

第一章 概述

建立远程通信

本机可以通过 USB、LAN、RS232 接口与计算机之间进行远程通信。

使用 USB 接口时，本机被识别为 USB-TMC 设备，编程资源描述符类似为：

`"USB0::0x0483::0x5740::UDP51183557335E::INSTR"`。

使用 LAN 接口时，本机为 VXI 兼容设备，编程资源描述符类似为：

`"TCPIP::192.168.10.142::INSTR"`。

使用 RS232 接口时，命令结束符为 "\n"。

命令列表：

1.APPLy

`:APPLy [CH1|CH2|CH3|SER|PARA],[<volt>|MINimum|MAXimum], [<curr>|MINimum|MAXimum]`

`:APPLy? [CH1|CH2|CH3|SER|PARA],[CURRent|VOLTage]`

2.INSTrument

`:INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}`

`:INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}`

`:INSTrument[:SELEct] ?`

`:INSTrument[:SELEct] ?`

`:INSTrument:NSELEct {1|2|3|5|6}`

`:INSTrument:NSELEct?`

3.SOURce

`:SOURce:Mode {NORMal|SER|PARA}`

`:SOURce:Mode?`

`[<SOURce#>:VOLTage[:LEVEl][:IMMediate][:AMPLitude] {<vol>|MINimum|MAXimum}`

`[<SOURce#>:VOLTage[:LEVEl][:IMMediate][:AMPLitude]?`

`[<SOURce#>:VOLTage:PROTEction[:LEVEl] {<vol>|MINimum|MAXimum}`

`[<SOURce#>:VOLTage:PROTEction[:LEVEl]?`

[:SOURce#]:VOLTage:PROTection:STATe {0|1|OFF|ON}
[:SOURce#]:VOLTage:PROTection:STATe?
[:SOURce#]:CURRent [:LEVel][:IMMediate][:AMPLitude] {<curr>|MINimum|MAXimum}
[:SOURce#]:CURRent [:LEVel][:IMMediate][:AMPLitude]?
[:SOURce#]:CURRent:PROTection[:LEVel] {<curr>|MINimum|MAXimum}
[:SOURce#]:CURRent:PROTection[:LEVel]?
[:SOURce#]:CURRent:PROTection:STATe {0|1|OFF|ON}
[:SOURce#]:CURRent:PROTection:STATe?

4. OUTPut

:OUTPut[:STATe] [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}
:OUTPut[:STATe]? [CH1|CH2|CH3|SER|PARA]
:OUTPut:CVCC? [CH1|CH2|CH3|SER|PARA]
:OUTPut:OVP:VALue [CH1|CH2|CH3|SER|PARA,] {<vol>|MINimum|MAXimum}
:OUTPut:OVP:VALue? [CH1|CH2|CH3|SER|PARA]
:OUTPut:OVP[:STATe] [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}
:OUTPut:OVP[:STATe]? [CH1|CH2|CH3|SER|PARA]
:OUTPut:OCP:VALue [CH1|CH2|CH3|SER|PARA,] {<curr>|MINimum|MAXimum}
:OUTPut:OCP:VALue? [CH1|CH2|CH3|SER|PARA]
:OUTPut:OCP[:STATe] [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}
:OUTPut:OCP[:STATe]? [CH1|CH2|CH3|SER|PARA]

5. MEASure

:MEASure:ALL[:DC]? [CH1|CH2|CH3|SER|PARA]
:MEASure[:VOLTage][:DC]? [CH1|CH2|CH3|SER|PARA]
:MEASure:CURRent[:DC]? [CH1|CH2|CH3|SER|PARA]
:MEASure:POWER[:DC]? [CH1|CH2|CH3|SER|PARA]

6. LISTout

:LISTout[:STATe] {0|1|OFF|ON}
:LISTout[:STATe]?
:LISTout:BASE <StrN>, <Grpn>, <CycN>, {OFF|LAST}
:LISTout:BASE?
:LISTout:PARAMeter <index>, <volt>, <curr>, <time>
:LISTout:PARAMeter? <index> [,<count>]
:LISTout:TEMPLet:SElect {SINE|PULSE|RAMP|UP|DN|UPDN|RISE|FALL}
:LISTout:TEMPLet:SElect?
:LISTout:TEMPLet:OBJect {V|C}

:LISTout:TEMPlet:OBJect? {V|C}
:LISTout:TEMPlet:START < index>
:LISTout:TEMPlet:START?
:LISTout:TEMPlet:POINTs <point>
:LISTout:TEMPlet:POINTs?
:LISTout:TEMPlet:MAXValue {<value>|MINimum|MAXimum}
:LISTout:TEMPlet:MAXValue?
:LISTout:TEMPlet:MINValue{<value>|MINimum|MAXimum}
:LISTout:TEMPlet:MINValue?
:LISTout:TEMPlet:INTERval <value>
:LISTout:TEMPlet:INTERval?
:LISTout:TEMPlet:INVErt {0|1|OFF|ON}
:LISTout:TEMPlet:INVErt?
:LISTout:TEMPlet:WIDTh <value>
:LISTout:TEMPlet:WIDTh?
:LISTout:TEMPlet:PERIod <value>
:LISTout:TEMPlet:PERIod?
:LISTout:TEMPlet:SYMMetry<value>
:LISTout:TEMPlet:SYMMetry?
:LISTout:TEMPlet:EXPRate<value>
:LISTout:TEMPlet:EXPRate?
:LISTout:TEMPlet:CONSTRUCT

7. DELAY

:DELAY[:STATe] {0|1|OFF|ON}
:DELAY[:STATe]?
:DELAY:START <StrN>
:DELAY:START?
:DELAY:GROUPs <Grpn>
:DELAY:GROUPs?
:DELAY:CYCLEs <CycN>
:DELAY:CYCLEs?
:DELAY:ENDState { ON|OFF|LAST}
:DELAY:ENDState?
:DELAY:STOP {NONE|<V|>V|<C|>C|<P|>P} [,<value>]
:DELAY:STOP?
:DELAY:PARAMeter <index>, {ON|OFF}, <time>
:DELAY:PARAMeter? <index> [,<count>]

:DELAY:GENERate:STAT < index>, <point>, {01P|10P}
:DELAY:GENERate:FIX< index>, <point>, <time_on>, <time_off>
:DELAY:GENERate:INC< index>, <point>, <time_base>, <time_step>
:DELAY:GENERate:DEC < index>, <point>, <time_base>, <time_step>
:DELAY:GENERate?

8. PRESet

:PRESet#[:APPLy]
:PRESet#:SET:VOLTage {CH1|CH2|CH3|SER|PARA}, {<volt>|MINimum|MAXimum}
:PRESet#:SET:VOLTage? {CH1|CH2|CH3|SER|PARA}
:PRESet#:SET:CURRent{CH1|CH2|CH3|SER|PARA}, {<curr>|MINimum|MAXimum}
:PRESet#:SET:CURRent? {CH1|CH2|CH3|SER|PARA}
:PRESet#:SET:OVP {CH1|CH2|CH3|SER|PARA}, {0|1|OFF|ON} [, {<volt>|MINimum|MAXimum}]
:PRESet#:SET:OVP? {CH1|CH2|CH3|SER|PARA}
:PRESet#:SET:OCP {CH1|CH2|CH3|SER|PARA}, {0|1|OFF|ON} [, {<curr>|MINimum|MAXimum}]
:PRESet#:SET:OCP? {CH1|CH2|CH3|SER|PARA}

9. MONItor

:MONItor[:STATe] {0|1|OFF|ON}
:MONItor[:STATe]?
:MONItor:VOLTage {<V|>V|NONE} [, <volt>|MINimum|MAXimum]
:MONItor:VOLTage?
:MONItor:CURRent {<C|>C|NONE} [, <curr>|MINimum|MAXimum]
:MONItor:CURRent?
:MONItor:POWER {<P|>P|NONE} [, <watt>|MINimum|MAXimum]
:MONItor:POWER?
:MONItor:LOGic {1|2}, {AND|OR}
:MONItor:LOGic? {1|2}
:MONItor:STOPway {OUTOFF|MSG|BEEPER}, {ON|OFF}
:MONItor:STOPway?

