



SCPI 协议编程手册

N3600 系列宽范围可编程直流电源

恩智（上海）测控技术有限公司

版本: V1.5

2023-12-18

目录

1. 通讯配置	3
2. 编程命令概述	4
2.1. 编程命令简介	4
2.2. 编程命令语法	4
2.2.1. 命令关键字	5
2.2.2. 分隔符	6
2.2.3. 查询指示符	6
2.2.4. 命令结束符	6
2.2.5. 参数格式	6
3. 命令详解	7
3.1. IEEE 488.2 公共命令	7
*IDN?	7
3.2. 设置输出电压与电流值	8
SOURce:VOLTage	8
SOURce:VOLTage:SLEW	8
SOURce:VOLTage:LIMit:HIGH	8
SOURce:VOLTage:LIMit:LOW	9
SOURce:CURREnt	9
SOURce:CURREnt:SLEW	9
SOURce:CURREnt:LIMit:HIGH	10
SOURce:CURREnt:LIMit:LOW	10
3.3. 输出控制及状态	10
OUTPut:ONOFF	10
OUTPut:MODE	11
OUTPut:EVENT	11
3.4. 读取输出电压电流及功率值	12
MEASure:VOLTage?	12
MEASure:CURREnt?	12
MEASure:POWer?	12
MEASure:MAXimum:VOLTage?	12
MEASure:MAXimum:CURREnt?	13
MEASure:MAXimum:POWer?	13
3.5. 保护功能	13
PROTect:VOLTage	13
PROTect:CURREnt	14
PROTect:POWer	14
3.6. 恒功率功能	14

CPOWer:VOLTage	14
CPOWer:CURRent	15
CPOWer:POWer	15
3.7. 外部编程功能	15
APROgram:MODE	15
CONFigure:APG:VOLTage	16
CONFigure:INHibit	16
3.8. 级联功能	16
CONFigure:MSSL:ID	16
CONFigure:MSSL:PARSER	17
CONFigure:MSSL:NUMSLV	17
CONFigure:MSSL:CONTRol	17
3.9. 序列运行功能	18
SEQuence:RUN:FILE	18
SEQuence:STATus	18
SEQuence:CYCLe	18
3.10. 序列编辑功能	19
SEQuence:EDITe:FILE	19
SEQuence:EDITe:LENGth	19
SEQuence:EDITe:CYCLe	19
SEQuence:EDITe:LFILE	19
SEQuence:EDITe:STEP	20
SEQuence:EDITe:VOLTage	20
SEQuence:EDITe:CURRent	20
SEQuence:EDITe:DWELl	20
SEQuence:EDITe:VSLEW	21
SEQuence:EDITe:CSLEW	21
4. 编程实例	22
4.1. 实例编程说明	22
4.1.1. 电压电流模式	22
4.1.2. 恒功率模式	22
4.1.3. SEQ 测试	23
4.1.4. 读取回显值	24
4.2. 出错信息	25
4.2.1. 语法错误	25
4.2.2. 执行错误	27

1. 通讯配置

通讯协议：标准 SCPI 通讯协议

通讯方式：LAN、串口 RS232

出厂默认 IP 地址：192.168.0.123

出厂默认设备 ID：160

TCP、UDP 端口号：7000

出厂默认串口波特率：9600（可设置，重启生效）

2. 编程命令概述

2.1. 编程命令简介

设备 SCPI 命令包括两种类型：IEEE488.2 公共命令和设备相关 SCPI 命令。

IEEE 488.2 公共命令定义了仪器仪表通用的一些控制和查询命令。可通过公共命令实现对设备的基本操作，如复位、状态查询等。所有 IEEE 488.2 公共命令由“*”和三个字母的助记符组成，如：*RST、*IDN?、*OPC?等。

SCPI 命令实现了对设备的大部分测试、设置、校准和测量等功能。此类命令以命令树形式组织。每个命令可包含多个助记符，命令树的各节点之间以字符“:”分隔，如图 4-1 所示。命令树顶端“ROOT”被称作“根”。从“根”到叶节点的全路径为一条完整的编程命令。

