

UNI-T

MSO8000HD 系列混合信号示波器

20GSa/s | 8GHz | 2Gpts | 1,000,000wfms/s



编程手册 REV.1.0

2025 年 3 月

引言

本编程手册包括 MSO8000HD 系列示波器有关的 SCPI 指令和相关说明。

版权信息

优利德科技（中国）股份有限公司版权所有。

如果原购买者自购买该产品之日起三年内，将该产品出售或转让给第三方，则保修期应为自原购买者从优利德或授权的优利德分销商购买该产品之日起三年内。探头及其他附件和保险丝等不受此保证的保护。如果在适用的保修期内证明产品有缺陷，优利德可自行决定是修复有缺陷的产品且不收部件和人工费用，或用同等产品（由优利德决定）更换有缺陷的产品。优利德用作保修用途的部件、模块和更换产品可能是全新的，或者经修理具有相当于新产品的性能。所有更换的部件、模块和产品将成为优利德的财产。

以下提到的“客户”是指据声明本保证所规定权利的个人或实体。为获得本保证承诺的服务，“客户”必须在适用的保修期内向优利德通报缺陷，并为服务的履行做适当安排。客户应负责将有缺陷的产品装箱并运送到优利德指定的维修中心，同时预付运费并提供原购买者的购买证明副本。如果产品要运送到优利德维修中心所在国范围内的地点，优利德应支付向客户送返产品的费用。如果产品送返到任何其他地点，客户应负责支付所有的运费、关税、税金及任何其他费用。

本保证不适用于由于意外、机器部件的正常磨损、在产品规定的范围之外使用或使用不当或者维护保养不当或不足而造成的任何缺陷、故障或损坏。优利德根据本保证的规定无义务提供以下服务：

- 修理由非优利德服务代表人员对产品进行安装、修理或维护所导致的损坏；
- 修理由于使用不当或与不兼容的设备连接造成的损坏；
- 修理由于使用不符合本说明书要求的电源而造成的任何损坏或故障；
- 维修已改动或者与其他产品集成的产品（如果这种改动或集成会增加产品维修的时间或难度）；

本保证由优利德针对本产品而订立，用于替代任何其他的明示或暗示的保证，优利德及其经销商拒绝对于特殊目的的适销性或适用性做任何暗示的保证。对于违反本保证的情况，优利德负责修理或更换有缺陷产品是提供给客户的唯一和全部补救措施。无论优利德及其经销商是否被预先告知可能发生任何间接、特殊、偶然或必然的损坏，优利德及其经销商对这些损坏均概不负责。

商标信息

UNI-T 是优利德科技（中国）股份有限公司[UNI-TREND TECHNOLOGY(CHINA)CO., LTD]的注册商标。

文档版本

MSO8000HD-V1.0

声明

- 优利德产品受中国或其他国家专利权的保护，包括已取得或正在申请的专利。
- 优利德保留更改产品规格和价格的权利。
- 优利德保留所有权利。许可软件产品由优利德及其子公司或提供商所有，受国家版权法及国际条约规定的保护。本文中的信息将取代所有以前出版的资料中的信息。
- 技术数据如有变更，恕不另行通告。

软件版本

V1.00.0000

软件升级可能更改或增加产品功能，请关注 优利德官方网站:www.uni-trend.com.cn 获取最新版本手册和固件信息。

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

下面三种符号不是 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: :ACQUIRE:TYPE <type> 命令, type 可选 {NORMAL | PEAK | HIGHres | AVERages | ENvelope}。

■ 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: :ACQUIRE:TYPE <type> { NORMAL | PEAK | HIGHres | AVERages | ENvelope} 命令。

■ 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 :ACQUIRE:TYPE NORMAL 的形式发送 :ACQUIRE:TYPE <type> 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、整数型、实数型、离散型、ASCII 字符串。

- **布尔型**

参数取值为“ON” (1) 或“OFF” (0)。例如: :AWG1:ENABle <enable> 命令中的<enable>为布尔型{ON | OFF}或{1 | 0}。

- **整数型**

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: :DISPlay:SCReen:BRIGtness <value>命令中的参数<value>可取 5 到 100 范围内的任一整数。

- **实数型**

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, CHANnel1: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: :ACQuire:TYPe <type> {NORMal | PEAK | HIGHres | AVERages | ENvelope} 命令只能取参数: NORMal、PEAK、HIGHres、AVERages、ENvelope。

- **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

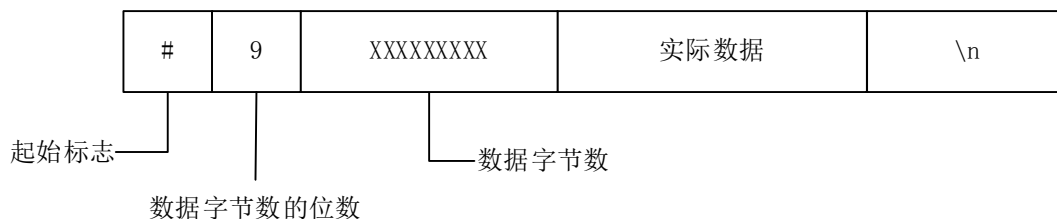
数据返回分为单个数据和批量数据返回。

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, 返回频率和时间值时 E 前部分小数点后面保留十三位数据, E 部分保留三位数据, 返回其它数值时 E 前部分小数点后面保留五位数据, E 部分保留三位数据;

批量数据返回格式为“数据块报头+数据块”, 其中数据块报头具有如下的格式: #后面的第一位数字 (9) 表示数据块报头中剩下的数字位数; 数据块报头中剩下的数字表示本次传输的数据字节数 (不足 9 位在前面补 0), 例如: 发送 1000 字节的数据块报头为#9000001000:

数据块格式:

DATA 是数据流,其他为 ASCII 字符,如下图所示: <#9XXXXXXXX + DATA + \n>



注意:当返回无效数据是 ASCII 类型数据时,用*表示;如果是实型数据类型,使用实型数据类型能表示的最大值数据返回。

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、示波器型号、产品序列号和软件版本号。

■ 返回格式:

制造商名称,示波器型号,产品序列号,由点号分隔的软件版本号。

■ 举例:

UNI-T, MSO8000HD, MSO8804, 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

用于对示波器进行最基本的操作,主要包括运行控制、全键盘锁定、错误队列和系统设置数据的操作。

:RUN**■ 命令格式:**

:RUN

■ 功能描述:

用于开始示波器波形采样工作，如需停止工作，需要执行:STOP 命令。

:STOP**■ 命令格式:**

:STOP

■ 功能描述:

用于停止示波器波形采样工作，如需恢复工作，需要执行:RUN 命令。

:AUTO**■ 命令格式:**

:AUTO

■ 功能描述:

自动设置示波器将根据输入信号自动调整垂直档位、水平时基以及触发方式，使波形显示达到最佳状态注意，应用自动设置要求被测信号的频率不小于 20 Hz，占空比大于 1%，且幅度至少为 10 mVpp。

注：当打开某些高级功能时，该命令不可用

:SYSTem:SINGle**■ 命令格式:**

:SYSTem:SINGle

■ 功能描述:

将示波器设置成单次触发方式。

:SYSTem:TFORce**■ 命令格式:**

:SYSTem:TFORce

■ 功能描述:

强制产生一个触发信号。强制触发适用于“普通”和“单次”触发方式。

:SYSTem:CALibration■ **命令格式:**

:SYSTem:CALibration

■ **功能描述:**

示波器自校正。

:SYSTem:CLEar■ **命令格式:**

:SYSTem:CLEar

■ **功能描述:**

清除示波器测量结果等。

:SYSTem:LANGuage■ **命令格式:**

:SYSTem:LANGuage <language>

:SYSTem:LANGuage ?

■ **功能描述:**

设置或查询软件语言类型。

■ **参数:**

language:离散型{SIMPLifiedchinese | ENGLish}

SIMPLifiedchinese:简体中文

ENGLish:英文

■ **返回格式:**

查询返回软件语言类型。

■ **举例:**

:SYSTem:LANGuage ENGLish 设置语言为英文

:SYSTem:LANGuage ? 查询返回 ENGLish

:SYSTem:PRTSc■ **命令格式:**

:SYSTem:PRTSc ?

■ **功能描述:**

截图。

■ 返回格式:

查询返回表示一张图片的字节数组。

读取的数据格式为 头+波形数据点+结束符。头为#NXXXXXX 的形式，#为规定的头标志符，N 表示后面含有 N 个字节，以 ASCII 字符的形式描述波形数据点的长度，结束符用于表示通讯的终止。例如，一次读取的数据为#9000001000XXXX 表示 9 个字节描述数据的长度，000001000 表示波形数据的长度，即 1000 字节。

:SYSTem:TOUChlock

■ 命令格式:

:SYSTem:TOUChlock <active>

:SYSTem:TOUChlock ?

■ 功能描述:

设置或查询触摸锁开关状态。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:SYSTem:TOUChlock ON 设置触摸锁打开

:SYSTem:TOUChlock ? 查询返回 1

:SYSTem:AUTO:TIMebase

■ 命令格式:

:SYSTem:AUTO:TIMebase <status>

:SYSTem:AUTO:TIMebase ?

■ 功能描述:

用于自动设置时，水平时基档位属性是否保持当前状态。

■ 参数:

status:{AUTO | KEEP}

AUTO:自动

KEEP:保持

■ 返回格式:

查询返回“AUTO”、“KEEP”。

■ **举例:**

:SYSTem:AUTO:TIMEbase KEEP	设置水平时基档位属性保持当前状态
:SYSTem:AUTO:TIMEbase ?	查询返回 KEEP

:SYSTem:AUTO:VERTical

■ **命令格式:**

```
:SYSTem:AUTO:VERTical <status>
:SYSTem:AUTO:VERTical ?
```

■ **功能描述:**

用于自动设置时，模拟通道垂直档位属性是否保持当前状态。

■ **参数:**

```
status:{AUTO| KEEP}
AUTO:自动
KEEP:保持
```

■ **返回格式:**

查询返回“AUTO”、“KEEP”。

■ **举例:**

:SYSTem:AUTO:VERTical KEEP	设置模拟通道垂直档位属性保持当前状态
:SYSTem:AUTO:VERTical ?	查询返回 KEEP

:SYSTem:AUTO:ACQuire

■ **命令格式:**

```
:SYSTem:AUTO:ACQuire <status>
:SYSTem:AUTO:ACQuire ?
```

■ **功能描述:**

用于自动设置时，采集模式是否保持当前状态。

■ **参数:**

```
status:{AUTO | KEEP}
AUTO:自动
KEEP:保持
```

■ **返回格式:**

查询返回“AUTO”、“KEEP”。

■ **举例:**

:SYSTem:AUTO:ACQuire KEEP	设置采集模式属性保持当前状态
---------------------------	----------------

:SYSTem:AUTO:ACQuire ?

查询返回 KEEP

:SYSTem:AUTO:TRIGger

■ 命令格式:

:SYSTem:AUTO:TRIGger <status>

:SYSTem:AUTO:TRIGger ?

■ 功能描述:

用于自动设置时，触发模式、触发源是否保持当前状态。

■ 参数:

status:{AUTO| KEEP}

AUTO:自动

KEEP:保持

■ 返回格式:

查询返回“AUTO”、“KEEP”。

■ 举例:

:SYSTem:AUTO:TRIGger KEEP

设置触发模式、触发源属性保持当前状态

:SYSTem:AUTO:TRIGger ?

查询返回 KEEP

:SYSTem:AUTO:CHANel:COUPling

■ 命令格式:

:SYSTem:AUTO:CHANel:COUPling <status>

:SYSTem:AUTO:CHANel:COUPling ?

■ 功能描述:

用于自动设置时，通道耦合是否保持当前状态。

■ 参数:

status:{AUTO | KEEP}

AUTO:自动

KEEP:保持

■ 返回格式:

查询返回“AUTO”、“KEEP”。

■ 举例:

:SYSTem:AUTO:CHANel:COUPling KEEP

设置通道耦合保持当前状态

:SYSTem:AUTO:CHANel:COUPling ?

查询返回 KEEP

:SYSTem:AUTO:CHANel:ACTive**■ 命令格式:**

:SYSTem:AUTO:CHANel:ACTive <status>

:SYSTem:AUTO:CHANel:ACTive ?

■ 功能描述:

用于自动设置时，通道打开或关闭是否保持当前状态。

■ 参数:

status:{AUTO| KEEP}

AUTO:自动

KEEP:保持

■ 返回格式:

查询返回“AUTO”、“KEEP”。

■ 举例:

:SYSTem:AUTO:CHANel:ACTive KEEP

设置通道开关保持当前状态

:SYSTem:AUTO:CHANel:ACTive ?

查询返回 KEEP

:SYSTem:AUTO:DISPlay:MODE**■ 命令格式:**

:SYSTem:AUTO:DISPlay:MODE <status>

:SYSTem:AUTO:DISPlay:MODE ?

■ 功能描述:

用于自动设置时，通道是否堆叠显示。

■ 参数:

status:{AUTO | KEEP}

AUTO:自动平铺显示

KEEP:保持堆叠显示

■ 返回格式:

查询返回“AUTO”、“KEEP”。

■ 举例:

:SYSTem:AUTO:DISPlay:MODE KEEP

设置通道堆叠显示

:SYSTem:AUTO:DISPlay:MODE ?

查询返回 KEEP

采集

:ACQuire:STATus

- **命令格式:**

:ACQuire:STATus ?

- **功能描述:**

查询采样状态。

- **返回格式:**

查询返回“STOP”、“ARMED”、“READY”、“TRIGED”、“AUTO”、“ROLL”、“RESET”。

STOP:停止

ARMED:已配备

READY:就绪

TRIGED:已触发

AUTO:自动

ROLL:扫描

RESET:重置

- **举例:**

:ACQuire:STATus ?

查询返回 READY

:ACQuire:TYPe

- **命令格式:**

:ACQuire:TYPe <type>

:ACQuire:TYPe ?

- **功能描述:**

设置或查询采样模式。

- **参数:**

type:{NORMal | PEAK | HIGHres | AVERage | ENvelope}

NORMal:普通

PEAK:峰值检测

HIGHres:高分辨率

AVERage:平均

ENvelope:包络

- **返回格式:**

查询返回“NORMal”、“PEAK”、“HIGHres”、“AVERage”、“ENVELOpe”。

■ **举例:**

:ACQUIRE:TYPE NORMal	设置采样模式为普通
:ACQUIRE:TYPE ?	查询返回 NORMal

:ACQUIRE:AVERage

■ **命令格式:**

:ACQUIRE:AVERage <count>
:ACQUIRE:AVERage ?

■ **功能描述:**

设置或查询平均采样模式下的平均次数。

■ **参数:**

count:整数型, 范围 2~65536。

■ **返回格式:**

查询返回一个表示平均次数的整数。

■ **举例:**

:ACQUIRE:AVERage 20	设置平均次数为 20 次
:ACQUIRE:AVERage ?	查询返回 20

:ACQUIRE:ENVELOpe

■ **命令格式:**

:ACQUIRE:ENVELOpe <count>
:ACQUIRE:ENVELOpe ?

■ **功能描述:**

设置或查询包络采样模式下的包络次数。

■ **参数:**

count:整数型, 范围 2~65536。

■ **返回格式:**

查询返回包络采样模式下包络次数的整数。

■ **举例:**

:ACQUIRE:ENVELOpe 30	设置包络次数为 30 次
:ACQUIRE:ENVELOpe ?	查询返回 30

:ACquire:INTerpolation:MODE**■ 命令格式:**

:ACquire:INTerpolation:MODE <mode>

:ACquire:INTerpolation:MODE ?

■ 功能描述:

设置或查询插值方式。

■ 参数:

mode:{LINE | SINX}

LINE:线性

SINX:正弦

■ 返回格式:

查询返回“LINE”、“SINX”。

■ 举例:

:ACquire:INTerpolation:MODE LINE

设置插值方式为线性

:ACquire:INTerpolation:MODE ?

查询返回 LINE

:ACquire:CLKSrc**■ 命令格式:**

:ACquire:CLKSrc <clksrc>

:ACquire:CLKSrc ?

■ 功能描述:

设置或查询 10MHz 时钟信号来源。

■ 参数:

clksrc:{INNER | OUTER}

INNER:内部

OUTER:外部

■ 返回格式:

查询返回“INNER”、“OUTER”。

■ 举例:

:ACquire:CLKSrc OUTER

设置 10MHz 时钟信号来源为“外部”

:ACquire:CLKSrc ?

查询返回 OUTER

:ACQuire:EXTLocked■ **命令格式:**

:ACQuire:EXTLocked ?

■ **功能描述:**

查询外部输入 10MHz 时钟频率是否锁定。

■ **返回格式:**

查询返回“1”、“0”。

1:锁定

0:未锁定

:ACQuire:FAST■ **命令格式:**

:ACQuire:FAST <active>

:ACQuire:FAST ?

■ **功能描述:**

设置或查询快采的开关状态。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

ON:开启

OFF:关闭

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:ACQuire:FAST ON 设置快采打开

:ACQuire:FAST ? 查询返回 1

:ACQuire:STORage■ **命令格式:**

:ACQuire:STORage <type>

:ACQuire:STORage ?

■ **功能描述:**

设置或查询存储深度。

■ 参数:

type:离散型, 默认单位 Pts。

选择 AUTO 时, 示波器根据当前的采样率自动选择存储深度。

10G 模式下: {50k | 500k | 5M | 50M | 500M | 1G}

20G 模式下: {100k | 1M | 10M | 100M | 1G | 2G}

■ 返回格式:

查询返回实际存储深度。

■ 举例:

:ACQUIRE:STORAGE 50k 设置存储深度 50kPts

:ACQUIRE:STORAGE ? 查询返回 500000E+004

:ACQUIRE:ERES:ENABLE**■ 命令格式:**

:ACQUIRE:ERES:ENABLE <enable>

:ACQUIRE:ERES:ENABLE ?

■ 功能描述:

设置或查询增强分辨率打开状态。

■ 参数:

enable:布尔型{ON | OFF}或{1 | 0}

ON:开启

OFF:关闭

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:ACQUIRE:ERES:ENABLE ON 设置增强分辨率打开

:ACQUIRE:ERES:ENABLE ? 查询返回 1

:ACQUIRE:ERES:BIT**■ 命令格式:**

:ACQUIRE:ERES:BIT <bit>

:ACQUIRE:ERES:BIT ?

■ 功能描述:

设置或查询增强分辨率增强位数。

- **参数:**

bit:实数型, 范围 0.5~3, 步进为 0.5

- **返回格式:**

查询返回增强分辨率增强位数。

- **举例:**

:ACquire:ERES:BIT 1	设置增强分辨率增强位数 1
:ACquire:ERES:BIT ?	查询返回 1.00000E+000

任意波形发生器

:AWG<n>:LOAD

- **命令格式:**

:AWG<n>:LOAD <load>

:AWG<n>:LOAD ?

- **功能描述:**

设置或查询指定通道的负载阻抗。

- **参数:**

load:离散型{HIGH | LOW}

HIGH:1MΩ

LOW:50Ω

- **返回格式:**

查询返回“HIGH”、“LOW”。

- **举例:**

:AWG1:LOAD HIGH	设置通道 1 的负载阻抗为高阻
:AWG1:LOAD ?	查询返回通道 1 的负载阻抗为 HIGH

:AWG<n>:ENABle

- **命令格式:**

:AWG<n>:ENABle <enable>

:AWG<n>:ENABle ?

- **功能描述:**

设置或查询打开或关闭指定 AWG 通道的输出。

- **参数:**

enable:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:AWG1:ENABle ON 设置通道 1 输出打开

:AWG1:ENABle ? 查询返回 1

:AWG<n>:MODE

■ **命令格式:**

:AWG<n>:MODE <mode>

:AWG<n>:MODE ?

■ **功能描述:**

设置或查询指定 AWG 通道的输出模式。

■ **参数:**

mode:{CONTinuous | MODulation | SWEep}

CONTinuous:连续波

MODulation:调制波

SWEep:扫频

■ **返回格式:**

查询返回“CONTinuous”、“MODulation”、“SWEep”。

■ **举例:**

:AWG1:MODE MODulation 设置 AWG1 的输出模式为调制波

:AWG1:MODE ? 查询返回 MODulation

连续波

:AWG<n>:BASe:WAVE

■ **命令格式:**

:AWG<n>:BASe:WAVE <type>

:AWG<n>:BASe:WAVE ?

■ **功能描述:**

设置或查询指定通道的基本波形类型。

■ **参数:**

type:离散型{SINusoid | SQUare | PULSe | RAMP | NOISe | DC | SINCl EXPRIse | EXPFall |

LOREntz | HAVersine | GAUSSian | ECG | ARbitrary}

SINusoid:正弦波

SQUare:方波

PULSe:脉冲波

RAMP:斜坡

NOISe:噪声

DC:直流

SINC:Sinc

EXPRise:指数上升

EXPFall:指数下降

LOREntz:洛伦兹

HAVersine:半正矢

GAUSSian:高斯

ECG:心电图

ARbitrary:任意波

■ 返回格式:

查询返回“SINusoid”、“SQUare”、“PULSe”、“RAMP”、“NOISe”、“DC”、“SINC”、“EXPRise”、“EXPFall”、“LOREntz”、“HAVersine”、“GAUSSian”、“ECG”、“ARbitrary”。

■ 举例:

:AWG1:BASe:WAVE SINusoid

设置通道 1 的基本波形为正弦波

:AWG1:BASe:WAVE ?

查询返回通道 1 的基本波形为 SINusoid

:AWG<n>:BASe:FREQuency

■ 命令格式:

:AWG<n>:BASe:FREQuency <freq>

:AWG<n>:BASe:FREQuency ?

■ 功能描述:

设置或查询指定通道的基本波形频率。

■ 参数:

freq:实数型, 单位:uHz,mHz,Hz,kHz,MHz, 不输入单位则默认单位为 Hz

■ 返回格式:

查询返回指定通道的基本波形频率。

■ 举例:

:AWG1:BASe:FREQuency 20 设置通道 1 的波形频率为 20Hz
:AWG1:BASe:FREQuency ? 查询返回通道 1 的波形频率为 2.0000000000000E+001

:AWG<n>:BASe:PHASe

■ 命令格式:

:AWG<n>:BASe:PHASe <phase>
:AWG<n>:BASe:PHASe ?

■ 功能描述:

设置或查询指定通道的输出相位。

■ 参数:

phase:实数型,表示相位,单位°,范围 0°~359.99°

■ 返回格式:

查询返回指定通道的基本波形的相位。

■ 举例:

:AWG1:BASe:PHASe 20 设置通道 1 的波形相位为 20°
:AWG1:BASe:PHASe ? 查询返回通道 1 的波形相位为 2.00000E+001

:AWG<n>:BASe:AMPLitude

■ 命令格式:

:AWG<n>:BASe:AMPLitude <amp>
:AWG<n>:BASe:AMPLitude ?

■ 功能描述:

设置或查询指定通道的输出幅度。

■ 参数:

amp:实数型,表示电压,单位 mVpp,Vpp,不输入单位则默认单位为 Vpp。高阻输出范围:20mVpp ~ 6Vpp;低阻输出范围:10mVpp ~ 3Vpp。

■ 返回格式:

查询返回指定通道的基本波形的幅度。

■ 举例:

:AWG1:BASe:AMPLitude 2 设置通道 1 的波形幅度为 2Vpp
:AWG1:BASe:AMPLitude ? 查询返回通道 1 的波形幅度为 2.00000E+000

:AWG<n>:BASe:OFFSet

■ 命令格式:

:AWG<n>:BAsE:OFFSet <offset>

:AWG<n>:BAsE:OFFSet ?

■ **功能描述:**

设置或查询指定通道的输出偏移。

■ **参数:**

offset:实数型,表示电压,单位 mV,V,不输入单位则默认单位为 V。高阻输出范围:-3V ~ 3V; 低阻输出范围:-1.5V ~ 1.5V。

■ **返回格式:**

查询返回指定通道的基本波形的输出偏移。

■ **举例:**

:AWG1:BAsE:OFFSet 2

设置通道 1 的波形偏置为 2V

:AWG1:BAsE:OFFSet ?

查询返回通道 1 的波形偏置为 2.00000E+000

:AWG<n>:BAsE:HIGH

■ **命令格式:**

:AWG<n>:BAsE:HIGH <voltage>

:AWG<n>:BAsE:HIGH ?

■ **功能描述:**

设置或查询指定通道信号输出高值。

■ **参数:**

voltage:实数型,表示电压,单位 mV,V,不输入单位则默认单位为 V。高阻输出范围:-3V ~ 3V; 低阻输出范围:-1.5V ~ 1.5V。

■ **返回格式:**

查询返回指定通道信号输出高值。

■ **举例:**

:AWG1:BAsE:HIGH 1

设置通道 1 的信号输出高值为 1V

:AWG1:BAsE:HIGH ?

查询返回通道 1 的信号输出高值为 1.00000E+000

:AWG<n>:BAsE:LOW

■ **命令格式:**

:AWG<n>:BAsE:LOW <voltage>

:AWG<n>:BAsE:LOW ?

■ **功能描述:**

设置或查询指定通道信号输出低值。

■ **参数:**

voltage:实数型,表示电压,单位 V,单位 mV,V,不输入单位则默认单位为 V。高阻输出范围:-3V ~ 3V;低阻输出范围:-1.5V ~ 1.5V。

■ **返回格式:**

查询返回指定通道信号输出低值。

■ **举例:**

:AWG1:BASe:LOW 1

设置通道 1 的信号输出低值为 1V

:AWG1:BASe:LOW ?

查询返回通道 1 的信号输出低值为 1.00000E+000

:AWG<n>:PULSe:RISe

■ **命令格式:**

:AWG<n>:PULSe:RISe <time>

:AWG<n>:PULSe:RISe ?

■ **功能描述:**

设置或查询指定通道脉冲波的上升沿时间。

■ **参数:**

time:实数型,单位 ns,us,默认单位为 s。范围 5ns~399.999us。

■ **返回格式:**

查询返回指定通道脉冲波的上升沿时间。

■ **举例:**

:AWG1:PULSe:RISe 10ns

设置通道 1 的脉冲波上升时间为 10ns

:AWG1:PULSe:RISe ?

查询返回 1.00000000000000E-008

说明: 该指令仅在脉冲波下使用

:AWG<n>:PULSe:FALL

■ **命令格式:**

:AWG<n>:PULSe:FALL <time>

:AWG<n>:PULSe:FALL ?

■ **功能描述:**

设置或查询指定通道脉冲波的下降沿时间。

■ **参数:**

time:实数型,单位 ns,us,默认单位为 s。范围 5ns~399.999us。

■ **返回格式:**

查询返回指定通道脉冲波的下降沿时间。

■ 举例:

:AWG1:PULSe:FALL 10ns	设置通道 1 的脉冲波下降时间为 10ns
:AWG1:PULSe:FALL ?	查询返回 1.00000000000000E-008

说明: 该指令仅在脉冲波下使用

:AWG<n>:BAsE:DUTY**■ 命令格式:**

:AWG<n>:BAsE:DUTY <duty>
:AWG<n>:BAsE:DUTY ?

■ 功能描述:

设置或查询指定通道信号输出占空比。

■ 参数:

Duty:实数型,表示占空比, 单位%, 范围 0~100。

■ 返回格式:

查询返回指定通道信号输出占空比。

■ 举例:

:AWG1:BAsE:DUTY 50	设置通道 1 的信号输出占空比为 50%
:AWG1:BAsE:DUTY ?	查询返回通道 1 的信号输出占空比 5.00000E+001

说明: 该指令仅在方波和脉冲波下使用

:AWG<n>:BAsE:SYMMetry**■ 命令格式:**

:AWG<n>:BAsE:SYMMetry <duty>
:AWG<n>:BAsE:SYMMetry ?

■ 功能描述:

设置或查询指定通道信号输出对称性。

■ 参数:

Duty:实数型,表示占空比, 单位%, 范围 0~100。

■ 返回格式:

查询返回指定通道信号输出对称性。

■ 举例:

:AWG1:BAsE:SYMMetry 50	设置通道 1 的信号输出对称性为 50%
:AWG1:BAsE:SYMMetry ?	查询返回通道 1 的信号输出对称性 5.00000E+001

说明: 该指令仅在斜波下使用

:AWG<n>:BAsE:ARBitrary**■ 命令格式:**

:AWG<n>:BAsE:ARBitrary path

:AWG<n>:BAsE:ARBitrary ?

■ 功能描述:

设置或查询指定通道任意波文件读取路径。

■ 参数:

path:AWG 任意波打开路径

■ 返回格式:

查询返回指定通道当前任意波文件的路径。

■ 举例:

:AWG1:BAsE:ARBitrary "D://U2 DSO//WaveForm//Uni-t000.bin"

设置通道 1 任意波文件读取路径为 D://U2 DSO//WaveForm//Uni-t000.bin

:AWG1:BAsE:ARBitrary ?

查询返回 D://U2 DSO//WaveForm//Uni-t000.bin

:AWG<n>:BAsE:PPOint**■ 命令格式:**

:AWG<n>:BAsE:PPOint <active>

:AWG<n>:BAsE:PPOint ?

■ 功能描述:

设置或查询指定通道任意波逐点输出开关。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回指定通道任意波逐点输出开关状态。

■ 举例:

:AWG1:BAsE:PPOint ON

设置通道 1 的任意逐点输出打开

:AWG1:BAsE:PPOint ?

查询返回 1

说明: 该指令仅在任意波下使用

调制波

:AWG<n>:MODulate:MODE

■ 命令格式:

:AWG<n>:MODulate:MODE <mode>

:AWG<n>:MODulate:MODE ?

■ 功能描述:

设置或查询指定通道信号调制模式。

■ 参数:

mode:离散型{AM | PM | FM}

■ 返回格式:

查询返回“AM”、“PM”、“FM”。

■ 举例:

:AWG1:MODulate:MODE AM

设置通道 1 信号为调幅模式

:AWG1:MODulate:MODE ?

查询返回 AM

:AWG<n>:MODulate:WAVE

■ 命令格式:

:AWG<n>:MODulate:WAVE <type>

:AWG<n>:MODulate:WAVE ?

■ 功能描述:

设置或查询指定通道信号调制波形。

■ 参数:

type:离散型{SINusoid | SQUare | RAMP | NOISe | ARBitrary}

SINusoid:正弦波

SQUare:方波

RAMP:斜波

NOISe:噪声

ARBitrary:任意波

■ 返回格式:

查询返回“SINusoid”、“SQUare”、“RAMP”、“NOISe”、“ARBitrary”。

■ 举例:

:AWG1:MODulate:WAVE SINusoid

设置通道 1 信号调制波类型为正弦波

:AWG1:MODulate:WAVE ?

查询返回 SINusoid

:AWG<n>:MODulate:FREQuency

■ 命令格式:

:AWG<n>:MODulate:FREQuency <freq>

:AWG<n>:MODulate:FREQuency ?

■ 功能描述:

设置或查询指定通道信号调制频率。

■ 参数:

Freq:实数型, 表示频率, 单位 mHz,Hz,kHz, 不输入单位则默认单位为 Hz。

■ 返回格式:

查询返回指定通道信号的调制频率。

■ 举例:

:AWG1:MODulate:FREQuency 2000

设置通道 1 信号调制频率为 2KHz

:AWG1:MODulate:FREQuency ?

查询返回 2.00000000000000E+003

:AWG<n>:FM:FREQuency:DEV

■ 命令格式:

:AWG<n>:FM:FREQuency DEV <freq>

:AWG<n>:FM:FREQuency ?

■ 功能描述:

设置或查询指定通道波形的频率偏移。

■ 参数:

freq:实数型, 100mHz~12.5MHz, 单位 Hz

■ 返回格式:

查询返回指定通道波形的频率偏移。

■ 举例:

:AWG1:FM:FREQuency:DEV 2

设置通道 1 频率偏移为 2Hz

:AWG1:FM:FREQuency:DEV ?

查询返回通道 1 频率偏移为 2.00000000000000E+000

:AWG<n>:MODulate:DEPTh

■ 命令格式:

:AWG<n>:MODulate:DEPTh <depth>

:AWG<n>:MODulate:DEPTh ?

■ **功能描述:**

设置或查询指定通道信号调制深度。

■ **参数:**

depth:实数型, 表示调制深度, 单位%。AM 调制深度为 1%~120%

■ **返回格式:**

查询返回指定通道信号的调制深度。

■ **举例:**

:AWG1:MODulate:DEPT h 50

设置通道 1 信号调制深度为 50%

:AWG1:MODulate:DEPT h ?

查询返回 5.00000E+001

:AWG<n>:PM:PHASe:DEV

■ **命令格式:**

:AWG<n>:PM:PHASe:DEV <value>

:AWG<n>:PM:PHASe:DEV ?

■ **功能描述:**

设置或查询指定通道信号相位调制相偏。

■ **参数:**

value:实数型, 范围 0°~359.99°, 单位 ° (度)。

■ **返回格式:**

查询返回指定通道信号相位调制相偏。

■ **举例:**

:AWG1:PM:PHASe:DEV 50

设置通道 1 信号调制相偏为 50°

:AWG1:PM:PHASe:DEV ?

查询返回 5.00000E+001

扫频

:AWG<n>:SWEep:TYPe

■ **命令格式:**

:AWG<n>:SWEep:TYPe <type>

:AWG<n>:SWEep:TYPe ?

■ **功能描述:**

设置或查询指定通道扫频类型。

■ **参数:**

type:{LINear | LOGarithmic}

LINear :线性

LOGarithmic:对数

■ **返回格式:**

查询返回“LINear ”、“LOGarithmic”。

■ **举例:**

:AWG1:SWEep:TYPe LOGarithmic

设置通道 1 扫频类型为对数

:AWG1:SWEep:TYPe ?

查询返回 LOGarithmic

:AWG<n>:SWEep:DURatio

■ **命令格式:**

:AWG<n>:SWEep:DURatio <time>

:AWG<n>:SWEep:DURatio ?

■ **功能描述:**

设置或查询指定通道扫频时间。

■ **参数:**

time:实数型, 单位 ms, s, 默认单位 s。

■ **返回格式:**

查询返回指定通道扫频时间。

■ **举例:**

:AWG1:SWEep:DURatio 1

设置通道 1 扫频时间 1s

:AWG1:SWEep:DURatio ?

查询返回 1.00000000000000E+000

:AWG<n>:SWEep:FREQuency:STARt

■ **命令格式:**

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

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

■ **功能描述:**

设置或查询指定通道扫频起始频率。

■ **参数:**

freq:实数型, 单位 mHz、Hz、kHz、MHz, 默认单位 Hz。

■ **返回格式:**

查询返回指定通道扫频起始频率。

■ **举例:**

:AWG1:SWEep:FREQuency:STARt 1

设置通道 1 扫频起始频率 1Hz

:AWG1:SWEep:FREQuency:START ?

查询返回 1.0000000000000E+000

:AWG<n>:SWEep:FREQuency:END

■ 命令格式:

:AWG<n>:SWEep:FREQuency:END <freq>

:AWG<n>:SWEep:FREQuency:END ?

■ 功能描述:

设置或查询指定通道扫频结束频率。

■ 参数:

freq:实数型, 单位 mHz、Hz、kHz、MHz, 默认单位 Hz。

■ 返回格式:

查询返回指定通道扫频结束频率。

■ 举例:

:AWG1:SWEep:FREQuency:END 1

设置通道 1 扫频结束频率 1Hz

:AWG1:SWEep:FREQuency:END ?

查询返回 1.0000000000000E+000

:AWG<n>:SWEep:TRIGger:SOURce

■ 命令格式:

:AWG<n>:SWEep:TRIGger:SOURce <source>

:AWG<n>:SWEep:TRIGger:SOURce ?

■ 功能描述:

设置或查询指定通道扫频触发源。

■ 参数:

source:{INSide | MANual | OUTSide}

INSide:内部

MANual:手动

OUTSide:外部

■ 返回格式:

查询返回指定通道扫频触发源。

■ 举例:

:AWG1:SWEep:TRIGger:SOURce OUTSide

设置通道 1 扫频触发源为外部

:AWG1:SWEep:TRIGger:SOURce ?

查询返回 OUTSide

:AWG<n>:SWEep:TRIGger:ENABle**■ 命令格式:**

:AWG<n>:SWEep:TRIGger:ENABle <enable>

:AWG<n>:SWEep:TRIGger:ENABle ?

■ 功能描述:

设置或查询指定通道扫频触发输出的开启或关闭。

■ 参数:

enable:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:AWG1:SWEep:TRIGger:ENABle OFF

设置通道 1 扫频触发输出关闭

:AWG1:SWEep:TRIGger:ENABle ?

查询返回 0

:AWG<n>:SWEep:TRIGger:MANualtrig**■ 命令格式:**

:AWG<n>:SWEep:TRIGger:MANualtrig

■ 功能描述:

设置指定通道扫频触发源为“手动”时触发一次。

通道**:CHANnel<n>:BWLimit****■ 命令格式:**

:CHANnel<n>:BWLimit <bwlimit>

:CHANnel<n>:BWLimit ?

■ 功能描述:

设置或查询通道的带宽限制。

■ 参数:

bwlimit:离散型{FULL | 20MHz}

FULL:带宽限制为满带宽

20MHz:带宽限制为 20MHz

■ **返回格式:**

查询返回“FULL”、“20MHz”。

■ **举例:**

:CHANnel1:BWLimit FULL

设置通道 1 带宽限制为满带宽

:CHANnel1:BWLimit ?

查询返回 FULL

:CHANnel<n>:COUPling

■ **命令格式:**

:CHANnel<n>:COUPling <coupling>

:CHANnel<n>:COUPling ?

■ **功能描述:**

设置或查询通道输入的耦合方式为“DC1M Ω ”、“AC1M Ω ”、“DC50 Ω ”、“GND”。

注: MSO8804HD 型号不支持 DC1M Ω 和 AC1M Ω 耦合方式。

■ **参数:**

coupling:离散型{DC1M | AC1M | DC50 | GND}

DC:耦合方式为直流

AC:耦合方式为交流

■ **返回格式:**

查询返回“DC1M”、“AC1M”、“DC50”、“GND”。

■ **举例:**

:CHANnel1:COUPling DC1M

设置通道 1 耦合方式为 DC1M Ω

:CHANnel1:COUPling ?

查询返回 DC1M

:CHANnel<n>:DISPlay

■ **命令格式:**

:CHANnel<n>:DISPlay <active>

:CHANnel<n>:DISPlay ?

■ **功能描述:**

关闭或打开通道的显示。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:CHANnel1:DISPlay ON	打开通道 1 显示
:CHANnel1:DISPlay ?	查询返回 1

:CHANnel<n>:INVert**■ 命令格式:**

:CHANnel<n>:INVert <invert>
:CHANnel<n>:INVert ?

