

万用表、毫伏表编程手册

优利德科技（中国）股份有限公司

版本:

版本	修改项
V1.0	正式版
V1.1	添加” disp?;” 指令; 增加 UT8802\UT8802N、UT8803\UT8803N 的相关描述; 增加代码示例;

声明:

支持的机型:

支持机型	物理接口	通信接口	PID&VID	设备地址
UT632\UT632N	USB	HID	0xe008&0x1a86	[C:DM][D:T632][T:HID][PID:0xe008][VID:0x1a86]
UT803\UT803N	USB	HID	0xe008&0x1a86	[C:DM][D:T803][T:HID][PID:0xe008][VID:0x1a86]
UT804\UT804N	USB	HID	0xe008&0x1a86	[C:DM][D:T804][T:HID][PID:0xe008][VID:0x1a86]
UT805A\UT805N	USB	COM (串口)		[C:DM][D:T805A][T:COM][PORT:8][BAUD:9600] [PARITY:N][STOP:1][DATA:7]
UT8802\UT8802N	USB	HID	0xea80&0x10c4	[C:DM][D:T8802][T:HID][PID:0xea80][VID:0x10c4]
UT8803\UT8803N	USB	HID	0xea80&0x10c4	[C:DM][D:T8803][T:HID][PID:0xea80][VID:0x10c4]

注意: 表格中是 PID&VID, 而不是 VID&PID, 请注意顺序。

驱动说明:

USB-HID 通信接口的设备, 是免驱动的, 如果文档中注明的 USB-HID 设备通过 USB 线接入电脑识别到的是 USB 转串口的驱动, 请先卸载掉驱动, 然后使用 HID 方式驱动。

UT805A\UT805N 采用的是串口通信, 使用 USB 链接时, 需要安装 USB 转串口驱动, 在电脑上识别到 COM 口:

 Silicon Labs CP210x USB to UART Bridge (COM8)。

如何获取 USB 转串口驱动?

1. 请检查随机附带的产品手册中的驱动安装章节;
通常都是存放在随机附带的光盘或网盘中。
2. 第 1 步无法解决时, 也可以在线搜索 CP210X 的 USB 转串口驱动, 下载试用;
3. 如果以上两个步骤都无法解决, 请联系我们!

获取设备：

查询设备

如果想主动查询有哪些设备连接到电脑，请使用 UCI 库的查询接口：

- a) `uci_QueryNodes`
- b) `uci_QueryNodesX`

请根据需求选择合适的接口进行设备查询。查询接口的详细说明和代码示例可以参看文档《UCI 帮助文档.pdf》，也可以参看 `examples` 文件夹下的示例工程。

在填写查询设备信息时，要添加你想查询的设备 VID 和 PID，各机型的 VID 和 PID 请参看“支持的机型”章节中的表格。

注意：查询目前只支持基于 HID 通信的设备。COM 通信的还未添加查询机制。

订阅设备插拔通知

如果只是想感知设备的插拔，然后再做对应的打开、关闭等逻辑操作，可以使用 `uci` 的设备插拔订阅机制，而不是定时查询设备以获知设备是否在线。

相关 `uci` 接口，请参阅《UCI 帮助文档.pdf》的接口 `uci_SetAttribute` 和 `uci_SetNotify` 说明。

指令集：

所有指令文本都不区分大小写。

Data? - 读取当前测量值：

指令名	IO	数据	说明
<code>data?;</code>	只读	Double(8Bytes)	读取当前测量数据。

接口：

使用[读数据](#)接口。

注意：

对 `UT805A\UT805N` 接口的缓冲区大小可以为 8Bytes 也可以为 16Bytes，这是由于 `UT805A\UT805N` 是有主显和副显示，如果需要副显示就可以使用 16Bytes（2 个 `double` 类型数据依次存放）读取数据。其它机型使用 1 个 `double` 读取数即可。

Disp? - 读取显示信息:

指令名	IO	数据	说明
disp?;	只读	Double(8Bytes)	读取当前测量数据和状态信息，即当前屏幕上显示的读数。

数据结构:

```

//@brief : 万用表和毫伏表数据帧
//@remark:
struct DMFRM {
    //@brief : 主显字符串 (20个字符)
    TCHAR MainDisp[20];
    //@brief : 副显字符串或 (20个字符)
    TCHAR AuxDisp[20];
    //@brief : 主显数值 (8Bytes)
    double MainValue;
    //@brief : 副显数值 (8Bytes)
    double AuxValue;
    //@brief : 标记位 (8Bytes)
    unsigned long long Flags;
};

```

其中 `TCHAR` 定义如下:

