 (自定义) > 自定义数据
示例	:RADio:ZIGBee:PN:DATA? PN9 返回值: 111111111000001111011111000101110011001000001001010011101

```

101000111100111110011011000101010010001110001101101010111
000100110001000100000000100001000110000100111001010101100
00110111101001101110010001010000101011010011111011001001
00101101111100100110101001100110000000110001100101000110
10010111111010001011000111010110010110011110001111101110
10000011010110110111011000001011010111101010101000000101
00101011110010111011100000011001110100100111101011101010
001001000011001110000101110110110011010000111011110000n

```

#### 3.4.16.3.22 保存自定义 PN 数据 ([:SOURce]:RADio:ZWAVE:PN:SAVE)

命令格式	[:SOURce]:RADio:ZWAVE:PN:SAVE <"file_name">
功能描述	此命令将 Z-Wave PPDU Payload 字段的自定义 PN 数据保存进文件。
参数类型	字符串
参数范围	无
对应菜单	<input type="checkbox"/> IQ > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据类型 (自定义) > 自定义数据 > 保存
示例	:RADio:ZWAVE:PN:SAVE "user_pn.udata"

#### 3.4.16.3.23 加载自定义 PN 数据 ([:SOURce]:RADio:ZWAVE:PN:LOAD)

命令格式	[:SOURce]:RADio:ZWAVE:PN:LOAD <"file_name">
功能描述	此命令从文件中加载 Z-Wave PPDU Payload 字段的自定义 PN 数据。
参数类型	字符串
参数范围	无
对应菜单	<input type="checkbox"/> IQ > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据类型 (自定义) > 自定义数据 > 加载
示例	:RADio:ZWAVE:PN:LOAD "user_pn.udata"

#### 3.4.16.3.24 数据长度 ([:SOURce]:RADio:ZWAVE:DATA:LENGth)

命令格式	[:SOURce]:RADio:ZWAVE:DATA:LENGth <val> [:SOURce]:RADio:ZWAVE:DATA:LENGth?
功能描述	此命令设置/查询 Z-Wave PPDU Payload 字段的数据长度。
参数类型	整型, 单位: octets
参数范围	取决于 PSDU 和 MSDU 字段的最大长度
返回值	整型, 单位: octets
默认值	32
对应菜单	<input type="checkbox"/> IQ > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据长度

示例	<code>:RADio:ZWAVe:DATA:LENGth 10</code> <code>:RADio:ZWAVe:DATA:LENGth?</code>
返回值:	<code>10\n</code>

#### 3.4.16.3.25 数据连续模式 ([:SOURce]:RADio:ZWAVe:CONTInuous:STATe)

命令格式	<code>[:SOURce]:RADio:ZWAVe:CONTInuous:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZWAVe:CONTInuous:STATe?</code>
功能描述	该命令启用或禁用数据连续状态。连续模式将使数据位连续分布在多帧中，截断模式将对所有帧具有相同的有效负载数据位。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 数据模式
示例	<code>:RADio:ZWAVe:CONTInuous:STATe 1</code> <code>:RADio:ZWAVe:CONTInuous:STATe?</code>
返回值:	<code>1\n</code>

#### 3.4.16.3.26 MAC FCS 开关 ([:SOURce]:RADio:ZWAVe:FCS:STATe)

命令格式	<code>[:SOURce]:RADio:ZWAVe:FCS:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZWAVe:FCS:STATe?</code>
功能描述	在 PSDU 中启用或禁用 MAC FCS。当关闭时，它可用于模拟无效 FCS 情况，因为 FCS 部分实际上填充了用户数据位。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	1
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > PPDU 设置 > MAC FCS
示例	<code>:RADio:ZWAVe:FCS:STATe 0</code> <code>:RADio:ZWAVe:FCS:STATe?</code>
返回值:	<code>0\n</code>

#### 3.4.16.3.27 帧尾定界符开关 ([:SOURce]:RADio:ZWAVe:EOF:STATe)

命令格式	<code>[:SOURce]:RADio:ZWAVe:EOF:STATe ON OFF 1 0</code> <code>[:SOURce]:RADio:ZWAVe:EOF:STATe?</code>
------	--

功能描述	在 PPDU 中启用或禁用 EHR。对于数据速率 R2 和 R3，此设置为关闭和只读状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	1
对应菜单	<input type="checkbox"/> IQ > IoT > 协议类型 (Z-Wave) > PPDU 设置 > 帧尾定界符
示例	<code>:RADio:ZWAVE:EOF:STATE 0</code> <code>:RADio:ZWAVE:EOF:STATE?</code> 返回值: <code>0\n</code>

#### 3.4.16.3.28 符号率偏差 ([:SOURce]:RADio:ZWAVE:SYMBle:TIMIng:ERROr)

命令格式	<code>[:SOURce]:RADio:ZWAVE:SYMBle:TIMIng:ERROr &lt;val&gt;</code> <code>[:SOURce]:RADio:ZWAVE:SYMBle:TIMIng:ERROr?</code>
功能描述	此命令设置/查询 Z-Wave 调制的符号率偏差。
参数类型	整型，单位：ppm
参数范围	-300 ~ 300
返回值	整型，单位：ppm
默认值	0
对应菜单	<input type="checkbox"/> IQ > IoT > 协议类型 (Z-Wave) > 损伤 > 符号率偏差
示例	<code>:RADio:ZWAVE:SYMBle:TIMIng:ERROr -10</code> <code>:RADio:ZWAVE:SYMBle:TIMIng:ERROr?</code> 返回值: <code>-10\n</code>

#### 3.4.16.3.29 载波频率偏移 ([:SOURce]:RADio:ZWAVE:FREQ:OFFSET)

命令格式	<code>[:SOURce]:RADio:ZWAVE:FREQ:OFFSET &lt;val&gt;</code> <code>[:SOURce]:RADio:ZWAVE:FREQ:OFFSET?</code>
功能描述	此命令设置或获取 Z-Wave 调制的标称载波频率的偏移（以 Hz 为单位）。
参数类型	浮点型，单位：Hz
参数范围	-200000 ~ 200000
返回值	浮点型，单位：Hz
默认值	0
对应菜单	<input type="checkbox"/> IQ > IoT > 协议类型 (Z-Wave) > 损伤 > 载波频率偏移
示例	<code>:RADio:ZWAVE:FREQ:OFFSET 1000</code> <code>:RADio:ZWAVE:FREQ:OFFSET?</code>

返回值:

1000\n

#### 3.4.16.3.30 频偏缩放 ([:SOURce]:RADio:ZWAVE:FREQ:DEVlation:SCALE)

命令格式	[:SOURce]:RADio:ZWAVE:FREQ:DEVlation:SCALE <val> [:SOURce]:RADio:ZWAVE:FREQ:DEVlation:SCALE?
功能描述	将附加标度设置为标称 FSK 频率偏差，这相当于对 FSK 调制指数应用缩放。
参数类型	浮点型
参数范围	0.5 ~ 1.5
返回值	浮点型
默认值	1
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 损伤 > 频偏缩放
示例	:RADio:ZWAVE:FREQ:DEVlation:SCALE 0.8 :RADio:ZWAVE:FREQ:DEVlation:SCALE? 返回值: 0.8\n

