

# 9321 系列耐压测试仪 编程手册

1	命令介绍 .....	1
1.1	符号的约定和说明 .....	1
1.2	命令及参数的缩写 .....	1
2	命令系统 .....	2
2.1	公用命令 .....	3
2.2	显示系统命令 .....	6
2.3	启动停止系统命令 .....	6
2.4	查询系统命令 .....	7
2.5	测试步骤系统命令 .....	11
2.6	AC 交流耐压系统命令 .....	12
2.7	DC 直流耐压系统命令 .....	14
2.8	IR 绝缘电阻系统命令 .....	17
2.9	OS 开短路系统命令 .....	20
2.10	PA 系统命令 .....	21
2.11	PRESET 测量配置系统命令 .....	22
2.12	保存/调出系统命令 .....	25
2.13	按键系统命令 .....	26
2.14	系统配置系统命令 .....	26
3	出错信息 .....	29
4	编程实例 .....	30
4.1	Visual C++ 6.0 编程实例 .....	32
4.2	Visual Basic 6.0 编程实例 .....	37
4.3	LabVIEW 8.5 编程实例 .....	40

编程手册的目的是使您利用我们现有的指令对仪器进行编程操作。主要的内容包括符号的约定和说明、命令及参数的缩写、主要命令的介绍和索引。

您可以通过这些指令控制仪器工作或者进行二次开发。

## 1 命令介绍

### 1.1 符号的约定和说明

- 冒号: 代表命令的层次, 表示进入命令的下一层。
- 问号? 表示查询命令的执行状态。
- 分号; 表示开始多重命令。
- 星号\* 星号后的命令是公用命令。
- 逗号, 逗号是多参数的分隔符。
- 空格 空格是命令和参数的分隔符。
- 尖括号<> 尖括号包含的字符表示程序代码参数。
- 方括号[] 方括号表示包含的项目是可选的。
- 大括号{} 大括号表示当包含几个项目时, 只能从几个项目中选择一个。
- NR1 整数, 例如: 12。
- NR2 定点数, 例如: 12.3
- NR3 浮点数, 例如: 2.000000e-03。
- NL 表示换行符, ACSII 码是 10, 是字符输入输出的结束符。
- 注: 每个命令串后面必须加上 NL(ASCII 码是 10)作为命令结束符。**

### 1.2 命令及参数的缩写

为了方便记忆和书写长格式的命令或参数, 我们根据下面的规则, 对长格式的命令或参数进行缩写。

如果命令或参数的字符数小于等于 4 个时, 不缩写。

例如: TYPE 缩写为 TYPE。

如果命令或参数的字符数大于 4 个时, 分两种情况考虑:

1. 如果第 4 个字符是元音, 则缩写取前 3 个字符。
2. 如果第 4 个字符不是元音, 则缩写取前 4 个字符。

例如: LIMit 缩写为 LIM。

DISPlay 缩写为 DISP。

如果命令或参数是由两个及以上单词组成的, 先按照取用第一个单词的首字符和最后一个单词的全部的原则, 取出长格式; 然后再对长格式使用上面的缩写规则进行缩写。

例如: SaveTYPe 长格式为 STYPe, 缩写为 STYP。

## 2 命令系统

9320 系列耐压测试仪的命令系统主要包括下列系统命令：

- ◆公用命令
- ◆显示系统命令
- ◆启动停止系统命令
- ◆查询系统命令
- ◆测试步骤系统命令
- ◆AC 交流耐压系统命令
- ◆DC 直流耐压系统命令
- ◆IR 绝缘电阻系统命令
- ◆OS 开短路系统命令
- ◆PA 系统命令
- ◆PRESET 测量配置系统命令
- ◆保存/调出系统命令
- ◆按键系统命令
- ◆系统配置系统命令

## 2.1 公用命令

公用命令是由 IEEE488.2-1987 标准定义的，也是本仪器命令系统中最基本的命令。它既可以与其它指令组成指令集，也可以单独完成特定的功能。

本仪器的指令系统中用到的公用指令如表 2-1-1 所示。

命令	查询	查询返回
N/A	*IDN?	Eucol Electronic Technology Co.,Ltd.,<仪器型号>,<仪器序列号>,<仪器软件版本>
*RST	N/A	N/A
*RCL <value>	N/A	N/A
*SAV <value>	N/A	N/A
*CLS	N/A	N/A
*ESE <0-255>	*ESE?	事件状态使能寄存器
N/A	*ESR?	事件状态寄存器
*OPC	*OPC?	返回1
*SRE <0-255>	*SRE?	服务请求使能寄存器
N/A	*STB?	服务请求寄存器

### 1. \*IDN?

返回仪器信息，包括公司名称、型号、产品序列号、软件版本号。

查询语法：\*IDN?

查询返回：Eucol Electronic Technology Co.,Ltd.,<仪器型号>,<仪器序列号>,<仪器软件版本>

### 2. \*RST (Reset)

使仪器处于稳定的状态，这里直接使仪器调用出厂设置。

命令语法：\*RST

### 3. \*RCL<value>

调出 value 指定的内部设置文件。value 范围：1~100。

命令语法：\*RCL <value>

例如：\*RCL 1 调出内部存储器位置1处的设置文件

### 4. \*SAV<value>

保存 value 指定的内部设置文件。value 范围：1~100。

命令语法：\*SAV <value>

例如： \*SAV 1 将设置文件保存到内部存储器位置1处

### 5. \*CLS

清除状态寄存器，并且清除输出缓冲区和清除 OPC 位。

命令语法： \*CLS

### 6. \*ESE<0-255>

设定事件状态使能寄存器。状态使能寄存器控制状态寄存器中被使能的相应位，当状态使能寄存器中的相应位置为 1 时，使能状态寄存器的相应位。

ESE（事件状态使能寄存器）

PON		CME	EXE		QYE		OPC
-----	--	-----	-----	--	-----	--	-----

事件说明

位	名称	描述	当置为1时，使能
7	PON	上电	当由断变为通时，事件发生
5	CME	命令错误	当检测到一个命令错误时，事件发生
4	EXE	执行错误	当检测到一个执行错误时，事件发生
2	QYE	输出数据丢失	输出缓冲中有数据，但是又有命令时，事件发生
0	OPC	操作完成	当一个操作完成时，事件发生

命令语法： \*ESE <0-255>

查询语法： \*ESE?

查询返回： <NR1><NL>返回事件状态使能寄存器值

### 7. \*ESR?

对应ESE中的状态位,返回所有已使能的相应位状态信息，同时清除ESR寄存器。

ESR（事件状态寄存器）

位	名称	描述	当置为 1 时，指明
7	PON	上电	已上电
5	CME	命令错误	已检测到一个命令错误
4	EXE	执行错误	已检测到一个执行错误
2	QYE	输出数据丢失	已检测到输出数据丢失
0	OPC	操作完成	操作已完成

查询语法： \*ESR?

查询返回： <NR1><NL>对应 ESE 中的状态位,返回状态信息

## 8. \*OPC

在所有命令解析完成后,将 ESR 寄存器中的 OPC 置位。

命令语法: \*OPC

查询语法: \*OPC?

查询返回: <1><NL>

注: 接口必须等待\*OPC?返回后才能执行下一步的操作。

## 9. \*SRE<0-255>

设定服务请求使能寄存器。当置 1 时, 使能状态字节寄存器中的相应位。

SRE (服务请求使能寄存器)

		ESB	MAV				
--	--	-----	-----	--	--	--	--

事件说明

位	名称	描述	当置为1时, 使能
5	ESB	事件状态位	ESR中使能的事件产生时产生中断
4	MAV	有效信息	输出缓冲区中数据有效时产生中断

命令语法: \*SRE <0-255>

查询语法: \*SRE?

查询返回: <NR1><NL>返回服务请求使能寄存器

## 10. \*STB?

查询返回服务请求寄存器。

	RQS	ESB	MAV				
--	-----	-----	-----	--	--	--	--

事件说明

位	名称	描述	当置为1时, 指明
6	RQS	服务请求	点名中用来表明是否是服务请求源
5	ESB	事件状态位	ESR中使能的事件产生
4	MAV	有效信息	输出缓冲区中数据有效

查询语法: \*STB?

查询返回: <NR1><NL>返回服务请求寄存器值

## 2.2 显示系统命令

**DISPlay:PAGE** 设定仪器显示页面，**DISPlay:PAGE?**查询当前 LCD 屏上显示的页面。

命令语法: **DISPlay:PAGE**<page name>

<page name>具体如下:

<b>MEASurement</b>	设定显示页面至: 测量显示页面( <b>MEASurement</b> )
<b>LIST</b>	设定显示页面至: 列表显示页面( <b>LIST</b> )
<b>MSETup</b>	设定显示页面至: 测量设置页面( <b>MeasSETup</b> )
<b>MCONfig</b>	设定显示页面至: 测量配置页面( <b>MeasCONfig</b> )
<b>SYSTem</b>	设定显示页面至: 系统设置页面( <b>SYSTem</b> )
<b>INFO</b>	设定显示页面至: 系统信息显示页面( <b>INFO</b> )
<b>FLISt</b>	设定显示页面至: 文件列表显示页面( <b>File LISt</b> )

查询语法: **DISPlay:PAGE?**

查询返回: {**MEAS** | **LIST** | **MSET** | **MCON** | **SYST** | **INFO** | **FLIS**}<NL>

ⓘ注意: 如果正在测量之中, 则不会响应此命令, 即命令被忽略!

## 2.3 启动停止系统命令

**[SOURce:]SAFEty:STARt[:ONCE]**启动测试。

命令语法: **[SOURce:]SAFEty:STARt[:ONCE]** 此命令用于启动测试。

**[SOURce:]SAFEty:STARt:OFFSet GET** 启动清零。

命令语法: **[SOURce:]SAFEty:STARt:OFFSet GET**

该命令启动清零动作, 清零完成后仪器自动打开清零开关。

**[SOURce:]SAFEty:STARt:OFFSet {ON | OFF}**

命令语法: **[SOURce:]SAFEty:STARt:OFFSet{OFF | ON}**

该命令用于打开或者关闭清零开关

查询语法: **[SOURce:]SAFEty:STARt:OFFSet?**

查询返回: {0 | 1}<NL> 0 表示清零关闭, 1 表示清零打开。

**[SOURce:]SAFEty:STOP** 停止测试。

命令语法: **[SOURce:]SAFEty:STOP** 此命令用于停止测试。

**[SOURce:]SAFEty:STATus?**查询仪器运行状态。

命令语法:**[SOURce:]SAFEty:STATus?**

查询返回: {**RUNNING** | **STOPPED**}<NL>



## 2.4 查询系统命令

**[SOURCE:]SAFETY:FETCh?** 用于查询测量参数。

命令语法: [SOURCE:]SAFETY:FETCh? [<item>]{,<item>}

此命令用于查询测量参数, <item>表示需要查询的测量参数。

此命令查询参数回传数据如下列表格所示:

Item 字符数据	说明
STEP	目前的测试步骤, 停止时为 0
MODE	目前的测试模式, 停止时为 NONE
OMETerage	目前 output meter 读值 (电压)
MMETerage	目前 measure meter 读值 (测量值)
RELapsed	目前已执行的电压爬升时间
RLEFt	目前未执行的电压爬升时间
TELapsed	目前已执行的测试时间
TLEFt	目前未执行的测试时间
FELapsed	目前已执行的电压下降时间
FLEFt	目前未执行的电压下降时间
FELapsed	目前已执行的 DWELL 持续时间
FLEFt	目前未执行的 DWELL 持续时间
CHANnel	目前扫描通道的状态

例如[SOURCE:]SAFETY:FETCh? STEP,MODE,OMET

仪器返回“1,AC,+5.000000E+02”, 表示查询此时 STEP,MODE 和输出值的结果是 STEP 1,AC MODE,0.500kV。

**[SOURCE:]SAFETY:RESult:ALL[:JUDGment]?**查询所有步骤的判别结果。

命令语法: [SOURCE:]SAFETY:RESult:ALL[:JUDGment]?

例如返回“116”, 表示判别结果为合格, 详见后续表格。

**[SOURCE:]SAFETY:RESult:ALL:OMETerage?**查询所有步骤 output meter 读值。

命令语法: [SOURCE:]SAFETY:RESult:ALL:OMETerage?