10. TRIGger

:TRIGger:IN[:ENABLE] {D0|D1|D2|D3}, {0|1|OFF|ON}
:TRIGger:IN[:ENABLE]? {D0|D1|D2|D3}
:TRIGger:IN:SOURce {D0|D1|D2|D3}, {CH1|CH2|CH3|SER|PARA}, {CH1|CH2|CH3|SER|PARA} [, {CH1|CH2|CH3|SER|PARA}]
:TRIGger:IN:SOURce? {D0|D1|D2|D3}
:TRIGger:IN:TYPE {D0|D1|D2|D3}, {RISE|FALL|HIGH|LOW}
:TRIGger:IN:TYPE? {D0|D1|D2|D3}

:TRIGger:IN:SENSitivity {D0|D1|D2|D3},{LOW|MID|HIGH}
:TRIGger:IN:SENSitivity? {D0|D1|D2|D3}
:TRIGger:IN:RESPonse {D0|D1|D2|D3},{ON|OFF|ALTER}
:TRIGger:IN:RESPonse? {D0|D1|D2|D3}
:TRIGger:OUT[:ENABLE] {D0|D1|D2|D3},{0|1|OFF|ON}
:TRIGger:OUT[:ENABLE]? {D0|D1|D2|D3}
:TRIGger:OUT:SOURce {D0|D1|D2|D3},{CH1|CH2|CH3|SER|PARA}
:TRIGger:OUT:SOURce? {D0|D1|D2|D3},
:TRIGger:OUT:CONDition {D0|D1|D2|D3},{AUTO|OUTOFF|OUTON|>V|<V|=V|>C|<C|=C|>P|<P|=P},{<value>}
:TRIGger:OUT:CONDition? {D0|D1|D2|D3}
:TRIGger:OUT:POLARity {D0|D1|D2|D3},{POSItive|NEGAtive}
:TRIGger:OUT:POLARity? {D0|D1|D2|D3}

11. SYSTem

:SYSTem:BEEPer[:STATe] {0|1|OFF|ON}
:SYSTem:BEEPer[:STATe]?
:SYSTem:COMMunicate:LAN:APPLY
:SYSTem:COMMunicate:LAN:DHCP[:STATe] {0|1|OFF|ON}
:SYSTem:COMMunicate:LAN:DHCP[:STATe]?
:SYSTem:COMMunicate:LAN:IPADdress "x.x.x.x"
:SYSTem:COMMunicate:LAN:IPADdress?
:SYSTem:COMMunicate:LAN:SMASK "x.x.x.x"
:SYSTem:COMMunicate:LAN:SMASK?
:SYSTem:COMMunicate:LAN:GATEway "x.x.x.x"
:SYSTem:COMMunicate:LAN:GATEway?
:SYSTem:COMMunicate:RS232:BAUD {4800|7200|9600|14400|19200|38400|57600|115200|128000}
:SYSTem:COMMunicate:RS232:BAUD?
:SYSTem:BRIGhtness <value>
:SYSTem:BRIGhtness?

第二章 命令详解

1.APPLy

:APPLy [CH1|CH2|CH3|SER|PARA],[<volt>|MINimum|MAXimum], [<curr>|MINimum|MAXimum]

功能: 设置指定通道为当前通道, 同时可设置该通道的电压、电流设置值。

说明: 只有在普通模式时才能选择“CH1、CH2”, 只有在串联模式时才能选择“SER”, 只有在并联模式时才能选择“PARA”。同时省略“volt”和“curr”参数时, 只设置指定通道为当前通道, 不改变通道的电压、电流设置值。

举例: :APPLy CH1,15.00V, 2.000A

:APPLy? [CH1|CH2|CH3|SER|PARA],[CURRent|VOLTage]

功能: 查询某通道的电压、电流设置值。

说明: 如果省略通道参数, 则默认指定当前通道为需要查询的通道。如果指定“CURRent”, 则返回电流设置值; 如果指定“VOLTage”, 则返回电压设置值; 如果不指定, 则同时返回电压、电流设置值。

举例: :APPLy? CH1, VOLT

返回: CH1, 15.00

2.INSTrument

:INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}

:INSTrument[:SELEct] {CH1|CH2|CH3|SER|PARA}

功能： 设置指定通道为当前通道。

说明： 只有在普通模式时才能选择 “CH1、CH2” ，只有在串联模式时才能选择 “SER” ，只有在并联模式时才能选择 “PARA” 。

设置电源的模式请参见命令 “:SOURce:Mode {NORMal|SER|PARA}”

举例： :INSTrument:SELE Ch3

:INSTrument[:SELEct] ?

:INSTrument[:SELEct] ?

功能： 查询当前通道。

说明： 返回 “CH1|CH2|CH3|SER|PARA” 。

举例： :INSTrument:SELE?

返回： CH1

:INSTrument:NSELEct {1|2|3|5|6}

功能： 设置指定通道为当前通道。

说明： 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。只有在普通模式时才能选择 “1、2” ，只有在串联模式时才能选择 “5” ，只有在并联模式时才能选择 “6” 。

举例： :INSTrument:NSELEct 3

:INSTrument:NSELEct?

功能： 查询当前通道。

说明： 返回 “1|2|3|5|6” 。

举例： :INSTrument:NSELEct?

返回： 3

3.SOURce

:SOURce:Mode {NORMAl|SER|PARA}

功能: 设置电源的工作模式。

说明: NORMAl 设置为普通独立模式, SER 设置为串联模式, PARA 设置为并联模式。电源切换模式时需要一段时间, 在此期间执行一些与电源工作模式相关的命令会导致命令执行失败; 因此, 切换电源工作模式后, 至少间隔 500 毫秒后再执行新的命令。

举例: :SOURce:Mode SER

:SOURce:Mode?

功能: 查询电源当前的工作模式。

说明: 返回 "NORMAl|SER|PARA" 。

举例: :SOURce:Mode?

返回: SER

[:SOURce#]:VOLTage[:LEVel][:IMMediate][:AMPLitude] {<vol>|MINimum|MAXimum}

功能: 设置某通道的电压设置值, 并设置该通道为当前通道。

说明: #的可选值为 "1|2|3|5|6", 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略 [:SOURce#]或#, 则默认值为 CH1。只有在普通模式时才能选择 "1、2", 只有在串联模式时才能选择 "5", 只有在并联模式时才能选择 "6" 。

举例: :SOURce1:VOLTage 25.00

[:SOURce#]:VOLTage[:LEVel][:IMMediate][:AMPLitude]?

功能: 查询某通道的电压设置值。

说明: #的可选值为 "1|2|3|5|6", 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略 [:SOURce#]或#, 则默认值为 CH1。

举例: :SOURce1:VOLTage?

返回: 25.00

[:SOURce#]:VOLTage:PROTection[:LEVel] {<vol>|MINimum|MAXimum}

功能: 设置某通道的过压保护值, 并设置该通道为当前通道。

说明: #的可选值为 "1|2|3|5|6", 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略 [:SOURce#]或#, 则默认值为 CH1。只有在普通模式时才能选择 "1、2", 只有在串联模式时才能选择 "5", 只有在并联模式时才能选择 "6" 。

举例: :SOURce1:VOLTage:PROTection 30.00

[:SOURce#]:VOLTage:PROTection[:LEVel]?

功能: 查询某通道的过压保护值。

说明: #的可选值为 "1|2|3|5|6", 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略 [:SOURce#]或#, 则默认值为 CH1。

举例: :SOURce1:VOLTage:PROTection?

返回: 30.00

[[:SOURce#]:VOLTage:PROTection:STATe {0|1|OFF|ON}

功能: 设置某通道的过压保护开关, 并设置该通道为当前通道。

说明: #的可选值为“1|2|3|5|6”, 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略[:SOURce#]或#, 则默认值为 CH1。只有在普通模式时才能选择“1、2”, 只有在串联模式时才能选择“5”, 只有在并联模式时才能选择“6”。

举例: :SOURce1:VOLTage:PROTection:STATe 1

[[:SOURce#]:VOLTage:PROTection:STATe?

功能: 查询某通道的过压保护开关。

说明: #的可选值为“1|2|3|5|6”, 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略[:SOURce#]或#, 则默认值为 CH1。返回“ON | OFF”

举例: :SOURce1:VOLTage:PROTection:STATe?

返回: ON

[[:SOURce#]:CURRent [:LEVel][:IMMediate][:AMPLitude] {<curr>|MINimum|MAXimum}

功能: 设置某通道的电流设置值, 并设置该通道为当前通道。

说明: #的可选值为“1|2|3|5|6”, 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略[:SOURce#]或#, 则默认值为 CH1。只有在普通模式时才能选择“1、2”, 只有在串联模式时才能选择“5”, 只有在并联模式时才能选择“6”。

举例: :SOURce1:CURRent 5.000

[[:SOURce#]:CURRent [:LEVel][:IMMediate][:AMPLitude]?

