

# UNI-T

## 编程手册

### UTG9000T系列可编程信号源

2020-01-06

UNI-T TECHNOLOGIES, INC.



---

## 保证和声明

### 版权

2017 优利德中国科技有限公司

### 商标信息

**UNI-T**是优利德中国科技有限公司的注册商标。

### 文档编号

### 软件版本

00.00.01

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

### 声明

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

### 产品认证

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

### 联系我们

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

电子邮箱：

网址：

## SCPI 指令简介

SCPI (Standard Commands for Programmable Instruments, 即可编程仪器标准命令集) 是一种建立在现有标准 IEEE 488.1 和 IEEE 488.2 基础上, 并遵循了 IEEE754 标准中浮点运算规则、ISO646 信息交换 7 位编码符号(相当于 ASCII 编程)等多种标准的标准化仪器编程语言。本节简介 SCPI 命令的格式、符号、参数和缩写规则。

### 指令格式

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。命令关键字和第一个参数之间以空格分开。命令字符串必须以一个 <换行> (<NL>) 字符结尾。命令行后面添加问号“?”通常表示对此功能进行查询。

### 符号说明

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

- **大括号 { }**  
大括号中通常包含多个可选参数, 发送命令时必须选择其中一个参数。  
如: DISPLAY:GRID:MODE { FULL | GRID | CROSS | NONE } 命令。
- **竖线 |**  
竖线用于分隔多个参数选项, 发送命令时必须选择其中一个参数。  
如: DISPLAY:GRID:MODE { FULL | GRID | CROSS | NONE } 命令。
- **方括号 [ ]**  
方括号中的内容(命令关键字)是可省略的。如果省略参数, 仪器将该参数设置为默认值。  
例如: 对于:MEASURE:NDUTY? [<source>] 命令, [<source>] 表示当前通道。
- **三角括号 < >**  
三角括号中的参数必须用一个有效值来替换。例如: 以 DISPLAY:GRID:BRIGHTNESS 30 的形式发送 DISPLAY:GRID:BRIGHTNESS <count> 命令。

## 参数说明

本手册介绍的命令中所含的参数可以分为以下 5 种类型：布尔型、整型、实型、离散型、ASCII 字符串。

- **布尔型**

参数取值为“ON”(1) 或“OFF”(0)。例如：:SYSTem:LOCK {{1 | ON} | {0 | OFF}}。

- **整型**

除非另有说明，参数在有效值范围内可以取任意整数。注意：此时，请不要设置参数为小数格式，否则将出现异常。例如：:DISPlay:GRID:BRIGhtness <count>命令中的参数< count > 可取 0 到 100 范围内的任一整数。

- **实型**

除非另有说明，参数在有效值范围内可以取任意值。

例如：对于 CH1，CHANnel:OFFSet <offset>命令中的参数<offset>的取值为实型。

- **离散型**

参数只能取指定的几个数值或字符。例如：:DISPlay:GRID:MODE { FULL | GRID | CROSS | NONE}命令的参数只能为 FULL、GRID、CROSS、NONE。

- **ASCII 字符串**

字符串参数实际上可包含所有 ASCII 字符集。字符串必须以配对的引号开始和结尾；可以用单引号或双引号。引号分隔符也可以作为字符串的一部分，只需键入两次并且不在中间添加任何字符，例如设置IP：SYST:COMM:LAN:IPAD "192.168.1.10"。

## 简写规则

所有命令对大小写都能识别，可以全部采用大写或小写。如果要缩写，必须输完命令格式中的所有大写字母。

## 数据返回

数据返回分为单个数据和批量数据返回，单个数据返回相对应的参数类型，其中实型返回用科学计数法表示，e 前部分小数点后面保留三位数据，e 部分保留三位数据；批量数据返回必须符合 IEEE 488.2 #格式的字符串数据，其格式：'#' + 长度所占的字符位数[固定为一个字符] + 有效数据长度的 ASCII 值 + 有效数据 + 结束符['\n']，例如#3123xxxxxxxxxxxxxxxxxxxx\n 表示的具有 123 个字节有效批量数据返回格式，其中 '3' 表示“123”占 3 个字符位。

---

# SCPI 指令详解

## IEEE488.2 通用命令

### \*IDN?

➤ **命令格式:**

\*IDN?

➤ **功能描述:**

用于查询制造商名称、电源型号、产品序列号和软件版本号。

➤ **返回格式:**

制造商名称，电源型号，产品序列号，由点号分隔的软件版本号。

**注意:** 返回的型号要与铭牌信息一致。

➤ **举例:**

UNI-T Technologies, UTG5000, 000000001, 00.00.01

### \*RST

➤ **命令格式:**

\*RST

➤ **功能描述:**

用于恢复出厂设置并清空所有的错误信息及发送接收队列缓冲。

## SYSTem 命令

用于对信号源进行最基本的操作，主要包括全键盘锁定、系统设置数据的操作。

### :SYSTem:INFo?

➤ **命令格式:**

:SYSTem:INFo?

➤ **功能描述:**

用于查询内置的通讯协议版本号等相关系统信息。

➤ **返回格式:**

{协议版本号},{软件版本号},{硬件版本号},{仪器类型},{机型内部代号},{带宽},{采样率},{通道数量},{显示名称},{渠道信息},{序列号}。

说明：用逗号隔开，按位置填充指定的信息。

➤ **举例:**

:SYSTem:INFo? 1,1.02.3,1.00.1,DSO,BG,100MHz,1GSa/s,2CH,UTD7000BG,\*\*,12345678

返回格式详细说明:

◇ **1: 协议版本号**

内置的版本号，每次修改都要递增，维护一个整数值而不是传统的 1.02.3 的模式，这样管理系统就可以通过判断这个版本号，来确定是否要提醒用户升级固件或者提示某些功能不可用。

◇ **1.02.3: 软件版本号**

◇ **1.00.1: 硬件版本号**

◇ **DSO: 仪器类别**

示例中：DSO 代表示波器

◇ **BG: 内置机型代号，每一种系列机型都有一个固定的机型代码，以便上位机区分。**

示例中：BG 就代表 UTD7000BG，UPO0001 就代表 UPO3000CS 系列，UPO0002 就代表新的 UPO2000CS 系列

◇ **100MHz: 带宽**

示例中：100MHz

◇ **1GSa/s : 采样率**

示例中：1GS/s

◇ **2CH: 通道数量**

示例中：2 通道

◇ **UTG7000BG: 显示名称**

示例中：UTG7000BG

◇ **\*\* : 渠道信息**

示例中：\*\*表示经外销，EN 表示教育渠道

◇ **12345678: 序列号**

示例中：12345678

**:SYSTem:LOCK**➤ **命令格式:**

```
:SYSTem:LOCK {{1 | ON} | {0 | OFF}}
```

```
:SYSTem:LOCK?
```

➤ **功能描述:**

用于锁定或者解锁全键盘按键。

➤ **返回格式:**

查询返回全键盘锁定状态，0 表示未锁定，1 表示锁定。

➤ **举例:**

```
:SYSTem:LOCK ON           全键盘锁定
:SYSTem:LOCK OFF         全键盘解锁
:SYSTem:LOCK?            查询返回 1，表示锁定
```

**:SYSTem:CONFigure**➤ **命令格式:**

```
:SYSTem:CONFigure <file>
```

```
:SYSTem:CONFigure?
```

➤ **功能描述:**

用于读写配置文件，先发送该指令，然后发送配置文件数据到信号源。

<file>表示配置文件。

➤ **返回格式:**

查询返回信号源当前配置文件数据。

➤ **举例:**

```
:SYSTem:CONFigure          写入配置文件数据到信号源中并使其加载
:SYSTem:CONFigure?        查询返回信号源当前配置文件数据二进制流
```

**:SYSTem:PHASe:MODE**➤ **命令格式:**

```
:SYSTem:PHASe:MODE {INDePendent | SYNChronization}
```

```
:SYSTem:PHASe:MODE?
```

➤ **功能描述:**

控制通道间的相位模式，若为同步，则表示两个通道起始相位保持同步，否则相位独立。

➤ **返回格式:**

查询返回通道间的相位模式。

➤ **举例:**

```
:SYSTem:PHASe:MODE INDePendent  设置通道间为独立相位模式
:SYSTem:PHASe:MODE?            查询返回 INDePendent
```



## **:SYSTem:LANGUage**

➤ **命令格式:**

:SYSTem:LANGUage {ENGLish|CHINese}

:SYSTem:LANGUage?

➤ **功能描述:**

控制系统显示语言。

➤ **返回格式:**

查询返回系统显示语言。

➤ **举例:**

:SYSTem:LANGUage ENGLish

设置英文为系统显示语言

:SYSTem:LANGUage?

查询返回 ENGLish

## **:SYSTem:BEEP**

➤ **命令格式:**

:SYSTem:BEEP {{1 | ON} | {0 | OFF}}

:SYSTem:BEEP?

➤ **功能描述:**

控制系统蜂鸣器开关

➤ **返回格式:**

查询返回蜂鸣器开关状态。

➤ **举例:**

:SYSTem:BEEP ON

打开蜂鸣器

:SYSTem:BEEP?

查询返回 1

## **:SYSTem:NUMBer:FORMat**

➤ **命令格式:**

:SYSTem:NUMBer:FORMat {COMMA|SPACE|NONE}

:SYSTem:NUMBer:FORMat?

➤ **功能描述:**

控制系统数字格式的分隔符

➤ **返回格式:**

查询返回系统数字格式的分隔符。

➤ **举例:**

:SYSTem:NUMBer:FORMat NONE

设置无系统数字格式

:SYSTem:NUMBer:FORMat?