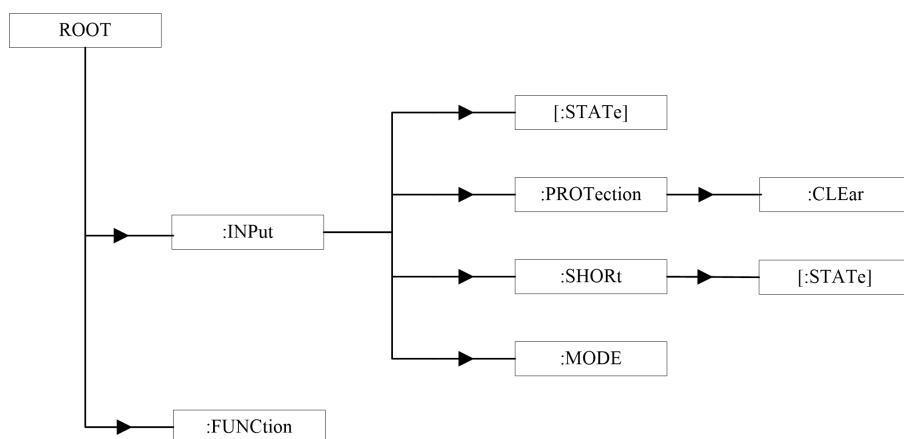


图 4-1 命令树示例

2.2. 编程命令语法

SCPI 命令是对 IEEE 488.2 命令规范的继承和扩充。SCPI 命令由命令关键字、分隔符、参数域和结束符等部分构成，以下面的命令为例：

STATic:CC:RANGe 1

在这条命令中，STATic、CC、RANGe 是命令关键字；字符“:”和空格为分隔符，1 为参数（部分命令具有多个参数，参数间以“,”分隔），命令后的回车符为命令结束符。

通道数据同时读或写则需要如下指令获取负载的通道板参数，格式例子为：

MEASure:VOLTage?(@1,2), STATic:CV:HIGH:LEVel 5(@1,2)

某一个通道数据读或写则需要如下指令获取负载的通道板参数，格式例子为：

MEASure3:VOLTage?, STATic3:CV:HIGH:LEVel 5

同时访问 1,2 通道的回显电压值，@后面的代表转发的通道号，通道与通道用', '号 隔开。

12 通道数据同时读，读取 1~12 通道的回显电压值，格式例子为：

MEASure:VOLTage?(@1,2,3,4,5,6,7,8,9,10,11,12)或

MEASure:VOLTage?(@1:12)

12 通道数据同时写，设定 1~12 通道的恒压值为 5V，格式例子为：

STATic:CV:HIGH:LEVel 5(@1,2,3,4,5,6,7,8,9,10,11,12)

或 STATic:CV:HIGH:LEVel 5(@1:12)

为方便描述，后续章节中各符号采用如下约定：

方括号 ([]) 表示可选关键字或参数，可省略。

花括号 ({ }) 表示命令串中的参数选项。

尖括号 (< >) 表示必须提供一个数值参数。

垂直线 (|) 用于分隔多个可选参数的选项。

2.2.1. 命令关键字

每个命令关键字均有长型助记符和短型助记符两种格式，短型助记符是长型助记符的缩写。每个助记符不超过 12 个字符（包括任何可能出现的数字后缀）。

N3600 只接受精确的长型或短型助记符。助记符的生成规则如下：

长型助记符由一个单词或短语构成。若是单词，则整个单词构成助记符。

CURRENT —— CURRent

短型助记符一般由长型助记符的前 4 个字符构成。

CURRent —— CURR

若长型助记符的字符长度小于等于 4，则长短型助记符相同；若长型助记符的长度大于 4，且第四个字符是元音，则其短型助记符将舍弃这个元音而变成 3

个字符。

MODE —— MODE

POWer —— POW

助记符不区分大小写。

2.2.2. 分隔符

✧ 冒号 “:”

分隔命令中相邻的两个关键字，如分隔编程命令“STATic:CV:HIGH:LEVel4”中的“STATic”和“CV”；也可为命令的第一个字符，表示将从命令树顶层节点开始寻路。

✧ 空格

分隔命令域与参数域。

2.2.3. 查询指示符

问号“?”用于标记命令的查询功能，标记位置为紧跟命令域的最后一个关键字。例如，要查询1通道CV恒压值，查询命令为“STATic:CV:HIGH:LEVel?”。假设恒压值为5V，则设备将返回字符串“5”。

设备收到查询命令并完成解析后，即执行该命令并生成响应字符串。响应字符串首先写入输出缓冲区，若当前远程接口为GPIB接口，则等待控制器读取该响应；否则，立即发送响应字符串到接口。