■ 功能描述:

关闭或打开通道的反相显示。

■ 参数:

invert:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:CHANnel1:INVert ON	打开通道 1 反相显示
:CHANnel1:INVert ?	查询返回 1

:CHANnel<n>:OFFSet**■ 命令格式:**

:CHANnel<n>:OFFSet <offset>
:CHANnel<n>:OFFSet ?

■ 功能描述:

设置或查询指定通道波形显示的垂直偏移。

■ 参数:

offset:实数型, -5.5~5.5, 居中为 0, 单位 div

■ 返回格式:

查询返回指定通道的垂直偏移。

■ 举例:

:CHANnel1:OFFSet 1	设置通道 1 垂直偏移 1div
:CHANnel1:OFFSet ?	查询返回 1.00000E+000

:CHANnel<n>:PROBe:MODeI**■ 命令格式:**

:CHANnel<n>:PROBe:MODeI ?

■ 功能描述:

查询探头型号。

:CHANnel<n>:PROBe:SERailnumber**■ 命令格式:**

:CHANnel<n>:PROBe:SERailnumber ?

■ 功能描述:

查询探头序列号。

:CHANnel<n>:PROBe:CURRent**■ 命令格式:**

:CHANnel<n>:PROBe:CURRent <probe>

:CHANnel<n>:PROBe:CURRent ?

■ 功能描述:

设置或查询探头衰减比。

■ 参数:

probe:离散型{1 | 10 | 100 | CUSTom}

1:探头衰减比为 1

10:探头衰减比为 10

100:探头衰减比为 100

CUSTom:自定义

■ 返回格式:

查询返回探头衰减比。

■ 举例:

:CHANnel1:PROBe:CURRent 10

设置通道 1 探头衰减比为 10

:CHANnel1:PROBe:CURRent ?

查询返回 10

:CHANnel<n>:PROBe:CUSTom**■ 命令格式:**

:CHANnel<n>:PROBe:CUSTom <count>

:CHANnel<n>:PROBe:CUSTom ?

■ **功能描述:**

设置或查询自定义探头倍率。

■ **参数:**

count:实数型, 设置范围 0.001X-1000X

■ **返回格式:**

查询返回自定义的探头倍率。

■ **举例:**

:CHANnel1:PROBe:CUSTom 10

设置通道 1 探头倍率为 10

:CHANnel1:PROBe:CUSTom ?

查询返回 1.00000E+001

:CHANnel<n>:PROBe:UNIT

■ **命令格式:**

:CHANnel<n>:PROBe:UNIT <unit>

:CHANnel<n>:PROBe:UNIT ?

■ **功能描述:**

设置或查询探头单位。

■ **参数:**

unit:字符串

■ **返回格式:**

查询返回探头单位字符串。

■ **举例:**

:CHANnel1:PROBe:UNIT "V"

设置通道 1 探头单位为 V

:CHANnel1:PROBe:UNIT ?

查询返回 V

:CHANnel<n>:PROBe:BUTTon

■ **命令格式:**

:CHANnel<n>:PROBe:BUTTon <button>

:CHANnel<n>:PROBe:BUTTon ?

■ **功能描述:**

设置或查询探头按钮类型。

■ **参数:**

button:离散型{HEADlight | RUNorstop | CLEAr | FORCetrig | NOOPs}

HEADlight:前灯

RUNorstop:运行/停止

CLEAr:清除

FORCetrig 强制触发:

NOOPs:无操作

■ 返回格式:

查询返回“HEADlight”、“RUNorstop”、“CLEAr”、“FORCetrig”、“NOOPs”。

■ 举例:

:CHANnel1:PROBe:BUtTon NOOPs

设置通道 1 探头按钮类型为无操作

:CHANnel1:PROBe:BUtTon ?

查询返回 NOOPs

:CHANnel<n>:SCALe

■ 命令格式:

:CHANnel<n>:SCALe <scale>

:CHANnel<n>:SCALe ?

■ 功能描述:

设置或查询指定通道波形显示的垂直档位。

■ 参数:

scale:实数型, 单位:自定义, 不输入单位则默认单位为 V

当输入阻抗为 1MΩ时, 可调范围在 1mV ~10V

当输入阻抗为 50Ω时, 可调范围在 1mV ~1V

■ 返回格式:

查询返回以科学计数形式的垂直档位值, 默认单位 V。

■ 举例:

:CHANnel1:SCALe 1V

设置通道 1 波形显示的垂直档位为 1V

:CHANnel1:SCALe ?

查询返回 1.00000E+000

:CHANnel<n>:FINE

■ 命令格式:

:CHANnel<n>:FINE <active>

:CHANnel<n>:FINE ?

■ 功能描述:

设置或查询指定通道的垂直档位细调是否打开。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:CHANnel1:FINE ON

设置通道 1 的垂直档位细调打开

:CHANnel1:FINE ?

查询返回 1

:CHANnel<n>:UNIT

■ **命令格式:**

:CHANnel<n>:UNIT <unit>

:CHANnel<n>:UNIT ?

■ **功能描述:**

设置或查询指定通道的幅度显示单位。

■ **参数:**

unit:字符串

■ **返回格式:**

查询返回默认单位 V 或用户自定义单位。

■ **举例:**

:CHANnel1:UNIT "V"

设置通道 1 波形显示的幅度单位为 V

:CHANnel1:UNIT ?

查询返回 V

:CHANnel<n>:BIAS

■ **命令格式:**

:CHANnel<n>:BIAS <bias>

:CHANnel<n>:BIAS ?

■ **功能描述:**

设置或查询偏置电压。

■ **参数:**

bias:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V

■ **返回格式:**

查询返回以科学计数形式的偏置电压值。

■ 举例:

:CHANnel1:BIAS 1V

设置通道 1 波形的偏置电压为 1V

:CHANnel1:BIAS ?

查询返回 1.00000E+000

:CHANnel<n>:LABel

■ 命令格式:

:CHANnel<n>:LABel <label>

:CHANnel<n>:LABel ?

■ 功能描述:

设置或查询通道标签。

■ 参数:

label:通道名称字符串

■ 返回格式:

查询返回通道名称字符串。

■ 举例:

:CHANnel1:LABel "abcd"

设置通道 1 标签为 abcd

:CHANnel1:LABel ?

查询返回 abcd

光标**:CURSor:ACTive**

■ 命令格式:

:CURSor:ACTive <active>

:CURSor:ACTive ?

■ 功能描述:

设置或查询光标是否打开。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:CURSor:ACTive ON

设置打开光标

:CURSor:ACTive ?

查询返回 1

:CURSor:TYPe

■ 命令格式:

:CURSor:TYPe <type>

:CURSor:TYPe ?

■ 功能描述:

设置或查询光标模式的光标类型。

■ 参数:

type:离散型{AMPLitude | TIme | SCReen | CLOSe}

AMPLitude:幅度（水平）

TIme:时间（垂直）

SCReen:屏幕（水平&垂直）

CLOSe:关闭

■ 返回格式:

查询返回“AMPLitude”、“TIme”、“SCReen”、“CLOSe”。

■ 举例:

:CURSor:TYPe AMPLitude 设置光标类型为幅度

:CURSor:TYPe ? 查询返回 AMPLitude

:CURSor:SOURce

■ 命令格式:

:CURSor:SOURce <source>

:CURSor:SOURce ?

■ 功能描述:

设置或查询光标模式的源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”、“M1”、“M2”、“M3”、“M4”、“M5”、“M6”、“M7”、“M8”、“R1”、“R2”、“R3”、“R4”。

■ 举例:

:CURSor:SOURce C1	设置光标测量源为通道 1
:CURSor:SOURce ?	查询返回 C1

:CURSor:MOVE:SYNC

■ 命令格式:

```
:CURSor:MOVE:SYNC <enable>
:CURSor:MOVE:SYNC ?
```

■ 功能描述:

设置或查询光标的同步移动开启或关闭。

■ 参数:

enable:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:CURSor:MOVE:SYNC ON	设置光标的同步移动开启
:CURSor:MOVE:SYNC ?	查询返回 1

:CURSor:UNIT:VCURsor

■ 命令格式:

```
:CURSor:UNIT:VCURsor <unit>
:CURSor:UNIT:VCURsor ?
```

■ 功能描述:

设置或查询垂直光标单位。

■ 参数:

unit:{SECond | DEGRee | PERCent}

SECond:秒

DEGRee:度

PERCent:百分比

■ 返回格式:

查询返回“SECond”、“DEGRee”、“PERCent”。

■ 举例:

:CURSor:UNIT:VCURsor SECond	设置垂直光标单位为秒
:CURSor:UNIT:VCURsor ?	查询返回 SECond

:CURSor:UNIT:HCURsor■ **命令格式:**

:CURSor:UNIT:HCURsor <unit>

:CURSor:UNIT:HCURsor ?

■ **功能描述:**

设置或查询水平光标单位。

■ **参数:**

unit:{BAsE | PERCent}

BAsE:基

PERCent:百分比

说明: BAsE 取决于光标所在通道的单位。

■ **返回格式:**

查询返回“BAsE”、“PERCent”。

■ **举例:**

:CURSor:UNIT:HCURsor BAsE 设置水平光标单位为基

:CURSor:UNIT:HCURsor ? 查询返回 BAsE

:CURSor:CAX■ **命令格式:**

:CURSor:CAX <value>

:CURSor:CAX ?

■ **功能描述:**

设置或查询垂直光标 A 位置。

■ **参数:**

value:实数型, 以 div 为单位, 调节范围 0~10。

■ **返回格式:**

查询返回垂直光标线 A 的位置。

■ **举例:**

:CURSor:CAX 3 设置垂直光标 A 的位置 3div

:CURSor:CAX ? 查询返回 3.00000E+000

:CURSor:CBX

■ 命令格式:

```
:CURSor:CBX <value>
```

```
:CURSor:CBX ?
```

■ 功能描述:

设置或查询垂直光标 B 位置。

■ 参数:

value:实数型，以 div 为单位，调节范围 0~10。

■ 返回格式:

查询返回垂直光标线 B 的位置。

■ 举例:

```
:CURSOR:CBX 3          设置垂直光标 B 的位置 3div
```

:CURSor:CBX ? 查询返回 3.00000E+000

:CURSor:CAY

■ 命令格式:

```
:CURSor:CAY <value>
```

```
:CURSor:CAY ?
```

■ 功能描述:

设置或查询水平光标 A 位置。

■ **参数:**

value:实数型, 以 div 为单位, 调节范围-4~4。

■ 返回格式:

查询返回水平光标线 A 的位置。

■ 举例:

```
:CURSOR: CAY 3
```

:CURSor:CAY ? 查询返回 3.00000E+000

:CURSor:CBY

■ 命令格式:

```
:CURSor:CBY <value>
```

```
:CURSOR:CBY ?
```

■ 功能描述:

设置或查询水平光标 B 位置。

■ 参数:

value:实数型, 以 div 为单位, 调节范围-4~4。

■ 返回格式:

查询返回水平光标线 B 的位置。

■ 举例:

:CURSor:CBY 3

设置水平光标 B 的位置 3div

:CURSor:CBY ?

查询返回 3.00000E+000

:CURSor:AXValue**■ 命令格式:**

:CURSor:AXValue ?

■ 功能描述:

查询垂直光标 A 的测量值。

■ 返回格式:

查询返回垂直光标 A 的测量值。

:CURSor:BXValue**■ 命令格式:**

:CURSor:BXValue ?

■ 功能描述:

查询垂直光标 B 的测量值。

■ 返回格式:

查询返回垂直光标 B 的测量值。

:CURSor:AYValue**■ 命令格式:**

:CURSor:AYValue ?

■ 功能描述:

查询水平光标 A 的测量值。

■ 返回格式:

查询返回水平光标 A 的测量值。

:CURSor:BYValue

- **命令格式:**
:CURSor:BYValue ?
- **功能描述:**
查询水平光标 B 的测量值。
- **返回格式:**
查询返回水平光标 B 的测量值。

:CURSor:DATA:MODE

- **命令格式:**
:CURSor:DATA:MODE <active>
:CURSor:DATA:MODE ?
- **功能描述:**
设置或查询光标数据悬浮是否打开。
- **参数:**
active:布尔型{ON | OFF}或{1 | 0}
ON:悬浮
OFF:固定
- **返回格式:**
查询返回“1”、“0”分别代表“ON”、“OFF”。
- **举例:**
:CURSor:DATA:MODE ON 设置打开光标数据悬浮
:CURSor:DATA:MODE ? 查询返回 1

时基**:TIMebase:SCALE**

- **命令格式:**
:TIMebase:SCALE <scale>
:TIMebase:SCALE ?
- **功能描述:**

设置或查询主时基档位。

■ **参数:**

scale:实数型, 单位:ps,ns,us,ms,s,ks, 不输入单位则默认单位为 s。设置范围:50ps~1ks。

■ **返回格式:**

查询以科学计数形式返回水平档位值,默认单位为 s。

■ **举例:**

:TIMebase:SCALE 1s

设置水平时基档为 1s

:TIMebase:SCALE ?

查询返回 1.00000000000000E+000

:TIMebase:DElay

■ **命令格式:**

:TIMebase:DElay <delay>

:TIMebase:DElay ?

■ **功能描述:**

设置或查询延迟扫描。

■ **参数:**

delay:实数型, 单位 fs,ps,ns,us,ms,s,ks, 不输入单位则默认单位为 s

■ **返回格式:**

查询以科学计数形式返回延迟扫描。

■ **举例:**

:TIMebase:DElay 1s

设置延迟扫描为 1s

:TIMebase:DElay ?

查询返回 1.00000000000000E+000

触发

:TRIGger:TYPE

■ **命令格式:**

:TRIGger:TYPE <type>

:TRIGger:TYPE ?

■ **功能描述:**

设置或查询触发类型。

■ **参数:**

type:离散型{EDGE | PWIDth | VIDEo | PATTeRn | TIMeout | SETuphold | RUNt | TRANsition | DELay | NEDGE | SUSTaIntime | SERIal}

EDGE:边沿

PWIDth:脉宽

VIDEo:视频

PATTeRn:码型

TIMeout:超时

SETuphold:建立保持

RUNt:欠幅

TRANsition:斜率

DELay:延迟

NEDGE:N 边沿

SUSTaIntime:持续时间

SERIal:总线

■ 返回格式:

查询返回“EDGE”、“PWIDth”、“VIDEo”、“PATTeRn”、“TIMeout”、“SETuphold”、“RUNt”、“TRANsition”、“DELay”、“NEDGE”、“SUSTaIntime”、“SERIal”。

■ 举例:

:TRIGger:TYPe EDGE 设置触发类型为边沿触发

:TRIGger:TYPe ? 查询返回 EDGE

:TRIGger:HOLDoff

■ 命令格式:

:TRIGger:HOLDoff <holdoff>

:TRIGger:HOLDoff ?

■ 功能描述:

设置或查询触发释抑时间。

■ 参数:

holdoff:实数型, 单位:ns,us,ms,s, 默认单位为 s

■ 返回格式:

查询返回释抑时间。

■ 举例:

:TRIGger:HOLDoff 1s 输出选择触发释抑 1s

:TRIGger:HOLDoff ?

查询返回 1.00000000000000E+000

:TRIGger:MODE

■ 命令格式:

:TRIGger:MODE <mode>

:TRIGger:MODE ?

■ 功能描述:

设置或查询触发模式。

■ 参数:

mode:离散型{AUTO | NORMal | SINGle}

AUTO:自动

NORMal:正常

SINGle:单次

■ 返回格式:

查询返回“AUTO”、“NORMal”、“SINGle”。

■ 举例:

:TRIGger:MODE NORMal 设置触发模式为正常模式

:TRIGger:MODE ? 查询返回 NORMal

触发-边沿

:TRIGger:EDGE:COUPling

■ 命令格式:

:TRIGger:EDGE:COUPling <coupling>

:TRIGger:EDGE:COUPling ?

■ 功能描述:

设置或查询触发耦合。

■ 参数:

coupling:离散型{DC | AC | LFReject | HFReject | NREject}

DC:直流

AC:交流

LFReject:低频抑制

HFReject:高频抑制

NREject:噪声抑制

■ 返回格式:

查询返回“DC”、“AC”、“LFReject”、“HFReject”、“NREject”。

■ 举例:

:TRIGger:EDGE:COUPling DC 设置触发耦合为 DC 耦合

:TRIGger:EDGE:COUPling ? 查询返回 DC

:TRIGger:EDGE:SOURce

■ 命令格式:

:TRIGger:EDGE:SOURce <source>

:TRIGger:EDGE:SOURce?

■ 功能描述:

设置或查询边沿触发源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | Ext | Ext5 | AC | AUXin}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”、“D0”、“D1”、“D2”、“D3”、“D4”、“D5”、“D6”、“D7”、“D8”、“D9”、“D10”、“D11”、“D12”、“D13”、“D14”、“D15”、“Ext”、“Ext5”、“AC”、“AUXin”。

■ 举例:

:TRIGger:EDGE:SOURce C1 设置边沿触发源为通道 1

:TRIGger:EDGE:SOURce ? 查询返回 C1

:TRIGger:EDGE:SLOPe

■ 命令格式:

:TRIGger:EDGE:SLOPe <type>

:TRIGger:EDGE:SLOPe ?

■ 功能描述:

设置或查询边沿触发边沿类型。

■ 参数:

type:离散型{RISe | FALL | ANY}

RISe:上升沿

FALL:下降沿

ANY:任意沿

■ **返回格式:**

查询返回“RISe”、“FALL”、“ANY”。

■ **举例:**

:TRIGger:EDGE:SLOPe RISe

设置边沿触发边沿类型为上升沿

:TRIGger:EDGE:SLOPe ?

查询返回 RISe

:TRIGger:EDGE:LEVel

■ **命令格式:**

:TRIGger:EDGE:LEVel <level>

:TRIGger:EDGE:LEVel ?

■ **功能描述:**

设置或查询边沿触发的触发电平。

■ **参数:**

level:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询以科学计数形式返回触发电平。

■ **举例:**

:TRIGger:EDGE:LEVel 1V

设置边沿触发电平为 1V

:TRIGger:EDGE:LEVel ?

查询返回 1.00000E+000

:TRIGger:EDGE:EXTer:n:IMPedance

■ **命令格式:**

:TRIGger:EDGE:EXTer:n:IMPedance <impedance >

:TRIGger:EDGE:EXTer:n:IMPedance ?

■ **功能描述:**

设置或查询外触发的输入阻抗。

■ **参数:**

impedance:离散型{HIGH | LOW}

HIGH:1MΩ

LOW:50Ω

■ **返回格式:**

查询返回外触发的输入阻抗。

■ 举例:

:TRIGger:EDGE:EXtern:IMPedance HIGH

设置外触发输入阻抗为 1MΩ

:TRIGger:EDGE:EXtern:IMPedance ?

查询返回 HIGH

触发-建立保持

:TRIGger:SHOLd:DSOURce

■ 命令格式:

:TRIGger:SHOLd:DSOURce <source>

:TRIGger:SHOLd:DSOURce ?

■ 功能描述:

设置或查询建立保持数据源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”。

■ 举例:

:TRIGger:SHOLd:DSOURce C1

设置建立保持数据源为通道 1

:TRIGger:SHOLd:DSOURce ?

查询返回 C1

说明: 数据源和时钟源不能相同!!!

:TRIGger:SHOLd:CSOURce

■ 命令格式:

:TRIGger:SHOLd:CSOURce <source>

:TRIGger:SHOLd:CSOURce ?

■ 功能描述:

设置或查询建立保持时钟源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”。

■ 举例:

:TRIGger:SHOLd:CSOURce C2

设置建立保持时钟源为通道 2

:TRIGger:SHOLd:CSOUrce ?

查询返回 C2

说明：时钟源和数据源不能相同！！

:TRIGger:SHOLd:TYPE

■ 命令格式:

:TRIGger:SHOLd:TYPE <type>

:TRIGger:SHOLd:TYPE ?

■ 功能描述:

设置或查询建立保持触发的时钟边沿。

■ 参数:

type:离散型{RISe | FALL}

RISe:上升

FALL:下降

■ 返回格式:

查询返回“RISe”、“FALL”。

■ 举例:

:TRIGger:SHOLd:TYPE RISe

设置建立保持触发的时钟边沿为上升沿

:TRIGger:SHOLd:TYPE ?

查询返回 RISe

:TRIGger:SHOLd:CLEVel

■ 命令格式:

:TRIGger:SHOLd:CLEVel <value>

:TRIGger:SHOLd:CLEVel?

■ 功能描述:

设置或查询建立保持触发的时钟电平。

■ 参数:

value:实数，单位:与通道垂直刻度单位一致，默认单位为 V。

■ 返回格式:

查询返回建立保持触发的时钟电平。

■ 举例:

:TRIGger:SHOLd:CLEVel 1V

设置建立保持触发的时钟电平为 1V

:TRIGger:SHOLd:CLEVel ?

查询返回 1.00000E+000

:TRIGger:SHOLd:DLEVel**■ 命令格式:**

:TRIGger:SHOLd:DLEVel <value>

:TRIGger:SHOLd:DLEVel?

■ 功能描述:

设置或查询建立保持触发的数据电平。

■ 参数:

value:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ 返回格式:

查询返回建立保持触发的数据电平。

■ 举例:

:TRIGger:SHOLd:DLEVel 1V

设置建立保持触发的数据电平为 1V

:TRIGger:SHOLd:DLEVel ?

查询返回 1.00000E+000

:TRIGger:SHOLd:PATtern**■ 命令格式:**

:TRIGger:SHOLd:PATtern <type>

:TRIGger:SHOLd:PATtern ?

■ 功能描述:

设置或查询建立保持触发的码型条件。

■ 参数:

type:离散型{HIGH | LOW}

HIGH:高

LOW:低

■ 返回格式:

查询返回“HIGH”、“LOW”。

■ 举例:

:TRIGger:SHOLd:PATtern HIGH

设置建立保持触发的码型条件为 HIGH

:TRIGger:SHOLd:PATtern ?

查询返回 HIGH

:TRIGger:SHOLd:VIOLation**■ 命令格式:**

:TRIGger:SHOLd:VIOLation <type>

:TRIGger:SHOLd:VIOLation ?

■ **功能描述:**

设置或查询保持触发的触发条件。

■ **参数:**

type:离散型{SETup | HOLD | SETHold}

SETup:建立

HOLD:保持

SETHold:建立保持

■ **返回格式:**

查询返回“SETup”、“HOLD”、“SETHold”。

■ **举例:**

:TRIGger:SHOLd:VIOLation SETup

设置保持触发的触发条件为建立

:TRIGger:SHOLd:VIOLation ?

查询返回 SETup

:TRIGger:SHOLd:SETTime

■ **命令格式:**

:TRIGger:SHOLd:SETTime <value>

:TRIGger:SHOLd:SETTime ?

■ **功能描述:**

设置或查询保持触发的建立时间。

■ **参数:**

value:实数, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回保持触发的建立时间。

■ **举例:**

:TRIGger:SHOLd:SETTime 1s

设置保持触发的建立时间为 1s

:TRIGger:SHOLd:SETTime ?

查询返回 1.00000000000000E+000

:TRIGger:SHOLd:HOLDtime

■ **命令格式:**

:TRIGger:SHOLd:HOLDtime <value>

:TRIGger:SHOLd:HOLDtime ?

■ **功能描述:**

设置或查询保持触发的保持时间。

■ **参数:**

value:实数, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回保持触发的保持时间。

■ **举例:**

:TRIGger:SHOLd:HOLDtime 1

设置保持触发的保持时间为 1s

:TRIGger:SHOLd:HOLDtime ?

查询返回 1.00000000000000E+000

触发-码型

:TRIGger:PATtern:LOGic:ALLCondition

■ **命令格式:**

:TRIGger:PATtern:LOGic:ALLCondition <condition>

■ **功能描述:**

设置码型触发逻辑资格的所有通道触发条件。

■ **参数:**

condition:离散型{HIGH | LOW | ANY | RISE | FALL}

HIGH:高

LOW:低

ANY:任意

RISe :上升沿

FALL:下降沿

■ **举例:**

:TRIGger:PATtern:LOGic:ALLCondition HIGH

设置所有通道触发条件为 HIGH

:TRIGger:PATtern:LOGic:CONDition:C<n>

■ **命令格式:**

:TRIGger:PATtern:LOGic:CONDition:C<n> <type>

:TRIGger:PATtern:LOGic:CONDition:C<n> ?

■ **功能描述:**

设置或查询码型触发逻辑资格的单个模拟通道触发条件。

■ 参数:

type:离散型{HIGH | LOW | ANY | RISE | FALL}

HIGH:高

LOW:低

ANY:任意

RISe :上升沿

FALL:下降沿

■ 返回格式:

查询返回“HIGH”、“LOW”、“ANY”、“RISE”、“FALL”。

■ 举例:

:TRIGger:PATtern:LOGic:CONDition:C1 HIGH 设置 C1 触发条件为 HIGH

:TRIGger:PATtern:LOGic:CONDition:C1 ? 查询返回 HIGH

:TRIGger:PATtern:LOGic:GATe:C<n>

■ 命令格式:

:TRIGger:PATtern:LOGic:GATe:C<n> <value>

:TRIGger:PATtern:LOGic:GATe:C<n> ?

■ 功能描述:

设置或查询码型触发逻辑资格的单个模拟通道门限。

■ 参数:

value:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ 返回格式:

查询返回码型触发逻辑资格的单个模拟通道门限。

■ 举例:

:TRIGger:PATtern:LOGic:GATe:C1 1V 设置 C1 触发门限为 1V

:TRIGger:PATtern:LOGic:GATe:C1 ? 查询返回 1.00000E+000

触发-脉宽

:TRIGger:PWIDth:CONDition

■ 命令格式:

:TRIGger:PWIDth:CONDition <condition>

:TRIGger:PWIDth:CONDition ?

■ **功能描述:**

设置或查询脉宽触发条件。

■ **参数:**

condition:离散型{GREaterthan | LESSthan | RANGE}

GREaterthan:大于

LESSthan:小于

RANGE:范围内

■ **返回格式:**

查询返回“GREaterthan”、“LESSthan”、“RANGE”

■ **举例:**

:TRIGger:PWIDth:CONDition GREaterthan 设置脉宽触发条件为 GREaterthan

:TRIGger:PWIDth:CONDition ? 查询返回 GREaterthan

:TRIGger:PWIDth:TiMe:LOWer

■ **命令格式:**

:TRIGger:PWIDth:TiMe:LOWer <time>

:TRIGger:PWIDth:TiMe:LOWer ?

■ **功能描述:**

设置或查询脉宽触发的脉宽下限。

■ **参数:**

time:实数, 单位:ns,us,ms,s, 不输入单位则默认单位为 s

■ **返回格式:**

查询返回以秒为单位。

■ **举例:**

:TRIGger:PWIDth:TiMe:LOWer 1 设置触发脉宽下限为 1s

:TRIGger:PWIDth:TiMe:LOWer ? 查询返回 1.00000000000000E+000

:TRIGger:PWIDth:TiMe:UPPer

■ **命令格式:**

:TRIGger:PWIDth:TiMe:UPPer <time>

:TRIGger:PWIDth:TiMe:UPPer ?

■ **功能描述:**

设置或查询脉宽触发的脉宽上限。

■ **参数:**

time:实数, 单位:ns,us,ms,s, 不输入单位则默认单位为 s

■ **返回格式:**

查询返回以秒为单位。

■ **举例:**

:TRIGger:PWIDth:TIMe:UPPer 1

设置触发脉宽上限为 1s

:TRIGger:PWIDth:TIMe:UPPer ?

查询返回 1.00000000000000E+000

:TRIGger:PWIDth:POLarity

■ **命令格式:**

:TRIGger:PWIDth:POLarity <polarity>

:TRIGger:PWIDth:POLarity ?

■ **功能描述:**

设置或查询脉宽触发的脉宽极性。

■ **参数:**

polarity:离散型{POSitive | NEGative}

POSitive:正脉宽

NEGative:负脉宽

■ **返回格式:**

查询返回“POSitive”、“NEGative”。

■ **举例:**

:TRIGger:PWIDth:POLarity POSitive

设置脉宽触发极性为 POSitive

:TRIGger:PWIDth:POLarity ?

查询返回 POSitive

:TRIGger:PWIDth:SOURce

■ **命令格式:**

:TRIGger:PWIDth:SOURce <source>

:TRIGger:PWIDth:SOURce ?

■ **功能描述:**

设置或查询脉宽触发的源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11

| D12 | D13 | D14 | D15}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”、“D0”、“D1”、“D2”、“D3”、“D4”、“D5”、“D6”、“D7”、“D8”、“D9”、“D10”、“D11”、“D12”、“D13”、“D14”、“D15”。

■ **举例:**

:TRIGger:PWIDth:SOURce C1	设置脉宽触发源为 C1
:TRIGger:PWIDth:SOURce ?	查询返回 C1

:TRIGger:PWIDth:LEVel

■ **命令格式:**

:TRIGger:PWIDth:LEVel <value>
:TRIGger:PWIDth:LEVel ?

■ **功能描述:**

设置或查询脉宽触发的时钟电平。

■ **参数:**

value:实数型, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回脉宽触发的时钟电平。

■ **举例:**

:TRIGger:PWIDth:LEVel 10mV	设置脉宽触发的时钟电平为 10mV
:TRIGger:PWIDth:LEVel ?	查询返回 1.00000E-002

触发-欠幅

:TRIGger:RUNT:SOURce

■ **命令格式:**

:TRIGger:RUNT:SOURce <source>
:TRIGger:RUNT:SOURce ?

■ **功能描述:**

设置或查询欠幅触发源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:TRIGger:RUNT:SOURce C1	设置欠幅触发源为 C1
:TRIGger:RUNT:SOURce ?	查询返回 C1

:TRIGger:RUNT:CONDition

■ **命令格式:**

```
:TRIGger:RUNT:CONDition <type>
:TRIGger:RUNT:CONDition ?
```

■ **功能描述:**

设置或查询欠幅触发条件。

■ **参数:**

type:离散型{GREaterthan | LESSthan | RANGE | NRANge}

GREaterthan :大于

LESSthan:小于

RANGE:范围内

NRANge:范围外

■ **返回格式:**

查询返回“GREaterthan”、“LESSthan”、“RANGE”、“NRANge”。

■ **举例:**

:TRIGger:RUNT:CONDition GREaterthan	设置欠幅触发条件为 GREaterthan
:TRIGger:RUNT:CONDition ?	查询返回 GREaterthan

:TRIGger:RUNT:LEVel:UPPer

■ **命令格式:**

```
:TRIGger:RUNT:LEVel:UPPer <value>
:TRIGger:RUNT:LEVel:UPPer ?
```

■ **功能描述:**

设置或查询欠幅触发的高电平。

■ **参数:**

value:实数型, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回欠幅触发的高电平。

■ 举例:

:TRIGger:RUNT:LEVel:UPPer 1

设置欠幅触发高电平为 1V

:TRIGger:RUNT:LEVel:UPPer ?

查询返回 1.00000E+000

:TRIGger:RUNT:LEVel:LOWer**■ 命令格式:**

:TRIGger:RUNT:LEVel:LOWer <value>

:TRIGger:RUNT:LEVel:LOWer ?

■ 功能描述:

设置或查询欠幅触发的低电平。

■ 参数:

value:实数型, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ 返回格式:

查询返回欠幅触发的低电平。

■ 举例:

:TRIGger:RUNT:LEVel:LOWer 1

设置欠幅触发低电平为 1V

:TRIGger:RUNT:LEVel:LOWer ?

查询返回 1.00000E+000

:TRIGger:RUNT:POLarity**■ 命令格式:**

:TRIGger:RUNT:POLarity <polarity>

:TRIGger:RUNT:POLarity ?

■ 功能描述:

设置或查询欠幅触发的极性。

■ 参数:

Polarity:离散型{POSitive | NEGative}

POSitive:正极性

NEGative:负极性

■ 返回格式:

查询返回“POSitive”、“NEGative”。

■ 举例:

:TRIGger:RUNT:POLarity POSitive

设置欠幅触发极性为正极性

:TRIGger:RUNT:POLarity ?

查询返回 POSitive

:TRIGger:RUNT:TiMe:UPPer■ **命令格式:**

:TRIGger:RUNT:TiMe:UPPer <time>

:TRIGger:RUNT:TiMe:UPPer ?

■ **功能描述:**

设置或查询欠幅触发的欠幅时间上限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回欠幅触发的欠幅时间上限。

■ **举例:**

:TRIGger:RUNT:TiMe:UPPer 1

设置欠幅触发的欠幅时间上限为 1s

:TRIGger:RUNT:TiMe:UPPer ?

查询返回 1.00000000000000E+000

:TRIGger:RUNT:TiMe:LOWer■ **命令格式:**

:TRIGger:RUNT:TiMe:LOWer <time>

:TRIGger:RUNT:TiMe:LOWer ?

■ **功能描述:**

设置或查询欠幅触发的欠幅时间下限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回欠幅触发的欠幅时间下限。

■ **举例:**

:TRIGger:RUNT:TiMe:LOWer 1

设置欠幅触发的欠幅时间下限为 1s

:TRIGger:RUNT:TiMe:LOWer ?

查询返回 1.00000000000000E+000

触发-斜率**:TRIGger:SLOPe:SOURce**■ **命令格式:**

:TRIGger:SLOPe:SOURce <source>

:TRIGger:SLOPe:SOURce ?

■ **功能描述:**

设置或查询斜率触发源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:TRIGger:SLOPe:SOURce C1 设置斜率触发源为通道 1

:TRIGger:SLOPe:SOURce ? 查询返回 C1

:TRIGger:SLOPe:CONDition

■ **命令格式:**

:TRIGger:SLOPe:CONDition <type>

:TRIGger:SLOPe:CONDition ?

■ **功能描述:**

设置或查询斜率触发条件。

■ **参数:**

type:离散型{GREaterthan | LESSthan | RANGe}

GREaterthan:大于

LESSthan:小于

RANGe:范围内

■ **返回格式:**

查询返回“GREaterthan”、“LESSthan”、“RANGe”。

■ **举例:**

:TRIGger:SLOPe:CONDition GREaterthan 设置斜率触发条件为 GREaterthan

:TRIGger:SLOPe:CONDition ? 查询返回 GREaterthan

:TRIGger:SLOPe:LEVel:UPPer

■ **命令格式:**

:TRIGger:SLOPe:LEVel:UPPer <value >

:TRIGger:SLOPe:LEVel:UPPer ?

■ **功能描述:**

设置或查询斜率触发的高电平。

■ **参数:**

value:实数型, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回斜率触发的高电平。

■ **举例:**

:TRIGger:SLOPe:LEVel:UPPer 1

设置斜率触发高电平为 1V

:TRIGger:SLOPe:LEVel:UPPer ?

查询返回 1.00000E+000

:TRIGger:SLOPe:LEVel:LOWer

■ **命令格式:**

:TRIGger:SLOPe:LEVel:LOWer <value>

:TRIGger:SLOPe:LEVel:LOWer ?

■ **功能描述:**

设置或查询斜率触发的低电平。

■ **参数:**

value:实数型, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回斜率触发的低电平。

■ **举例:**

:TRIGger:SLOPe:LEVel:LOWer 1

设置斜率触发低电平为 1V

:TRIGger:SLOPe:LEVel:LOWer ?

查询返回 1.00000E+000

:TRIGger:SLOPe:SLOPe

■ **命令格式:**

:TRIGger:SLOPe:SLOPe <type>

:TRIGger:SLOPe:SLOPe ?

■ **功能描述:**

设置或查询斜率触发的斜率类型。

■ **参数:**

type:离散型{RISe | FALL}

RISe:上升

FALL:下降

■ **返回格式:**

查询返回“RISe”、“FALL”。

■ **举例:**

:TRIGger:SLOPe:SLOPe RISe

设置斜率触发类型为 RISe

:TRIGger:SLOPe:SLOPe ?

查询返回 RISe

:TRIGger:SLOPe:TIME:UPPer

命令格式:

:TRIGger:SLOPe:TIME:UPPer <time>

:TRIGger:SLOPe:TIME:UPPer ?

■ **功能描述:**

设置或查询斜率触发的斜率时间上限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回斜率触发的斜率时间上限。

■ **举例:**

:TRIGger:SLOPe:TIME:UPPer 1

设置斜率触发斜率时间上限为 1s

:TRIGger:SLOPe:TIME:UPPer ?

查询返回 1.00000000000000E+000

:TRIGger:SLOPe:TIME:LOWer

命令格式:

:TRIGger:SLOPe:TIME:LOWer <time>

:TRIGger:SLOPe:TIME:LOWer ?

■ **功能描述:**

设置或查询斜率触发的斜率时间下限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回斜率触发的斜率时间下限。

■ **举例:**

:TRIGger:SLOPe:TIME:LOWer 1

设置斜率触发斜率时间下限为 1s

:TRIGger:SLOPe:TIMe:LOWer ?

查询返回 1.00000000000000E+000

触发-超时

:TRIGger:TIMeout:SOURce

■ 命令格式:

:TRIGger:TIMeout:SOURce <source>

:TRIGger:TIMeout:SOURce ?

■ 功能描述:

设置或查询超时触发源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11
| D12 | D13 | D14 | D15}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”、“D0”、“D1”、“D2”、“D3”、“D4”、“D5”、“D6”、“D7”、“D8”、“D9”、“D10”、“D11”、“D12”、“D13”、“D14”、“D15”。

■ 举例:

:TRIGger:TIMeout:SOURce C1

设置超时触发源为通道 1

:TRIGger:TIMeout:SOURce ?

查询返回 C1

:TRIGger:TIMeout:SLOPe

■ 命令格式:

:TRIGger:TIMeout:SLOPe <type>

:TRIGger:TIMeout:SLOPe ?

■ 功能描述:

设置或查询超时边沿类型。

■ 参数:

type:离散型{POSitive | NEGative | ANY}

POSitive:上升沿

NEGative:下降沿

ANY:任意沿

■ 返回格式:

查询返回“POSitive”、“NEGative”、“ANY”。

■ **举例:**

:TRIGger:TIMEout:SLOPe POSitive	设置超时触发类型为上升沿
:TRIGger:TIMEout:SLOPe ?	查询返回 POSitive

TRIGger:TIMEout:LEVel

■ **命令格式:**

```
:TRIGger:TIMEout:LEVel <level>
:TRIGger:TIMEout:LEVel ?
```

■ **功能描述:**

设置或查询超时触发的触发电平。

■ **参数:**

level:实数型, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回超时触发的触发电平。

■ **举例:**

:TRIGger:TIMEout:LEVel 1	设置超时触发电平为 1V
:TRIGger:TIMEout:LEVel ?	查询返回 1.00000E+000

:TRIGger:TIMEout:TIME

■ **命令格式:**

```
:TRIGger:TIMEout:TIME <time>
:TRIGger:TIMEout:TIME ?
```

■ **功能描述:**

设置或查询超时触发的持续时间。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回超时触发的持续时间。

■ **举例:**

:TRIGger:TIMEout:TIME 1	设置超时触发持续时间为 1s
:TRIGger:TIMEout:TIME ?	查询返回 1.00000000000000E+000

触发-视频

:TRIGger:VIDeo:SOURce

■ **命令格式:**

:TRIGger:VIDeo:SOURce <source>

:TRIGger:VIDeo:SOURce ?

