

MPS-150601
MiniDAQ 数据采集器
使用说明

Ver. 1.1

第一章 产品概述



一、 产品简介

MPS-150601 (MiniDAQ) 是一款基于 USB 总线的精巧型信号采集器。MPS-150601 外形与 U 盘相似，但其内部集成有采集多种信号的高精度 AD 与一个复合功能的双向数字端口。MPS-150601 能够采集 0-10V 之间的电压、0-20mA 之间的电流、0-10K Ω 电阻或是 0-250KHz 的频率信号，是针对传感器多种不同的输出类型所设计的多功能信号采集器。MPS-150601 具备极佳的采样精度，内部使用 24 位超高分辨率的 AD 转换器，采样噪声被抑制在 0.005% 以内，同时内置高达 128 倍的可变增益放大器 (PGA)，小信号也可达到极佳的采集效果。MPS-150601 每秒钟最多可以采集 10 个样点，实际采样频率完全受计算机软件控制，可根据不同信号的测量要求灵活设定。此外，MPS-150601 还具有一个复合型的数字输入输出端口，既可以采集外部的高低数字电平信号，也可以用于对外输出高低电平，当输出高电平时还可以作为 5V 电源来驱动外部器件。在外观上，MPS-150601 美观小巧，便于携带，在提高工作效率的同时，不会造成工程师的额外负担，是随身携带的测量利器。

MPS-150601 采用 USB2.0 全速总线接口，无需外部电源，连接到计算机即可使用。同一台计算机上可以连接多个 MPS-150601，多个采集器既可以独立工作，也可受软件控制协调工作。

MPS-150601 采用跨平台的动态链接库作为驱动函数接口，可工作在 Win9X/Me、Win2000/XP、VISTA、WIN7、WIN8、WIN10 等常用操作系统下，支持 64 位系统，支持 VB, VC, C++Builder, C#, Dephi, LabVIEW, Matlab 等绝大多数编程语言，驱动函数简洁明晰，编程简单高效。随卡附送功能强大的应用软件，常用的功能可以直接实现，不懂编程的用户也可使用。

二、 性能指标

2.1、USB 总线

- USB2.0 全速总线传输
- 支持热插拔和即插即用

2.2、电压采集通道

- 输入信号：单端电压信号
- 输入端口：U - 电压输入端；GND - 地线输入端。
- 输入量程：0-10V

- 输入阻抗: $49\text{K}\Omega$
- 采样分辨率: 24bit
- 噪声波动幅度: $< \pm 0.0002\text{V}$ (PGA 倍数 = 1); $< \pm 0.00001\text{V}$ (PGA 倍数 = 128)
- 采样时间: $< 100\text{ms}$

2.3、电流采集通道

- 输入信号: 单端电流信号
- 输入端口: I - 电流正极输入端; GND - 电流负极 (地线) 输入端
- 输入量程: 0-20mA
- 输入阻抗: 100Ω
- 采样分辨率: 24bit
- 噪声波动幅度: $< \pm 0.0002\text{mA}$ (PGA 倍数 = 1); $< \pm 0.00002\text{mA}$ (PGA 倍数 = 128)
- 采样时间: $< 100\text{ms}$

2.4、电阻采集通道

- 输入信号: 开路电阻
- 输入端口: R - 电阻输入端一; GND - 电阻输入端二 (地线)
- 输入量程: 0-10K Ω
- 激励电流: 0.2mA
- 采样分辨率: 24bit
- 噪声波动幅度: $< \pm 0.2\Omega$ (PGA 倍数 = 1); $< \pm 0.02\Omega$ (PGA 倍数 = 128)
- 采样时间: $< 100\text{ms}$

2.5、频率采集通道

- 输入信号: 数字方波信号
- 输入端口: DIO - 数字信号输入端; GND - 地线输入端
- 输入量程: 20Hz-250KHz
- 输入电平: $0\text{V} \leq \text{低电平} \leq 1.5\text{V}$, $2.5\text{V} \leq \text{高电平} \leq 5\text{V}$
- 输入电流: $< 50\mu\text{A}$ (输入状态下 DIO 漏电流)
- 采样时间: $< 100\text{ms}$

2.6、数字电平采集通道

- 输入信号: 数字电平高低状态
- 输入端口: DIO - 数字信号输入端; GND - 地线输入端
- 输入电平: $0\text{V} \leq \text{低电平} \leq 1.5\text{V}$, $2.5\text{V} \leq \text{高电平} \leq 5\text{V}$
- 输入电流: $< 50\mu\text{A}$ (输入状态下 DIO 漏电流)
- 采样时间: $< 100\text{ms}$

2.7、数字电平输出

- 输出电平: 高电平 5V, 低电平 0V (典型值)
- 输出端口: DIO - 数字信号输出端; GND - 地线
- 输出电流: $< 100\text{mA}$ (高电平下 DIO 输出电流)
- 输入电流: $< 20\text{mA}$ (低电平下 DIO 输入电流)
- 采样时间: $< 100\text{ms}$

2.8、工作温度

- $-40^\circ\text{C} - 70^\circ\text{C}$

三、应用领域

传感器信号采集
变送器输出采集

四、 软件资源

Windows95/98/NT/2000/XP/VISTA/Window 7/Window 8/ Window 10/ Window 11 下的驱动程序；通用 DLL 动态链接库；LabVIEW、VB、VC 环境下的编程参考代码；配套应用软件等。