#### 3.4.16.3.31 Gaussian BT ([:SOURce]:RADio:ZWAVE:GAUSSian:BT)

命令格式	[:SOURce]:RADio:ZWAVE:GAUSSian:BT <val> [:SOURce]:RADio:ZWAVE:GAUSSian:BT?
功能描述	此命令设置或获取应用于 FSK 调制的高斯滤波器的高斯滤波器的 BT 乘积。
参数类型	浮点型
参数范围	0.1 ~ 10
返回值	浮点型
默认值	0.6
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 损伤 > Gaussian BT
示例	:RADio:ZWAVE:GAUSSian:BT 0.8 :RADio:ZWAVE:GAUSSian:BT? 返回值: 0.8\n

#### 3.4.16.3.32 标识 1 ([:SOURce]:RADio:ZWAVE:MARKer:ONE:SOURce)

命令格式	[:SOURce]:RADio:ZWAVE:MARKer:ONE:SOURce WAVEform  FRAME [:SOURce]:RADio:ZWAVE:MARKer:ONE:SOURce?
功能描述	此命令设置/查询 Z-Wave 调制波的标记 1 的源。 波形开始 - 指示波形的开始， 帧开始 - 表示波形中每帧的开始。

参数类型	枚举
参数范围	WAVEform  FRAME
返回值	枚举
默认值	FRAME
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 标识设置 > 标识 1
示例	:RADio:ZWAVE:MARKer:ONE:SOURce WAVEform :RADio:ZWAVE:MARKer:ONE:SOURce? 返回值: WAVEform\n

#### 3.4.16.3.33 标识 2 ([:SOURce]:RADio:ZWAVE:MARKer:TWO:SOURce)

命令格式	[:SOURce]:RADio:ZWAVE:MARKer:TWO:SOURce WAVEform  FRAME [:SOURce]:RADio:ZWAVE:MARKer:TWO:SOURce?
功能描述	此命令设置/查询 Z-Wave 调制波的标记 2 的源。 波形开始 - 指示波形的开始, 帧开始 - 表示波形中每帧的开始。
参数类型	枚举
参数范围	WAVEform  FRAME
返回值	枚举
默认值	FRAME
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 标识设置 > 标识 2
示例	:RADio:ZWAVE:MARKer:TWO:SOURce WAVEform :RADio:ZWAVE:MARKer:TWO:SOURce? 返回值: WAVEform\n

#### 3.4.16.3.34 Z-Wave 触发类型 ([:SOURce]:RADio:ZWAVE:TRIGger:TYPE)

命令格式	[:SOURce]:RADio:ZWAVE:TRIGger:TYPE CONTInous  SINGle  GATE [:SOURce]:RADio:ZWAVE:TRIGger:TYPE?
功能描述	此命令设置/查询 Z-Wave 触发的触发类型。
参数类型	枚举
参数范围	CONTInous  SINGle  GATE
返回值	枚举
默认值	CONTInous
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 触发设置 > 触发类型
示例	:RADio:ZWAVE:TRIGger:TYPE SINGle :RADio:ZWAVE:TRIGger:TYPE?

返回值：  
SINGle\n

#### 3.4.16.3.35 Z-Wave 连续触发模式 ([:SOURce]:RADio:ZWAVE:TRIGger:CONTInous)

命令格式	[:SOURce]:RADio:ZWAVE:TRIGger:CONTInous FREErun   RUNIgnored [:SOURce]:RADio:ZWAVE:TRIGger:CONTInous?
功能描述	此命令设置/查询 Z-Wave 连续触发的触发模式。
参数类型	枚举
参数范围	FREErun   RUNIgnored
返回值	枚举
默认值	FREErun
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 触发设置 > 连续模式
示例	:RADio:ZWAVE:TRIGger:CONTInous RUNIgnored :RADio:ZWAVE:TRIGger:CONTInous? 返回值： RUNIgnored\n

#### 3.4.16.3.36 Z-Wave 门选通触发模式 ([:SOURce]:RADio:ZWAVE:TRIGger:GATE)

命令格式	[:SOURce]:RADio:ZWAVE:TRIGger:GATE LOW   HIGH [:SOURce]:RADio:ZWAVE:TRIGger:GATE?
功能描述	此命令设置/查询 Z-Wave 门选通触发的触发模式。
参数类型	枚举
参数范围	LOW   HIGH
返回值	枚举
默认值	LOW
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 触发设置 > 门选通模式
示例	:RADio:ZWAVE:TRIGger:GATE HIGH :RADio:ZWAVE:TRIGger:GATE? 返回值： HIGH\n

#### 3.4.16.3.37 Z-Wave 触发源 ([:SOURce]:RADio:ZWAVE:TRIGger:SOURce)

命令格式	[:SOURce]:RADio:ZWAVE:TRIGger:SOURce KEY  BUS  EXT [:SOURce]:RADio:ZWAVE:TRIGger:SOURce?
功能描述	此命令设置/查询 Z-Wave 调制的触发源。
参数类型	枚举
参数范围	KEY  BUS  EXT

返回值	枚举
默认值	KEY
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 触发设置 > 触发源
示例	<code>:RADio:ZWAVe:TRIGger:SOURce BUS</code> <code>:RADio:ZWAVe:TRIGger:SOURce?</code> 返回值: <code>BUS\n</code>

#### 3.4.16.3.38 Z-Wave 外触发极性 ([:SOURce]:RADio:ZWAVe:TRIGger:POL)

命令格式	<code>[:SOURce]:RADio:ZWAVe:TRIGger:POL POS   NEG</code> <code>[:SOURce]:RADio:ZWAVe:TRIGger:POL?</code>
功能描述	此命令设置/查询 Z-Wave 外触发的触发极性。
参数类型	枚举
参数范围	POS   NEG
返回值	枚举
默认值	POS
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 触发设置 > 外触发极性
示例	<code>:RADio:ZWAVe:TRIGger:POL NEG</code> <code>:RADio:ZWAVe:TRIGger:POL?</code> 返回值: <code>NEG\n</code>

#### 3.4.16.3.39 Z-Wave 外触发延迟采样点数 ([:SOURce]:RADio:ZWAVe:TRIGger:DELAy:SAMPlE)

命令格式	<code>[:SOURce]:RADio:ZWAVe:TRIGger:DELAy:SAMPlE &lt;samples&gt;</code> <code>[:SOURce]:RADio:ZWAVe:TRIGger:DELAy:SAMPlE?</code>
功能描述	此命令设置/查询 Z-Wave 外触发的延迟触发采样点数。
参数类型	整型
参数范围	0 ~ 100000000
返回值	整型
默认值	0
对应菜单	<b>IQ</b> > IoT > 协议类型 (Z-Wave) > 触发设置 > 延迟采样点数
示例	<code>:RADio:ZWAVe:TRIGger:DELAy:SAMPlE 1000</code> <code>:RADio:ZWAVe:TRIGger:DELAy:SAMPlE?</code> 返回值: <code>1000\n</code>