编码方式	定义	说明
UNICODE	<code>wchar_t</code>	16 位 UNICODE 编码字符
ASCII	<code>char</code>	8 位 ASCII 编码字符

具体是指 `wchar_t` 或 `char`，请根据编译环境和 `uci.dll` 版本而定。

区别:

对比项	Data?	Disp?
数据	Double (8Bytes)	<code>struct DMFRM</code>
获取状态码	<code>uci_Read\ uci_ReadX</code> 接口返回值 > 0 时即为状态码。具体见 读数据 接口说明。	<code>DMFR.Flags</code> 是状态码，各比特位的意义请参照： 状态位列表
适用场景	只想获取浮点数类型的测量值和简单的状态信息	要获取浮点数类型数据和屏幕完全一致的显示数据和完整的状态信息

接口：

读数据：

```
u_status uci_Read(u_session _session, PRParams _params,
                 u_byte* _data, u_size _dataLen);
u_status uci_ReadX(u_session _session, u_cstring _msg, u_uint32 _timeout,
                  u_byte* _data, u_size _dataLen);
```

具体接口使用说明，请见《UCI 帮助文档.pdf》，这里重点对返回值进行说明：

返回值	意义
<0	错误码
>0	“Data?”：状态位编码，需转换位 32 位无符号整数后解析，具体见 状态位列表 。
	“disp?”：代表读取到的数据量，即 DMFRM 结构体的大小（Bytes）。
=0	不存在此种结果

示例：

在 VC 以 UNICODE 环境测试 UT8802N:

```
struct DMFRM
{
    TCHAR MainDisp[20];
    TCHAR AuxDisp[20];
    double MainValue;
    double AuxValue;
    unsigned long long Flags;
};
```

注意：UNICODE 环境下TCHAR就是宽字符，即一个字符两个字节；

```
#define Bits(_status, _offset, _mask) ((_status >> _offset) & _mask)
```

```
std::wstring& GetUnit(unsigned char _type, unsigned char _scale, std::wstring& _s)
{
    std::wstring unit[] = {
        _T("V"), _T("A"), _T("Ω"), _T("Hz"), _T("°C"), _T("°F"),
        _T(" "), _T("F"), _T("β"), _T("%"), _T(" ")
    };
    std::wstring scale[] = {
        _T("n"), _T("μ"), _T("m"), _T(" "), _T("k"), _T("M"), _T("G"), _T(" ")
    };
};
```

```

    _s = _T("");
    if (_scale < sizeof(scale) / sizeof(scale[0]))
        _s += scale[_scale];
    if (_type < sizeof(unit) / sizeof(unit[0]))
        _s += unit[_type];
    return _s;
}

void _main_UT8802N_cpp()
{
    double dv = 0.0;
    DMFRM dfrm;

    //////////////////////////////////////////disp?////////////////////////////////////
    //read
    u_status r = uci_ReadX(m_session, _T("disp?;"), 2000,
        (unsigned char*)&dfrm, sizeof(dfrm));
    if (r <= 0)
    {
        _tprintf(_T("[read : disp?]Error, r = %d; msg = %s\n"),
            r, uci_GetLastError());
        return;//error
    }

    //parse
    unsigned char unit_scale = Bits(dfrm.Flags, 8, 0xf);
    unsigned char unit_type = Bits(dfrm.Flags, 12, 0x7);
    std::wstring str_unit;
    GetUnit(unit_scale, unit_type, str_unit);

    //show
    _tprintf(_T("disp? : main = %s, aux = %s, mv = %f %s, av = %f, flags = 0x%x\n"),
        dfrm.MainDisp, dfrm.AuxDisp,
        dfrm.MainValue, str_unit.c_str(), dfrm.AuxValue, dfrm.Flags);

    //////////////////////////////////////////data?////////////////////////////////////
    r = uci_ReadX(m_session, _T("data?;"), 2000, (unsigned char*)&dv, sizeof(dv));
    if (r <= 0)
    {
        _tprintf(_T("[read : data?]Error, r = %d; msg = %s\n"), r, uci_GetLastError());
        return;//error
    }

    unit_scale = Bits(dfrm.Flags, 8, 0xf);

```

```

unit_type = Bits(dfrm.Flags, 12, 0x7);
GetUnit(unit_scale, unit_type, str_unit);

_tprintf(_T("data? : value = %f %s, flags = 0x%x\n"),
        dv, str_unit.c_str(), dfrm.Flags);
}