大部分设置命令都具有相应的查询语法。若收到不可查询的命令，设备将报告错误信息“-115 Command can not query”，且无任何返回。

2.2.4. 命令结束符

命令结束符有换行符（ASCII字符LF，数值为10）和EOI（仅适用于GPIB接口）两种格式。结束符的功能是结束当前命令串并将命令路径复位到根路径。

2.2.5. 参数格式

编程参数有数值型、字符型、布尔型等多种类型，详见表4-1，但无论哪种类型，均以ASCII码串表示。

表 4- 1 参数格式

符 号	说 明	例 子
<NR1>	整数数值。	123
<NR2>	浮点数值。	123., 12.3, 0.12, 1.23E4
<NRf>	数值可能是 NR1 或 NR2。	
<NRf+>	扩展的数值格式，包括<NRf>和 MIN、MAX。	
<Bool>	布尔数据。	1 0 ON OFF
<CRD>	字符数据，如 CURR。	
<AARD>	返回 ASCII 码数据。允许返回未定义的 7 位 ASCII 码。此数据格式隐含一个命令结束符。	

3. 命令详解

3.1. IEEE 488.2 公共命令

公共命令是 IEEE 488.2 规范要求仪器必须支持的通用命令，用于控制仪器的通用功能，如复位、状态查询等，其语法和语义遵循 IEEE 488.2 协议规范。IEEE 488.2 公共命令无层次结构。

注：公用命令目前只支持“*IDN?”。

*IDN?

读取直流电源相关信息。该命令的返回值包含三个以逗号分开的字符串，其含义依次为制造商、产品型号，保留区和软件版本号。

查询语法：*IDN?

参数：无

返回值：<AARD>	字符串	描述
	NGI	制造商
	N3600	产品型号
	0	保留区
	XX.XX	软件版本号

返回值示例：NGI,N3600,0,V1.00

3.2. 设置输出电压与电流值

SOURce:VOLTage

功能：设置输出电压值。

语法：SOURce:VOLTage <NR2>

参数：0-额定电压值。

举例：SOURce:VOLTage 10

表示设置输出电压为 10V。

查询：SOURce:VOLTage?

返回：10V

表示查询到输出电压值设置为 10V。

SOURce:VOLTage:SLEW

功能：设置电压斜率。

语法：SOURce:VOLTage:SLEW <NRf>

参数：0.001-5000V/s。

举例：SOURce:VOLTage:SLEW 5000

表示设置电压斜率为 5000V/s。

查询：SOURce:VOLTage:SLEW?

返回：5000V/s

表示查询到电压斜率设置为 5000V/s。

SOURce:VOLTage:LIMit:HIGH

功能：电压输出设置上限，防止用户误输入，以保护被测设备。

语法：SOURce:VOLTage:LIMit:HIGH <NRf>

参数：0-额定电压值。

举例：SOURce:VOLTage:LIMit:HIGH 25

表示设置电压输出上限为 25V。

查询：SOURce:VOLTage:LIMit:HIGH?

返回：25V

表示查询到电压输出上限为 25V。

SOURce:VOLTage:LIMit:LOW

功能：电压输出设置下限，防止用户误输入，以保护被测设备。

语法：SOURce:VOLTage:LIMit:LOW <NRf>

参数：0-额定电压值。

举例：SOURce:VOLTage:LIMit:LOW 10

表示设置电压输出下限为 10V。

查询：SOURce:VOLTage:LIMit:LOW?

返回：10V

表示查询到电压输出下限为 10V。

SOURce:CURREnt

功能：设置输出电流值。

语法：SOURce:CURREnt <NR2>

参数：0-额定电流值。

举例：SOURce:CURREnt 10

表示设置输出电流值为 10A。

查询：SOURce:CURREnt?

返回：10A

表示查询到输出电流值设定为 10A。

SOURce:CURREnt:SLEW

功能：设置电流斜率。

语法：SOURce:CURREnt:SLEW <NRf>

参数：0.001-2000A/s。

举例：SOURce:CURREnt:SLEW 2000

表示设置电流斜率为 2000A/s。

查询：SOURce:CURREnt:SLEW?