### 3.4.16.3.40 Z-Wave 总线触发 ([:SOURce]:RADio:ZWAVE:TRG)

命令格式	[:SOURce]:RADio:ZWAVE:TRG
功能描述	此命令发送 Z-Wave 总线触发源的触发信号。
对应菜单	无
示例	:RADio:ZWAVE:TRG

## 3.4.17 多音调制设置

### 3.4.17.1 多音状态开关 ([:SOURce]:RADio:MTONE:ARB[:STATe])

命令格式	[:SOURce]:RADio:MTONE:ARB[:STATe] ON OFF 1 0 [:SOURce]:RADio:MTONE:ARB[:STATe]?
功能描述	此命令设置/查询多音调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	0
对应菜单	<b>IQ</b> > 多音 > 多音状态
示例	:RADio:MTONE:ARB ON :RADio:MTONE:ARB? 返回值: 1\n

### 3.4.17.2 多音个数 ([:SOURce]:RADio:MTONE:ARB:SETup:TABLE:NTONes)

命令格式	[:SOURce]:RADio:MTONE:ARB:SETup:TABLE:NTONes <num> [:SOURce]:RADio:MTONE:ARB:SETup:TABLE:NTONes?
功能描述	该命令设置/查询多音波形中的音调数量。
参数类型	整型
参数范围	1 ~ 20
返回值	整型
默认值	2
对应菜单	<b>IQ</b> > 多音 > 多音个数
示例	:RADio:MTONE:ARB:SETup:TABLE:NTONes 5 :RADio:MTONE:ARB:SETup:TABLE:NTONes? 返回值: 5\n

## 3.4.17.3 单边状态 ([:SOURce]:RADio:MTONe:ARB:SETup:TABLE:SINGle)

命令格式	[:SOURce]:RADio:MTONe:ARB:SETup:TABLE:SINGle ON OFF 1 0 [:SOURce]:RADio:MTONe:ARB:SETup:TABLE:SINGle?
功能描述	该命令启用或禁用多音单侧。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>IQ</b> > 多音 > 单边
示例	:RADio:MTONe:ARB:SETup:TABLE:SINGle ON :RADio:MTONe:ARB:SETup:TABLE:SINGle? 返回值: 1\n

## 3.4.17.4 采样率 ([:SOURce]:RADio:MTONe:ARB:SCLock:RATE)

命令格式	[:SOURce]:RADio:MTONe:ARB:SCLock:RATE <val> [:SOURce]:RADio:MTONe:ARB:SCLock:RATE?
功能描述	此命令设置/查询多音调制的采样时钟速率。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	500 Hz ~ 240 MHz
返回值	浮点型, 单位: Hz
默认值	2 MHz
对应菜单	<b>IQ</b> > 多音 > 采样率
示例	:RADio:MTONe:ARB:SCLock:RATE 4 MHz :RADio:MTONe:ARB:SCLock:RATE? 返回值: 400000\n

## 3.4.17.5 频率间隔 ([:SOURce]:RADio:MTONe:ARB:SETup:TABLE:FSPacing)

命令格式	[:SOURce]:RADio:MTONe:ARB:SETup:TABLE:FSPacing <val> [:SOURce]:RADio:MTONe:ARB:SETup:TABLE:FSPacing?
功能描述	此命令设置/查询总音调之间的频率间隔。
参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	多音个数*采样率/4096 ~ min (120M, 采样率/1.28)
返回值	浮点型, 单位: Hz

默认值	1 MHz
对应菜单	<b>IQ</b> > 多音 > 频率间隔
示例	<code>:RADio:MTONe:ARB:SETup:TABLE:FSPacing 2 MHz</code> <code>:RADio:MTONe:ARB:SETup:TABLE:FSPacing?</code> 返回值: <code>2000000n</code>

#### 3.4.17.6 保存状态 ([:SOURce]:RADio:MTONe:ARB:SETup:STORe)

命令格式	<code>[:SOURce]:RADio:MTONe:ARB:SETup:STORe &lt;"file_name"&gt;</code>
功能描述	该命令将当前多音波形设置存储在 MULSTATE 文件中。
参数类型	字符串
参数范围	命名规则请参阅用户手册。
对应菜单	<b>IQ</b> > 多音 > 保存状态
示例	<code>:RADio:MTONe:ARB:SETup:STORe "test.mulstate"</code>

#### 3.4.17.7 加载状态 ([:SOURce]:RADio:MTONe:ARB:SETup)

命令格式	<code>[:SOURce]:RADio:MTONe:ARB:SETup &lt;"file_name"&gt;</code>
功能描述	该命令从 MULSTATE 文件中加载多音波形设置。
参数类型	字符串
参数范围	无
对应菜单	<b>IQ</b> > 多音 > 加载状态
示例	<code>:RADio:MTONe:ARB:SETup "test.mulstate"</code>

### 3.4.18 AWGN 调制设置

#### 3.4.18.1 AWGN 状态开关 ([:SOURce]:RADio:AWGN:RT[:STATe])

命令格式	<code>[:SOURce]:RADio:AWGN:RT[:STATe] ON OFF 1 0</code> <code>[:SOURce]:RADio:AWGN:RT[:STATe]?</code>
功能描述	此命令设置/查询 AWGN 调制的开关状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	布尔型
默认值	0

对应菜单	<b>IQ</b> > AWGN > AWGN 状态
示例	<code>:RADio:AWGN:RT ON</code> <code>:RADio:AWGN:RT?</code> 返回值: <code>1\n</code>

### 3.4.18.2 带宽 ([:SOURce]:RADio:AWGN:RT:BWIDth)

命令格式	<code>[:SOURce]:RADio:AWGN:RT:BWIDth &lt;bandwidth&gt;</code> <code>[:SOURce]:RADio:AWGN:RT:BWIDth?</code>
功能描述	此命令设置/查询实时 AWGN 的带宽值。
参数类型	浮点型, 单位: Hz
参数范围	1 Hz ~ 150 MHz
返回值	浮点型, 单位: Hz
默认值	10 MHz
对应菜单	<b>IQ</b> > 多音 > 带宽
示例	<code>:RADio:AWGN:RT:BWIDth 5000000</code> <code>:RADio:AWGN:RT:BWIDth?</code> 返回值: <code>5000000\n</code>

## 3.5 SENSE 命令子系统

### 3.5.1 功率计设置

#### 3.5.1.1 查询功率计信息 ([:SENSe[:POWER]:TYPE?])

命令格式	<code>:SENSe[:POWER]:TYPE?</code>
功能描述	查询连接到射频/微波信号源的功率计的型号。
返回值	字符串
默认值	无
对应菜单	<b>HOME</b> > POWER SENSOR > 功率计信息
示例	<code>SENSe:TYPE?</code> 返回值: <code>NRP6A\n</code>

### 3.5.1.2 功率计状态 (:SENSe[:POWer]:STATus)

命令格式	:SENSe[:POWer]:STATus OFF ON 0 1 :SENSe[:POWer]:STATus?
功能描述	设置功率计的测量状态为打开或关闭。 查询功率计的测量状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<input type="button" value="HOME"/> > POWER SENSOR > 功率计状态
示例	<i>SENSe:STATus ON</i> <i>SENSe:STATus?</i> 返回值: <i>1\n</i>