五、 配件清单

- [1] MPS-150601 信号采集器一套；
- [2] 保修卡一张；

六、 售后服务

一年保修。

第二章 设备安装

一、 MPS-150601 信号采集卡硬件接口

USB:	接至计算机的 USB 插口
GND:	采集器地线端口
U:	电压输入端口
I:	电流输入端口
R:	电阻输入端口
DIO:	复合型数字输入输出端口

二、 接口说明

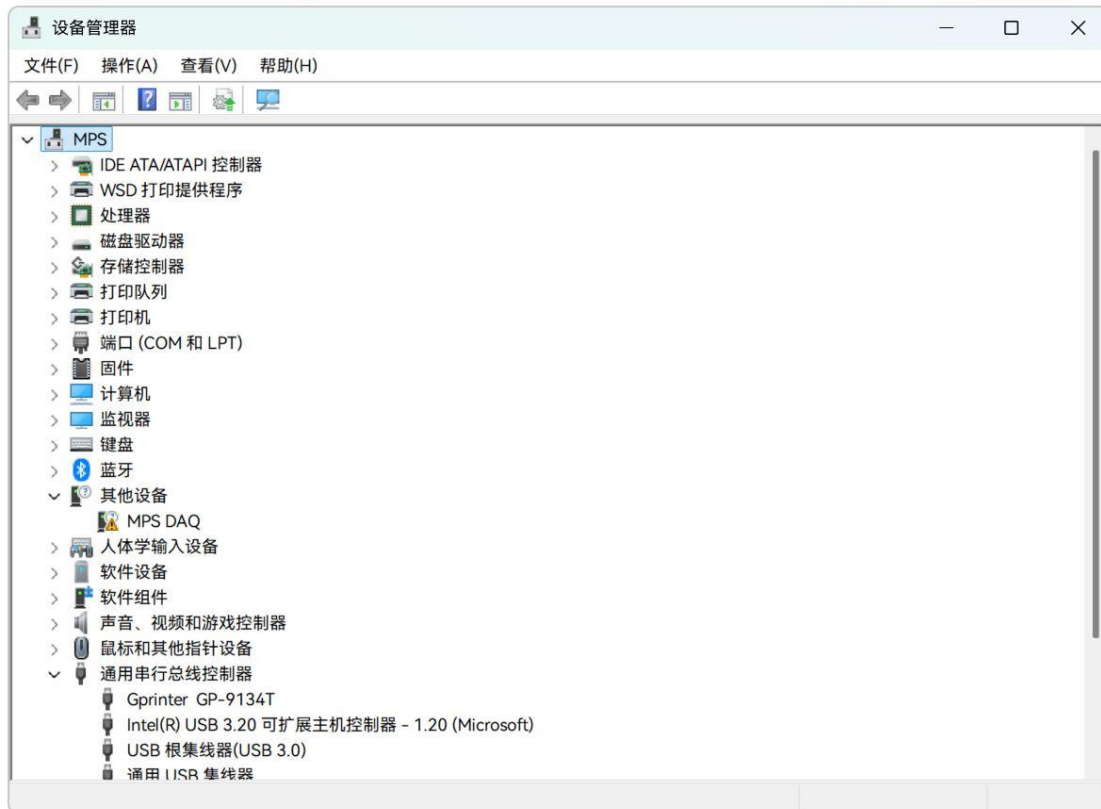
- 使用设备时请通过 USB 数据线将设备与计算机连接，设备正常连接后，可在 WINDOWS 的设备管理器中查询到设备信息。
- 设备的信号输入端与信号线接通，设备地线与信号源的地线接通。建议先将设备连接到计算机，再接入外部信号（或给信号源上电）。
- U 端口接入电压的范围为 0-10V，禁止接入超出该范围的电压或反向电压。
- I 端口接入电压的范围为 0-20mA，禁止接入超出该范围的电流或反向电流。
- R 端口接入电压的范围为 0-10K Ω ，超出该范围的电阻可接入，但无法测量。
- 复合型数字端口 DIO 配置为输入模式时，被内部弱上拉至 3.3V，端口输入电压在 2.5V 以上时识别为高电平，1.5V 以下时识别为低电平。
- 复合型数字端口 DIO 配置为 5V 输出时，内部与 USB 总线 5V 电源连通，最大输出 100mA 的电流。

三、 指示灯说明

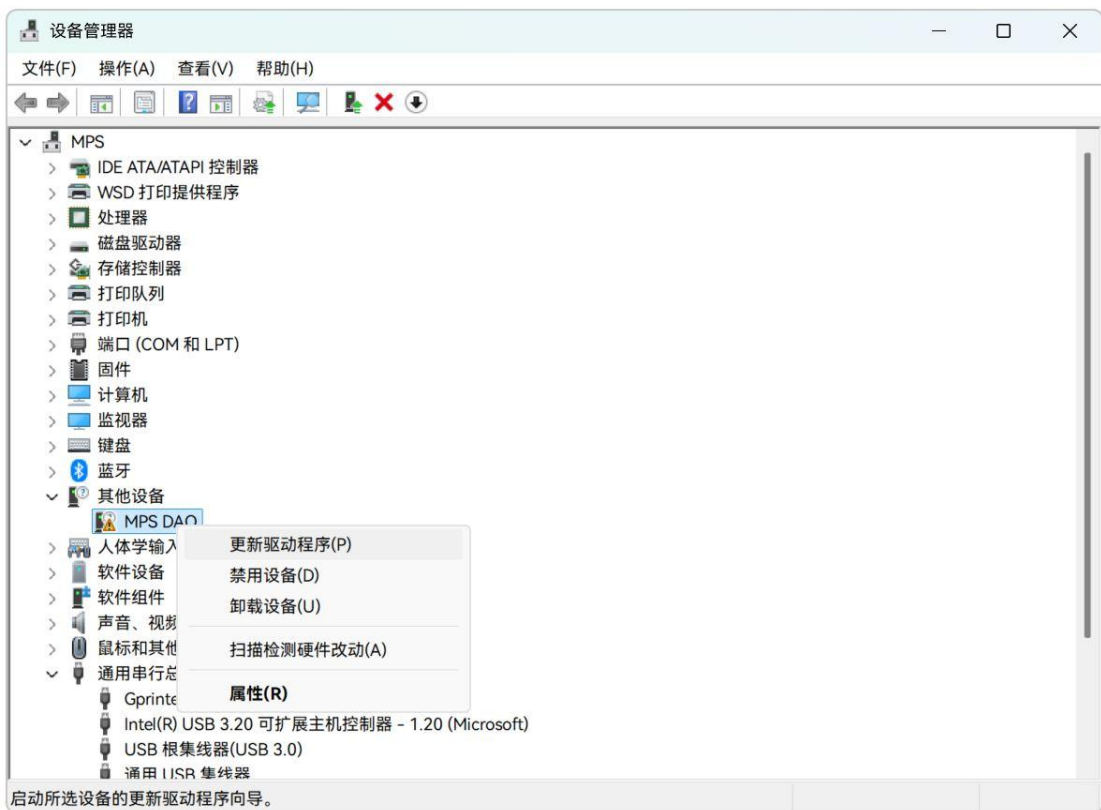
- MPS-150601 内置蓝色和绿色两只用于指示采集器的工作状态的指示灯。
- 上电时，蓝色指示灯亮起，说明采集器自检成功。当采集器进行数据采集时，绿色指示灯闪烁，当采集停止进入空闲状态时，绿色指示灯熄灭。