功能: 查询某通道的电流设置值。

说明: #的可选值为“1|2|3|5|6”, 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略[:SOURce#]或#, 则默认值为 CH1。

举例: :SOURce1:CURRent?

返回: 5.000

[[:SOURce#]:CURRent:PROTection[:LEVel] {<curr>|MINimum|MAXimum}

功能: 设置某通道的过流保护值, 并设置该通道为当前通道。

说明: #的可选值为“1|2|3|5|6”, 1 对应 CH1, 2 对应 CH2, 3 对应 CH3, 5 对应 SER, 6 对应 PARA。如果省略[:SOURce#]或#, 则默认值为 CH1。只有在普通模式时才能选择“1、2”, 只有在串联模式时才能选择“5”, 只有在并联模式时才能选择“6”。

举例: :SOURce1:CURRent:PROTection 5.000

[[:SOURce#]:CURRent:PROTection[:LEVel]?

功能: 查询某通道的过流保护值。

说明: #的可选值为“1|2|3|5|6”，1对应CH1，2对应CH2，3对应CH3，5对应SER，6对应PARA。如果省略[:SOURce#]或#，则默认值为CH1。

举例: :SOURce1:CURRent:PROTection?

返回: 5.000

[[:SOURce#]:CURRent:PROTection:STATe {0|1|OFF|ON}

功能: 设置某通道的过流保护开关，并设置该通道为当前通道。

说明: #的可选值为“1|2|3|5|6”，1对应CH1，2对应CH2，3对应CH3，5对应SER，6对应PARA。如果省略[:SOURce#]或#，则默认值为CH1。只有在普通模式时才能选择“1、2”，只有在串联模式时才能选择“5”，只有在并联模式时才能选择“6”。

举例: :SOURce1:CURRent:PROTection:STATe 1

[[:SOURce#]:CURRent:PROTection:STATe?

功能: 查询某通道的过流保护开关。

说明: #的可选值为“1|2|3|5|6”，1对应CH1，2对应CH2，3对应CH3，5对应SER，6对应PARA。如果省略[:SOURce#]或#，则默认值为CH1。返回“ON | OFF”。

举例: :SOURce1:CURRent:PROTection:STATe?

返回: ON

4.OUTPUT

:OUTPUT[:STATe] [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}

功能： 打开或关闭某通道的输出开关，并设置该通道为当前通道。

说明： 如果省略通道参数，则默认指定当前通道为需要控制的通道。只有在普通模式时才能选择“CH1、CH2”，只有在串联模式时才能选择“SER”，只有在并联模式时才能选择“PARA”。

举例： :OUTPUT:STATe CH1, ON

:OUTPUT[:STATe]? [CH1|CH2|CH3|SER|PARA]

功能： 查询某通道的输出开关状态。

说明： 如果省略通道参数，则默认指定当前通道为需要查询的通道。返回“ON | OFF”。

举例： :OUTPUT:STATe? CH1

返回： ON

:OUTPUT:CVCC? [CH1|CH2|CH3|SER|PARA]

功能： 查询某通道的恒压、恒流状态。

说明： 如果省略通道参数，则默认指定当前通道为需要查询的通道。返回“CV | CC”。

举例： :OUTPUT:CVCC? CH1

返回： CV

:OUTPUT:OVP:VALue [CH1|CH2|CH3|SER|PARA,] {<vol>|MINimum|MAXimum}

功能： 设置某通道的过压保护值，并设置该通道为当前通道。

说明： 如果省略通道参数，则默认指定当前通道为需要控制的通道。只有在普通模式时才能选择“CH1、CH2”，只有在串联模式时才能选择“SER”，只有在并联模式时才能选择“PARA”。

举例： :OUTPUT:OVP:VALue CH1, 5

:OUTPUT:OVP:VALue? [CH1|CH2|CH3|SER|PARA]

功能： 查询某通道的过压保护值。

说明： 如果省略通道参数，则默认指定当前通道为需要查询的通道。

举例： :OUTPUT:OVP:VALue? CH1

返回： 5.00

:OUTPUT:OVP[:STATe] [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}

功能： 设置某通道的过压保护开关，并设置该通道为当前通道。

说明： 如果省略通道参数，则默认指定当前通道为需要控制的通道。只有在普通模式时才能选择“CH1、CH2”，只有在串联模式时才能选择“SER”，只有在并联模式时才能选择“PARA”。

举例： :OUTPUT:OVP:STATe CH1, ON

:OUTPut:OVP[:STATe]? [CH1|CH2|CH3|SER|PARA]

功能： 查询某通道的过压保护开关。

说明： 如果省略通道参数，则默认指定当前通道为需要查询的通道。返回 “ON | OFF”

举例： :OUTPut:OVP:STATe? CH1

返回： ON

:OUTPut:OCP:VALue [CH1|CH2|CH3|SER|PARA,] {<curr>|MINimum|MAXimum}

功能： 设置某通道的过流保护值，并设置该通道为当前通道。

说明： 如果省略通道参数，则默认指定当前通道为需要控制的通道。只有在普通模式时才能选择 “CH1、CH2” ，只有在串联模式时才能选择 “SER” ，只有在并联模式时才能选择 “PARA” 。

举例： :OUTPut:OCP:VALue CH1, 5.1

:OUTPut:OCP:VALue? [CH1|CH2|CH3|SER|PARA]

功能： 查询某通道的过流保护值。

说明： 如果省略通道参数，则默认指定当前通道为需要查询的通道。

举例： :OUTPut:OCP:VALue? CH1

返回： 5.100

:OUTPut:OCP[:STATe] [CH1|CH2|CH3|SER|PARA,] {0|1|OFF|ON}

功能： 设置某通道的过流保护开关，并设置该通道为当前通道。

说明： 如果省略通道参数，则默认指定当前通道为需要控制的通道。只有在普通模式时才能选择 “CH1、CH2” ，只有在串联模式时才能选择 “SER” ，只有在并联模式时才能选择 “PARA” 。

举例： :OUTPut:OCP:STATe CH1, ON

:OUTPut:OCP[:STATe]? [CH1|CH2|CH3|SER|PARA]

功能： 查询某通道的过流保护开关。

说明： 如果省略通道参数，则默认指定当前通道为需要查询的通道。返回 “ON | OFF”

举例： :OUTPut:OCP:STATe? CH1

返回： ON

5. MEASure

:MEASure:ALL[:DC]? [CH1|CH2|CH3|SER|PARA]

功能: 查询指定通道的输出端子上的实际电压、电流、功率值。

说明: 如果省略通道参数, 则默认指定当前通道为需要查询的通道。

举例: :MEASure:ALL? CH1

返回: 05.10,0.089,00.45

:MEASure[:VOLTage][:DC]? [CH1|CH2|CH3|SER|PARA]

功能: 查询指定通道的输出端子上的实际电压值。

说明: 如果省略通道参数, 则默认指定当前通道为需要查询的通道。

举例: :MEASure:VOLTage? CH1

返回: 05.10

:MEASure:CURRent[:DC]? [CH1|CH2|CH3|SER|PARA]

功能: 查询指定通道的输出端子上的实际电流值。

说明: 如果省略通道参数, 则默认指定当前通道为需要查询的通道。

举例: :MEASure:CURRent? CH1

返回: 0.089

:MEASure:POWEr[:DC]? [CH1|CH2|CH3|SER|PARA]

功能: 查询指定通道的输出端子上的实际功率值。

说明: 如果省略通道参数, 则默认指定当前通道为需要查询的通道。

举例: :MEASure:POWEr? CH1

返回: 00.45

6. LISTout

:LISTout[:STATE] {0|1|OFF|ON}

功能: 启动或停止当前通道的列表输出模式。

说明: 可通过“INSTrument”章节的指令设置和查询当前通道。

举例: :LISTout:STATe ON

:LISTout[:STATE]?

功能: 查询当前通道的列表输出模式的状态。

说明: 返回“运行状态(ON|PAUSED|OFF), 剩余时间, 当前点, 终止点, 剩余循环次数, 停止状态(OFF|LAST)”。

举例: :LISTout[:STATE]?