查询返回 NONE

---

**:SYSTem:BRIGhtness**➤ **命令格式:**

```
:SYSTem:BRIGhtness { 30|40|50|60|70|80|90|100}
```

```
:SYSTem:BRIGhtness?
```

➤ **功能描述:**

控制系统背光亮度等级

➤ **返回格式:**

查询返回系统背光亮度等级

➤ **举例:**

```
:SYSTem:BRIGhtness 30          设置系统背光亮度 30%
```

```
:SYSTem:BRIGhtness?          查询返回 30
```

**:SYSTem:SLEEP:TIME**➤ **命令格式:**

```
:SYSTem:SLEEP:TIME { CLOSe | 1 | 5 | 15 | 30 | 60}
```

```
:SYSTem:SLEEP:TIME?
```

➤ **功能描述:**

控制系统休眠时间，单位是分钟

➤ **返回格式:**

查询返回休眠时间

➤ **举例:**

```
:SYSTem:SLEEP:TIME 1          设置系统 1 分钟之后自动休眠
```

```
:SYSTem:SLEEP:TIME?          查询返回 1
```

**:SYSTem:CYMometer**➤ **命令格式:**

```
:SYSTem:CYMometer {{1 | ON} | {0 | OFF}}
```

```
:SYSTem:CYMometer?
```

➤ **功能描述:**

控制系统频率计开关状态。

注意：打开该功能会关闭通道的同步输出。

➤ **返回格式:**

查询返回系统频率计开关状态，0 表示关闭，1 表示打开。

➤ **举例:**

```
:SYSTem:CYMometer ON          打开系统频率计
```

```
:SYSTem:CYMometer?          查询返回 1
```

---

**:SYSTem:CYMometer:FREQuency?**

- **命令格式:**  
:SYSTem:CYMometer:FREQuency?
- **功能描述:**  
获取频率计的当前测量的频率。
- **返回格式:**  
查询返回获取频率计的当前测量的频率，单位 Hz，采样科学计数法返回数据。
- **举例:**  
:SYSTem:CYMometer:FREQuency?            查询返回 2e+3

**:SYSTem:CYMometer:PERiod?**

- **命令格式:**  
:SYSTem:CYMometer:PERiod?
- **功能描述:**  
获取频率计的当前测量的周期。
- **返回格式:**  
查询返回获取频率计的当前测量的周期，单位 S，采样科学计数法返回数据。
- **举例:**  
:SYSTem:CYMometer:PERiod?            查询返回 2e-3

**:SYSTem:CYMometer:DUTY?**

- **命令格式:**  
:SYSTem:CYMometer:DUTY?
- **功能描述:**  
获取频率计的当前测量的占空比。
- **返回格式:**  
查询返回获取频率计的当前测量的占空比，单位%。
- **举例:**  
:SYSTem:CYMometer:DUTY?            查询返回 20，表示占空比 20%

**:SYSTem:CYMometer:PWIDTh?**

- **命令格式:**  
:SYSTem:CYMometer:PWIDTh?
- **功能描述:**  
获取频率计的当前测量的正脉宽宽度。
- **返回格式:**  
查询返回获取频率计的当前测量的正脉宽宽度，单位 s。
- **举例:**

:SYSTem:CYMometer: PWIDTH?

查询返回 1e-3, 表示占空比 1 毫秒。

### **:SYSTem:CYMometer:NWIDTH?**

➤ **命令格式:**

:SYSTem:CYMometer: NWIDTH?

➤ **功能描述:**

获取频率计的当前测量的负脉宽宽度。

➤ **返回格式:**

查询返回获取频率计的当前测量的负脉宽宽度, 单位 s。

➤ **举例:**

:SYSTem:CYMometer: NWIDTH?

查询返回 1e-3, 表示占空比 1 毫秒

## CHANnel 命令

用于设置信号源通道相关功能。

### :CHANnel<n>:MODE

➤ **命令格式:**

```
:CHANnel<n>:MODE {CONTInue | MODulation| SWEep| BURSt }
:CHANnel<n>:MODE?
```

➤ **功能描述:**

设置指定通道信号模式，分别为 CONTInue、MODulation、SWEep、BURSt。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道信号模式。

➤ **举例:**

```
:CHANnel1:MODE AM          设置通道 1 信号幅度调制模式输出
:CHANnel1:MODE?           查询返回 AM
```

### :CHANnel<n>:OUTPut

➤ **命令格式:**

```
:CHANnel<n>:OUTPut {{1 | ON} | {0 | OFF}}
:CHANnel<n>:OUTPut?
```

➤ **功能描述:**

设置打开或关闭指定通道的输出。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道的输出状态，0 表示关闭，1 表示打开。

➤ **举例:**

```
:CHANnel1:OUTPut ON        设置打开通道 1 输出
:CHANnel1:OUTPut?         查询返回 1
```

### :CHANnel<n>:INVersion

➤ **命令格式:**

```
:CHANnel<n>:INVersion {{1 | ON} | {0 | OFF}}
:CHANnel<n>:INVersion?
```

➤ **功能描述:**

设置打开或关闭指定通道反向。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道的反向状态，0 表示关闭，1 表示打开。

➤ **举例:**

:CHANnel1:INVersion ON                    设置打开通道 1 反向输出  
:CHANnel1:INVersion?                    查询返回 1

### **:CHANnel<n>:OUTPut:SYNC**

➤ **命令格式:**

:CHANnel<n>:OUTPut:SYNC {{1 | ON} | {0 | OFF}}

:CHANnel<n>:OUTPut:SYNC?

➤ **功能描述:**

设置通道同步输出状态。

注意：设备只有一个同步输出接口，同时只能打开一个通道的同步输出。

<n>: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道的同步输出状态，0 表示关闭，1 表示打开。

➤ **举例:**

:CHANnel1:OUTPut:SYNC ON                    设置打开通道 1 同步输出

:CHANnel1:OUTPut:SYNC?                    查询返回 1

### **:CHANnel<n>:OUTPut:SYNC:INVersion**

➤ **命令格式:**

:CHANnel<n>:OUTPut:SYNC:INVersion {{1 | ON} | {0 | OFF}}

:CHANnel<n>:OUTPut:SYNC:INVersion?

➤ **功能描述:**

设置通道同步输出反相。

注意：设备只有一个同步输出接口，同时只能打开一个通道的同步输出。

<n>: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道的反向状态，0 表示关闭，1 表示打开。

➤ **举例:**

:CHANnel1:OUTPut:SYNC:INVersion ON                    设置通道 1 同步输出反相

:CHANnel1:OUTPut:SYNC:INVersion?                    查询返回 1

### **:CHANnel<n>:LIMit:ENABLE**

➤ **命令格式:**

:CHANnel<n>:LIMit:ENABLE {{1 | ON} | {0 | OFF}}

:CHANnel<n>:LIMit:ENABLE?

➤ **功能描述:**

设置指定通道限幅开关。

<n>: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道的限幅状态。

➤ **举例:**

:CHANnel1:LIMit:ENABle ON                    设置打开通道 1 限幅  
:CHANnel1:LIMit:ENABle?                    查询返回 1

### **:CHANnel<n>:LIMit:LOWer**

- **命令格式:**  
:CHANnel<n>:LIMit:LOWer {<voltage>}  
:CHANnel<n>:LIMit:LOWer?
- **功能描述:**  
设置指定通道限幅下限值。  
<voltage>表示电压，单位当前通道指定单位。  
<n>: 通道号，n 取值 1、2，3，4。
- **返回格式:**  
查询返回指定通道的限幅下限值，采用科学计数法返回。
- **举例:**  
:CHANnel1:LIMit:LOWer 2                    设置通道 1 限幅下限 2V  
:CHANnel1:LIMit:LOWer?                    查询返回 2e+0

### **:CHANnel<n>:LIMit:UPPer**

- **命令格式:**  
:CHANnel<n>:LIMit:UPPer {<voltage>}  
:CHANnel<n>:LIMit:UPPer?
- **功能描述:**  
设置指定通道限幅上限值。  
<voltage>表示电压，单位当前通道指定单位。  
<n>: 通道号，n 取值 1、2，3，4。
- **返回格式:**  
查询返回指定通道的限幅上限值，采用科学计数法返回。
- **举例:**  
:CHANnel1:LIMit:UPPer 2                    设置通道 1 限幅上限 2V  
:CHANnel1:LIMit:UPPer?                    查询返回 2e+0

### **:CHANnel<n>:AMPLitude:UNIT**

- **命令格式:**  
:CHANnel<n>:AMPLitude:UNIT {VPP | VRMS | DBM}  
:CHANnel<n>:AMPLitude:UNIT?
- **功能描述:**  
设置指定通道输出幅度单位。  
<n>: 通道号，n 取值 1、2，3，4。
- **返回格式:**  
查询返回指定通道的输出幅度单位。
- **举例:**

:CHANnel1:AMPLitude:UNIT VPP                    设置通道 1 输出幅度单位为 VPP  
:CHANnel1:AMPLitude:UNIT?                    查询返回 VPP

### **:CHANnel<n>:LOAD**

➤ **命令格式:**

:CHANnel<n>:LOAD <resistance>  
:CHANnel<n>:LOAD?