四、 驱动安装

1. 设备接入计算机 USB 后，在“此电脑”上点右键，选择“设备管理器”或者点击“属性”-“管理”-“设备管理器”来打开 WINDOWS 设备管理器，并在“其他设备”下找到本设备。



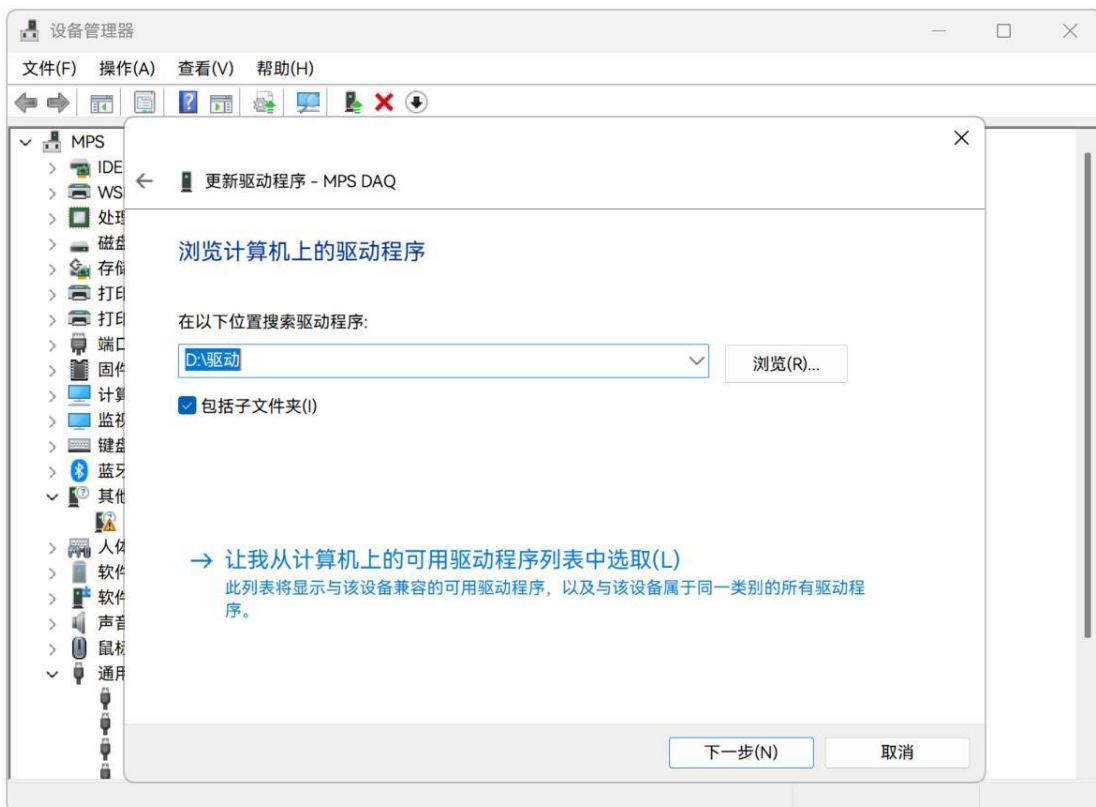
2. 在设备名称上点右键，选择“更新驱动程序”。