返回：2000A/s

表示查询到电流斜率设置为 2000A/s。

SOURce:CURRent:LIMit:HIGH

功能：电流输出设置上限，防止用户误输入，以保护被测设备。

语法：SOURce:CURRent:LIMit:HIGH <NRf>

参数：0-额定电流值。

举例：SOURce:CURRent:LIMit:HIGH 10

表示设置电流输出上限为 10A。

查询：SOURce:CURRent:LIMit:HIGH?

返回：10A

表示查询到电流输出上限为 10A。

SOURce:CURRent:LIMit:LOW

功能：电流输出设置下限，防止用户误输入，以保护被测设备。

语法：SOURce:CURRent:LIMit:LOW <NRf>

参数：0-额定电流值。

举例：SOURce:CURRent:LIMit:LOW 10

表示设置电流输出下限为 10A。

查询：SOURce:CURRent:LIMit:LOW?

返回：10A

表示查询到电流输出下限为 10A。

3.3. 输出控制及状态

OUTPut:ONOFF

功能：控制电源输出开关。

语法：OUTPut:ONOFF <Bool>

参数：0 | 1 | OFF | ON | 。

举例：OUTPut:ONOFF 1 或者 OUTPut:ONOFF ON

表示开启电源输出。

查询：OUTPut:ONOFF?

返回：ON

表示查询到电源开关状态为开启。

OUTPut:MODE

功能：设置电源工作模式。

语法：OUTPut:MODE <CRD>

参数：0|1|2|。

参数解释：0 为普通电压电流模式；1 为 SEQ 序列模式；2 为 CP 恒功率模式。

举例：OUTPut:MODE 0

表示设置电源为普通电压电流模式。

查询：OUTPut:MODE?

返回：0

表示查询到电源工作模式为普通电压电流模式。

OUTPut:EVENT

功能：获取电源告警状态及清除告警。

语法：OUTPut:EVENT?

返回：16

返回参数解释：

BIT4：OCP 告警，1 为有 OCP 告警，0 为无 OCP 告警。

BIT5：OVP 告警，1 为有 OVP 告警，0 为无 OVP 告警。

BIT6：OPP 告警，1 为有 OPP 告警，0 为无 OPP 告警。

BIT7：OTP 告警，1 为有 OTP 告警，0 为无 OTP 告警。

其他 BIT 位忽略。

返回 16 表示 OCP 告警。

16 转换为二进制为 b0001 0000 最右边为最低位 BIT0。

语法：OUTPut:EVENT 0

功能：清除告警。

3.4. 读取输出电压电流及功率值

MEASure:VOLTage?

功能：回读电压。

语法：MEASure:VOLTage?

参数：无。

举例：MEASure:VOLTage?

返回：<NRf>

单位：V。

MEASure:CURREnt?

功能：回读电流。

语法：MEASure:CURREnt?

参数：无。

举例：MEASure:CURREnt?

返回：<NRf>

单位：A。

MEASure:POWer?

功能：回读功率。

语法：MEASure:POWer?

参数：无。

举例：MEASure:POWer?

返回：<NRf>

单位：W。

MEASure:MAXimum:VOLTage?

功能：获取额定电压值。

语法：MEASure:MAXimum:VOLTage?

参数：无。

举例：MEASure:MAXimum:VOLTage?

返回：<NRf>

单位：V。

MEASure:MAXimum:CURRent?

功能：获取额定电流值。

语法：MEASure:MAXimum:CURRent?

参数：无。

举例：MEASure:MAXimum:CURRent?

返回：<NRf>

单位：A。

MEASure:MAXimum:POWer?

功能：获取额定功率值。

语法：MEASure:MAXimum:POWer?

参数：无。

举例：MEASure:MAXimum:POWer?

返回：<NRf>

单位：W。

3.5. 保护功能

PROTect:VOLTage

功能：设置及读取保护电压参数。

语法：PROTect:VOLTage <NR2>

举例：PROTect:VOLTage 10

解释：保护电压设置为 10V，当输出电压超过 10V 后，电源会关闭输出并提示 OVP 告警。

查询：PROTect:VOLTage?

返回：10V

单位：V。

PROTect:CURRent

功能：设置及读取保护电流参数。

语法：PROTect:CURRent <NR2>

举例：PROTect:CURRent 1

解释：保护电流设置为 1A，当输出电流超过 1A 后，电源会关闭输出并提示 OCP 告警。

查询：PROTect:CURRent?