➤ **功能描述:**

设置指定通道输出负载。

<resistance>表示负载电阻值，单位为  $\Omega$

<n>: 通道号，n 取值 1、2、3、4。

注意：阻值取值范围为 1~10000,其中 10000 对应于高阻。

➤ **返回格式:**

查询返回指定通道的负载阻值，采用科学计数法返回。

➤ **举例:**

:CHANnel1:LOAD 50                    设置通道 1 输出负载 50  $\Omega$   
:CHANnel1:LOAD?                    查询返回 50e+0

### **:CHANnel<n>:PNCode**

➤ **命令格式:**

:CHANnel<n>:PNCode <code>  
:CHANnel<n>:PNCode?

➤ **功能描述:**

设置指定通道 PN 码，该指令只对具备 PN 码功能的波形有效。

<code>: 表示 PN 码，如下所示:

{PN3|PN5|PN7|PN9|PN11|PN13|PN15|PN17|PN21|PN23|PN25|PN27|PN29|PN31|PN33}

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道 PN 码。

➤ **举例:**

:CHANnel1:PNCode PN9                    设置通道 1 PN 码为 PN9  
:CHANnel1:PNCode?                    查询返回 PN9

### **:CHANnel<n>:TRIGger:SOURce**

➤ **命令格式:**

:CHANnel<n>:TRIGger:SOURce {INTernal|EXTRise|EXTFall|MANUal}  
:CHANnel<n>:TRIGger:SOURce?

➤ **功能描述:**

设置指定通道触发源，该指令只对扫频和猝发功能有效。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**



查询返回指定通道触发源。

➤ **举例：**

:CHANnel1:TRIGger:SOURce INTernal      设置通道 1 内部触发源  
:CHANnel1:TRIGger:SOURce?              查询返回 INTernal

**:CHANnel<n>:TRIGger:OUTPut**

➤ **命令格式：**

:CHANnel<n>:TRIGger:OUTPut {CLOSe|RISe|FALL}  
:CHANnel<n>:TRIGger:OUTPut?

➤ **功能描述：**

设置指定通道触发输出模式，该指令只对扫频和猝发功能有效。

<n>：通道号，n 取值 1、2，3，4。

➤ **返回格式：**

查询返回指定通道触发输出模式。

➤ **举例：**

:CHANnel1:TRIGger:OUTPut RISe      设置通道 1 上升沿触发输出模式  
:CHANnel1:TRIGger:OUTPut?              查询返回 RISe

**:CHANnel<n>:NS**

➤ **命令格式：**

:CHANnel<n>:NS {{1|ON}|{0|OFF}}  
:CHANnel<n>:NS?

➤ **功能描述：**

设置打开或关闭指定通道噪声叠加，该指令只对具备噪声叠加功能的波形有效。

<n>：通道号，n 取值 1、2，3，4。

➤ **返回格式：**

查询返回指定通道的输出状态，0 表示关闭，1 表示打开。

➤ **举例：**

:CHANnel1:NS ON                      打开通道 1 噪声叠加  
:CHANnel1:NS?                      查询返回 1

**:CHANnel<n>:NS:SNR:UNIT**

➤ **命令格式：**

:CHANnel<n>:NS:SNR:UNIT {DBM|RATIO}  
:CHANnel<n>:NS:SNR:UNIT?

➤ **功能描述：**

设置通道噪声叠加信噪比单位，该指令只对具备噪声叠加功能的波形有效。

<n>：通道号，n 取值 1、2，3，4。

➤ **返回格式：**

查询返回指定通道噪声叠加信噪比单位。

➤ **举例：**

:CHANnel1:NS:SNR:UNIT DBM                    设置通道 1 噪声叠加信噪比单位 dB  
:CHANnel1:NS:SNR:UNIT?                    查询返回 DBM

### :CHANnel<n>:NS:SNR

➤ **命令格式:**

:CHANnel<n>:NS:SNR <value>

:CHANnel<n>:NS:SNR?

➤ **功能描述:**

设置通道噪声叠加信噪比值，该指令只对具备噪声叠加功能的波形有效。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道的噪声叠加信噪比值，采用科学计数法返回。

➤ **举例:**

:CHANnel1:NS:SNR 3

设置通道 1 噪声叠加信噪比 3dB

:CHANnel1:NS:SNR?

查询返回 3e+0

### :CHANnel<n>:MERge

➤ **命令格式:**

:CHANnel<n>:MERge {{1 | ON} | {0 | OFF}}

:CHANnel<n>:MERge?

➤ **功能描述:**

设置指定通道合并输出开关。

1、2 通道合并只能从 1 和 2 通道输出。

3、4 通道合并只能从 3 和 4 通道输出。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道合并开合状态。

➤ **举例:**

:CHANnel1:MERge ON

1 和 2 通道合并信号且信号从 1 通道输出

:CHANnel1:MERge?

返回 1

### :CHANnel:COUPle<m>:FREQuency

➤ **命令格式:**

:CHANnel:COUPle<m>:FREQuency {{1 | ON} | {0 | OFF}}

:CHANnel:COUPle<m>:FREQuency?

➤ **功能描述:**

设置通道频率耦合开关，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

- **返回格式:**  
查询返回通道耦合开关状态, 0 表示关闭, 1 表示开启。

- **举例:**  
:CHANnel:COUPle1:FREQuency ON                    开启 1 和 2 通道耦合  
:CHANnel:COUPle1:FREQuency?                    返回 1

### **:CHANnel:COUPle<m>:FREQuency:SCALE**

- **命令格式:**  
:CHANnel:COUPle<m>:FREQuency:SCALE <scale>  
:CHANnel:COUPle<m>:FREQuency:SCALE?

- **功能描述:**  
设置通道耦合频率比例, 通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。  
<scale >: 耦合频率比例。  
<m>: 通道号, m 取值 1、2。  
1 表示一和二通道耦合; 2 表示三和四通道耦合

- **返回格式:**  
查询返回通道耦合频率比例, 返回科学计数法。

- **举例:**  
:CHANnel:COUPle1:FREQuency:SCALE 0.1            设置 2 通道比 1 通道耦合比例为 0.1  
:CHANnel:COUPle1:FREQuency:SCALE?            返回 1e-1

### **:CHANnel:COUPle<m>:FREQuency:DEV**

- **命令格式:**  
:CHANnel:COUPle<m>:FREQuency:DEV <dev >  
:CHANnel:COUPle<m>:FREQuency:DEV?

- **功能描述:**  
设置通道耦合频率偏差, 通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。  
<scale >: 耦合频率偏差, 单位 Hz。  
<m>: 通道号, m 取值 1、2。  
1 表示一和二通道耦合; 2 表示三和四通道耦合

- **返回格式:**  
查询返回通道耦合频率偏差, 返回科学计数法。

- **举例:**  
:CHANnel:COUPle1:FREQuency:DEV 100            设置 2 通道比 1 通道耦合偏差为 100Hz  
:CHANnel:COUPle1:FREQuency:DEV?            返回 1e+2

## **:CHANnel:COUPle<m>:PHASe**

➤ **命令格式:**

:CHANnel:COUPle<m>:PHASe {{1|ON}|{0|OFF}}

:CHANnel:COUPle<m>:PHASe?

➤ **功能描述:**

设置通道相位耦合开关，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

➤ **返回格式:**

查询返回通道耦合开关状态，0 表示关闭，1 表示开启。

➤ **举例:**

:CHANnel:COUPle1:PHASe ON                    开启 1 和 2 通道耦合

:CHANnel:COUPle1:PHASe?                    返回 1

## **:CHANnel:COUPle<m>:PHASe:SCALE**

➤ **命令格式:**

:CHANnel:COUPle<m>:PHASe:SCALE <scale>

:CHANnel:COUPle<m>:PHASe:SCALE?

➤ **功能描述:**

设置通道耦合相位比例，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<scale>: 耦合相位比例。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

➤ **返回格式:**

查询返回通道耦合相位比例，返回科学计数法。

➤ **举例:**

:CHANnel:COUPle1:PHASe:SCALE 0.1        设置 2 通道比 1 通道耦合比例为 0.1

:CHANnel:COUPle1:PHASe:SCALE?        返回 1e-1

## **:CHANnel:COUPle<m>:PHASe:DEV**

➤ **命令格式:**

:CHANnel:COUPle<m>:PHASe:DEV <dev >

:CHANnel:COUPle<m>:PHASe:DEV?

➤ **功能描述:**

设置通道耦合相位偏差，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<scale>: 耦合相位偏差，单位°。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

➤ **返回格式:**

查询返回通道耦合相位偏差，返回科学计数法。

➤ **举例:**

:CHANnel:COUPle1:PHASe:DEV 100            设置 2 通道比 1 通道耦合偏差为 100°  
:CHANnel:COUPle1:PHASe:DEV?            返回 1e+2

**:CHANnel:COUPle<m>:AMPLitude**

➤ **命令格式:**

:CHANnel:COUPle<m>:AMPLitude {{1|ON}|{0|OFF}}  
:CHANnel:COUPle<m>:AMPLitude?

➤ **功能描述:**

设置通道幅度耦合开关，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

➤ **返回格式:**

查询返回通道耦合幅度开关状态，0 表示关闭，1 表示开启。

➤ **举例:**

:CHANnel:COUPle1:AMPLitude ON            开启 1 和 2 通道耦合幅度  
:CHANnel:COUPle1:AMPLitude?            返回 1

**:CHANnel:COUPle<m>:AMPLitude:SCALe**

➤ **命令格式:**

:CHANnel:COUPle<m>:AMPLitude:SCALe <scale>  
:CHANnel:COUPle<m>:AMPLitude:SCALe?