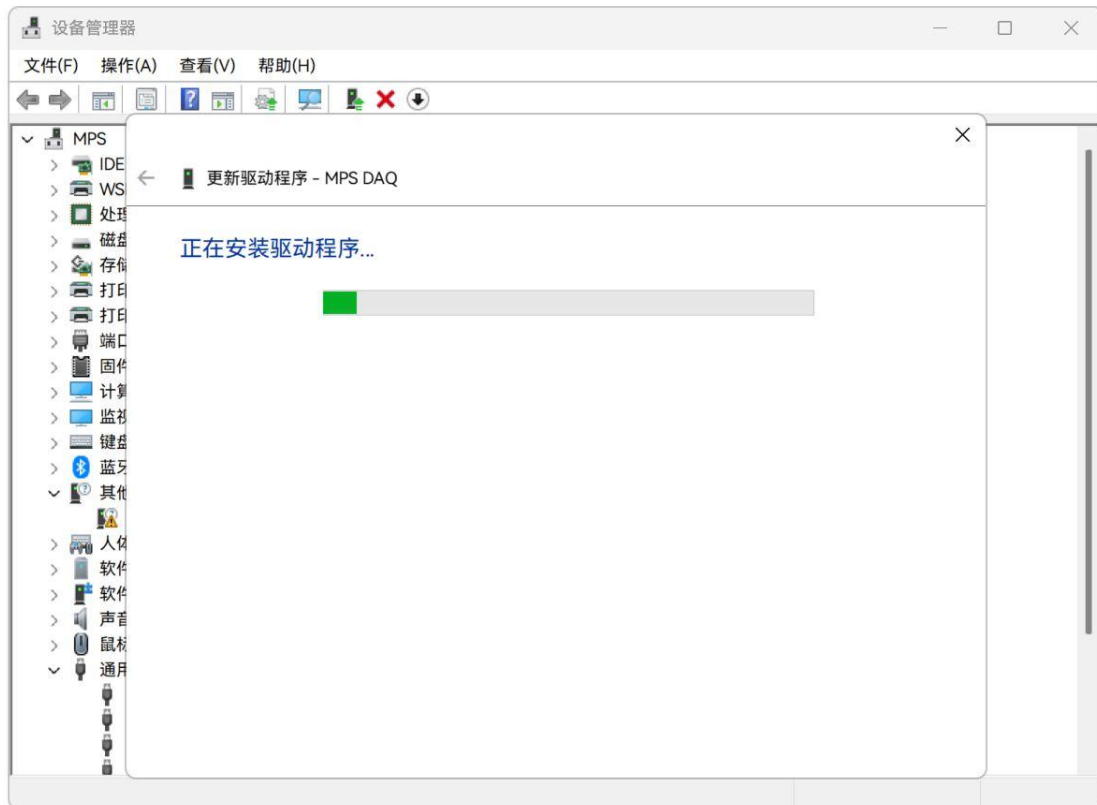
3. 在弹出的驱动安装向导界面中，选择“浏览我的电脑以查找驱动程序”。



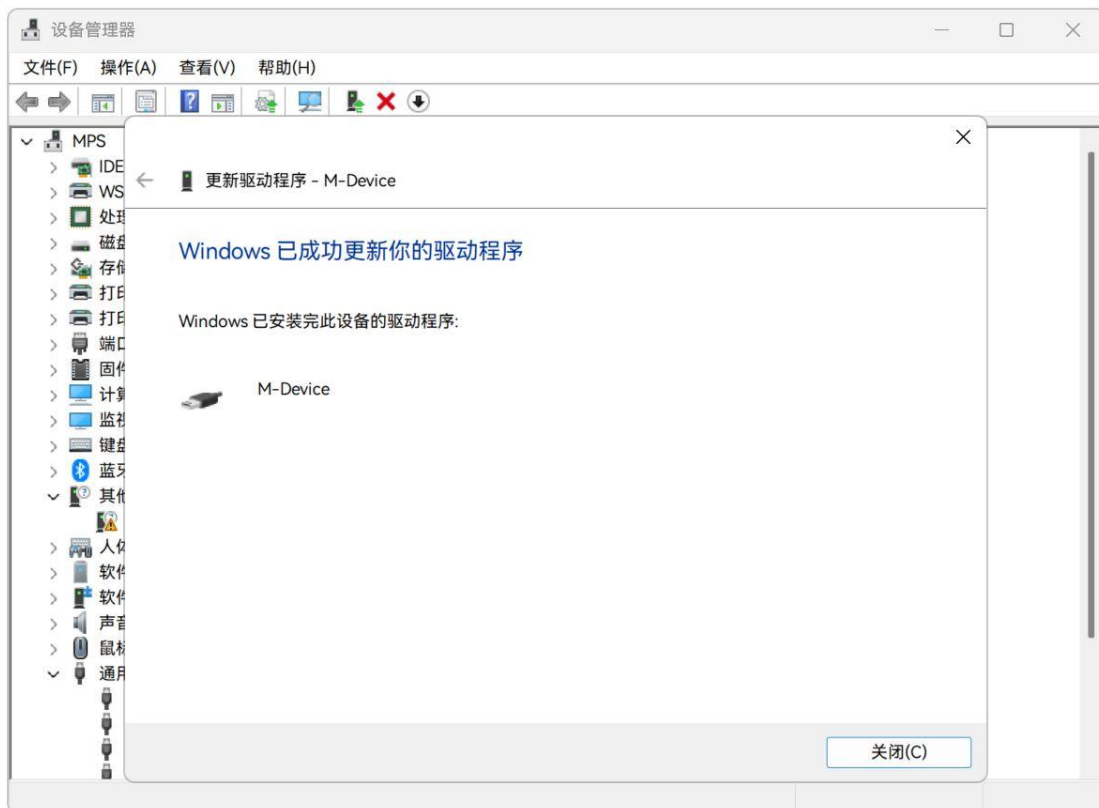
4. 点击“浏览”按钮，选择采集卡驱动所在的文件夹。如果驱动为压缩包，需要先解压缩后再进行操作。