■ **功能描述:**

设置或查询视频触发源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:TRIGger:VIDeo:SOURce C1 设置视频触发源为通道 1

:TRIGger:VIDeo:SOURce ? 查询返回 C1

:TRIGger:VIDeo:STANdard

■ **命令格式:**

:TRIGger:VIDeo:STANdard <standard>

:TRIGger:VIDeo:STANdard ?

■ **功能描述:**

设置或查询视频标准类型。

■ **参数:**

standard:{NTSC | PAL}

■ **返回格式:**

查询返回“NTSC”、“PAL”。

举例:

:TRIGger:VIDeo:STANdard NTSC 设置视频标准类型为 NTSC

:TRIGger:VIDeo:STANdard ? 查询返回 NTSC

:TRIGger:VIDeo:SYNC

■ **命令格式:**

:TRIGger:VIDeo:SYNC <sync>

:TRIGger:VIDeo:SYNC ?

■ **功能描述:**

设置或查询视频触发的同步方式。

■ **参数:**

sync:{ODD | EVEn | ALL | SPECified}

ODD:偶数场

EVEn:奇数场

ALL:所有行

SPECified:指定行

■ **返回格式:**

查询返回视频触发的同步方式。

■ **举例:**

:TRIGger:VIDeo:SYNC ALL

设置视频触发的同步方式为所有行

:TRIGger:VIDeo:SYNC ?

查询返回所有行

:TRIGger:VIDeo:LINE

■ **命令格式:**

:TRIGger:VIDeo:LINE <line>

:TRIGger:VIDeo:LINE ?

■ **功能描述:**

设置或查询视频触发指定行的视频行数。

■ **参数:**

line:整数型, 范围 1~625

■ **返回格式:**

查询返回视频触发指定行的视频行数。

■ **举例:**

:TRIGger:VIDeo:LINE 10

设置视频触发指定行的视频行数为 10

:TRIGger:VIDeo:LINE ?

查询返回 10

:TRIGger:VIDeo:LEVel

■ **命令格式:**

:TRIGger:VIDeo:LEVel <LEVel >

:TRIGger:VIDeo:LEVel ?

■ **功能描述:**

设置或查询视频触发的电平。

■ **参数:**

LEVel :视频触发电平, 单位:与通道垂直刻度单位一致, 默认单位为 V

■ **返回格式:**

查询返回电平值。

■ **举例:**

:TRIGger:VIDeo:LEVel 1

设置视频触发电平为 1V

:TRIGger:VIDeo:LEVel ?

查询返回查询返回 1.00000E+000

触发-N 边沿

:TRIGger:NEDGE:SLOPe

■ **命令格式:**

:TRIGger:NEDGE:SLOPe <type>

:TRIGger:NEDGE:SLOPe ?

■ **功能描述:**

设置或查询 N 边沿触发的边沿类型。

■ **参数:**

type:离散型{RISe | FALL }

RISe:上升沿

FALL:下降沿

■ **返回格式:**

查询返回“RISe”、“FALL”。

■ **举例:**

:TRIGger:NEDGE:SLOPe RISe

设置 N 边沿触发的边沿类型为 RISe

:TRIGger:NEDGE:SLOPe ?

查询返回 RISe

:TRIGger:NEDGE:SOURce

■ **命令格式:**

:TRIGger:NEDGE:SOURce <source>

:TRIGger:NEDGE:SOURce ?

■ 功能描述:

设置或查询 N 边沿触发的触发源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”、“D0”、“D1”、“D2”、“D3”、“D4”、“D5”、“D6”、“D7”、“D8”、“D9”、“D10”、“D11”、“D12”、“D13”、“D14”、“D15”。

■ 举例:

:TRIGger:NEDGE:SOURce C1	设置 N 边沿触发源为通道 1
:TRIGger:NEDGE:SOURce ?	查询返回 C1

:TRIGger:NEDGE:DURATION

■ 命令格式:

:TRIGger:NEDGE:DURATION <time>

:TRIGger:NEDGE:DURATION ?

■ 功能描述:

设置或查询 N 边沿触发的空闲时间。

■ 参数:

time:实数, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。设置范围:3.1ns~10s。

■ 返回格式:

查询返回 N 边沿触发的空闲时间。

■ 举例:

:TRIGger:NEDGE:DURATION 1s	设置 N 边沿触发空闲时间 1s
:TRIGger:NEDGE:DURATION ?	查询返回 1.00000000000000E+000

:TRIGger:NEDGE:LEVEL

■ 命令格式:

:TRIGger:NEDGE:LEVEL <level>

:TRIGger:NEDGE:LEVEL ?

■ 功能描述:

设置或查询 N 边沿触发电平。

■ 参数:

level:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V

■ **返回格式:**

查询返回 N 边沿触发电平值。

■ **举例:**

:TRIGger:NEDGE:LEVel 1V 设置 N 边沿触发电平为 1V

:TRIGger:NEDGE:LEVel ? 查询返回 1.00000E+000

TRIGger:NEDGE:EDGE

■ **命令格式:**

:TRIGger:NEDGE:EDGE <count>

:TRIGger:NEDGE:EDGE ?

■ **功能描述:**

设置或查询 N 边沿触发的边沿数。

■ **参数:**

count:整数型, 设置范围为 1~65.536k

■ **返回格式:**

查询返回边沿数。

■ **举例:**

:TRIGger:NEDGE:EDGE 5 设置 N 边沿触发的边沿数为 5

:TRIGger:NEDGE:EDGE ? 查询返回 5

触发-延迟

:TRIGger:DElay:SOURce<n>

■ **命令格式:**

:TRIGger:DElay:SOURce<n> <source>

:TRIGger:DElay:SOURce<n> ?

■ **功能描述:**

设置或查询延迟触发信源。

■ **参数:**

n:实数{1 | 2}, 表示延迟触发的信源 1 和信源 2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:TRIGger:DElay:SOURce1 C1 设置延迟触发信源 1 的源为 C1

:TRIGger:DElay:SOURce1 ? 查询返回 C1

说明: 信源 1 和信源 2 不可设置为同一个源

:TRIGger:DElay:SLOPe<n>

■ **命令格式:**

:TRIGger:DElay:SLOPe<n> <type>

:TRIGger:DElay:SLOPe<n> ?

■ **功能描述:**

设置或查询延迟触发信源的边沿类型。

■ **参数:**

n:实数{1 | 2}, 表示延迟触发的信源 1 和信源 2

type:离散型{RISe | FALL}

RISe:上升沿

FALL:下降沿

■ **返回格式:**

查询返回“RISe”、“FALL”。

■ **举例:**

:TRIGger:DElay:SLOPe1 FALL 设置延迟触发信源 1 边沿类型为 FALL

:TRIGger:DElay:SLOPe1 ? 查询返回 FALL

:TRIGger:DElay:LEVel<n>

■ **命令格式:**

:TRIGger:DElay:LEVel<n> <level>

:TRIGger:DElay:LEVel<n> ?

■ **功能描述:**

设置或查询延迟触发的信源电平。

■ **参数:**

n:实数{1 | 2}, 表示延迟触发的信源 1 和信源 2

level:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回信源电平值。

■ **举例:**

:TRIGger:DElay:LEVel1 100mV

设置延迟触发信源 1 的电平为 100mV

:TRIGger:DElay:LEVel1 ?

查询返回 1.00000E-001

:TRIGger:DElay:TIME:UPPer

■ **命令格式:**

:TRIGger:DElay:TIME:UPPer <time>

:TRIGger:DElay:TIME:UPPer ?

■ **功能描述:**

设置或查询延迟触发时间上限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回延迟触发时间上限。

■ **举例:**

:TRIGger:DElay:TIME:UPPer 1

设置延迟触发时间上限为 1s

:TRIGger:DElay:TIME:UPPer ?

查询返回 1.0000000000000E+000

:TRIGger:DElay:TIME:LOWer

■ **命令格式:**

:TRIGger:DElay:TIME:LOWer <time>

:TRIGger:DElay:TIME:LOWer ?

■ **功能描述:**

设置或查询延迟触发时间下限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回延迟触发时间下限。

■ **举例:**

:TRIGger:DElay:TIME:LOWer 1

设置延迟触发时间下限为 1s

:TRIGger:DElay:TIME:LOWer ?

查询返回 1.0000000000000E+000

:TRIGger:DElay:CONDition**■ 命令格式:**

:TRIGger:DElay:CONDition <type>

:TRIGger:DElay:CONDition ?

■ 功能描述:

设置或查询延迟触发条件。

■ 参数:

type:离散型{GREaterthan | LESSthan | RANGE | NRANGE}

GREaterthan:大于

LESSthan:小于

RANGE:范围内

NRANGE:范围外

■ 返回格式:

查询返回“GREaterthan”、“LESSthan”、“RANGE”、“NRANGE”。

■ 举例:

:TRIGger:DElay:CONDition GREaterthan

设置延迟触发条件为 GREaterthan

:TRIGger:DElay:CONDition ?

查询返回 GREaterthan

触发-持续时间**:TRIGger:SUSTaintime:CONDition****■ 命令格式:**

:TRIGger:SUSTaintime:CONDition <type>

:TRIGger:SUSTaintime:CONDition ?

■ 功能描述:

设置或查询持续时间触发条件。

■ 参数:

type:离散型{GREaterthan | LESSthan | RANGE}

GREaterthan:大于

LESSthan:小于

RANGE:范围内

■ 返回格式:

查询返回“GREaterthan”、“LESSthan”、“RANGe”。

■ **举例:**

:TRIGger:SUSTaintime:CONDition GREaterthan

设置持续时间触发条件为 GREaterthan

:TRIGger:SUSTaintime:CONDition ?

查询返回 GREaterthan

:TRIGger:SUSTaintime:TIME:UPPer

■ **命令格式:**

:TRIGger:SUSTaintime:TIME:UPPer <time>

:TRIGger:SUSTaintime:TIME:UPPer ?

■ **功能描述:**

设置或查询持续时间触发时间上限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回持续时间触发时间上限。

■ **举例:**

:TRIGger:SUSTaintime:TIME:UPPer 1

设置持续时间触发时间上限为 1s

:TRIGger:SUSTaintime:TIME:UPPer ?

查询返回 1.00000000000000E+000

:TRIGger:SUSTaintime:TIME:LOWer

■ **命令格式:**

:TRIGger:SUSTaintime:TIME:LOWer <time>

:TRIGger:SUSTaintime:TIME:LOWer ?

■ **功能描述:**

设置或查询持续时间触发时间下限。

■ **参数:**

time:实数型, 单位:ns,us,ms,s, 不输入单位则默认单位为 s。

■ **返回格式:**

查询返回持续时间触发时间下限。

■ **举例:**

:TRIGger:SUSTaintime:TIME:LOWer 1

设置持续时间触发时间下限为 1s

:TRIGger:SUSTaintime:TIME:LOWer ?

查询返回 1.00000000000000E+000

:TRIGger:SUST:LOGic:ALLCondition■ **命令格式:**

:TRIGger:SUST:LOGic:ALLCondition <condition>

■ **功能描述:**

设置持续时间触发逻辑资格的所有通道触发条件。

■ **参数:**

condition:离散型{HIGH | LOW | ANY}

HIGH:高

LOW:低

ANY:任意

■ **举例:**

:TRIGger:SUST:LOGic:ALLCondition HIGH 设置所有通道触发条件为 HIGH

:TRIGger:SUST:LOGic:CONDition:C<n>■ **命令格式:**

:TRIGger:SUST:LOGic:CONDition:C<n> <type>

:TRIGger:SUST:LOGic:CONDition:C<n> ?

■ **功能描述:**

设置或查询持续时间触发逻辑资格的单个模拟通道触发条件。

■ **参数:**

type:离散型{HIGH | LOW | ANY}

HIGH:高

LOW:低

ANY:任意

■ **返回格式:**

查询返回“HIGH”、“LOW”、“ANY”。

■ **举例:**

:TRIGger:SUST:LOGic:CONDition:C1 HIGH 设置单个模拟通道触发条件为 HIGH

:TRIGger:SUST:LOGic:CONDition:C1 ? 查询返回 HIGH

:TRIGger:SUST:LOGic:GATe:C<n>■ **命令格式:**

:TRIGger:SUST:LOGic:GATe:C<n> <value>

:TRIGger:SUST:LOGic:GATe:C<n> ?

■ **功能描述:**

设置或查询持续时间触发逻辑资格的单个模拟通道门限。

■ **参数:**

value:实数, 单位:与通道垂直刻度单位一致, 默认单位为 V。

■ **返回格式:**

查询返回持续时间触发逻辑资格的单个模拟通道门限。

■ **举例:**

:TRIGger:SUST:LOGic:GATe:C1 1V 设置单个模拟通道触发门限为 1V

:TRIGger:SUST:LOGic:GATe:C1 ? 查询返回 1.00000E+000

总线

:TRIGger:BUS

■ **命令格式:**

:TRIGger:BUS <type>

:TRIGger:BUS ?

■ **功能描述:**

设置或查询总线触发的总线通道。

■ **参数:**

type:离散型{B1 | B2}

■ **返回格式:**

查询返回总线触发的总线通道。

■ **举例:**

:TRIGger:BUS B1 设置总线触发的触发通道为 B1

:TRIGger:BUS ? 查询返回 B1

RS232

:TRIGger:RS232:CONDition

■ **命令格式:**

:TRIGger:RS232:CONDition <type>

:TRIGger:RS232:CONDition ?

■ **功能描述:**

设置或查询 RS232 的触发条件。

■ **参数:**

type:离散型{START | END | ERRor | DATA}

START:起始位

END:停止位

ERRor:校验错

DATA:数据

■ **返回格式:**

查询返回 RS232 的触发条件。

■ **举例:**

:TRIGger:RS232:CONDition DATA

设置 RS232 触发条件为 DATA

:TRIGger:RS232:CONDition ?

查询返回 DATA

:TRIGger:RS232:DATA

■ **命令格式:**

:TRIGger:RS232:DATA <value>

:TRIGger:RS232:DATA ?

■ **功能描述:**

设置或查询 RS232 触发条件为数据时的数据值。

■ **参数:**

value:参数为 0 或 1 表示的进制字符串数据, 其范围与[:BUS<n>:RS232:WIDTh](#) 指令设置的值相关, 范围为: $0 \sim 2^n - 1$, n 为当前的数据宽度。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:RS232:DATA "01111111"

设置 RS232 触发数据值为 0x7F

:TRIGger:RS232:DATA ?

查询返回 01111111

I2C

:TRIGger:I2C:CONDition

■ 命令格式:

:TRIGger:I2C:CONDition <type>

:TRIGger:I2C:CONDition ?

■ 功能描述:

设置或查询 I2C 的触发条件。

■ 参数:

type:离散型{START | REStart | STOP | LOST | ADDRess | DATA | ADDAta}

START:起始位

REStart:重启

STOP:停止位

LOST:响应失败

ADDRess:地址

DATA:数据

ADDAta:地址和数据

■ 返回格式:

查询返回 I2C 的触发条件。

■ 举例:

:TRIGger:I2C:CONDition DATA

设置 RS232 触发条件为 DATA

:TRIGger:I2C:CONDition ?

查询返回 DATA

:TRIGger:I2C:DATA

■ 命令格式:

:TRIGger:I2C:DATA <value>

:TRIGger:I2C:DATA ?

■ 功能描述:

设置或查询 I2C 触发条件为数据和地址数据时的数据值。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:I2C:DATA "10001110"

设置 I2C 触发数据值为 0x8E

:TRIGger:I2C:DATA ?

查询返回 10001110

:TRIGger:I2C:DIRection

■ **命令格式:**

:TRIGger:I2C:DIRection <direction>

:TRIGger:I2C:DIRection ?

■ **功能描述:**

设置或查询 I2C 触发条件为地址和地址数据时的数据方向。

■ **参数:**

direction:离散型{WRITE | READ}

WRITE:写

READ:读

■ **返回格式:**

查询返回 I2C 触发条件为地址和地址数据时的数据方向。

■ **举例:**

:TRIGger:I2C:DIRection WRITE

设置 I2C 触发数据方向为 WRITE

:TRIGger:I2C:DIRection ?

查询返回 WRITE

:TRIGger:I2C:ADDRess

■ **命令格式:**

:TRIGger:I2C:ADDRess <value>

:TRIGger:I2C:ADDRess ?

■ **功能描述:**

设置或查询 I2C 触发条件为地址或地址数据时的地址值。

■ **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:I2C:ADDRess "100101"

设置 I2C 触发地址值为 0x25

:TRIGger:I2C:ADDRess ?

查询返回 100101

SPI

:TRIGger:SPI:CONDition

- **命令格式:**

:TRIGger:SPI:CONDition <condition>

:TRIGger:SPI:CONDition ?

- **功能描述:**

设置或查询 SPI 触发条件。

- **参数:**

condition:离散型{CS | DATA}

CS:片选有效

DATA:数据

- **返回格式:**

查询返回 SPI 触发条件。

- **举例:**

:TRIGger:SPI:CONDition DATA

设置 SPI 触发条件为 DATA

:TRIGger:SPI:CONDition ?

查询返回 DATA

:TRIGger:SPI:DATA

- **命令格式:**

:TRIGger:SPI:DATA <value>

:TRIGger:SPI:DATA ?

- **功能描述:**

设置或查询 SPI 触发条件为数据时的数据值。

- **参数:**

value:参数为 0 或 1 表示的二进制字符串数据，其范围与[:TRIGger:SPI:FRAMecount](#) 指令设置的帧数相关。

- **返回格式:**

查询返回二进制字符串数据。

- **举例:**

:TRIGger:SPI:DATA "10001"

设置 SPI 触发数据值为 0x001

:TRIGger:SPl:DATA ?

查询返回 10001

:TRIGger:SPl:DACHannel

■ 命令格式:

:TRIGger:SPl:DACHannel <source>

:TRIGger:SPl:DACHannel ?

■ 功能描述:

设置或查询 SPI 触发条件为数据时的数据通道。

■ 参数:

source:离散型{MISO | MOSI | MOMI}

■ 返回格式:

查询返回 SPI 触发条件为数据时的数据通道。

■ 举例:

:TRIGger:SPl:DACHannel MOMI

设置 SPI 触发数据通道为 MOMI

:TRIGger:SPl:DACHannel ?

查询返回 MOMI

:TRIGger:SPl:FRAMecount

■ 命令格式:

:TRIGger:SPl:FRAMecount <value>

:TRIGger:SPl:FRAMecount ?

■ 功能描述:

设置或查询 SPI 触发条件为数据时的帧数。

■ 参数:

value:整数型

■ 返回格式:

查询返回 SPI 触发条件为数据时的帧数。

■ 举例:

:TRIGger:SPl:FRAMecount 2

设置 SPI 触发帧数为 2

:TRIGger:SPl:FRAMecount ?

查询返回 2

:TRIGger:CAN:CONDition

■ 命令格式:

:TRIGger:CAN:CONDition <condition>

:TRIGger:CAN:CONDition ?

■ 功能描述:

设置或查询 CAN 触发方式。

■ 参数:

condition:离散型{STARTframe | TYPEframe | ID | DATA | IDData | ENDFrame | ERROR}

STARTframe:帧起始

TYPEframe:帧类型

ID:ID

DATA:数据

IDData:ID 和数据

ENDFrame:帧结尾

ERROR:错误

■ 返回格式:

查询返回 CAN 触发方式。

■ 举例:

:TRIGger:CAN:CONDition DATA

设置 CAN 触发方式为 DATA

:TRIGger:CAN:CONDition ?

查询返回 DATA

:TRIGger:CAN:DATA

■ 命令格式:

:TRIGger:CAN:DATA <value>

:TRIGger:CAN:DATA ?

■ 功能描述:

设置或查询 CAN 触发条件为数据或 ID 和数据时的数据值。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:CAN:DATA "1010100"

设置 CAN 触发数据值为 54h

:TRIGger:CAN:DATA ?

查询返回 1010100

:TRIGger:CAN:DIRection

■ **命令格式:**

:TRIGger:CAN:DIRection <direction>

:TRIGger:CAN:DIRection ?

■ **功能描述:**

设置或查询 CAN 触发条件为 ID 或 ID 和数据时的帧方向。

■ **参数:**

direction:离散型{READ | WRITe | BOTH}

READ:读

WRITe:写

BOTH:读写

■ **返回格式:**

查询返回 CAN 触发条件为 ID 或 ID 和数据时的帧方向。

■ **举例:**

:TRIGger:CAN:DIRection WRITe

设置 CAN 触发帧方向为 WRITe

:TRIGger:CAN:DIRection ?

查询返回 WRITe

:TRIGger:CAN:IDSTandard

■ **命令格式:**

:TRIGger:CAN:IDSTandard <standard>

:TRIGger:CAN:IDSTandard ?

■ **功能描述:**

设置或查询 CAN 触发条件为 ID 或 ID 和数据时的 ID 标准。

■ **参数:**

standard:离散型{STANdard | EXTended}

STANdard:标准

EXTended:扩展

■ **返回格式:**

查询返回 CAN 触发条件为 ID 或 ID 和数据时的 ID 标准。

■ **举例:**

:TRIGger:CAN:IDSTandard EXTended

设置 CAN 触发 ID 标准为 EXTended

:TRIGger:CAN:IDSTandard ?

查询返回 EXTended

:TRIGger:CAN:STANdardid

■ **命令格式:**

:TRIGger:CAN:STANdardid <value>

:TRIGger:CAN:STANdardid ?

■ **功能描述:**

设置或查询 CAN 触发条件为 ID 或 ID 和数据时的标准 ID。

■ **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:CAN:STANdardid "101000"

设置 CAN 触发标准 ID 为 028h

:TRIGger:CAN:STANdardid ?

查询返回 101000

:TRIGger:CAN:FTYPE

■ **命令格式:**

:TRIGger:CAN:FTYPE <type>

:TRIGger:CAN:FTYPE ?

■ **功能描述:**

设置或查询 CAN 触发条件为帧类型时的帧类型。

■ **参数:**

type:离散型{DATA | REMote | ERRor | OVERload}

DATA:数据帧

REMote:远程帧

ERRor:错误帧

OVERload:超载帧

■ **返回格式:**

查询返回 CAN 触发条件为帧类型时的帧类型。

■ **举例:**

:TRIGger:CAN:FTYPE DATA	设置 CAN 触发帧类型为 DATA
:TRIGger:CAN:FTYPE ?	查询返回 DATA

:TRIGger:CAN:ETYPe

■ **命令格式:**

```
:TRIGger:CAN:ETYPe <type>
:TRIGger:CAN:ETYPe ?
```

■ **功能描述:**

设置或查询 CAN 触发条件为错误时的错误包类型。

■ **参数:**

type:离散型{ACKLose | BITFillerror}

ACKLose:响应失败

BITFillerror:位填充错误

■ **返回格式:**

查询返回 CAN 触发条件为错误时的错误包类型。

■ **举例:**

:TRIGger:CAN:ETYPe ACKLose	设置 CAN 触发错误包类型为 ACKLose
:TRIGger:CAN:ETYPe ?	查询返回 ACKLose

CAN-FD

:TRIGger:CAN_FD:CONDition

■ **命令格式:**

```
:TRIGger:CAN_FD:CONDition <condition>
:TRIGger:CAN_FD:CONDition ?
```

■ **功能描述:**

设置或查询 CAN_FD 触发方式。

■ **参数:**

condition:离散型{STARTframe | TYPEframe | ID | DATA | IDData | ENDFrame | ERROR | BITFILLERROR | CRCERROR | ALLERROR}

STARTframe:帧起始

TYPEframe:帧类型

ID:ID

DATA:数据

IDData:ID 和数据

ENDFrame:帧结尾

ERRor:丢失确认

BITFILLERRor:位填充错误

CRCERRor:CRC 错误

ALLERRor:所有错误

■ 返回格式:

查询返回 CAN_FD 触发方式。

■ 举例:

:TRIGger:CAN_FD:CONDition IDData

设置 CAN_FD 触发方式为 IDData

:TRIGger:CAN_FD:CONDition ?

查询返回 IDData

:TRIGger:CAN_FD:DATA

■ 命令格式:

:TRIGger:CAN_FD:DATA <value>

:TRIGger:CAN_FD:DATA ?

■ 功能描述:

设置或查询 CAN_FD 触发条件为数据或 ID 和数据时的数据值。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ 举例:

:TRIGger:CAN_FD:DATA "1010100"

设置 CAN_FD 数据值为 54h

:TRIGger:CAN_FD:DATA ?

查询返回 1010100

:TRIGger:CAN_FD:DIRection

■ 命令格式:

:TRIGger:CAN_FD:DIRection <direction>

:TRIGger:CAN_FD:DIRection ?

■ 功能描述:

设置或查询 CAN 触发条件为 ID 或 ID 和数据时的帧方向。

■ **参数:**

direction:离散型{READ | WRITe | BOTH}

READ:读

WRITe:写

BOTH:读写

■ **返回格式:**

查询返回 CAN_FD 触发条件为 ID 或 ID 和数据时的帧方向。

■ **举例:**

:TRIGger:CAN_FD:DIRection WRITe

设置 CAN_FD 触发帧方向为 WRITe

:TRIGger:CAN_FD:DIRection ?

查询返回 WRITe

:TRIGger:CAN_FD:IDSTandard

■ **命令格式:**

:TRIGger:CAN_FD:IDSTandard <standard>

:TRIGger:CAN_FD:IDSTandard ?

■ **功能描述:**

设置或查询 CAN_FD 触发条件为 ID 或 ID 和数据时的 ID 标准。

■ **参数:**

standard:离散型{STANdard | EXTended}

STANdard:标准

EXTended:扩展

■ **返回格式:**

查询返回 CAN_FD 触发条件为 ID 或 ID 和数据时的 ID 标准。

■ **举例:**

:TRIGger:CAN_FD:IDSTandard EXTended

设置 CAN_FD 触发 ID 标准为 EXTended

:TRIGger:CAN_FD:IDSTandard ?

查询返回 EXTended

:TRIGger:CAN_FD:STANdardid

■ **命令格式:**

:TRIGger:CAN_FD:STANdardid <value>

:TRIGger:CAN_FD:STANdardid ?

■ **功能描述:**

设置或查询 CAN_FD 触发条件为 ID 或 ID 和数据时的标准 ID。

■ **参数:**

value: 参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:CAN_FD:STANdardid "101000"	设置 CAN_FD 触发标准 ID 为 0x28
:TRIGger:CAN_FD:STANdardid ?	查询返回 101000

:TRIGger:CAN_FD:FTYPE

■ **命令格式:**

:TRIGger:CAN_FD:FTYPE <type>

:TRIGger:CAN_FD:FTYPE ?

■ **功能描述:**

设置或查询 CAN_FD 触发条件为帧类型时的帧类型。

■ **参数:**

type: 离散型{DATA | REMote | ERRor | OVERload}

DATA: 数据帧

REMote: 远程帧

ERRor: 错误帧

OVERload: 超载帧

■ **返回格式:**

查询返回 CAN_FD 触发条件为帧类型时的帧类型。

■ **举例:**

:TRIGger:CAN_FD:FTYPE DATA	设置 CAN_FD 触发帧类型为 DATA
:TRIGger:CAN_FD:FTYPE ?	查询返回 DATA

:TRIGger:CAN_FD:ETYPE

■ **命令格式:**

:TRIGger:CAN_FD:ETYPE <type>

:TRIGger:CAN_FD:ETYPE ?

■ **功能描述:**

设置或查询 CAN_FD 触发条件为丢失确认时的错误包类型。

■ 参数:

type:离散型{ACKLose | BITFillerror}

ACKLose:响应失败

BITFillerror:位填充错误

■ 返回格式:

查询返回 CAN_FD 触发条件为丢失确认时的错误包类型。

■ 举例:

:TRIGger:CAN_FD:ETYPe ACKLose

设置 CAN_FD 触发错误包类型为 ACKLose

:TRIGger:CAN_FD:ETYPe ?

查询返回 ACKLose

LIN

:TRIGger:LIN:CONDition

■ 命令格式:

:TRIGger:LIN:CONDition <condition>

:TRIGger:LIN:CONDition ?

■ 功能描述:

设置或查询 LIN 触发方式。

■ 参数:

condition:离散型{STARTframe | ID | DATA | IDData | WAKEup | SLEEp | SYERror | IDCRcerror
| SUMCrcerror}

STARTframe:帧起始

ID:ID

DATA:数据

IDData:ID 和数据

WAKEup:唤醒帧

SLEEp:休眠帧

SYERror:同步错误

IDCRcerror:ID 校验错

SUMCrcerror:校验和错

■ 返回格式:

查询返回 LIN 触发方式。

■ 举例:

:TRIGger:LIN:CONDition IDData

设置 LIN 触发方式为 IDData

:TRIGger:LIN:CONDition ?

查询返回 IDData

:TRIGger:LIN:DATA

■ 命令格式:

:TRIGger:LIN:DATA <value>

:TRIGger:LIN:DATA ?

■ 功能描述:

设置或查询 LIN 触发条件为数据或 ID 和数据时的数据值。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ 举例:

:TRIGger:LIN:DATA "1010100"

设置 LIN 数据值为 54h

:TRIGger:LIN:DATA ?

查询返回 1010100

:TRIGger:LIN:ID

■ 命令格式:

:TRIGger:LIN:ID <value>

:TRIGger:LIN:ID ?

■ 功能描述:

设置或查询 LIN 触发条件为 ID 或 ID 和数据时的标准 ID。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ 举例:

:TRIGger:LIN:ID "101000"

设置 LIN 标准 ID 为 0x28

:TRIGger:LIN:ID ?

查询返回 101000

:TRIGger:FR:CONDition

■ 命令格式:

:TRIGger:FR:CONDition <condition>

:TRIGger:FR:CONDition ?

■ 功能描述:

设置或查询 FLEXRAY 触发方式。

■ 参数:

condition:离散型{HEADframe | INDicator | ID | CIRCulate | DATA | IDData | TAILframe | ERRor}

HEADframe:帧头

INDicator:指示位

ID:ID

CIRCulate:周期数

DATA:数据

IDData:ID 和数据

TAILframe:帧结尾

ERRor:错误

■ 返回格式:

查询返回 FLEXRAY 触发方式。

■ 举例:

:TRIGger:FR:CONDition DATA

设置 TRIGger 的 FLEXRAY 触发方式为 DATA

:TRIGger:FR:CONDition ?

查询返回 DATA

:TRIGger:FR:DATA

■ 命令格式:

:TRIGger:FR:DATA <value>

:TRIGger:FR:DATA ?

■ 功能描述:

设置或查询 FLEXRAY 触发条件为数据或 ID 和数据时的数据值。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据吗, 其范围与:TRIGger:FR:DCOUnt 指令设置的值

相关。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:FR:DATA "1010100"

设置 FLEXRAY 触发数据值为 54h

:TRIGger:FR:DATA ?

查询返回 1010100

:TRIGger:FR:DCOUnt

■ **命令格式:**

:TRIGger:FR:DCOUnt <value>

:TRIGger:FR:DCOUnt ?

■ **功能描述:**

设置或查询 FLEXRAY 触发条件为数据或 ID 和数据时的数据长度。

■ **参数:**

value:整数型

■ **返回格式:**

查询返回 FLEXRAY 触发条件为数据或 ID 和数据时的数据长度。

■ **举例:**

:TRIGger:FR:DCOUnt 3

设置 FLEXRAY 触发数据长度为 3

:TRIGger:FR:DCOUnt ?

查询返回 3

:TRIGger:FR:ID

■ **命令格式:**

:TRIGger:FR:ID <value>

:TRIGger:FR:ID ?

■ **功能描述:**

设置或查询 FLEXRAY 触发条件为 ID 或 ID 和数据时的标准 ID。

■ **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:FR:ID "101000"

设置 FLEXRAY 触发数据值为 28h

:TRIGger:FR:ID ?

查询返回 101000

:TRIGger:FR:INDicator

■ 命令格式:

:TRIGger:FR:INDicator <type>

:TRIGger:FR:INDicator ?

■ 功能描述:

设置或查询 FLEXRAY 触发条件为指示位时指示位类型。

■ 参数:

type:离散型{NMFRame | PLFRame | EPFRame | SNFRame | STFRame}

NMFRame:正常帧

PLFRame:净荷帧

EPFRame:空帧

SNFRame:同步帧

STFRame:启动帧

■ 返回格式:

查询返回 FLEXRAY 触发条件为指示位时指示位类型。

■ 举例:

:TRIGger:FR:INDicator NMFRame

设置 FLEXRAY 触发数据值为 NMFRame

:TRIGger:FR:INDicator ?

查询返回 NMFRame

:TRIGger:FR:CYCLedata

■ 命令格式:

:TRIGger:FR:CYCLedata <value>

:TRIGger:FR:CYCLedata ?

■ 功能描述:

设置或查询 FLEXRAY 触发条件为周期数时的循环数据。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ 举例:

:TRIGger:FR:CYCLedata "100000"

设置 FLEXRAY 触发循环数据值为 0x20

:TRIGger:FR:CYCLedata ?

查询返回 100000

:TRIGger:FR:EDFType

■ 命令格式:

:TRIGger:FR:EDFType <type>

:TRIGger:FR:EDFType ?

■ 功能描述:

设置或查询 FLEXRAY 触发条件为结尾帧时的结尾帧类型。

■ 参数:

type:离散型{STATic | DYNAmic | ALL}

STATic:静态

DYNAmic:动态

ALL:全部

■ 返回格式:

查询返回 FLEXRAY 触发条件为结尾帧时的结尾帧类型。

■ 举例:

:TRIGger:FR:EDFType DYNAmic

设置 FLEXRAY 触发结尾帧类型为 DYNAmic

:TRIGger:FR:EDFType ?

查询返回 DYNAmic

:TRIGger:FR:ETYPe

■ 命令格式:

:TRIGger:FR:ETYPe <type>

:TRIGger:FR:ETYPe ?

■ 功能描述:

设置或查询 FLEXRAY 触发条件为错误时的错误包类型。

■ 参数:

type:离散型{PGHeadcrc | PGTail | SEFRame | DEFRame | SYFRame | STFRame}

PGHeadcrc:标头 CRC

PGTail:帧尾

SEFRame:空帧, 静态

DEFRame:空帧, 动态

SYFRame:同步帧

STFRame:启动帧 (无同步)

- **返回格式:**

查询返回 FLEXRAY 触发条件为错误时的错误包类型。

- **举例:**

:TRIGger:FR:ETYPe SEFRame

设置 FLEXRAY 触发错误包类型为 SEFRame

:TRIGger:FR:ETYPe ?

查询返回 SEFRame

AUDIOBUS

:TRIGger:AB:CONDition

- **命令格式:**

:TRIGger:AB:CONDition <type>

:TRIGger:AB:CONDition ?

- **功能描述:**

设置或查询 AUDIOBUS 触发方式。

- **参数:**

type:离散型{SYNC | DATA}

SYNC:同步

DATA:数据

- **返回格式:**

查询返回 AUDIOBUS 触发方式。

- **举例:**

:TRIGger:AB:CONDition DATA

设置 AUDIOBUS 触发方式为 DATA

:TRIGger:AB:CONDition ?

查询返回 DATA

:TRIGger:AB:DATA

- **命令格式:**

:TRIGger:AB:DATA <value>

:TRIGger:AB:DATA ?

- **功能描述:**

设置或查询 AUDIOBUS 触发条件为数据时的数据值。

- **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

- **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:AB:DATA "1010100"

设置 AUDIOBUS 触发数据值为 0x00000054

:TRIGger:AB:DATA ?

查询返回 1010100

MIL_STD_1553

:TRIGger:MS:CONDition

■ **命令格式:**

:TRIGger:MS:CONDition <condition>

:TRIGger:MS:CONDition ?

■ **功能描述:**

设置或查询 MIL_STD_1553 触发条件。

■ **参数:**

condition:离散型{CMD | STATus | DATA | CRCError}

CMD:命令帧

STATus:状态帧

DATA:数据帧

CRCError:CRC 校验错

■ **返回格式:**

查询返回 MIL_STD_1553 触发条件。

■ **举例:**

:TRIGger:MS:CONDition STATus

设置 MIL_STD_1553 触发条件为 STATus

:TRIGger:MS:CONDition ?

查询返回 STATus

ARINC429

:TRIGger:A429:CONDition

■ **命令格式:**

:TRIGger:A429:CONDition <condition>

:TRIGger:A429:CONDition ?

■ **功能描述:**

设置或查询 ARINC429 触发方式。

■ 参数:

condition:离散型
 HDFrame:帧起始
 LABel:标签
 SDI:源或目的标识符
 DATA:数据
 SSM:标志和状态位
 CKError:校验错

■ 返回格式:

查询返回 ARINC429 触发方式。

■ 举例:

:TRIGger:A429:CONDition DATA	设置 ARINC429 触发方式为 DATA
:TRIGger:A429:CONDition ?	查询返回 DATA

:TRIGger:A429:DATA

■ 命令格式:

:TRIGger:A429:DATA <value>
 :TRIGger:A429:DATA ?

■ 功能描述:

设置或查询 ARINC429 触发条件为数据时的数据值。

■ 参数:

value:参数为 0 或 1 表示的二进制字符串数据。

■ 返回格式:

查询返回二进制字符串数据。

■ 举例:

:TRIGger:A429:DATA "1010100"	设置 ARINC429 触发数据值为 0x54
:TRIGger:A429:DATA ?	查询返回 1010100

:TRIGger:A429:SDI

■ 命令格式:

:TRIGger:A429:SDI <value>
 :TRIGger:A429:SDI ?

■ 功能描述:

设置或查询 ARINC429 触发条件为源或目的标识符时的值。

■ **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:A429:SDI "100011"	设置 ARINC429 触发 SDI 值为 0x23
:TRIGger:A429:SDI ?	查询返回 100011

:TRIGger:A429:SSM

■ **命令格式:**

:TRIGger:A429:SSM <value>

:TRIGger:A429:SSM ?

■ **功能描述:**

设置或查询 ARINC429 触发条件为标志和状态位的值。

■ **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:A429:SSM "100011"	设置 ARINC429 触发 SSM 值为 0x23
:TRIGger:A429:SSM ?	查询返回 100011

:TRIGger:A429:LABel

■ **命令格式:**

:TRIGger:A429:LABel <value>

:TRIGger:A429:LABel ?