➤ **功能描述:**

设置通道耦合幅度比例，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<scale>: 耦合幅度比例。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

➤ **返回格式:**

查询返回通道耦合幅度比例，返回科学计数法。

➤ **举例:**

:CHANnel:COUPle1:AMPLitude:SCALe 0.1    设置 2 通道比 1 通道耦合比例为 0.1  
:CHANnel:COUPle1:AMPLitude:SCALe?    返回 1e-1

**:CHANnel:COUPle<m>:AMPLitude:DEV**

➤ **命令格式:**

:CHANnel:COUPle<m>:AMPLitude:DEV <dev >

:CHANnel:COUPle<m>:AMPLitude:DEV?

➤ **功能描述:**

设置通道耦合幅度偏差，通道只存在 1、2 通道耦合和 3、4 通道耦合两种类型。

<scale>: 耦合幅度偏差，单位 V。

<m>: 通道号，m 取值 1、2。

1 表示一和二通道耦合；2 表示三和四通道耦合

➤ **返回格式:**

查询返回通道耦合幅度偏差，返回科学计数法。

➤ **举例:**

:CHANnel:COUPle1:AMPLitude:DEV 100                    设置 2 通道比 1 通道耦合偏差为 100Hz

:CHANnel:COUPle1:AMPLitude:DEV?                    返回 1e+2

## 连续

### :CHANnel<n>:BASE:WAVE

➤ **命令格式:**

:CHANnel<n>:BASE:WAVE { SINE | SQUARE | PULSE | RAMP | ARB | NOISE | DC | HARMONIC | PRBS }

:CHANnel<n>:BASE:WAVE?

➤ **功能描述:**

设置指定通道基波类型。分别为正弦波、方波、脉冲波、三角波、任意波、噪声、直流、谐波、伪随机二进制序列。

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道的基波类型。

➤ **举例:**

:CHANnel1:BASE:WAVE SINE	设置通道 1 基本类型为正弦波
:CHANnel1:BASE:WAVE?	查询返回 SINE

### :CHANnel<n>:BASE:FREQuency

➤ **命令格式:**

:CHANnel<n>:BASE:FREQuency {<freq>}

:CHANnel<n>:BASE:FREQuency?

➤ **功能描述:**

设置指定通道输出频率。

<freq>表示频率值, 单位 Hz。(1e-6s ~ 当前波形允许最大频率)

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道的输出频率, 采用科学计数法返回。

➤ **举例:**

:CHANnel1:BASE:FREQuency 2000	设置通道 1 输出频率 2KHz
:CHANnel1:BASE:FREQuency?	查询返回 2e+3

**:CHANnel<n>:BASE:PERiod****➤ 命令格式:**

:CHANnel<n>:BASE:PERiod { <period>}

:CHANnel<n>:BASE:PERiod?

**➤ 功能描述:**

设置指定通道输出周期。

<period>表示周期，单位 S。

若为正弦波：范围为（当前允许最大时间 ~ 1e3s）

<n>：通道号，n 取值 1、2，3，4。

**➤ 返回格式:**

查询返回指定通道的限幅上限值，采用科学计数法返回。

**➤ 举例:**

:CHANnel1:BASE:PERiod 0.002                    设置通道 1 输出周期 2ms

:CHANnel1:BASE:PERiod?                        查询返回 2e-3

**:CHANnel<n>:BASE:PHASe****➤ 命令格式:**

:CHANnel<n>:BASE:PHASe { <phase>}

:CHANnel<n>:BASE:PHASe?

**➤ 功能描述:**

设置指定通道输出相位。

<phase>表示相位，单位°，范围-360~360。

<n>：通道号，n 取值 1、2，3，4。

**➤ 返回格式:**

查询返回指定通道的输出相位。

**➤ 举例:**

:CHANnel1:BASE:PHASe 20                        设置通道 1 输出相位为 20°

:CHANnel1:BASE:PHASe?                        查询返回 20



**:CHANnel<n>:BASE:AMPLitude****➤ 命令格式:**

```
:CHANnel<n>:BASE:AMPLitude { <amp;gt;}
:CHANnel<n>:BASE:AMPLitude?
```

**➤ 功能描述:**

设置指定通道输出幅度。

<amp;gt;表示电压，单位当前通道指定单位。1mVpp ~ 当前负载下输出的最大值。

若当前单位为 VPP，当前负载下最大值=当前负载\*20/(50+当前负载)

<n>: 通道号，n 取值 1、2、3、4。

**➤ 返回格式:**

查询返回指定通道的输出幅度，采用科学计数法返回。

**➤ 举例:**

```
:CHANnel1:BASE:AMPLitude 2           设置通道 1 输出幅度为 2V
:CHANnel1:BASE:AMPLitude?           查询返回 2e+0
```

**:CHANnel<n>:BASE:OFFSet****➤ 命令格式:**

```
:CHANnel<n>:BASE:OFFSet { <voltage>}
:CHANnel<n>:BASE:OFFSet?
```

**➤ 功能描述:**

设置指定通道输出直流偏移。

<voltage>表示电压，单位 V。范围为：0~±当前负载下最大直流。

当前负载下的最大直流= 当前负载\*10/(50+当前负载) - 当前交流最小值/2;

交流最小值为 2mVpp,直流模式取 0;

<n>: 通道号，n 取值 1、2、3、4。

**➤ 返回格式:**

查询返回指定通道的输出直流偏移，采用科学计数法返回。

**➤ 举例:**

```
:CHANnel1:BASE:OFFSet 2           设置通道 1 输出直流偏移为 2V
:CHANnel1:BASE:OFFSet?           查询返回 2e+0
```

**:CHANnel<n>:BASE:HIGH****➤ 命令格式:**

```
:CHANnel<n>:BASE:HIGH { <voltage>}
```

```
:CHANnel<n>:BASE:HIGH?
```

**➤ 功能描述:**

设置指定通道信号输出高值。

<voltage>表示电压，单位当前通道指定单位。

<n>: 通道号，n 取值 1、2，3，4。

**➤ 返回格式:**

查询返回指定通道信号输出高值，采用科学计数法返回。

**➤ 举例:**

```
:CHANnel1:BASE:HIGH 2
```

设置通道 1 信号输出高值为 2V

```
:CHANnel1:BASE:HIGH?
```

查询返回 2e+0

**:CHANnel<n>:BASE:LOW****➤ 命令格式:**

```
:CHANnel<n>:BASE:LOW { <voltage>}
```

```
:CHANnel<n>:BASE:LOW?
```

**➤ 功能描述:**

设置指定通道信号输出低值。

<voltage>表示电压，单位当前通道指定单位。

<n>: 通道号，n 取值 1、2，3，4。

**➤ 返回格式:**

查询返回指定通道信号输出低值，采用科学计数法返回。

**➤ 举例:**

```
:CHANnel1:BASE:LOW 2
```

设置通道 1 信号输出低值为 2V

```
:CHANnel1:BASE:LOW?
```

查询返回 2e+0

**:CHANnel<n>:BASE:DUTY**➤ **命令格式:**

:CHANnel&lt;n&gt;:BASE:DUTY { &lt;duty&gt;}

:CHANnel&lt;n&gt;:BASE:DUTY?

➤ **功能描述:**

设置指定通道信号输出占空比。

&lt;duty&gt;表示占空比，单位%，范围 0~100。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道信号输出占空比。

➤ **举例:**

:CHANnel1:BASE:DUTY 20

设置通道 1 信号输出占空比为 20%

:CHANnel1:BASE:DUTY?

查询返回 20

**:CHANnel<n>:BASE:BITRatio**➤ **命令格式:**

:CHANnel&lt;n&gt;:BASE:BITRatio &lt;ratio&gt;

:CHANnel&lt;n&gt;:BASE:BITRatio?

➤ **功能描述:**

设置指定通道位率值，该指令只对具备位率功能的波形有效。

&lt;ratio&gt;表示位率，单位为 bps

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道位率值，采用科学计数法返回。

➤ **举例:**

:CHANnel1:BASE:BITRatio 1000000

设置通道 1 位率 100Kbps

:CHANnel1:BASE:BITRatio?

查询返回 1e+6

**:CHANnel<n>:BASE:ARB**➤ **命令格式:**

:CHANnel&lt;n&gt;:BASE:ARB &lt;source&gt;,&lt;filename&gt;

:CHANnel&lt;n&gt;:BASE:ARB?

➤ **功能描述:**

设置指定通道加载基波任意波源下某文件任意波形数据。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

&lt;source&gt;: {INTERNAL|EXTERNAL|USER}，分别内部、外部、自定义三种。

&lt;filename&gt;: 任意波形文件名称。



**:CHANnel<n>:PULSe:FALL**➤ **命令格式:**

:CHANnel&lt;n&gt;:PULSe:FALL {&lt;width&gt;}

:CHANnel&lt;n&gt;:PULSe:FALL?

➤ **功能描述:**

设置指定通道信号脉冲波下降沿脉宽。

&lt;width&gt;表示脉宽，单位 S。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道信号脉冲波下降沿脉宽，采用科学计数法返回。

➤ **举例:**

:CHANnel1:PULSe:FALL 0.002

设置通道 1 信号下降沿脉宽为 2ms

:CHANnel1:PULSe:FALL?

查询返回 2e-3

**:CHANnel<n>:NOISe:BANDwith**➤ **命令格式:**

:CHANnel&lt;n&gt;:NOISe:BANDwith {&lt;width&gt;}

:CHANnel&lt;n&gt;:NOISe:BANDwith?