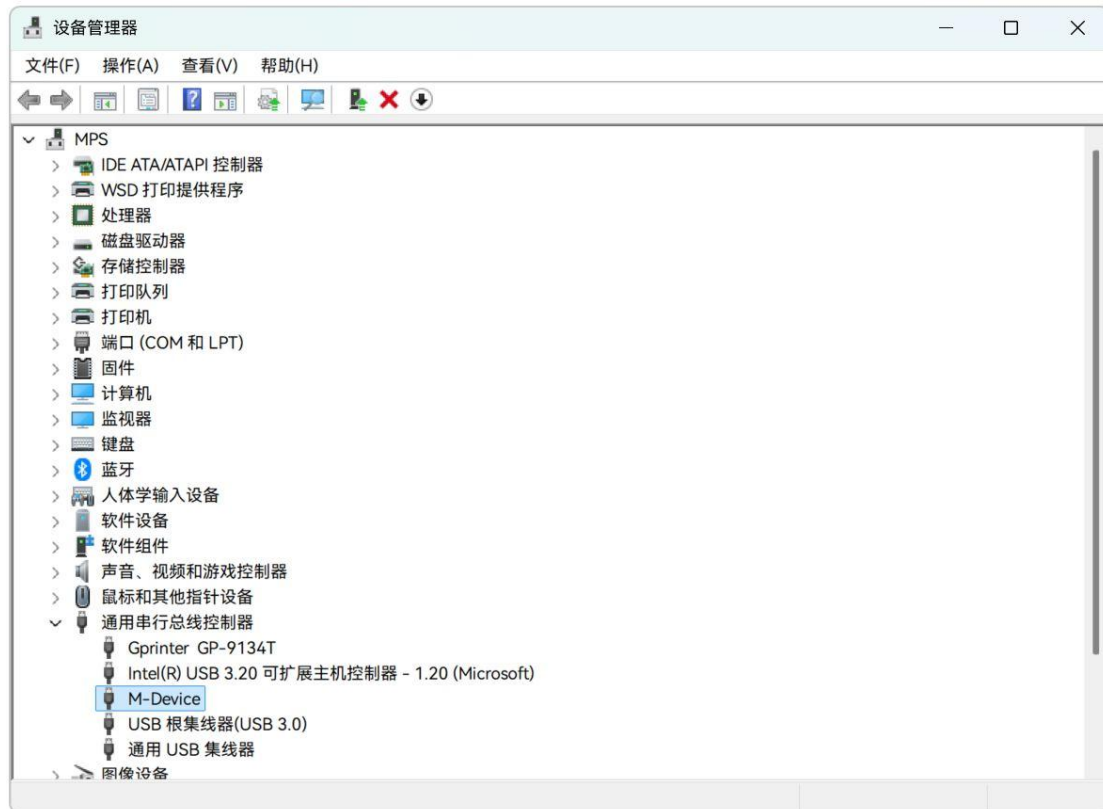
5. 点击下一步，出现如下驱动安装界面。过程中如果出现信任驱动的安全提示，勾选信任选项并点击确认。



6. 驱动安装完成，出现如下提示框。



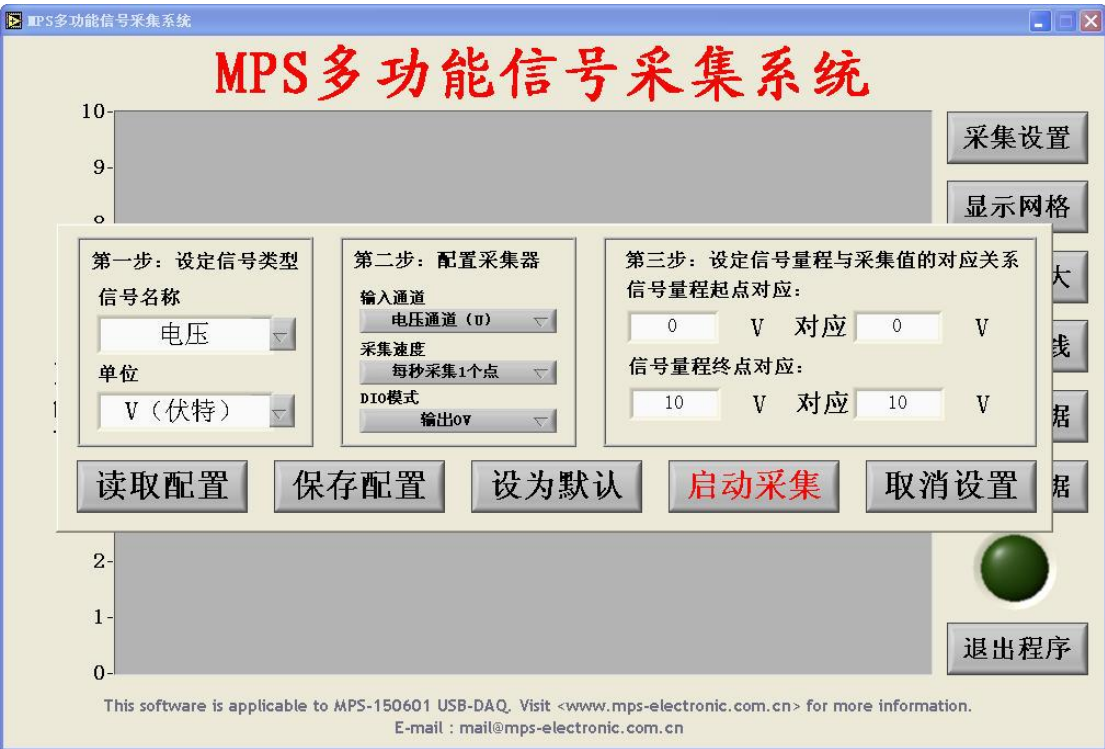
7. 驱动安装后，可在设备管理器 “通用串行总线控制器” 类目下看到设备，名为 “MPS-Device USB Data Acquisition”，驱动成功安装。



