

# **UT3200 系列多路温度测试仪 SCPI 协议**

**包含型号 UT3208、UT3216、UT3224 和 UT3232**

# 目录

目录.....	2
1. SCPI 命令参考.....	2
1.1 命令串解析.....	2
1.1.1 命令解析规则.....	2
1.1.2 符号约定和定义.....	3
1.1.3 命令树结构.....	3
1.2 命令和参数.....	3
1.2.1 命令.....	3
1.2.2 参数.....	3
1.2.3 分隔符.....	4
1.3 命令参考.....	4
1.3.1 MEAS 子系统.....	5
1.3.2 SYST 子系统.....	7
1.3.3 FETCH 子系统.....	8
1.3.4 ERROR 子系统.....	8
1.3.5 IDN 子系统.....	8

## 1. SCPI 命令参考

本章包含编程 UT3200 系列多路温度测试仪的 SCPI 命令的参考信息。

- 命令解析器——了解命令解析器的一些规则。
- 命令和参数——命令行的书写规则
- 命令参考

本章节提供了仪器使用的所有的 SCPI 命令，通过这些 SCPI 命令，可以完全控制仪器所有功能。

### 1.1 命令串解析

主机可以发送一串命令给仪器，仪器命令解析器在捕捉到结束符（\n）或输入缓冲区溢出后开始解析。

例如：合法的命令串：

```
AAA:BBB CCC;DDD EEE;:FFF
```

仪器命令解析器负责所有命令解析和执行，在编写程序前您必须首先对其解析规则有所了解。

#### 1.1.1 命令解析规则

1. 命令解析器只对 ASCII 码数据进行解析和响应。
2. **SCPI 命令串必须以 NL（'\n' ASCII 0x0A）为结束符**，命令解析器在收到结束符后或缓冲区溢出才开始执行命令串。
3. 如果指令握手打开，命令解析器在每接受到一个字符后，立即将该字符回送给主机，主机只有接收到这个回送字符后才能继续发送下一个字符。

4. 命令解析器在解析到错误后，立即终止解析，当前指令作废。
5. 命令解析器在解析到查询命令后，终止本次命令串解析，其后字符串被忽略。
6. 命令解析器对命令串的解析不区分大小写。
7. 命令解析器支持命令缩写形式，缩写规格参见之后章节。

### 1.1.2 符号约定和定义

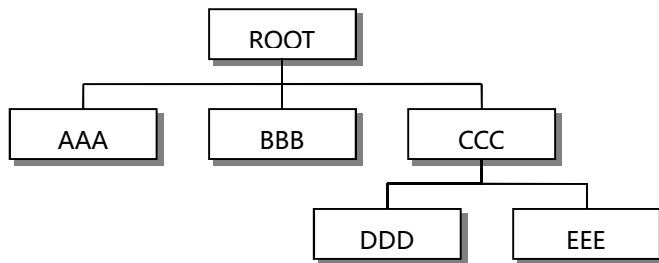
本章使用了一些符号，这些符号并不是命令树的一部分，只是为了能更好的对命令串的理解。

- <> 尖括号中的文字表示该命令的参数
  - [] 方括号中的文字表示可选命令
  - { } 当大括号包含几个参数项目时，表示只能从中选择一个项目。
  - () 参数的缩写形式放在小括号中。
- 大写字母 命令的缩写形式。

### 1.1.3 命令树结构

对 SCPI 命令采用树状结构的，可向下三级（注：此仪器的命令解析器可向下解析任意层），在这里最高级称为子系统命令。只有选择了子系统命令，该其下级命令才有效，SCPI 使用冒号 (:) 来分隔高级命令和低级命令。

图 1-1-3 命令树结构



#### 举例说明

```

ROOT:CCC:DDD PPP
ROOT  子系统命令
CCC   第二级
DDD   第三级
PPP   参数
  
```

## 1.2 命令和参数

一条命令树由 **命令和[参数]** 组成，中间用 1 个空格 (ASCII: 20H) 分隔。

#### 举例说明

```

AAA:BBB 1.234
命令    [参数]
  
```

### 1.2.1 命令

命令字可以是长命令格式或缩写形式，使用长格式便于工程师更好理解命令串的含义；缩写形式适合书写。

### 1.2.2 参数

- 单命令字命令，无参数。  
例如：AAA:BBB
- 参数可以是字符串形式，其缩写规则仍遵循上节的“命令缩写规则”。  
例如：AAA:BBB CCC

- 参数可以是数值形式
  - `<integer>` 整数 123, +123, -123
  - `<float>` 浮点数
    1. `<fixfloat>`: 定点浮点数: 1.23, -1.23
    2. `<Sciloat>`: 科学计数法浮点数: 1.23E+4, +1.23e-4
    3. `<mpfloat>`: 倍率表示的浮点数: 1.23k, 1.23M, 1.23G, 1.23u