返回: ON,1,7,2047,0,OFF

:LISTout:BASE <StrN>, <Grpn>, <CycN>, {OFF|LAST}

功能: 设置当前通道的列表输出模式的基本参数。

说明: StrN: 输出起始组号, 范围 0-2047, StrN + Grpn 不能大于 2048。

Grpn: 输出组数, 范围 1-2048, StrN + Grpn 不能大于 2048。

CycN: 循环次数, 范围 1-99999。

OFF|LAST: 停止状态, 关闭输出或保持最后的输出状态。

注意: 列表输出模式处于运行状态时, 无法设置参数。

举例: :LISTout:BASE 1,100,1,OFF

:LISTout:BASE?

功能: 查询当前通道的列表输出模式的基本参数。

说明: 返回“ StrN, Grpn, CycN, {OFF|LAST}”

StrN: 输出起始组号, 范围 0-2047。

Grpn: 输出组数, 范围 1-2048。

CycN: 循环次数, 范围 1-99999。

OFF|LAST: 停止状态, 关闭输出或保持最后的输出状态。

举例: :LISTout:BASE?

返回: 1,100,1,OFF

:LISTout:PARAMeter <index>, <volt>, <curr>, <time>

功能: 设置当前通道的列表输出模式的组参数。

说明: index: 需要设置参数的组的组序号, 范围 0-2047。

volt: 该组的输出电压, 单位为伏。

curr: 该组的输出电流, 单位为安。

time: 该组的输出时长, 单位为秒。

注意: 列表输出模式处于运行状态时, 无法设置参数。

举例: :LISTout:PARAMeter 0, 10.00, 3.00, 10

:LISTout:PARAMeter? <index> [,<count>]

功能: 查询当前通道的列表输出模式的组参数。

说明: **index:** 从此序号的组开始查询设置参数, 范围 0-2047。

count: 需要查询的组的个数, 范围 1-10。省略此参数时, 默认查询 1 组。

返回格式: 以数据块格式返回数据, 数据段为各组的序号、电压、电流、时间。如: “#2180,10.000,3.000,10;” , #218 表示数据段有 18 个数据—— “0,10.000,3.000,10;” , 数据内容显示组号 0, 电压 10.000 , 电流 3.000, 时间 10。

举例: :LISTout:PARAMeter? 0

返回: #2180,10.000,3.000,10;

:LISTout:TEMPlet:SElect {SINE|PULSE|RAMP|UP|DN|UPDN|RISE|FALL}

功能: 为当前通道的列表输出模式选择模板类型。

说明: **SINE:** 正弦模板;

PULSE: 脉冲模板;

RAMP: 斜坡模板;

UP: 阶梯上升模板;

DN: 阶梯下降模板;

UPDN: 上下阶梯模板;

RISE: 指数上升模板;

FALL: 指数下降模板。

举例: :LISTout:TEMPlet:SElect SINE

:LISTout:TEMPlet:SElect?

功能: 查询当前通道的列表输出模式的模板类型。

说明: 返回{SINE|PULSE|RAMP|UP|DN|UPDN|RISE|FALL}。

举例: :LISTout:PARAMeter?

返回: SINE

:LISTout:TEMPlet:OBject {V|C}

功能: 设置当前通道的列表输出模式的模板的构建对象。

说明: **V:** 电压; **C:** 电流;

举例: :LISTout:TEMPlet:OBject V

:LISTout:TEMPlet:OBject? {V|C}

功能: 查询当前通道的列表输出模式的模板的构建对象。

说明: 返回{V|C}, **V:** 电压; **C:** 电流。

举例: :LISTout:TEMPlet:OBject?

返回: V

:LISTout:TEMPLet:STARTt < index >

功能: 设置当前通道的列表输出模式的模板的构建起始组号, 以此组为第一个模板构建点。

说明: index, 起始构建组, 范围 0 至 (2048 减去需要构建的组数) 。

举例: :LISTout:TEMPLet:STARTt 0

:LISTout:TEMPLet:START?

功能: 查询当前通道的列表输出模式的模板的构建起始组号, 此组为第一个模板构建点。

说明: 返回<index>, 起始构建组, 范围 0 至 (2048 减去需要构建的组数) 。

举例: :LISTout:TEMPLet:START?

返回: 0

:LISTout:TEMPLet:POINTs <point >

功能: 设置当前通道的列表输出模式的模板的构建组数。

说明: point, 构建组数, 从起始组开始构建、共构建 point 指定的组数, 不改变其他组的设置值。脉冲模板最少构建 2 个组, 其他模板最少构建 10 个组。

举例: :LISTout:TEMPLet:POINTs 50

:LISTout:TEMPLet:POINTs?

功能: 查询当前通道的列表输出模式的模板的构建组数。

说明: 返回< point >, 构建组数。

举例: :LISTout:TEMPLet:POINTs?

返回: 50

:LISTout:TEMPLet:MAXValue {<value>|MINimum|MAXimum}

功能: 设置当前通道的列表输出模式的模板的构建最大值。

说明: value: 实型数值, 指定一个数值。

MINimum: 0。

MAXimum: 各通道的最大输出值。

举例: :LISTout:TEMPLet:MAXValue 5.55

:LISTout:TEMPLet:MAXValue?

功能: 查询当前通道的列表输出模式的模板的构建最大值。

说明: 返回< value >, 实型数值。

举例: :LISTout:TEMPLet:MAXValue?

返回: 5.55

:LISTout:TEMPlet:MINValue{<value>|MINimum|MAXimum}

功能：设置当前通道的列表输出模式的模板的构建最小值。

说明：value：实型数值，指定一个数值。

MINimum：0。

MAXimum：各通道的最大输出值。

举例：:LISTout:TEMPlet:MINValue 1.11

:LISTout:TEMPlet:MINValue?

功能：查询当前通道的列表输出模式的模板的构建最小值。

说明：返回< value >，实型数值。

举例：:LISTout:TEMPlet:MINValue?

返回：1.11

:LISTout:TEMPlet:INTERval <value>

功能：设置当前通道的列表输出模式的模板的构建间隔时间。

说明：value：整型数值，指定组参数的持续时间，范围：1——99999。

举例：:LISTout:TEMPlet:INTERval 100

:LISTout:TEMPlet:INTERval?

功能：查询当前通道的列表输出模式的模板的构建间隔时间。

说明：返回< value >，整型数值。

举例：:LISTout:TEMPlet:INTERval?

返回：100

:LISTout:TEMPlet:INVErt {0|1|OFF|ON}

功能：设置当前通道的列表输出模式的模板的构建反相开关。

说明：只有正弦、脉冲、斜坡模板具有反相功能，在这些模板时才能使用此命令。

举例：:LISTout:TEMPlet:INVErt ON

:LISTout:TEMPlet:INVErt?

功能：查询当前通道的列表输出模式的模板的构建反相开关。

说明：返回{OFF|ON}。

举例：:LISTout:TEMPlet:INVErt?

返回：ON

:LISTout:TEMPlet:WIDTh <value>

功能：设置当前通道的列表输出模式的脉冲模板的脉冲宽度。

说明：value：整型数值，指定脉冲宽度，单位为秒，范围：1——（周期-1）。只有在脉冲模板时才能使用此命令。

举例：:LISTout:TEMPlet:WIDTh 5

:LISTout:TEMPlet:WIDTh?

功能： 查询当前通道的列表输出模式的脉冲模板的脉冲宽度。

说明： 返回<value>， 整型数值。

举例： :LISTout:TEMPlet:WIDTh?

返回： 5

:LISTout:TEMPlet:PERIod <value>

功能： 设置当前通道的列表输出模式的脉冲模板的周期。

说明： value： 整型数值， 指定脉冲的周期， 单位为秒， 范围：（脉冲宽度 + 1）——99999。只有在脉冲模板时才能使用此命令。

举例： :LISTout:TEMPlet:PERIod 10

:LISTout:TEMPlet:PERIod?

功能： 查询当前通道的列表输出模式的脉冲模板的周期。

说明： 返回<value>， 整型数值。

举例： :LISTout:TEMPlet:PERIod?

返回： 10

:LISTout:TEMPlet:SYMMetry <value>

功能： 设置当前通道的列表输出模式的斜坡模板的对称性。

说明： value： 整型数值， 指定斜坡的对称性， 范围： 0——100。只有在斜坡模板时才能使用此命令。

举例： :LISTout:TEMPlet:SYMMetry 50

:LISTout:TEMPlet:SYMMetry?

功能： 查询当前通道的列表输出模式的斜坡模板的对称性。

说明： 返回<value>， 整型数值。

举例： :LISTout:TEMPlet:SYMMetry?