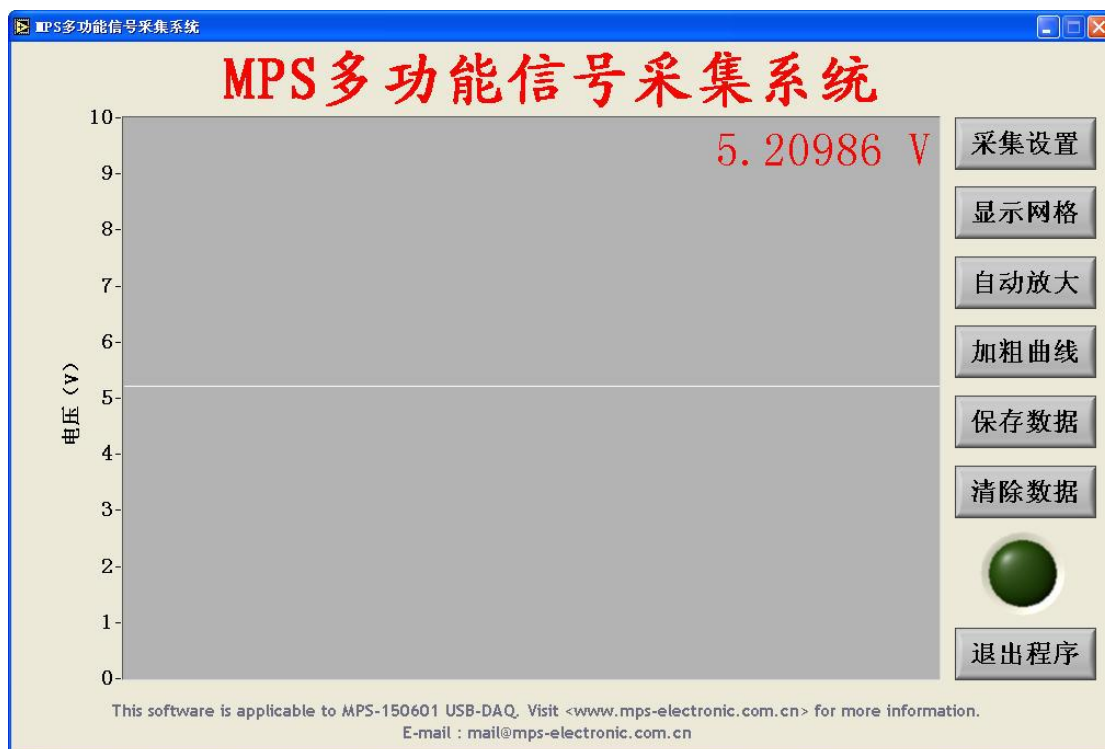
五、 功能测试



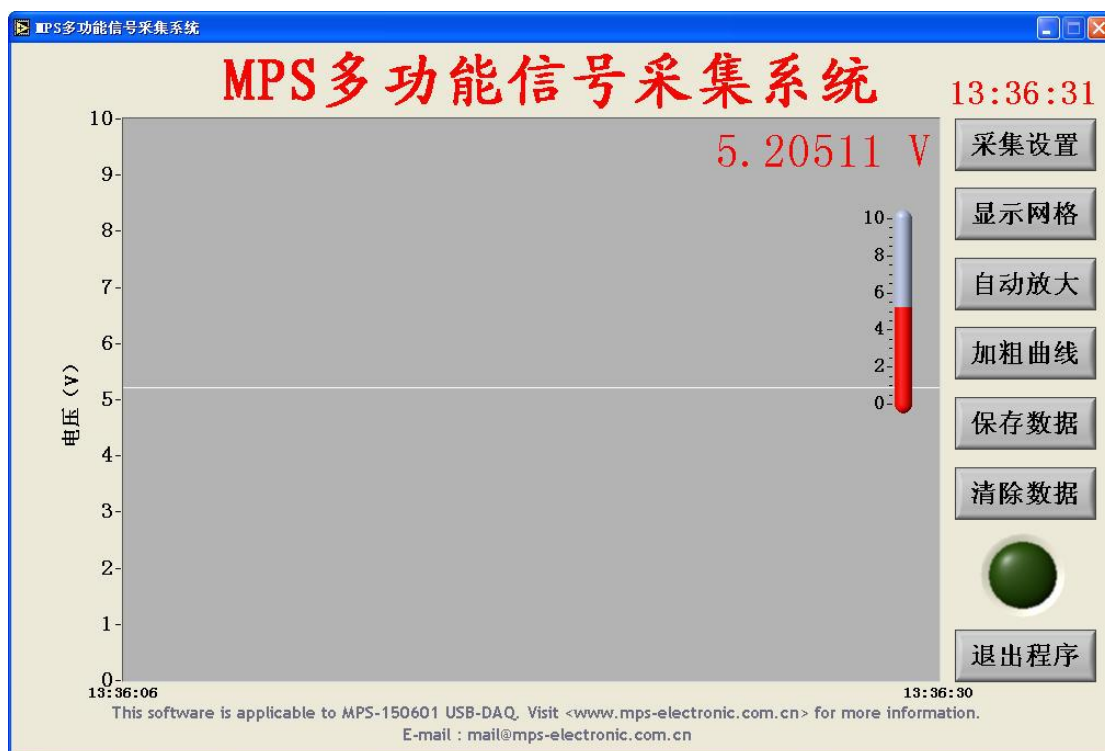
MPS-150601 数据采集软件 图 1



MPS-150601 数据采集软件 图 2



MPS-150601 数据采集软件 图 3



MPS-150601 数据采集软件 图 4

1. 将 MPS-150601 数据采集卡与计算机 USB 连接。
2. 按第三小节所描述的过程安装驱动程序。
3. 下载并解压安装 MPS 应用软件运行库 “MPSSoftwareRuntimeLib.rar”。
4. 打开测试程序 “MPS-150601 Data Acquisition V1.0.exe”， 出现如图 1 所示的界面。