表 1-2-2 倍率缩写

数值	倍率
1E18 (EXA)	EX
1E15 (PETA)	PE
1E12 (TERA)	T
1E9 (GIGA)	G
1E6 (MEGA)	MA
1E3 (KILO)	K
1E-3 (MILLI)	M
1E-6 (MICRO)	U
1E-9 (NANO)	N
1E-12 (PICO)	P
1E-15 (PEMTO)	F
1E-18 (ATTO)	A



提示: 倍率不区分大小写, 其写法与标准名称不同。

### 1.2.3 分隔符

仪器命令解析器只接受允许的分隔符, 除此之外的分隔符命令解析器将产生“Invalid separator(非法分割符)”错误。这些分隔符包括:

- ; 分号, 用于分隔两条命令。  
例如: AAA:BBB 100.0;CCC:DDD
- : 冒号, 用于分隔命令树, 或命令树重新启动。  
例如: AAA:BBB:CCC 123.4:DDD:EEE 567.8
- ? 问号, 用于查询。  
例如: AAA?
- 空格, 用于分隔参数。  
例如: AAA:BBB□1.234

主机可以发送一串命令给仪器, 仪器命令解析器在捕捉到结束符 (\n) 或输入缓冲区溢出后开始解析。

**例如:** 合法的命令串:

AAA:BBB CCC;DDD EEE::FFF

仪器命令解析器负责所有命令解析和执行, 在编写程序前您必须首先对其解析规则有所了解。

## 1.3 命令参考

所有命令都是按子系统命令顺序进行解释, 下面列出了所有子系统

- MEAS 测量设置子系统

- SYST 系统设置子系统
- FETCH 获取数据子系统
- ERROR 错误子系统
- IDN? 查询子系统

### 1.3.1 MEAS 子系统

MEAS 子系统用来设置测量页面

表 1-3-1 MEAS 命令树

Meas	:MODEL	{tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b}
	:RATE	{fast,med,slow}
	:KEYLOCK	{on,off}
	:START	{on,off}
	:CMODEL	<para>,<level>
	:CHANON	<para>,<on,off>
	:LOW	<level>
	:CLOW	<para>,<level>
	:HIGH	<level>
	:CHIGH	<para>,<level>
	:SENSOR	{tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b}
	:FONT	{fong24,font18,font16,font6x9}

#### MEAS:MODEL

MEAS:MODEL 用来设置传感器型号

Command Syntax	MEAS:MODEL <tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b>
Example	SEND> MEAS:MODEL TC-T<NL> //设置传感器型号为 T 型热电偶
Query Syntax	MEAS:MODEL?
Query Response	< tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b ><NL>
Example	SEND> MEAS:MODEL?<NL> RET> tc-t<NL>

#### MEAS:RATE

MEAS:RATE 用来设置采样速度

Command Syntax	MEAS:RATE <fast,med,slow>
Example	SEND> MEAS:RATE fast<NL> //设置采样速度为快速
Query Syntax	MEAS:RATE?
Query Response	<fast,med,slow><NL>
Example	SEND> MEAS:RATE?<NL> RET> fast<NL>

#### MEAS:KEYLOCK

MEAS:KEYLOCK 用来设置键盘锁

Command Syntax	MEAS:KEYLOCK <on,off>
Example	SEND> MEAS:KEYLOCK off<NL> //设置键盘锁关闭
Query Syntax	MEAS:KEYLOCK?
Query Response	<on,off><NL>
Example	SEND> MEAS:KEYLOCK?<NL> RET> on<NL>

## MEAS:START

MEAS :START 用来启动采样

Command Syntax	MEAS:START <on,off>
Example	SEND> MEAS:START off<NL> //设置采样结束
Query Syntax	MEAS:START?
Query Response	<on,off><NL>
Example	SEND> MEAS:START?<NL> RET> on<NL>

## MEAS:CMODEL

MEAS :CMODEL 用来设置各通道的传感器型号

Command Syntax	MEAS:MODEL <para>,<tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b>
Example	SEND> MEAS:CMODEL 0 TC-T<NL> //设置通道 001 的传感器为 T 型
Query Syntax	MEAS:CMODEL? //获取所有通道的传感器型号 MEAS:CMODEL? <int> //获取单通道传感器型号, 最小通道号 1
Query Response	<tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b><NL>
Example	SEND> MEAS:CMODEL?<NL> RET> <tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b><NL>  SEND> MEAS:CMODEL? 1<NL> //获取 001 通道的传感器型号 RET> <tc-t><NL>

## MEAS:CHANON

MAEAS:CHANON 用来设置各通道状态.