```

UT8802N 设备端显示的是： 50.23 Hz

程序打印：

disp? : main = 50.23, aux = , mv = 50.230000 Hz, av = 0.000000, flags = 0x2013345

data? : value = 50.230000 Hz, flags = 0x2013345

附录：

状态位列表：

“通用” 状态位列表适用于所有机型。

通用：

比特位	位数	意义	数据
D0~D3	4	功能编码	见 编码表
D4~D5	2	AC&DC 状态	见 编码表
D6	1	是否是 Auto Range	0: 不是; 1: 是。
D7	1	是否是 Over load(超出范围)	0: 不是; 1: 是。
D8~D11	4	物理单位类型	见 编码表
D12~D14	3	物理单位量级	见 编码表
D15	1	是否电池低电量	0: 不是; 1: 是。
D16	1	是否是 USB 通信	0: 不是; 1: 是。
D17	1	是否是 Under 状态	0: 不是; 1: 是。
D18	1	是否是 Over 状态	0: 不是; 1: 是。
D19	1	是否显示负数符号	0: 不显; 1: 显示。
D20~D23	4	档位编码	见 编码表 , 只在需要时使用。
D24~D27	4	小数点位置	以 1 起始的小数点位置
D28	1	是否是最大值(MAX)	0: 不是; 1: 是。{UT8802\UT8803 适用}
D29	1	是否是最小值(MIN)	0: 不是; 1: 是。{UT8802\UT8803 适用}
D30	1	是否相对测量值(REL)	0: 不是; 1: 是。{UT8802\UT8803 适用}
D31	1	是否是保持模式(HOLD)	0: 不是; 1: 是。{UT8802\UT8803 适用}

DMFR.Flags 高 32 位状态位列表:

此编码表只针对 UT8802\UT8802N 、 UT8803\UT8803N

比特位	位数	意义	数据
D0	1	错误标志	1: 当前数值错误或当前显示为 Err
D1	1	L , C 档位测量模式	1: 串行模式 , 对应屏幕显示 SER; 0: 并行模式 , 对应屏幕显示 PAL
D2	1	二极管, 晶闸管方向指示 => 从右到左	1 : 有效, 0 : 无效
D3	1	二极管, 晶闸管方向指示 => 从左到右	1 : 有效, 0 : 无效
D4	1	电感品质元素测量	1: 是; 0: 不是;
D5	1	等效电阻测量	1: 是; 0: 不是;
D6	1	电容损耗元素测量	1: 是; 0: 不是;
D7	1	电容-等效电阻测量	1: 是; 0: 不是;
D8~D15	8	档位	具体见: UT8802 功能编码 和 UT8803 功能编码
D16~D31	16	保留	
D31	1	是否是保持模式(HOLD)	0: 不是; 1: 是。{UT8802\UT8803 适用}

功能编码表:

下面给出通用档位编码和具体机型的编码表, 两种类型都可以完全代表文档所给出的机型的所有档位, 请根据实际情况选择。

通用:

此功能表编码是适用于所有机型的。

类型	编码
电压测量 (Voltage)	0
电阻测量 (OHM)	1
二极管测量 (DIODE)	2
电路通断测试(Continuity)	3
电容测量 (Capacitance)	4
频率(FREQ)	5
温度测试 华氏度(TEMPERATURE FAHRENHEIT)	6
温度测试 摄氏度(TEMPERATURE CENTIGRADE)	7
三极管 hFE 测量, 以 β 为单位(hFE)	8
电流测量(Current)	9
%(4-20mA)	10
占空比测量(Duty)	11
晶闸管测量(SCR)	12
电感测量 (包含 Q : “电感品质因素”、R : “等效电阻”)	13

注意:

功能编码不再区分 DCV、ACV、DCV+ACV 或者 DCI、ACI、DCI+ACI，而是抽象为电压和电流。
可以通过 AC&DC 状态码和单位类型和量级来区分细类别。

对于其它有细分类型的也是通过标记为来区分，比如电感测量的 Q,R 测量，有单独的标记为来区分。

UT8803\UT8803N

此功能表只适用于 UT8803\UT8803N，数据来自“disp?”指令数据的 DMFR.Flags 字段的高 32 位中的 D8~D15 标记位。

档位	编码
交流电压测量 (ACV)	0
直流电压测量 (DCV)	1
交流电流微安档 (AC μ A)	2
交流电流毫安档 (AC mA)	3
交流电流标准档 (AC A)	4
直流电流微安档 (DC μ A)	5
直流电流毫安档 (DC mA)	6
直流电流标准档 (DC A)	7
电阻测量 (OHM)	8
电路通断测试(Continuity)	9
二极管测量 (DIODE)	10
电感测量 (L)	11
电感测量 (Q)	12
电感测量 (R)	13
电容测量 (Capacitance) (C)	14
电容测量 (Capacitance) (D)	15
电容测量 (Capacitance) (R)	16
三极管 hFE 测量，以 β 为单位(hFE)	17
晶闸管测量(SCR)	18
温度测试 摄氏度(TEMPERATURE CENTIGRADE)	19
温度测试 华氏度(TEMPERATURE FAHRENHEIT)	20
频率(FREQ)	21
占空比测量(Duty)	22