例如返回“+5.100000E+01”表示查询 output meter 的结果是 0.051kV。

**[SOURCE:]SAFETY:RESult:ALL:MMETerage?**查询所有步骤 measure meter 读值。

命令语法: [SOURCE:]SAFETY:RESult:ALL:MMETerage?

例如返回“+6.1200000E-05”表示查询 measure meter 的结果是 0.612mA。

**[SOURCE:]SAFETy:RESult:ALL:TIME[:TEST]?**查询所有步骤测试所需时间。

命令语法: [SOURCE:]SAFETy:RESult:ALL:TIME[:TEST]?

例如返回"+3.000000E+00"表示测试所需时间是 3 秒。

**[SOURCE:]SAFETy:RESult:ALL:TIME:RAMP?**查询所有步骤其测试上升至设定电压所需时间。

命令语法: [SOURCE:]SAFETy:RESult:ALL:TIME:RAMP?

例如返回"+1.000000E+00"表示其测试上升至设定电压所需时间是 3 秒。

**[SOURCE:]SAFETy:RESult:ALL:TIME:FALL?**查询所有步骤其由设定电压下降至零所需时间。

命令语法: [SOURCE:]SAFETy:RESult:ALL:TIME:FALL?

例如返回"+1.000000E+00"表示其由设定电压下降至零所需时间是 3 秒。

**[SOURCE:]SAFETy:RESult:ALL:TIME:DWELI?**查询所有步骤测试的 DWELL 所需时间。

命令语法: [SOURCE:]SAFETy:RESult:ALL:TIME:DWELI?

例如返回"+2.500000E+00"表示测试的 DWELL 所需时间是 2.5 秒。

**[SOURCE:]SAFETy:RESult:ALL:MODE?**查询所有步骤的测试模式。

命令语法: [SOURCE:]SAFETy:RESult:ALL:MODE?

查询返回: {AC | DC | IR | PA | OS}<NL>。

例如返回 DC 时表示当前 MODE 设定的是直流耐压 DC 模式。

**[SOURCE:]SAFETy:RESult:COMPLete?** 查询仪器是否执行完所有设定的测试步骤。

命令语法:[SOURCE:]SAFETy:RESult:COMPLete?

查询返回: {0 | 1}<NL>,1 表示完成, 0 表示未完成。

**[SOURCE:]SAFETy:RESult:AREPort[:JUDGment][:MESSAge]**设定测试结束后是否自动返回测试结果。

命令语法:[SOURCE:]SAFETy:RESult:AREPort[:JUDGment][:MESSAge] {{1 | ON}| {0 | OFF}}

当设定为 ON 或者 1 时, 测试完成后返回信息"PASS"或"FAIL"字符串数据, 当设定为 OFF 或者 0 时, 不自动返回结果。

查询语法: [SOURCE:]SAFETy:RESult:AREPort[:JUDGment][:MESSAge]?

查询返回: {1 | 0}<NL>

**[SOURCE:]SAFETy:RESult:AREPort:OMETerage** 设定测试结束后是否自动返回 OUTPUT METER 测试结果。

命令语法:[SOURCE:]SAFETy:RESult:AREPort:OMETerage {{1 | ON}| {0 | OFF}}

当设定为 ON 或者 1 时，测试完成后返回所有步骤的 OUTPUT METER 测试结果，如果其中某些步骤并未测试，则这些步骤返回+9.910000E+37；当设定为 OFF 或者 0 时，不会自动返回结果。

查询语法: [SOURCE:]SAFETy:RESult:AREPort:OMETerage?

查询返回: {1 | 0}<NL>

**[SOURCE:]SAFETy:RESult:AREPort:MMETerage** 设定测试结束后是否自动返回 MEASURE METER 测试结果。

命令语法:[SOURCE:]SAFETy:RESult:AREPort:MMETerage {{1 | ON}| {0 | OFF}}

当设定为 ON 或者 1 时，测试完成后返回所有步骤的 MEASURE METER 测试结果，如果其中某些步骤并未测试，则这些步骤返回+9.910000E+37；当设定为 OFF 或者 0 时，不会自动返回结果。

查询语法: [SOURCE:]SAFETy:RESult:AREPort:MMETerage?

查询返回: {1 | 0}<NL>

**[SOURCE:]SAFETy:RESult[:LAST][:JUDGment]?**查询最后一个步骤的判别结果。

命令语法: [SOURCE:]SAFETy:RESult[:LAST][:JUDGment]?

例如返回"116"，表示最后一个步骤判别结果为合格，详见后续表格。

**[SOURCE:]SAFETy:RESult:LAST:OMETerage?**查询最后一个步骤 output meter 读值。

命令语法: [SOURCE:]SAFETy:RESult:LAST:OMETerage?

例如返回"+5.100000E+01"表示最后一个步骤 output meter 的结果是 0.051kV。

**[SOURCE:]SAFETy:RESult:LAST:MMETerage?**查询最后一个步骤 measure meter 读值。

命令语法: [SOURCE:]SAFETy:RESult:LAST:MMETerage?

例如返回"+6.1200000E-05"表示最后一个步骤 measure meter 的结果是 0.612mA。

**[SOURCE:]SAFETy:RESult:STEP<n>[:JUDGment]?**查询指定步骤 n 的判别结果。

命令语法:[SOURCE:]SAFETy:RESult:STEP<n>[:JUDGment]?

其中 n 是介于 1~99 的十进制数值，下同。

例如返回"116"，表示步骤 n 判别结果为合格，详见后续表格。

**[SOURce:]SAFEty:RESult:STEP<n>:OMETerage?** 查询指定步骤 n output meter 读值。

命令语法: [SOURce:]SAFEty:RESult:STEP<n>:OMETerage?

例如返回"+5.100000E+01"表示步骤 n output meter 的结果是 0.051kV。

**[SOURce:]SAFEty:RESult:STEP<n>:MMETerage?** 查询步骤 n measure meter 读值。

命令语法: [SOURce:]SAFEty:RESult:STEP<n>:MMETerage?

例如返回"+6.1200000E-05"表示步骤 n measure meter 的结果是 0.612mA。

#### 一般判别结果代码表

判别结果	代码（十六进制）	代码（十进制）
PASS	74	116
USER STOP	71	113
CAN NOT STOP	72	114
TESTING	73	115
STOP	70	112

#### 判别结果不良代码表

	AC MODE		DC MODE		IR MODE	
	HEX	DEC	HEX	DEC	HEX	DEC
HI	11	17	21	33	31	49
LO	12	18	22	34	32	50
ARC	13	19	23	35	33	----
CHECK LOW	----	----	25	37	----	----
TRIPPED	79	121	79	121	79	121
RANGE	16	22	26	38	36	54
VOLT LOW	16	22	26	38	36	54

## 2.5 测试步骤系统命令

**[SOURCE:]SAFETy:SNUMber?**用于查询设定的测试步骤数目。

命令语法: [SOURCE:]SAFETy:SNUMber?

查询返回: <NR1><NL>

例如返回"+2"时表示当前已经设定了 2 个测试步骤。

**[SOURCE:]SAFETy:STEP<n>:DELeTe** 删除指定步骤 n。

命令语法: [SOURCE:]SAFETy:STEP<n>:DELeTe

其中 n 是介于 1~99 的十进制数值。

**[SOURCE:]SAFETy:STEP<n>:SET?**查询指定步骤的设置值。

命令语法: [SOURCE:]SAFETy:STEP<n>:SET?

查询返回:

“1,AC,+5.000000E+01,+5.000000E-04,+8.000000E-06,+2.000000E-04,+3.000000E+00,+1.000000E+00,+2.000000E+00,+3.000000E-04,(@0),(@0),1”

表示

STEP:1,MODE:AC,VOLT:0.5kV,HIGH:0.500mA,LOW:0.008mA,ARC:2.0mA;TIME:3.0s,RAMP:1.0s,FALL:2.0s,SCAN HIGH:0,SCAN LOW:0,DUT:1

不同的测试模式返回参数不同，参数如下：

STEP,AC,VOLT,HIGH,LOW,ARC,TIME,RAMP,FALL,CHH,CHL,DUT

STEP,DC,VOLT,HIGH,LOW,ARC,TIME,RAMP,FALL,DWELL,CHH,CHL,DUT

STEP,IR,VOLT,HIGH,LOW,TIME,RAMP,FALL,RANGE,CHH,CHL,DUT

STEP,OS,SHORT,OPEN,CHH,CHL,DUT

STEP,PA,MESSAGE,UNDER TEST SIGNAL,TIME

**[SOURCE:]SAFETy:STEP<n>:MODE?**查询指定步骤 n 的测试模式。

命令语法: [SOURCE:]SAFETy:STEP<n>:MODE?

查询返回: {AC | DC | IR | OS | PA}<NL>

## 2.6 AC 交流耐压系统命令

**[SOURCE:]SAFETy:STEP<n>:AC[:LEVel]** 设定所选步骤交流耐压电压值，

**[SOURCE:]SAFETy:STEP<n>:AC[:LEVel]?** 查询所选步骤交流耐压电压值。

命令语法: **[SOURCE:]SAFETy:STEP<n>:AC[:LEVel] <level>** level 是 NR3 格式

level 范围是 50~5000，单位为 V。

查询语法: **[SOURCE:]SAFETy:STEP<n>:AC[:LEVel]?**

查询返回: <NR3><NL>

**[SOURCE:]SAFETy:STEP<n>:AC:LIMit[:HIGH]** 设定所选步骤交流耐压漏电流上限值。

**[SOURCE:]SAFETy:STEP<n>:AC:LIMit[:HIGH]?** 查询所选步骤交流耐压漏电流上限值。

命令语法: **[SOURCE:]SAFETy:STEP<n>:AC:LIMit[:HIGH]<value>** value 是 NR3 格式

查询语法: **[SOURCE:]SAFETy:STEP<n>:AC:LIMit[:HIGH]?**

查询返回: <NR3><NL>

例如:

**SAFETy:STEP1:AC:LIMit:HIGH 0.01** 设定步骤 1 其交流耐压电流上限值为 10mA。

**SAFETy:STEP1:AC:LIMit:HIGH?** 查询返回+1.000000E-2 表示步骤 1 其交流耐压电流上限值是 10mA

**[SOURCE:]SAFETy:STEP<n>:AC:LIMit:LOW** 设定所选步骤交流耐压漏电流下限值。

**[SOURCE:]SAFETy:STEP<n>:AC:LIMit:LOW?** 查询所选步骤交流耐压漏电流下限值。

命令语法: **[SOURCE:]SAFETy:STEP<n>:AC:LIMit:LOW<value>** value 是 NR3 格式

查询语法: **[SOURCE:]SAFETy:STEP<n>:AC:LIMit:LOW?**

查询返回: <NR3><NL>

例如:

**SAFETy:STEP1:AC:LIMit:LOW 0.0001** 设定步骤 1 其交流耐压电流下限值为 0.1mA。

**SAFETy:STEP1:AC:LIMit:LOW?** 查询返回+1.000000E-4 表示步骤 1 其交流耐压电流下限值是 0.1mA

**[SOURCE:]SAFETy:STEP<n>:AC:LIMit:ARC[:LEVel]** 设定所选步骤交流耐压电弧 ARC 检测值。**[SOURCE:]SAFETy:STEP<n>:AC:LIMit:ARC[:LEVel]?** 查询所选步骤交流耐压电弧 ARC 检测值

命令语法: **[SOURCE:]SAFETy:STEP<n>:AC:LIMit:ARC[:LEVel] <value>**

value 是 NR3 格式

查询语法: **[SOURCE:]SAFETy:STEP<n>:AC:LIMit:ARC[:LEVel]?**

查询返回: <NR3><NL>

**[SOURCE:]SAFEty:STEP<n>:AC:TIME[:TEST]** 设定所选步骤交流耐压测试时间。  
[SOURCE:]SAFEty:STEP<n>:AC:TIME[:TEST]? 查询所选步骤交流耐压测试时间。  
命令语法:[SOURCE:]SAFEty:STEP<n>:AC:TIME[:TEST] <time> time 是 NR3 格式  
查询语法: [SOURCE:]SAFEty:STEP<n>:AC:TIME[:TEST]?  
查询返回: <NR3><NL>

例如:

SAFEty:STEP1:AC:TIME:TEST 3 设定步骤 1 其交流耐压测试时间为 3s。  
SAFEty:STEP1:AC:TIME:TEST? 查询返回+3.000000E+00 表示其测试时间是 3s。