### 3.5.1.3 查询功率计测量值 (:SENSe[:POWer]:VALue?)

命令格式	:SENSe[:POWer]:VALue?
功能描述	查询功率计的测量值。
返回值	浮点型, 单位: dBm
默认值	无
对应菜单	<input type="button" value="HOME"/> > POWER SENSOR > 测量功率
示例	<i>SENSe:VALue?</i> 返回值: <i>-0.02600282\n</i>

### 3.5.1.4 功率计测量统计状态 (:SENSe[:POWer]:STATIStics:STATe)

命令格式	:SENSe[:POWer]:STATIStics:STATe ON OFF 1 0 :SENSe[:POWer]:STATIStics:STATe?
功能描述	设置功率计的测量统计状态为打开或关闭。 查询功率计的测量统计状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0

对应菜单  > POWER SENSOR > 统计功能

示例 `SENSe:STATISTICS:STATe ON`  
`SENSe:STATISTICS:STATe?`

返回值:

`1\n`

### 3.5.1.5 查询功率计测量统计值 (:READ[:POWer]?)

命令格式 `:READ[:POWer]?`

功能描述 查询功率计测量统计值的平均值和最大值。

返回值 字符串

字符串形式: 平均值, 最大值

默认值 无

对应菜单  > POWER SENSOR > 统计功能 > 平均值/最大值

示例 `READ?`

返回值:

`-0.05,-0.04\n`

### 3.5.1.6 查询功率计测量统计的最大值 (:SENSe[:POWer]:STATISTICS:MAX?)

命令格式 `:SENSe[:POWer]:STATISTICS:MAX?`

功能描述 查询功率计测量统计的最大值。

返回值 浮点型, 单位: dBm

默认值 无

对应菜单  > POWER SENSOR > 统计功能 > 最大值

示例 `SENSe:STATISTICS:MAX?`

返回值:

`-0.03117205\n`

### 3.5.1.7 查询功率计测量统计的最小值 (:SENSe[:POWer]:STATISTICS:MIN?)

命令格式 `:SENSe[:POWer]:STATISTICS:MIN?`

功能描述 查询功率计测量统计的最小值。

返回值 浮点型, 单位: dBm

默认值 无

对应菜单  > POWER SENSOR > 统计功能 > 最小值

示例 `SENSe:STATIStics:MIN?`  
返回值:  
`-0.06101395\n`

### 3.5.1.8 查询功率计测量统计的平均值 (:SENSe[:POWer]:STATIStics:AVG?)

命令格式	:SENSe[:POWer]:STATIStics:AVG?
功能描述	查询功率计测量统计的平均值。
返回值	浮点型, 单位: dBm
默认值	无
对应菜单	<code>[HOME]</code> > POWER SENSOR > 统计功能 > 平均值
示例	<code>SENSe:STATIStics:AVG?</code> 返回值: <code>-0.0322136383619456\n</code>

### 3.5.1.9 查询功率计测量统计的计数 (:SENSe[:POWer]:STATIStics:COUNt?)

命令格式	:SENSe[:POWer]:STATIStics:COUNt?
功能描述	查询功率计测量统计的计数。
返回值	整型
默认值	无
对应菜单	<code>[HOME]</code> > POWER SENSOR > 统计功能 > 计数
示例	<code>SENSe:STATIStics:COUNt?</code> 返回值: <code>2035\n</code>

### 3.5.1.10 清空功率计的测量统计 (:SENSe[:POWer]:STATIStics:CLEAR)

命令格式	:SENSe[:POWer]:STATIStics:CLEAR
功能描述	清空功率计的测量统计值。
对应菜单	<code>[HOME]</code> > POWER SENSOR > 统计功能 > 清空
示例	<code>SENSe:STATIStics:CLEAR</code>

### 3.5.1.11 功率计功率控制状态 (:SENSe[:POWer]:LEV:CTL:STATe)

命令格式	:SENSe[:POWer]:LEV:CTL:STATe ON OFF 1 0 :SENSe[:POWer]:LEV:CTL:STATe?
------	--

功能描述	设置功率计测量的功率控制状态。 查询功率计测量的功率控制状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
等效命令	<code>[:SOURce]:POWER:SPC:STATe ON OFF 1 0</code> <code>[:SOURce]:POWER:SPC:STATe?</code>
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制
示例	<code>:SENSe:LEV:CTL:STATe OFF</code> <code>:SENSe:LEV:CTL:STATe?</code> 返回值: <code>0\n</code>

### 3.5.1.12 功率控制目标功率 (:SENSe[:POWER]:SPC:TARGet)

命令格式	<code>:SENSe[:POWER]:SPC:TARGet &lt;power&gt;</code> <code>:SENSe[:POWER]:SPC:TARGet?</code>
功能描述	设置功率计功率控制的目标功率。 查询功率计功率控制的目标功率。
参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	-120 dBm ~ 20 dBm
返回值	浮点型, 单位: dBm
默认值	0 dBm
等效命令	<code>[:SOURce]:POWER:SPC:TARGet &lt;power&gt;</code> <code>[:SOURce]:POWER:SPC:TARGet?</code>
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制 > 目标功率
示例	<code>SENSe:SPC:TARGet -6</code> <code>SENSe:SPC:TARGet?</code> 返回值: <code>-6\n</code>

### 3.5.1.13 功率控制限制功率 (:SENSe[:POWER]:LIMit)

命令格式	<code>:SENSe[:POWER]:LIMit &lt;power&gt;</code> <code>:SENSe[:POWER]:LIMit?</code>
功能描述	设置功率计功率控制的限制功率。 查询功率计功率控制的限制功率。

参数类型	浮点型, 单位: dBm, dBuV, uV, mV, V, nW, uW, mW 或 W, 默认为 dBm
参数范围	-120 dBm ~ 20 dBm
返回值	浮点型, 单位: dBm
默认值	0 dBm
等效命令	[[:SOURce]:POWER:LIMit <power> [:SOURce]:POWER:LIMit?
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制 > 限制功率
示例	<i>SENSe:LIMit 2</i> <i>SENSe:LIMit?</i> 返回值: <i>2\n</i>

### 3.5.1.14 功率控制捕获范围 (:SENSe[:POWER]:SPC:CRANge)

命令格式	:SENSe[:POWER]:SPC:CRANge <power> :SENSe[:POWER]:SPC:CRANge?
功能描述	设置功率计功率控制的捕获范围。 查询功率计功率控制的捕获范围。
参数类型	浮点型, 单位: dB
参数范围	0 dB ~ 50 dB
返回值	浮点型, 单位: dB
默认值	0 dB
等效命令	[[:SOURce]:POWER:SPC:CRANge <power> [:SOURce]:POWER:SPC:CRANge?
对应菜单	<b>HOME</b> > POWER SENSOR > 功率控制 > 捕获范围
示例	<i>:SENSe:SPC:CRANge 10</i> <i>:SENSe:SPC:CRANge?</i> 返回值: <i>10\n</i>