➤ **功能描述:**

设置指定通道噪声信号带宽。

&lt;width&gt;表示带宽，单位 Hz。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道噪声信号带宽，采用科学计数法返回。

➤ **举例:**

:CHANnel1:NOISe:BANDwith 2MHz

设置通道 1 噪声信号带宽 2MHz

:CHANnel1:NOISe:BANDwith?

查询返回 2e+6

**:CHANnel<n>:HARMonic:TYPe?**➤ **命令格式:**

:CHANnel&lt;n&gt;:HARMonic:TYPe {ODD|EVEN|ALL|USER}

:CHANnel&lt;n&gt;:HARMonic:TYPe?

➤ **功能描述:**

设置指定通道谐波类型。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道谐波类型。

- **举例:**  
:CHANnel1:HARMonic:TYPe ODD                    设置通道 1 谐波类型为偶次谐波  
:CHANnel1:HARMonic:TYPe?                        查询返回 ODD

#### **:CHANnel<n>:HARMonic:MAX:ORDER?**

- **命令格式:**  
:CHANnel<n>:HARMonic:MAX:ORDer <order>  
:CHANnel<n>:HARMonic:MAX:ORDer?
- **功能描述:**  
设置指定通道最大谐波次数。  
< order >: 谐波次数, 范围 2~16。  
<n>: 通道号, n 取值 1、2, 3, 4。
- **返回格式:**  
查询返回指定通道最大谐波次数, 返回整型数据。
- **举例:**  
:CHANnel1:HARMonic:TOTal:ORDer 2                设置通道 1 最大谐波 2 次  
:CHANnel1:HARMonic:TOTal:ORDer?                查询返回 2

#### **:CHANnel<n>:HARMonic:USER:TYPe?**

- **命令格式:**  
:CHANnel<n>:HARMonic:USER:TYPe #H<order>  
:CHANnel<n>:HARMonic:USER:TYPe?
- **功能描述:**  
设置指定通道自定义谐波类型。  
< order >: 自定义谐波类型, #H 代表 16 进制数。X0111 1111 1111 1111 位分别表示谐波开关。  
<n>: 通道号, n 取值 1、2, 3, 4。
- **返回格式:**  
查询返回指定通道自定义谐波类型, 返回整型数据。
- **举例:**  
:CHANnel1:HARMonic:USER:ORDer #H7FFF        设置通道 1 自定义谐波类型  
:CHANnel1:HARMonic:USER:ORDer?                查询返回 32767

#### **:CHANnel<n>:HARMonic:ORDer<m>:AMPLitude?**

- **命令格式:**  
:CHANnel<n>:HARMonic:ORDer<m>:AMPLitude <amp>  
:CHANnel<n>:HARMonic:ORDer<m>:AMPLitude?
- **功能描述:**

设置指定通道下指定谐波次数的幅度值。

< amp >: 幅度值, 单位 Vpp。

<n>: 通道号, n 取值 1、2, 3, 4。

<m>: 谐波次数, m 取值 2~16。

➤ **返回格式:**

查询返回指定通道下指定谐波次数的幅度值, 采用科学计数法返回。

➤ **举例:**

:CHANnel1:HARM:ORDER2:AMP 0.02      设置通道 1 下 2 次谐波的幅度值 20mVpp  
:CHANnel1:HARM:ORDER2:AMP?      查询返回 2e-2

**:CHANnel<n>:HARMonic:ORDER<m>:PHASe?**

➤ **命令格式:**

:CHANnel<n>:HARMonic:ORDER<m>:PHASe <phase>  
:CHANnel<n>:HARMonic:ORDER<m>:PHASe?

➤ **功能描述:**

设置指定通道下指定谐波次数的相位值。

<phase>: 相位值, 单位°。

<n>: 通道号, n 取值 1、2, 3, 4。

<m>: 谐波次数, m 取值 2~16。

➤ **返回格式:**

查询返回指定通道下指定谐波次数的相位值, 采用科学计数法返回。

➤ **举例:**

:CHANnel1:HARM:ORDER2:PHASe 20      设置通道 1 下 2 次谐波的相位值 20°  
:CHANnel1:HARM:ORDER2:PHASe?      查询返回 2e+1

**:CHANnel<n>:ARB:MODE**

➤ **命令格式:**

:CHANnel<n>:ARB:MODE {DDS | POINTS }  
:CHANnel<n>:ARB:MODE?

➤ **功能描述:**

设置指定通道任意波输出模式, 分别为 DDS 和逐点模式。

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道任意波模式。

➤ **举例:**

:CHANnel1:ARB:MODE DDS      设置通道 1 任意波模式 DDS 输出模式  
:CHANnel1:ARB:MODE?      查询返回 DDS

## **:CHANnel<n>:ARB:FILTer**

➤ **命令格式:**

:CHANnel<n>:ARB:FILTer { ZEROHOLD | LINE }  
:CHANnel<n>:ARB:FILTer?

➤ **功能描述:**

设置指定通道任意波输出插值方式，分别为零点保持和线性插值。  
<n>: 通道号，n 取值 1、2。

➤ **返回格式:**

查询返回指定通道任意波输出插值方式。

➤ **举例:**

:CHANnel1:ARB:FILTer LINE	设置通道 1 任意波以线性插值方式输出
:CHANnel1:ARB:FILTer?	查询返回 LINE

## **调制**

### **:CHANnel<n>:MODulate:TYPE**

➤ **命令格式:**

:CHANnel<n>:MODulate:TYPE <type>  
:CHANnel<n>:MODulate:TYPE?

➤ **功能描述:**

设置指定通道信号调制类型。  
<type>: { AM|BAM|QAM|ASK|FM|FSK|ThreeFSK|FourFSK|PM|PSK|BPSK|QPSK|OSK|PWM|SUM }  
分别为调幅、双边调幅、正交调制、幅移键控、调频、频移键控、三频键控、四频键控、调相、相移键控、双相移键控、四相移键控、震荡键控、脉宽调制、总和调制。  
<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道信号调制类型。

➤ **举例:**

:CHANnel1:MODulate:TYPE AM	设置通道 1 信号 AM 调制
:CHANnel1:MODulate:TYPE?	查询返回 AM

### **:CHANnel<n>:MODulate:WAVE**

➤ **命令格式:**

:CHANnel<n>:MODulate:WAVE { SINE|SQUare|UPRamp|DNRamp|ARB|NOISe }  
:CHANnel<n>:MODulate:WAVE?

➤ **功能描述:**

设置指定通道信号调制波类型，分别为正弦波、方波、上三角、下三角、任意波、噪声。



<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道信号调制波类型。

➤ **举例:**

:CHANnel1:MODulate:WAVe SINE

设置通道 1 信号调制波类型为正弦波

:CHANnel1:MODulate:WAVe?

查询返回 SINE

**:CHANnel<n>:MODulate:SOURce**

➤ **命令格式:**

:CHANnel<n>:MODulate:SOURce { INTernal|EXTernal }

:CHANnel<n>:MODulate:SOURce?

➤ **功能描述:**

设置指定通道调制源, 分别内部、外部两种。

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道调制源。

➤ **举例:**

:CHANnel1:MODulate:SOURce INTernal

设置通道一调制源为内部

:CHANnel1:MODulate:SOURce?

查询返回 INTernal

**:CHANnel<n>:MODulate:FREQuency**➤ **命令格式:**

:CHANnel&lt;n&gt;:MODulate:FREQuency {&lt;freq&gt;}

:CHANnel&lt;n&gt;:MODulate:FREQuency?

➤ **功能描述:**

设置指定通道信号调制频率。

&lt;freq&gt;表示频率，单位 Hz。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道信号调制频率，返回采样科学计数法表示。

➤ **举例:**

:CHANnel1:MODulate:FREQuency 2000                    设置通道 1 信号调制频率 2KHz

:CHANnel1:MODulate:FREQuency?                    查询返回 2e+3

**:CHANnel<n>:MODulate:IQMap**➤ **命令格式:**

:CHANnel&lt;n&gt;:MODulate: IQMap {&lt;IQ TYPE&gt;}

:CHANnel&lt;n&gt;:MODulate: IQMap?

➤ **功能描述:**

设置指定 QAM 的 IQ 类型可以为:

4QAM, 8QAM, 16QAM, 32QAM, 64QAM, 128QAM, 256QAM。

&lt; IQ TYPE &gt;表示 IO 映射类型。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道的 IQ 类型

➤ **举例:**

:CHANnel1:MODulate:IQMap 32QAM                    设置通道 1 调制 IQ 映射为 32QAM

:CHANnel1:MODulate:IQMap?                    查询返回 32QAM

**:CHANnel<n>:MODulate:ARB**➤ **命令格式:**

:CHANnel&lt;n&gt;:MODulate:ARB &lt;source&gt;,&lt;filename&gt;

:CHANnel&lt;n&gt;:MODulate:ARB?

➤ **功能描述:**

设置指定通道加载调制任意波源下某文件任意波形数据。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

&lt;source&gt;: {INTernal|EXTernal|USER}, 分别内部、外部、自定义三种。

<filename>: 任意波形文件名称。

➤ **举例:**

:CHANnel1:MODulate:ARB INTernal, "test.bsv"

### **:CHANnel<n>:MODulate:DEPTh**

➤ **命令格式:**

:CHANnel<n>:MODulate:DEPTh { <depth> }

:CHANnel<n>:MODulate:DEPTh?

➤ **功能描述:**

设置指定通道调制深度。

<depth>表示调制深度，单位%。0% ~ 100%，其中 AM 调制深度为 0% ~ 120%

<n>: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道调制深度。

➤ **举例:**

:CHANnel1:MODulate:DEPTh 50

设置通道一调制深度为 50%

:CHANnel1:MODulate:DEPTh?