返回：1A

单位：A。

PROTect:POWer

功能：设置及读取保护功率参数。

语法：PROTect:POWer <NR2>

举例：PROTect:POWer 10

解释：保护功率设置为 10W，当输出功率超过 10W 后，电源会关闭输出并提示 OPP 告警。

查询：PROTect:POWer?

返回：10W

单位：W。

3.6. 恒功率功能

CPOWer:VOLTag

功能：设置及读取恒功率电压参数。

语法：CPOWer:VOLTag <NR2>

举例：CPOWer:VOLTag 10

查询：CPOWer:VOLTag?

返回：10V

单位：V。

CPOWer:CURRent

功能：设置及读取恒功率电流参数。

语法：CPOWer:CURRent <NR2>

举例：CPOWer:CURRent 1

查询：CPOWer:CURRent?

返回：1A

单位：A。

CPOWer:POWer

功能：设置及读取恒功率功率参数。

语法：CPOWer:POWer <NR2>

举例：CPOWer:POWer 10

查询：CPOWer:POWer?

返回：10W

单位：W。

3.7. 外部编程功能

APRogram:MODE

功能：设置模拟编程模式。

语法：APRogram:MODE <NR1>

参数：0 | 1 | 2 | 3 | 。

参数解释：0 表示模拟编程设为 OFF；1 表示模拟编程设为 V；2 表示模拟编程设为 I；3 表示模拟编程设为 V and I。

举例：APRogram:MODE 1

表示设置模拟编程为电压编程

举例：APRogram:MODE?

返回：1

表示查询到模拟编程为电压编程

CONFigure:APG:VOLTage

功能：设置电源模拟编程参考电压。

语法：CONFigure:APG:VOLTage <NR1>

参数：0 | 1 | REF5 | REF10 | 。

举例：CONFigure:APG:VOLTage 1

表示设置电源模拟编程参考电压为 10V。

查询：CONFigure:APG:VOLTage?

返回：10V

表示查询到电源模拟编程参考电压为 10V。

CONFigure:INHibit

功能：设置电源输出外部控制方式。

语法：CONFigure:INHibit <NR1>

参数：0 | 1 | 2 | OFF | Toggle | Hold | 。

举例：CONFigure:INHibit 1

表示设置电源输出外部控制方式为 Toggle

查询：CONFigure:INHibit?

返回：Toggle

表示查询到电源输出外部控制方式为 Toggle

3.8. 级联功能

CONFigure:MSSL:ID

功能：设置电源在级联扩展时的身份属性，此参数亦作为级联通讯地址。

语法：CONFigure:MSSL:ID <NR1>

参数：0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 。

参数解释：0 表示设置电源在级联扩展时的身份属性为 **Master**；1 表示设置电源在级联扩展时的身份属性为 **Slave1**；2 表示设置电源在级联扩展时的身份属性为 **Slave2**。以此类推，9 表示设置电源在级联扩展时的身份属性为 **Slave9**。

举例：CONFigure:MSSL:ID 0

表示设置电源在级联扩展时的身份属性为 **Master**

查询: CONFigure:MSSL:ID?

返回: Master

表示设置电源在级联扩展时的身份属性为 Master

CONFigure:MSSL:PARSER

功能: 设置电源并联或串联。

语法: CONFigure:MSSL:PARSER <NR1>

参数: 0 | 1 | Parallel | Series | 。

参数解释: 0 表示并联; 1 表示串联。

举例: CONFigure:MSSL:PARSER 1

表示设置电源在级联时为串联。

查询: CONFigure:MSSL:PARSER?

返回: Series

表示查询到电源在级联时设置为串联。

CONFigure:MSSL:NUMSLV

功能: 设置级联扩展时从机的数目。

语法: CONFigure:MSSL:NUMSLV <NR1>

参数: 1~9。

举例: CONFigure:MSSL:NUMSLV 1

查询: CONFigure:MSSL:NUMSLV?

返回: 1

表示查询到电源在级联扩展时从机的数目设置为 1 台。

CONFigure:MSSL:CONTRol

功能: 使能或失能级联扩展。

语法: CONFigure:MSSL:CONTRol <NR1>

参数: 0 | 1 | OFF | ON | 。

举例: CONFigure:MSSL:CONTRol 0

查询: CONFigure:MSSL:CONTRol?