**[SOURCE:]SAFEty:STEP<n>:AC:TIME:RAMP** 设定所选步骤交流耐压上升时间。  
[SOURCE:]SAFEty:STEP<n>:AC:TIME:RAMP? 查询所选步骤交流耐压上升时间。  
命令语法:[SOURCE:]SAFEty:STEP<n>:AC:TIME:RAMP<time> time 是 NR3 格式  
查询语法: [SOURCE:]SAFEty:STEP<n>:AC:TIME:RAMP?  
查询返回: <NR3><NL>

例如:

SAFEty:STEP1:AC:TIME:RAMP0.2 设定步骤 1 其交流耐压上升时间为 0.2s。  
SAFEty:STEP1:AC:TIME:RAMP? 查询返回+2.000000E-01 表示其上升时间是 0.2s。

**[SOURCE:]SAFEty:STEP<n>:AC:TIME:FALL** 设定所选步骤交流耐压下降时间。  
[SOURCE:]SAFEty:STEP<n>:AC:TIME:FALL? 查询所选步骤交流耐压下降时间。  
命令语法:[SOURCE:]SAFEty:STEP<n>:AC:TIME:FALL<time>time 是 NR3 格式  
查询语法:[SOURCE:]SAFEty:STEP<n>:AC:TIME:FALL?  
查询返回: <NR3><NL>

例如:

SAFEty:STEP1:AC:TIME:FALL 0.1 设定步骤 1 其交流耐压下降时间为 0.1s。  
SAFEty:STEP1:AC:TIME:FALL? 查询返回+1.000000E-01 表示其下降时间是 0.1s。

**[SOURCE:]SAFEty:STEP<n>:AC:CHANnel[:HIGH]** 设定扫描测试高压输出通道状态。  
[SOURCE:]SAFEty:STEP<n>:AC:CHANnel[:HIGH]? 查询扫描测试高压输出通道状态。  
命令语法:[SOURCE:]SAFEty:STEP<n>:AC:CHANnel[:HIGH] (@(ch,ch,...))  
查询语法:[SOURCE:]SAFEty:STEP<n>:AC:CHANnel[:HIGH]?

例如:

SAFEty:STEP1:AC:CHANnel:HIGH @(1,3) 表示将步骤 1 其扫描测试输出通道的通道 1 和通道 3 设置为 HIGH 高输出。

SAFEty:STEP1:AC:CHANnel:HIGH @(0) 表示将步骤 1 其扫描测试输出通道的 HIGH 高输出的通道设置为关闭 OFF。

SAFEty:STEP1:AC:CHANnel:HIGH? 返回(@{1,3})表示步骤 1 其扫描测试输出通道的通道 1 和通道 3 是 HIGH 高输出。

**[SOURce:]SAFEty:STEP<n>:AC:CHANnel:LOW** 设定扫描测试 LOW 输出通道状态。

[SOURce:]SAFEty:STEP<n>:AC:CHANnel:LOW? 查询扫描测试 LOW 输出通道状态。

命令语法:[SOURce:]SAFEty:STEP<n>:AC:CHANnel:LOW (@{ch,ch,..})

查询语法:[SOURce:]SAFEty:STEP<n>:AC:CHANnel:LOW?

例如:

SAFEty:STEP1:AC:CHANnel:LOW (@{2,4}) 表示将步骤 1 其扫描测试输出通道的通道 2 和通道 4 设置为 LOW 低输出。

SAFEty:STEP1:AC:CHANnel:LOW (@{0}) 表示将步骤 1 其扫描测试输出通道的 LOW 低输出的通道设置为关闭 OFF。

SAFEty:STEP1:AC:CHANnel:LOW? 返回(@{2,4})表示步骤 1 其扫描测试输出通道的通道 2 和通道 4 是 LOW 低输出。

## 2.7 DC 直流耐压系统命令

**[SOURce:]SAFEty:STEP<n>:DC[:LEVel]** 设定所选步骤直流耐压电压值，

[SOURce:]SAFEty:STEP<n>:DC[:LEVel]? 查询所选步骤直流耐压电压值。

命令语法: [SOURce:]SAFEty:STEP<n>:DC[:LEVel] <level> level 是 NR3 格式  
level 范围是 50~6000，单位为 V。

查询语法: [SOURce:]SAFEty:STEP<n>:DC[:LEVel]?

查询返回: <NR3><NL>

**[SOURce:]SAFEty:STEP<n>:DC:LIMit[:HIGH]** 设定所选步骤直流耐压漏电流上限值。

[SOURce:]SAFEty:STEP<n>:DC:LIMit[:HIGH]? 查询所选步骤直流耐压漏电流上限值。

命令语法: [SOURce:]SAFEty:STEP<n>:DC:LIMit[:HIGH]<value>value 是 NR3 格式

查询语法: [SOURce:]SAFEty:STEP<n>:DC:LIMit[:HIGH]?

查询返回: <NR3><NL>

例如:

SAFEty:STEP2:DC:LIMit:HIGH 0.003 设定步骤 2 其直流耐压电流上限值为 3mA。

SAFEty:STEP2:DC:LIMit:HIGH? 查询返回+3.000000E-3 表示步骤 2 其直流耐压电流上限值是 3mA

**[SOURce:]SAFEty:STEP<n>:DC:LIMit:LOW** 设定所选步骤直流耐压漏电流下限值。

[SOURce:]SAFEty:STEP<n>:DC:LIMit:LOW? 查询所选步骤直流耐压漏电流下限值。

命令语法: [SOURce:]SAFEty:STEP<n>:DC:LIMit:LOW<value>value 是 NR3 格式



查询语法: [SOURce:]SAFEty:STEP<n>:DC:LIMit:LOW?

查询返回: <NR3><NL>

例如:

SAFEty:STEP2:DC:LIMit:LOW 0.00001 设定步骤 2 其直流耐压电流下限值为 0.01mA。

SAFEty:STEP2:DC:LIMit:LOW? 查询返回+1.000000E-5 表示步骤 2 其直流耐压电流下限值是 0.01mA

**[SOURce:]SAFEty:STEP<n>:DC:LIMit:ARC[:LEVel]** 设定所选步骤直流耐压电弧 ARC 检测值。**[SOURce:]SAFEty:STEP<n>:DC:LIMit:ARC[:LEVel]**? 查询所选步骤直流耐压电弧 ARC 检测值

命令语法:[SOURce:]SAFEty:STEP<n>:DC:LIMit:ARC[:LEVel] <value>  
value 是 NR3 格式

查询语法: [SOURce:]SAFEty:STEP<n>:DC:LIMit:ARC[:LEVel]?

查询返回: <NR3><NL>

**[SOURce:]SAFEty:STEP<n>:DC:TIME[:TEST]** 设定所选步骤直流耐压测试时间。

**[SOURce:]SAFEty:STEP<n>:DC:TIME[:TEST]**? 查询所选步骤直流耐压测试时间。

命令语法:[SOURce:]SAFEty:STEP<n>:DC:TIME[:TEST] <time> time 是 NR3 格式

查询语法: [SOURce:]SAFEty:STEP<n>:DC:TIME[:TEST]?

查询返回: <NR3><NL>

例如:

SAFEty:STEP2:DC:TIME:TEST 3 设定步骤 2 其直流耐压测试时间为 3s。

SAFEty:STEP2:DC:TIME:TEST? 查询返回+3.000000E+00 表示其测试时间是 3s。

**[SOURce:]SAFEty:STEP<n>:DC:TIME:RAMP** 设定所选步骤直流耐压上升时间。

**[SOURce:]SAFEty:STEP<n>:DC:TIME:RAMP?** 查询所选步骤直流耐压上升时间。

命令语法:[SOURce:]SAFEty:STEP<n>:DC:TIME:RAMP <time> time 是 NR3 格式

查询语法: [SOURce:]SAFEty:STEP<n>:DC:TIME:RAMP?

查询返回: <NR3><NL>

例如:

SAFEty:STEP2:DC:TIME:RAMP 0.2 设定步骤 2 其直流耐压上升时间为 0.2s。

SAFEty:STEP2:DC:TIME:RAMP? 查询返回+2.000000E-01 表示其上升时间是 0.2s。

**[SOURce:]SAFEty:STEP<n>:DC:TIME:FALL** 设定所选步骤直流耐压下降时间。

**[SOURce:]SAFEty:STEP<n>:DC:TIME:FALL?** 查询所选步骤直流耐压下降时间。

命令语法:[SOURce:]SAFEty:STEP<n>:DC:TIME:FALL <time> time 是 NR3 格式

查询语法: [SOURce:]SAFEty:STEP<n>:DC:TIME:FALL?

查询返回: <NR3><NL>

例如:

SAFEty:STEP2:DC:TIME:FALL 0.1 设定步骤 2 其直流耐压下降时间为 0.1s。

SAFEty:STEP2:DC:TIME:FALL? 查询返回+1.000000E-01 表示其下降时间是 0.1s。

**[SOURce:]SAFEty:STEP<n>:DC:TIME:DWELI** 设定所选步骤直流耐压持续时间。

[SOURce:]SAFEty:STEP<n>:DC:TIME:DWELI? 查询所选步骤直流耐压持续时间。

命令语法:[SOURce:]SAFEty:STEP<n>:DC:TIME:DWELI<time> time 是 NR3 格式

查询语法: [SOURce:]SAFEty:STEP<n>:DC:TIME:DWELI?

查询返回: <NR3><NL>

例如:

SAFEty:STEP2:DC:TIME:DWELI 1.0 设定步骤 2 其直流耐压持续时间为 1.0s。

SAFEty:STEP2:DC:TIME:DWELI? 查询返回+1.000000E+00 表示其持续时间是 1.0s。

**[SOURce:]SAFEty:STEP<n>:DC:CHANnel[:HIGH]** 设定扫描测试高压输出通道状态。

[SOURce:]SAFEty:STEP<n>:DC:CHANnel[:HIGH]? 查询扫描测试高压输出通道状态。

命令语法:[SOURce:]SAFEty:STEP<n>:DC:CHANnel[:HIGH] (@(ch,ch,...))

查询语法:[SOURce:]SAFEty:STEP<n>:DC:CHANnel[:HIGH]?

例如:

SAFEty:STEP2:DC:CHANnel:HIGH @(1,3) 表示将步骤 2 其扫描测试输出通道的通道 1 和通道 3 设置为 HIGH 高输出。

SAFEty:STEP2:DC:CHANnel:HIGH @(0) 表示将步骤 2 其扫描测试输出通道的 HIGH 高输出的通道设置为关闭 OFF。

SAFEty:STEP2:DC:CHANnel:HIGH? 返回(@(1,3))表示步骤 2 其扫描测试输出通道的通道 1 和通道 3 是 HIGH 高输出。

**[SOURce:]SAFEty:STEP<n>:DC:CHANnel:LOW** 设定扫描测试 LOW 输出通道状态。

[SOURce:]SAFEty:STEP<n>:DC:CHANnel:LOW? 查询扫描测试 LOW 输出通道状态。

命令语法:[SOURce:]SAFEty:STEP<n>:DC:CHANnel:LOW (@(ch,ch,...))

查询语法:[SOURce:]SAFEty:STEP<n>:DC:CHANnel:LOW?