查询返回 50

### **:CHANnel<n>:MODulate:BITRatio**

➤ **命令格式:**

:CHANnel<n>:MODulate:BITRatio <ratio>

:CHANnel<n>:MODulate:BITRatio?

➤ **功能描述:**

设置指定通道调制位率值，该指令只对具备位率功能调制类型有效。

<ratio>表示位率，单位为 bps

<n>: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道调制位率值，采用科学计数法返回。

➤ **举例:**

:CHANnel1:MODulate:BITRatio 1000000

设置通道 1 位率 100Kbps

:CHANnel1:MODulate:BITRatio?

查询返回 1e+6

### **:CHANnel<n>:OSK:TRIGger:SOURce**

➤ **命令格式:**

:CHANnel<n>:OSK:TRIGger:SOURce { INTernal|EXTernal }

:CHANnel<n>:OSK:TRIGger:SOURce?

➤ **功能描述:**

设置指定通道震荡键控触发源，分别为内部和外部。

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道震荡键控触发源。

➤ **举例:**

设置通道一震荡键控内部触发源

:CHANnel1:OSK:TRIGger:SOURce?

查询返回 INTERNAL

**:CHANnel<n>:FM:FREQuency:DEV**

➤ **命令格式:**

:CHANnel<n>:FM:FREQuency:DEV { <freq> }

:CHANnel<n>:FM:FREQuency:DEV?

➤ **功能描述:**

设置指定通道频率偏差。

<freq>表示频率偏移, 单位 Hz。0Hz ~ 当前基波频率

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道频率偏移, 采样科学计数法返回数据。

➤ **举例:**

:CHANnel1:FM:FREQuency:DEV 2000

设置通道一频率偏移 2KHz

:CHANnel1:FM:FREQuency:DEV?

查询返回 2e+3

**:CHANnel<n>:PM:PHASe:DEV**➤ **命令格式:**

:CHANnel&lt;n&gt;:PM:PHASe:DEV { &lt;phase&gt;}

:CHANnel&lt;n&gt;:PM:PHASe:DEV?

➤ **功能描述:**

设置指定通道输出相位偏差。

&lt; phase &gt;表示相位偏移, 单位°, 范围 0~360。

&lt;n&gt;: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道输出相位偏移。

➤ **举例:**

:CHANnel1:PM:PHASe:DEV 30

设置通道一相位偏移 30°

:CHANnel1:PM:PHASe:DEV?

查询返回 30

**:CHANnel<n>:PWM:DUTY:DEV**➤ **命令格式:**

:CHANnel&lt;n&gt;:PWM:DUTY:DEV { &lt;duty&gt;}

:CHANnel&lt;n&gt;:PWM:DUTY:DEV?

➤ **功能描述:**

设置指定通道输出脉宽调制下脉宽偏差。

&lt; duty &gt;表示脉宽偏差, 单位%, 范围 0~100。

&lt;n&gt;: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道脉宽调制下脉宽偏差, 以科学计数法返回数据。

➤ **举例:**

:CHANnel1:PWM:DUTY:DEV 10

设置通道一脉宽偏差 10%

:CHANnel1:PWM:DUTY:DEV?

查询返回 1e+1

**:CHANnel<n>:FSK:FREQuency<m>**➤ **命令格式:**

:CHANnel&lt;n&gt;:FSK:FREQuency&lt;m&gt; { &lt;freq&gt;}

:CHANnel&lt;n&gt;:FSK:FREQuency&lt;m&gt;?

➤ **功能描述:**

设置指定通道输出多频移键控跳频频率, 必须要提前指定调制方式, 此指令才可生效。

&lt; freq &gt;表示频率, 单位 Hz。

&lt;n&gt;: 通道号, n 取值 1、2, 3, 4。

&lt;m&gt;: 频率序号, 2FSK 时取值 1; 3FSK 时取值 1、2; 4FSK 时取值 1、2、3;

➤ **返回格式:**

查询返回指定通道输出跳频频率，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:FSK:FREQ1 2000                    设置通道一输出跳频频率 2KHz  
:CHANnel1:FSK:FREQ1?                        查询返回 2e+3

**:CHANnel<n>:PSK:PHASe<m>**

➤ **命令格式:**

:CHANnel<n>:PSK:PHASe<m> { < phase > }  
:CHANnel<n>:PSK:PHASe<m>?

➤ **功能描述:**

设置指定通道输出多相移键控相位值，必须要提前指定调制方式，此指令才可生效。

< phase >表示相位，单位°，范围 0~360。

<n>: 通道号，n 取值 1、2、3、4。

<m>: 相位序号，BPSK 时取值 1；QPSK 时取值 1、2、3；

➤ **返回格式:**

查询返回指定通道相移键控相位值，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:PSK:PHAS1 90                    设置通道一输出相位 90°  
:CHANnel1:PSK:PHAS1?                        查询返回 9e+1

**:CHANnel<n>:OSK:TIME**

➤ **命令格式:**

:CHANnel<n>:OSK:TIME { < time > }  
:CHANnel<n>:OSK:TIME?

➤ **功能描述:**

设置指定通道调制模式下震荡键控的震荡时间。

< time >表示震荡时间，单位 S。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道调制模式下震荡键控的震荡时间，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:OSK:TIME 2ms                    设置通道一震荡键控的震荡时间 2ms  
:CHANnel1:OSK:TIME?                        查询返回 2e-3

# 扫频

## **:CHANnel<n>:SWEep:TYPe**

➤ **命令格式:**

:CHANnel<n>:SWEep:TYPe { LINE|LOG|STEP|LIST }

:CHANnel<n>:SWEep:TYPe?

➤ **功能描述:**

设置指定通道扫频模式，分别为线性扫频、对数扫频、步进扫频、列表扫频。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道扫频模式。

➤ **举例:**

:CHANnel1:SWEep:TYPe LINE                      设置通道一线性扫频模式

:CHANnel1:SWEep:TYPe?                            查询返回 LINE

## **:CHANnel<n>:SWEep:FREQuency:STARt**

➤ **命令格式:**

:CHANnel<n>:SWEep:FREQuency:STARt <freq>

:CHANnel<n>:SWEep:FREQuency:STARt?

➤ **功能描述:**

设置指定通道扫频的起始频率。

<freq>表示频率，单位 Hz。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道扫频的起始频率，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:SWE:FREQ:STAR 2000                      设置通道一输出扫频的起始频率 2KHz

:CHANnel1:SWE:FREQ:STAR?                            查询返回 2e+3

## **:CHANnel<n>:SWEep:FREQuency:STOP**

➤ **命令格式:**

:CHANnel<n>:SWEep:FREQuency:STOP <freq>

:CHANnel<n>:SWEep:FREQuency:STOP?

➤ **功能描述:**

设置指定通道扫频的截止频率。

<freq>表示频率，单位 Hz。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式:**

查询返回指定通道输出扫频的截止频率，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:SWE:FREQ:STOP 2000

设置通道一输出扫频的截止频率 2KHz

:CHANnel1:SWE:FREQ:STOP?

查询返回 2e+3



**:CHANnel<n>:SWEep:TIME**➤ **命令格式:**

:CHANnel&lt;n&gt;:SWEEP:TIME &lt;time&gt;

:CHANnel&lt;n&gt;:SWEEP:TIME?

➤ **功能描述:**

设置指定通道扫频时的扫描时间。

&lt;time&gt;表示时间，单位 S。范围为：1ms ~ 500s

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道扫频时的扫描时间，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:SWEEP:TIME 2

置通道一扫频时的扫描时间为 2S

:CHANnel1:SWEEP:TIME?

查询返回 2e+0

**:CHANnel<n>:SWEep:HOLD**➤ **命令格式:**

:CHANnel&lt;n&gt;:SWEEP:HOLD &lt;time&gt;

:CHANnel&lt;n&gt;:SWEEP:HOLD?

➤ **功能描述:**

设置指定通道扫频驻留时间，该指令只在步进和列表扫频时有效。

&lt;time&gt;表示时间，单位 S。

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道扫频驻留时间，以科学计数法返回数据。

➤ **举例:**

:CHANnel1:SWEEP:HOLD 2

置通道一扫频时的扫描时间为 2S

:CHANnel1:SWEEP:HOLD?

查询返回 2e+0

**:CHANnel<n>:SWEep:STEPS**➤ **命令格式:**

:CHANnel&lt;n&gt;:SWEEP:STEPS &lt;steps&gt;

:CHANnel&lt;n&gt;:SWEEP:STEPS?

➤ **功能描述:**

设置指定通道步进扫频时的总步数，该指令只在步进扫频时有效。

&lt;steps&gt;: 步数

&lt;n&gt;: 通道号，n 取值 1、2，3，4。

➤ **返回格式:**

查询返回指定通道扫频时的扫描时间，返回整型数据。

➤ **举例：**

:CHANnel1:SWEEP:STEPS 10

设置指定通道步进扫频 10 步

:CHANnel1:SWEEP:STEPS?

查询返回 10

### **:CHANnel<n>:SWEep:TRIGger**

➤ **命令格式：**

:CHANnel<n>:SWEep:TRIGger

➤ **功能描述：**

触发指定通道扫频输出，该参数仅在触发模式设置为手动触发时候有效。

➤ **举例：**

:CHANnel1:SWEep:TRIGger

触发一次扫频信号输出

## 猝发

### **:CHANnel<n>:BURSt:TYPE**

➤ **命令格式：**

:CHANnel<n>:BURSt:TYPE {NCYC|GATe|INFinit}

:CHANnel<n>:BURSt:TYPE?