返回： 50

:LISTout:TEMPlet:EXPRate <value>

功能： 设置当前通道的列表输出模式的指数模板的指数。

说明： value： 整型数值， 范围： 0——10。在指数上升模板时， 使用此命令设置上升指数； 在指数下降模板时， 使用此命令设置下降指数； 其他模板下不可使用此命令。

举例： :LISTout:TEMPlet:EXPRate 5

:LISTout:TEMPlet:EXPRate?

功能： 查询当前通道的列表输出模式的斜坡模板的对称性。

说明： 返回<value>, 整型数值。

举例： :LISTout:TEMPlet:EXPRate?

返回： 5

:LISTout:TEMPlet:CONStruct

功能： 启动构建列表组参数。

说明： 仪器接收到此命令后, 按已设置的模板参数开始构建列表输出的组参数。

举例： :LISTout:TEMPlet:CONStruct

7. DELAY

:DELAY[:STATe] {0|1|OFF|ON}

功能：启动或停止当前通道的延时器。

说明：可通过“INSTrument”章节的指令设置和查询当前通道。

举例： :DELAY:STATe ON

:DELAY[:STATe]?

功能：查询当前通道的延时器状态。

说明：返回“运行状态(ON|OFF), 剩余时间, 当前点, 终止点, 剩余循环次数, 停止状态(OFF|LAST|ON)”。

举例： :DELAY[:STATe]?

返回： ON,1,12,2047,0,OFF

:DELAY:STARTt <StrN>

功能：设置当前通道的延时器输出起始组号。

说明： StrN: 输出起始组号, 整形数值, 范围 0-2047, (起始组号 + 输出组数) 不能大于 2048。

注意：延时器处于运行状态时, 无法设置参数。

举例： :DELAY:STARTt 0

:DELAY:START?

功能：查询当前通道的延时器输出起始组号。

说明：返回“StrN”, 整形数值。

举例： :DELAY:START?

返回： 0

:DELAY:GROUPs <Grpn>

功能：设置当前通道的延时器的输出组数。

说明： Grpn: 输出组数, 整形数值, 范围 1-2048, (起始组号 + 输出组数) 不能大于 2048。

注意：延时器处于运行状态时, 无法设置参数。

举例： :DELAY:GROUPs 100

:DELAY:GROUPs?

功能：查询当前通道的延时器的输出组数。

说明：返回“Grpn”, 整形数值。

举例： :DELAY:GROUPs?

返回： 100

:DELAY:CYCLEs <CycN>

功能: 设置当前通道的延时器的循环次数。

说明: CycN: 循环次数, 整形数值, 范围 1-99999。

注意: 延时器处于运行状态时, 无法设置参数。

举例: :DELAY:CYCLEs 1

:DELAY:CYCLEs?

功能: 查询当前通道的延时器的循环次数。

说明: 返回 "CycN: 循环次数, 整形数值。"

举例: :DELAY:CYCLEs?

返回: 1

:DELAY:ENDState { ON|OFF|LAST}

功能: 设置当前通道的延时器的停止状态。

说明: ON: 延时器停止时, 打开输出。

OFF: 延时器停止时, 关闭输出。

LAST: 延时器停止时, 保持最后的输出状态。

注意: 延时器处于运行状态时, 无法设置参数。

举例: :DELAY:ENDState OFF

:DELAY:ENDState?

功能: 查询当前通道的延时器的停止状态。

说明: 返回 "{ ON|OFF|LAST }"

ON: 延时器停止时, 打开输出。

OFF: 延时器停止时, 关闭输出。

LAST: 延时器停止时, 保持最后的输出状态。

举例: :DELAY:ENDState?

返回: OFF

:DELAY:STOP {NONE|<V>>V|<C>>C|<P>>P} [,<value>]

功能: 设置当前通道的延时器的停止条件。

说明: 延时器运行时, 满足设置的停止条件后、延时器将自动停止。省略<value>参数时, 只设置判断条件、不改变原有判断值。

注意: 延时器处于运行状态时, 无法设置参数。

举例: :DELAY:STOP >V, 10.00

:DELAY:STOP?

功能： 查询当前通道的延时器的停止条件。

说明： 返回“ (NONE|<V|>V|<C|>C|<P|>P [,<value>])” ， 返回的第一个参数为“NONE” 时， 将没有第二个参数。

举例： :DELAY:STOP?

返回： >V,10.000

:DELAY:PARAMeter <index>, {ON|OFF}, <time>

功能： 设置当前通道的延时器的组参数。

说明： index: 需要设置参数的组的组序号， 范围 0-2047。

ON|OFF: 该组的输出状态。

time: 该组的运行时长， 单位为秒。

注意： 延时器处于运行状态时， 无法设置参数。

举例： :DELAY:PARAMeter 0, ON, 10

:DELAY:PARAMeter? <index> [,<count>]

功能： 查询当前通道的延时器的组参数。

说明： index: 从此序号的组开始查询设置参数， 范围 0-2047。

count: 需要查询的组的个数， 范围 1-10。省略此参数时， 默认查询 1 组。

返回格式： 以数据块格式返回数据， 数据段为各组的序号、输出开关状态、时间。如：“#180,ON,10;” ， #18 表示数据段有 8 个数据——“0,ON,10;” ， 数据内容显示组号 0， 输出打开， 时间 10 秒。

举例： :DELAY:PARAMeter? 0

返回： #180,ON,10;

:DELAY:GENerate:STAT < index>, <point>, {01P|10P}

功能： 按状态生成模式自动生成延时器的组参数。

说明： 仪器接收到此命令后， 按状态生成模式生成延时器的组参数， 未指定的组参数不受影响。

index: 第一个点的组号， 从此点开始生成；

point: 需要生成的点数；

01P|10P: 指定按 01 码或 10 码生成；

注意： 延时器处于运行状态时， 不可使用此命令。

举例： :DELAY:GENerate:STAT 0, 10, 01P

:DELAY:GENerate:FIX< index>, <point>, <time_on>, <time_off>

功能: 按固定时间生成模式自动生成定时器的组参数。

说明: 仪器接收到此命令后, 按固定时间生成模式生成定时器的组参数, 未指定的组参数不受影响。

index: 第一个点的组号, 从此点开始生成;

point: 需要生成的点数;

time_on: 输出状态为打开的组时, 需要运行的时长;

time_off: 输出状态为关闭的组时, 需要运行的时长;

注意: 定时器处于运行状态时, 不可使用此命令。

举例: :DELAY:GENerate:FIX 0, 10, 5, 10

:DELAY:GENerate:INC< index>, <point>, <time_base>, <time_step>

功能: 按单调上升生成模式自动生成定时器的组参数。

说明: 仪器接收到此命令后, 按单调上升生成模式生成定时器的组参数, 未指定的组参数不受影响。

index: 第一个点的组号, 从此点开始生成;

point: 需要生成的点数;

time_base: 时间基值;

time_step: 步进值;

注意: 定时器处于运行状态时, 不可使用此命令。

举例: :DELAY:GENerate:INC 0, 10, 10, 2

:DELAY:GENerate:DEC < index>, <point>, <time_base>, <time_step>

功能: 按单调下降生成模式自动生成定时器的组参数。

说明: 仪器接收到此命令后, 按单调下降生成模式生成定时器的组参数, 未指定的组参数不受影响。

index: 第一个点的组号, 从此点开始生成;

point: 需要生成的点数;

time_base: 时间基值;

time_step: 步进值;

注意: 定时器处于运行状态时, 不可使用此命令。

举例: :DELAY:GENerate:DEC 0, 10, 100, 1

:DELAY:GENerate?

功能: 查询当前通道的定时器的自动生成参数。

说明: 返回: "STAT, <index>,<point> , {01P|10P}" 或
"FIX, <index>,<point> , <time_on>, <time_off>" 或
"{INC|DEC}, <index>,<point> , <time_base>, <time_step>"

举例: :DELAY:GENerate?