例如:

SAFEty:STEP2:DC:CHANnel:LOW @(2,4) 表示将步骤 2 其扫描测试输出通道的通道 2 和通道 4 设置为 LOW 低输出。

SAFEty:STEP2:DC:CHANnel:LOW @(0) 表示将步骤 2 其扫描测试输出通道的 LOW 低输出的通道设置为关闭 OFF。

SAFEty:STEP2:DC:CHANnel:LOW? 返回(@(2,4))表示步骤 2 其扫描测试输出通道的通道 2 和通道 4 是 LOW 低输出。

## 2.8 IR 绝缘电阻系统命令

**[SOURce:]SAFEty:STEP<n>:IR[:LEVel]** 设定所选步骤绝缘电阻测试电压值，  
**[SOURce:]SAFEty:STEP<n>:IR[:LEVel]?** 查询所选步骤绝缘电阻测试电压值。  
命令语法: **[SOURce:]SAFEty:STEP<n>:IR[:LEVel] <level>** level 是 NR3 格式  
level 范围是 50~1000，单位为 V。

查询语法: **[SOURce:]SAFEty:STEP<n>:IR[:LEVel]?**

查询返回: <NR3><NL>

**[SOURce:]SAFEty:STEP<n>:IR:LIMit:HIGH** 设定所选步骤绝缘电阻上限值。  
**[SOURce:]SAFEty:STEP<n>:IR:LIMit[:HIGH]?** 查询所选步骤绝缘电阻上限值。  
命令语法: **[SOURce:]SAFEty:STEP<n>:IR:LIMit:HIGH<value>** value 是 NR3 格式  
查询语法: **[SOURce:]SAFEty:STEP<n>:IR:LIMit:HIGH?**

查询返回: <NR3><NL>

例如:

**SAFEty:STEP3:IR:LIMit:HIGH 1.0e+10** 设定步骤 3 其绝缘电阻上限值为 10GΩ。

**SAFEty:STEP3:IR:LIMit:HIGH?** 查询返回+1.000000E+10 表示步骤 3 其绝缘电阻上限值是 10GΩ。

**[SOURce:]SAFEty:STEP<n>:IR:LIMit[:LOW]** 设定所选步骤绝缘电阻下限值。  
**[SOURce:]SAFEty:STEP<n>:IR:LIMit[:LOW]?** 查询所选步骤绝缘电阻下限值。  
命令语法: **[SOURce:]SAFEty:STEP<n>:IR:LIMit[:LOW]<value>** value 是 NR3 格式  
查询语法: **[SOURce:]SAFEty:STEP<n>:IR:LIMit[:LOW]?**

查询返回: <NR3><NL>

例如:

**SAFEty:STEP3:IR:LIMit:LOW 100000000** 设定步骤 3 其绝缘电阻下限值为 100MΩ。

**SAFEty:STEP3:IR:LIMit:LOW?** 查询返回+1.000000E+08 表示步骤 3 其绝缘电阻下限值是 100MΩ

**[SOURce:]SAFEty:STEP<n>:IR:TIME[:TEST]** 设定所选步骤绝缘电阻测试时间。  
**[SOURce:]SAFEty:STEP<n>:IR:TIME[:TEST]?** 查询所选步骤绝缘电阻测试时间。  
命令语法: **[SOURce:]SAFEty:STEP<n>:IR:TIME[:TEST] <time>** time 是 NR3 格式  
查询语法: **[SOURce:]SAFEty:STEP<n>:IR:TIME[:TEST]?**

查询返回: <NR3><NL>

例如:

**SAFEty:STEP3:IR:TIME:TEST 3** 设定步骤 3 其绝缘电阻测试时间为 3s。

**SAFEty:STEP3:IR:TIME:TEST?** 查询返回+3.000000E+00 表示其测试时间是 3s。

**[SOURCE:]SAFETY:STEP<n>:IR:TIME:RAMP** 设定所选步骤绝缘电阻上升时间。

**[SOURCE:]SAFETY:STEP<n>:IR:TIME:RAMP?** 查询所选步骤绝缘电阻上升时间。

命令语法:**[SOURCE:]SAFETY:STEP<n>:IR:TIME:RAMP <time>** time 是 NR3 格式

查询语法: **[SOURCE:]SAFETY:STEP<n>:IR:TIME:RAMP?**

查询返回: <NR3><NL>

例如:

**SAFETY:STEP3:IR:TIME:RAMP 0.2** 设定步骤 3 其绝缘电阻上升时间为 0.2s。

**SAFETY:STEP3:IR:TIME:RAMP?** 查询返回+2.000000E-01 表示其上升时间是 0.2s。

**[SOURCE:]SAFETY:STEP<n>:IR:TIME:FALL** 设定所选步骤绝缘电阻下降时间。

**[SOURCE:]SAFETY:STEP<n>:IR:TIME:FALL?** 查询所选步骤绝缘电阻下降时间。

命令语法:**[SOURCE:]SAFETY:STEP<n>:IR:TIME:FALL <time>** time 是 NR3 格式

查询语法: **[SOURCE:]SAFETY:STEP<n>:IR:TIME:FALL?**

查询返回: <NR3><NL>

例如:

**SAFETY:STEP3:IR:TIME:FALL 0.1** 设定步骤 3 其绝缘电阻下降时间为 0.1s。

**SAFETY:STEP3:IR:TIME:FALL?** 查询返回+1.000000E-01 表示其下降时间是 0.1s。

**[SOURCE:]SAFETY:STEP<n>:IR:AGC** 设定所选步骤 AGC 软键增益自动调整开关。

**[SOURCE:]SAFETY:STEP<n>:IR:AGC?** 查询所选步骤 AGC 软键增益自动调整开关。

命令语法:**[SOURCE:]SAFETY:STEP<n>:IR:AGC {{1 | ON} | {0 | OFF}}**

查询语法: **[SOURCE:]SAFETY:STEP<n>:IR:AGC?**

查询返回: {1 | 0}<NL>

**[SOURCE:]SAFETY:STEP<n>:IR:RANGE[:UPPER]** 设定所选步骤绝缘电阻测试量程。

**[SOURCE:]SAFETY:STEP<n>:IR:RANGE[:UPPER]?** 查询所选步骤绝缘电阻测试量程。

命令语法:**[SOURCE:]SAFETY:STEP<n>:IR:RANGE[:UPPER] <value>**

此命令根据使用者输入的电流值，来选取高于该电流所能测量的量程。例如

**SAFETY:STEP3:IR:RANG 0.0003** 表示步骤 3 能够测量的电流值是 300uA，所以此时被选取高于该电流能测量的量程是 3mA。

查询语法: **[SOURCE:]SAFETY:STEP<n>:IR:RANGE[:UPPER]?**

查询返回: <NR3><NL>

例如返回+3.000000E-03 表示其设定的量程是 3mA。

**[SOURCE:]SAFETY:STEP<n>:IR:RANGE:LOWER** 设定所选步骤绝缘电阻测试量程。

**[SOURCE:]SAFETY:STEP<n>:IR:RANGE:LOWER?** 查询所选步骤绝缘电阻测试量程。

命令语法:**[SOURCE:]SAFETY:STEP<n>:IR:RANGE:LOWER<value>**

此命令根据使用者输入的电流值，来选取低于该电流所能测量的量程。例如 SAFETy:STEP3:IR:RANG:LOW 0.0003 表示步骤 3 能够测量的电流值是 300uA，所以此时被选取低于该电流能测量的量程是 300uA。

查询语法: [SOURce:]SAFETy:STEP<n>:IR:RANGe:LOWer?

查询返回: <NR3><NL>

例如返回+3.000000E-04 表示其设定的量程是 300uA。

**[SOURce:]SAFETy:STEP<n>:IR:RANGe:AUTO** 设定所选步骤绝缘电阻自动量程开关。

[SOURce:]SAFETy:STEP<n>:IR:RANGe:AUTO? 查询所选步骤绝缘电阻自动量程开关。

命令语法:[SOURce:]SAFETy:STEP<n>:IR:RANGe:AUTO {{1 | ON}| {0 | OFF}}

查询语法: [SOURce:]SAFETy:STEP<n>:IR:RANGe:AUTO?

查询返回: {1 | 0}<NL>

**[SOURce:]SAFETy:STEP<n>:IR:CHANnel[:HIGH]** 设定扫描测试高压输出通道状态。

[SOURce:]SAFETy:STEP<n>:IR:CHANnel[:HIGH]? 查询扫描测试高压输出通道状态。

命令语法:[SOURce:]SAFETy:STEP<n>:IR:CHANnel[:HIGH] (@(ch,ch,...))

查询语法:[SOURce:]SAFETy:STEP<n>:IR:CHANnel[:HIGH]?

例如:

SAFETy:STEP3:IR:CHANnel[:HIGH] @(1,3) 表示将步骤 3 其扫描测试输出通道的通道 1 和通道 3 设置为 HIGH 高输出。

SAFETy:STEP3:IR:CHANnel[:HIGH] @(0) 表示将步骤 3 其扫描测试输出通道的 HIGH 高输出的通道设置为关闭 OFF。

SAFETy:STEP3:IR:CHANnel[:HIGH]? 返回(@(1,3))表示步骤 3 其扫描测试输出通道的通道 1 和通道 3 是 HIGH 高输出。

**[SOURce:]SAFETy:STEP<n>:IR:CHANnel:LOW** 设定扫描测试 LOW 输出通道状态。

[SOURce:]SAFETy:STEP<n>:IR:CHANnel:LOW? 查询扫描测试 LOW 输出通道状态。

命令语法:[SOURce:]SAFETy:STEP<n>:IR:CHANnel:LOW (@(ch,ch,...))

查询语法:[SOURce:]SAFETy:STEP<n>:IR:CHANnel:LOW?

例如:

SAFETy:STEP3:IR:CHANnel:LOW @(2,4) 表示将步骤 3 其扫描测试输出通道的通道 2 和通道 4 设置为 LOW 低输出。

SAFETy:STEP3:IR:CHANnel:LOW @(0) 表示将步骤 3 其扫描测试输出通道的 LOW 低输出的通道设置为关闭 OFF。

SAFETy:STEP3:IR:CHANnel:LOW? 返回(@(2,4))表示步骤 3 其扫描测试输出通道的通道 2 和通道 4 是 LOW 低输出。

## 2.90S 开短路系统命令

**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:OPEN** 设定所选步骤 OS 测试时开路判断所设定的百分比值。**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:OPEN?** 查询开路判断所设定的百分比值。

命令语法:**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:OPEN <value>** value 是 NR3 格式

查询语法:**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:OPEN?**

查询返回: <NR3><NL>

例如

**SAFETY:STEP4:OSC:LIM:OPEN 0.3** 设定开路判断所设定的百分比为 30%。

**SAFETY:STEP4:OSC:LIM:OPEN?** 返回+3.000000E-01 表示开路判断所设定的百分比为 30%。

**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:SHORT** 设定所选步骤 OS 测试时短路判断所设定的百分比值。**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:SHORT?** 查询短路判断所设定的百分比值。

命令语法:**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:SHORT<value>** value 是 NR3 格式

查询语法:**[SOURCE:]SAFETY:STEP<n>:OSC[:LIMit]:SHORT?**

查询返回: <NR3><NL>

例如

**SAFETY:STEP4:OSC:LIM:SHORT3.0** 设定短路判断所设定的百分比为 300%。

**SAFETY:STEP4:OSC:LIM:SHORT?** 返回+3.000000E+00 表示短路判断所设定的百分比为 300%。

**[SOURCE:]SAFETY:STEP<n>:OSC:GET** 执行所选步骤 OS 测试的获取标准功能。

命令语法:**[SOURCE:]SAFETY:STEP<n>:OSC:GET** 此命令用于执行一次获取标准(Get Cs)测试，并将测试结果写入标准中。

**[SOURCE:]SAFETY:STEP<n>:OSC:CHANnel[:HIGH]** 设定扫描测试高压输出通道状态。

**[SOURCE:]SAFETY:STEP<n>:OSC:CHANnel[:HIGH]?** 查询扫描测试高压输出通道状态。

命令语法:**[SOURCE:]SAFETY:STEP<n>:OSC:CHANnel[:HIGH] (@(ch,ch,...))**

查询语法:**[SOURCE:]SAFETY:STEP<n>:OSC:CHANnel[:HIGH]?**

例如:

**SAFETY:STEP4:OSC:CHANnel[:HIGH] @(1,3)** 表示将步骤 4 其扫描测试输出通道的通道 1 和通道 3 设置为 HIGH 高输出。

**SAFETY:STEP4:OSC:CHANnel[:HIGH] @(0)** 表示将步骤 4 其扫描测试输出通道的 HIGH 高输出的通道设置为关闭 OFF。

SAFEty:STEP4:OSC:CHANnel[:HIGH]? 返回(@{1,3})表示步骤 4 其扫描测试输出通道的通道 1 和通道 3 是 HIGH 高输出。

**[SOURCE:]SAFEty:STEP<n>:OSC:CHANnel:LOW** 设定扫描测试 LOW 输出通道状态。

[SOURCE:]SAFEty:STEP<n>:OSC:CHANnel:LOW? 查询扫描测试 LOW 输出通道状态。

命令语法:[SOURCE:]SAFEty:STEP<n>:OSC:CHANnel:LOW (@{ch,ch,..})

查询语法:[SOURCE:]SAFEty:STEP<n>:OSC:CHANnel:LOW?