### 3.5.1.15 功率计自动调零类型 (:CALibration:ZERO:TYPE)

命令格式	:CALibration:ZERO:TYPE INTernal EXTernal :CALibration:ZERO:TYPE?
功能描述	设置功率计的自动调零类型。 查询功率计的自动调零类型。
参数类型	枚举
参数范围	INTernal EXTernal

返回值	枚举
默认值	INTernal
对应菜单	<b>HOME</b> > POWER SENSOR > 自动调零
示例	<code>:CALibration:ZERO:TYPE EXTernal</code> <code>:CALibration:ZERO:TYPE?</code> 返回值: <code>EXTernal\n</code>

### 3.5.1.16 执行清零 (:SENSe[:POWer]:ZERO)

命令格式	<code>:SENSe[:POWer]:ZERO</code>
功能描述	对功率计执行清零操作。
对应菜单	<b>HOME</b> > POWER SENSOR > 执行清零
示例	<code>:SENSe:ZERO</code>

### 3.5.1.17 功率计测量频率类型 (:SENSe[:POWer]:SOURce)

命令格式	<code>:SENSe[:POWer]:SOURce RF USER</code> <code>:SENSe[:POWer]:SOURce?</code>
功能描述	设置功率计的测量频率类型。 查询功率计的测量频率类型。
参数类型	枚举
参数范围	RF USER
返回值	枚举
默认值	RF
对应菜单	<b>HOME</b> > POWER SENSOR > 测量频率
示例	<code>SENSe:SOURce USER</code> <code>SENSe:SOURce?</code> 返回值: <code>USER\n</code>

### 3.5.1.18 功率计手动测量频率 (:SENSe[:POWer]:FREQuency)

命令格式	<code>:SENSe[:POWer]:FREQuency &lt;freq&gt;</code> <code>:SENSe[:POWer]:FREQuency?</code>
功能描述	设置功率计的手动测量频率。 查询功率计的手动测量频率。

参数类型	浮点型, 单位: Hz, kHz, MHz 或 GHz, 默认为 Hz
参数范围	功率计测量量程
返回值	浮点型, 单位: Hz
默认值	无
对应菜单	<b>HOME</b> > POWER SENSOR > 测量频率 > 手动
示例	<i>SENSe:FREQuency 1 MHz</i> <i>SENSe:FREQuency?</i> 返回值: <i>1000000\n</i>

### 3.5.1.19 功率计幅度偏移状态 (:SENSe[:POWER]:OFFSet:STATe)

命令格式	:SENSe[:POWER]:OFFSet:STATe ON OFF 1 0 :SENSe[:POWER]:OFFSet:STATe?
功能描述	设置功率计的幅度偏移状态。 查询功率计的幅度偏移状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>HOME</b> > POWER SENSOR > 幅度偏移
示例	<i>SENSe:OFFSet:STATe ON</i> <i>SENSe:OFFSet:STATe?</i> 返回值: <i>1\n</i>

### 3.5.1.20 功率计幅度偏移 (:SENSe[:POWER]:OFFSet)

命令格式	:SENSe[:POWER]:OFFSet <power> :SENSe[:POWER]:OFFSet?
功能描述	设置功率计的幅度偏移。 查询功率计的幅度偏移。
参数类型	浮点型, 单位: dB
参数范围	-200 ~ 200
返回值	浮点型, 单位: dB
默认值	0

对应菜单	<code>HOME</code> > POWER SENSOR > 幅度偏移
示例	<code>SENSe:OFFSet 10</code> <code>SENSe:OFFSet?</code> 返回值: <code>10\n</code>

### 3.5.1.21 功率计平均类型 (:SENSe[:POWER]:FILTer:TYPE)

命令格式	<code>:SENSe[:POWER]:FILTer:TYPE AUTO USER NSRatio</code> <code>:SENSe[:POWER]:FILTer:TYPE?</code>
功能描述	设置功率计的测量平均类型。 查询功率计的测量平均类型。
参数类型	枚举
参数范围	AUTO USER NSRatio
返回值	枚举
默认值	AUTO
对应菜单	<code>HOME</code> > POWER SENSOR > 平均类型
示例	<code>SENSe:FILTer:TYPE USER</code> <code>SENSe:FILTer:TYPE?</code> 返回值: <code>USER\n</code>

### 3.5.1.22 功率计平均次数 (:SENSe[:POWER]:FILTer:LENGth)

命令格式	<code>:SENSe[:POWER]:FILTer:LENGth &lt;length&gt;</code> <code>:SENSe[:POWER]:FILTer:LENGth?</code>
功能描述	设置功率计的测量平均次数。 查询功率计的测量平均次数。
参数类型	整型
参数范围	1 ~ 65536
返回值	整型
默认值	无
对应菜单	<code>HOME</code> > POWER SENSOR > 平均次数
示例	<code>SENSe:FILTer:LENGth 10</code> <code>SENSe:FILTer:LENGth?</code> 返回值: <code>10\n</code>

## 3.5.1.23 功率计固定噪声 (:SENSE[:POWER]:FILTER:NSRatio)

命令格式	:SENSE[:POWER]:FILTER:NSRatio <noise> :SENSE[:POWER]:FILTER:NSRatio?
功能描述	功率计的测量平均类型为固定噪声时，设置功率计的固定噪声。 查询功率计的固定噪声。
参数类型	浮点型，单位：dB
参数范围	无
返回值	浮点型，单位：dB
默认值	无
对应菜单	<b>HOME</b> > POWER SENSOR > 固定噪声
示例	<i>SENSE:FILTer:NSRatio 1</i> <i>SENSE:FILTer:NSRatio?</i> 返回值： <i>1\n</i>

## 3.5.1.24 功率计日志状态 (:SENSE[:POWER]:LOGGing:STATe)

命令格式	:SENSE[:POWER]:LOGGing:STATe ON OFF 1 0 :SENSE[:POWER]:LOGGing:STATe?
功能描述	设置功率计的日志记录状态。 查询功率计的日志记录状态。
参数类型	布尔型
参数范围	ON OFF 1 0
返回值	1 0
默认值	0
对应菜单	<b>HOME</b> > POWER SENSOR > 日志
示例	<i>SENSe:LOGGing:STATe ON</i> <i>SENSe:LOGGing:STATe?</i> 返回值： <i>1\n</i>

## 3.6 MEMory 命令子系统

### 3.6.1 ARB 调制设置

#### 3.6.1.1 保存波形段 (:MEMory:COPIY[:NAME])

命令格式	:MEMory:COPIY[:NAME] <"segment">,<"file_name">
功能描述	将波形段从易失性存储器保存到非易失性存储器。
参数类型	字符串, 字符串
参数范围	segment: ARB 波形段列表中的波形, file_name: 保存的波形文件。
对应菜单	<span style="border: 1px solid black; padding: 2px;">IQ</span> > ARB > 波形段 > 保存
示例	:MEMory:COPIY "SINE_WAVE","SINE_test.arb"