■ **功能描述:**

设置或查询 ARINC429 触发条件为标签时的标签值。

■ **参数:**

value:无符号整数型

■ **返回格式:**

查询返回 ARINC429 触发条件为标签时的标签值。

■ **举例:**

:TRIGger:A429:LABel 22	设置 ARINC429 触发标签值为 22
:TRIGger:A429:LABel ?	查询返回 22

SENT

:TRIGger:SENT:CONDition

■ **命令格式:**

```
:TRIGger:SENT:CONDition <condition>
:TRIGger:SENT:CONDition ?
```

■ **功能描述:**

设置或查询 SENT 触发条件。

■ **参数:**

condition:离散型{SYNC | START | DATA | CRCErrors}

SYNC:同步位

START:帧起始

DATA:数据

CRCErrors:CRC 校验错

■ **返回格式:**

查询返回 SENT 触发条件。

■ **举例:**

:TRIGger:SENT:CONDition DATA	设置 SENT 触发条件为 DATA
:TRIGger:SENT:CONDition ?	查询返回 DATA

:TRIGger:SENT:DATA

■ **命令格式:**

```
:TRIGger:SENT:DATA <value>
:TRIGger:SENT:DATA ?
```

■ **功能描述:**

设置或查询 SENT 触发条件为数据时的数据值。

■ **参数:**

value:参数为 0 或 1 表示的二进制字符串数据。

■ **返回格式:**

查询返回二进制字符串数据。

■ **举例:**

:TRIGger:SENT:DATA "1010100"

设置 SENT 触发条件为 0x54

:TRIGger:SENT:DATA ?

查询返回 1010100

波形数据

:WAVeform:SOURce

■ **命令格式:**

:WAVeform:SOURce <source>

:WAVeform:SOURce ?

■ **功能描述:**

设置或查询波形数据读取的通道源。

当通道源为 math 通道时，只有 ASCII 格式数据有效。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”、“M1”、“M2”、“M3”、“M4”、“M5”、“M6”、“M7”、“M8”。

■ **举例:**

:WAVeform:SOURce C1

设置波形数据读取的通道源为通道 1

:WAVeform:SOURce ?

查询返回 C1

:WAVeform:MODE

■ **命令格式:**

:WAVeform:MODE <mode>

:WAVeform:MODE ?

■ **功能描述:**

设置或查询读取波形数据的类型。

■ **参数:**

mode:离散型{NORMal | RAW}

NORMal:屏幕显示数据

RAW:内存波形数据

■ **返回格式:**

查询返回“NORMal”、“RAW”。

■ **举例:**

:WAVEform:MODE NORMal

设置读取波形数据的类型为 NORMal

:WAVEform:MODE ?

查询返回 NORMal

:WAVEform:FORMat

■ **命令格式:**

:WAVEform:FORMat <format>

:WAVEform:FORMat ?

■ **功能描述:**

设置或查询波形数据的返回格式。

■ **参数:**

format:{BYTE | ASCII | DWORD}

BYTE: 单字节数据, 如果单个波形点 AD 值占一个字节, 只有 8 位 ADC 时有效, 仅支持“C1”、“C2”、“C3”、“C4”, 返回模拟通道波形数据。

DWORD: 双字节数据, 如果单个波形点 AD 值占两个字节, 8 位 ADC 时, 低 8 位有效, 高 8 位为 0。

ASCII: 以科学计数形式返回各波形点的实际电压值, 各电压值之间以逗号隔开且符合[数据块格式](#)。当通道源为 math 通道时, 只有 ASCII 格式数据有效, 返回屏幕显示的数学通道波形电压数据。

■ **返回格式:**

查询返回“BYTE”、“DWORD”、“ASCII”。

■ **举例:**

:WAVEform:FORMat BYTE

设置波形数据的返回格式为 BYTE。

:WAVEform:FORMat ?

查询返回 BYTE

:WAVEform:START

■ **命令格式:**

:WAVEform:START <value>

:WAVEform:START ?

■ **功能描述:**

设置或查询欲读取的波形起始位置。

■ **参数:**

value:整数型。

■ **返回格式:**

查询返回欲读取的波形起始位置。

■ **举例:**

:WAVeform:STARt 0	设置读取波形起始位置为 0（即第一个点）
:WAVeform:STARt ?	查询返回 0

:WAVeform:POINts

■ **命令格式:**

:WAVeform:POINts <point>
:WAVeform:POINts ?

■ **功能描述:**

设置或查询欲读取的波形点数。

■ **参数:**

point:整数型 。

■ **返回格式:**

查询返回欲读取的波形点数。

■ **举例:**

:WAVeform:POINts 10000	设置欲读取的波形点数为 10000。
:WAVeform:POINts ?	查询返回 10000

:WAVeform:DATA

■ **命令格式:**

:WAVeform:DATA ?

■ **功能描述:**

读取当前屏幕显示波形数据；

■ **返回格式:**

查询返回波形数据。返回格式与当前选择的波形数据返回格式:WAVeform:FORMat 有关

■ **举例:**

获得指定数据源的波形数据指令顺序执行如下:

获得屏幕波形数据流程

:WAVeform:SOURce C1	设置当前要查询波形数据的信号源为通道一
:WAVeform: MODE NORMal	设置读取屏幕显示波形数据

:WAVEform:FORMat DWORD 波形数据的返回格式为双字模式

:WAVEform:DATA ? 获得波形数据

获得内存波形数据流程，当存储深度过大时，需分块读取，此流程只有在示波器停止状态下有效

:WAVEform:SOURce C1 设置当前要查询波形数据的信号源为通道一

:WAVEform: MODE RAW 设置读取内存波形数据

:WAVEform:FORMat DWORD 波形数据的返回格式为双字模式

:WAVEform:POINts 5000 读取内存被形点数为 5000

循环发送读取波形指令以获取整个内存数据

:WAVEform:DATA ? 获得内存中一块的波形数据

:WAVEform: START ? 直到波形数据读取的起始位置等于-1 表示已到最后一个点

测量

测量参数表

类型	参数	描述	多信源
垂直	最大值 (MAX)	采样点的最大值，指波形显示的最高值	
垂直	最小值 (MIN)	采样点的最小值，指波形显示的最低值	
垂直	峰峰值 (PKPK)	最大值与最小值之间的差值	
垂直	顶值 (TOP)	波形中点以上出现频率最高的值	
垂直	底值 (BASE)	波形中点以下出现频率最高的值	
垂直	中值 (MID)	顶端值与底端值之间的平均值	
垂直	幅度 (AMPL)	顶端值与底端值之间的差值	
垂直	平均值 (AVE)	波形区域内所有采样点数值的算术平均值	
垂直	有效值 (RMS)	波形区域内所有波形数据电压值的均方根值	
垂直	标准差 (STDD)	波形区域内所有波形数据移除直流分量后的电压值均	

		方根值	
垂直	正过冲 (POST)	上升沿跃迁后的失真	
垂直	负过冲 (NOST)	下降沿跃迁后的失真	
垂直	周期最大值 (CMAX)	波形在一个周期内的最大值	
垂直	周期最小值 (CMIN)	波形在一个周期内的最小值	
垂直	周期有效值 (CRMS)	波形在一个周期内的均方根值	
垂直	周期平均值 (CAVE)	波形在一个周期内的平均值	
垂直	周期峰峰值 (CPK)	波形在一个周期内的峰峰值	
垂直	周期中值 (CMID)	波形在一个周期内中间值	
垂直	正预冲 (PPST)	波形的正预冲是指上升沿跃迁前的失真	
垂直	负预冲 (NPST)	波形的负预冲是指下降沿跃迁前的失真	
水平	周期 (PRD)	周期是完成一个重复周期所需的时间	
水平	频率 (FREQ)	频率定义为周期的倒数	
水平	上升时间 (RISE)	边沿从最低参考电平 (RB) 升到最高参考电平 (RT) 所需的时间	
水平	下降时间 (FALL)	边沿从最高参考电平 (RT) 降到最低参考电平 (RB) 所需的时间	
水平	正脉宽 (PWID)	上升沿的中间阈值到下一个下降沿的中间阈值的时间	
水平	负脉宽 (NWID)	下降沿的中间阈值到下一个上升沿的中间阈值的时间	
水平	正占空比 (DUTY)	波形的正脉冲宽度和周期的比率	
水平	负占空比 (NDTY)	波形的负脉冲宽度和周期的比率	

水平	时间@Max (TMAX)	从波形区域的左侧开始第一次出现波形最大值时的水平轴值	
水平	时间@Min (TMIN)	从波形区域的左侧开始第一次出现波形最小值时的水平轴值	
水平	上升时间@Lv (LRIS)	在指定电平处波形的上升时间	
水平	下降时间@Lv (LFAL)	在指定电平处波形的下降时间	
水平	周期@Lv (LPRD)	在指定电平处波形的周期时间	
水平	频率@Lv (LFRQ)	在指定电平处波形的频率	
水平	脉宽@Lv (LWID)	在指定电平处波形的脉冲宽度	
水平	占空比@Lv (LDUT)	在指定电平处波形的占空比	
水平	相位差@Lv (LPHA)	在指定电平处两个源第一个上升沿 50%处的相位差	✓
水平	RRD@Lv (LRRD)	计算两个源第一个上升沿指定电平处的时间差	✓
水平	FFD@Lv (LFFD)	计算两个源第一个下降沿指定电平处的时间差	✓
水平	RFD@Lv (LRFD)	计算从第一个源上升沿到第二个源下降沿指定电平处的时间差	✓
水平	FRD@Lv (LFRD)	计算从第一个源下降沿到第二个源上升沿指定电平处的时间差	✓
水平	偏移 (SKEW)	计算从第一个源的上升沿 50%处到第二个源的上升沿 50%处的时间差	✓
水平	数据点数 (WLEN)	参与测量的波形采样点的样本数目	
水平	周期数 (CYCL)	周期波形在波形区域内的循环次数	

水平	建立时间 (STPT)	从超过数据源上指定中间参考电平到最近一次超过时钟源上的指定中间参考电平的时间	✓
水平	保持时间 (HLDT)	从超过时钟源上的指定中间参考电平到最近一次超过数据源上的指定中间参考电平的时间	✓
其他	面积 (AREA)	波形的面积是指测量波形和接地电平之间的面积	
其他	周期面积 (CAR)	波形在一个周期内的面积	

参数快照

:MEASure:QKMEasure:DISPlay

■ 命令格式:

:MEASure:QKMEasure:DISPlay <active>

:MEASure:QKMEasure:DISPlay?

■ 功能描述:

设置或查询参数快照测量开关状态。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:MEASure:QKMEasure:DISPlay ON

设置参数快照测量开启

:MEASure:QKMEasure:DISPlay ?

查询返回 1

:MEASure:QKMEasure:SOURce

■ 命令格式:

:MEASure:QKMEasure:SOURce <source>

:MEASure:QKMEasure:SOURce ?

■ 功能描述:

设置或查询参数快照测量源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:MEASure:QKMEasure:SOURce C1

设置参数快照测量源为通道 1

:MEASure:QKMEasure:SOURce ?

查询返回 C1

:MEASure:QKMEasure:DATA

■ **命令格式:**

:MEASure:QKMEasure:DATA ?

■ **功能描述:**

查询参数快照中所有测量参数的结果。

其中参数快照包含以下测量参数，并按照如下排列：请参考[测量参数表](#)

"MAX", "MIN", "PKPK", "TOP", "BASE", "MID", "AMPL", "AVE", "RMS", "STDD", "POST", "NOST",
"AREA", "PRD", "FREQ", "RISE", "FALL", "PWID", "NWID", "DUTY", "NDTY", "TMAX", "TMIN",
"CMAX", "CMIN", "CRMS", "CAVE", "CPK", "CAR", "CMID", "WLEN", "CYCL", "PPST", "NPST"

■ **返回格式:**

查询返回相应通道参数快照测量所有结果，符合[数据块格式](#)，以科学计数法表示，并按顺序以逗号按分隔排列，无效值使用实型数据最大值表示。

例如：“#90000004475.26667E-001,-5.20000E-001,1.04667E+000,5.07246E-001,...”

参数测量

:MEASure:ALL:ACTive

■ **命令格式:**

:MEASure:ALL:ACTive <active>

■ **功能描述:**

设置所有测量项是否激活。

■ **参数:**

active:{ON | OFF}或{1 | 0}

■ **举例:**

:MEASure:ALL:ACTive ON

设置所有测量项都激活

:MEASure:ULTRaAcq

■ **命令格式:**

:MEASure:ULTRaAcq <mode>

■ **功能描述:**

设置快速开启垂直类型或水平类型的多个测量项。

■ **参数:**

mode:离散型{VERTical | HORizontal}

VERTical:垂直

HORizontal:水平

■ **举例:**

:MEASure:ULTRaAcq VERTical

设置快速开启垂直类型的多个测量项

:MEASure:GATe

■ **命令格式:**

:MEASure:GATe <type>

:MEASure:GATe ?

■ **功能描述:**

设置或查询测量门限。

■ **参数:**

type:离散型{SCReen | CURSor}

SCReen:屏幕

CURSor:光标

■ **返回格式:**

查询返回“SCReen”、“CURSor”。

■ **举例:**

:MEASure:GATe SCReen

设置测量门限为屏幕

:MEASure:GATe ?

查询返回 SCReen

:MEASure:INDicator

■ **命令格式:**

:MEASure:INDicator <indicator>

:MEASure:INDicator ?

■ **功能描述:**

设置或查询指示器。

■ **参数:**

indicator:离散型{OFF | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10}

■ **返回格式:**

查询返回“OFF”、“P1”、“P2”、“P3”、“P4”、“P5”、“P6”、“P7”、“P8”、“P9”、“P10”。

■ **举例:**

:MEASure:INDicator P1

设置 P1 的指示器打开

:MEASure:INDicator ?

查询返回 P1

:MEASure:ITEM<n>:DISPlay

■ **命令格式:**

:MEASure:ITEM<n>:DISPlay <active>

:MEASure:ITEM<n>:DISPlay ?

■ **功能描述:**

设置或查询单个测量项是否激活。

■ **参数:**

active:{ON | OFF}或者{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:MEASure:ITEM1:DISPlay ON

设置单个测量项 P1 打开

:MEASure:ITEM1:DISPlay ?

查询返回 1

:MEASure:STATistics:DISPlay

■ **命令格式:**

:MEASure:STATistics:DISPlay <active>

:MEASure:STATistics:DISPlay ?

■ **功能描述:**

设置或查询统计是否打开。

■ **参数:**

active:{ON | OFF}或者{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:MEASure:STATistics:DISPlay ON

设置统计打开

:MEASure:STATistics:DISPlay ?

查询返回 1

:MEASure:STATistics:RESet

■ **命令格式:**

:MEASure:STATistics:RESet

■ **功能描述:**

设置统计复位。

:MEASure:ITEM<n>:TYPE

■ **命令格式:**

:MEASure:ITEM<n>:TYPE <type>

:MEASure:ITEM<n>:TYPE ?

■ **功能描述:**

设置或查询测量项类型。

■ **参数:**

n: 离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

type:离散型, 表示测量参数类型, 请参考[测量参数表](#)。

{MAX | MIN | PKPK | TOP | BASE | MID | AMPL | AVE | RMS | STDD | POST | NOST | PPST | NPST | AREA | PRD | FREQ | RISE | FALL | PWID | NWID | DUTY | NDTY | CYCL | TMAX | TMIN | CMAX | CMIN | CRMS | CAVE | CPK | CMID | CAR | LRIS | LFAL | LPRD | LFRQ | LWID | LDUT | LPHA | LRRD | LFFD | LRFD | LFRD | SKEW | WLEN | STPT | HLDT}

■ **返回格式:**

查询返回“MAX”、“MIN”、“PKPK”、“TOP”、“BASE”、“MID”、“AMPL”、“AVE”、“RMS”、“STDD”、“POST”、“NOST”、“PPST”、“NPST”、“AREA”、“PRD”、“FREQ”、“RISE”、“FALL”、“PWID”、“NWID”、“DUTY”、“NDTY”、“CYCL”、“TMAX”、“TMIN”、“CMAX”、“CMIN”、“CRMS”、“CAVE”、“CPK”、“CMID”、“CAR”、“LRIS”、“LFAL”、“LPRD”、“LFRQ”、“LWID”、“LDUT”、“LPHA”、“LRRD”、“LFFD”、“LRFD”、“LFRD”、“SKEW”、“WLEN”、“STPT”、“HLDT”。

■ **举例:**

:MEASure:ITEM1:TYPe MAX

设置测量位 P1 的测量项为最大值

:MEASure:ITEM1:TYPe ?

查询返回 MAX

:MEASure:ITEM<n>:SOURce<ch>

■ 命令格式:

:MEASure:ITEM<n>:SOURce<ch> <source>

:MEASure:ITEM<n>:SOURce<ch> ?

■ 功能描述:

设置或查询测量位测量源。

■ 参数:

n: 离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

source:离散型{C1 | C2 | C3 | C4 | R1 | R2 | R3 | R4}

ch:离散型{1 | 2}, 表示信源 1 或信源 2。

如果测量参数是单信源则只需设置信源 1, 信源 2 可以省略。

如果测量参数是多信源则需要设置信源 1 和信源 2, 信源 2 不可以省略。

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”、“R1”、“R2”、“R3”、“R4”。

■ 举例:

:MEASure:ITEM1:SOURce1 C1

设置测量位 P1 的信源 1 为通道 1

:MEASure:ITEM1:SOURce1 ?

查询返回 C1

:MEASure:ITEM<n>:THRold:TYPe

■ 命令格式:

:MEASure:ITEM<n>:THRold:TYPe <type>

:MEASure:ITEM<n>:THRold:TYPe ?

■ 功能描述:

设置或查询测量位参考阈值-寻顶方式, 在特定测量项类型下才能设置成功。

■ 参数:

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

type:寻顶方式{BASetop | MINMax | ZERMin | ZERMax}

BASetop:底值-顶值

MINMax:最小-最大值

ZERMin:零-最小值

ZERMax:零-最大值

■ **返回格式:**

查询返回“BASetop”、“MINMax”、“ZERMin”、“ZERMax”。

■ **举例:**

:MEASure:ITEM1:THRold:TYPe BASetop	设置测量位 P1 的参考阈值-寻顶方式为顶底值
:MEASure:ITEM1:THRold:TYPe ?	查询返回 BASetop

:MEASure:ITEM<n>:THRold:TOP

■ **命令格式:**

:MEASure:ITEM<n>:THRold:TOP <top>

:MEASure:ITEM<n>:THRold:TOP ?

■ **功能描述:**

设置或查询测量项阈值高参考值，有效值 25%~95%，在特定测量项类型下才能设置成功。

■ **参数:**

n:离散型{1|2|3|4|5|6|7|8|9|10}。表示第 n 项测量参数。

top:整型，高参考值

■ **返回格式:**

查询返回阈值。

■ **举例:**

:MEASure:ITEM1:THRold:TOP 50	设置测量位 P1 的参考阈值-顶的阈值为 50%
:MEASure:ITEM1:THRold:TOP ?	查询返回 50

:MEASure:ITEM<n>:THRold:MIDDLE

■ **命令格式:**

:MEASure:ITEM<n>:THRold:MIDDLE <middle>

:MEASure:ITEM<n>:THRold:MIDDLE ?

■ **功能描述:**

设置或查询测量项阈值中参考值，有效值 15%~85%，在特定测量项类型下才能设置成功。

■ **参数:**

n:离散型{1|2|3|4|5|6|7|8|9|10}。表示第 n 项测量参数。

middle:整型，中参考值

■ **返回格式:**

查询返回阈值。

■ **举例:**

:MEASure:ITEM1:THRold:MIDDLE 50

设置测量位 P1 的参考阈值-中的阈值为 50%

:MEASure:ITEM1:THRold:MIDDLE ?

查询返回 50

:MEASure:ITEM<n>:THRold:BOTTOM

■ **命令格式:**

:MEASure:ITEM<n>:THRold:BOTTOM <bottom>

:MEASure:ITEM<n>:THRold:BOTTOM ?

■ **功能描述:**

设置或查询测量项阈值低参考值，有效值 5%~75%，在特定测量项类型下才能设置成功。

■ **参数:**

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

bottom:整型，低参考值

■ **返回格式:**

查询返回阈值。

■ **举例:**

:MEASure:ITEM1:THRold:BOTTOM 50

设置测量位 P1 的参考阈值-底的阈值为 50%

:MEASure:ITEM1:THRold:BOTTOM ?

查询返回 50

:MEASure:ITEM<n>:VALue

■ **命令格式:**

:MEASure:ITEM<n>:VALue ?

■ **功能描述:**

查询测量项测量值。

■ **参数:**

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

■ **返回格式:**

查询返回测量值。

:MEASure:ITEM<n>:COUNTER:MAX

■ **命令格式:**

:MEASure:ITEM<n>:COUNTER:MAX ?

■ 功能描述:

查询测量项统计值-最大值。

■ 参数:

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

■ 返回格式:

查询返回测量项统计值的最大值。

:MEASure:ITEM<n>:COUNter:MIN**■ 命令格式:**

:MEASure:ITEM<n>:COUNter:MIN ?

■ 功能描述:

查询测量项统计值-最小值。

■ 参数:

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

■ 返回格式:

查询返回测量项统计值的最小值。

:MEASure:ITEM<n>:COUNter:AVG**■ 命令格式:**

:MEASure:ITEM<n>:COUNter:AVG ?

■ 功能描述:

查询测量项统计值-平均值。

■ 参数:

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

■ 返回格式:

查询返回测量项统计值的平均值。

:MEASure:ITEM<n>:COUNter:DEV**■ 命令格式:**

:MEASure:ITEM<n>:COUNter:DEV ?

■ 功能描述:

查询测量项统计值-标准差。

- **参数:**

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

- **返回格式:**

查询返回测量项统计值的标准差。

:MEASure:ITEM<n>:COUNter:POP

- **命令格式:**

:MEASure:ITEM<n>:COUNter:POP?

- **功能描述:**

查询测量项统计-计数值。

- **参数:**

n:离散型{1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 }。表示第 n 项测量参数。

- **返回格式:**

查询返回测量项统计的计数值。

:MEASure:ITEM<n>:HISTgram

- **命令格式:**

:MEASure:ITEM<n>:HISTgram <active>

:MEASure:ITEM<n>:HISTgram ?

- **功能描述:**

设置或查询直方图是否打开。

- **参数:**

active:{ON | OFF}或者{1 | 0}

- **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

- **举例:**

:MEASure:ITEM1:HISTgram ON

设置直方图打开

:MEASure:ITEM1:HISTgram ?

查询返回 1

:MEASure:ITEM<n>:TREND

- **命令格式:**

:MEASure:ITEM<n>:TREND <active>

:MEASure:ITEM<n>:TRENd ?

■ **功能描述:**

设置或查询趋势图是否打开。

■ **参数:**

active:{ON | OFF}或者{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:MEASure:ITEM1:TRENd ON

设置趋势图打开

:MEASure:ITEM1:TRENd ?

查询返回 1

:MEASure:ITEM<n>:TRACK

■ **命令格式:**

:MEASure:ITEM<n>:TRACk <active>

:MEASure:ITEM<n>:TRACk ?

■ **功能描述:**

设置或查询追踪是否打开。

■ **参数:**

active:{ON | OFF}或者{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:MEASure:ITEM1:TRACk ON

设置追踪打开

:MEASure:ITEM1:TRACk ?

查询返回 1

数学

:MATH<n>:ACTive

■ **命令格式:**

:MATH<n>:ACTive <active>

:MATH<n>:ACTive ?

■ **功能描述:**

设置或查询指定数学通道开启状态。

- **参数:**

active:布尔型{ON | OFF}或{1 | 0}

- **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

- **举例:**

:MATH1:ACTive ON 设置数学通道 1 打开

:MATH1:ACTive ? 查询返回 1

:MATH<n>:LABel:NAME

- **命令格式:**

:MATH<n>:LABel:NAME <label>

:MATH<n>:LABel:NAME ?

- **功能描述:**

设置或查询数学通道标签。

- **参数:**

Label:通道名称字符串

- **返回格式:**

查询返回通道名称字符串

- **举例:**

:MATH1:LABel:NAME "abcd" 设置数学通道 1 标签为 abcd

:MATH1:LABel:NAME ? 查询返回 abcd

:MATH<n>:UNIT:ENABLE

- **命令格式:**

:MATH<n>:UNIT:ENABLE <enable>

:MATH<n>:UNIT:ENABLE ?

- **功能描述:**

设置或查询数学通道自定义单位开启状态。

- **参数:**

enable:布尔型{ON | OFF}或{1 | 0}

- **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

- **举例:**

:MATH1:UNIT:ENABLE ON

设置数学通道 1 自定义单位打开

:MATH1:UNIT:ENABLE ?

查询返回 1

:MATH<n>:UNIT:NAME

■ 命令格式:

:MATH<n>:UNIT:NAME <unit>

:MATH<n>:UNIT:NAME ?

■ 功能描述:

设置或查询指定数学通道显示单位名称，在自定义单位开启的时候才能设置成功。

■ 参数:

unit:单位字符串

■ 返回格式:

查询返回指定数学通道的显示单位名称

■ 举例:

:MATH1:UNIT:NAME "A"

设置数学通道 1 单位为 A

:MATH1:UNIT:NAME ?

查询返回 A

:MATH<n>:TYPE

■ 命令格式:

:MATH<n>:TYPE <type>

:MATH<n>:TYPE ?

■ 功能描述:

设置或查询数学通道数学运算类型。

■ 参数:

type:离散型{BINary | FFT | FILTer | ERES | CUSTom | ADVanced}

BINary:基本运算

FFT:快速傅里叶变化

FILTer:滤波

ERES:增强分辨率

CUSTom:自定义

ADVanced:高级运算

■ 返回格式:

查询返回“BINary”、“FFT”、“FILTer”、“ERES”、“CUSTom”、“ADVanced”。

■ **举例:**

:MATH1:TYPE FFT

设置数学通道 1 数学运算类型为 FFT

:MATH1:TYPE ?

查询返回 FFT

:MATH<n>:VERTical:OFFSet

■ **命令格式:**

:MATH<n>:VERTical:OFFSet <offset>

:MATH<n>:VERTical:OFFSet ?

■ **功能描述:**

设置或查询数学通道垂直位移。

■ **参数:**

offset:实数型, 单位 div, 范围为-4div-4div。

■ **返回格式:**

查询返回数学通道垂直位移。

■ **举例:**

:MATH1:VERTical:OFFSet 1

设置数学通道 1 垂直位移 1div

:MATH1:VERTical:OFFSet ?

查询返回 1.00000E+000

:MATH<n>:VERTical:SCALE

■ **命令格式:**

:MATH<n>:VERTical:SCALE <scale>

:MATH<n>:VERTical:SCALE ?

■ **功能描述:**

设置或查询数学通道垂直档位。

■ **参数:**

scale:实数型, 单位:自定义单位, 不输入单位则默认单位为 V。

■ **返回格式:**

查询返回数学通道垂直档位。

■ **举例:**

:MATH1:VERTical:SCALE 1

设置数学通道 1 垂直档位 1V

:MATH1:VERTical:SCALE ?

查询返回 1.00000E+000

:MATH<n>:HORizontal:OFFSet■ **命令格式:**

:MATH<n>:HORizontal:OFFSet <offset>

:MATH<n>:HORizontal:OFFSet ?

■ **功能描述:**

设置或查询数学通道水平位移。

■ **参数:**

offset:实数型, 单位 div, 范围为 0div-10div。

■ **返回格式:**

查询返回数学通道水平位移。

■ **举例:**

:MATH1:HORizontal:OFFSet 1

设置数学通道 1 水平位移 1div

:MATH1:HORizontal:OFFSet ?

查询返回 1.000000E+000

:MATH<n>:HORizontal:SCALe■ **命令格式:**

:MATH<n>:HORizontal:SCALe <scale>

:MATH<n>:HORizontal:SCALe ?

■ **功能描述:**

设置或查询数学通道水平档位。

■ **参数:**

scale:实数型, 单位:默认单位为 s, 范围:与主时基大小相关。

■ **返回格式:**

查询返回数学通道水平档位。

■ **举例:**

:MATH1:HORizontal:SCALe 1us

设置数学通道 1 水平档位 1us

:MATH1:HORizontal:SCALe ?

查询返回 1.0000000000000000E-006

:MATH<n>:SOURce<ch>■ **命令格式:**

:MATH<n>:SOURce<ch> <source>

:MATH<n>:SOURce<ch> <source> ?

■ 功能描述:

设置或查询数学通道运算源。

■ 参数:

ch:{1 | 2},表示信源 1 或者信源 2

source:离散型{C1 | C2 | C3 | C4 | R1 | R2 | R3 | R4}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”、“R1”、“R2”、“R3”、“R4”。

■ 举例:

:MATH1:SOURce1 C1 设置数学通道 1 运算源为通道 1

:MATH1:SOURce1 ? 查询返回 C1

说明：若只有一个源可以写 source1，也可以直接写 source

基本运算

:MATH<n>:OPERation

■ 命令格式:

:MATH<n>:OPERation <type>

:MATH<n>:OPERation ?

■ 功能描述:

设置或查询数学通道基础运算类型。

■ 参数:

type:离散型{ADD | SUBTract | MULTiPLY | DIVide}

ADD:加

SUBTract:减

MULTiPLY:乘

DIVide:除

■ 返回格式:

查询返回“ADD”、“SUBTract”、“MULTiPLY”、“DIVide”。

■ 举例:

:MATH1:OPERation ADD 设置数学通道 1 基础运算类型为加

:MATH1:OPERation ? 查询返回 ADD

:MATH<n>:FFT:WINDow

■ 命令格式:

:MATH<n>:FFT:WINDow <type>

:MATH<n>:FFT:WINDow ?

■ 功能描述:

运算类型为快速傅里叶变换（FFT）时，设置或查询数学通道窗函数。

■ 参数:

type:离散型{RECTangle | HANN | HAMMing | BLACkman | FLATtop}

RECTangle:矩形窗

HANN:汉宁窗

HAMMing:汉明窗

BLACkman:布莱克曼窗

FLATtop:平顶窗

■ 返回格式:

查询返回“RECTangle”、“HANN”、“HAMMing”、“BLACkman”、“FLATtop”。

■ 举例:

:MATH1:FFT:WINDow RECTangle

设置数学通道 1 窗函数为矩形窗

:MATH1:FFT:WINDow ?

查询返回 RECTangle

:MATH<n>:FFT:TYPE

■ 命令格式:

:MATH<n>:FFT:TYPE <type>

:MATH<n>:FFT:TYPE ?

■ 功能描述:

运算类型为快速傅里叶变换（FFT）时，设置或查询数学通道输出类型。

■ 参数:

type:离散型{AMPLtd | POWER | PSD | REAL | IMAGinary | PHASe}

AMPLtd :幅度谱

POWER:功率谱

PSD:Psd

REAL:实部

IMAGinary 虚部

PHASe:相位谱

■ 返回格式:

查询返回“AMPLtd”、“POWeR”、“PSD”、“REAL”、“IMAGinary”、“PHASe”。

■ 举例:

:MATH1:FFT:TYPe AMPLtd

设置数学通道 1 输出类型为幅度谱

:MATH1:FFT:TYPe ?

查询返回 AMPLtd

:MATH<n>:FFT:UNIT

■ 命令格式:

:MATH<n>:FFT:UNIT <type>

:MATH<n>:FFT:UNIT ?

■ 功能描述:

快速傅里叶变换（FFT）输出类型为幅度谱时，设置或查询数学通道 FFT 单位。

■ 参数:

type:离散型{VRMS | DBM | DBUW | DBMV | DBUV | DBMA | DBUA}

■ 返回格式:

查询返回“VRMS”、“DBM”、“DBUW”、“DBMV”、“DBUV”、“DBMA”、“DBUA”。

■ 举例:

:MATH1:FFT:UNIT VRMS

设置数学通道 1 的 FFT 单位为 VRMS

:MATH1:FFT:UNIT ?

查询返回 VRMS

:MATH<n>:FFT:POINT

■ 命令格式:

:MATH<n>:FFT:POINT <type>

:MATH<n>:FFT:POINT ?

■ 功能描述:

运算类型为快速傅里叶变换（FFT）时，设置或查询数学通道扫描点数。

■ 参数:

type:离散型{1k | 2k | 4k | 8k | 16k | 32k | 64k | 128k | 256k | 512k | 1M}, 默认单位 Pts。

■ 返回格式:

查询返回“1k”、“2k”、“4k”、“8k”、“16k”、“32k”、“64k”、“128k”、“256k”、“512k”、“1M”。

■ 举例:

:MATH1:FFT:POINT 1k 设置数学通道 1 扫描点数为 1Kpts

:MATH1:FFT:POINT ? 查询返回 1K

:MATH<n>:FFT:FREQuency:CENTer

■ 命令格式:

:MATH<n>:FFT:FREQuency:CENTer <freq>

:MATH<n>:FFT:FREQuency:CENTer ?

■ 功能描述:

运算类型为快速傅里叶变换(FFT)时, 设置或查询指定 FFT 的中心频率, 默认单位 Hz。

■ 参数:

freq:实数型, 单位 mHz、Hz、kHz、MHz

■ 返回格式:

查询返回中心频率。

■ 举例:

:MATH1:FFT:FREQuency:CENTer 100 设置数学通道 1FFT 中心频率 100Hz

:MATH1:FFT:FREQuency:CENTer ? 查询返回 1.00000000000000E+002

:MATH<n>:FFT:FREQuency:SPAN

■ 命令格式:

:MATH<n>:FFT:FREQuency:SPAN

:MATH<n>:FFT:FREQuency:SPAN ?

■ 功能描述:

运算类型为快速傅里叶变换(FFT)时, 设置或查询指定 FFT 的跨度, 默认单位 Hz。

■ 参数:

span:实数型, 单位 Hz、kHz、MHz

■ 返回格式:

查询返回频率跨度。

■ 举例:

:MATH1:FFT:FREQuency:SPAN 100000 设置数学通道 1FFT 频率跨度 100kHz

:MATH1:FFT:FREQuency:SPAN ? 查询返回 1.00000000000000E+005

说明:频率跨度实际可设置范围与当前时基刻度有关。

自定义

:MATH<n>:EXPRession

■ 命令格式:

:MATH<n>:EXPRession <expression>

:MATH<n>:EXPRession ?

■ 功能描述:

利用自由组合表达式进行数学计算。

表达式格式可见示波器下 MATH 菜单下自定义类型公式编辑窗，<expression>属于 ASCII 字符串参数。

■ 参数:

expression:自定义运算字符串。

自定义运算仅支持模拟通道和参考通道。

■ 举例:

:MATH1:EXPRession "C1 * C2" 表示将通道一和二相乘

:MATH1:EXPRession ? 查询返回 C1 * C2

注意：公式中所含函数表达式均需小写（通道字母固定大写）

ERes

:MATH<n>:ERES:BIT

■ 命令格式:

:MATH<n>:ERES:BIT <bit>

:MATH<n>:ERES:BIT ?

■ 功能描述:

运算类型为 ERes 时，设置或查询增强位数。

■ 参数:

bit:实数型，范围 0.5-3，步进 0.5

■ 返回格式:

查询返回增强位数。

■ 举例:

:MATH1:ERES:BIT 1 设置数学通道 1 的增强位数为 1

:MATH1:ERES:BIT ? 查询返回 1.00000E+000

滤波

:MATH<n>:FILTer:TYPe

■ 命令格式:

:MATH<n>:FILTer:TYPe <type>

:MATH<n>:FILTer:TYPe ?

■ 功能描述:

运算类型为 Filter 时，设置或查询滤波类型。

■ 参数:

type:离散型{LOWPass | HIGHpass | BNDPass | BNDStop}

LOWPass:低通

HIGHpass:高通

BNDPass:带通

BNDStop:带阻

■ 返回格式:

查询返回“LOWPass”、“HIGHpass”、“BNDPass”、“BNDStop”。

■ 举例:

:MATH1:FILTer:TYPe BNDStop 设置数学通道 1 的滤波类型为带阻

:MATH1:FILTer:TYPe ? 查询返回 BNDStop

:MATH<n>:FILTer:FREQuency<n>:STOP

■ 命令格式:

:MATH<n>:FILTer:FREQuency<n>:STOP <freq>

:MATH<n>:FILTer:FREQuency<n>:STOP ?

■ 功能描述:

运算类型为 Filter 时，设置或查询截止频率，默认单位 Hz。

■ 参数:

n:离散型{1 | 2}，表示截止频率 1 和截至频率 2

freq:实数型，单位 mHz、Hz、kHz、MHz

■ 返回格式:

查询返回截至频率。

■ 举例:

:MATH1:FILTer:FREQuency1:STOP 100

设置数学通道 1 的滤波截止频率 1 为 100Hz

:MATH1:FILTer:FREQuency1:STOP ?

查询返回 1.00000000000000E+002

:MATH<n>:FILTer:CUSTom:ENABle

■ 命令格式:

:MATH<n>:FILTer:CUSTom:ENABle <enable>

:MATH<n>:FILTer:CUSTom:ENABle ?

■ 功能描述:

运算类型为 Filter 时，设置或查询自定义滤波器的打开状态。

■ 参数:

enable:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:MATH1:FILTer:CUSTom:ENABle ON

设置数学通道 1 自定义滤波器打开

:MATH1:FILTer:CUSTom:ENABle ?

查询返回 1

频率计

:CYMometer:ACTive

■ 命令格式:

:CYMometer:ACTive <active>

:CYMometer:ACTive ?

■ 功能描述:

设置或查询频率计开启状态。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:CYMometer:ACTive ON

设置频率计打开

:CYMometer:ACTive ?

查询返回 1

:CYMometer:FREQuency

- **命令格式:**
:CYMometer:FREQuency ?
- **功能描述:**
查询返回频率计的值。
- **返回格式:**
查询返回频率计测量值，以 Hz 为单位。

电压表**:DVM:ACTive**

- **命令格式:**
:DVM:ACTive <active>
:DVM:ACTive ?
- **功能描述:**
设置或查询电压表开启状态。
- **参数:**
active:布尔型{ON | OFF}或{1 | 0}
- **返回格式:**
查询返回“1”、“0”分别代表“ON”、“OFF”。
- **举例:**

:DVM:ACTive ON	设置电压表打开
:DVM:ACTive ?	查询返回 1

:DVM:SOURce

- **命令格式:**
:DVM:SOURce <source>
:DVM:SOURce ?
- **功能描述:**
设置或查询电压表源。
- **参数:**
source:{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”。

■ 举例:

:DVM:SOURce C1	设置电压表源为 C1
:DVM:SOURce ?	查询返回 C1

:DVM:MODE**■ 命令格式:**

:DVM:MODE <mode>

:DVM:MODE ?

■ 功能描述:

设置或查询电压表模式。

■ 参数:

mode:{DC | ACRMs | DCACrms}

■ 返回格式:

查询返回“DC”、“ACRMs”、“DCACrms”。

■ 举例:

:DVM:MODE DC	设置电压表模式为 DC
:DVM:MODE ?	查询返回 DC

:DVM:AUTO**■ 命令格式:**

:DVM:AUTO <active>

:DVM:AUTO ?