返回: DEC,0,10,100,1

8. PRESet

:PRESet#[:APPLy]

功能：将指定组的预设值参数应用到输出设置参数。

说明：#的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。

举例：:PRESet1:APPLy

:PRESet#:SET:VOLTage {CH1|CH2|CH3|SER|PARA}, {<volt>|MINimum|MAXimum}

功能：设置预设值组的电压值。

说明：#的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。

举例：:PRESet1:SET:VOLTage CH1, 5.00

:PRESet#:SET:VOLTage? {CH1|CH2|CH3|SER|PARA}

功能：查询预设值组的电压值。

说明：#的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。返回电压的实型数值。

举例：:PRESet1:SET:VOLTage? CH1

返回: 05.00

:PRESet#:SET:CURRent{CH1|CH2|CH3|SER|PARA}, {<curr>|MINimum|MAXimum}

功能：设置预设值组的电流值。

说明：#的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。

举例：:PRESet1:SET:CURRent CH1, 1.258

:PRESet#:SET:CURRent? {CH1|CH2|CH3|SER|PARA}

功能：查询预设值组的电流值。

说明：#的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。返回电流的实型数值。

举例：:PRESet1:SET:CURRent? CH1

返回: 1.258

:PRESet#:SET:OVP {CH1|CH2|CH3|SER|PARA}, {0|1|OFF|ON} [, {<volt>|MINimum|MAXimum}]

功能：设置预设值组的过压保护开关、过压保护值。

说明：#的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。省略{<volt>|MINimum|MAXimum}参数时，只设置过压开关、不修改保护值。

举例：:PRESet1:SET:OVP CH1, OFF

:PRESet1:SET:OVP CH1, ON

:PRESet1:SET:OVP CH1, OFF, 20.00

:PRESet1:SET:OVP CH1, ON, 15.00

:PRESet#:SET:OVP? {CH1|CH2|CH3|SER|PARA}

功能: 查询预设值组的过压保护开关、过压保护值。

说明: #的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。返回“{OFF|ON}, <value>”。

OFF|ON: 过压保护的开关状态;

<value>: 实型数值，过压保护值。

举例: :PRESet1:SET:OVP? CH1

返回: ON,15.000

:PRESet#:SET:OCP {CH1|CH2|CH3|SER|PARA}, {0|1|OFF|ON} [, {<curr>|MINimum|MAXimum}]

功能: 设置预设值组的过流保护开关、过流保护值。

说明: #的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。省略{<curr>|MINimum|MAXimum}参数时，只设置过流开关、不修改保护值。

举例: :PRESet1:SET:OCP CH1, OFF

:PRESet1:SET:OCP CH1, ON

:PRESet1:SET:OCP CH1, OFF, 2.000

:PRESet1:SET:OCP CH1, ON, 1.500

:PRESet#:SET:OCP? {CH1|CH2|CH3|SER|PARA}

功能: 查询预设值组的过流保护开关、过流保护值。

说明: #的可选值为“1|2|3|4|5”、不可省略，对应相应的预设值组。返回“{OFF|ON}, <value>”。

OFF|ON: 过流保护的开关状态;

<value>: 实型数值，过流保护值。

举例: :PRESet1:SET:OCP? CH1

返回: ON,1.500

9. MONItor

:MONItor[:STATe] {0|1|OFF|ON}

功能：启动或停止当前通道的监测器。

说明：可通过“INSTRument”章节的指令设置和查询当前通道。

举例： :MONItor:STATe ON

:MONItor[:STATe]?

功能：查询当前通道的监测器状态。

说明：返回 “{ON|OFF}”。

举例： :MONItor:STATe?

返回： ON

:MONItor:VOLTage {<V>>V|NONE} [, <volt>|MINimum|MAXimum]

功能：设置当前通道的监测条件中的电压条件。

说明：省略第二个参数时，只设置判断条件、不改变原有的判断值。

<V>：启用电压判断，当电压小于判断值时返回“真”；

>V：启用电压判断，当电压大于判断值时返回“真”；

NONE：禁用电压判断，不可以同时禁用电压判断、电流判断、功率判断；

<volt>：实型数值，电压判断值。

举例： :MONItor:VOLTage >V, 15.58

:MONItor:VOLTage?

功能：查询当前通道的监测条件中的电压条件。

说明：返回 “{<V>>V|NONE}, <volt>”；<volt>：实型数值，电压判断值。

举例： :MONItor:VOLTage?

返回： >V,15.58

:MONItor:CURRent {<C>>C|NONE} [, <curr>|MINimum|MAXimum]

功能：设置当前通道的监测条件中的电流条件。

说明：省略第二个参数时，只设置判断条件、不改变原有的判断值。

<C>：启用电流判断，当电流小于判断值时返回“真”；

>C：启用电流判断，当电流大于判断值时返回“真”；

NONE：禁用电流判断，不可以同时禁用电压判断、电流判断、功率判断；

<curr>：实型数值，电流判断值。

举例： :MONItor:CURRent >C, 3.555

:MONItor:CURRent?

功能: 查询当前通道的监测条件中的电流条件。

说明: 返回 “{<C|>C|NONE}, <curr>” ; <curr>: 实型数值, 电流判断值。

举例: :MONItor:CURRent?

返回: >C,3.555

:MONItor:POWER {<P|>P|NONE} [, <watt>|MINimum|MAXimum]

功能: 设置当前通道的监测条件中的功率条件。

说明: 省略第二个参数时, 只设置判断条件、不改变原有的判断值。

<P: 启用功率判断, 当功率小于判断值时返回 “真”;

>P: 启用功率判断, 当功率大于判断值时返回 “真”;

NONE: 禁用功率判断, 不可以同时禁用电压判断、电流判断、功率判断;

<watt>: 实型数值, 功率判断值。

举例: :MONItor:POWER >P, 60.00

:MONItor:POWER?

功能: 查询当前通道的监测条件中的功率条件。

说明: 返回 “{<P|>P|NONE}, <watt>” ; <watt>: 实型数值, 功率判断值。

举例: :MONItor:POWER?

返回: >P,60.00

:MONItor:LOGic {1|2}, {AND|OR}

功能: 设置当前通道的监测条件中的逻辑符号。

说明: {1|2}: 1 指定设置第一个逻辑符号; 2 指定设置第二个逻辑符号;

AND: 设置指定的逻辑符号为逻辑 “与”;

OR: 设置指定的逻辑符号为逻辑 “或”。

举例: :MONItor:LOGic 1, AND

:MONItor:LOGic 2, OR

:MONItor:LOGic? {1|2}

功能: 查询当前通道的监测条件中的逻辑符号。

说明: {1|2}: 1 指定查询第一个逻辑符号; 2 指定查询第二个逻辑符号;

返回: “{AND|OR}” 。

举例: :MONItor:LOGic? 1

返回: AND

:MONItor:LOGic? 2

返回: OR

:MONItor:STOPway {OUTOFF|MSG|BEEPER}, {ON|OFF}

功能：设置当前通道的监测器的停止处理方式。

说明： **OUTOFF：** 设置是否关闭输出, **ON：** 关闭输出, **OFF：** 不关闭输出;
MSG： 设置是否弹出消息提示, **ON：** 弹出消息, **OFF：** 不弹出消息;
BEEPER： 设置是否发出蜂鸣声音, **ON：** 开启蜂鸣音, **OFF：** 关闭蜂鸣音。

举例： :MONItor:STOPway OUTOFF, ON 监测条件满足时, 关闭输出
:MONItor:STOPway MSG, OFF 监测条件满足时, 不弹出消息提示
:MONItor:STOPway BEEPER, ON 监测条件满足时, 蜂鸣器发出声音

:MONItor:STOPway?

功能：查询当前通道的监测器的停止处理方式。

说明：返回: "OutputOff:{ON|OFF}, Msg:{ON|OFF}, Beep:{ON|OFF}" 。

举例： :MONItor:STOPway?

返回: OutputOff:ON,Msg:OFF,Beep:ON

10. TRIGger

:TRIGger:IN[:ENABLE] {D0|D1|D2|D3},{0|1|OFF|ON}

功能: 启动或停止输入模式触发器; 如果触发器当前处于输出模式, 触发器会切换到输入模式。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

举例: :TRIGger:IN:ENABLE D0, OFF 停止 IO1 的触发器, IO1 作为输入模式
:TRIGger:IN:ENABLE D0, ON 启用 IO1 的触发器, IO1 作为输入模式

:TRIGger:IN[:ENABLE]? {D0|D1|D2|D3}

功能: 查询输入模式触发器开启状态。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

当相应 IO 处于输出模式时, 一定返回 "OFF" ;

当相应 IO 处于输入模式时, 根据触发器的开启状态返回 "OFF" 或 "ON" 。

举例: :TRIGger:IN:ENABLE? D0

返回: ON

:TRIGger:IN:SOURce {D0|D1|D2|D3}, {CH1|CH2|CH3|SER|PARA}[,{CH1|CH2|CH3|SER|PARA}] [{CH1|CH2|CH3|SER|PARA}]

功能: 设置输入模式触发器的受控源, 当输入 IO 产生触发信号时, 受控源按设置的响应方式执行动作。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