#### 3.6.1.2 创建波形段 (:MEM:DATA)

命令格式	:MEM:DATA <"file_name">,<data_block>
功能描述	该命令使用 <data_block> 参数将波形数据加载到信号发生器内存中, 并将波形数据保存到 <"file_name"> 变量指定的文件中。
参数类型	file_name: 字符串, 目标波形文件名称, 包括目录路径。 data_block: 字符串, 该参数代表波形数据和文件长度参数, 文件中的数据由该变量表示。
参数范围	file_name: 命名规则请参阅用户手册, data_block: 无。
对应菜单	无
示例	:MEM:DATA "IQ_Data.arb",#14Y9oL

注:

#14Y9oL - 数据块

# - 该字符表示数据块的开始

1 - 字节数中的位数

4 - 字节数

Y9oL - 4 字节波形数据 (比特数据的 ascii 表示)

### 3.6.1.3 加载用户比特数据 (:MEM:DATA:BIT)

<b>命令格式</b>	:MEM:DATA:BIT <"file_name">,<bit_count>,<data_block>
<b>功能描述</b>	该命令使用 <bit_count> 和 <data_block> 参数将比特数据加载到信号发生器内存中，并将数据保存到由 <"file_name"> 变量指定的文件中。
<b>参数类型</b>	file_name: 字符串，该变量命名目标文件和目录路径。 bit_count: 整型，该数字表示数据块中的位数。 data_block: 字符串。该参数代表数据和文件长度参数，文件中的数据由该变量表示，信号发生器使用文件长度参数来分配内存。
<b>参数范围</b>	file_name: 命名规则请参阅用户手册， bit_count: 无， data_block: 无。
<b>对应菜单</b>	无
<b>示例</b>	:MEM:DATA:BIT "Test_Data.udata",16,#12Qz 注： #12Qz - 数据块 # - 该字符表示数据块的开始 1 - 字节数中的位数 2 - 字节数 Qz - 16 位用户数据（比特数据的 ascii 表示）

## 4 编程示例

本章为程序员提供了一些示例。在这些示例中，您可以了解如何将 VISA 或 Sockets 以及上述 SCPI 命令来控制射频/射频信号发生器。通过遵循这些示例，您可以开发更多的应用程序。

### 4.1 使用 VISA 的编程示例

#### 4.1.1 VC++ 示例

**系统环境：**Windows 10, 64 位操作系统。

**编程软件：**Visual Studio

**示例内容：**使用 NI-VISA 通过 USBTMC 或 TCP/IP 访问控制设备，并执行读写操作。

请按照以下步骤完成示例：

1. 打开 Visual Studio，新建一个 VC++ win32 控制台项目。
2. 设置项目环境使用 NI-VISA 库，使用 NI-VISA 库有两种方式，静态或自动：

(1) 静态：

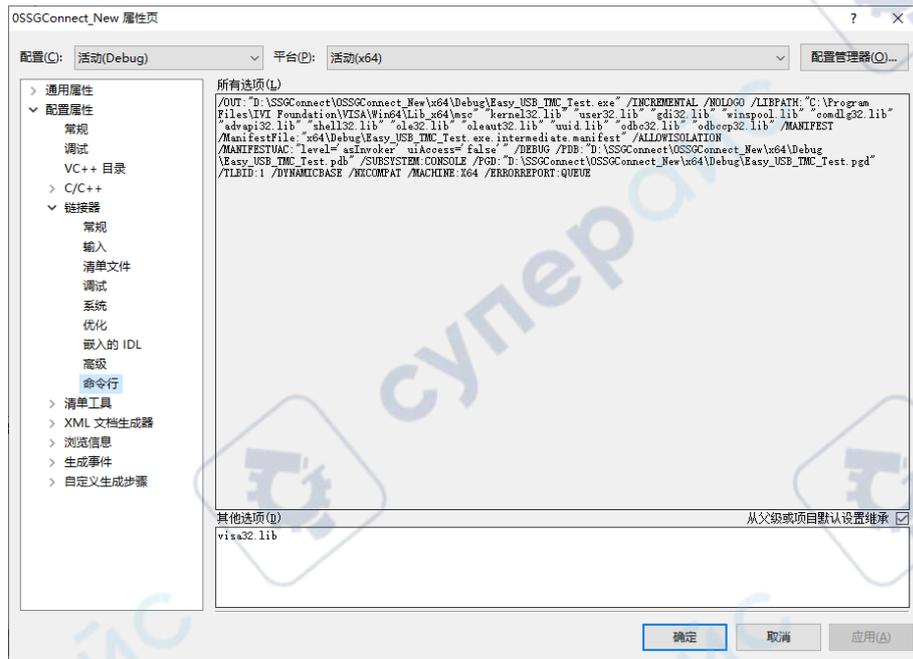
在 NI-VISA 的安装路径中查找文件：visa.h、visatype.h、visa32.lib，其中 NI-VISA 的默认安装路径为：C:\Program Files\IVI Foundation\VISA\Win64\。将上述文件复制到您的项目中，并将它们添加到项目中。然后在项目.cpp 文件中添加以下两行代码：

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

(2) 动态：

设置.h 头文件的包含目录。NI-VISA 的头文件默认安装路径为：C:\Program Files\IVI Foundation\VISA\Win64\Include。在项目 --- 属性 --- C/C++ --- 常规 --- 附加包含目录中填入头文件安装路径，如下图所示：





最后在项目.cpp 文件中引用头文件：

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

### 3. 增加代码

#### (1) USBTMC 接口访问代码：

```
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];
int i;
/*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);

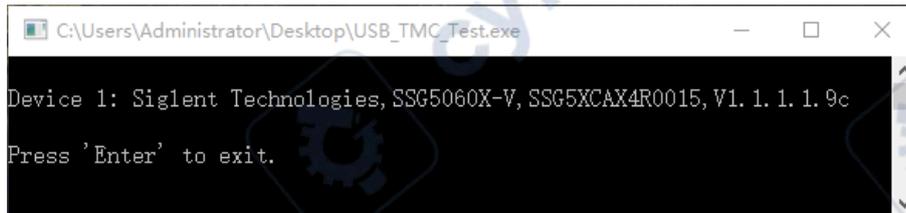
/**现在, 我们将关闭会话使用 viClose 仪器。此操作释放所有系统资源。*/
status = viClose (defaultRM);
printf("Press 'Enter' to exit.");
fflush(stdin);
getchar();
return 0;
}
```

```

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

```

运行结果：



(2) TCP/IP 接口访问代码：

```

int TCP_IP_Test(char *pIP)
{
    char outputBuffer[VI_FIND_BUFLLEN];
    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:\Users\Administrator\Desktop\TCP_IP_test.exe
Please input IP address:10.11.14.25
Messeage read from device: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
Press 'Enter' to exit.
```