■ 功能描述:

设置或查询电压表自动量程开启状态。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:DVM:AUTO ON	设置电压表自动量程打开
:DVM:AUTO ?	查询返回 1

:DVM:CURRent

- **命令格式:**
:DVM:CURRent ?
- **功能描述:**
查询返回电压表的值。
- **返回格式:**
查询返回电压表测量值，以 V 为单位。

显示**:DISPlay:PERsist**

- **命令格式:**
:DISPlay:PERsist <persist>
:DISPlay:PERsist ?
- **功能描述:**
设置或查询无限余辉状态。
- **参数:**
persist:离散型 {CLOSe | AUTO | INFInity}
CLOSe:关闭
AUTO:自动
INFInity:无限
- **返回格式:**
查询返回无限余辉状态“CLOSe”、“AUTO”、“INFInity”。
- **举例:**
:DISPlay:PERsist CLOSe 设置无限余辉关闭
:DISPlay:PERsist ? 查询返回 CLOSe

:DISPlay:MARK

- **命令格式:**
:DISPlay:MARK <mark>
:DISPlay:MARK?
- **功能描述:**

设置或查询显示标记。

■ **参数:**

mark:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:DISPlay:MARK ON 设置显示标记打开

:DISPlay:MARK ? 查询返回 1

:DISPlay:MKVert

■ **命令格式:**

:DISPlay:MKVert <mkvert>

:DISPlay:MKVert ?

■ **功能描述:**

设置或查询垂直标记位置。

■ **参数:**

mkvert:{LEFT | RIGHT}

LEFT:左

RIGHT:右

■ **返回格式:**

查询返回垂直标记位置“LEFT”、“RIGHT”。

■ **举例:**

:DISPlay:MKVert LEFT 设置垂直标记位置为左

:DISPlay:MKVert ? 查询返回 LEFT

:DISPlay:MKHori

■ **命令格式:**

:DISPlay:MKHori <mkHori>

:DISPlay:MKHori ?

■ **功能描述:**

设置或查询水平标记位置。

■ **参数:**

mkHori:{TOP | BOTTom}

TOP:上

BOTTOM:下

■ **返回格式:**

查询返回水平标记位置“TOP”、“BOTTOM”。

■ **举例:**

:DISPlay:MKHori TOP 设置水平标记位置为上

:DISPlay:MKHori ? 查询返回 TOP

:DISPlay:GRID:STYLE

■ **命令格式:**

:DISPlay:GRID:STYLE <style>

:DISPlay:GRID:STYLE ?

■ **功能描述:**

设置或查询网格样式。

■ **参数:**

style:{FULL | BRIef | NONe}

FULL:满

BRIef:简单

NONe:无

■ **返回格式:**

查询返回网格样式“FULL”、“BRIef”、“NONe”。

■ **举例:**

:DISPlay:GRID:STYLE FULL 设置网格样式为满

:DISPlay:GRID:STYLE ? 查询返回 FULL

:DISPlay:GRID:BRIGtness

■ **命令格式:**

:DISPlay:GRID:BRIGtness <value>

:DISPlay:GRID:BRIGtness ?

■ **功能描述:**

设置或查询网格亮度。

■ **参数:**

value:整数型, 范围: 0~100。

■ 返回格式:

查询返回网格亮度值。

■ 举例:

:DISPlay:GRID:BRIGtness 50

设置网格亮度值 50

:DISPlay:GRID:BRIGtness ?

查询返回 50

:DISPlay:WAVe:STYLe**■ 命令格式:**

:DISPlay:WAVe:STYLe <style>

:DISPlay:WAVe:STYLe ?

■ 功能描述:

设置或查询波形样式。

■ 参数:

style:{VECTor | DOT}

VECTor:矢量

DOT:点

■ 返回格式:

查询返回波形样式“VECTor”、“DOT”。

■ 举例:

:DISPlay:WAVe:STYLe VECTor

设置波形样式为矢量

:DISPlay:WAVe:STYLe ?

查询返回 VECTor

:DISPlay:WAVe:BRIGtness**■ 命令格式:**

:DISPlay:WAVe:BRIGtness <value>

:DISPlay:WAVe:BRIGtness ?

■ 功能描述:

设置或查询波形亮度。

■ 参数:

value:整数型, 范围: 0~100。

■ 返回格式:

查询返回波形亮度值。

■ 举例:

:DISPlay:WAVe:BRIGtness 50

设置波形亮度值 50

:DISPlay:WAVe:BRIGtness ?

查询返回 50

:DISPlay:SCReen:BRIGtness

■ 命令格式:

:DISPlay:SCReen:BRIGtness <value>

:DISPlay:SCReen:BRIGtness ?

■ 功能描述:

设置或查询屏幕亮度。

■ 参数:

value:整数型, 范围: 5~100。

■ 返回格式:

查询返回屏幕亮度值。

■ 举例:

:DISPlay:SCReen:BRIGtness 50

设置屏幕亮度值 50

:DISPlay:SCReen:BRIGtness ?

查询返回 50

:DISPlay:SCReen:CONTRast

■ 命令格式:

:DISPlay:SCReen:CONTRast <value>

:DISPlay:SCReen:CONTRast ?

■ 功能描述:

设置或查询屏幕对比度。

■ 参数:

value:整数型, 范围: 50~100。

■ 返回格式:

查询返回屏幕对比度值。

■ 举例:

:DISPlay:SCReen:CONTRast 50

设置屏幕对比度值 50

:DISPlay:SCReen:CONTRast ?

查询返回 50

存储

波形

:FILE:WAVE:FORMat

■ 命令格式:

:FILE:WAVE:FORMat <format>

:FILE:WAVE:FORMat ?

■ 功能描述:

设置或查询波形保存格式。

■ 参数:

format:{BINary | TEXT | MATlab | EXCel | CSV | TSV | DAT}

■ 返回格式:

查询返回波形保存格式“BINary”、“TEXT”、“MATlab”、“EXCel”、“CSV”、“TSV”、“DAT”。

■ 举例:

:FILE:WAVE:FORMat TEXT

设置波形保存格式 TEXT

:FILE:WAVE:FORMat ?

查询返回 TEXT

:FILE:WAVE:TXTEncoding

■ 命令格式:

:FILE:WAVE:TXTEncoding <encoding>

:FILE:WAVE:TXTEncoding ?

■ 功能描述:

设置或查询当波形保存格式为文本时的文本编码格式。

■ 参数:

encoding:{ASCII | GB2312 | UTF8 | UTF32 | UINCode}

■ 返回格式:

查询返回文本编码格式“ASCII”、“GB2312”、“UTF8”、“UTF32”、“UINCode”。

■ 举例:

:FILE:WAVE:TXTEncoding ASCII

设置文本编码格式 ASCII

:FILE:WAVE:TXTEncoding ?

查询返回 ASCII

:FILE:WAVE:SOURce■ **命令格式:**

:FILE:WAVE:SOURce <source>

:FILE:WAVE:SOURce ?

■ **功能描述:**

设置或查询波形保存的源。

■ **参数:**

source:{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回波形保存源“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:FILE:WAVE:SOURce C1 设置波形保存源 C1

:FILE:WAVE:SOURce ? 查询返回 C1

:FILE:WAVE:DATetime:ENABLE■ **命令格式:**

:FILE:WAVE:DATetime:ENABLE <enable>

:FILE:WAVE:DATetime:ENABLE ?

■ **功能描述:**

设置或查询波形保存名称是否追加日期。

■ **参数:**

enable:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:FILE:WAVE:DATetime:ENABLE OFF 设置波形保存名称不追加日期

:FILE:WAVE:DATetime:ENABLE ? 查询返回 0

:FILE:WAVE:NAME■ **命令格式:**

:FILE:WAVE:NAME <name>

:FILE:WAVE:NAME ?

■ **功能描述:**

设置或查询波形保存名称。

■ **参数:**

name:可自定义输入保存波形名称。若不输入名称则默认名称自动叠加序号。

■ **返回格式:**

查询返回波形名称。

■ **举例:**

:FILE:WAVE:NAME abcd

设置波形保存名称 abcd

:FILE:WAVE:NAME ?

查询返回 abcd

:FILE:WAVE:PATH

■ **命令格式:**

:FILE:WAVE:PATH ?

■ **功能描述:**

查询波形保存路径。

■ **返回格式:**

查询返回波形路径。

:FILE:WAVE:SAVe

■ **命令格式:**

:FILE:WAVE:SAVe ?

■ **功能描述:**

保存波形数据并返回是否保存成功。

■ **返回格式:**

查询返回离散型{SUCCess | FAIL}。

SUCCess:保存成功

FAIL: 保存失败

截屏

:FILE:PICTure:REGion

■ 命令格式:

:FILE:PICTure:REGion <region>

:FILE:PICTure:REGion ?

■ 功能描述:

设置或查询图片保存区域。

■ 参数:

region:{WINDow | APPLication}。

WINDow:栅格

APPLication:屏幕

■ 返回格式:

查询返回“WINDow”、“APPLication”。

■ 举例:

:FILE:PICTure:REGion WINDow

设置图片保存栅格区域

:FILE:PICTure:REGion ?

查询返回 WINDow

:FILE:PICTure:COLor

■ 命令格式:

:FILE:PICTure:COLor <color>

:FILE:PICTure:COLor ?

■ 功能描述:

设置或查询图片保存颜色。

■ 参数:

color:{STANdard | BLACkwhite | REVerse}。

STANdard:标准

BLACkwhite:黑白

REVerse:反色

■ 返回格式:

查询返回“STANdard ”、“BLACkwhite”、“REVerse”。

■ 举例:

:FILE:PICTure:COLor STANDARD	设置图片保存颜色为标准
:FILE:PICTure:COLor ?	查询返回 STANDard

:FILE:PICTure:FORMat

■ 命令格式:

:FILE:PICTure:FORMat <format>

:FILE:PICTure:FORMat ?

■ 功能描述:

设置或查询图片保存格式。

■ 参数:

format:{BMP | TIFF | GIF | Png | JPEG}。

■ 返回格式:

查询返回“BMP”、“TIFF”、“GIF”、“Png”、“JPEG”。

■ 举例:

:FILE:PICTure:FORMat JPEG	设置图片保存格式为 JPEG
:FILE:PICTure:FORMat ?	查询返回 JPEG

:FILE:PICTure:NAME

■ 命令格式:

:FILE:PICTure:NAME <name>

:FILE:PICTure:NAME ?

■ 功能描述:

设置或查询图片保存名称。

■ 参数:

name:离散型，可自定义输入保存波形名称。若不输入名称则默认名称自动叠加序号。

■ 举例:

:FILE:PICTure:NAME abcd	设置图片保存名称 abcd
:FILE:PICTure:NAME ?	查询返回 abcd

:FILE:PICTure:DATetime:ENABLE

■ 命令格式:

:FILE:PICTure:DATetime:ENABLE <enable>

:FILE:PICTure:DATetime:ENABle ?

■ **功能描述:**

设置或查询图片保存名称是否追加日期。

■ **参数:**

enable:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:FILE:PICTure:DATetime:ENABle OFF

设置图片保存名称不追加日期

:FILE:PICTure:DATetime:ENABle ?

查询返回 0

:FILE:PICTure:PATH

■ **命令格式:**

:FILE:PICTure:PATH ?

■ **功能描述:**

查询图片保存路径。

■ **返回格式:**

查询返回图片路径。

:FILE:PICTure:SAVe

■ **命令格式:**

:FILE:PICTure:SAVe ?

■ **功能描述:**

保存图片数据并返回是否保存成功。

■ **返回格式:**

查询返回离散型{SUCCess | FAIL}。

SUCCess:保存成功

FAIL:保存失败

打印

:PRINT:REGion

■ **命令格式:**

:PRINT:REGion <region>

:PRINT:REGion ?

■ **功能描述:**

设置或查询打印机打印区域。

■ **参数:**

region:{SCReen | GRID}

SCReen:屏幕

GRID:栅格

■ **返回格式:**

查询返回“SCReen”、“GRID”。

■ **举例:**

:PRINT:REGion SCReen

设置打印区域为屏幕

:PRINT:REGion ?

查询返回 SCReen

:PRINT:COLor

■ **命令格式:**

:PRINT:COLor <color>

:PRINT:COLor ?

■ **功能描述:**

设置或查询打印机打印颜色。

■ **参数:**

color:{STANdard | BLACkwhite | REVerse}。

STANdard:标准

BLACkwhite:黑白

REVerse:反色

■ **返回格式:**

查询返回“STANdard ”、“BLACkwhite”、“REVerse”。

■ **举例:**

:PRINT:COLor STANdard

设置打印颜色为“标准”

:PRINT:COLor ?

查询返回 STANdard

:PRINT:ORlent

■ **命令格式:**

:PRINT:ORlent <orient>

:PRINT:ORlent ?

■ **功能描述:**

设置或查询打印机打印方向。

■ **参数:**

orient:{HOR | VER}。

HOR:横向

VER:纵向

■ **返回格式:**

查询返回“HOR ”、“VER”。

■ **举例:**

:PRINT:ORlent HOR

设置打印方向为横向

:PRINT:ORlent ?

查询返回 HOR

LA

:LA:ITEM<n>:DISPlay

■ **命令格式:**

:LA:ITEM<n>:DISPlay <active>

:LA:ITEM<n>:DISPlay ?

■ **功能描述:**

设置或查询指定逻辑通道打开状态。

■ **参数:**

n:整数型，设置范围 0~15，表示逻辑通道 D0~D15。

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:LA:ITEM1:DISPlay ON

设置逻辑通道 1 打开

:LA:ITEM1:DISPlay ?

查询返回 1

:LA:ITEM<n>:OFFSet

■ **命令格式:**

:LA:ITEM<n>:OFFSet <offset>

:LA:ITEM<n>:OFFSet?

■ **功能描述:**

设置或查询指定逻辑通道的垂直位置偏移。

■ **参数:**

n:整数型, 设置范围 0~15, 表示逻辑通道 D0~D15。

offset:实数型, 设置范围-3.5div~4div。

■ **返回格式:**

查询返回指定逻辑通道的垂直位置偏移。

■ **举例:**

:LA:ITEM1:OFFSet 3 设置逻辑通道 1 的垂直位置偏移 3div

:LA:ITEM1:OFFSet ? 查询返回 3.00000E+000

:LA:ITEM<n>:LABel

■ **命令格式:**

:LA:ITEM<n>:LABel <lable>

:LA:ITEM<n>:LABel ?

■ **功能描述:**

设置或查询指定逻辑通道的标签。

■ **参数:**

n:整数型, 设置范围 0~15, 表示逻辑通道 D0~D15。

lable:通道名称字符串。

■ **返回格式:**

查询返回指定逻辑通道的标签。

■ **举例:**

:LA:ITEM1:LABel "abcd" 设置逻辑通道 1 的标签 abcd

:LA:ITEM1:LABel ? 查询返回 abcd

:LA:ITEM<n>:THReshold

■ **命令格式:**

:LA:ITEM<n>:THReshold <threshold>

:LA:ITEM<n>:THReshold ?

■ **功能描述:**

设置或查询指定逻辑通道的阈值电平属性。

■ **参数:**

n:整数型, 设置范围 0~15, 表示逻辑通道 D0~D15。

threshold:{TTL | CMOS5000 | CMOS3300 | CMOS2500 | CMOS1800 | ECL | PECL | LVDS | USER}。

■ **返回格式:**

查询返回“TTL”、“CMOS5000”、“CMOS3300”、“CMOS2500”、“CMOS1800”、“ECL”、“PECL”、“LVDS”、“USER”。

■ **举例:**

:LA:ITEM1:THReshold CMOS3300	设置逻辑通道 1 的阈值电平属性 CMOS3300
:LA:ITEM1:THReshold ?	查询返回 CMOS3300

:LA:ITEM<n>:LEVel

■ **命令格式:**

:LA:ITEM<n>:LEVel <level>

:LA:ITEM<n>:LEVel ?

■ **功能描述:**

设置或查询指定逻辑通道的阈值电平。

■ **参数:**

n:整数型, 设置范围 0~15, 表示逻辑通道 D0~D15。

level:实数型, 单位 mV、V, 默认单位 V。

■ **返回格式:**

查询返回逻辑通道的阈值电平。

■ **举例:**

:LA:ITEM1:LEVel 3	设置逻辑通道 1 的阈值电平 3
:LA:ITEM1:LEVel ?	查询返回 3.00000E+000

:LA:HEIGht

■ **命令格式:**

:LA:HEIGht <height>

:LA:HEIGht ?

■ **功能描述:**

设置或查询逻辑通道的高度。

■ 参数:

height:{SMALL | MEDium | LARGe}。

■ 返回格式:

查询返回“SMALL”、“MEDium”、“LARGe”。

■ 举例:

:LA:HEIGHt LARGe 设置逻辑通道的高度 LARGe

:LA:HEIGHt ? 查询返回 LARGe

:LA:AUTO:LOCate**■ 命令格式:**

:LA:AUTO:LOCate <auto>

■ 功能描述:

设置逻辑通道自动排序。

■ 参数:

auto:布尔型{ON | OFF}或{1 | 0}。

■ 举例:

:LA:AUTO:LOCate ON 设置逻辑通道的自动排序

:LA:ALL:DISPlay**■ 命令格式:**

:LA:ALL:DISPlay <active>

■ 功能描述:

设置所有逻辑通道打开。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}。

■ 举例:

:LA:ALL:DISPlay ON 设置所有逻辑通道打开

顺序模式**:SEGement:ACTive****■ 命令格式:**

:SEGement:ACTive <active>

:SEGement:ACTive ?

■ **功能描述:**

设置或查询顺序模式开启状态，在开启顺序模式的时候会自动打开快采。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:SEGement:ACTive ON 设置顺序模式打开

:SEGement:ACTive ? 查询返回 1

:SEGement:RSTacq

■ **命令格式:**

:SEGement:RSTacq <acq>

■ **功能描述:**

设置顺序模式是否重采。

■ **参数:**

acq:布尔型{ON | OFF}或{1 | 0}

ON:重采

OFF:不重采

■ **举例:**

:SEGement:RSTacq ON 设置顺序模式重采

:SEGement:MODE

■ **命令格式:**

:SEGement:MODE <mode>

:SEGement:MODE ?

■ **功能描述:**

设置或查询顺序模式的工作模式。

■ **参数:**

mode:{SINGle | SEQuent}

SINGle:单帧

SEQuent:连续帧

■ **返回格式:**

查询返回“SINGLe”、“SEQuent”。

■ **举例:**

:SEGement:MODE SEQuent 设置顺序模式的工作模式为连续帧模式

:SEGement:MODE ? 查询返回 SEQuent

:SEGement:FRAME:COUNT

■ **命令格式:**

:SEGement:FRAME:COUNT <count>

:SEGement:FRAME:COUNT ?

■ **功能描述:**

设置或查询顺序模式的总帧数。

■ **参数:**

count:整数型

■ **返回格式:**

查询返回顺序模式的总帧数。

■ **举例:**

:SEGement:FRAME:COUNT 100 设置顺序模式的的总帧数 100

:SEGement:FRAME:COUNT ? 查询返回 100

:SEGement:SEQuent:TYPE

■ **命令格式:**

:SEGement:SEQuent:TYPE <mode>

:SEGement:SEQuent:TYPE ?

■ **功能描述:**

设置或查询顺序模式在连续帧下的波形显示类型。

■ **参数:**

mode:{ANGLE | STACK | SUPERimpose | SPLiced | NONE}

ANGLE:45°

STACK:堆叠

SUPERimpose:叠加

SPLiced:拼接

NONE:无

■ **返回格式:**

查询返回“ANGLE”、“STACK”、“SUPERimpose”、“SPLiced”、“NONE”。

■ **举例:**

:SEGement:SEQuent:TYPe ANGLE

设置顺序模式的在连续帧下的波形显示类型为 45°

:SEGement:SEQuent:TYPe ?

查询返回 ANGLE

:SEGement:FRAME:START

■ **命令格式:**

:SEGement:FRAME:START <count>

:SEGement:FRAME:START ?

■ **功能描述:**

设置或查询顺序模式在连续帧模式下的起始帧。

■ **参数:**

count:整数型

■ **返回格式:**

查询返回顺序模式在连续帧模式下的起始帧。

■ **举例:**

:SEGement:FRAME:START 1

设置顺序模式的起始帧 1

:SEGement:FRAME:START ?

查询返回 1

:SEGement:FRAME:END

■ **命令格式:**

:SEGement:FRAME:END <count>

:SEGement:FRAME:END ?

■ **功能描述:**

设置或查询顺序模式在连续帧模式下的结束帧。

■ **参数:**

count:整数型

■ **返回格式:**

查询返回顺序模式在连续帧模式下的结束帧。

■ **举例:**

:SEGement:FRAME:END 15

设置顺序模式的结束帧 15

:SEGement:FRAME:END ?

查询返回 15

:SEGement:FRAME:CURRent■ **命令格式:**

:SEGement:FRAME:CURRent <count>

:SEGement:FRAME:CURRent ?

■ **功能描述:**

设置或查询顺序模式在单帧模式下的选定帧。

■ **参数:**

count:整数型

■ **返回格式:**

查询返回顺序模式在单帧模式下的选定帧。

■ **举例:**

:SEGement:FRAME:CURRent 1 设置顺序模式的选定帧 1

:SEGement:FRAME:CURRent ? 查询返回 1

:SEGement:FRAME:REF■ **命令格式:**

:SEGement:FRAME:REF <count>

:SEGement:FRAME:REF ?

■ **功能描述:**

设置或查询顺序模式在单帧模式下的参考帧。

■ **参数:**

count:整数型

■ **返回格式:**

查询返回顺序模式在单帧模式下的参考帧。

■ **举例:**

:SEGement:FRAME:REF 15 设置顺序模式的的参考帧 15

:SEGement:FRAME:REF ? 查询返回 15

:SEGement:REFactive■ **命令格式:**

:SEGement:REFactive <active>

:SEGement:REFactive ?

■ **功能描述:**

设置或查询顺序模式下参考帧开启状态。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:SEGement:REFactive ON

设置顺序模式参考帧打开

:SEGement:REFactive ?

查询返回 1

:SEGement:CALLback

■ **命令格式:**

:SEGement:CALLback <callback>

:SEGement:CALLback ?

■ **功能描述:**

设置或查询顺序模式在单帧模式下的回放状态。

■ **参数:**

callback:布尔型{ON | OFF}或{1 | 0}

ON:开始回放

OFF:停止回放

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:SEGement:CALLback ON

设置顺序模式回放

:SEGement:CALLback ?

查询返回 1

李沙育

:XY<n>:ACTive

■ **命令格式:**

:XY<n>:ACTive <active>

:XY<n>:ACTive ?

■ **功能描述:**

设置或查询指定 XY 是否打开。

■ **参数:**

n:整型, 范围 1~4

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:XY1:ACTive ON 设置 XY1 打开

:XY1:ACTive ? 查询返回 1

:XY<n>:SOURce:X

■ **命令格式:**

:XY<n>:SOURce:X <source>

:XY<n>:SOURce:X ?

■ **功能描述:**

设置或查询指定 XY 的源 X。

■ **参数:**

n:整型, 范围 1~4

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:XY1:SOURce:X C1 设置 XY1 的源 X 为 C1

:XY1:SOURce:X ? 查询返回 C1

:XY<n>:SOURce:Y

■ **命令格式:**

:XY<n>:SOURce:Y <source>

:XY<n>:SOURce:Y ?

■ **功能描述:**

设置或查询指定 XY 的源 Y。

■ **参数:**

n:整型, 范围 1~4

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回“C1”、“C2”、“C3”、“C4”。

■ **举例:**

:XY1:SOURce:Y C2 设置 XY1 的源 Y 为 C2

:XY1:SOURce:Y ? 查询返回 C2

:XY<n>:CURSor:ACTive

■ **命令格式:**

:XY<n>:CURSor:ACTive <active>

:XY<n>:CURSor:ACTive ?

■ **功能描述:**

设置或查询指定 XY 的光标是否打开。

■ **参数:**

n:整型, 范围 1~4

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:XY1:CURSor:ACTive ON 设置 XY1 的光标打开

:XY1:CURSor:ACTive ? 查询返回 1

:XY<n>:TYPe

■ **命令格式:**

:XY<n>:TYPe <type>

:XY<n>:TYPe ?

■ **功能描述:**

设置或查询指定 XY 的标记类型。

■ **参数:**

n:整型, 范围 1~4

type:离散型{CURSor | WAVE}

CURSor:光标

WAVE:波形

■ **返回格式:**

查询返回“CURSor”、“WAVe”。

■ **举例:**

:XY1:TYPE CURSor 设置 XY1 标记类型为光标

:XY1:TYPE ? 查询返回 CURSor

:XY<n>:DATA:TYPE

■ **命令格式:**

:XY<n>:DATA:TYPE <type>

:XY<n>:DATA:TYPE ?

■ **功能描述:**

设置或查询指定 XY 的数据类型。

■ **参数:**

n:整型, 范围 1~4

type:离散型{RECTangle | POLar}

RECTangle:矩形

POLar:极性

■ **返回格式:**

查询返回“RECTangle”、“POLar”。

■ **举例:**

:XY1:DATA:TYPE POLar 设置 XY1 数据类型为极性

:XY1:DATA:TYPE ? 查询返回 POLar

:XY<n>:CURSor:CAX

■ **命令格式:**

:XY<n>:CURSor:CAX <value>

:XY<n>:CURSor:CAX ?

■ **功能描述:**

设置或查询指定 XY 标记在光标模式下垂直光标 A 位置。

■ **参数:**

n:整型, 范围 1~4

value:布尔型, 以 div 为单位的实数,范围 0~10

■ **返回格式:**

查询返回指定 XY 标记在光标模式下垂直光标 A 位置。

■ **举例:**

:XY1:CURSOr:CAX 2

设置 XY1 在光标模式下垂直光标 A 的位置为 2div

:XY1:CURSOr:CAX ?

查询返回 2.00000E+000

:XY<n>:CURSor:CBX

■ **命令格式:**

:XY<n>:CURSor:CBX <value>

:XY<n>:CURSor:CBX ?

■ **功能描述:**

设置或查询指定 XY 标记在光标模式下垂直光标 B 位置。

■ **参数:**

n:整型, 范围 1~4

value:布尔型, 以 div 为单位的实数,范围 0~10

■ **返回格式:**

查询返回指定 XY 标记在光标模式下垂直光标 B 位置。

■ **举例:**

:XY1:CURSOr:CBX 2

设置 XY1 在光标模式下垂直光标 B 的位置为 2div

:XY1:CURSOr:CBX ?

查询返回 2.00000E+000

:XY<n>:CURSor:CAY

■ **命令格式:**

:XY<n>:CURSor:CAY <value>

:XY<n>:CURSor:CAY ?

■ **功能描述:**

设置或查询指定 XY 标记在光标模式下水平光标 A 位置。

■ **参数:**

n:整型, 范围 1~4

value:布尔型, 以 div 为单位的实数,范围-4~4

■ **返回格式:**

查询返回指定 XY 标记在光标模式下水平光标 A 位置。

■ **举例:**

:XY1:CURSOr:CAY 2

设置 XY1 在光标模式下水平光标 A 的位置为 2div

:XY1:CUSor:CAy ?

查询返回 2.00000E+000

:XY<n>:CURSor:CBY

■ 命令格式:

:XY<n>:CURSor:CBY <value>

:XY<n>:CURSor:CBY ?

■ 功能描述:

设置或查询指定 XY 标记在光标模式下水平光标 B 位置。

■ 参数:

n:整型, 范围 1~4

value:布尔型, 以 div 为单位的实数,范围-4~4

■ 返回格式:

查询返回指定 XY 标记在光标模式下水平光标 B 位置。

■ 举例:

:XY1:CUSor:CBY 2

设置 XY1 在光标模式下水平光标 B 的位置为 2div

:XY1:CUSor:CBY ?

查询返回 2.00000E+000

:XY<n>:CURSor:AXValue

■ 命令格式:

:XY<n>:CURSor:AXValue ?

■ 功能描述:

查询指定 XY 标记在光标模式下垂直光标 A 的矩形数据测量值, 单位与源 X 保持一致。

■ 参数:

n:整型, 范围 1~4

■ 返回格式:

查询返回指定 XY 标记在光标模式下垂直光标 A 的矩形数据测量值, 默认单位 V。

:XY<n>:CURSor:BXValue

■ 命令格式:

:XY<n>:CURSor:BXValue ?

■ 功能描述:

查询指定 XY 标记在光标模式下垂直光标 B 的矩形数据测量值, 单位与源 X 保持一致。

■ 参数:

n:整型, 范围 1~4

■ **返回格式:**

查询返回指定 X 标记在光标模式下垂直光标 B 的矩形数据测量值, 默认单位 V。

:XY<n>:CURSor:AYValue

■ **命令格式:**

:XY<n>:CURSor:AYValue ?

■ **功能描述:**

查询指定 XY 标记在光标模式下水平光标 A 的矩形数据测量值, 单位与源 Y 保持一致。

■ **参数:**

n:整型, 范围 1~4

■ **返回格式:**

查询返回指定 XY 标记在光标模式下水平光标 A 的矩形数据测量值, 默认单位 V。

:XY<n>:CURSor:BYValue

■ **命令格式:**

:XY<n>:CURSor:BYValue ?

■ **功能描述:**

查询指定 XY 标记在光标模式下水平光标 B 的矩形数据测量值, 单位与源 Y 保持一致。

■ **参数:**

n:整型, 范围 1~4

■ **返回格式:**

查询返回指定 XY 标记在光标模式下水平光标 B 的矩形数据测量值, 默认单位 V。

:XY<n>:WAVe:AXValue

■ **命令格式:**

:XY<n>:WAVe:AXValue ?

■ **功能描述:**

查询指定 XY 标记在波形模式下垂直光标 A 的矩形数据测量值, 单位与源 X 保持一致。

■ **参数:**

n:整型, 范围 1~4

■ **返回格式:**

查询返回指定 XY 标记在波形模式下垂直光标 A 的矩形数据测量值。

:XY<n>:WAVE:BXValue**■ 命令格式:**

:XY<n>:WAVE:BXValue ?

■ 功能描述:

查询指定 XY 标记在波形模式下垂直光标 B 的矩形数据测量值，单位与源 X 保持一致。

■ 参数:

n:整型，范围 1~4

■ 返回格式:

查询返回指定 XY 标记在波形模式下垂直光标 B 的矩形数据测量值。

:XY<n>:WAVE:AYValue**■ 命令格式:**

:XY<n>:WAVE:AYValue ?

■ 功能描述:

查询指定 XY 标记在波形模式下水平光标 A 的矩形数据测量值，单位与源 Y 保持一致。

■ 参数:

n:整型，范围 1~4

■ 返回格式:

查询返回指定 XY 标记在波形模式下水平光标 A 的矩形数据测量值。

:XY<n>:WAVE:BYValue**■ 命令格式:**

:XY<n>:WAVE:BYValue ?

■ 功能描述:

查询指定 XY 标记在波形模式下水平光标 B 的矩形数据测量值，单位与源 Y 保持一致。

■ 参数:

n:整型，范围 1~4

■ 返回格式:

查询返回指定 XY 标记在波形模式下水平光标 B 的矩形数据测量值。

:XY<n>:RADius:A**■ 命令格式:**

:XY<n>:RADius:A ?

■ **功能描述:**

查询指定 XY 的源 X 和源 Y 在光标 A 处电压值组成的极坐标系半径，单位与源 X 保持一致。

■ **参数:**

n:整型，范围 1~4

■ **返回格式:**

查询返回指定 XY 的源 X 和源 Y 在光标 A 处电压值组成的极坐标系半径。

:XY<n>:RADius:B

■ **命令格式:**

:XY<n>:RADius:B ?

■ **功能描述:**

查询指定 XY 的源 X 和源 Y 在光标 B 处电压值组成的极坐标系半径，单位与源 X 保持一致。

■ **参数:**

n:整型，范围 1~4

■ **返回格式:**

查询返回指定 XY 的源 X 和源 Y 在光标 B 处电压值组成的极坐标系半径。

:XY<n>:ANGLE:A

■ **命令格式:**

:XY<n>:ANGLE:A ?

■ **功能描述:**

查询指定 XY 的源 X 和源 Y 在光标 A 处电压值组成的极坐标系夹角，单位度。

■ **参数:**

n:整型，范围 1~4

■ **返回格式:**

查询返回指定 XY 的源 X 和源 Y 在光标 A 处电压值组成的极坐标系夹角。

:XY<n>:ANGLE:B

■ **命令格式:**

:XY<n>:ANGLE:B ?

■ **功能描述:**

查询指定 XY 的源 X 和源 Y 在光标 B 处电压值组成的极坐标系夹角，单位度。

■ 参数:

n:整型, 范围 1~4

■ 返回格式:

查询返回指定 XY 的源 X 和源 Y 在光标 B 处电压值组成的极坐标系夹角。

ZOOM

:ZOOM:ACTive

■ 命令格式:

:ZOOM:ACTive <active>

:ZOOM:ACTive ?

■ 功能描述:

设置或查询扩展是否打开。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回扩展状态, “1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:ZOOM:ACTive ON 设置打开扩展

:ZOOM:ACTive ? 查询返回 1

:ZOOM:CENTer:X

■ 命令格式:

:ZOOM:CENTer:X <value>

:ZOOM:CENTer:X ?

■ 功能描述:

设置或查询扩展中心点垂直位置。

■ 参数:

value:实数型, 范围 0-10000

■ 返回格式:

查询返回扩展中心点垂直位置。

■ 举例:

:ZOOM:CENTer:X 5000 设置打开扩展垂直中心点为 5000

:ZOOM:CENTer:X ?

查询返回 5.00000E+003

:ZOOM:CENTer:Y

■ 命令格式:

:ZOOM:CENTer:Y <value>

:ZOOM:CENTer:Y ?

■ 功能描述:

设置或查询扩展中心点水平位置。

■ 参数:

value:实数型, 范围-4000-4000

■ 返回格式:

查询返回扩展中心点水平位置。

■ 举例:

:ZOOM:CENTer:Y 0

设置打开扩展水平中心点为 0

:ZOOM:CENTer:Y ?

查询返回 0.00000E+000

:ZOOM:SCALe:X

■ 命令格式:

:ZOOM:SCALe:X <value>

:ZOOM:SCALe:X ?

■ 功能描述:

设置或查询扩展水平方向比例。

■ 参数:

value:实数型, 范围 1-1000

■ 返回格式:

查询返回扩展水平方向比例。

■ 举例:

:ZOOM:SCALe:X 10

设置扩展水平方向比例为 10

:ZOOM:SCALe:X ?

查询返回 1.00000E+001

:ZOOM:SCALe:Y

■ 命令格式:

:ZOOM:SCALe:Y <value>

:ZOOM:SCALE:Y ?

■ **功能描述:**

设置或查询扩展垂直方向比例。

■ **参数:**

value:实数型, 范围 1-1000

■ **返回格式:**

查询返回扩展垂直方向比例。

■ **举例:**

:ZOOM:SCALE:Y 10

设置打开扩展水平中心点为 10

:ZOOM:SCALE:Y ?

查询返回 1.00000E+001

抖动分析

:JITTer:ACTive

■ **命令格式:**

:JITTer:ACTive <active>

:JITTer:ACTive ?

■ **功能描述:**

设置或查询抖动分析是否打开。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回抖动分析打开状态, “1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:JITTer:ACTive ON

设置打开抖动分析

:JITTer:ACTive ?

查询返回 1

:JITTer:SOURce

■ **命令格式:**

:JITTer:SOURce <source>

:JITTer:SOURce ?

■ **功能描述:**

设置或查询抖动分析数据源。

■ 参数:

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回“C1”、“C2”、“C3”、“C4”。

■ 举例:

:JITTer:SOURce C1

设置抖动分析数据源为 C1

:JITTer:SOURce ?

查询返回 C1

:JITTer:SIGNal:TYPE**■ 命令格式:**

:JITTer:SIGNal:TYPE <type>

:JITTer:SIGNal:TYPE ?

■ 功能描述:

设置或查询抖动分析信号类型。

■ 参数:

type:离散型{CUSTom | CLOCK}

CUSTom:数据信号

CLOCK:时钟信号

■ 返回格式:

查询返回“CUSTom”、“CLOCK”。

■ 举例:

:JITTer:SIGNal:TYPE CLOCK

设置抖动分析信号类型为时钟信号

:JITTer:SIGNal:TYPE ?

查询返回 CLOCK

:JITTer:DTALength**■ 命令格式:**

:JITTer:DTALength <value>

:JITTer:DTALength ?

■ 功能描述:

设置或查询数据信号类型下的数据长度。

■ 参数:

value:整数型, 范围 0~4Gbit, 默认单位 bit

■ 返回格式:

查询返回数据信号类型下的数据长度。

■ **举例:**

:JITTer:DTALength 127 设置数据信号类型下的数据长度为 127bit

:JITTer:DTALength ? 查询返回 127

:JITTer:THReshold

■ **命令格式:**

:JITTer:THReshold <value>

:JITTer:THReshold ?

■ **功能描述:**

设置或查询抖动分析的比较阈值,阈值范围 45%~55%。

■ **参数:**

value:实数型

■ **返回格式:**

查询返回抖动分析的比较阈值。

■ **举例:**

:JITTer:THReshold 50 设置抖动分的比较阈值为 50%

:JITTer:THReshold ? 查询返回 5.00000E+001

:JITTer:HYSteresis

■ **命令格式:**

:JITTer:HYSteresis <value>

:JITTer:HYSteresis ?

■ **功能描述:**

设置或查询抖动分析的迟滞。

■ **参数:**

value:实数型

■ **返回格式:**

查询返回抖动分析的迟滞。

■ **举例:**

:JITTer:HYSteresis 30 设置抖动分的迟滞为 30%

:JITTer:HYSteresis ? 查询返回 3.00000E+001

:JITter:BITRate**■ 命令格式:**

:JITter:BITRate <bit>

:JITter:BITRate?

■ 功能描述:

设置或查询抖动分析的比特率。

■ 参数:

value:实数型, 范围 0~5Gbps, 默认单位 bps

■ 返回格式:

查询返回抖动分析的比特率。

■ 举例:

:JITter:BITRate 20 设置抖动分的比特率为 20

:JITter:BITRate ? 查询返回 2.00000E+001

:JITter:FNDBitrate**■ 命令格式:**

:JITter:FNDBitrate <active>

■ 功能描述:

设置寻找比特率。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 举例:

:JITter:FNDBitrate ON 设置寻找比特率

:JITter:CLOCK:TYPE**■ 命令格式:**

:JITter:CLOCK:TYPE <type>

:JITter:CLOCK:TYPE ?

■ 功能描述:

设置或查询抖动分析的时钟恢复方法。

■ 参数:

type:离散型{CONStant | PLL}

CONStant:常频时钟

PLL:锁相环

■ 返回格式:

查询返回“CONStant”、“PLL”。

■ 举例:

:JITTer:CLOCK:TYPe PLL

设置抖动分析的时钟恢复方法为 PLL

:JITTer:CLOCK:TYPe ?

查询返回 PLL

:JITTer:PLL:TYPe

■ 命令格式:

:JITTer:PLL:TYPe <type>

:JITTer:PLL:TYPe ?