5. 如计算机未连接 MPS-150601、驱动未安装或连接出错，软件会出现报错提示，此时请检查上述环节。
6. 将待测电压的信号线连接到 MPS-150601 标有 U 字样的输入端子，同时将待测电压的地线接到标有 GND 字样的端子。
7. 点击“采集设置”按钮，出现如图 2 所示界面，按照提示步骤，逐步设置信号类型、采集器配置和对应关系等。对于电压信号采集，使用默认设置即可。设定完成后，点击“启动采集”按钮。
8. 软件开始采集信号，于波形图框右上角显示采集到的电压数值，并于波形图中显示信号曲线。
9. 点击“显示网格”波形图增加坐标网格显示，再次点击取消显示。
10. 点击“自动放大”波形图 Y 轴显示的范围自动与信号的最大最小值进行匹配，再次点击恢复默认满量程。
11. 点击“加粗曲线”绘制波形的曲线会加粗。
12. 点击“保存数据”会将当前波形图中显示的所有数据保存至硬盘文件。
13. 点击绿色指示灯，会弹出自动报警功能设置框，启用报警功能后，当信号值达到设定值时就会触发报警，指示灯闪烁并播放报警提示音。
14. 点击“清除数据”清空当前波形图曲线，重新开始采集。
15. 点击“退出程序”关闭软件。
16. 在波形图右部点击，可调出图形化的数值显示，再次点击取消；在波形图下部点击，可调出 X 轴的时间信息，再次点击取消；在软件窗口的右上角部点击，可调出时间显示模块，再次点击取消。如图 4 所示。
17. 若需采集电流信号，将待测电流信号正极接入标有 I 字样的端子，电流信号负极接入标有 GND 字样的端子。并点击“采集设置”，将信号名称改为电流、输入通道改为“电流通道”，启动采集即可。
18. 若需采集电阻信号，将待测电阻两端分别接入标有 R 字样的端子和标有 GND 的端子。点击“采集设置”，将信号名称改为电阻、输入通道改为“电阻通道”，启动采集即可。
19. 若需采集数字频率信号，为确保 DIO 模式正确，建议首先点击“采集设置”，将信号名称改为频率、输入通道改为“频率通道”，并将“DIO 模式”改为输入模式，启动采集。然后将待测信号接入标有 DIO 字样的端子，地线接入 GND 端子，采集到频率结果后，再点击“清除数据”复位重新采集即可。
20. 若需采集数字电平信号，为确保 DIO 模式正确，建议首先点击“采集设置”，将信号名称改为电平、输入通道改为“电平通道”，并将“DIO 模式”改为输入模式，启动采集。然后将待测信号接入标有 DIO 字样的端子，地线接入 GND 端子，采集到电平状态后，点击“清除数据”复位重新采集即可。
21. 若传感器的输出电信号与对应的物理量之间为线性关系，可在采集设置的窗口内修改信号量程与采集值的对应关系，将采集值直接换算为物理量。首先在选择物理量的名称和单位（如不在列表内也可以自定义输入），然后根据传感器所标注的变换关系，确定待测物理量变化的起点与终点，并找出与之对应的传感器输出电信号的起点与终点，并将其分别填入采集设置的窗口对应的输入框，点击启动采集即可。此时软件会根据设定自动进行线性运算，并将运算结果进行显示。

第三章 用户编程

一、 动态链接库 (DLL)

MPS-150601 采用 DLL (Dynamic Linkable Library, 动态链接库) 的方式来进行编程驱动。DLL 的编制与具体的编程语言及编译器无关, 只要遵循约定的 DLL 接口规范和调用方式, 用各种语言编写的 DLL 都可以相互调用。

DLL 可以方便的在 VC、VB、LabVIEW 等语言下被调用, 具体方式分别为:

- VC 下调用 DLL

```
typedef void ( * FUNC )(void);           //定义一个函数指针
FUNC Func;                               //定义一个函数指针变量
HINSTANCE hDLL=LoadLibrary("DllTest.dll"); //加载 dll
Func=(FUNC)GetProcAddress(hDLL, "FuncInDLL"); //找到 dll 中的函数
Func();                                  //调用 dll 里的函数
```

- VB 下调用 DLL

```
[Public | Private] Declare Function name Lib " libname " [Alias "
aliasname " ] [(arglist)] [ As type] "
```

Public (可选) 用于声明在所有模块中的所有过程都可以使用的函数; **Private** (可选) 用于声明只能在包含该声明的模块中使用的函数。

Name (必选) 任何合法的函数名。动态链接库的入口处 (entry points) 区分大小写。

Libname (必选) 包含所声明的函数动态链接库名或代码资源名。

Alias (可选) 表示将被调用的函数在动态链接库 (DLL) 中还有另外的名称。当外部函数名与某个函数重名时, 就可以使用这个参数。当动态链接库的函数与同一范围内的公用变量、常数或任何其它过程的名称相同时, 也可以使用 **Alias**。如果该动态链接库函数中的某个字符不符合动态链接库的命名约定时, 也可以使用 **Alias**。

Aliasname (可选) 动态链接库。如果首字符不是数字符号 (#), 则 **aliasname** 是动态链接库中该函数入口处的名称。如果首字符是 (#), 则随后的字符必须指定该函数入口处的顺序号。

Arglist (可选) 代表调用该函数时需要传递参数的变量表。

Type (可选) **Function** 返回值的数据类型; 可以是 **Byte**、**Boolean**、**Integer**、**Long**、**Currency**、**Single**、**Double**、**Decimal** (目前尚不支持)、**Date**、**String** (只支持变长) 或 **Variant**, 用户定义类型, 或对象类型。

arglist 参数的语法如下:

```
[Optional] [ByVal | ByRef] [ParamArray] varname [()] [As type]
```