例如:

SAFEty:STEP4:OSC:CHANnel:LOW (@{2,4}) 表示将步骤 4 其扫描测试输出通道的通道 2 和通道 4 设置为 LOW 低输出。

SAFEty:STEP4:OSC:CHANnel:LOW (@{0}) 表示将步骤 4 其扫描测试输出通道的 LOW 低输出的通道设置为关闭 OFF。

SAFEty:STEP4:OSC:CHANnel:LOW? 返回(@{2,4})表示步骤 4 其扫描测试输出通道的通道 2 和通道 4 是 LOW 低输出。

## 2.10 PA 系统命令

**[SOURCE:]SAFEty:STEP<n>:PAuse[:MESSAge]**用于设置 message 信息字符串。

[SOURCE:]SAFEty:STEP<n>:PAuse[:MESSAge]?查询 message 信息字符串。

命令语法:[SOURCE:]SAFEty:STEP<n>:PAuse[:MESSAge] "message"

注意: message 必须使用双引号。

查询语法:[SOURCE:]SAFEty:STEP<n>:PAuse[:MESSAge]?

查询返回: <"message"><NL>返回字符串有双引号。

**[SOURCE:]SAFEty:STEP<n>:Pause:UTSignal** 设定 under test signal 状态。

[SOURCE:]SAFEty:STEP<n>:Pause:UTSignal?查询 under test signal 状态。

命令语法:[SOURCE:]SAFEty:STEP<n>:Pause:UTSignal {{1 | ON} | {0 | OFF}}

查询语法:[SOURCE:]SAFEty:STEP<n>:Pause:UTSignal?

查询返回: {0 | 1}<NL>

**[SOURCE:]SAFEty:STEP<n>:Pause:TIME** 设定 PA 测试所需时间。

[SOURCE:]SAFEty:STEP<n>:Pause:TIME?查询 PA 测试所需时间。

命令语法:[SOURCE:]SAFEty:STEP<n>:Pause:TIME<time> time 是 NR3 格式

查询语法:[SOURCE:]SAFEty:STEP<n>:Pause:TIME?

查询返回: <NR3><NL>

例如: SAFEty:STEP5:PA:TIME 5 设定步骤 5 的 PA 测试时间为 5s。

SAFEty:STEP5:PA:TIME?返回 5.000000E+00 表示步骤 5 测试设定时间是 5s。

## 2.11 PRESET 测量配置系统命令

**[SOURCE:]SAFETy:PRESet:TIME:PASS** 用于设定 PASS 时蜂鸣器响声持续时间。

**[SOURCE:]SAFETy:PRESet:TIME:PASS?** 查询 PASS 时蜂鸣器响声持续时间。

命令语法: **[SOURCE:]SAFETy:PRESet:TIME:PASS<time>time** 是 NR3 格式, 范围是 0.2~99.9s

查询语法: **[SOURCE:]SAFETy:PRESet:TIME:PASS?**

查询返回: <NR3><NL>

例如:

**SAFE:PRE:TIME:PASS 1.0** 设定 PASS 时蜂鸣器响声持续时间为 1s。

**SAFE:PRE:TIME:PASS?** 返回 1.000000E+00 表示 PASS 时蜂鸣器响声持续时间是 1s。

**[SOURCE:]SAFETy:PRESet:TIME:STEP** 用于设定测试步骤间的间隔时间。

**[SOURCE:]SAFETy:PRESet:TIME:STEP?** 查询测试步骤间的间隔时间。

命令语法: **[SOURCE:]SAFETy:PRESet:TIME:STEP{time | KEY}**

time 是 NR3 格式, 范围是 0.0~99.9s

查询语法: **[SOURCE:]SAFETy:PRESet:TIME:STEP?**

查询返回: {NR3 | KEY}<NL>

例如:

**SAFETy:PRESet:TIME:STEP0.5** 设定测试步骤间的间隔时间为 0.5s。

**SAFETy:PRESet:TIME:STEP?** 返回 5.000000E-01 表示测试步骤间的间隔时间是 0.5s。

**[SOURCE:]SAFETy:PRESet:TIME:SDElay** 用于设定测试启动前的延时时间。

**[SOURCE:]SAFETy:PRESet:TIME:SDElay?** 查询测试启动前的延时时间。

命令语法: **[SOURCE:]SAFETy:PRESet:TIME:SDElay<time>**

time 是 NR3 格式, 范围是 0.0~99.9s

查询语法: **[SOURCE:]SAFETy:PRESet:TIME:SDElay?**

查询返回: <NR3><NL>

例如:

**SAFETy:PRESet:TIME:SDEL 0.5** 设定测试启动前的延时时间为 0.5s。

**SAFETy:PRESet:TIME:SDEL?** 返回 5.000000E-01 表示测试启动前的延时时间是 0.5s。

**[SOURCE:]SAFETy:PRESet:RJUDgment** 用于设定上升时间判别功能开关。

**[SOURCE:]SAFETy:PRESet:RJUDgment?** 查询上升时间判别功能开关。

命令语法: **[SOURCE:]SAFETy:PRESet:RJUDgment {{1 | ON} | {0 | OFF}}**

查询语法: **[SOURCE:]SAFETy:PRESet:RJUDgment?**

查询返回: {0 | 1}<NL>



**[SOURCE:]SAFETy:PRESet:AC:FREQuency** 用于设定交流耐压测试信号频率。

**[SOURCE:]SAFETy:PRESet:AC:FREQuency?** 查询交流耐压测试信号频率。

命令语法: **[SOURCE:]SAFETy:PRESet:AC:FREQuency {50 | 60}**

查询语法: **[SOURCE:]SAFETy:PRESet:AC:FREQuency?**

查询返回: {50 | 60}<NL>

例如:

**SAFETy:PRESet:AC:FREQuency 50** 设定交流耐压测试信号频率是 50Hz。

**SAFETy:PRESet:AC:FREQuency?** 返回 5.000000E+01 表示频率是 50Hz。

**[SOURCE:]SAFETy:PRESet:WRANge[:AUTO]** 用于设定耐压测试自动量程开关。

**[SOURCE:]SAFETy:PRESet:WRANge[:AUTO]?** 查询耐压测试自动量程开关。

命令语法: **[SOURCE:]SAFETy:PRESet:WRANge[:AUTO] {{1 | ON}} {0 | OFF}}**

查询语法: **[SOURCE:]SAFETy:PRESet:WRANge[:AUTO]?**

查询返回: {0 | 1}<NL>

**[SOURCE:]SAFETy:PRESet:DAGC** 用于设定 DC50V 测试增益自动调整开关。

**[SOURCE:]SAFETy:PRESet:DAGC?** 查询 DC50V 测试增益自动调整开关。

命令语法: **[SOURCE:]SAFETy:PRESet:DAGC {{1 | ON}} {0 | OFF}}**

查询语法: **[SOURCE:]SAFETy:PRESet:DAGC?**

查询返回: {0 | 1}<NL>

**[SOURCE:]SAFETy:PRESet:GFI[:SWITCh]** 用于设定 GFI 触电保护开关。

**[SOURCE:]SAFETy:PRESet:GFI[:SWITCh]?** 查询 GFI 触电保护开关。

命令语法: **[SOURCE:]SAFETy:PRESet:GFI[:SWITCh] {{1 | ON}} {0 | OFF}}**

查询语法: **[SOURCE:]SAFETy:PRESet:GFI[:SWITCh]?**

查询返回: {0 | 1}<NL>

**[SOURCE:]SAFETy:PRESet:FAIL[:OPERation]** 用于设定测试不合格 FAIL 后的动作。

**[SOURCE:]SAFETy:PRESet:FAIL[:OPERation]?** 查询测试不合格 FAIL 后的动作。

命令语法:

**[SOURCE:]SAFETy:PRESet:FAIL[:OPERation]{STOP|CONTInue|REStArt|NEXT }**

其中: STOP 表示停止测试;

CONTInue 表示继续测试;

REStArt 表示发生 FAIL 后, 再按下 START 则重新测试;

NEXT 表示发生 FAIL 后, 再按下 START 执行下一步测试。

查询语法: **[SOURCE:]SAFETy:PRESet:FAIL[:OPERation]?**

查询返回: {STOP|CONTINUE|RESTART|NEXT}<NL>

**[SOURce:]SAFETy:PRESet:ARC[:MODE]**用于设定电弧测试模式。

**[SOURce:]SAFETy:PRESet:ARC[:MODE]?**查询电弧测试模式。

命令语法:**[SOURce:]SAFETy:PRESet:ARC[:MODE] {CURRENT|LEVEL }**

其中: **CURRENT** 表示电流模式;

**LEVEL** 表示等级模式。

查询语法: **[SOURce:]SAFETy:PRESet:ARC[:MODE]?**

查询返回:**{CURRENT|LEVEL}<NL>**

## 2.12 保存/调出系统命令

**MMEMory:LOAD:STATe** 命令用于加载已保存的文件。

命令语法: MMEMory:LOAD:STATe <file number>

<file number>是文件序号, 范围是 1~600, NR1 数据格式, 无后缀参数。

- ①注意: 1. 如果加载的文件不存在, 仪器将显示“没有数据加载”报警信息。
- 2. 如果指定的文件号超出范围, 仪器将显示“数据超出范围”的报警信息。此项同样适用于下面两个命令。

命令语法: MMEMory:LOAD:STATe <“filename”>

该命令直接通过输入的文件名“filename”来查找文件并进行加载。

- ①注意: 使用文件名加载文件仅限于内部文件。

**MMEMory:SAVE:STATe** 或 **MMEMory:STORe:STATe** 命令用于保存当前仪器的设置到一个文件。

命令语法: MMEMory:STORe:STATe <file number> [, <“filename”>]

<file number> 是文件序号, 范围是 1~600, NR1 数据格式, 无后缀参数。

<“filename”>是要保存的文件名, 可用 20 个以内的 ASCII 字符表示, 注意要用引号将文件名括起来。如果未指定文件名, 仪器以缺省的<Unnamed>命名。

- ①注意: 本仪器在覆盖已存在的文件记录时并不会提示。

☞提醒: 总线指定的文件名总是被原样引用, 因此通过总线可以得到一些仪器面板上不能输入的字符, 如一些特殊符号等。

**MMEMory:DELeTe:STATe** 命令用于删除一个文件。

命令语法: MMEMory:DELeTe:STATe <file number>

<file number>是文件序号, 范围是 1~600, NR1 数据格式, 无后缀参数。

- ①注意: 仪器删除存在的文件记录时并不会提示。

命令语法: Mass MEMory:DELeTe:STATe “filename”

按照文件名删除文件, “filename” 是要删除文件名称, 可用 20 个以内的 ASCII 字符表示, 注意要用引号将文件名括起来。

- ①注意: 使用文件名删除文件仅限于内部文件。

## 2.13 按键系统命令

KEY 按键系统命令等同于前面板上的按键和旋钮。

KEY:LOCAL	返回本地
KEY:KEYLOCK	LOCK 键锁/解锁按键操作
KEY:MEASURE	测量显示按键
KEY:SETUP	设置按键
KEY:UPPER	上按键
KEY:DOWN	下按键
KEY:LEFT	左按键
KEY:RIGHT	右按键
KEY:NUM<n>	数字键, n 为 0~9
KEY:DOT	小数点按键
KEY:SIGN	负号按键
KEY:ENTER	回车键
KEY:BACKSPACE	退格键
KEY:ESC	取消键
KEY:HARD	保存按键
KEY:F<n>	软键, n 为 1~5

## 2.14 系统配置系统命令

**SYSTEM:SDATE** 用于设定年月日。SYSTEM:SDATE?查询当前年月日。

命令语法:SYSTEM:SDATE<year>,<month>,<day>year,month,day 是 NR1 格式。

其中 month 也可以是字符串格式: {JANuary | FEBruary | MARch | APRil | MAY | JUNe | JULy | AUGust | SEPtember | OCTober | NOVember | DECember}

查询语法: SYSTEM:SDATE?

查询返回: <NR1>,<NR1>,<NR1><NL>

**SYSTEM:STIME** 设定时分秒。SYSTEM:STIME?查询时分秒。

命令语法:SYSTEM:STIME<hour>,<minute>,<second>

hour, minute, second 是 NR1 格式。

查询语法: SYSTEM:STIME?

查询返回: <NR1>,<NR1>,<NR1><NL>

**SYSTEM:STYPE** 用于设定 **SAVE** 按键保存文件类型。SYSTEM:STYPE?查询 **SAVE** 按键保存文件类型。

命令语法:SYSTEM:STYPE {CSV | GIF | BMP | PNG}

查询语法:SYSTem:STYPe?