## 4.1.2 VB 示例

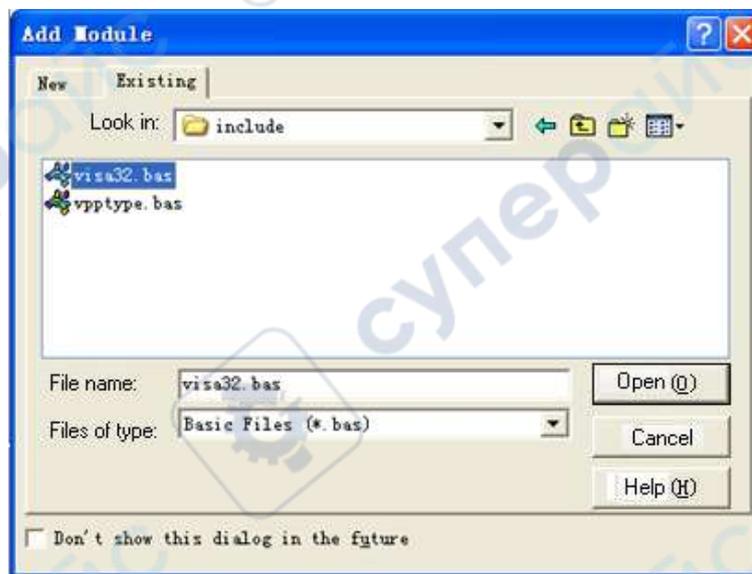
系统环境：Windows 7

编程软件：Microsoft Visual Basic 6.0

示例内容：使用 NI-VISA 通过 USBTMC 或 TCP/IP 访问控制设备，并执行读写操作。

请按照以下步骤完成示例：

1. 打开 Visual Basic，建立一个标准的应用程序项目（标准 EXE）。
2. 设置项目环境中使用 NI-VISA 库。单击项目的现有标签>>添加模块，在 NI-VISA 安装路径下的 include 文件夹中搜索 visa32.bas 文件并添加该文件。这将使得 VISA 功能和 VISA 的数据类型在程序中使用。



3. 添加代码

- (1) USBTMC 接口访问代码

`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 接口访问代码

```
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

```

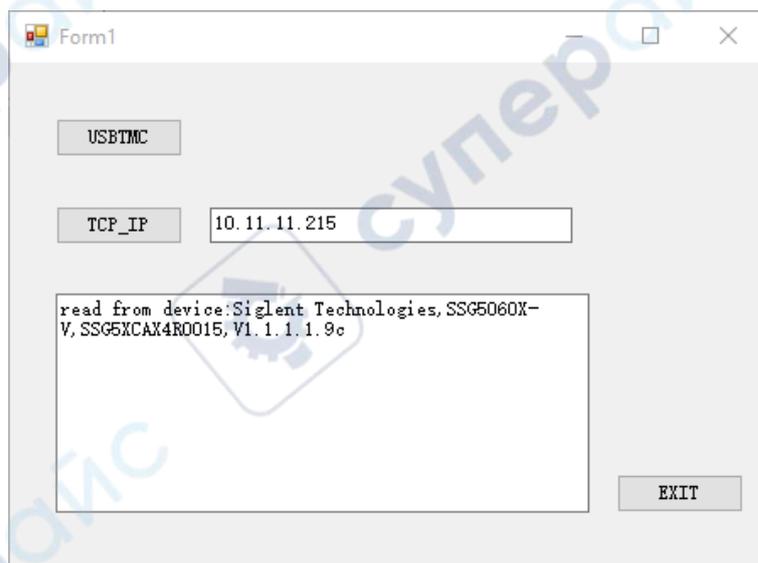
Private Sub exitBtn_Click()
    End
End Sub

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

```

运行结果：



### 4.1.3 MATLAB 示例

系统环境：Windows 7

编程软件：MATLAB R2013a

示例内容：使用 NI-VISA 通过 USBTMC 或 TCP/IP 访问控制设备，并执行读写操作。

请按照以下步骤完成示例：

1. 打开 MATLAB，修改当前目录。在本例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 接口访问代码

```
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::0x1501::0123456789::INSTR');

%打开创建的 VISA 对象
fopen(vu);

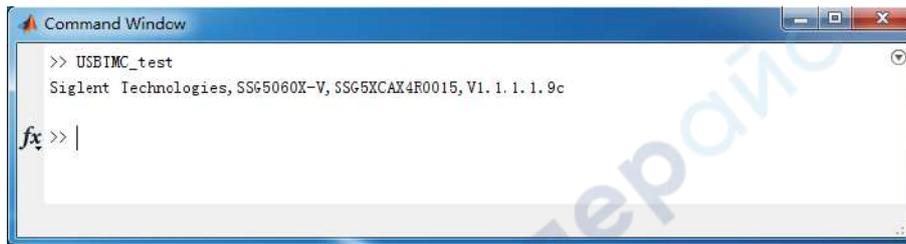
%发送字符串"* IDN? "，查询设备信息。
fprintf(vu, '*IDN?');

%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
>> USBIMC_test
Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
fx >> |
```

(2) TCP/IP 接口访问代码

```
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-TCP/IP 对象连接到配置了 IP 地址的仪器。

```
vt = visa('ni', ['TCP/IP0::', 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

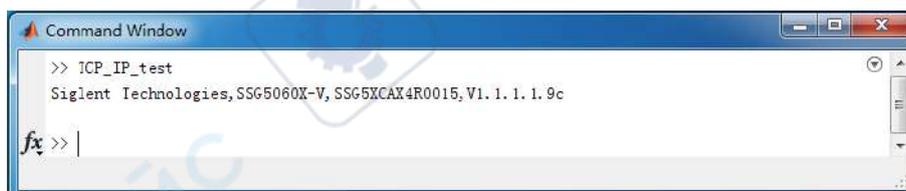
```
outputbuffer = fscanf(vt);
disp(outputbuffer);
```

%关闭 VISA 对象

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