返回: off

表示查询到主从控制为失能级联扩展即主从控制为 off 状态。

3.9. 序列运行功能

SEQuence:RUN:FILE

功能：选择需运行的序列文件和获取当前运行的序列文件。

语法：SEQuence:RUN:FILE <NR1>

参数：1-100。

举例：SEQuence:RUN:FILE 1

解释：选择运行序列文件 1。

查询：SEQuence:RUN:FILE?

返回：1

解释：获取当前运行的序列文件编号为 1。

SEQuence:STATus

功能：查看序列运行的文件和运行工步。

查询：SEQuence:STATus?

返回：1,2

解释：其中 1 表示当前运行的文件，2 表示当前运行的工步

SEQuence:CYCLe

功能：获取序列运行当前文件的循环次数。

查询：SEQuence:RUN:FILE?

返回：1

解释：表示当前序列运行的循环次数为 1。第 1 次

3.10. 序列编辑功能

SEquence:EDITe:FILE

功能：选择需编辑的序列文件编号。

语法：SEquence:EDITe:FILE <NR1>

参数：1-100。

举例：SEquence:EDITe:FILE 1

查询：SEquence:EDITe:FILE?

返回：<NR1>

SEquence:EDITe:LENGth

功能：设置本序列文件大小。

语法：SEquence:EDITe:LENGth <NR1>

参数：1-100。

举例：SEquence:EDITe:LENGth 1

查询：SEquence:EDITe:LENGth?

返回：<NR1>

SEquence:EDITe:CYCLe

功能：设置本序列文件运行次数。

语法：SEquence:EDITe:CYCLe <NR1>

举例：SEquence:EDITe:CYCLe 1

查询：SEquence:EDITe:CYCLe?

返回：1

SEquence:EDITe:LFILE

功能：设置本序列文件链接到某序列文件。

语法：SEquence:EDITe:LFILE <NR1>

举例：SEquence:EDITe:LFILE 3

解释：假设本序列文件为 1，则运行完序列文件 1 后会继续运行序列文件 3。

查询: SEQuence:EDITe:LFILE?

返回: 3

SEQuence:EDITe:STEP

功能: 选择本序列文件的第 N 步。

语法: SEQuence:EDITe:STEP <NR1>

范围: 1-本序列文件总步数。

举例: SEQuence:EDITe:STEP 1

查询: SEQuence:EDITe:STEP?

返回: 1

SEQuence:EDITe:VOLTage

功能: 设置本序列文件第 N 步的输出电压值。

语法: SEQuence:EDITe:VOLTage <NR2>

范围: 1-额定电压值。

举例: SEQuence:EDITe:VOLTage 10

查询: SEQuence:EDITe:VOLTage?

返回: 10V

单位: V

SEQuence:EDITe:CURRent

功能: 设置本序列文件第 N 步的电流值

语法: SEQuence:EDITe:CURRent <NR2>

范围: 1-额定电流值

举例: SEQuence:EDITe:CURRent 10

查询: SEQuence:EDITe:CURRent?

返回: 10A

单位: A

SEQuence:EDITe:DWELl

功能: 设置本序列文件第 N 步的运行时间

语法: SEQuence:EDITe:DWEL1 <NR2>

举例: SEQuence:EDITe:DWEL1 100

查询: SEQuence:EDITe:DWEL1?

返回: 100.00

单位: 秒

SEQuence:EDITe:VSLEw

功能: 设置本序列文件第 N 步的电压斜率

语法: SEQuence:EDITe:VSLEw <NR2>

举例: SEQuence:EDITe:VSLEw 5000

查询: SEQuence:EDITe:VSLEw?

返回: 5000V/s

单位: V/s

SEQuence:EDITe:CSLEw

功能: 设置本序列文件第 N 步的电流斜率

语法: SEQuence:EDITe:CSLEw <NR2>

举例: SEQuence:EDITe:CSLEw 2000

查询: SEQuence:EDITe:CSLEw?