UT8802\UT8802N

此功能表只适用于 UT8802\UT8802N，数据来自“disp?”指令数据的 DMFR.Flags 字段的高 32 位中的 D8~D15 标记位。

档位	编码
交流电流毫安档 (AC 2mA)	0x10
交流电流毫安档 (AC 20mA)	0x13
交流电流毫安档 (AC 200mA)	0x14
交流电流标准档 (AC 20A)	0x18
直流电流微安档 (DC 200 μ A)	0x0D
直流电流毫安档 (DC 2mA)	0x0E
直流电流毫安档 (DC 20mA)	0x11
直流电流毫安档 (DC 200mA)	0x12
直流电流标准档 (DC 2A)	0x16
频率 (FREQ Hz)	0x2B
频率 (FREQ kHz)	0x2C
频率 (FREQ MHz)	0x2D
占空比测量 (Duty)	0x22
电路通断测试 (Continuity)	0x24
二极管测量 (DIODE)	0x23
电阻测量 (OHM 200 Ω)	0x19
电阻测量 (OHM 2k Ω)	0x1A
电阻测量 (OHM 20k Ω)	0x1B
电阻测量 (OHM 200k Ω)	0x1C
电阻测量 (OHM 2M Ω)	0x1D
电阻测量 (OHM 200M Ω)	0x1F
三极管 hFE 测量, 以 β 为单位 (hFE)	0x25
晶闸管测量 (SCR)	0x2A
直流电压测量 (200mV)	0x01
直流电压测量 (2V)	0x03
直流电压测量 (20V)	0x04
直流电压测量 (200V)	0x05
直流电压测量 (1000V)	0x06
交流电压测量 (2V)	0x09
交流电压测量 (20V)	0x0A
交流电压测量 (200V)	0x0B
交流电压测量 (750V)	0x0C
电容测量 (Capacitance nF)	0x27
电容测量 (Capacitance μ F)	0x28
电容测量 (Capacitance mF)	0x29

物理单位类型编码表:

类型	编码
电压 (V)	0
电流 (A)	1
电阻 (Ω)	2
频率 (Hz)	3
摄氏度 ($^{\circ}\text{C}$)	4
华氏度 ($^{\circ}\text{F}$)	5
RPM(转速)保留	6
电容 (F)	7
三极管 hFE	8
百分比 (%)	9
不显示	0xf

物理单位量级编码表

类型	编码
n	0
μ	1
m	2
标准	3
K	4
M	5
G	6

AC 和 DC 状态编码表

类型	编码
OFF	0
AC	1
DC	2
AC+DC	3

量程编码表

UT805A\UT805N:

编码	DCV	ACV&ACV+DCV	DCI	ACI&ACI+DCI	OHM	CAP	FEQ	其它
0	200mV	200mV	2mA	2mA	200 Ω	6nF	6KHz	不分 量程
1	2V	2V	200mA	200mA	2K Ω	60nF	60KHz	
2	20V	20V	10A	10A	20K Ω	600nF	600KHz	
3	200V	200V			200K Ω	6 μ F	6MHz	
4	1000V	750V			2M Ω	60 μ F	60MHz	
5					20M Ω	600 μ F		
6						6mF		
7								
8								
9								

DCV : 功能编码为 0 (Voltage), AC&DC 编码为 2, 即为 DCV;

ACV : 功能编码为 0 (Voltage), AC&DC 编码为 1, 即为 ACV;

ACV+DCV : 功能编码为 0 (Voltage), AC&DC 编码为 3, 即为 ACV+DCV;

DCI : 功能编码为 9 (Current), AC&DC 编码为 2, 即为 DCV;

ACI : 功能编码为 9 (Current), AC&DC 编码为 1, 即为 ACV;

ACI+DCI: 功能编码为 9 (Current), AC&DC 编码为 3, 即为 ACI+DCI;

UT804\UT804N:

编码	ACV	DCV	OHM	CAP	℃	μ A	mA	10A	Diode	FREQ	℉
0	400mV	400mV	400	--	1000	400 μ	40mA	10A		40	1832
1	4V	4V	4K	40nF		4000 μ	400mA			400	
2	40V	40V	40K	400nF						4K	
3	400V	400V	400K	4 μ F						40K	
4	1000V	1000V	4M	40 μ F						400K	
5			40M	400 μ F						4M	
6				4mF						40M	
7				40mF						400M	
8											