```
end
```

运行结果：



```
Command Window
>> TCP_IP_test
Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
fx >> |
```

#### 4.1.4 LabVIEW 示例

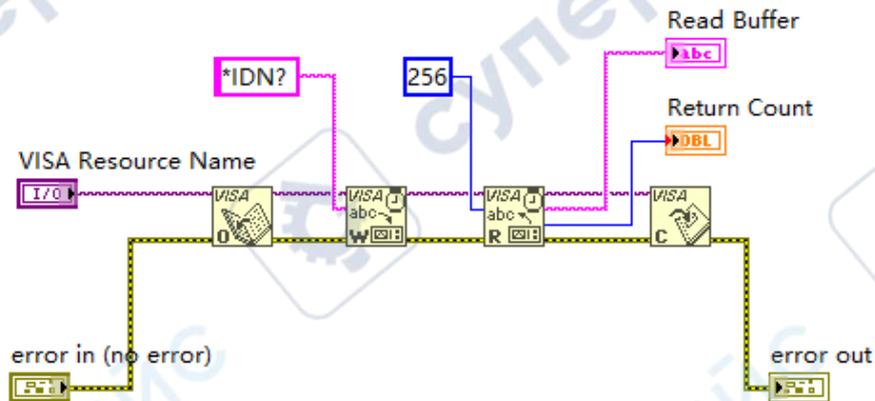
系统环境：Windows 7

编程软件：LabVIEW 2011

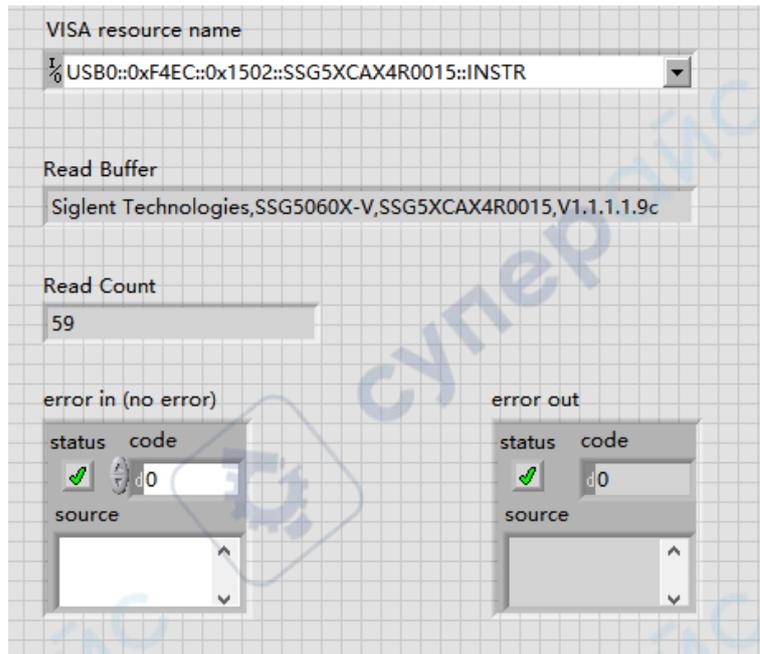
示例内容：使用 NI-VISA 通过 USBTMC 或 TCP/IP 访问控制设备，并执行读写操作。

请按照以下步骤完成示例：

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：



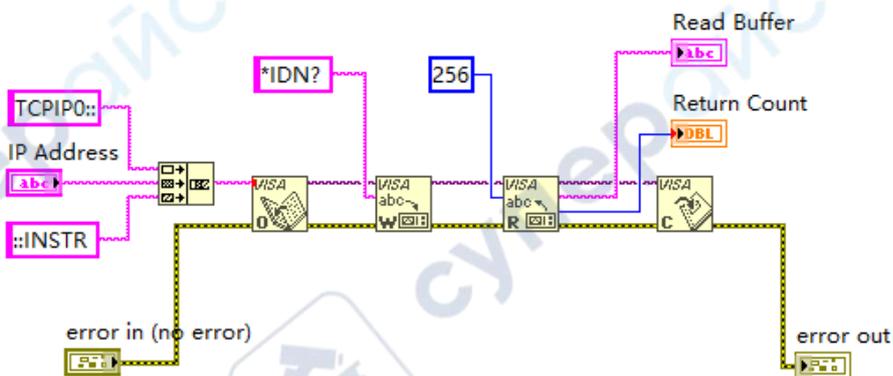
5. 从 VISA 资源名称列表框中选择设备资源然后运行程序。



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。

The screenshot displays a network configuration window with a grid background. It contains the following fields and sections:

- IP address:** A text box containing the value "10.11.14.25".
- Read Buffer:** A text box containing the string "Siglent Technologies,SSG5060X-V,SSG5XCAX4R0015,V1.1.1.1.9c".
- Read Count:** A text box containing the value "59".
- error in (no error):** A section containing a table with columns "status" and "code". The "status" column has a green checkmark icon, and the "code" column has the value "d0". Below this is a "source" label and an empty dropdown menu.
- error out:** A section containing a table with columns "status" and "code". The "status" column has a green checkmark icon, and the "code" column has the value "d0". Below this is a "source" label and an empty dropdown menu.

## 4.2 使用 Socket 的编程示例

Windows 操作系统自身支持 Sockets 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可移植。Python 有一个底层网络模块，可以提供对 Socket 接口的访问，端口为 5025。可以为 Socket 接口编写 Python 脚本来执行各种测试和测量任务。

**系统环境：** Windows 10，64 位操作系统。

**编程软件：** Python v3.6.5

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

脚本内容：

```
#!/usr/bin/env python
#-*- coding:utf-8 -*-
#-----
# The short script is an 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 = "10.11.22.27" # should match the instrument's IP address
port = 5025 # the port number of the instrument service

def SocketConnect():
    try:
        #create an AF_INET, STREAM socket (TCP)
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    except socket.error:
        input("Failed to create socket. \nPress "Enter" to exit: ")
        sys.exit()
    try:
        #Connect to remote server
        s.connect((remote_ip , port))
    except socket.error:
        input("Failed to connect to ip %s!\nPress "Enter" to exit: ' % remote_ip)
    s.close()
```

```
sys.exit()
    return s

def SocketQuery(Sock, cmd):
    try :
        #Send cmd string
        Sock.sendall(cmd)
        time.sleep(1)
    except socket.error:
        #Send failed
        input('Send failed!\nPress "Enter" to exit: ')
    SocketClose(Sock)
    sys.exit()
reply = Sock.recv(4096)
reply = reply.decode()
    return reply

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

def main():
    # Body: send the SCPI commands *IDN? 10 times and print the return message
    s = SocketConnect()

    count = 0
    for i in range(10):
        qStr = SocketQuery(s, b'*IDN?\n')
        print (str(count) + ":: " + qStr)
        count = count + 1
    SocketClose(s)
    input('Press "Enter" to exit')

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

运行结果：

```
C:\Python27\python2.exe
0:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
1:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
2:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
3:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
4:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
5:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
6:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
7:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
8:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
9:: Siglent Technologies, SSG5060X-V, SSG5XCAX4R0015, V1.1.1.1.9c
Press "Enter" to exit
```

## 关于鼎阳

鼎阳科技（SIGLENT）是通用电子测试测量仪器领域的行业领军企业，A 股上市公司。

2002 年，鼎阳科技创始人开始专注于示波器研发，2005 年成功研制出鼎阳第一款数字示波器。历经多年发展，鼎阳产品已扩展到数字示波器、手持示波表、函数/任意波形发生器、频谱分析仪、矢量网络分析仪、射频/微波信号源、台式万用表、直流电源、电子负载等基础测试测量仪器产品，是全球极少数能够同时研发、生产、销售数字示波器、信号发生器、频谱分析仪和矢量网络分析仪四大通用电子测试测量仪器主力产品的厂家之一，国家重点“小巨人”企业。同时也是国内主要竞争对手中极少数同时拥有这四大主力产品并且四大主力产品全线进入高端领域的厂家。公司总部位于深圳，在美国克利夫兰、德国奥格斯堡、日本东京成立了子公司，在成都成立了分公司，产品远销全球 80 多个国家和地区，SIGLENT 已经成为全球知名的测试测量仪器品牌。

## 联系我们

深圳市鼎阳科技股份有限公司  
全国免费服务热线：400-878-0807  
网址：[www.siglent.com](http://www.siglent.com)

## 声明

 SIGLENT® 鼎阳 是深圳市鼎阳科技股份有限公司的注册商标，事先未经过允许，不得以任何形式或通过任何方式复制本手册中的任何内容。本资料中的信息代替原先的此前所有版本。技术数据如有变更，恕不另行通告。

## 技术许可

对于本文中描述的硬件和软件，仅在得到许可的情况下才会提供，并且只能根据许可进行使用或复制。