返回: 2000A/s

单位: A/s

4. 编程实例

本章将举例说明如何使用编程命令控制直流电源，包括正常恒压电流模式、恒功率模式和 SEQ 测试。

◎◎注意：

本章节部分编程命令后存在以“//”开始的注释语句，这些注释语句并不属于直流电源可以识别的语法范围，仅仅为方便理解对应的编程命令，因此在实际操作直流电源时不可输入包括“//”在内的注释语句。

4.1. 实例编程说明

4.1.1. 电压电流模式

例：将直流电源设置为普通电压电流模式，设定输出电压值为 10V，输出电流值为 1A，开启电源输出，编程命令如下：

```
OUTPut:ONOFF 0           //关闭电源输出
OUTPut:MODE 0             //设置电源进入正常恒压电流模式
SOURce:VOLTage 10         //设定输出电压为 10V
SOURce:CURREnt 1          //设定输出电流为 1A
OUTPut:ONOFF 1           //开启电源输出
```

4.1.2. 恒功率模式

例：将直流电源设置为恒功率模式，设定恒功率模式下输出电压值为 10V，输出电流值为 1A，设定恒功率为 10W，开启电源输出，编程命令如下：

```
OUTPut:ONOFF 0           //关闭电源输出
OUTPut:MODE 2             //设置电源进入恒功率模式
CPOWer:VOLTage 10         //设定恒功率模式下电压为 10V
CPOWer:CURREnt 1          //设定恒功率模式下电流为 1A
CPOWer:POWer 10           //设定恒功率模式下功率为 10W
```

OUTPut:ONOFF 1 //开启电源输出

4.1.3. SEQ 测试

SEQ 测试主要根据当前选择的运行文件判断步大小，按顺序根据设置的每步的输出参数进行顺序执行，序列文件之间也可以嵌套，可独立设置相应的循环模式。

例：将电源设置为 SEQ 序列模式输出，选择当前运行文件 1，总步数为 3，文件运行 1 次，序列编辑的各项设定输出参数如下列表：

序列文件	文件大小	运行次数	链接文件	步编号	输出电压	输出电流	单步延时	电压斜率	电流斜率
1	3	1	0	1	1V	1A	5s	5000V/s	2000A/s
				2	2V	1A	10s	5000V/s	2000A/s
				3	3V	1A	15s	5000V/s	2000A/s

OUTPut:ONOFF 0 //关闭电源输出

OUTPut:MODE 1 //设置电源进入 SEQ 模式

SEQuence:EDITe:FILE 1 //设定序列文件为 1

SEQuence:EDITe:LENGTh 3 //设定序列文件 1 的文件大小为 3

SEQuence:EDITe:CYCLe 1 //设定序列文件 1 的运行次数为 1

SEQuence:EDITe:LFILE 0 //设定序列文件 1 的链接文件为 0

SEQuence:EDITe:STEP 1 //设定序列文件 1 的步编号为 1

SEQuence:EDITe:VOLTagE 1 //设定序列文件 1 步编号 1 的电压为 1V

SEQuence:EDITe:CURREnt 1 //设定序列文件 1 步编号 1 的电流为 1A

SEQuence:EDITe:DWELl 5 //设定序列文件 1 步编号 1 的单步延时为 5s

SEQuence:EDITe:VSLEw 5000 //设定序列文件 1 步编号 1 的电压斜率为 5000V/s

SEQuence:EDITe:CSLEw 2000 //设定序列文件 1 步编号 1 的电流斜率为 2000A/s

SEQuence:EDITe:STEP 2 //设定序列文件 1 的步编号为 2

SEQuence:EDITe:VOLTagE 2 //设定序列文件 1 步编号 2 的电压为 2V

SEQuence:EDITe:CURREnt 1 //设定序列文件 1 步编号 2 的电流为 1A

SEQuence:EDITe:DWELl 10 //设定序列文件 1 步编号 2 的单步延时为 10s

SEQuence:EDITe:VSLEw 5000 //设定序列文件 1 步编号 2 的电压斜率为 5000V/s

SEQuence:EDITe:CSLEw 2000	//设定序列文件 1 步编号 2 的电流斜率为 2000A/s
SEQuence:EDITe:STEP 3	//设定序列文件 1 的步编号为 3
SEQuence:EDITe:VOLTage 3	//设定序列文件 1 步编号 3 的电压为 3V
SEQuence:EDITe:CURREnt 1	//设定序列文件 1 步编号 3 的电流为 1A
SEQuence:EDITe:DWELl 15	//设定序列文件 1 步编号 3 的运行时间为 15s
SEQuence:EDITe:VSLEw 5000	//设定序列文件 1 步编号 3 的电压斜率为 5000V/s
SEQuence:EDITe:CSLEw 2000	//设定序列文件 1 步编号 3 的电流斜率为 2000A/s
SEQuence:RUN:FILE 1	//设置电源运行序列文件 1
OUTPut:ONOFF ON	//开启电源输出