➤ **功能描述：**

设置指定通道猝发类型，分别为 N 周期、门控、无限。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式：**

查询返回指定通道猝发类型。

➤ **举例：**

:CHANnel1:BURSt:TYPE NCYC

设置通道一通道 N 周期猝发

:CHANnel1:BURSt:TYPE?

查询返回 2e+0

### **:CHANnel<n>:BURSt:PERiod**

➤ **命令格式：**

:CHANnel<n>:BURSt:PERiod <period >

:CHANnel<n>:BURSt:PERiod?

➤ **功能描述：**

设置指定通道猝发周期。

< period >表示时间，单位 S。

<n>: 通道号，n 取值 1、2、3、4。

➤ **返回格式：**

查询返回指定通道猝发周期，以科学计数法返回数据。

- **举例:**  
:CHANnel1:BURSt:PERiod 5ms                    设置通道一猝发周期 5ms  
:CHANnel1:BURSt:PERiod?                        查询返回 5e-3

### **:CHANnel<n>:BURSt:PHASe**

- **命令格式:**  
:CHANnel<n>:BURSt:PHASe <phase>  
:CHANnel<n>:BURSt:PHASe?
- **功能描述:**  
设置指定通道猝发相位。  
< phase >表示相位，单位°。范围为：0 ~ 360  
<n>: 通道号，n 取值 1、2，3，4。
- **返回格式:**  
查询返回指定通道猝发相位，以科学计数法返回数据。
- **举例:**  
:CHANnel1:BURSt:PHASe 18                        设置通道一猝发相位 18°  
:CHANnel1:BURSt:PHASe?                        查询返回 1.8e+1

### **:CHANnel<n>:BURSt:CYCLes**

- **命令格式:**  
:CHANnel<n>:BURSt:CYCLes <cycles>  
:CHANnel<n>:BURSt:CYCLes?
- **功能描述:**  
设置指定通道猝发循环次数。  
< cycles >表示循环次数，整型数据。  
<n>: 通道号，n 取值 1、2，3，4。
- **返回格式:**  
查询返回指定通道猝发循环次数。
- **举例:**  
:CHANnel1:BURSt:CYCLes 2                        设置指定通道猝发循环次数为 2  
:CHANnel1:BURSt:CYCLes?                        查询返回 2

### **:CHANnel<n>:BURSt:GATe:POLarity**

- **命令格式:**  
:CHANnel<n>:BURSt:GATe:POLarity {POSitive|NEGative}  
:CHANnel<n>:BURSt:GATe:POLarity?
- **功能描述:**  
设置指定通道门控猝发极性，分别为正极性、负极性。

<n>: 通道号, n 取值 1、2, 3, 4。

➤ **返回格式:**

查询返回指定通道门控猝发极性。

➤ **举例:**

:CHANnel1:BURSt:GATe:POLarity POSitive      设置通道一门控猝发极性为正极性

:CHANnel1:BURSt:GATe:POLarity?      查询返回 POSitive

**:CHANnel<n>:BURSt:TRIGger**

➤ **命令格式:**

:CHANnel<n>:BURSt:TRIGger

➤ **功能描述:**

触发指定通猝发输出, 该参数仅在触发模式设置为手动触发时候有效。

➤ **举例:**

:CHANnel1:BURSt:TRIGger      触发一次猝发信号输出

## WARB 命令

用于写任意波形文件指令，包括基本任意波形和调制任意波形写配置。

### :WARB<n>:MODulate

➤ **命令格式:**

:WARB<n>:MODulate <arb file>

➤ **功能描述:**

用于写调制任意波形，先发送该指令，然后发送任意波形文件数据到信号源。

<arb file>表示任意波形文件名称，只支持 bsv 文件格式。

➤ **举例:**

:WARB1:MODulate "test.bsv"                   写通道一调制任意波形文件

### :WARB<n>:CARRier

➤ **命令格式:**

:WARB<n>:CARRier <arb file>

➤ **功能描述:**

用于写基波任意波形，先发送该指令，然后发送任意波形文件数据到信号源。

<arb file>表示任意波形文件名称，只支持 bsv 文件格式。

➤ **举例:**

:WARB1:CARRier "test.bsv"                   写通道一基波任意波形文件

### :WFREQLIST<n>

➤ **命令格式:**

:WFREQLIST<n> <arb file>

➤ **功能描述:**

用于写频率列表文件，先发送该指令，然后发送频率列表文件数据到信号源。

<arb file>表示频率列表文件名称，只支持 csv 文件格式。

➤ **举例:**

:WFREQLIST1     "freq.csv"                   写通道一频率列表文件

## DIGital 命令

用于输出数字通讯信号，例如 UART、SPI、I2C 等通讯信号。

### :DIGital

➤ **命令格式:**

```
:DIGital {{1 | ON} | {0 | OFF}}
```

```
:DIGital?
```

➤ **功能描述:**

设置指定通道打开或关闭数字通讯信号功能。

➤ **返回格式:**

查询返回数字通讯信号功能打开或关闭，0 表示关闭，1 表示打开。

➤ **举例:**

```
:DIGital ON                打开数字通讯信号功能
```

```
:DIGital?                  查询返回 1
```

### :DIGital:TYPE

➤ **命令格式:**

```
:DIGital:TYPE {UART|I2C|SPI }
```

```
:DIGital:TYPE?
```

➤ **功能描述:**

设置指定通道输出数字通讯信号类型，分别为 UART、I2C、SPI。

➤ **返回格式:**

查询返回指定通道输出数字通讯信号类型。

➤ **举例:**

```
:DIGital:TYPE UART        设置输出 UART 通讯信号
```

```
:DIGital:TYPE?           查询返回 UART
```

### :DIGital:AMPLitude

➤ **命令格式:**

```
:DIGital:AMPLitude <amp;>
```

```
:DIGital:AMPLitude?
```

➤ **功能描述:**

设置指定通道输出数字通讯信号幅度。

<amp;>:幅度，单位 V。

➤ **返回格式:**

查询返回指定通道输出数字通讯信号类型，采用科学计数法返回。

➤ **举例:**

```
:DIGital:AMPLitude 3      设置通道输出 3V 通讯信号
```

```
:DIGital:AMPLitude?     查询返回 3e+0
```

## **:DIGital:FORMat**

➤ **命令格式:**

:DIGital:FORMat { HEX|CHAR }

:DIGital:FORMat?

➤ **功能描述:**

设置指定通道输出数字通讯信号数据格式，分别为十六进制数据和 ASCII 数据。

➤ **返回格式:**

查询返回指定通道输出数字通讯信号数据格式。

➤ **举例:**

:DIGital:FORMat HEX

设置数字通讯信号数据格式为十六进制数据

:DIGital:FORMat?

查询返回 HEX

## **:DIGital:AS**

➤ **命令格式:**

:DIGital:AS {{1|ON}|{0|OFF}}

:DIGital:AS?

➤ **功能描述:**

设置指定通道输出数字通讯信号发送模式。OFF 表示手动发送；ON 表示自动发送。

➤ **返回格式:**

查询返回指定通道输出数字通讯信号是否开启自动发送模式，1 表示开启，0 表示未开启。

➤ **举例:**

:DIGital:AS ON

设置数字通讯信号自动发送

:DIGital:AS?

查询返回 1

## **:DIGital:AS:INTerval**

➤ **命令格式:**

:DIGital:AS:INTerval <time>

:DIGital:AS:INTerval?

➤ **功能描述:**

设置指定通道自动输出数字通讯信号间隔时间。

<time>: 间隔时间，单位 S。

➤ **返回格式:**

查询返回指定通道输出数字通讯信号间隔时间，采用科学计数法返回。

➤ **举例:**

:DIGital:AS:INTerval 10ms

设置自动输出数字通讯信号间隔时间 10ms

:DIGital:AS:INTerval?

查询返回 1e-2

## **:DIGital:TRIGger**

➤ **命令格式:**

:DIGital:TRIGger

- **功能描述:**  
触发指定通道输出数字通讯信号发送, 在信号手动发送模式下有效。
- **举例:**  
:DIGital:TRIGger                          触发一次数字信号发送

#### **:DIGital:DATA**

- **命令格式:**  
:DIGital:DATA <data>  
:DIGital:DATA?
- **功能描述:**  
设置指定通道自动输出数字通讯信号数据。  
<data>: 二进制字节流数据。
- **返回格式:**  
查询返回指定通道输出数字通讯信号数据, 返回二进制字节流数据。
- **举例:**  
:DIGital:DATA                          写入数字通讯信号数据到信号源中  
:DIGital:DATA?                         查询返回二进制字节流

### **UART**

#### **:DIGital:UART:BAUDrate**

- **命令格式:**  
:CHANnel:UART:BAUDrate <baudrate>  
:CHANnel:UART:BAUDrate?
- **功能描述:**  
设置指定通道输出数字 UART 通讯信号的波特率。  
<baudrate>: 波特率, 单位 bps, 整型类型数据。
- **返回格式:**  
查询返回指定通道输出数字 UART 通讯信号的波特率, 返回整型数据。
- **举例:**  
:DIGital:UART:BAUDrate 115200        设置 UART 通讯信号的波特率 115200  
:DIGital:UART:BAUDrate?                查询返回 115200

#### **:DIGital:UART:DATA**

- **命令格式:**  
:CHANnel:UART:DATA <bit >  
:CHANnel:UART:DATA?
- **功能描述:**  
设置指定通道输出数字 UART 通讯信号的数据位。



<bit>: 数据位, 整型类型数据, 范围 4~8。

➤ **返回格式:**

查询返回指定通道输出数字 UART 通讯信号的数据位, 返回整型数据。

➤ **举例:**

:DIGital:UART:DATA 4	设置 UART 通讯信号的数据位 4
:DIGital:UART:DATA?	查询返回 4

### **:DIGital:UART:STOP**

➤ **命令格式:**

:CHANnel:UART:STOP <bit>

:CHANnel:UART:STOP?