■ 功能描述:

设置或查询抖动分析的 PLL 模式。

■ 参数:

type:离散型{GOLDeN | SEConDOrder}

GOLDeN:黄金分割频率锁相环

SEConDOrder:二阶锁相环

■ 返回格式:

查询返回“GOLDeN”、“SEConDOrder”。

■ 举例:

:JITTer:PLL:TYPe GOLDeN

设置抖动分析的 PLL 模式为 GOLDeN

:JITTer:PLL:TYPe ?

查询返回 GOLDeN

:JITTer:PLL:DIVisor:CUToff

■ 命令格式:

:JITTer:PLL:DIVisor:CUToff <value>

:JITTer:PLL:DIVisor:CUToff ?

■ 功能描述:

设置或查询抖动分析黄金分割频率锁相环的截止系数。

■ 参数:

value:实数型

■ 返回格式:

查询返回抖动分析黄金分割频率锁相环的截止系数。

■ 举例:

:JITTer:PLL:DIVisor:CUToff 100

设置抖动分析黄金分割频率锁相环的截止系数为 100

:JITTer:PLL:DIVisor:CUToff ?

查询返回 1.000000E+002

:JITTer:PLL:FREQ:CUToff**■ 命令格式:**

:JITTer:PLL:FREQ:CUToff <value>

:JITTer:PLL:FREQ:CUToff ?

■ 功能描述:

设置或查询抖动分析黄金分割频率锁相环的截止频率，默认单位 Hz。

■ 参数:

value:实数型，单位 mHz、Hz

■ 返回格式:

查询返回抖动分析黄金分割频率锁相环的截止频率。

■ 举例:

:JITTer:PLL:FREQ:CUToff 10

设置抖动分析黄金分割频率锁相环的截止频率为 10Hz

:JITTer:PLL:FREQ:CUToff ?

查询返回 1.000000000000000E+001

:JITTer:PLL:FREQ:NATural**■ 命令格式:**

:JITTer:PLL:FREQ:NATural <value>

:JITTer:PLL:FREQ:NATural ?

■ 功能描述:

设置或查询抖动分析二阶锁相环的自然频率，默认单位 Hz。

■ 参数:

value:实数型，单位 mHz、Hz

■ 返回格式:

查询返回抖动分析二阶锁相环的自然频率。

■ 举例:

:JITTer:PLL:FREQ:NATural 10

设置抖动分析二阶锁相环的自然频率为 10Hz

:JITTer:PLL:FREQ:NATural ?

查询返回 1.000000000000000E+001

:JITTer:PLL:FACTor**■ 命令格式:**

:JITTer:PLL:FACTor <value>

:JITTer:PLL:FACTor ?

■ 功能描述:

设置或查询抖动分析二阶锁相环的阻尼因子。

■ 参数:

value:实数型

■ 返回格式:

查询返回二阶锁相环的阻尼因子。

■ 举例:

:JITTer:PLL:FACTor 0.6

设置抖动分析二阶锁相环的阻尼因子为 0.6

:JITTer:PLL:FACTor ?

查询返回 6.00000E-001

:JITTer:PARams:JITTer**■ 命令格式:**

:JITTer:PARams:JITTer <active>

:JITTer:PARams:JITTer ?

■ 功能描述:

设置或查询抖动分析抖动参数是否打开。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回抖动分析抖动参数打开状态，“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:JITTer:PARams:JITTer ON

设置抖动分析抖动参数打开

:JITTer:PARams:JITTer ?

查询返回 1

:JITTer:PARams:EYE**■ 命令格式:**

:JITTer:PARams:EYE <active>

:JITTer:PARams:EYE ?

■ **功能描述:**

设置或查询抖动分析眼图参数是否打开。

■ **参数:**

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回抖动分析眼图参数打开状态,“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:JITTer:PARams:EYE ON

设置抖动分析眼图参数打开

:JITTer:PARams:EYE ?

查询返回 1

:JITTer:HISTogram:BINNum

■ **命令格式:**

:JITTer:HISTogram:BINNum <value>

:JITTer:HISTogram:BINNum ?

■ **功能描述:**

设置或查询抖动分析直方图柱体数。

■ **参数:**

value:离散型{25 | 50 | 100 | 250 | 500 | 2000 | MAX}

■ **返回格式:**

查询返回抖动分析直方图柱体数。

■ **举例:**

:JITTer:HISTogram:BINNum 25

设置抖动分析直方图柱体数 25

:JITTer:HISTogram:BINNum ?

查询返回 25

:JITTer:GRAPh

■ **命令格式:**

:JITTer:GRAPh <type> <enable>

■ **功能描述:**

设置抖动分析图形是否打开。

■ **参数:**

type:离散型{TRENd | SPECTrum | HISTogram | EYE | BATHtub}

TRENd:趋势图

SPECTrum:频谱图

HISTogram:直方图

EYE:眼图

BATHtub:浴盆曲线

enable:布尔型{ON | OFF}或{1 | 0}

■ 举例:

:JITTer:GRAPh EYE,ON 设置抖动分析眼图打开

:JITTer:DATA:JITTer

■ 命令格式:

:JITTer:DATA:JITTer ?

■ 功能描述:

查询抖动分析参数结果，并按照如下排列：

“TIE”、“TJ”、“RJ”、“DJ”、“PJ”、“DDJ”、“DCD”、“ISI”

TIE:时间间隔误差

TJ:特定误码率下的总体抖动

RJ:随机抖动

DJ:确定性抖动

PJ:周期性抖动

DDJ:数据相关性抖动

DCD:占空比失真

ISI:码间干扰

■ 返回格式:

查询返回抖动分析参数结果值，符合[数据块格式](#)，以科学计数法表示，并按顺序以逗号按分隔排列，无效值使用实型数据最大值表示。

例如：“#90000001001.20000E+000,2.00000E+002,1.20000E+003.....”

:JITTer:DATA:EYE

■ 命令格式:

:JITTer:DATA:EYE ?

■ 功能描述:

查询眼图参数结果，并按照如下排列：

“ZeroLevel”、“OneLevel”、“EyeAmplitude”、“EyeHeight”、“EyeWidth”、“ExtinctionRatio”、“QFactor”、“EyeCrossRatio”

ZeroLevel:0 电平

OneLevel:1 电平

EyeAmplitude:眼幅度

EyeHeight:眼高

EyeWidth:眼宽

ExtinctionRatio:消光比

QFactor:Q 因子

EyeCrossRatio:眼交叉比

■ 返回格式:

查询返回眼图参数结果值，符合[数据块格式](#)，以科学计数法表示，并按顺序以逗号按分隔排列，无效值使用实型数据最大值表示。

例如：“#90000001001.20000E+000,2.00000E+002,1.20000E+003.....”

PASS/FAIL

:PF:ACTive

■ 命令格式:

:PF:ACTive <active>

:PF:ACTive ?

■ 功能描述:

设置或查询通过失败测试是否打开。

■ 参数:

active:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回通过失败测试打开状态，“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:PF:ACTive ON 设置打开通过失败测试

:PF:ACTive ? 查询返回 1

:PF:VISible

■ 命令格式:

:PF:VISible <visible>

:PF:VISible ?

■ **功能描述:**

设置或查询通过失败测试是否显示测试模板。

■ **参数:**

visible:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回通过失败测试是否显示测试模板，“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:PF:VISible ON 设置显示通过失败测试显示模板

:PF:VISible ? 查询返回 1

:PF:MODE

■ **命令格式:**

:PF:MODE <mode>

:PF:MODE ?

■ **功能描述:**

设置或查询通过失败测试类型。

■ **参数:**

mode:离散型{LIMitMode | STDMaskmode}

LIMitMode:极限测试

STDMaskmode:标准测试

■ **返回格式:**

查询返回通过失败测试类型。

■ **举例:**

:PF:MODE LIMitMode 设置通过失败测试类型为极限测试

:PF:MODE ? 查询返回 LIMitMode

:PF:SOURce

■ **命令格式:**

:PF:SOURce <source>

:PF:SOURce ?

■ **功能描述:**

设置或查询通过失败测试数据源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4 }

■ **返回格式:**

查询返回通过失败测试数据源。

■ **举例:**

:PF:SOURce C1 设置通过失败测试数据源为 C1

:PF:SOURce ? 查询返回 C1

:PF:LIMit:SOURce

■ **命令格式:**

:PF:LIMit:SOURce <source>

:PF:LIMit:SOURce ?

■ **功能描述:**

设置或查询通过失败极限测试参考源。

■ **参数:**

source:离散型{C1 | C2 | C3 | C4 }

■ **返回格式:**

查询返回通过失败极限测试参考源。

■ **举例:**

:PF:LIMit:SOURce C1 设置通过失败极限测试参考源为 C1

:PF:LIMit:SOURce ? 查询返回 C1

:PF:LIMit:VTOLerance

■ **命令格式:**

:PF:LIMit:VTOLerance <value>

:PF:LIMit:VTOLerance?

■ **功能描述:**

设置或查询通过失败极限测试模板垂直容量，默认单位 mdiv。

■ **参数:**

value:整数型，范围 1mdiv~1div

■ **返回格式:**

查询返回通过失败极限测试模板垂直容量。

■ **举例:**

:PF:LIMit:VTOLerance 100 设置通过失败极限测试模板垂直容量 100mdiv

:PF:LIMit:VTOLerance ?

查询返回 100

:PF:LIMit:HTOLerance

■ 命令格式:

:PF:LIMit:HTOLerance <value>

:PF:LIMit:HTOLerance ?

■ 功能描述:

设置或查询通过失败极限测试模板水平容量，默认单位 mdiv。

■ 参数:

value:整数型，单位范围 1mdiv~500mdiv

■ 返回格式:

查询返回通过失败极限测试模板水平容量。

■ 举例:

:PF:LIMit:HTOLerance 100

设置通过失败极限测试模板水平容量 100mdiv

:PF:LIMit:HTOLerance ?

查询返回 100

:PF:LIMit:CREate

■ 命令格式:

:PF:LIMit:CREate

■ 功能描述:

创建通过失败极限测试模板。

:PF:STANdard:TYPe

■ 命令格式:

:PF:STANdard:TYPe <type>

:PF:STANdard:TYPe ?

■ 功能描述:

设置或查询通过失败标准测试类型。

■ 参数:

type:离散型{ANSI | ITU | USB}

■ 返回格式:

查询返回通过失败标准测试类型。

■ 举例:

:PF:STANdard:TYPE ANSI

设置通过失败标准测试类型 ANSI

:PF:STANdard:TYPE ?

查询返回 ANSI

:PF:STANdard:LOCK

■ 命令格式:

:PF:STANdard:LOCK <lock>

:PF:STANdard:LOCK ?

■ 功能描述:

设置或查询通过失败标准测试模板是否锁定。

■ 参数:

lock:布尔型{ON | OFF}或{1 | 0}

ON 或 1:锁定

OFF 或 0:未锁定

■ 返回格式:

查询返回通过失败标准测试模板锁定状态，“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:PF:STANdard:LOCK ON

设置通过失败标准测试模板锁定

:PF:STANdard:LOCK ?

查询返回 1

:PF:STANdard:INdEx

■ 命令格式:

:PF:STANdard:INdEx <index>

:PF:STANdard:INdEx ?

■ 功能描述:

设置或查询通过失败标准测试模板索引。

■ 参数:

index:整数型

■ 返回格式:

查询返回通过失败标准测试模板索引。

■ 举例:

:PF:STANdard:INdEx 10

设置通过失败标准测试模板索引为 10

:PF:STANdard:INdEx ?

查询返回 10

:PF:STOP:TYPe**■ 命令格式:**

:PF:STOP:TYPe <type>

:PF:STOP:TYPe ?

■ 功能描述:

设置或查询通过失败测试结束条件。

■ 参数:

type:离散型{WFMs | TIme}

WFMs:波形总数

TIme:总时间

■ 返回格式:

查询返回通过失败测试结束条件。

■ 举例:

:PF:STOP:TYPe WFMs

设置通过失败测试结束条件为 WFMs

:PF:STOP:TYPe ?

查询返回 WFMs

:PF:WFMS:COUNT**■ 命令格式:**

:PF:WFMS:COUNT <value>

:PF:WFMS:COUNT ?

■ 功能描述:

设置或查询通过失败测试结束条件为波形时的波形总数。

■ 参数:

value:整数型, 范围 1-100000

■ 返回格式:

查询返回通过失败测试结束条件为波形时的波形总数。

■ 举例:

:PF:WFMS:COUNT 1000

设置通过失败测试的波形总数为 1000

:PF:WFMS:COUNT ?

查询返回 1000

:PF:TIme:DURation**■ 命令格式:**

:PF:TIME:DURation <value>

:PF:TIME:DURation ?

■ **功能描述:**

设置或查询通过失败测试结束条件为时间时的总时间，默认单位 s。

■ **参数:**

value:实数型，范围 100ms-1Ms

■ **返回格式:**

查询返回通过失败测试结束条件为时间时的总时间。

■ **举例:**

:PF:TIME:DURation 100ms 设置通过失败测试的总时间为 100ms

:PF:TIME:DURation ? 查询返回 1.00000000000000E-001

:PF:VIOLations:COUNT

■ **命令格式:**

:PF:VIOLations:COUNT <value>

:PF:VIOLations:COUNT ?

■ **功能描述:**

设置或查询通过失败测试总违例次数。

■ **参数:**

value:整数型，范围 1-1000

■ **返回格式:**

查询返回通过失败测试总违例次数。

■ **举例:**

:PF:VIOLations:COUNT 10 设置通过失败测试的总违例次数为 10

:PF:VIOLations:COUNT ? 查询返回 10

:PF:OPTion:SAVe

■ **命令格式:**

:PF:OPTion:SAVe <save>

:PF:OPTion:SAVe ?

■ **功能描述:**

设置或查询通过失败测试失败操作是否保存。

■ **参数:**

save:布尔型{ON | OFF}或{1 | 0}

ON:保存

OFF:不保存

■ **返回格式:**

查询返回通过失败测试操作失败是否保存,“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:PF:OPTion:SAVe ON

设置通过失败测试失败保存

:PF:OPTion:SAVe ?

查询返回 1

:PF:OPTion:BEEP

■ **命令格式:**

:PF:OPTion:BEEP <beep>

:PF:OPTion:BEEP ?

■ **功能描述:**

设置或查询通过失败测试失败操作是否报警。

■ **参数:**

beep:布尔型{ON | OFF}或{1 | 0}

ON:报警

OFF:不报警

■ **返回格式:**

查询返回通过失败测试操作失败是否报警,“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:PF:OPTion:BEEP ON

设置通过失败测试失败报警

:PF:OPTion:BEEP ?

查询返回 1

:PF:OPTion:PULSe

■ **命令格式:**

:PF:OPTion:PULSe <pulse>

:PF:OPTion:PULSe ?

■ **功能描述:**

设置或查询通过失败测试失败操作是否输出脉冲。

■ **参数:**

pulse:布尔型{ON | OFF}或{1 | 0}

ON:输出脉冲

OFF:不打开脉冲

■ 返回格式:

查询返回通过失败测试失败操作是否输出脉冲,“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:PF:OPTion:PULSe ON

设置通过失败测试失败输出脉冲

:PF:OPTion:PULSe ?

查询返回 1

:PF:OPTion:HARDcopy

■ 命令格式:

:PF:OPTion:HARDcopy <hardcopy>

:PF:OPTion:HARDcopy ?

■ 功能描述:

设置或查询通过失败测试失败操作是否硬拷贝。

■ 参数:

hardcopy:布尔型{ON | OFF}或{1 | 0}

ON:硬拷贝

OFF:不硬拷贝

■ 返回格式:

查询返回通过失败测试失败操作是否硬拷贝,“1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:PF:OPTion:HARDcopy ON

设置通过失败测试失败打开硬拷贝

:PF:OPTion:HARDcopy ?

查询返回 1

:PF:RUN

■ 命令格式:

:PF:RUN <run>

:PF:RUN ?

■ 功能描述:

设置或查询通过失败测试运行状态。

■ 参数:

run:布尔型{ON | OFF}或{1 | 0}

ON:运行

OFF:停止

■ **返回格式:**

查询返回通过失败测试运行状态,“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:PF:RUN ON 设置通过失败测试运行

:PF:RUN ? 查询返回 1

:PF:DATA

■ **命令格式:**

:PF:DATA ?

■ **功能描述:**

查询通过失败测试状态表结果。

■ **返回格式:**

查询返回通过失败测试结果值,符合[数据块格式](#),并按顺序以逗号按分隔排列,无效值使用实型数据最大值表示。

例如:“#90000000147

Test Status,OFF,

Total Waveform,0,

Total Violations,0,

Testing Time,0 s,

Number of Hits,0,

Hits Per Segment:

1,0

2,0

3,0

4,0

5,0

6,0

7,0”

搜索

:SEARCh<n>:ACTive■ **命令格式:**

:SEARCh<n>:ACTive <active>

:SEARCh<n>:ACTive ?

■ **功能描述:**

设置或查询指定 search 是否打开。

■ **参数:**

n:整型, 范围 0~10

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回指定 search 的打开状态, “1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:SEARCh1:ACTive ON 设置打开 search1

:SEARCh1:ACTive ? 查询返回 1

:SEARCh<n>:SOURce■ **命令格式:**

:SEARCh<n>:SOURce <source>

:SEARCh<n>:SOURce ?

■ **功能描述:**

设置或查询指定 search 的数据源。

■ **参数:**

n:整型, 范围 0~10

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回指定 search 的数据源。

■ **举例:**

:SEARCh1:SOURce C1 设置 search1 的数据源为 C1

:SEARCh1:SOURce ? 查询返回 C1

:SEARCh<n>:TYPe**■ 命令格式:**

:SEARCh<n>:TYPe <type>

:SEARCh<n>:TYPe ?

■ 功能描述:

设置或查询指定 search 的搜索类型。

■ 参数:

n:整型, 范围 0~10

type:离散型{EDGE | PULSE}

EDGE:边沿

PULSE:脉宽

■ 返回格式:

查询返回指定 search 的搜索类型。

■ 举例:

:SEARCh1:TYPe EDGE 设置 search1 的搜索类型为 EDGE

:SEARCh1:TYPe ? 查询返回 EDGE

:SEARCh<n>:VISible**■ 命令格式:**

:SEARCh<n>:VISible <enable>

:SEARCh<n>:VISible ?

■ 功能描述:

设置或查询指定 search 的标记是否打开。

■ 参数:

n:整型, 范围 0~10

enable:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回指定 search 的标记打开状态, “1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:SEARCh1:VISible ON 设置 search1 的标记打开

:SEARCh1:VISible ? 查询返回 1

:SEARCh<n>:EVENT**■ 命令格式:**

:SEARCh<n>:EVENT <enable>

:SEARCh<n>:EVENT ?

■ 功能描述:

设置或查询指定 search 的事件列表是否打开。

■ 参数:

n:整型, 范围 0~10

enable:布尔型{ON | OFF}或{1 | 0}

■ 返回格式:

查询返回指定 search 的事件列表打开状态, “1”、“0”分别代表“ON”、“OFF”。

■ 举例:

:SEARCh1:EVENT ON 设置 search1 的事件列表打开

:SEARCh1:EVENT ? 查询返回 1

:SEARCh<n>:EDGE:SLOPe**■ 命令格式:**

:SEARCh<n>:EDGE:SLOPe <slope>

:SEARCh<n>:EDGE:SLOPe ?

■ 功能描述:

设置或查询指定 search 为边沿搜索时的边沿类型。

■ 参数:

n:整型, 范围 0~10

slope:离散型{RISe | FALL | ANY}

RISe:上升沿

FALL:下降沿

ANY:任意沿

■ 返回格式:

查询返回指定 search 为边沿搜索时的边沿类型。

■ 举例:

:SEARCh1:EDGE:SLOPe FALL 设置 search1 边沿搜索的边沿类型为下降沿

:SEARCh1:EDGE:SLOPe ? 查询返回 FALL

:SEARCh<n>:EDGE:LEVel**■ 命令格式:**

:SEARCh<n>:EDGE:LEVel <level>

:SEARCh<n>:EDGE:LEVel ?

■ 功能描述:

设置或查询指定 search 为边沿搜索时的搜索电平，默认单位 V。

■ 参数:

n:整型，范围 0~10

level:实数型，与通道垂直刻度单位一致，默认单位为 V

■ 返回格式:

查询返回指定 search 为边沿搜索时的搜索电平。

■ 举例:

:SEARCh1:EDGE:LEVel 100mV

设置 search1 边沿搜索的搜索电平为 100mV

:SEARCh1:EDGE:LEVel ?

查询返回 1.00000E-001

:SEARCh<n>:PULSe:POLarity**■ 命令格式:**

:SEARCh<n>:PULSe:POLarity <polarity>

:SEARCh<n>:PULSe:POLarity ?

■ 功能描述:

设置或查询指定 search 为脉宽搜索时的搜索极性。

■ 参数:

n:整型，范围 0~10

polarity:离散型{POSitive | NEGative}

POSitive:正

NEGative:负

■ 返回格式:

查询返回指定 search 为脉宽搜索时的搜索极性。

■ 举例:

:SEARCh1:PULSe:POLarity NEGative

设置 search1 脉宽搜索的搜索极性为 NEGative

:SEARCh1:PULSe:POLarity ?

查询返回 NEGative

:SEARCh<n>:PULSe:CONDition■ **命令格式:**

:SEARCh<n>:PULSe:CONDition <condition>

:SEARCh<n>:PULSe:CONDition ?

■ **功能描述:**

设置或查询指定 search 为脉宽搜索时的搜索条件。

■ **参数:**

n:整型, 范围 0~10

condition:离散型{GREATERthan | LESSthan | RANGE}

GREATERthan:大于

LESSthan:小于

RANGE:范围内

■ **返回格式:**

查询返回指定 search 为脉宽搜索时的搜索条件。

■ **举例:**

:SEARCh1:PULSe:CONDition LESSthan

设置 search1 脉宽搜索的搜索条件为 LESSthan

:SEARCh1:PULSe:CONDition ?

查询返回 LESSthan

:SEARCh<n>:PULSe:TIME:UPPer■ **命令格式:**

:SEARCh<n>:PULSe:TIME:UPPer <width>

:SEARCh<n>:PULSe:TIME:UPPer ?

■ **功能描述:**

设置或查询指定 search 为脉宽搜索时的脉冲宽度上限, 默认单位 s。

■ **参数:**

n:整型, 范围 0~10

width:实数型, 单位 ns、us、ms、s

■ **返回格式:**

查询返回指定 search 为脉宽搜索时的脉冲宽度上限。

■ **举例:**

:SEARCh1:PULSe:TIME:UPPer 2us

设置 search1 脉宽搜索的脉冲宽度上限为 2us

:SEARCh1:PULSe:TIME:UPPer ?

查询返回 2.00000000000000E-006

:SEARCh<n>:PULSe:TiMe:LOWer■ **命令格式:**

:SEARCh<n>:PULSe:TiMe:LOWer <width>

:SEARCh<n>:PULSe:TiMe:LOWer ?

■ **功能描述:**

设置或查询指定 search 为脉宽搜索时的脉冲宽度下限，默认单位 s。

■ **参数:**

n:整型，范围 0~10

width:实数型，单位 ns、us、ms、s

■ **返回格式:**

查询返回指定 search 为脉宽搜索时的脉冲宽度下限。

■ **举例:**

:SEARCh1:PULSe:TiMe:LOWer 1us 设置 search1 脉宽搜索的脉冲宽度下限为 1us

:SEARCh1:PULSe:TiMe:LOWer ? 查询返回 1.00000000000000E-006

:SEARCh<n>:PULSe:LEVel■ **命令格式:**

:SEARCh<n>:PULSe:LEVel <level>

:SEARCh<n>:PULSe:LEVel ?

■ **功能描述:**

设置或查询指定 search 为脉宽搜索时的阈值电平，默认单位 V。

■ **参数:**

n:整型，范围 0~10

level:实数型，与通道垂直刻度单位一致，默认单位为 V

■ **返回格式:**

查询返回指定 search 为脉宽搜索时的阈值电平。

■ **举例:**

:SEARCh1:PULSe:LEVel 100mV 设置 search1 脉宽搜索的阈值电平为 100mV

:SEARCh1:PULSe:LEVel ? 查询返回 1.00000E-001

:SEARCh<n>:DATA■ **命令格式:**

:SEARCh<n>:DATA ?

■ **功能描述:**

设查询指定 search 事件列表结果值。

■ **返回格式:**

查询返回指定 search 事件列表所有结果值，符合[数据块格式](#)，以科学计数法表示，并按顺序以逗号按分隔排列，无效值使用实型数据最大值表示。

例如：“#9000000549index,type,position,delta,description,
0,Edge,-2.40000000000000E-006,0.00000000000000E+000,Edge,
1,Edge,-2.30000000000000E-006,1.00000000000000E-007,Edge,
2,Edge,-2.20000000000000E-006,1.00000000000000E-007,Edge,”

电源分析

:PWAS:ADD

■ **命令格式:**

:PWAS:ADD <type>,<vsource>,<csource>

■ **功能描述:**

添加一个指定类型的电源分析。

■ **参数:**

type:{QUALity | HARMonic | RIPple | SWITchingloss | SAFeoperationarea | LOOPanalysis}

QUALity:电源质量

HARMonic:谐波分析

RIPple:纹波分析

SWITchingloss:开关损耗

SAFeoperationarea:安全工作区

LOOPanalysis:环路分析

vsource:电压源，离散型{C1 | C2 | C3 | C4}

csource:电流源，离散型{C1 | C2 | C3 | C4}

■ **举例:**

:PWAS:ADD QUALity,C1,C2 添加电源质量分析，电压源为 C1，电流源为 C2

:PWAS:ALLPoweranalysis

■ **命令格式:**

:PWAS:ALLPoweranalysis ?

■ **功能描述:**

查询所有已打开的电源分析。

■ **返回格式:**

查询返回所有已打开的电源分析, POWER1,电源分析类型, ..., POWER<n>,电源分析类型。

■ **举例:**

:PWAS:ALLPoweranalysis ?

查询返回 POWER1, QUALity, POWER2, RIPple

:PWAS:DELeTe

■ **命令格式:**

:PWAS:DELeTe <n>

■ **功能描述:**

删除一个指定电源分析。

■ **参数:**

n:整型, n 的值小于或等于已经打开的电源分析的个数, 范围 1~8

■ **举例:**

:PWAS:DELeTe 1

删除 POWER1

:POWeR<n>:MODe

■ **命令格式:**

:POWeR<n>:MODe ?

■ **功能描述:**

查询指定 POWER 的电源分析类型。

■ **参数:**

n:整型, 范围 1~8

■ **返回格式:**

查询返回指定 POWER 的电源分析类型。

QUALity:电源质量

HARMonic:谐波分析

RIPple:纹波分析

SWITchingloss:开关损耗

SAFeoperationarea:安全工作区

LOOPanalysis:环路分析

:POWer<n>:SOURce:VOLTage**■ 命令格式:**

:POWer<n>:SOURce:VOLTage <source>

:POWer<n>:SOURce:VOLTage ?

■ 功能描述:

设置或查询指定 POWER 的电压源。

■ 参数:

n:整型, 范围 1~8

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回指定 POWER 的电压源。

■ 举例:

:POWer1:SOURce:VOLTage C1

设置 POWER1 的电压源为 C1

:POWer1:SOURce:VOLTage ?

查询返回 C1

:POWer<n>:SOURce:CURREnt**■ 命令格式:**

:POWer<n>:SOURce:CURREnt <source>

:POWer<n>:SOURce:CURREnt ?

■ 功能描述:

设置或查询指定 POWER 的电流源。

■ 参数:

n:整型, 范围 1~8

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回指定 POWER 的电流源。

■ 举例:

:POWer1:SOURce:CURREnt C2

设置 POWER1 的电流源为 C2

:POWer1:SOURce:CURREnt ?

查询返回 C2

:POWer<n>:DATA**■ 命令格式:**

:POWer<n>:DATA ?

■ **功能描述:**

查询指定 POWER 的测量结果值。

■ **参数:**

n:整型, 范围 1~8

■ **返回格式:**

查询返回指定 POWER 的测量结果值, 符合[数据块格式](#), 以科学计数法表示, 并按顺序以逗号按分隔排列, 无效值使用实型数据最大值表示。

例如: “#9000000564QUALity,

Value,Average,Maxumum,Minimum,

1.78612E-003,1.99920E-003,2.47743E-003,1.53923E-003,

1.95003E+000,2.51101E+000,4.00265E+000,1.22652E+000,

1.79769E+308,1.79769E+308,1.79769E+308,1.79769E+308,

1.78612E-003,1.99920E-003,2.47743E-003,1.53923E-003,

1.95003E+000,2.51199E+000,4.00265E+000,1.22652E+000,

4.02983E-006,4.00127E-006,4.18383E-006,3.83827E-006,

3.19021E-006,4.01689E-006,6.13767E-006,2.36924E-006,

NaN,1.90541E-006,4.68634E-006,8.00995E-008,

1.26319E+000,1.01649E+000,1.65799E+000,6.45764E-001,

NaN,2.47287E+001,4.97770E+001,1.16113E+000,”

电源质量

:QUALity<n>:FREQuency:REFeRence

■ **命令格式:**

:QUALity<n>:FREQuency:REFeRence <type>

:QUALity<n>:FREQuency:REFeRence ?

■ **功能描述:**

设置或查询指定电源质量分析的频率参考。

■ **参数:**

n:整型, 范围 1~8

type:离散型{VOLTage | CURRent}

VOLTage:电压源

CURRent:电流源

■ **返回格式:**

查询返回指定电源质量分析的频率参考。

■ **举例:**

:QUALity1:FREQuency:REFerence CURRent

设置 QUALity1 的频率参考为电流源

:QUALity1:FREQuency:REFerence ?

查询返回 CURRent

谐波分析

:HARMonic<n>:COUNT

■ **命令格式:**

:HARMonic<n>:COUNT <value>

:HARMonic<n>:COUNT ?

■ **功能描述:**

设置或查询指定谐波分析的谐波个数。

■ **参数:**

value:整数型, 范围 40~200

■ **返回格式:**

查询返回指定谐波分析的谐波个数。

■ **举例:**

:HARMonic1:COUNT 50

设置 HARMonic1 的谐波个数为 50

:HARMonic1:COUNT ?

查询返回 50

:HARMonic<n>:FREQuency:REFerence

■ **命令格式:**

:HARMonic<n>:FREQuency:REFerence <type>

:HARMonic<n>:FREQuency:REFerence ?

■ **功能描述:**

设置或查询指定谐波分析的频率参考。

■ **参数:**

n:整型, 范围 1~8

type:离散型{HARMonicsrc | VOLTage | CURRent | CONStant}

HARMonicsrc:谐波源

VOLTage:电压源

CURRent:电流源

CONStant:固定

返回格式:

查询返回指定谐波分析的频率参考。

■ 举例:

:HARMonic1:FREQuency:REFeRence CURRent

设置 HARMonic1 的频率参考为电流源

:HARMonic1:FREQuency:REFeRence ?

查询返回 CURRent

:HARMonic<n>:FREQuency:CONStant

■ 命令格式:

:HARMonic<n>:FREQuency:CONStant <value>

:HARMonic<n>:FREQuency:CONStant ?

■ 功能描述:

设置或查询指定谐波分析的频率参考为固定时的频率，默认单位 Hz。

■ 参数:

n:整型，范围 1~8

value:实数型，单位 Hz、KHz、MHz、GHz。

■ 返回格式:

查询返回指定谐波分析的频率参考为固定时的频率。

■ 举例:

:HARMonic1:FREQuency:CONStant 120

设置 HARMonic1 的固定频率参考为 120Hz

:HARMonic1:FREQuency:CONStant ?

查询返回 1.20000000000000E+002

:HARMonic<n>:SOURce

■ 命令格式:

:HARMonic<n>:SOURce <source>

:HARMonic<n>:SOURce ?

■ 功能描述:

设置或查询指定谐波分析的谐波源。

■ 参数:

n:整型，范围 1~8

source:离散型{VOLTage | CURRent}

VOLTage:电压

CURRent:电流

■ 返回格式:

查询返回指定谐波分析的谐波源。

■ 举例:

:HARMonic1:SOURce CURRent

设置 HARMonic1 的谐波源为电流

:HARMonic1:SOURce ?

查询返回 CURRent

纹波分析

:RIPPlE<n>:SOURce

■ 命令格式:

:RIPPlE<n>:SOURce <source>

:RIPPlE<n>:SOURce ?

■ 功能描述:

设置或查询指定纹波分析的纹波源。

■ 参数:

n:整型, 范围 1~8

source:离散型{VOLTage | CURRent}

VOLTage:电压源

CURRent:电流源

■ 返回格式:

查询返回指定纹波分析的纹波源。

■ 举例:

:RIPPlE1:SOURce CURRent

设置 HARMonic1 的纹波源为电流源

:RIPPlE1:SOURce ?

查询返回 CURRent

安全工作区

:SAFeoperationarea<n>:STOPonfail

■ 命令格式:

:SAFeoperationarea<n>:STOPonfail <active>

:SAFeoperationarea<n>:STOPonfail ?

■ **功能描述:**

设置或查询指定安全工作区违例时是否停止。

■ **参数:**

n:整型, 范围 1~8

active:布尔型{ON | OFF}或{1 | 0}

ON:停止

OFF:不停止

■ **返回格式:**

查询返回指定安全工作区违例时是否停止, “1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:SAFeoperationarea1:STOPonfail ON

设置 SAFeoperationarea1 违例时停止

:SAFeoperationarea1:STOPonfail ?

查询返回 1

:SAFeoperationarea<n>:AXIS:XMAX

■ **命令格式:**

:SAFeoperationarea<n>:AXIS:XMAX <value>

:SAFeoperationarea<n>:AXIS:XMAX ?

■ **功能描述:**

设置或查询指定安全工作区 X 轴最大值, 默认单位 V。

■ **参数:**

n:整型, 范围 1~8

value:实数型, 范围-3.99V~10kV

■ **返回格式:**

查询返回指定安全工作区 X 轴最大值。

■ **举例:**

:SAFeoperationarea1:AXIS:XMAX 5

设置 SAFeoperationarea1 的 X 轴最大值为 5V

:SAFeoperationarea1:AXIS:XMAX ?

查询返回 5.00000E+000

:SAFeoperationarea<n>:AXIS:XMIN

■ **命令格式:**

:SAFeoperationarea<n>:AXIS:XMIN <value>

:SAFeoperationarea<n>:AXIS:XMIN ?

■ **功能描述:**

设置或查询指定安全工作区 X 轴最小值，默认单位 V。

■ **参数:**

n:整型，范围 1~8

value:实数型，范围-10kV~7.99V

■ **返回格式:**

查询返回指定安全工作区 X 轴最小值。

■ **举例:**

:SAFeoperationarea1:AXIS:XMIN 1

设置 SAFeoperationarea1 的 X 轴最小值为 1V

:SAFeoperationarea1:AXIS:XMIN ?

查询返回 1.00000E+000

:SAFeoperationarea<n>:AXIS:YMAX

■ **命令格式:**

:SAFeoperationarea<n>:AXIS:YMAX <value>

:SAFeoperationarea<n>:AXIS:YMAX ?

■ **功能描述:**

设置或查询指定安全工作区 Y 轴最大值，默认单位 A。

■ **参数:**

n:整型，范围 1~8

value:实数型，范围 10mA~10kA

■ **返回格式:**

查询返回指定安全工作区 Y 轴最大值。

■ **举例:**

:SAFeoperationarea1:AXIS:YMAX 5

设置 SAFeoperationarea1 的 Y 轴最大值为 5A

:SAFeoperationarea1:AXIS:YMAX ?

查询返回 5.00000E+000

:SAFeoperationarea<n>:AXIS:YMIN

■ **命令格式:**

:SAFeoperationarea<n>:AXIS:YMIN <value>

:SAFeoperationarea<n>:AXIS:YMIN ?

■ **功能描述:**

设置或查询指定安全工作区 Y 轴最小值，默认单位 A。

■ **参数:**

n:整型, 范围 1~8

value:实数型, 范围-10kA~5.29A

■ 返回格式:

查询返回指定安全工作区 Y 轴最小值。

■ 举例:

:SAFeoperationarea1:AXIS:YMIN 1

设置 SAFeoperationarea1 的 Y 轴最小值为 1A

:SAFeoperationarea1:AXIS:YMIN ?

查询返回 1.00000E+000

环路分析

:LOOPAnalysis<n>:SOURce:AWG

■ 命令格式:

:LOOPAnalysis<n>:SOURce:AWG <source>

:LOOPAnalysis<n>:SOURce:AWG ?

■ 功能描述:

设置或查询指定环路分析的 AWG 源。

■ 参数:

n:整型, 范围 1~8

source:离散型{G1 | G2}

■ 返回格式:

查询返回指定环路分析的 AWG 源。

■ 举例:

:LOOPAnalysis1:SOURce:AWG G1

设置 LOOPAnalysis1 的 AWG 源为 G1

:LOOPAnalysis1:SOURce:AWG ?

查询返回 G1

:LOOPAnalysis<n>:IMPedance

■ 命令格式:

:LOOPAnalysis<n>:IMPedance <type>

:LOOPAnalysis<n>:IMPedance ?

■ 功能描述:

设置或查询指定环路分析的阻抗。

■ 参数:

n:整型, 范围 1~8

type:离散型{LOW | HIGH}

LOW:50Ω

HIGH:高阻

■ 返回格式:

查询返回指定环路分析的阻抗。

■ 举例:

:LOOPAnalysis1:IMPedance HIGH

设置 LOOPAnalysis1 阻抗为高阻

:LOOPAnalysis1:IMPedance ?

查询返回 HIGH

:LOOPAnalysis<n>:FREQUENCY:START

■ 命令格式:

:LOOPAnalysis<n>:FREQUENCY:START <value>

:LOOPAnalysis<n>:FREQUENCY:START ?

■ 功能描述:

设置或查询指定环路分析的开始频率，默认单位 Hz。

■ 参数:

n:整型，范围 1~8

value:实数型，单位 Hz、kHz、MHz

■ 返回格式:

查询返回指定环路分析的开始频率。

■ 举例:

:LOOPAnalysis1:FREQUENCY:START 100

设置 LOOPAnalysis1 的开始频率为 100Hz

:LOOPAnalysis1:FREQUENCY:START ?

查询返回 1.00000000000000E+002

:LOOPAnalysis<n>:FREQUENCY:STOP

■ 命令格式:

:LOOPAnalysis<n>:FREQUENCY:STOP <value>

:LOOPAnalysis<n>:FREQUENCY:STOP ?

■ 功能描述:

设置或查询指定环路分析的停止频率，默认单位 Hz。

■ 参数:

n:整型，范围 1~8

value:实数型，单位 Hz、kHz、MHz

■ **返回格式:**

查询返回指定环路分析的停止频率。

■ **举例:**

:LOOPAnalysis1:FREQUENCY:STOP 1MHz

设置 LOOPAnalysis1 的停止频率为 1MHz

:LOOPAnalysis1:FREQUENCY:STOP ?