Command Syntax	MEAS:CHANON <para>,<integer>
Example	SEND> MEAS:CHANON 1, ON<NL> //设置通道 001 打开
Query Syntax	MEAS:CHANON?
Query Response	<on,off><NL>
Example	SEND> MEAS:CHANON?<NL> RET> <on,off,on,on,on,on,on,on,on><NL>

## MEAS:LOW

MAEAS:LOW 用来设置全通道下限值.

Command Syntax	MEAS:LOW <float>
Example	SEND> MEAS:LOW -200.0<NL> //设置所有通道下限为-200.0
Query Syntax	MEAS:LOW?
Query Response	<float,float><NL>
Example	SEND> MEAS:CLOW?<NL> RET> <-2.00000e+02, -2.00000e+02 ><NL>

## MEAS:CLOW

MAEAS:CLOW 用来设置各通道下限值.

Command Syntax	MEAS:CLOW <para>,<float>
Example	SEND> MEAS:CLOW 1, -200.0<NL> //设置通道 001 下限为-200.0

## MEAS:HIGH

MAEAS:HIGH 用来设置全通道上限值.

Command Syntax	MEAS:CHIGH <para>,<float>
Example	SEND> MEAS:CHIGH 1, 1800.0<NL> //设置所有通道上限为 1800.0
Query Syntax	MEAS:HIGH?
Query Response	<float,float><NL>
Example	SEND> MEAS:HIGH?<NL> RET> <1.80000e+03, 1.80000e+03 ><NL>

## MEAS:CHIGH

MAEAS:CHIGH 用来设置各通道上限值.

Command Syntax	MEAS:CLOW <para>,<float>
Example	SEND> MEAS:CHIGH 1, 1800.0<NL> //设置通道 001 下限为 1800.01

## MEAS:SENSOR

MAEAS:SENSOR 用来获取各通道分度号.

Query Syntax	MEAS: SENSOR
Query Response	< tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b ><NL>
Example	SEND> MEAS:SENSOR<NL> RET> < tc-t,tc-k,tc-j,tc-n,tc-e,tc-s,tc-r,tc-b ><NL>

## MEAS:FONT

MAEAS:FONT 用来设置显示页面.

Command Syntax	MEAS:FONT < font24,font18,font16,font6x9>
Example	SEND> MEAS:FONT font24<NL> //设置显示为 24 号字体

### 1.3.2 SYST 子系统

SYST 子系统用来设置 SETUP 页面

表 1-3-2 SYST 命令树

SYST	:COMP	{on,off}
	:BEEP	{on,off}
	:UNIT	{cel,kel,fah}

## SYST:COMP

SYST :COMP 用来设置比较器状态

Command Syntax	MEAS:COMP <on,off>
Example	SEND> SYST:comp on<NL> //设置比较器开
Query Syntax	MEAS:comp?
Query Response	<on,off><NL>
Example	SEND> SYST:comp?<NL> RET> on<NL>

## SYST:BEEP

SYST:BEEP 用来设置讯响状态

Command Syntax	MEAS:BEEP <on,off>
Example	SEND> SYST:beep on <NL> //设置蜂鸣器打开
Query Syntax	MEAS:beep?
Query Response	<on,off><NL>
Example	SEND> SYST:beep?<NL> RET> on<NL>

## SYST:UNIT

SYST:UNIT 用来设置温度单位

Command Syntax	MEAS:UNIT <cel, kel, fah>
Parameter	<cel, kel, fah> cel: 摄氏度 kel: 开尔文度 fah: 华氏度
Example	SEND> SYST:unit cel<NL> //设置温度单位为摄氏度
Query Syntax	MEAS:unit?
Query Response	<°C,K,F><NL>
Example	SEND> SYST:unit?<NL> RET> °C<NL>

### 1.3.3 FETCH 子系统

FETCH 子系统用来获取温度数据

表 1-3-3 FETCH 命令树

fetch?	
--------	--

## FETCH?

FETCH? 用来获取温度数据

Query Syntax	Fetch?
Query Response	<float,float,float><NL>
Example	SEND> fetch?<NL> RET> +1.00000e-05, +1.00000e-05, +1.00000e-05<NL>

### 1.3.4 ERROR 子系统

ERROR 子系统返回错误信息

Query Syntax	ERROR?
Query Response	Error string
Example	SEND> ERR?<NL> RET> no error<NL>

### 1.3.5 IDN 子系统

IDN?用来查询仪器 ID 号

Query Syntax	IDN? Or *IDN?
Query Response	<MODEL>,<Revision>,<SN>,< Manufacturer>