➤ **功能描述:**

设置指定通道输出数字 UART 通讯信号的停止位。

<bit>: 停止位, 整型类型数据, 范围 1~2。

➤ **返回格式:**

查询返回指定通道输出数字 UART 通讯信号的停止位, 返回整型数据。

➤ **举例:**

:DIGital:UART:STOP 1	设置 UART 通讯信号的停止位 1
:DIGital:UART:STOP?	查询返回 1

### **:DIGital:UART:PARity**

➤ **命令格式:**

:CHANnel:UART:PARity {NONE|EVEN|ODD}

:CHANnel:UART:PARity?

➤ **功能描述:**

设置指定通道输出数字 UART 通讯信号的校验位, 分别为无校验、奇校验、偶校验。

➤ **返回格式:**

查询返回指定通道输出数字 UART 通讯信号的校验位, 返回整型数据。

➤ **举例:**

:DIGital:UART:PARity NONE	设置 UART 通讯信号的无校验位
:DIGital:UART:PARity?	查询返回 NONE

## **IIC**

### **:DIGital: IIC:CLOCK**

➤ **命令格式:**

:CHANnel:IIC:CLOCK <freq>

:CHANnel:IIC:CLOCK?

➤ **功能描述:**

设置指定通道输出数字 IIC 通讯信号的时钟。

<freq>: 时钟, 单位 Hz。

➤ **返回格式:**

查询返回指定通道输出数字 IIC 通讯信号的时钟, 采用科学计数法返回。

➤ **举例:**

:DIGital:IIC:CLOCK 1000	设置 IIC 通讯信号的时钟 1KHz
:DIGital:IIC:CLOCK?	查询返回 1e+3

### **:DIGital:IIC:ADDRESS**

➤ **命令格式:**

:CHANnel:IIC:ADDRESS <address>  
:CHANnel:IIC:ADDRESS?

➤ **功能描述:**

设置指定通道输出数字 IIC 通讯信号的地址。

<address>: 地址, 整型类型数据。

➤ **返回格式:**

查询返回指定通道输出数字 IIC 通讯信号的地址。

➤ **举例:**

:DIGital:IIC:ADDRESS 3	设置 IIC 通讯信号的地址 3
:DIGital:IIC:ADDRESS?	查询返回 3

## **SPI**

### **:DIGital:SPI:CLOCK**

➤ **命令格式:**

:CHANnel:SPI:CLOCK <freq>  
:CHANnel:SPI:CLOCK?

➤ **功能描述:**

设置指定通道输出数字 SPI 通讯信号的时钟。

<freq>: 时钟, 单位 Hz。

➤ **返回格式:**

查询返回指定通道输出数字 SPI 通讯信号的时钟, 采用科学计数法返回。

➤ **举例:**

:DIGital:SPI:CLOCK 1000	设置 SPI 通讯信号的时钟 1KHz
:DIGital:SPI:CLOCK?	查询返回 1e+3

## **DISPlay 命令**

用于信号源显示相关信息。

## **:DISPlay:DATA?**

➤ **命令格式:**

:DISPlay:DATA?

➤ **功能描述:**

用于查询示波器当前屏幕的图像数据。

➤ **返回格式:**

查询返回图像数据，返回的数据符合 IEEE 488.2 #格式的二进制数据。

➤ **举例:**

:DISPlay:DATA?

查询返回图像数据

数据格式: #800012345+位图数据

## KEY 命令

用于控制电源操作面板上的按键和旋钮。

### :KEY:<key>

➤ **命令格式:**

:KEY:<key>

:KEY:<key>:LOCK { {1 | ON} | {0 | OFF} }

:KEY:<key>:LOCK?

:KEY:<key>:LED?

➤ **功能描述:**

用于设置按键功能及该按键的锁定/解锁。按键<key>的定义和描述，详见[附录 1: <key>列表](#)。

➤ **返回格式:**

查询返回按键锁定状态或者具有 LED 按键灯状态。

锁定状态: 0 表示未锁定, 1 表示锁定;

LED 灯状态: 0 表示不亮, 1 表示亮 (绿灯), 2 表示亮 (红灯)。

➤ **举例:**

:KEY:AUTO

自动设置示波器的各项控制值

:KEY:AUTO:LOCK ON/OFF

锁定/解锁按键

:KEY:AUTO:LOCK?

查询返回该按键锁定状态, 1 表示锁定

:KEY:AUTO:LED?

查询返回 LED 灯状态, 0 表示不亮

### :KEY:LOCK?

➤ **命令格式:**

:KEY:LOCK?

➤ **功能描述:**

用于查询所有按键的锁定状态。

➤ **返回格式:**

查询返回所有按键的锁定状态, 返回字符序列, 每个字符代表一个按键的锁定状态, 锁定为 ASCII '1', 未锁定为 ASCII '0', 按照[附录 1: <key>列表](#)顺序返回锁定状态。

➤ **举例:**

一共 10 个按键, 只有第 4、5 两个按键锁定, 返回 ASCII 字符串 0001100000

---

**:KEY:LED?**

- **命令格式:**  
:KEY:LED?
- **功能描述:**  
用于查询所有带灯按键灯状态。
- **返回格式:**  
查询返回所有按键的灯状态，返回字符序列，每个字符代表一个按键灯状态，亮为 ASCII' 1' ,不亮为 ASCII' 0' ，按照[附录 1: <key>列表](#)带灯的按键顺序返回灯状态。
- **举例:**  
一共 3 个按键带灯，只有 CH1、CH2 两个按键亮，CH3、CH4 两个按键不亮，返回 ASCII 字符串 1100

## 附录 1: <key>列表

按键命令关键字	功能描述	LED 灯
COUNtinue	连续 (CW)	
MOD	调制	
SWEep	扫描	
BURSt	猝发	
SINe	正弦波	
SQUare	方波	
RAMP	三角波	
PULSe	脉冲波	
ARB	任意波	
HARMonic	谐波	
NOISe	噪声	
DC	直流	
PRBS	伪随机二进制序列	
WARB	写任意波	
CH1	通道一按键	√
CH2	通道二按键	√
CH3	通道三按键	√
CH4	通道四按键	√
Utility	系统	
Right	方向键右	
Left	方向键左	
OK	确认键	
Up	方向键上	
Down	方向键下	
NUM0	数字键 0	
NUM1	数字键 1	
NUM2	数字键 2	
NUM3	数字键 3	
NUM4	数字键 4	
NUM5	数字键 5	
NUM6	数字键 6	
NUM7	数字键 7	
NUM8	数字键 8	
NUM9	数字键 9	
Dot	数字键小数点	
Symbol	数字键符号	

BACKspace	退格	
-----------	----	--

## 附录 2：按键锁定状态

位序	按键	状态
0	COUNtinue	0 表示未锁定, 1 表示锁定
1	MOD	0 表示未锁定, 1 表示锁定
2	SWEep	0 表示未锁定, 1 表示锁定
3	BURSt	0 表示未锁定, 1 表示锁定
4	SINe	0 表示未锁定, 1 表示锁定
5	SQUare	0 表示未锁定, 1 表示锁定
6	RAMP	0 表示未锁定, 1 表示锁定
7	PULSe	0 表示未锁定, 1 表示锁定
8	ARB	0 表示未锁定, 1 表示锁定
9	HARMonic	0 表示未锁定, 1 表示锁定
10	NOISe	0 表示未锁定, 1 表示锁定
11	DC	0 表示未锁定, 1 表示锁定
12	PRBS	0 表示未锁定, 1 表示锁定
13	WARB	0 表示未锁定, 1 表示锁定
14	CH1	0 表示未锁定, 1 表示锁定
15	CH2	0 表示未锁定, 1 表示锁定
16	CH3	0 表示未锁定, 1 表示锁定
17	CH4	0 表示未锁定, 1 表示锁定
18	Utility	0 表示未锁定, 1 表示锁定
19	Right	0 表示未锁定, 1 表示锁定
20	Left	0 表示未锁定, 1 表示锁定
21	OK	0 表示未锁定, 1 表示锁定
22	Up	0 表示未锁定, 1 表示锁定
23	Down	0 表示未锁定, 1 表示锁定
24	NUM0	0 表示未锁定, 1 表示锁定
25	NUM1	0 表示未锁定, 1 表示锁定
26	NUM2	0 表示未锁定, 1 表示锁定
27	NUM3	0 表示未锁定, 1 表示锁定
28	NUM4	0 表示未锁定, 1 表示锁定
29	NUM5	0 表示未锁定, 1 表示锁定
30	NUM6	0 表示未锁定, 1 表示锁定
31	NUM7	0 表示未锁定, 1 表示锁定
32	NUM8	0 表示未锁定, 1 表示锁定
33	NUM9	0 表示未锁定, 1 表示锁定
34	Dot	0 表示未锁定, 1 表示锁定
35	Symbol	0 表示未锁定, 1 表示锁定

36	BACKspace	0 表示未锁定，1 表示锁定
----	-----------	----------------