CH1: 通道一; CH2: 通道二; CH3: 通道三; SER: 串联通道; PARA: 并联通道。

后 3 个参数中, CH1\CH2 不可与 SER\PARA 同时出现, SER 与 PARA 也不可同时出现。

举例: :TRIGger:IN:SOURce D0, CH1, CH2, CH3 设置 IO1 的受控源为 CH1 和 CH2 和 CH3
:TRIGger:IN:SOURce D0, CH1, CH3 设置 IO1 的受控源为 CH1 和 CH3
:TRIGger:IN:SOURce D0, CH3, SER 设置 IO1 的受控源为 CH3 和 SER(串联通道)
:TRIGger:IN:SOURce D0, PARA 设置 IO1 的受控源为 PARA(并联通道)

:TRIGger:IN:SOURce? {D0|D1|D2|D3}

功能: 查询输入模式触发器的受控源。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回所有受控通道的组合, 如: "CH2,CH3", 返回了两个受控通道, 通道二和通道三。

CH1: 通道一; CH2: 通道二; CH3: 通道三; SER: 串联通道; PARA: 并联通道。

举例: :TRIGger:IN:SOURce? d0

返回: CH2,CH3

:TRIGger:IN:TYPE {D0|D1|D2|D3},{RISE|FALL|HIGH|LOW}

功能: 设置输入模式触发器的触发信号的类型。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

RISE: 上升沿信号;

FALL: 下降沿信号;

HIGH: 高电平信号;

LOW: 低电平信号。

举例: :TRIGger:IN:TYPE D0, FALL 设置 IO1 的触发输入信号类型为下降沿

:TRIGger:IN:TYPE? {D0|D1|D2|D3}

功能: 查询输入模式触发器的触发信号的类型。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回 "{RISE|FALL|HIGH|LOW}" 。

RISE: 上升沿信号;

FALL: 下降沿信号;

HIGH: 高电平信号;

LOW: 低电平信号。

举例: :TRIGger:IN:TYPE? D0 查询 IO1 的触发输入信号的类型

返回: FALL 返回结果: 下降沿

:TRIGger:IN:SENSitivity {D0|D1|D2|D3},{LOW|MID|HIGH}

功能: 设置输入模式触发器的触发信号的灵敏度。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

LOW: 低灵敏度;

MID: 中灵敏度;

HIGH: 高灵敏度;

举例: :TRIGger:IN:SENSitivity D0, MID 设置 IO1 的触发输入信号灵敏度为中等灵敏度

:TRIGger:IN:SENSitivity? {D0|D1|D2|D3}

功能: 查询输入模式触发器的触发信号的灵敏度。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回 "{LOW|MID|HIGH}" 。

LOW: 低灵敏度;

MID: 中灵敏度;

HIGH: 高灵敏度;

举例: :TRIGger:IN:SENSitivity? D0 查询 IO1 的触发输入信号的灵敏度

返回: MID 返回结果: 中等灵敏度

:TRIGger:IN:RESPonse {D0|D1|D2|D3},{ON|OFF|ALTER}

功能: 设置输入模式触发器的触发响应, 输入触发事情产生时、触发器执行触发响应。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

ON: 输入触发事情产生时、触发器打开受控通道的输出;

OFF: 输入触发事情产生时、触发器关闭受控通道的输出;

ALTER: 输入触发事情产生时、触发器翻转受控通道的输出。

举例: :TRIGger:IN:RESPonse D0, OFF 设置 IO1 的输入触发响应为关闭输出

:TRIGger:IN:RESPonse? {D0|D1|D2|D3}

功能: 查询输入模式触发器的触发响应。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回 "{ON|OFF|ALTER}", 代表输入触发事情产生时、触发器执行的触发响应。

ON: 输入触发事情产生时、触发器打开受控通道的输出;

OFF: 输入触发事情产生时、触发器关闭受控通道的输出;

ALTER: 输入触发事情产生时、触发器翻转受控通道的输出。

举例: :TRIGger:IN:RESPonse? D0 查询 IO1 的触发响应

返回: OFF 返回结果: 输入触发事情产生时、触发器关闭受控通道的输出

:TRIGger:OUT[:ENABLE] {D0|D1|D2|D3},{0|1|OFF|ON}

功能: 启动或停止输出模式触发器; 如果触发器当前处于输入模式, 触发器会切换到输出模式。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

举例: :TRIGger:OUT:ENABLE D0, OFF 停止 IO1 的触发器, IO1 作为输出模式

:TRIGger:OUT:ENABLE D0, ON 启用 IO1 的触发器, IO1 作为输出模式

:TRIGger:OUT[:ENABLE]? {D0|D1|D2|D3}

功能: 查询输出模式触发器开启状态。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

当相应 IO 处于输入模式时, 一定返回 "OFF";

当相应 IO 处于输出模式时, 根据触发器的开启状态返回 "OFF" 或 "ON" 。

举例: :TRIGger:OUT:ENABLE? D0

返回: ON

:TRIGger:OUT:SOURce {D0|D1|D2|D3},{CH1|CH2|CH3|SER|PARA}

功能: 设置输出模式触发器的控制源, 当控制源满足触发条件时, IO 按设置的输出信号输出响应的信号。

说明: D0|D1|D2|D3: 选择输出 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

只能选择一个通道作为控制源。

CH1: 通道一; CH2: 通道二; CH3: 通道三; SER: 串联通道; PARA: 并联通道。

举例: :TRIGger:OUT:SOURce D0, CH1 设置 CH1 为 IO1 的控制通道

:TRIGger:OUT:SOURce? {D0|D1|D2|D3},

功能: 查询输出模式触发器的控制源。

说明: D0|D1|D2|D3: 选择输出 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回: {CH1|CH2|CH3|SER|PARA}。

CH1: 通道一; CH2: 通道二; CH3: 通道三; SER: 串联通道; PARA: 并联通道。

举例: :TRIGger:OUT:SOURce? D0 查询 IO1 的控制通道

返回: CH1 返回结果: CH1 为 IO1 的控制通道

:TRIGger:OUT:CONDition {D0|D1|D2|D3},{AUTO|OUTOFF|OUTON}>V|<V|=V|>C|<C|=C|>P|<P|=P}[,<value>]

功能: 设置输出模式触发器的触发条件, 当控制源满足触发条件时, IO 按设置的输出信号输出响应的信号。

说明: D0|D1|D2|D3: 选择输出 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

AUTO: 自动触发, 此时仪器总是自动产生触发条件; 不可带有 <value> 参数;

OUTOFF: 控制源关闭输出时, 产生触发条件; 不可带有 <value> 参数;

OUTON: 控制源打开输出时, 产生触发条件; 不可带有 <value> 参数;

>V: 控制源电压大于判断值时, 产生触发条件; 不可省略 <value> 参数;

<V: 控制源电压小于判断值时, 产生触发条件; 不可省略 <value> 参数;

=V: 控制源电压等于判断值时, 产生触发条件; 不可省略 <value> 参数;

>C: 控制源电流大于判断值时, 产生触发条件; 不可省略 <value> 参数;

<C: 控制源电流小于判断值时, 产生触发条件; 不可省略 <value> 参数;

=C: 控制源电流等于判断值时, 产生触发条件; 不可省略 <value> 参数;

>P: 控制源功率大于判断值时, 产生触发条件; 不可省略 <value> 参数;

<P: 控制源功率小于判断值时, 产生触发条件; 不可省略 <value> 参数;

=P: 控制源功率等于判断值时, 产生触发条件; 不可省略 <value> 参数。

举例: :TRIGger:OUT:CONDition D0,>V,30.00 设置 IO1 的触发条件为: 电压大于 30.00 伏

:TRIGger:OUT:CONDition {D0|D1|D2|D3}

功能: 查询输出模式触发器的触发条件。

说明: D0|D1|D2|D3: 选择输出 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回: "{AUTO|OUTOFF|OUTON}>V|<V|=V|>C|<C|=C|>P|<P|=P} [,<value>]" 。

返回 "AUTO|OUTOFF|OUTON" 时, 不带有 <value> 参数;

返回 ">V|<V|=V|>C|<C|=C|>P|<P|=P" 时, 带有 <value> 参数。

举例: :TRIGger:OUT:CONDition? D0 查询 IO1 的输出触发条件

返回: >V,30.00 返回结果: 触发条件为-电压大于 30.00 伏

:TRIGger:OUT:POLARity {D0|D1|D2|D3},{POSItive|NEGAtive}

功能: 设置输出触发器的输出信号的极性。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

POSItive: 输出正极性信号;

NEGAtive: 输出负极性信号;

举例: :TRIGger:OUT:POLARity D0, POSItive 设置 IO1 的输出信号为正极性

:TRIGger:OUT:POLARity? {D0|D1|D2|D3}

功能: 查询输出触发器的输出信号的极性。

说明: D0|D1|D2|D3: 选择触发 IO, D0-IO1, D1-IO2, D2-IO3, D3-IO4。

返回: “{POSITIVE|NEGATIVE}”。

POSITIVE: 输出信号为正极性;

NEGATIVE: 输出信号为负极性。

举例: :TRIGger:OUT:POLARity? D0 查询 IO1 的输出信号极性

返回: POSITIVE 返回结果: IO1 的输出信号为正极性

11. SYSTem

:SYSTem:BEEPer[:STATe] {0|1|OFF|ON}

功能: 打开或关闭操作蜂鸣器提示音。

举例: :SYSTem:BEEPer:STATe OFF 关闭蜂鸣提示音

:SYSTem:BEEPer[:STATe]?