查询返回:{CSV | GIF | BMP | PNG}<NL>

**SYSTem:PRESet** 用于复位系统设置参数（不包含密码及日期时间）。

命令语法:SYSTem:PRESet

**SYSTem:BEEPer:PASS[:MODE]**用于设定 PASS 时蜂鸣器响声类型。

SYSTem:BEEPer:PASS[:MODE]? 查询 PASS 时蜂鸣器响声类型。

命令语法:SYSTem:BEEPer:PASS[:MODE] {CONTInue | GAP}

其中: CONTInue 表示连续声音;  
GAP 表示间断声音。

查询语法:SYSTem:BEEPer:PASS[:MODE]?

查询返回: {CONT | GAP}<NL>

**SYSTem:BEEPer:PASS:VOLUme** 用于设定 PASS 时蜂鸣器响声音量。

SYSTem:BEEPer:PASS:VOLUme? 查询 PASS 时蜂鸣器响声音量。

命令语法:SYSTem:BEEPer:PASS:VOLUme{OFF | LOW | MEDium | HIGH}

其中: OFF 表示不发出声音;  
LOW 表示低音;  
MEDium 表示中音;  
HIGH 表示高音。

查询语法:SYSTem:BEEPer:PASS:VOLUme?

查询返回:{OFF | LOW | MED | HIGH}<NL>

**SYSTem:BEEPer:FAIL[:MODE]**用于设定 FAIL 时蜂鸣器响声类型。

SYSTem:BEEPer:FAIL[:MODE]? 查询 FAIL 时蜂鸣器响声类型。

命令语法:SYSTem:BEEPer:FAIL[:MODE] {CONTInue | GAP}

其中: CONTInue 表示连续声音;  
GAP 表示间断声音。

查询语法:SYSTem:BEEPer:FAIL[:MODE]?

查询返回: {CONT | GAP}<NL>

**SYSTem:BEEPer:FAIL:VOLUme** 用于设定 FAIL 时蜂鸣器响声音量。

SYSTem:BEEPer:FAIL:VOLUme? 查询 FAIL 时蜂鸣器响声音量。

命令语法:SYSTem:BEEPer:FAIL:VOLUme {OFF | LOW | MEDium | HIGH}

其中: OFF 表示不发出声音;  
LOW 表示低音;

MEDium 表示中音；

HIGH 表示高音。

查询语法:SYSTem:BEEPer:FAIL:VOLume?

查询返回: {OFF | LOW | MED | HIGH}<NL>

**SYSTem:STYLe:SKIN** 用于设定显示风格。

SYSTem:STYLe:SKIN? 查询显示风格。

命令语法:SYSTem:STYLe:SKIN{GRAY | BLACK | BLUE | CYAN}

其中: GRAY 表示简洁灰；

BLACK 表示炫酷黑；

BLUE 表示典雅蓝；

CYAN 表示深青色。

查询语法:SYSTem:STYLe:SKIN?

查询返回:{GRAY | BLACK | BLUE | CYAN}<NL>

**SYSTem:STYLe:LANGuage** 用于设定系统语言。

SYSTem:STYLe:LANGuage? 查询系统语言。

命令语法:SYSTem:STYLe:LANGuage {CHinese | ENglish}

其中: Chinese 表示中文界面；

English 表示英文界面。

查询语法:SYSTem:STYLe:LANGuage?

查询返回: {CH | EN}<NL>

**SYSTem:STYLe:KSOUnd** 用于设定按键音开关。

SYSTem:STYLe:KSOUnd? 查询按键音开关。

命令语法:SYSTem:STYLe:KSOUnd{{1 | ON}| {0 | OFF}}

查询语法:SYSTem:STYLe:KSOUnd?

查询返回: {0 | 1}<NL>

### 3 出错信息

发给仪器的总线命令中，可能包含错误的命令或语法，或者不正确的参数。本仪器对命令串边分析边执行，如果遇到错误，则显示出错信息并中止命令分析，因此，在一个命令串中，错误之后的内容将被仪器忽略。如果提示的消息是命令忽略或者触发忽略等，那么仪器还是会继续执行分析程序。

下表是总线上常见的错误消息，出错消息将在信息提示区域显示。

错误信息	说明
Undefined message	未知消息，既无法识别的消息。 例如: DISP:PAG MEAS 应该为 DISP:PAGE MEAS
Data out of range	数据超限 例如:SAFE:STEP1:AC:LEV 6000, AC 耐压电压值超出范围
Invalid parameter	无法识别的命令参数。 例如: SAFE:PRES:AC:FREQ 100, 100 为不支持的 AC 耐压频率
Invalid suffix	后缀错误，一般是不匹配的单位。 例如:SAFE:STEP1:AC:TIME:RAMP 1000us, us 是时间不支持的单位。
Data too long	数据太长，例如文件名超过 20 个字符
Syntax error	系统错误，如 DISP 命令后面应该是冒号(:)，如果不是就是系统错误。
Command ignored	命令忽略。例如在测试进行过程中，DISP:PAGEMSET 命令被忽略。

## 4 编程实例

本章给出了基于VISA(Virtual Instrument Software Architecture)库,在Visual C++ 6.0、Visual Basic 6.0和LabVIEW 8.5软件开发环境中,通过USB接口传输命令实现仪器基本功能的编程实例,分别实现发送命令和读取返回数据。

VISA是一个用来与各种仪器总线进行通讯的高级应用编程接口(API)。在这里我们所说的VISA是指NI-VISA。NI-VISA是美国国家仪器有限公司根据VISA标准编写的应用程序接口。您可以通过使用NI-VISA,在GPIB、RS232、USB和LAN等接口上,实现仪器与PC的通信。用户无需了解接口总线如何工作,就可以对仪器进行控制。如果您要了解有关NI-VISA API的具体信息,请参考《NI-VISA 用户手册》与《NI-VISA程序员参考手册》。

一个典型的VISA应用分为以下几个步骤:

- (1)为已有的资源建立会话
- (2)对资源做一些配置(如波特率)
- (3)关闭会话资源



## 编程前的准备工作

如果没有安装NI-VISA，请先到NI的网站(<http://www.ni.com/china>)下载。我们采用的安装路径是C:\Program Files\IVI Foundation\VISA。

用USB数据线，一端连接仪器后面板上的USB DEVICE接口，一端连接PC机的USB接口。

然后,打开仪器的电源开关，PC机弹出硬件更新向导对话框，请根据安装向导的提示安装“USB Test and Measurement Device”软件。如图4-1所示。



图 4-1

## 4.1 Visual C++ 6.0 编程实例

请运行 Visual C++ 6.0 软件，并按如下步骤进行操作：

1. 创建一个基于 MFC 的工程。

2. 打开 Project->Settings 中的 C/C++ 选项卡，在 Category 框中选择 Code Generation，在 Use run-time library 框中选择 Debug Multithreaded DLL。点 OK 关闭对话框。如图 4-1-1 所示。

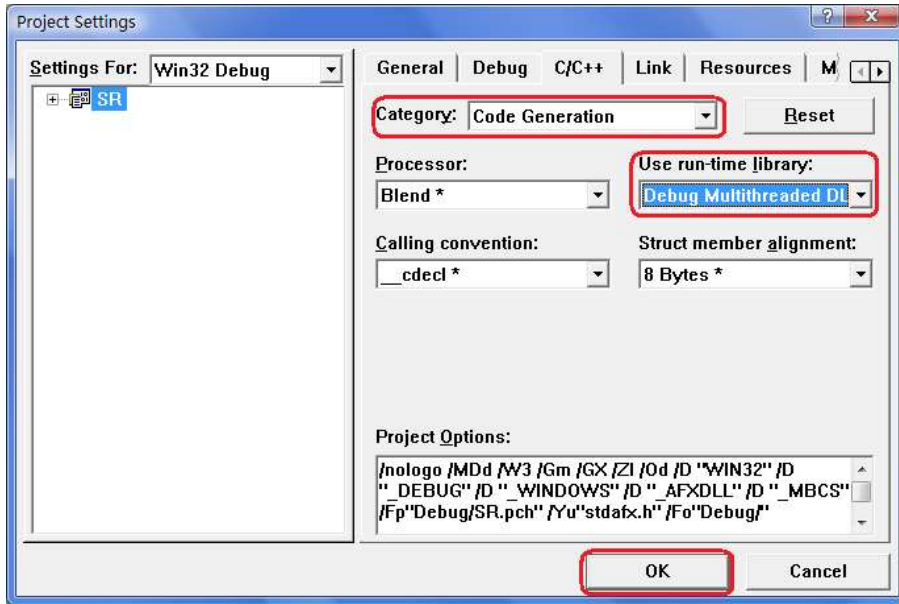


图 4-1-1

3. 打开 Project->Settings 中的 Link 选项卡，在 Object/library modules 框中手动添加 visa32.lib。点 OK 关闭对话框。如图 4-1-2 所示。

4. 打开 Tools->Options 中的 Directories 选项卡。

在 Show directories for 框中选择 Include files，双击 Directories 框中的空白处添加 Include 的路径：C:\Program Files\IVI Foundation\VISA\WinNT\include，如图 4-1-3 所示；

在 Show directories for 框中选择 Library files，双击 Directories 框中的空白处添加 Lib 的路径：C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc。