查询返回 1.00000000000000E+006

:LOOPAnalysis<n>:SCANnum

■ **命令格式:**

:LOOPAnalysis<n>:SCANnum <value>

:LOOPAnalysis<n>:SCANnum ?

■ **功能描述:**

设置或查询指定环路分析的扫描点数。

■ **参数:**

n:整型, 范围 1~8

value:整数型, 范围 1~1k

■ **返回格式:**

查询返回指定环路分析的扫描点数。

■ **举例:**

:LOOPAnalysis1:SCANnum 30

设置 LOOPAnalysis1 的扫描点数为 30

:LOOPAnalysis1:SCANnum ?

查询返回 30

:LOOPAnalysis<n>:AMPMode

■ **命令格式:**

:LOOPAnalysis<n>:AMPMode <mode>

:LOOPAnalysis<n>:AMPMode ?

■ **功能描述:**

设置或查询指定环路分析的幅度模式。

■ **参数:**

n:整型, 范围 1~8

mode:离散型{CONSTant | VARIABLE}

CONSTant:恒定

VARIABLE:可变

■ **返回格式:**

查询返回指定环路分析的幅度模式。

■ **举例:**

:LOOPanalysis1:AMPMode CONSTant

设置 LOOPanalysis1 的幅度模式为恒定

:LOOPanalysis1:AMPMode ?

查询返回 CONSTant

:LOOPanalysis<n>:AMPValue

■ **命令格式:**

:LOOPanalysis<n>:AMPValue <value>

:LOOPanalysis<n>:AMPValue ?

■ **功能描述:**

设置或查询指定环路分析的幅度，默认单位 V。

■ **参数:**

n:整型，范围 1~8

value:实数型，单位 mV、V

■ **返回格式:**

查询返回指定环路分析的幅度。

■ **举例:**

:LOOPanalysis1:AMPValue 1

设置 LOOPanalysis1 的幅度为 1V

:LOOPanalysis1:AMPValue ?

查询返回 1.00000E+000

注：MSO8804HD 型号无电源分析功能。

参考通道

:REFerence<n>:ACTive

■ **命令格式:**

:REFerence<n>:ACTive <active>, path

:REFerence<n>:ACTive ?

■ **功能描述:**

设置或查询指定参考通道是否打开。

■ **参数:**

n:整型，范围 0~4

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回指定参考通道的打开状态，“1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:REference1:ACTive ON,“C://Users//Administrator//Desktop//Uni-t000.bin”

设置打开 REference1

:REference1:ACTive ?

查询返回 1

:REference<n>:LABel

■ **命令格式:**

:REference<n>:LABel <labe>

:REference<n>:LABel ?

■ **功能描述:**

设置或查询指定参考通道的标签。

■ **参数:**

n:整型, 范围 0~4

labe:字符串

■ **返回格式:**

查询返回指定参考通道的标签。

■ **举例:**

:REference1:LABel “QWER”

设置 REF1 标签为 QWER

:REference1:LABel ?

查询返回 QWER

:REference<n>:VERTical:OFFSet

■ **命令格式:**

:REference<n>:VERTical:OFFSet <offset>

:REference<n>:VERTical:OFFSet ?

■ **功能描述:**

设置或查询指定参考通道的垂直位移。

■ **参数:**

n:整型, 范围 0~4

offset:实数型, 单位为 div, 范围为-4div-4div

■ **返回格式:**

查询返回指定参考通道的垂直位移。

■ **举例:**

:REfERENCE1:VERTical:OFFSet 1.2

设置 REF1 垂直位移为 1.2

:REfERENCE1:VERTical:OFFSet ?

查询返回 1.20000E+000

:REfERENCE<n>:VERTical:SCALE

■ 命令格式:

:REfERENCE<n>:VERTical:SCALE <scale>

:REfERENCE<n>:VERTical:SCALE ?

■ 功能描述:

设置或查询指定参考通道的垂直档位。

■ 参数:

n:整型, 范围 0~4

scale:实数型, 单位为保存波形时的单位

■ 返回格式:

查询返回指定参考通道的垂直档位。

■ 举例:

:REfERENCE1:VERTical:SCALE 1

设置 REF1 垂直档位为 1

:REfERENCE1:VERTical:SCALE ?

查询返回 1.00000E+000

:REfERENCE<n>:HORizontal:OFFSet

■ 命令格式:

:REfERENCE<n>:HORizontal:OFFSet <offset>

:REfERENCE<n>:HORizontal:OFFSet ?

■ 功能描述:

设置或查询指定参考通道的水平位移。

■ 参数:

n:整型, 范围 0~4

offset:实数型, 单位为 div, 范围为 0div-10div

■ 返回格式:

查询返回指定参考通道的水平位移。

■ 举例:

:REfERENCE1:HORizontal:OFFSet 1.2

设置 REF1 水平位移为 1.2

:REfERENCE1:HORizontal:OFFSet ?

查询返回 1.20000E+000

:REfERENCE<n>:HORizontal:SCALe■ **命令格式:**

:REfERENCE<n>:HORizontal:SCALe <scale>

:REfERENCE<n>:HORizontal:SCALe ?

■ **功能描述:**

设置或查询指定参考通道的水平档位。

■ **参数:**

n:整型, 范围 0~4

scale:实数型, 单位:us、ms、s

■ **返回格式:**

查询返回指定参考通道的水平档位。

■ **举例:**

:REfERENCE1:HORizontal:SCALe 1

设置 REF1 水平档位为 1

:REfERENCE1:HORizontal:SCALe ?

查询返回 1.00000000000000E+000

总线**:BUS<n>:ACTive**■ **命令格式:**

:BUS<n>:ACTive <active>

:BUS<n>:ACTive ?

■ **功能描述:**

设置或查询 BUS 是否打开。

■ **参数:**

n:整型, 范围 1~2

active:布尔型{ON | OFF}或{1 | 0}

■ **返回格式:**

查询返回指定 BUS 的打开状态, “1”、“0”分别代表“ON”、“OFF”。

■ **举例:**

:BUS1:ACTive 1 设置 BUS1 打开

:BUS1:ACTive ? 查询返回 1

:BUS<n>:TYPe**■ 命令格式:**

:BUS<n>:TYPe <type>

:BUS<n>:TYPe ?

■ 功能描述:

设置或查询指定 BUS 的解码类型。

■ 参数:

n:整型, 范围 1~2

type:离散型{Close | RS232 | I2C | SPI | CAN | CANFd (CAN-FD 总线) | LIN | FLEXray |
AUDioBus | MIL (MIL-STD-1553 总线) | ARINc429 | SENT}

■ 返回格式:

查询返回指定 BUS 的解码类型。

■ 举例:

:BUS1:TYPe RS232

设置 BUS1 的解码类型为 RS232

:BUS1:TYPe ?

查询返回 RS232

:BUS<n>:FORMat**■ 命令格式:**

:BUS<n>:FORMat <type>

:BUS<n>:FORMat ?

■ 功能描述:

设置或查询指定 BUS 的解码显示格式。

■ 参数:

n:整型, 范围 1~2

type:离散型{HEX | DEC | BINary | ASCII | AUTo}

■ 返回格式:

查询返回指定 BUS 的解码显示格式。

■ 举例:

:BUS1:FORMat HEX

设置 BUS1 的解码显示格式为 HEX

:BUS1:FORMat ?

查询返回 HEX

:BUS<n>:POSition**■ 命令格式:**

:BUS<n>:POSition <value>

:BUS<n>:POSition ?

■ 功能描述:

设置或查询指定 BUS 的解码显示位置，默认单位 div。

■ 参数:

n:整型，范围 1~2

value:实数型，-4div ~ 4div

■ 返回格式:

查询返回指定 BUS 的解码显示位置。

■ 举例:

:BUS1:POSition 2

设置 BUS1 的解码显示位置为 2div

:BUS1:POSition ?

查询返回 2.00000E+000

:BUS<n>:LABel**■ 命令格式:**

:BUS<n>:LABel <lable>

:BUS<n>:LABel ?

■ 功能描述:

设置或查询指定 BUS 的标签。

■ 参数:

n:整型，范围 1~2

lable:离散型

■ 返回格式:

查询返回指定 BUS 的标签。

■ 举例:

:BUS1:LABel abcd

设置 BUS1 的标签为 abcd

:BUS1:LABel ?

查询返回 abcd

:BUS<n>:DATA**■ 命令格式:**

:BUS<n>:DATA ?

■ **功能描述:**

查询指定 BUS 的事件列表结果。

■ **参数:**

n:整型, 范围 1~2

■ **返回格式:**

查询返回指定 BUS 事件列表所有结果值, 符合[数据块格式](#), 以科学计数法表示, 并按顺序以逗号按分隔排列, 无效值使用实型数据最大值表示。以 CAN 类型举例的返回格式如下:

“Index,Start Time,StandardID,ExpandID,DLC,Data,CRC,ACK,EOF,ERROR”

例如: “#9000000272CAN,

Index,Start Time,StandardID,ExpandID,DLC,Data,CRC,ACK,EOF,ERROR

1,-1.89960E-004,NaN,00 02 73 E6,0F,NaN,NaN,NaN,NaN,NaN

2,2.10040E-004,NaN,NaN,NaN,NaN,NaN,NaN,NaN,NaN

3,3.10040E-004,01 8F,NaN,00,NaN,30 67,01,03,CRC

4,5.60040E-004,NaN,NaN,NaN,NaN,NaN,NaN,NaN,NaN”

RS232

:BUS<n>:RS232:SOURce

■ **命令格式:**

:BUS<n>:RS232:SOURce <source>

:BUS<n>:RS232:SOURce ?

■ **功能描述:**

设置或查询 RS232 的解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 RS232 的解码信源。

■ **举例:**

:BUS1:RS232:SOURce C1

设置 BUS1 的 RS232 源为 C1

:BUS1:RS232:SOURce ?

查询返回 C1

:BUS<n>:RS232:BAUDrate:CURRent**■ 命令格式:**

:BUS<n>:RS232:BAUDrate:CURRent <value>

:BUS<n>:RS232:BAUDrate ?

■ 功能描述:

设置或查询 RS232 的解码波特率，默认单位 bps。

■ 参数:

n:整型，范围 1~2

value:离散型{CUSTom | 2400 | 4800 | 9600 | 192000 | 38400 | 57600 | 115200}

■ 返回格式:

查询返回 RS232 的解码波特率。

■ 举例:

:BUS1:RS232:BAUDrate:CURRent 115200 设置 BUS1 的 RS232 波特率为 115200bps

:BUS1:RS232:BAUDrate:CURRent ? 查询返回 115200

:BUS<n>:RS232:BAUdrate:CUSTom**■ 命令格式:**

:BUS<n>:RS232:BAUdrate:CUSTom <value>

:BUS<n>:RS232:BAUdrate:CUSTom ?

■ 功能描述:

设置或查询 RS232 的解码自定义波特率，默认单位 bps。

■ 参数:

n:整型，范围 1~2

value:整数型，范围 1kbps~1Gbps

■ 返回格式:

查询返回 RS232 的解码自定义波特率。

■ 举例:

:BUS1:RS232:BAUdrate:CUSTom 5500 设置 BUS1 的 RS232 自定义波特率为 5500bps

:BUS1:RS232:BAUdrate:CUSTom ? 查询返回 5500

:BUS<n>:RS232:LEVel**■ 命令格式:**

:BUS<n>:RS232:LEVel <value>

:BUS<n>:RS232:LEVel ?

■ **功能描述:**

设置或查询 RS232 的解码阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

value:实数型，单位与通道垂直刻度单位一致，默认单位为 V

■ **返回格式:**

查询返回 RS232 的解码阈值。

■ **举例:**

:BUS1:RS232:LEVel 1

设置 BUS1 的 RS232 解码阈值为 1V

:BUS1:RS232:LEVel ?

查询返回 1.00000E+000

:BUS<n>:RS232:ORDer

■ **命令格式:**

:BUS<n>:RS232:ORDer <order>

:BUS<n>:RS232:ORDer ?

■ **功能描述:**

设置或查询 RS232 的解码位顺序。

■ **参数:**

n:整型，范围 1~2

order:离散型{MSB | LSB}

■ **返回格式:**

查询返回 RS232 的解码位顺序。

■ **举例:**

:BUS1:RS232:ORDer LSB

设置 BUS1 的 RS232 解码位顺序为 LSB

:BUS1:RS232:ORDer ?

查询返回 LSB

:BUS<n>:RS232:WIDTh

■ **命令格式:**

:BUS<n>:RS232:WIDTh <width>

:BUS<n>:RS232:WIDTh ?

■ **功能描述:**

设置或查询 RS232 的解码位宽。

■ **参数:**

n:整型, 范围 1~2

width:离散型{5 | 6 | 7 | 8}

■ **返回格式:**

查询返回 RS232 的解码位宽。

■ **举例:**

:BUS1:RS232:WIDTH 8

设置 BUS1 的 RS232 解码位宽为 8

:BUS1:RS232:WIDTH ?

查询返回 8

:BUS<n>:RS232:STOP

■ **命令格式:**

:BUS<n>:RS232:STOP <stop>

:BUS<n>:RS232:STOP ?

■ **功能描述:**

设置或查询 RS232 的解码停止位。

■ **参数:**

n:整型, 范围 1~2

width:离散型{1 | 2}

■ **返回格式:**

查询返回 RS232 的解码停止位。

■ **举例:**

:BUS1:RS232:STOP 1

设置 BUS1 的 RS232 解码停止位为 1

:BUS1:RS232:STOP ?

查询返回 1

:BUS<n>:RS232:PARity

■ **命令格式:**

:BUS<n>:RS232:PARity <type>

:BUS<n>:RS232:PARity ?

■ **功能描述:**

设置或查询 RS232 的解码奇偶校验位。

■ **参数:**

n:整型, 范围 1~2

type:离散型{NONE | ODD | EVEN}

NONE:无校验

ODD:奇校验

EVEN:偶校验

■ 返回格式:

查询返回 RS232 的解码奇偶校验位。

■ 举例:

:BUS1:RS232:PARity ODD

设置 BUS1 的 RS232 解码奇偶校验位为 ODD

:BUS1:RS232:PARity ?

查询返回 ODD

:BUS<n>:RS232:POLarity

■ 命令格式:

:BUS<n>:RS232:POLarity <type>

:BUS<n>:RS232:POLarity ?

■ 功能描述:

设置或查询 RS232 的解码极性。

■ 参数:

n:整型, 范围 1~2

type:离散型{NEGative | POSitive}

NEGative:负极性

POSitive:正极性

■ 返回格式:

查询返回 RS232 的解码极性。

■ 举例:

:BUS1:RS232:POLarity POSitive

设置 BUS1 的 RS232 解码极性为 POSitive

:BUS1:RS232:POLarity ?

查询返回 POSitive

I2C

:BUS<n>:I2C:SDA

■ 命令格式:

:BUS<n>:I2C:SDA <source>

:BUS<n>:I2C:SDA ?

■ **功能描述:**

设置或查询 I2C 解码的数据源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 I2C 解码的数据源。

■ **举例:**

:BUS1:I2C:SDA C1	设置 BUS1 的 I2C 解码数据源 C1
:BUS1:I2C:SDA ?	查询返回 C1

:BUS<n>:I2C:SCL

■ **命令格式:**

:BUS<n>:I2C:SCL <source>

:BUS<n>:I2C:SCL ?

■ **功能描述:**

设置或查询 I2C 解码的时钟信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 I2C 解码的时钟信源。

■ **举例:**

:BUS1:I2C:SCL C2	设置 BUS1 的 I2C 解码时钟源 C2
:BUS1:I2C:SCL ?	查询返回 C2

:BUS<n>:I2C:DLEVel

■ **命令格式:**

:BUS<n>:I2C:DLEVel <level>

:BUS<n>:I2C:DLEVel ?

■ **功能描述:**

设置或查询 I2C 解码的数据阈值, 默认单位 V。

■ **参数:**

n:整型, 范围 1~2

level:实数型, 单位与通道垂直刻度单位一致, 默认单位为 V, 范围-8V~8V

■ **返回格式:**

查询返回 I2C 解码的数据阈值。

■ **举例:**

:BUS1:I2C:DLEVel 1 设置 BUS1 的 I2C 解码数据阈值为 1V

:BUS1:I2C:DLEVel ? 查询返回 1.00000E+000

:BUS<n>:I2C:CLEVel

■ **命令格式:**

:BUS<n>:I2C:CLEVel <level>

:BUS<n>:I2C:CLEVel ?

■ **功能描述:**

设置或查询 I2C 解码的时钟阈值, 默认单位 V。

■ **参数:**

n:整型, 范围 1~2

level:实数型, 单位与通道垂直刻度单位一致, 默认单位为 V, 范围-8V~8V

■ **返回格式:**

查询返回 I2C 解码的时钟阈值。

■ **举例:**

:BUS1:I2C:CLEVel 1 设置 BUS1 的 I2C 解码时钟阈值为 1V

:BUS1:I2C:CLEVel ? 查询返回 1.00000E+000

:BUS<n>:I2C:WIDTh

■ **命令格式:**

:BUS<n>:I2C:WIDTh <width>

:BUS<n>:I2C:WIDTh ?

■ **功能描述:**

设置或查询 I2C 解码的位宽。

■ **参数:**

n:整型, 范围 1~2

width:离散型{7 | 10}

■ **返回格式:**

查询返回 I2C 解码的位宽。

■ **举例:**

:BUS1:I2C:WIDTH 7

设置 BUS1 的 I2C 解码的位宽为 7bit

:BUS1:I2C:WIDTH ?

查询返回 7

SPI

:BUS<n>:SPI:DECHannel

■ **命令格式:**

:BUS<n>:SPI:DECHannel <source>

:BUS<n>:SPI:DECHannel ?

■ **功能描述:**

设置或查询 SPI 解码通道。

■ **参数:**

n:整型, 范围 1~2

source:离散型{MISO | MOSI | MOMI}

■ **返回格式:**

查询返回 SPI 解码通道。

■ **举例:**

:BUS1:SPI:DECHannel MISO

设置 BUS1 的 SPI 解码通道为 MISO

:BUS1:SPI:DECHannel ?

查询返回 MISO

:BUS<n>:SPI:SOURce:SCL

■ **命令格式:**

:BUS<n>:SPI:SOURce:SCL <source>

:BUS<n>:SPI:SOURce:SCL ?

■ **功能描述:**

设置或查询 SPI 解码的时钟源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 SPI 解码的时钟源。

■ **举例:**

:BUS1:SPI:SOURce:SCL C1

设置 BUS1 的 SPI 解码时钟源 C1

:BUS1:SPI:SOURce:SCL ?

查询返回 C1

:BUS<n>:SPI:SOURce:CS

■ **命令格式:**

:BUS<n>:SPI:SOURce:CS <source>

:BUS<n>:SPI:SOURce:CS ?

■ **功能描述:**

设置或查询 SPI 解码的片选源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 SPI 解码的片选源。

■ **举例:**

:BUS1:SPI:SOURce:CS C2

设置 BUS1 的 SPI 解码片选源 C2

:BUS1:SPI:SOURce:CS ?

查询返回 C2

:BUS<n>:SPI:SOURce:MOSI

■ **命令格式:**

:BUS<n>:SPI:SOURce:MOSI <source>

:BUS<n>:SPI:SOURce:MOSI ?

■ **功能描述:**

设置或查询 SPI 解码的 MOSI 源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 SPI 解码的 MOSI 源。

■ **举例:**

:BUS1:SPI:SOURce:MOSI C3

设置 BUS1 的 SPI 解码 MOSI 源 C3

:BUS1:SPI:SOURce:MOSI ?

查询返回 C3

:BUS<n>:SPI:SOURce:MISO**■ 命令格式:**

:BUS<n>:SPI:SOURce:MISO <source>

:BUS<n>:SPI:SOURce:MISO ?

■ 功能描述:

设置或查询 SPI 解码的 MISO 源。

■ 参数:

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回 SPI 解码的 MISO 源。

■ 举例:

:BUS1:SPI:SOURce:MISO C3

设置 BUS1 的 SPI 解码 MISO 源 C3

:BUS1:SPI:SOURce:MISO ?

查询返回 C3

:BUS<n>:SPI:LEVel:SCL**■ 命令格式:**

:BUS<n>:SPI:LEVel:SCL <level>

:BUS<n>:SPI:LEVel:SCL ?

■ 功能描述:

设置或查询 SPI 解码的时钟阈值, 默认单位 V。

■ 参数:

n:整型, 范围 1~2

level:实数型, 单位与通道垂直刻度单位一致, 默认单位为 V, 范围-8V~8V

■ 返回格式:

查询返回 SPI 解码的时钟阈值。

■ 举例:

:BUS1:SPI:LEVel:SCL 1

设置 BUS1 的 SPI 解码时钟阈值为 1V

:BUS1:SPI:LEVel:SCL ?

查询返回 1.00000E+000

:BUS<n>:SPI:LEVel:CS**■ 命令格式:**

:BUS<n>:SPI:LEVel:CS <level>

:BUS<n>:SPI:LEVel:CS ?

■ **功能描述:**

设置或查询 SPI 解码的片选阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-8V~8V

■ **返回格式:**

查询返回 SPI 解码的片选阈值。

■ **举例:**

:BUS1:SPI:LEVel:CS 1 设置 BUS1 的 SPI 解码片选阈值为 1V

:BUS1:SPI:LEVel:CS ? 查询返回 1.00000E+000

:BUS<n>:SPI:LEVel:MOSI

■ **命令格式:**

:BUS<n>:SPI:LEVel:MOSI <level>

:BUS<n>:SPI:LEVel:MOSI ?

■ **功能描述:**

设置或查询 SPI 解码的 MOSI 阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-8V~8V

■ **返回格式:**

查询返回 SPI 解码的 MOSI 阈值。

■ **举例:**

:BUS1:SPI:LEVel:MOSI 1 设置 BUS1 的 SPI 解码 MOSI 阈值为 1V

:BUS1:SPI:LEVel:MOSI ? 查询返回 1.00000E+000

:BUS<n>:SPI:LEVel:MISO

■ **命令格式:**

:BUS<n>:SPI:LEVel:MISO <level>

:BUS<n>:SPI:LEVel:MISO ?

■ **功能描述:**

设置或查询 SPI 解码的 MISO 阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-8V~8V

■ **返回格式:**

查询返回 SPI 解码的 MISO 阈值。

■ **举例:**

:BUS1:SPI:LEVel:MISO 1

设置 BUS1 的 SPI 解码 MISO 阈值为 1V

:BUS1:SPI:LEVel:MISO ?

查询返回 1.00000E+000

:BUS<n>:SPI:EDGe:SCL

■ **命令格式:**

:BUS<n>:SPI:EDGe:SCL <type>

:BUS<n>:SPI:EDGe:SCL ?

■ **功能描述:**

设置或查询 SPI 解码的时钟沿。

■ **参数:**

n:整型，范围 1~2

type:离散型{FALL | RISe}

FALL:下降沿

RISe:上升沿

■ **返回格式:**

查询返回 SPI 解码的时钟沿。

■ **举例:**

:BUS1:SPI:EDGe:SCL RISe

设置 BUS1 的 SPI 解码时钟沿为 RISe

:BUS1:SPI:EDGe:SCL ?

查询返回 RISe

:BUS<n>:SPI:EDGe:CS

■ **命令格式:**

:BUS<n>:SPI:EDGe:CS <type>

:BUS<n>:SPI:EDGe:CS ?

■ **功能描述:**

设置或查询 SPI 解码的片选沿。

■ **参数:**

n:整型, 范围 1~2

type:离散型{LOW | HIGH}

LOW:低电平

HIGH:高电平

■ **返回格式:**

查询返回 SPI 解码的片选沿。

■ **举例:**

:BUS1:SPI:EDGE:CS HIGH

设置 BUS1 的 SPI 解码片选沿为 HIGH

:BUS1:SPI:EDGE:CS ?

查询返回 HIGH

:BUS<n>:SPI:POLarity:MOSI

■ **命令格式:**

:BUS<n>:SPI:POLarity:MOSI <type>

:BUS<n>:SPI:POLarity:MOSI ?

■ **功能描述:**

设置或查询 SPI 解码的 MOSI 极性。

■ **参数:**

n:整型, 范围 1~2

type:离散型{NEGative | POSitive}

NEGative:负极性

POSitive:正极性

■ **返回格式:**

查询返回 SPI 解码的 MOSI 极性。

■ **举例:**

:BUS1:SPI:POLarity:MOSI POSitive

设置 BUS1 的 SPI 解码 MOSI 极性为 POSitive

:BUS1:SPI:POLarity:MOSI ?

查询返回 POSitive

:BUS<n>:SPI:POLarity:MISO

■ **命令格式:**

:BUS<n>:SPI:POLarity:MISO <type>

:BUS<n>:SPI:POLarity:MISO ?

■ **功能描述:**

设置或查询 SPI 解码的 MISO 极性。

■ **参数:**

n:整型, 范围 1~2

type:离散型{NEGative | POSitive}

NEGative:低电平

POSitive:高电平

■ **返回格式:**

查询返回 SPI 解码的 MISO 极性。

■ **举例:**

:BUS1:SPI:POLarity:MISO POSitive

设置 BUS1 的 SPI 解码 MISO 极性为 POSitive

:BUS1:SPI:POLarity:MISO ?

查询返回 POSitive

:BUS<n>:SPI:ORDer

■ **命令格式:**

:BUS<n>:SPI:ORDer <type>

:BUS<n>:SPI:ORDer ?

■ **功能描述:**

设置或查询 SPI 解码的位顺序。

■ **参数:**

n:整型, 范围 1~2

type:离散型{MSB | LSB}

MSB:低电平

LSB:高电平

■ **返回格式:**

查询返回 SPI 解码的位顺序。

■ **举例:**

:BUS1:SPI:ORDer LSB

设置 BUS1 的 SPI 解码位顺序为 LSB

:BUS1:SPI:ORDer ?

查询返回 LSB

CAN

:BUS<n>:CAN:SOURce

■ **命令格式:**

:BUS<n>:CAN:SOURce <source>

:BUS<n>:CAN:SOURce ?

■ **功能描述:**

设置或查询 CAN 解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 CAN 解码信源。

■ **举例:**

:BUS1:CAN:SOURce C1

设置 BUS1 的 CAN 解码源为 C1

:BUS1:CAN:SOURce ?

查询返回 C1

:BUS<n>:CAN:STYPe

■ **命令格式:**

:BUS<n>:CAN:STYPe <type>

:BUS<n>:CAN:STYPe ?

■ **功能描述:**

设置或查询 CAN 解码信号类型。

■ **参数:**

n:整型, 范围 1~2

type:离散型{L | H}

L:CAN_L

H:CAN_H

■ **返回格式:**

查询返回 CAN 解码信号类型。

■ **举例:**

:BUS1:CAN:STYPe H

设置 BUS1 的 CAN 解码信号类型为 CAN_H

:BUS1:CAN:STYPe ?

查询返回 H

:BUS<n>:CAN:LEVel

■ **命令格式:**

:BUS<n>:CAN:LEVel <level>

:BUS<n>:CAN:LEVel ?

■ **功能描述:**

设置或查询 CAN 解码阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-12V~12V

■ **返回格式:**

查询返回 CAN 解码阈值。

■ **举例:**

:BUS1:CAN:LEVel 1 设置 BUS1 的 CAN 解码阈值为 1V

:BUS1:CAN:LEVel ? 查询返回 1.00000E+000

:BUS<n>:CAN:BAUDrate:CURRent

■ **命令格式:**

:BUS<n>:CAN:BAUDrate:CURRent <value>

:BUS<n>:CAN:BAUDrate:CURRent ?

■ **功能描述:**

设置或查询 CAN 解码信号速率。

■ **参数:**

n:整型，范围 1~2

value:离散型{CUSTom | 10000 | 20000 | 33300 | 50000 | 62500 | 83300 | 100000 | 125000
| 1000000}

■ **返回格式:**

查询返回 CAN 解码信号速率。

■ **举例:**

:BUS1:CAN:BAUDrate:CURRent 10000 设置 BUS1 的 CAN 解码信号速率为 10k

:BUS1:CAN:BAUDrate:CURRent ? 查询返回 10000

:BUS<n>:CAN:BAUDrate:CUSTom

■ **命令格式:**

:BUS<n>:CAN:BAUDrate:CUSTom <value>

:BUS<n>:CAN:BAUDrate:CUSTom ?

■ **功能描述:**

设置或查询 CAN 解码自定义信号速率，默认单位 bps。

■ **参数:**

n:整型，范围 1~2

value:整数型，范围 10kbps~10Mbps

■ **返回格式:**

查询返回 CAN 解码自定义信号速率。

■ **举例:**

:BUS1:CAN:BAUDrate:CUSTom 10000

设置 BUS1 的 CAN 解码自定义信号速率为 10kbps

:BUS1:CAN:BAUDrate:CUSTom ?

查询返回 10000

:BUS<n>:CAN:SPOint

■ **命令格式:**

:BUS<n>:CAN:SPOint <value>

:BUS<n>:CAN:SPOint ?

■ **功能描述:**

设置或查询 CAN 解码采样点，默认单位%。

■ **参数:**

n:整型，范围 1~2

value:整数型，范围 30%~90%

■ **返回格式:**

查询返回 CAN 采样点。

■ **举例:**

:BUS1:CAN:SPOint 30

设置 BUS1 的 CAN 解码采样点为 30%

:BUS1:CAN:SPOint ?

查询返回 30

CAN-FD

:BUS<n>:CAN_FD:SOURce

■ **命令格式:**

:BUS<n>:CAN_FD:SOURce <source>

:BUS<n>:CAN_FD:SOURce ?

■ **功能描述:**

设置或查询 CAN_FD 解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 CAN_FD 解码信源。

■ **举例:**

:BUS1:CAN_FD:SOURce C1

设置 BUS1 的 CAN_FD 解码源为 C1

:BUS1:CAN_FD:SOURce ?

查询返回 C1

:BUS<n>:CAN_FD:STYPe

■ **命令格式:**

:BUS<n>:CAN_FD:STYPe <type>

:BUS<n>:CAN_FD:STYPe ?

■ **功能描述:**

设置或查询 CAN_FD 解码信号类型。

■ **参数:**

n:整型, 范围 1~2

type:离散型{L | H}

L:CAN-FD_L

H:CAN-FD_H

■ **返回格式:**

查询返回 CAN_FD 解码信号类型。

■ **举例:**

:BUS1:CAN_FD:STYPe H

设置 BUS1 的 CAN_FD 解码信号类型为 CAN_H

:BUS1:CAN_FD:STYPe ?

查询返回 H

:BUS<n>:CAN_FD:LEVel

■ **命令格式:**

:BUS<n>:CAN_FD:LEVel <level>

:BUS<n>:CAN_FD:LEVel ?

■ **功能描述:**

设置或查询 CAN_FD 解码阈值, 默认单位 V。

■ **参数:**

n:整型, 范围 1~2

level:实数型, 单位与通道垂直刻度单位一致, 默认单位为 V, 范围-12V~12V

■ 返回格式:

查询返回 CAN_FD 解码阈值。

■ 举例:

:BUS1:CAN_FD:LEVel 1 设置 BUS1 的 CAN_FD 解码阈值为 1V

:BUS1:CAN_FD:LEVel ? 查询返回 1.00000E+000

:BUS<n>:CAN_FD:BAUDRate:SD

■ 命令格式:

:BUS<n>:CAN_FD:BAUDRate:SDCURrent <value>

:BUS<n>:CAN_FD:BAUDRate:SDCURrent ?

■ 功能描述:

设置或查询 CAN_FD 解码 SD 信号速率。

■ 参数:

n:整型, 范围 1~2

value:离散型{CUSTom | 10000 | 20000 | 33300 | 50000 | 62500 | 83300 | 100000 | 125000
| 1000000}

■ 返回格式:

查询返回 CAN_FD 解码 SD 信号速率。

■ 举例:

:BUS1:CAN_FD:BAUDRate:SDCURrent 10000 设置 BUS1 的 CAN_FD 解码 SD 信号速率为
10k

:BUS1:CAN_FD:BAUDRate:SDCURrent ? 查询返回 10000

:BUS<n>:CAN_FD:BAUDRate:SDCUSTom

■ 命令格式:

:BUS<n>:CAN_FD:BAUDRate:SDCUSTom <value>

:BUS<n>:CAN_FD:BAUDRate:SDCUSTom ?

■ 功能描述:

设置或查询 CAN_FD 解码 SD 自定义信号速率, 默认单位 bps。

■ 参数:

n:整型, 范围 1~2

value:整数型, 范围 10kbps~1Mbps

■ **返回格式:**

查询返回 CAN_FD 解码 SD 自定义信号速率。

■ **举例:**

:BUS1:CAN_FD:BAUDrate:SDCUSTom 10000 设置 CAN_FD 解码 SD 自定义信号速率为 10k
:BUS1:CAN_FD:BAUDrate:SDCUSTom ? 查询返回 10000

:BUS<n>:CAN_FD:BAUDrate:FDCURrent

■ **命令格式:**

:BUS<n>:CAN_FD:BAUDrate:FDCURrent <value>

:BUS<n>:CAN_FD:BAUDrate:FDCURrent ?

■ **功能描述:**

设置或查询 CAN_FD 解码 FD 信号速率。

■ **参数:**

n:整型, 范围 1~2

value:离散型{CUSTom | 1000000 | 2000000 | 3000000 | 4000000 | 5000000 | 6000000
| 7000000 | 8000000}

■ **返回格式:**

查询返回 CAN_FD 解码 FD 信号速率。

■ **举例:**

:BUS1:CAN_FD:BAUDrate:FDCURrent 1000000 设置 CAN_FD 解码 FD 信号速率为 1M
:BUS1:CAN_FD:BAUDrate:FDCURrent ? 查询返回 1000000

:BUS<n>:CAN_FD:BAUDrate:FDCustom

■ **命令格式:**

:BUS<n>:CAN_FD:BAUDrate:FDCustom <value>

:BUS<n>:CAN_FD:BAUDrate:FDCustom ?

■ **功能描述:**

设置或查询 CAN_FD 解码 FD 自定义信号速率, 默认单位 bps。

■ **参数:**

n:整型, 范围 1~2

value:整数型, 范围 500kbps~16Mbps

■ **返回格式:**

查询返回 CAN_FD 解码 FD 自定义信号速率。

■ **举例:**

:BUS1:CAN_FD:BAUDrate:FDCustom 1000000 设置 CAN_FD 解码 FD 自定义信号速率为 1M

:BUS1:CAN_FD:BAUDrate:FDCustom ? 查询返回 1000000

:BUS<n>:CAN_FD:SPOint

■ **命令格式:**

:BUS<n>:CAN_FD:SPOint <value>

:BUS<n>:CAN_FD:SPOint ?

■ **功能描述:**

设置或查询 CAN_FD 解码采样点，默认单位%。

■ **参数:**

n:整型，范围 1~2

value:整数型，范围 30%~90%

■ **返回格式:**

查询返回 CAN_FD 采样点。

■ **举例:**

:BUS1:CAN_FD:SPOint 30 设置 BUS1 的 CAN 解码采样点为 30%

:BUS1:CAN_FD:SPOint ? 查询返回 30

LIN

:BUS<n>:LIN:SOURce

■ **命令格式:**

:BUS<n>:LIN:SOURce <source>

:BUS<n>:LIN:SOURce ?

■ **功能描述:**

设置或查询 LIN 解码信源。

■ **参数:**

n:整型，范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 LIN 解码信源。

■ **举例:**

:BUS1:LIN:SOURce C1	设置 BUS1 的 LIN 解码源为 C1
:BUS1:LIN:SOURce ?	查询返回 C1

:BUS<n>:LIN:LEVel

■ **命令格式:**

:BUS<n>:LIN:LEVel <level>

:BUS<n>:LIN:LEVel ?

■ **功能描述:**

设置或查询 LIN 解码阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-12V~12V

■ **返回格式:**

查询返回 LIN 解码阈值。

■ **举例:**

:BUS1:LIN:LEVel 1	设置 BUS1 的 LIN 解码阈值为 1V
:BUS1:LIN:LEVel ?	查询返回 1.00000E+000

:BUS<n>:LIN:POLarity

■ **命令格式:**

:BUS<n>:LIN:POLarity <type>

:BUS<n>:LIN:POLarity ?

■ **功能描述:**

设置或查询 LIN 解码极性。

■ **参数:**

n:整型，范围 1~2

type:离散型{POSitive | NEGative}

NEGative:负极性

POSitive:正极性

■ **返回格式:**

查询返回 LIN 解码极性。

■ **举例:**

:BUS1:LIN:POLarity POSitive	设置 BUS1 的 LIN 解码极性为 POSitive
:BUS1:LIN:POLarity ?	查询返回 POSitive

:BUS<n>:LIN:BAUDrate:CURRent

■ **命令格式:**

```
:BUS<n>:LIN:BAUDrate:CURRent <value>
:BUS<n>:LIN:BAUDrate:CURRent ?
```

■ **功能描述:**

设置或查询 LIN 解码波特率。

■ **参数:**

n:整型, 范围 1~2
value:离散型{CUSTom | 2400 | 4800 | 9600 | 19200}

■ **返回格式:**

查询返回 LIN 解码波特率。

■ **举例:**

:BUS1:LIN:BAUDrate:CURRent 9600	设置 BUS1 的 LIN 解码波特率为 9600
:BUS1:LIN:BAUDrate:CURRent ?	查询返回 9600

:BUS<n>:LIN:BAUDrate:CUSTom

■ **命令格式:**

```
:BUS<n>:LIN:BAUDrate:CUSTom <value>
:BUS<n>:LIN:BAUDrate:CUSTom ?
```

■ **功能描述:**

设置或查询 LIN 解码自定义波特率, 默认单位 bps。

■ **参数:**

n:整型, 范围 1~2
value:整数型, 范围 1kbps~200Mbps

■ **返回格式:**

查询返回 LIN 解码自定义波特率。

■ **举例:**

:BUS1:LIN:BAUDrate:CUSTom 10000	设置 BUS1 的 LIN 解码自定义波特率为 10kbps
:BUS1:LIN:BAUDrate:CUSTom ?	查询返回 10000

:BUS<n>:LIN:COUNT**■ 命令格式:**

:BUS<n>:LIN:COUNT <value>

:BUS<n>:LIN:COUNT ?

■ 功能描述:

设置或查询 LIN 解码字节数。

■ 参数:

n:整型, 范围 1~2

value:整数型, 范围 1~8

■ 返回格式:

查询返回 LIN 解码字节数。

■ 举例:

:BUS1:LIN:COUNT 8

设置 BUS1 的 LIN 解码字节数为 8

:BUS1:LIN:COUNT ?

查询返回 8

:BUS<n>:LIN:STANDARD**■ 命令格式:**

:BUS<n>:LIN:STANDARD <type>

:BUS<n>:LIN:STANDARD ?