功能: 查询操作蜂鸣器提示音的开关状态。

说明: 返回 "{ON|OFF}" 。

举例: :SYSTem:BEEPer:STATe? 查询蜂鸣提示音开关状态
返回: OFF 返回结果: 蜂鸣提示音为关闭状态

:SYSTem:COMMunicate:LAN:APPLY

功能: 应用已设置的网络参数。

说明: 使用 ":SYSTem:COMMunicate:LAN:xxxx" (xxxx 代表 LAN 下的其它网络设置命令)命令设置网络参数后, 相关设置参数不会立即生效、也不会被存入设备的非易失性存储器, 必须执行 ":SYSTem:COMMunicate:LAN:APPLY" 命令使参数生效和永久存储。可以执行多条命令修改多个参数后, 共同执行一条 ":SYSTem:COMMunicate:LAN:APPLY" 命令。

举例: :SYSTem:COMMunicate:LAN:DHCP:STATe ON 打开 DHCP 功能
:SYSTem:COMMunicate:LAN:APPLY 应用已设置的参数

:SYSTem:COMMunicate:LAN:DHCP[:STATe] {0|1|OFF|ON}

功能: 打开或关闭网络的 DHCP 功能。

说明: 使用此命令设置网络参数后, 设置参数不会立即生效、也不会被存入设备的非易失性存储器; 必须使用 ":SYSTem:COMMunicate:LAN:APPLY" 命令使参数生效和永久存储。可以执行多条命令修改多个参数后, 共同执行一条 ":SYSTem:COMMunicate:LAN:APPLY" 命令。

举例: :SYSTem:COMMunicate:LAN:DHCP:STATe ON 打开 DHCP 功能
:SYSTem:COMMunicate:LAN:APPLY 应用已设置的参数

:SYSTem:COMMunicate:LAN:DHCP[:STATe]?

功能: 查询网络的 DHCP 功能的开关状态。

说明: 在设置过网络参数后、并且在执行 ":SYSTem:COMMunicate:LAN:APPLY" 命令以前, 使用此命令查询到的结果是设置的临时参数, 反应的是执行 ":SYSTem:COMMunicate:LAN:APPLY" 命令时使用到的参数数据。否则, 使用此命令查询到的结果是设备正在运行使用的参数。

返回 "{ON|OFF}" 。

举例: :SYSTem:COMMunicate:LAN:DHCP:STATe?
返回: ON

:SYSTem:COMMunicate:LAN:IPADdress "x.x.x.x"

功能: 设置设备的网络 IP 地址。

说明: 使用此命令设置网络参数后, 设置参数不会立即生效、也不会被存入设备的非易失性存储器; 必须使用 ":SYSTem:COMMunicate:LAN:APPLy" 命令使参数生效和永久存储。可以执行多条命令修改多个参数后, 共同执行一条 ":SYSTem:COMMunicate:LAN:APPLy" 命令。

举例: :SYSTem:COMMunicate:LAN:IPADdress "192.168.10.142" 设置 IP 为 192.168.10.142
:SYSTem:COMMunicate:LAN:APPLy 应用已设置的参数

:SYSTem:COMMunicate:LAN:IPADdress?

功能: 查询设备的网络 IP 地址。

说明: 在设置过网络参数后、并且在执行 ":SYSTem:COMMunicate:LAN:APPLy" 命令以前, 使用此命令查询到的结果是设置的临时参数, 反应的是执行 ":SYSTem:COMMunicate:LAN:APPLy" 命令时使用到的参数数据。否则, 使用此命令查询到的结果是设备正在运行使用的参数。

返回类似: "192.168.10.142" 。

举例: :SYSTem:COMMunicate:LAN:IPADdress?

返回: "192.168.10.142"

:SYSTem:COMMunicate:LAN:SMASK "x.x.x.x"

功能: 设置设备的网络子网掩码。

说明: 使用此命令设置网络参数后, 设置参数不会立即生效、也不会被存入设备的非易失性存储器; 必须使用 ":SYSTem:COMMunicate:LAN:APPLy" 命令使参数生效和永久存储。可以执行多条命令修改多个参数后, 共同执行一条 ":SYSTem:COMMunicate:LAN:APPLy" 命令。

举例: :SYSTem:COMMunicate:LAN:SMASK "255.255.255.0" 设置子网掩码为 255.255.255.0
:SYSTem:COMMunicate:LAN:APPLy 应用已设置的参数

:SYSTem:COMMunicate:LAN:SMASK?

功能: 查询设备的网络子网掩码。

说明: 在设置过网络参数后、并且在执行 ":SYSTem:COMMunicate:LAN:APPLy" 命令以前, 使用此命令查询到的结果是设置的临时参数, 反应的是执行 ":SYSTem:COMMunicate:LAN:APPLy" 命令时使用到的参数数据。否则, 使用此命令查询到的结果是设备正在运行使用的参数。

返回类似: "255.255.255.0" 。

举例: :SYSTem:COMMunicate:LAN:SMASK?

返回: "255.255.255.0"

:SYSTem:COMMunicate:LAN:GATEway "x.x.x.x"

功能: 设置设备的网络网关。

说明: 使用此命令设置网络参数后, 设置参数不会立即生效、也不会被存入设备的非易失性存储器; 必须使用 ":SYSTem:COMMunicate:LAN:APPLY" 命令使参数生效和永久存储。可以执行多条命令修改多个参数后, 共同执行一条 ":SYSTem:COMMunicate:LAN:APPLY" 命令。

举例: :SYSTem:COMMunicate:LAN:GATEway "192.168.10.1" 设置子网掩码为 192.168.10.1
:SYSTem:COMMunicate:LAN:APPLY 应用已设置的参数

:SYSTem:COMMunicate:LAN:GATEway?

功能: 查询设备的网络网关。

说明: 在设置过网络参数后、并且在执行 ":SYSTem:COMMunicate:LAN:APPLY" 命令以前, 使用此命令查询到的结果是设置的临时参数, 反应的是执行 ":SYSTem:COMMunicate:LAN:APPLY" 命令时使用到的参数数据。否则, 使用此命令查询到的结果是设备正在运行使用的参数。

返回类似: "192.168.10.1" 。

举例: :SYSTem:COMMunicate:LAN:GATEway?
返回: "192.168.10.1"

:SYSTem:COMMunicate:RS232:BAUD {4800|7200|9600|14400|19200|38400|57600|115200|128000}

功能: 设置 RS232 接口的波特率。

举例: :SYSTem:COMMunicate:RS232:BAUD 9600 设置 RS232 的波特率为 9600

:SYSTem:COMMunicate:RS232:BAUD?

功能: 查询 RS232 接口的波特率。

说明: 返回 "{4800|7200|9600|14400|19200|38400|57600|115200|128000}"

举例: :SYSTem:COMMunicate:RS232:BAUD? 查询 RS232 的波特率为 9600
返回: 9600 返回结果: 波特率为 9600

:SYSTem:BRIGhtness <value>

功能: 设置 LCD 的背光亮度。

说明: value, 整型数值, 范围 1 至 100。

举例: :SYSTem:BRIGhtness 80 设置 LCD 的背光亮度为 80%

:SYSTem:BRIGhtness?

功能: 查询 LCD 的背光亮度。

说明: 返回 < value >, 整型数值, 范围 1 至 100。

举例: :SYSTem:BRIGhtness? 查询 LCD 的背光亮度
返回: 80 返回结果: LCD 的背光亮度 80%