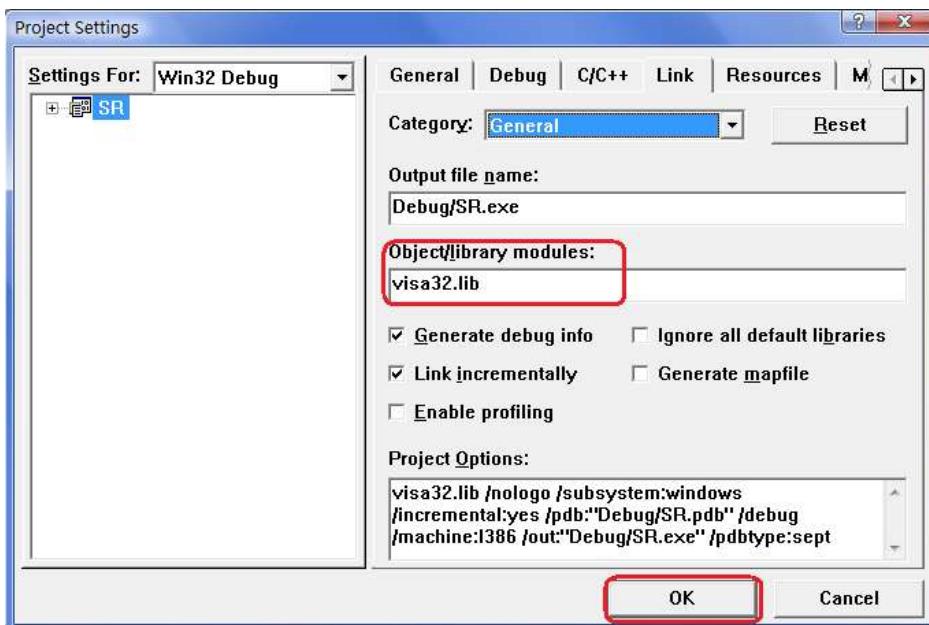


图 4-1-2

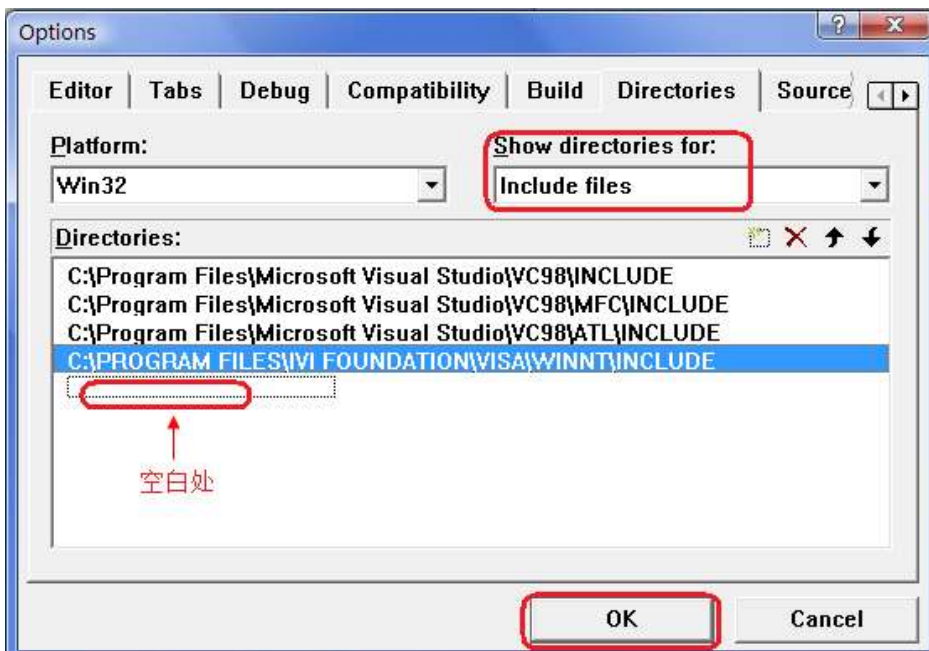


图 4-1-3

5.添加控件 Static Text、Edit 和 Button。如图 4-1-4 所示。

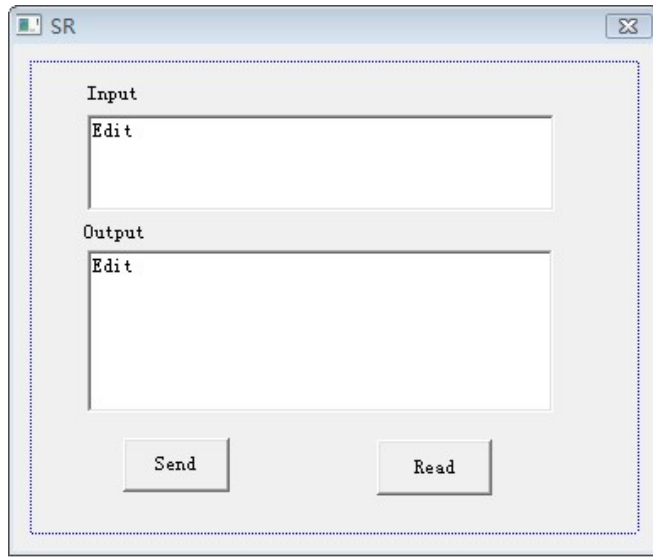


图 4-1-4

(1)添加两个 StaticText 控件，分别命名为 Input 和 Output。

(2)添加两个 Edit 控件，分别添加变量 m\_send 和 m\_read。如图 4-1-5 和 4-1-6 所示。

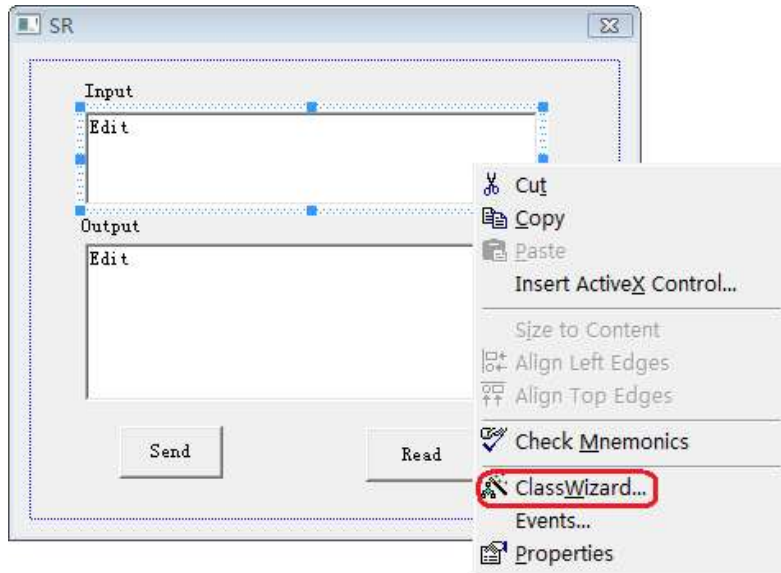


图 4-1-5

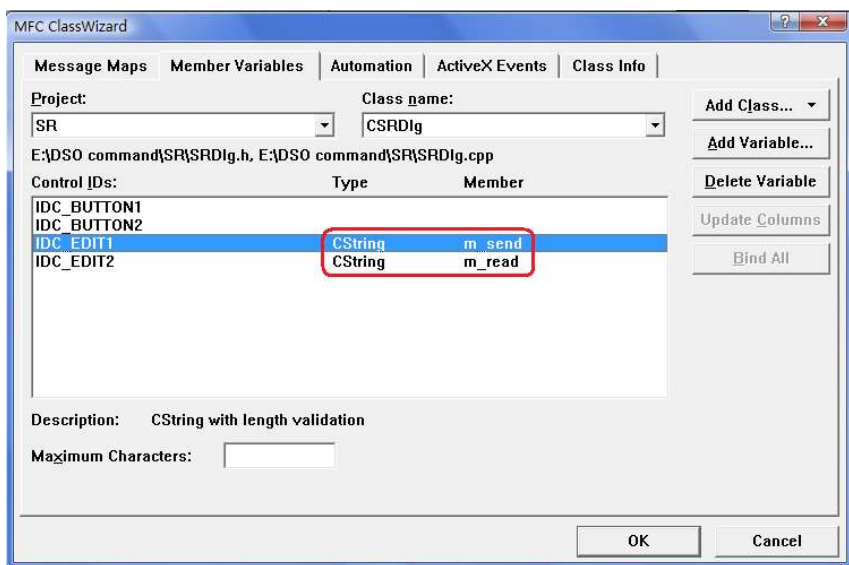


图 4-1-6

(3) 添加两个 Button 控件，分别命名为 Send 和 Read。

6. 双击”Send”按键，进入编程环境。

(1)在头文件处添加“#include “visa.h””,进行声明。

(2)在头文件后，添加如下代码，定义相关的变量。

```
ViSession defaultRM, vi;
char buf [256] = {0};
CString strTemp;
char* stringTemp;
ViChar buffer [VI_FIND_BUFLLEN];
ViRsrc matches=buffer;
ViUInt32 nmatches;
ViFindList list;
```

(3) 在::CSRDIg(CWnd\* pParent /\*=NULL\*/) : CDialog(CSRDIg::IDD, pParent)里，令 m\_send = \_T(“\*IDN?\n”);

(4)在::OnInitDialog()里，添加如下代码。

```
viOpenDefaultRM (&defaultRM);
//获取visa的USB资源
viFindRsrc(defaultRM, "USB?*", &list,&nmatches, matches);
```

```
viOpen (defaultRM,matches,VI_NULL,VI_NULL,&vi);
```

(5)在”Send”按键处，添加如下代码。

```
//发送接收到的命令  
UpdateData (TRUE);  
strTemp = m_send + "\n";  
stringTemp = (char *) (LPCTSTR)strTemp;  
viPrintf (vi,stringTemp);
```

(6) 在”Read”按键处，添加如下代码。

```
//读取结果  
viScanf (vi, "%t\n", &buf);  
//将结果显示出来  
m_read = buf;  
UpdateData (FALSE);
```

(7)在::OnQueryDragIcon()下，添加如下代码。

```
//关闭资源  
viClose (vi);  
viClose (defaultRM);
```

7. 保存、编译并运行程序，得到可执行文件。当仪器与PC机成功相连时，在Input下编辑框中输入一条命令，如\*IDN?(\*IDN?为默认输入命令)，按“Send”按键，再按“Read”按键，仪器返回的结果将显示在Output下编辑框中。如图4-1-7所示。

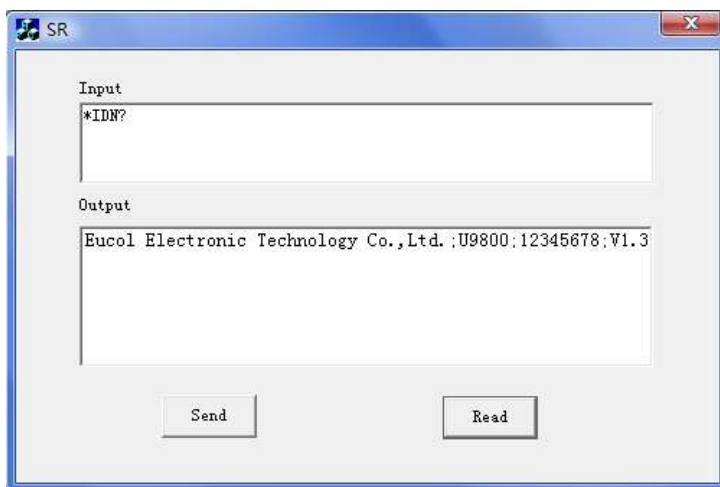


图 4-1-7

## 4.2 Visual Basic 6.0 编程实例

请运行 Visual Basic 6.0 软件，并按如下步骤进行操作：

1. 建立一个Standard EXE工程。

2. 打开 Project->Add Module 的 **Existing** 选项卡，打开 NI-VISA 安装路径下 C:\Program Files\IVI Foundation\VISA\WinNT\include 文件夹中的 **visa32.bas** 文件并添加。如图 4-2-1 所示。

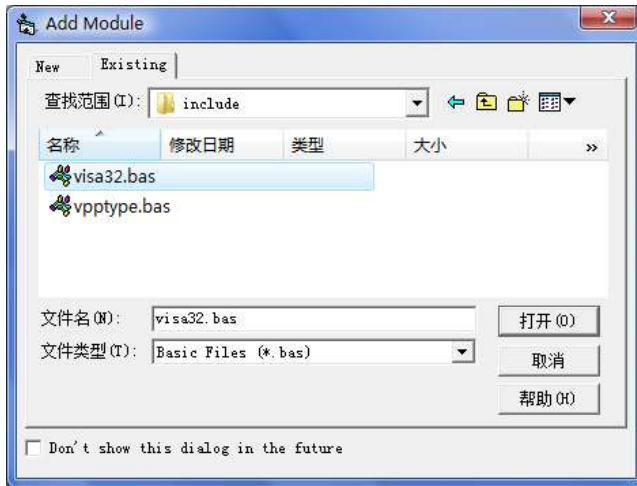


图 4-2-1

3. 添加名称分别为 Input 和 Output 的 Label、两个 TextBox 以及名称分别为 Send 和 Read 的 CommandButton。并把 Input 下的 TextBox 的属性中的 Text 设置为 \*IDN? 的初始状态。如图 4-2-2 所示。

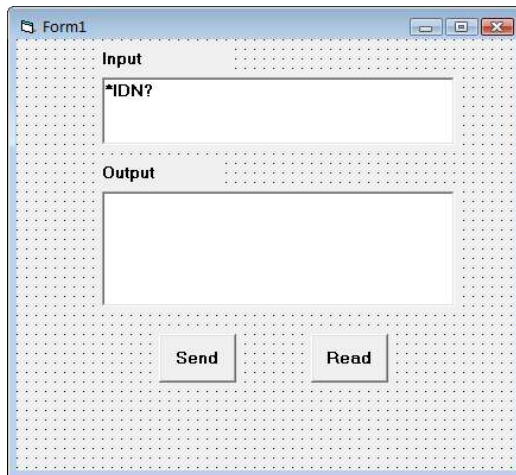


图 4-2-2

4.打开 Project->Project1 Properties 中的 **General** 选项卡，在 Startup Object 下拉框中选择 **Form1**。

5.双击 Send 按钮进入编程环境，添加如下代码。

```
Dim defrm As Long
```

```
Dim vi As Long
```

```
Dim list As Long
```

```
Dim nmatches As Long
```

```
Dim matches As String * 200 '保留获取设备号
```

```
Dim strRes As String * 200
```

```
Private Sub Cmd_Read_Click()
```

```
' 获取命令返回状态
```

```
Call viVscanf(vi, "%t", strRes)
```

```
Txt_output.Text = strRes
```

```
End Sub
```

```
Private Sub Cmd_Send_Click()
```

```
' 发送询问命令内容
```

```
Call viVprintf(vi, Txt_input.Text + Chr$(10), 0)
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
' 获得 visa 的 usb 资源
```

```
Call viOpenDefaultRM(defrm)
```

```
Call viFindRsrc(defrm, "USB?* ", list, nmatches, matches)
```

```
' 打开设备
```

```
Call viOpen(defrm, matches, 0, 0, vi)
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
' 关闭资源
```

```
Call viClose(vi)
```

```
Call viClose(defrm)
```



End Sub

6.保存并运行整个工程，得到单个可执行文件。当仪器与 PC 机成功相连时，在 Input 下文本编辑框中输入一条命令，如\*IDN? (\*IDN?为默认输入命令)，按“Send”按键，再按“Read”按键，仪器返回的结果将显示在 Output 下文本编辑框中。