4.1.4. 读取回显值

MEASure:VOLTage?	//读取电源的回显电压值
MEASure:CURREnt?	//读取电源的回显电流值
MEASure:POWer?	//读取电源的回显功率值

4.2. 出错信息

4.2.1. 语法错误

-100	Command error	未定义的语法错误
-101	Invalid character	程序信息字符串中出现无效字符
-102	Syntax error	存在未被识别的命令或数据类型
-103	Invalid separator	要求为分隔符，但是发送了不是分割符的字符
-104	Data type error	当前数据类型与所要求的类型不符
-105	GET not allowed	在程序信息中收到群执行触发(GET)
-106	Semicolon unwanted	存在一个或几个多余的分号
-107	Comma unwanted	参数列表中存在一个或多个多余的逗号
-108	Parameter not allowed	参数的数量超出了命令要求的数量
-109	Missing parameter	参数的数量少于命令要求的数量，或未输入参数
-110	Command header error	未定义的命令头错误
-111	Header separator error	在命令头分隔符的地方使用了非分隔符字符
-112	Program mnemonic too long	命令助记符的长度超过 12 个字符
-113	Undefined header	收到的命令虽然在语法结构上合乎规定，但并未在此仪器中定义
-114	Header suffix out of rang	命令头后缀超出范围

-115	Command can not query	命令不存在查询形式
-116	Command must query	命令必须为查询形式
-120	Numeric data error	未定义的数值数据错误
-121	Invalid character in number	在数值数据中出现了不被当先命令所接受的数据字符
-123	Exponent too large	指数的绝对值超过 32000
-124	Too many digits	十进制数据除开前置 0 后，数据的长度超过 255 个字符
-128	Numeric data not allowed	在不接受数值数据元素的位置收到正确格式的数值数据元素
-130	Suffix error	未定义的后缀错误
-131	Invalid suffix	后缀未遵循 IEEE 488.2 中定义的语法，或后缀不适合 E5071C
-134	Suffix too long	后缀长度超过 12 个字符
-138	Suffix not allowed	向不允许附加后缀的数值元素附加后缀
-140	Character data error	未定义的关键字数据错误
-141	Invalid character data	在关键字数据元素中找到无效字符，或收到无效关键字
-144	Character data too long	关键字数据的长度超过 12 个字符
-148	Character data not allowed	在此仪器不接受关键字数据元素的位置收到格式正确的关键字数据元素
-150	String data error	未定义的字符串数据错误

-151	Invalid string data	本应出现字符串数据，但是由于某些原因，出现的字符串数据无效
-158	String data not allowed	在此仪器不接受字符串数据元素的位置收到字符串数据元素
-160	Block data error	未定义的块数据错误
-161	Invalid block data	预期出现数据块，但是由于某些原因，出现的数据块无效
-168	Block data not allowed	在此仪器不接受数据块元素的位置收到数据块元素
-170	Expression error	未定义的表达式错误
-171	Invalid expression	表达式数据元素无效。例如，括号不成对或使用非法字符
-178	Expression data not allowed	在此仪器不接受表达式数据元素的位置收到表达式数据元素
-180	Macro error	未定义的宏错误
-181	Invalid outside macro definition	在宏定义之外遇到宏参数占位符“\$”
-183	Invalid inside macro definition	宏定义时(*DDT,*DMC)，宏的内容语法错误
-184	Macro parameter error	宏定义内的命令有错误的参数个数或参数类型

4.2.2. 执行错误

-200	Execution error	生成了与执行相关、且此仪器无法指定错误信息的错误
------	-----------------	--------------------------

-220	Parameter error	未定义的参数错误
-221	Setting conflict	命令参数已通过解析，但由于当前设备状态而不能执行
-222	Data out of range	数据超范围
-224	Illegal paramter value	参数不是当前命令可选参数列表中的一个
-225	Out of memory	此仪器中的可用内存不足以执行所选操作
-232	Invalid format	非法的数据格式
-240	Hardware error	未定义的硬件错误
-242	Calibration data lost	校准数据丢失
-243	NO reference	没有参考电压
-256	File name not found	文件名没找到
-259	Not selected file	无可选的文件
-295	Input buffer overflow	输入缓存溢出
-296	Output buffer overflow	输出缓存溢出