■ 功能描述:

设置或查询 LIN 解码信号标准。

■ 参数:

n:整型, 范围 1~2

type:离散型{V1 | V2}

V1:LIN1.0

V2:LIN2.0

■ 返回格式:

查询返回 LIN 解码信号标准。

■ 举例:

:BUS1:LIN:STANDARD V1

设置 BUS1 的 LIN 解码信号标准为 LIN1.0

:BUS1:LIN:STANDARD ?

查询返回 V1

:BUS<n>:LIN:CKPID■ **命令格式:**

:BUS<n>:LIN:CKPID <type>

:BUS<n>:LIN:CKPID ?

■ **功能描述:**

设置或查询 LIN 解码是否检查 PID 校验。

■ **参数:**

n:整型, 范围 1~2

type:布尔型{ON | OFF}或{1 | 0}

ON:检查

OFF:不检查

■ **返回格式:**

查询返回 LIN 解码是否检查 PID 校验。

■ **举例:**

:BUS1:LIN:CKPID ON

设置 BUS1 的 LIN 解码检查 PID 校验

:BUS1:LIN:CKPID ?

查询返回 1

FLEXRAY**:BUS<n>:FR:SOURce**■ **命令格式:**

:BUS<n>:FR:SOURce <source>

:BUS<n>:FR:SOURce ?

■ **功能描述:**

设置或查询 FLEXRAY 解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 FLEXRAY 解码信源。

■ **举例:**

:BUS1:FR:SOURce C1

设置 BUS1 的 FLEXRAY 解码源为 C1

:BUS1:FR:SOURce ?

查询返回 C1

:BUS<n>:FR:STYPe

■ 命令格式:

:BUS<n>:FR:STYPe <type>

:BUS<n>:FR:STYPe ?

■ 功能描述:

设置或查询 FLEXRAY 解码源类型。

■ 参数:

n:整型, 范围 1~2

type:离散型{BP | BM}

■ 返回格式:

查询返回 FLEXRAY 解码源类型。

■ 举例:

:BUS1:FR:STYPe BP

设置 BUS1 的 FLEXRAY 解码源类型为 BP

:BUS1:FR:STYPe ?

查询返回 BP

:BUS<n>:FR:BAUDrate:CURRent

■ 命令格式:

:BUS<n>:FR:BAUDrate:CURRent <value>

:BUS<n>:FR:BAUDrate:CURRent ?

■ 功能描述:

设置或查询 FLEXRAY 解码波特率, 默认单位 bps。

■ 参数:

n:整型, 范围 1~2

value:离散型{CUSTom | 1000000 | 5000000 | 10000000}

■ 返回格式:

查询返回 FLEXRAY 解码波特率。

■ 举例:

:BUS1:FR:BAUDrate:CURRent 1000000

设置 BUS1 的 FLEXRAY 解码波特率为 1Mbps

:BUS1:FR:BAUDrate:CURRent ?

查询返回 1000000

:BUS<n>:FR:BAUDRate:CUSTom■ **命令格式:**

:BUS<n>:FR:BAUDRate:CUSTom <value>

:BUS<n>:FR:BAUDRate:CUSTom ?

■ **功能描述:**

设置或查询 FLEXRAY 解码自定义波特率，默认单位 bps。

■ **参数:**

n:整型，范围 1~2

value:整数型，范围 1Mbps~10Mbps

■ **返回格式:**

查询返回 FLEXRAY 解码自定义波特率。

■ **举例:**

:BUS1:FR:BAUDRate:CUSTom 1000000 设置 BUS1 的 FLEXRAY 解码自定义波特率为 1Mbps

:BUS1:FR:BAUDRate:CUSTom ? 查询返回 1000000

:BUS<n>:FR:LEVel■ **命令格式:**

:BUS<n>:FR:LEVel <level>

:BUS<n>:FR:LEVel ?

■ **功能描述:**

设置或查询 FLEXRAY 解码阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-12V~12V

■ **返回格式:**

查询返回 FLEXRAY 解码阈值。

■ **举例:**

:BUS1:FR:LEVel 1 设置 BUS1 的 FLEXRAY 解码阈值为 1V

:BUS1:FR:LEVel ? 查询返回 1.00000E+000

:BUS<n>:FR:CHANnel■ **命令格式:**

:BUS<n>:FR:CHANnel <type>

:BUS<n>:FR:CHANnel ?

■ **功能描述:**

设置或查询 FLEXRAY 解码通道类型。

■ **参数:**

n:整型, 范围 1~2

type:离散型{A | B}

■ **返回格式:**

查询返回 FLEXRAY 解码通道类型。

■ **举例:**

:BUS1:FR:CHANnel A

设置 BUS1 的 FLEXRAY 解码通道类型为 A

:BUS1:FR:CHANnel ?

查询返回 A

AUDIOBUS

:BUS<n>:AB:TYPE

■ **命令格式:**

:BUS<n>:AB:TYPE <type>

:BUS<n>:AB:TYPE ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码协议类型。

■ **参数:**

n:整型, 范围 1~2

type:离散型{I2S | LJ | RJ | TDM}

■ **返回格式:**

查询返回 AUDIOBUS 解码协议类型。

■ **举例:**

:BUS1:AB:TYPE I2S

设置 BUS1 的 AUDIOBUS 解码协议类型为 I2S

:BUS1:AB:TYPE ?

查询返回 I2S

:BUS<n>:AB:DPOLarity

■ **命令格式:**

:BUS<n>:AB:DPOLarity <type>

:BUS<n>:AB:DPOLarity ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码数据极性。

■ **参数:**

n:整型, 范围 1~2

type:离散型{POSitive | NEGative}

POSitive:正极性

NEGative:负极性

■ **返回格式:**

查询返回 AUDIOBUS 解码数据极性。

■ **举例:**

:BUS1:AB:DPOLarity NEGative

设置 BUS1 的 AUDIOBUS 解码数据极性为 NEGative

:BUS1:AB:DPOLarity ?

查询返回 NEGative

:BUS<n>:AB:SPOLarity

■ **命令格式:**

:BUS<n>:AB:SPOLarity <type>

:BUS<n>:AB:SPOLarity ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码同步极性。

■ **参数:**

n:整型, 范围 1~2

type:离散型{POSitive | NEGative}

POSitive:正极性

NEGative:负极性

■ **返回格式:**

查询返回 AUDIOBUS 解码同步极性。

■ **举例:**

:BUS1:AB:SPOLarity NEGative

设置 BUS1 的 AUDIOBUS 解码同步极性为 NEGative

:BUS1:AB:SPOLarity ?

查询返回 NEGative

:BUS<n>:AB:CEdGe

■ **命令格式:**

:BUS<n>:AB:CEDGe <type>

:BUS<n>:AB:CEDGe?

■ **功能描述:**

设置或查询 AUDIOBUS 解码有效时钟沿。

■ **参数:**

n:整型, 范围 1~2

type:离散型{RISe | FALL}

RISe:上升沿

FALL:下降沿

■ **返回格式:**

查询返回 AUDIOBUS 解码有效时钟沿。

■ **举例:**

:BUS1:AB:CEDGe FALL 设置 BUS1 的 AUDIOBUS 解码有效时钟沿为 FALL

:BUS1:AB:CEDGe ? 查询返回 FALL

:BUS<n>:AB:ORDer

■ **命令格式:**

:BUS<n>:AB:ORDer <type>

:BUS<n>:AB:ORDer ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码位序。

■ **参数:**

n:整型, 范围 1~2

type:离散型{MSB | LSB}

■ **返回格式:**

查询返回 AUDIOBUS 解码位序。

■ **举例:**

:BUS1:AB:ORDer LSB 设置 BUS1 的 AUDIOBUS 解码位序为 LSB

:BUS1:AB:ORDer ? 查询返回 LSB

:BUS<n>:AB:SCType

■ **命令格式:**

:BUS<n>:AB:SCType <type>

:BUS<n>:AB:SCType ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码声道类型。

■ **参数:**

n:整型, 范围 1~2

type:离散型{LR | L | R}

LR:左/右声道

L:左声道

R:右声道

■ **返回格式:**

查询返回 AUDIOBUS 解码声道类型。

■ **举例:**

:BUS1:AB:SCType R 设置 BUS1 的 AUDIOBUS 解码声道类型为右声道

:BUS1:AB:SCType ? 查询返回 R

:BUS<n>:AB:SCCount

■ **命令格式:**

:BUS<n>:AB:SCCount <value>

:BUS<n>:AB:SCCount ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码声道数。

■ **参数:**

n:整型, 范围 1~2

value:整数型, 范围 2~32

■ **返回格式:**

查询返回 AUDIOBUS 解码声道数。

■ **举例:**

:BUS1:AB:SCCount 2 设置 BUS1 的 AUDIOBUS 解码声道数为 2

:BUS1:AB:SCCount ? 查询返回 2

:BUS<n>:AB:DBCount

■ **命令格式:**

:BUS<n>:AB:DBCount <value>

:BUS<n>:AB:DBCount ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码每声道数据位数。

■ **参数:**

n:整型, 范围 1~2

value:整数型, 范围 0~64

■ **返回格式:**

查询返回 AUDIOBUS 解码每声道数据位数。

■ **举例:**

:BUS1:AB:DBCount 8 设置 BUS1 的 AUDIOBUS 解码每声道数据位数为 8

:BUS1:AB:DBCount ? 查询返回 8

:BUS<n>:AB:CBCount

■ **命令格式:**

:BUS<n>:AB:CBCount <value>

:BUS<n>:AB:CBCount ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码每声道时钟位数。

■ **参数:**

n:整型, 范围 1~2

value:整数型, 范围 0~64

■ **返回格式:**

查询返回 AUDIOBUS 解码每声道时钟位数。

■ **举例:**

:BUS1:AB:CBCount 8 设置 BUS1 的 AUDIOBUS 解码每声道时钟位数为 8

:BUS1:AB:CBCount ? 查询返回 8

:BUS<n>:AB:PCCount

■ **命令格式:**

:BUS<n>:AB:PCCount <value>

:BUS<n>:AB:PCCount ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码每帧通道数。

■ **参数:**

n:整型, 范围 1~2

value:整数型, 范围 2~64

■ **返回格式:**

查询返回 AUDIOBUS 解码每帧通道数。

■ **举例:**

:BUS1:AB:PCCount 8 设置 BUS1 的 AUDIOBUS 解码每帧通道数为 8

:BUS1:AB:PCCount ? 查询返回 8

:BUS<n>:AB:SOURce:SCL

■ **命令格式:**

:BUS<n>:AB:SOURce:SCL <source>

:BUS<n>:AB:SOURce:SCL ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码时钟源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 AUDIOBUS 解码时钟源。

■ **举例:**

:BUS1:AB:SOURce:SCL C4 设置 BUS1 的 AUDIOBUS 解码时钟源为 C4

:BUS1:AB:SOURce:SCL ? 查询返回 C4

:BUS<n>:AB:SOURce:CS

■ **命令格式:**

:BUS<n>:AB:SOURce:CS <source>

:BUS<n>:AB:SOURce:CS ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码片选源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回 AUDIOBUS 解码片选源。

■ 举例:

:BUS1:AB:SOURce:CS C3

设置 BUS1 的 AUDIOBUS 解码片选源为 C3

:BUS1:AB:SOURce:CS ?

查询返回 C3

:BUS<n>:AB:SOURce:SDA

■ 命令格式:

:BUS<n>:AB:SOURce:SDA <source>

:BUS<n>:AB:SOURce:SDA ?

■ 功能描述:

设置或查询 AUDIOBUS 解码数据源。

■ 参数:

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回 AUDIOBUS 解码数据源。

■ 举例:

:BUS1:AB:SOURce:SDA C2

设置 BUS1 的 AUDIOBUS 解码数据源为 C2

:BUS1:AB:SOURce:SDA ?

查询返回 C2

:BUS<n>:AB:LEVel:SCL

■ 命令格式:

:BUS<n>:AB:LEVel:SCL <level>

:BUS<n>:AB:LEVel:SCL ?

■ 功能描述:

设置或查询 AUDIOBUS 解码时钟阈值, 默认单位 V。

■ 参数:

n:整型, 范围 1~2

level:实数型, 单位与通道垂直刻度单位一致, 默认单位为 V, 范围-20V~20V

■ 返回格式:

查询返回 AUDIOBUS 解码时钟阈值。

■ 举例:

:BUS1:AB:LEVel:SCL 1	设置 BUS1 的 AUDIOBUS 解码时钟阈值为 1V
:BUS1:AB:LEVel:SCL ?	查询返回 1.00000E+000

:BUS<n>:AB:LEVel:CS

■ 命令格式:

:BUS<n>:AB:LEVel:CS <level>

:BUS<n>:AB:LEVel:CS ?

■ 功能描述:

设置或查询 AUDIOBUS 解码片选阈值，默认单位 V。

■ 参数:

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-20V~20V

■ 返回格式:

查询返回 AUDIOBUS 解码片选阈值。

■ 举例:

:BUS1:AB:LEVel:CS 1	设置 BUS1 的 AUDIOBUS 解码片选阈值为 1V
:BUS1:AB:LEVel:CS ?	查询返回 1.00000E+000

:BUS<n>:AB:LEVel:SDA

■ 命令格式:

:BUS<n>:AB:LEVel:SDA <level>

:BUS<n>:AB:LEVel:SDA ?

■ 功能描述:

设置或查询 AUDIOBUS 解码数据阈值，默认单位 V。

■ 参数:

n:整型，范围 1~2

level:实数型，单位与通道垂直刻度单位一致，默认单位为 V，范围-20V~20V

■ 返回格式:

查询返回 AUDIOBUS 解码数据阈值。

■ 举例:

:BUS1:AB:LEVel:SDA 1	设置 BUS1 的 AUDIOBUS 解码数据阈值为 1V
:BUS1:AB:LEVel:SDA ?	查询返回 1.00000E+000

:BUS<n>:AB:BITDelay■ **命令格式:**

:BUS<n>:AB:BITDelay <value>

:BUS<n>:AB:BITDelay ?

■ **功能描述:**

设置或查询 AUDIOBUS 解码位延迟。

■ **参数:**

n:整型, 范围 1~2

value:整数型, 范围 0~64

■ **返回格式:**

查询返回 AUDIOBUS 解码位延迟。

■ **举例:**

:BUS1:AB:BITDelay 10 设置 BUS1 的 AUDIOBUS 解码位延迟为 10

:BUS1:AB:BITDelay ? 查询返回 10

MIL_STD_1553**:BUS<n>:MS:SOURce**■ **命令格式:**

:BUS<n>:MS:SOURce <source>

:BUS<n>:MS:SOURce ?

■ **功能描述:**

设置或查询 MIL_STD_1553 解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 MIL_STD_1553 解码信源。

■ **举例:**

:BUS1:MS:SOURce C4 设置 BUS1 的 MIL_STD_1553 解码信源为 C4

:BUS1:MS:SOURce ? 查询返回 C4

:BUS<n>:MS:BAUDRate:CURRent**■ 命令格式:**

:BUS<n>:MS:BAUDRate:CURRent <value>

:BUS<n>:MS:BAUDRate:CURRent ?

■ 功能描述:

设置或查询 MIL_STD_1553 解码波特率，默认单位为 bps。

■ 参数:

n:整型，范围 1~2

value:离散型{CUSTom | 1000000 | 10000000}

■ 返回格式:

查询返回 MIL_STD_1553 解码波特率。

■ 举例:

:BUS1:MS:BAUDRate:CURRent 1000000 设置 BUS1 的 MIL_STD_1553 解码波特率为 1Mbps

:BUS1:MS:BAUDRate:CURRent ? 查询返回 1000000

:BUS<n>:MS:BAUDRate:CUSTom**■ 命令格式:**

:BUS<n>:MS:BAUDRate:CUSTom <value>

:BUS<n>:MS:BAUDRate:CUSTom ?

■ 功能描述:

设置或查询 MIL_STD_1553 解码波特率，默认单位为 bps。

■ 参数:

n:整型，范围 1~2

value:整数型，范围 1kbps~20Mbps

■ 返回格式:

查询返回 MIL_STD_1553 解码波特率。

■ 举例:

:BUS1:MS:BAUDRate:CUSTom 1000000 设置 BUS1 的 MIL_STD_1553 解码波特率为 1Mbps

:BUS1:MS:BAUDRate:CUSTom ? 查询返回 1000000

:BUS<n>:MS:LEVel:HIGH**■ 命令格式:**

:BUS<n>:MS:LEVel:HIGH <value>

:BUS<n>:MS:LEVel:HIGH ?

■ **功能描述:**

设置或查询 MIL_STD_1553 解码高阈值，默认单位 V。

■ **参数:**

n:整型，范围 1~2

value:实数型，范围-30V~30V

■ **返回格式:**

查询返回 MIL_STD_1553 解码高阈值。

■ **举例:**

:BUS1:MS:LEVel:HIGH 2 设置 BUS1 的 MIL_STD_1553 解码高阈值为 2V

:BUS1:MS:LEVel:HIGH ? 查询返回 2.00000E+000

:BUS<n>:MS:LEVel:LOW

■ **命令格式:**

:BUS<n>:MS:LEVel:LOW <value>

:BUS<n>:MS:LEVel:LOW ?

■ **功能描述:**

设置或查询 MIL_STD_1553 解码低阈值,默认单位 V。

■ **参数:**

n:整型，范围 1~2

value:实数型，范围-30V~30V

■ **返回格式:**

查询返回 MIL_STD_1553 解码低阈值。

■ **举例:**

:BUS1:MS:LEVel:LOW 1 设置 BUS1 的 MIL_STD_1553 解码低阈值为 1V

:BUS1:MS:LEVel:LOW ? 查询返回 1.00000E+000

:BUS<n>:MS:POLarity

■ **命令格式:**

:BUS<n>:MS:POLarity <polarity>

:BUS<n>:MS:POLarity ?

■ **功能描述:**

设置或查询 MIL_STD_1553 解码极性。

■ **参数:**

n:整型, 范围 1~2

polarity:离散型{POSitive | NEGative}

POSitive:正极性

NEGative:负极性

■ **返回格式:**

查询返回 MIL_STD_1553 解码极性。

■ **举例:**

:BUS1:MS:POLarity POSitive 设置 BUS1 的 MIL_STD_1553 解码极性为正极性

:BUS1:MS:POLarity ? 查询返回 POSitive

ARINC429

:BUS<n>:A429:SOURce

■ **命令格式:**

:BUS<n>:A429:SOURce <source>

:BUS<n>:A429:SOURce ?

■ **功能描述:**

设置或查询 ARINC429 解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ **返回格式:**

查询返回 ARINC429 解码信源。

■ **举例:**

:BUS1:A429:SOURce C4 设置 BUS1 的 ARINC429 解码信源为 C4

:BUS1:A429:SOURce ? 查询返回 C4

:BUS<n>:A429:BAUDrate:CURRent

■ **命令格式:**

:BUS<n>:A429:BAUDrate:CURRent <value>

:BUS<n>:A429:BAUDrate:CURRent ?

■ **功能描述:**

设置或查询 ARINC429 解码波特率，默认单位为 bps。

■ **参数:**

n:整型，范围 1~2

value:离散型{CUSTom | 12.5k | 100k}

■ **返回格式:**

查询返回 ARINC429 解码波特率。

■ **举例:**

:BUS1:A429:BAUDrate:CURRent 12.5k	设置 BUS1 的 ARINC429 解码波特率为 12.5kbps
:BUS1:A429:BAUDrate:CURRent ?	查询返回 12.5k

:BUS<n>:A429:BAUDrate:CUSTom

■ **命令格式:**

:BUS<n>:A429:BAUDrate:CUSTom <value>

:BUS<n>:A429:BAUDrate:CUSTom ?

■ **功能描述:**

设置或查询 ARINC429 解码波特率，默认单位为 bps。

■ **参数:**

n:整型，范围 1~2

value:整数型，范围 1kbps~10Mbps

■ **返回格式:**

查询返回 ARINC429 解码波特率。

■ **举例:**

:BUS1:A429:BAUDrate:CUSTom 1000000	设置 BUS1 的 ARINC429 解码波特率为 1Mbps
:BUS1:A429:BAUDrate:CUSTom ?	查询返回 1000000

:BUS<n>:A429:LEVel:HIGH

■ **命令格式:**

:BUS<n>:A429:LEVel:HIGH <value>

:BUS<n>:A429:LEVel:HIGH ?

■ **功能描述:**

设置或查询 ARINC429 解码高阈值,默认单位 V。

■ **参数:**

n:整型, 范围 1~2

value:实数型, 范围-20V~20V

■ 返回格式:

查询返回 ARINC429 解码高阈值。

■ 举例:

:BUS1:A429:LEVel:HIGH 2 设置 BUS1 的 ARINC429 解码高阈值为 2V

:BUS1:A429:LEVel:HIGH ? 查询返回 2.00000E+000

:BUS<n>:A429:LEVel:LOW

■ 命令格式:

:BUS<n>:A429:LEVel:LOW <value>

:BUS<n>:A429:LEVel:LOW ?

■ 功能描述:

设置或查询 ARINC429 解码低阈值, 默认单位 V。

■ 参数:

n:整型, 范围 1~2

value:实数型, 范围-20V~20V

■ 返回格式:

查询返回 ARINC429 解码低阈值。

■ 举例:

:BUS1:A429:LEVel:LOW 1 设置 BUS1 的 ARINC429 解码低阈值为 1V

:BUS1:A429:LEVel:LOW ? 查询返回 1.00000E+000

:BUS<n>:A429:POLarity

■ 命令格式:

:BUS<n>:A429:POLarity <polarity>

:BUS<n>:A429:POLarity ?

■ 功能描述:

设置或查询 ARINC429 解码极性。

■ 参数:

n:整型, 范围 1~2

polarity:离散型{POSitive | NEGative}

POSitive:正极性

NEGative:负极性

■ **返回格式:**

查询返回 ARINC429 解码极性。

■ **举例:**

:BUS1:A429:POLarity POSitive

设置 BUS1 的 ARINC429 解码极性为 POSitive

:BUS1:A429:POLarity ?

查询返回 POSitive

:BUS<n>:A429:LEVel:MODE

■ **命令格式:**

:BUS<n>:A429:MODE <type>

:BUS<n>:A429:MODE ?

■ **功能描述:**

设置或查询 ARINC429 解码模式。

■ **参数:**

n:整型, 范围 1~2

type:离散型{19 | 21 | 23}

■ **返回格式:**

查询返回 ARINC429 解码模式。

■ **举例:**

:BUS1:A429:MODE 21

设置 BUS1 的 ARINC429 解码模式为 21BIT

:BUS1:A429:MODE ?

查询返回 21

SENT

:BUS<n>:SENT:SOURce

■ **命令格式:**

:BUS<n>:SENT:SOURce <source>

:BUS<n>:SENT:SOURce ?

■ **功能描述:**

设置或查询 SENT 解码信源。

■ **参数:**

n:整型, 范围 1~2

source:离散型{C1 | C2 | C3 | C4}

■ 返回格式:

查询返回 SENT 解码信源。

■ 举例:

:BUS1:SENT:SOURce C2 设置 BUS1 的 SENT 解码信源为 C2

:BUS1:SENT:SOURce ? 查询返回 C2

:BUS<n>:SENT:BAUDrate:CURRent**■ 命令格式:**

:BUS<n>:SENT:BAUDrate:CURRent <value>

:BUS<n>:SENT:BAUDrate:CURRent ?

■ 功能描述:

设置或查询 SENT 解码速率，默认单位为 bps。

■ 参数:

n:整型，范围 1~2

value:离散型{CUSTom | 10000 | 20000 | 33300 | 50000 | 62500 | 83300 | 100000 | 125000
| 1000000}

■ 返回格式:

查询返回 SENT 解码速率。

■ 举例:

:BUS1:SENT:BAUDrate:CURRent 1000000 设置 BUS1 的 SENT 解码速率为 1Mbps

:BUS1:SENT:BAUDrate:CURRent ? 查询返回 1000000

:BUS<n>:SENT:BAUDrate:CUSTom**■ 命令格式:**

:BUS<n>:SENT:BAUDrate:CUSTom <value>

:BUS<n>:SENT:BAUDrate:CUSTom ?

■ 功能描述:

设置或查询 SENT 解码速率，默认单位为 bps。

■ 参数:

n:整型，范围 1~2

value:整数型，范围 1kbps~10Mbps

■ 返回格式:

查询返回 SENT 解码速率。

■ 举例:

:BUS1:SENT:BAUDrate:CUSTom 1000000	设置 BUS1 的 SENT 解码速率为 1Mbps
:BUS1:SENT:BAUDrate:CUSTom ?	查询返回 1000000

:BUS<n>:SENT:POLarity**■ 命令格式:**

:BUS<n>:SENT:POLarity <polarity>
:BUS<n>:SENT:POLarity ?

■ 功能描述:

设置或查询 SENT 解码极性。

■ 参数:

n:整型, 范围 1~2
polarity:离散型{POSitive | NEGative}
POSitive:正极性
NEGative:负极性

■ 返回格式:

查询返回 SENT 解码极性。

■ 举例:

:BUS1:SENT:POLarity POSitive	设置 BUS1 的 SENT 解码极性为 POSitive
:BUS1:SENT:POLarity ?	查询返回 POSitive

:BUS<n>:SENT:THReshold**■ 命令格式:**

:BUS<n>:SENT:THReshold <value>
:BUS<n>:SENT:THReshold ?

■ 功能描述:

设置或查询 SENT 解码阈值, 默认单位 V。

■ 参数:

n:整型, 范围 1~2
value:实数型, 范围-12V~12V

■ 返回格式:

查询返回 SENT 解码阈值。

■ 举例:

:BUS1:SENT:THReshold 1	设置 BUS1 的 SENT 解码阈值为 1V
:BUS1:SENT:THReshold ?	查询返回 1.00000E+000

:BUS<n>:SENT:DLENgth

■ 命令格式:

:BUS<n>:SENT:DLENgth <value>

:BUS<n>:SENT:DLENgth ?

■ 功能描述:

设置或查询 SENT 解码数据长度。

■ 参数:

n:整型, 范围 1~2

value:离散型{1 | 2 | 3 | 4 | 5 | 6}

■ 返回格式:

查询返回 SENT 数据长度。

■ 举例:

:BUS1:SENT:DLENgth 2	设置 BUS1 的 SENT 解码数据长度为 2Nibbles
----------------------	---------------------------------

:BUS1:SENT:DLENgth ?	查询返回 2
----------------------	--------

:BUS<n>:SENT:PAUSE

■ 命令格式:

:BUS<n>:SENT:PAUSE <type>

:BUS<n>:SENT:PAUSE ?

■ 功能描述:

设置或查询 SENT 解码有无暂停位。

■ 参数:

n:整型, 范围 1~2

type:离散型{NO | YES}

NO:无

YES:有

■ 返回格式:

查询返回 SENT 解码暂停位。

■ 举例:

:BUS1:SENT:PAUSE YES	设置 BUS1 的 SENT 解码有暂停位
----------------------	-----------------------

:BUS1:SENT:PAUSE ?

查询返回 YES

:BUS<n>:SENT:CRC

■ 命令格式:

:BUS<n>:SENT:CRC <type>

:BUS<n>:SENT:CRC ?

■ 功能描述:

设置或查询 SENT 解码 CRC。

■ 参数:

n:整型, 范围 1~2

type:离散型{V2008 | V2010}

■ 返回格式:

查询返回 SENT 解码 CRC。

■ 举例:

:BUS1:SENT:CRC V2010

设置 BUS1 的 SENT 解码 CRC 为 V2010

:BUS1:SENT:CRC ?

查询返回 V2010

辅助输入输出

:AUX:INPut:TYPE

■ 命令格式:

:AUX:INPut:TYPE <type>

:AUX:INPut:TYPE ?

■ 功能描述:

设置或查询辅助输入类型。

■ 参数:

type:离散型{CLOSE | TRIGger | AWGSync}

CLOSE:关闭

TRIGger:触发同步

AWGSync:AWG 外触发

■ 返回格式:

查询返回辅助输入类型。

■ 举例:

:AUX:INPut:TYPe TRIGger

设置辅助输入为触发同步

:AUX:INPut:TYPe ?

查询返回 TRIGger

:AUX:INPut:POLarity

■ 命令格式:

:AUX:INPut:POLarity <type>

:AUX:INPut:POLarity ?

■ 功能描述:

设置或查询输入极性。

■ 参数:

type:离散型{POSitive | NEGative}

POSitive:正极性

NEGative:负极性

■ 返回格式:

查询返回输入极性。

■ 举例:

:AUX:INPut:POLarity NEGative

设置输入极性为负极性

:AUX:INPut:POLarity ?

查询返回 NEGative

:AUX:OUTPut:TYPe

■ 命令格式:

:AUX:OUTPut:TYPe <type>

:AUX:OUTPut:TYPe ?

■ 功能描述:

设置或查询辅助输出类型。

■ 参数:

type:离散型{CLOSE | TRIGger | AWGSync | PFTest}

CLOSE:关闭

TRIGger:触发同步

AWGSync:AWG 触发

PFTest:通过测试

■ 返回格式:

查询返回辅助输出类型。

■ **举例:**

:AUX:OUTPut:TYPe TRIGger	设置辅助输出为触发同步
:AUX:OUTPut:TYPe ?	查询返回 TRIGger

:AUX:OUTPut:POLarity

■ **命令格式:**

```
:AUX:OUTPut:POLarity <type>
:AUX:OUTPut:POLarity ?
```

■ **功能描述:**

设置或查询输出极性。

■ **参数:**

type:离散型{RISe | FALL}
 RISe:正极性
 FALL:负极性

■ **返回格式:**

查询返回输出极性。

■ **举例:**

:AUX:OUTPut:POLarity FALL	设置输出极性为负极性
:AUX:OUTPut:POLarity ?	查询返回 FALL

通信

:NETWork:LAN:COUNt

■ **命令格式:**

```
:NETWork:LAN:COUNt ?
```

■ **功能描述:**

查询当前处于已连接状态的以太网网络适配器的个数。

:NETWork:LAN:SElect

■ **命令格式:**

```
:NETWork:LAN:SElect <index>
:NETWork:LAN:SElect ?
```

■ **功能描述:**

设置或查询当前选中的以太网网络适配器。

■ **参数:**

index:整数型, index 小于或等于查询到的已连接的以太网网络适配器个数

■ **返回格式:**

查询返回当前选中的以太网网络适配器。

■ **举例:**

:NETWork:LAN:SElect 1 设置选中第一个以太网网络适配器

:NETWork:LAN:SElect ? 查询返回 1

:NETWork:LAN:RESet

■ **命令格式:**

:NETWork:LAN:RESet

■ **功能描述:**

用于立即重设当前设置的网络参数到默认参数。

:NETWork:LAN:APPLy

■ **命令格式:**

:NETWork:LAN:APPLy

■ **功能描述:**

设置已选中的适配器立即生效当前设置的网络参数。

:NETWork:LAN:IPADdress

■ **命令格式:**

:NETWork:LAN:IPADdress <adress>

:NETWork:LAN:IPADdress ?

■ **功能描述:**

用于设置或查询已选中的适配器的 IP 地址, 格式为 xxx.xxx.xxx.xxx。

■ **参数:**

adress:字符串类型

■ **返回格式:**

查询返回 IP 地址。

■ **举例:**

:NETWork:LAN:IPADdress "192.168.137.10" 设置 IP 地址为 192.168.137.10

:NETWork:LAN:IPADdress ?

查询返回 192.168.137.10

:NETWork:LAN:SMASK

■ 命令格式:

:NETWork:LAN:SMASK <smask>

:NETWork:LAN:SMASK ?

■ 功能描述:

设置或查询已选中的适配器的子网掩码，格式为 xxx.xxx.xxx.xxx。

■ 参数:

smask:字符串类型

■ 返回格式:

查询返回子网掩码。

■ 举例:

:NETWork:LAN:SMASK "255.255.255.0"

设置子网掩码为 255.255.255.0

:NETWork:LAN:SMASK ?

查询返回 255.255.255.0

:NETWork:LAN:GATEway

■ 命令格式:

:NETWork:LAN:GATEway <gateway>

:NETWork:LAN:GATEway ?

■ 功能描述:

设置或查询已选中的适配器的默认网关，格式为 xxx.xxx.xxx.xxx。

■ 参数:

gateway:字符串类型

■ 返回格式:

查询返回默认网关。

■ 举例:

:NETWork:LAN:GATEway "192.168.137.1"

设置默认网关为 192.168.137.1

:NETWork:LAN:GATEway ?

查询返回 192.168.137.1

:NETWork:LAN:DHCP

■ 命令格式:

:NETWork:LAN:DHCP <type>

:NETWork:LAN:DHCP ?

■ **功能描述:**

设置或查询已选中的适配器的配置模式（自动 IP）和（手动 IP）。

■ **参数:**

type:离散型{AUTO | MANual}

AUTO:自动

MANual:手动

■ **返回格式:**

查询返回动态配置模式。

■ **举例:**

:NETWork:LAN:DHCP AUTO 设置配置模式为 AUTO

:NETWork:LAN:DHCP ? 查询返回 AUTO

:NETWork:LAN:SPEed

■ **命令格式:**

:NETWork:LAN:SPEed ?

■ **功能描述:**

用于查询已选中的适配器的网络传输速度。

:NETWork:LAN:MAC

■ **命令格式:**

:NETWork:LAN:MAC ?

■ **功能描述:**

用于查询 MAC 物理地址。

:NETWork:WIFI:RESet

■ **命令格式:**

:NETWork:WIFI:RESet

■ **功能描述:**

用于立即重设 WIFI 当前设置的网络参数。

:NETWork:WiFi:APPLY**■ 命令格式:**

:NETWork:WiFi:APPLY

■ 功能描述:

用于立即生效当前 WiFi 设置的网络参数。

:NETWork:WiFi:IPADdress**■ 命令格式:**

:NETWork:WiFi:IPADdress <adress>

:NETWork:WiFi:IPADdress?

■ 功能描述:

设置或查询 WiFi 的 IP 地址，格式为 xxx.xxx.xxx.xxx。

■ 参数:

adress:字符串类型

■ 返回格式:

查询返回 WiFi 的 IP 地址。

■ 举例:

:NETWork:WiFi:IPADdress 192.168.100.11 设置子网掩码为 192.168.100.11

:NETWork:WiFi:IPADdress ? 查询返回 192.168.100.11

:NETWork:WiFi:SMASK**■ 命令格式:**

:NETWork:WiFi:SMASK <smask>

:NETWork:WiFi:SMASK ?

■ 功能描述:

设置或查询 WiFi 的子网掩码，格式为 xxx.xxx.xxx.xxx。

■ 参数:

smask:字符串类型

■ 返回格式:

查询返回子网掩码。

■ 举例:

:NETWork:WiFi:SMASK "255.255.255.0" 设置子网掩码为 255.255.255.0

:NETWork:WIFI:SMASK ?

查询返回 255.255.255.0

:NETWork:WIFI:GATEway

■ 命令格式:

:NETWork:WIFI:GATEway <gateway>

:NETWork:WIFI:GATEway ?

■ 功能描述:

设置或查询 WIFI 的默认网关，格式为 xxx.xxx.xxx.xxx。

■ 参数:

gateway:字符串类型

■ 返回格式:

查询返回默认网关。

■ 举例:

:NETWork:WIFI:GATEway "192.168.1.1"

设置子网掩码为 192.168.1.1

:NETWork:WIFI:GATEway ?

查询返回 192.168.1.1

:NETWork:WIFI:DHCP

■ 命令格式:

:NETWork:WIFI:DHCP <type>

:NETWork:WIFI:DHCP ?

■ 功能描述:

设置或查询 WIFI 的配置模式（自动 IP）和（手动 IP）。

■ 参数:

type:离散型{AUTO | MANual}

AUTO:自动

MANual:手动

■ 返回格式:

查询返回动态配置模式。

■ 举例:

:NETWork:WIFI:DHCP AUTO

设置子网掩码为 AUTO

:NETWork:WIFI:DHCP ?

查询返回 AUTO

:NETWork:WIFI:SPEed**■ 命令格式:**

:NETWork:WIFI:SPEed ?

■ 功能描述:

用于查询 WIFI 的网络传输速度。

:NETWork:WIFI:MAC**■ 命令格式:**

:NETWork:WIFI:MAC ?

■ 功能描述:

用于查询 WIFI 的 MAC 物理地址。



公司简介

优利德科技（中国）股份有限公司（以下简称为“优利德”）成立于 2003 年，总部位于东莞松山湖，是一家集仪器仪表自主研发、生产、销售为一体的国家高新技术企业，主要包括通用仪表、专业仪表、温度及环境测试仪表、测试仪器四大产品线，广泛应用于电子、家用电器、机电设备、节能环保、轨道交通、汽车制造、暖通、建筑工程、5G 新基建、新能源、物联网、大数据中心、人工智能、电力建设及维护、高等教育和科学研究等领域。

优利德拥有东莞、成都及常州三处研发中心。凭借较强的研发实力，参与四项国家标准的起草，获得三次中国专利优秀奖，连续两次被评为国家知识产权优势企业。截至 2023 年 6 月 30 日，累计获得专利 481 项，其中发明专利 64 项、实用新型专利 176 项、外观设计专利 216 项、软件著作权 24 项，拥有 22 项核心技术，具备一定的技术领先优势。

优利德自 2007 年至今一直被评定为“国家高新技术企业”，并在中国仪器仪表行业协会第五届至第八届理事会(2007 年-2023 年)任理事单位。2020 年，获得二十届中国专利优秀奖；2021 年，在上海证券交易所科创板上市(股票代码：688628)；2022 年，获批设立广东省博士工作站、博士后创新实践基地，入选广东省“2022 年创新型中小企业”名单，获评广东省“2022 年专精特新中小企业”，并被认定为“2022 年国家知识产权优势企业”，标志着优利德的持续创新能力、专业技术水平、研发能力及综合实力得到国家相关部门的认可，随着物联网大数据、云计算及人工智能和制造业的深度融合发展趋势，优利德基于企业多年积累的多元化智能测量传感器应用技术，为全球合作伙伴提供高精度工业测量智能传感设备及行业物联网解决方案，赋能物联网产业发展，为 IoT 生态合作伙伴创造更多价值。

企业规模

公司在东莞及河源合计拥有约 10 万平方米的生产基地，建立了先进、完备的产品生产和质量控制体系，合计设计年产能达到 1000 万台以上，确保为全球用户提供了持续稳定的产品供应保障。

公司理念

公司一直秉承为全球用户提供高质量、高安全性、高可靠性、高性价比的测试测量产品及综合解决方案，坚持以科技及人文为本，致力于成为世界一流的仪器仪表民族品牌。

销售网络

[优利德测试仪器销售网络](#)

联系我们

UNI-T 技术支持热线：400-876-7822

声明：

UNI-T 是优利德科技（中国）股份有限公司的英文名称和商标。

本文档中的产品信息可不经通知而变更，有关 UNI-T 最新的产品、

应用、服务等方面的信息请访问 UNI-T 官方网站：www.uni-trend.com.cn