### 4.3 LabVIEW 8.5 编程实例

运行 LabVIEW 8.5 软件，并按如下步骤进行操作：

1. 进入 Getting Started 画面。选择 New>>Blank VI，新建一个 VI。如图 4-3-1 所示。

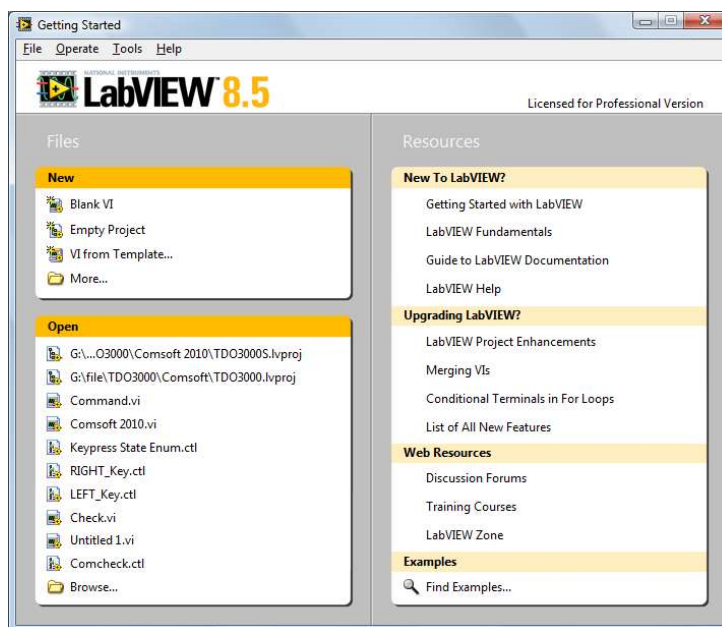


图 4-3-1

2. 从前面板选择 Controls>>Modern>>Boolean>> OK Button 放置 3 个按钮分别定义为 Write，Read 和 Stop。如图 4-3-2 所示。

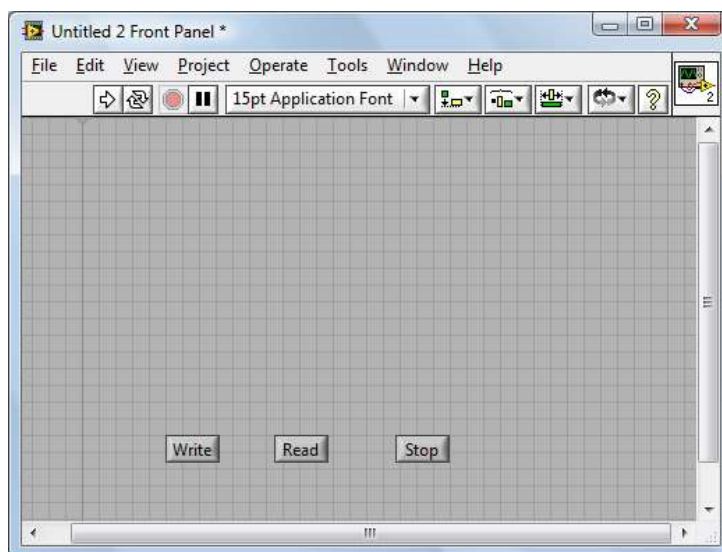


图 4-3-2

- 3.从程序框图的函数选板选择 Functions>>Programming>>Structures>>Event Structure 来放置一个 EventStructure 事件框。
4. 从程序框图右击事件框选择 Add Event Case..., 为 3 个按钮分别添加 Value Changed 事件。将按钮接线端分别放入各自的事件框。
- 5.选择 Write 按钮 ValueChange 事件框，从函数选板选择 Functions>>Instrument I/O>>VISA>>VISA Write 来放置一个 VISAWrite 函数到 Write 按钮的 ValueChange 事件框中。
6. 从函数选板选择 Functions>>Instrument I/O>>VISA>>VISA Advanced>>VISA Open 来放置一个 VISAOpen 函数到事件框的左侧。
7. 右键 VISAOpen 的 VISA resource name 接线端，在快捷键菜单中选择 Create>>Control 来创建一个输入控件 VISA resource name。
- 8.连线 VISA Open 的 VISA resource out 接线端和事件框内 VISAWrite 函数的 VISA resource name 接线端。连线 error out 和 error in 接线端。
- 9.右键 VISAWrite 的 writebuffer 接线端，在快捷键菜单中选择 Create>>Control 来创建一个输入控件 write buffer。如图 4-3-3 所示。

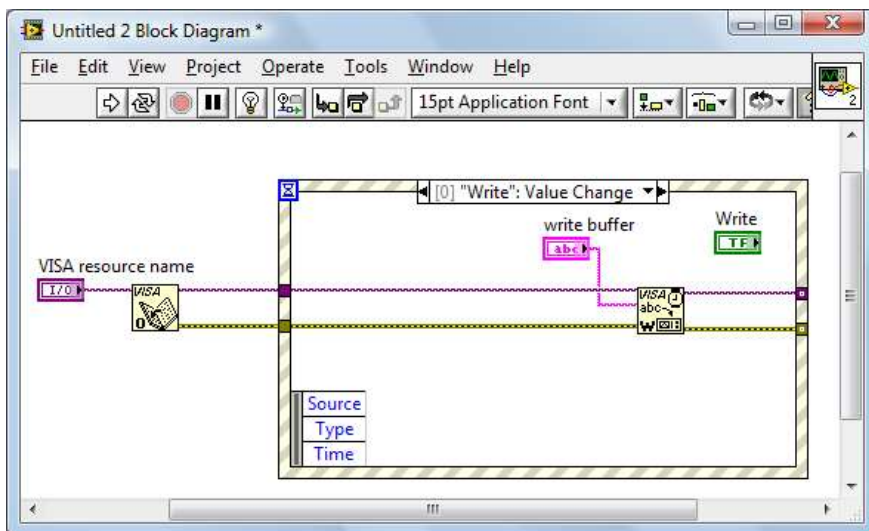


图 4-3-3

- 10.选择 Read 按钮 ValueChange 事件框，从函数选板选择 Functions>>Instrument I/O>>VISA>>VISA Read 来放置一个 VISARead 函数到 Read 按钮的 ValueChange 事件框中。
- 11.右键 VISARead 的 readbuffer 接线端，在快捷键菜单中选择 Create>>Indicator 来创建一个显示控件 read buffer。
- 12.右键 VISARead 的 bytecount 接线端，在快捷键菜单中选择 Create>>Constant 来创建一个常量。并将常量设为 1024。
- 13.连线 error 接线端，连线 VISA resource name 接线端。如图 4-3-4 所示。

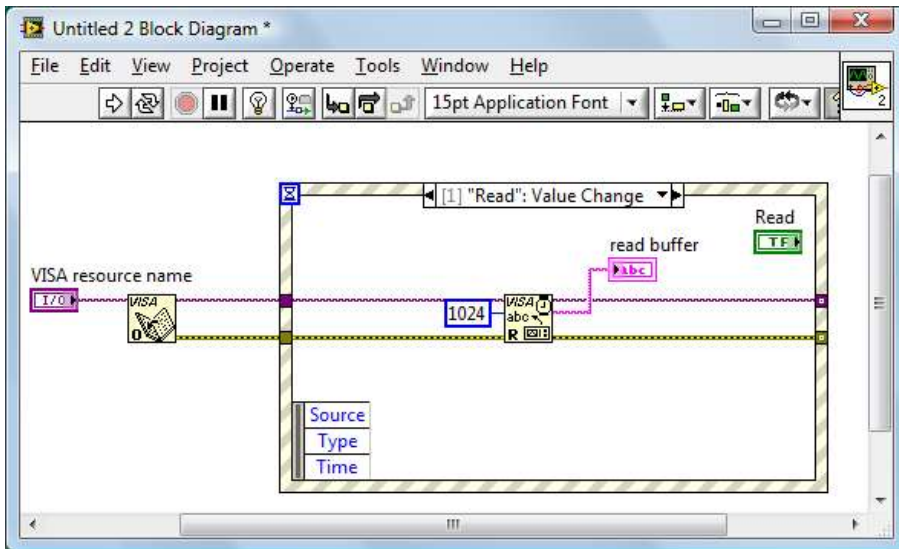


图 4-3-4

14. 选择 Stop 按键 ValueChange 事件框，从函数选板选择 Functions>>Instrument I/O>>VISA>>VISA Advanced>>VISA Close 来放置一个 VISA Close 函数到 Stop 按键的 ValueChange 事件框中。

15. 连线 error 接线端，连线 VISA resource name 接线端。

16. 从函数选板选择 Functions>>Programming>>Structures>>While Loop 来放置一个 While Loop 结构到事件框的周围。

17. 从函数选板选择 Functions>>Programming>>Boolean>>True Constant 来放置一个 True Constant 常量到 Stop 事件框的内。连接 TrueConstant 常量到 WhileLoop 循环结构的停止接线端。

18. 从函数选板选择 Functions>>Programming>>Dialog & User Interface>>Simple Error Handler 来放置一个 Simple Error Handler 函数。连线 error 接线端。

19. 右键 WhileLoop 循环结构与 error 连线的接线端 LoopTunnel，在快捷键菜单中选择 Replace with Shift Register 来创建循环移位寄存器对来替代循环隧道。同样也将 VISA resource name 的循环隧道替换成移位寄存器对。

20. 在前面板，调整控件的位置排列如图 4-3-5 所示。

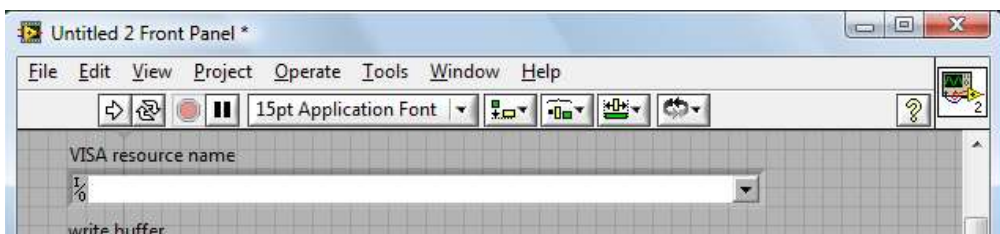


图 4-3-5

21.保存当前 VI。在运行本 VI 前从 VISA resource name 下拉框中选择正确的 VISA 资源名。

22.运行本 VI，在 writebuffer 中输入你想要发送的命令或查询，如：`*idn?`。点击 Write 键发送命令。然后按 Read 键读取返回信息。执行结果如图 4-3-6 所示。

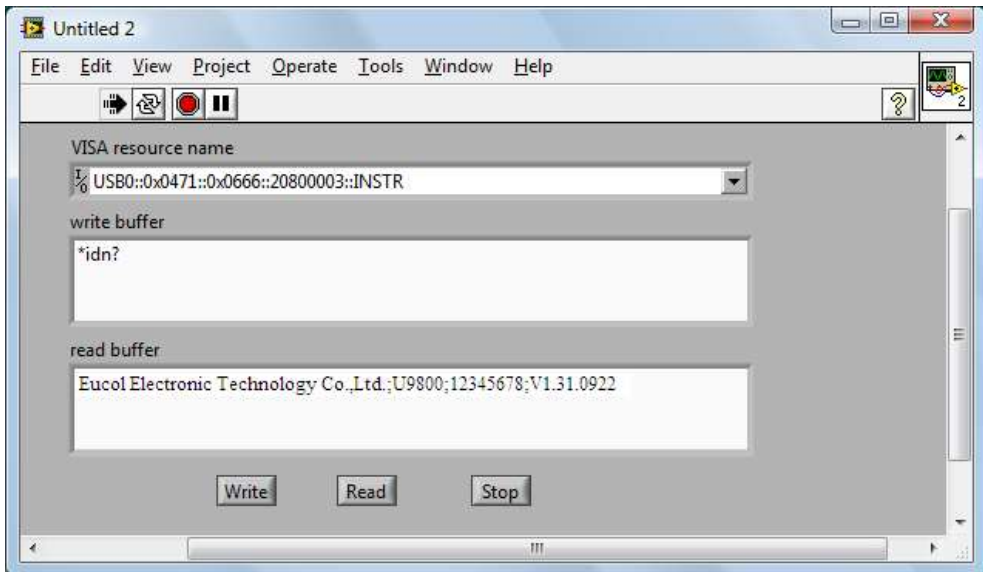


图 4-3-6

23.按 Stop 键退出程序执行。