Optional (可选) 表示参数不是必需的。如果使用该选项, 则 **arglist** 中的后续参数都必需是可选的, 而且必须都使用 **Optional** 关键字声明。如果使用了 **ParamArray**, 则任何参数都不能使用 **Optional**。

ByVal (可选) 表示该参数按值传递。

ByRef (可选) 表示该参数按地址传递。

- LabVIEW 下调用 DLL

在 LabVIEW 中, 调用 DLL 是通过 CLF 节点来完成的。所谓 CLF 节点 (Call Library Function, 调用函数库节点), 是指可以在 LabVIEW 调用其它语言封装的 DLL, CLF 节点位于位于 LabVIEW 功能模板中的 **Advanced** 子模板中, 其配置过程如下:

- 在 CLF 节点的右键菜单中选择“Configure”，弹出 CLF 节点配置对话框；
- 点击“Browse”按钮，在随后弹出的选择 DLL 文件对话框中找到你需要用的 DLL 文件，此时，LabVIEW 就会自动装载选定的 DLL 文件，并检测 DLL 文件中所包含函数。但是函数中的参数和参数的数据类型需要用户根据函数的输入、输出参数手动设置。因而在调用 DLL 文件时，要求用户对 DLL 文件有较为详细的了解。
- 在 FunctionName 下拉列表框中选定动态连接库中所包含的所需要 API 函数；
- 在 Calling Convention 下拉菜单中选择 StdCall (WINAPI) 和 C 两个选项，若用户选定的是 Windows API 函数，则选用 StdCall (WINAPI) 选项；若用户选用的 DLL 中的函数是非 Windows API 函数，则选用 C 选项；
- 设置函数的返回参数。函数参数的类型要与 DLL 中函数本身所定义的函数参数类型相对应，如果不对应，函数就会出现数据错误和强制类型转换；
- 根据所选函数的函数原型，设置函数的输入参数及数据类型。点击 Add a Parameter 按钮，即可以添加一个新的输入参数。

二、编程函数及参数

MPS-150601 提供的驱动 DLL 文件名为 MPS-150601.dll，软件程序员只需调用该 DLL 内的驱动函数即可实现对板卡的控制，编程与打包发布时该 DLL 必须被包含在工程内。

MPS-150601.dll 内部的驱动函数如下：

- `int __stdcall MPS_GetData(int CH, int DIO, int PGA, double *Data, int DeviceNumber)`

`int MPS_GetData` 函数执行采集数据的功能。每调用一次本函数，获取一个采样样点。函数运行成功返回 1，运行失败返回 0。

`int CH` 通道设置参数。本参数设置本次函数执行采集的通道。

CH = 0 对应采集电压通道；

CH = 1 对应采集电流通道；

CH = 2 对应采集电阻通道；

CH = 3 对应采集 DIO 通道频率；

CH = 4 对应采集 DIO 通道电平状态。

CH 为其它值时，自动校正为 CH = 0。

`int PGA` 增益设置参数。本参数设置本次采集中模拟量采集的输入信号预放大（增益）倍数。PGA 参数设定后，采集器会根据当前放大倍数对输入信号进行预先放大，使小信号更接近采集器满量程，从而充分利用采集器性能，获得良好的采集效果。

当采集模拟信号通道（电压、电流或电阻通道）时：

PGA = 0 为不对信号进行预放大，采集量程为默认量程；

PGA = 1 为对输入信号进行 2 倍放大，采集量程为默认量程的 1/2；

PGA = 2 为对输入信号进行 4 倍放大，采集量程为默认量程的 1/4；

PGA = 3 为对输入信号进行 8 倍放大，采集量程为默认量程的 1/8；

PGA = 4 为对输入信号进行 16 倍放大，采集量程为默认量程的 1/16；

PGA = 5 为对输入信号进行 32 倍放大，采集量程为默认量程的 1/32；

PGA = 6 为对输入信号进行 64 倍放大，采集量程为默认量程的 1/64；

PGA = 7 为对输入信号进行 128 倍放大，采集量程为默认量程的 1/128。

PGA 为其它值时，自动校正为 PGA = 0。

无论 PGA 设定为多少，函数获得的采样结果均为输入信号的真实值而非放大后的

值，内部已对放大后信号的采集值进行了相应的缩小还原处理。

注：当采集数字信号通道（DIO 通道频率或 DIO 通道电平状态）时，该参数无效。

int DIO 复合功能数字 IO 口（DIO）模式设置。本参数设置本次函数执行时 DIO 的模式，DIO 状态在采集前即被设置。

DIO = 0 对应 DIO 口设置为输出低电平，此时 DIO 口对外输出 0V；

DIO = 1 对应 DIO 口设置为输出高电平，此时 DIO 口对外输出 5V；

DIO = 3 对应 DIO 口设置为输入模式，此时 DIO 口为带 3.3V 弱上拉的高阻态输入模式，可读取接入到该端口上的外部电平状态值。

设置为其它值，自动校正为 DIO = 2。

注：当 CH = 4 或 CH = 5 时，数字端口被自动设置为输入模式，DIO 参数值无效。

double *Data 数据存储变量 Data 所在的地址（指针）。若 MPS_GetData 函数执行成功，函数向 Data 内写入采集到的数据；若函数执行失败，则 Data 内的数据无效。

写入 Data 的数据为一个浮点数，其数值为采样点的值：

若当前采集电压通道，Data 中的值为采集到的电压值，单位为伏特（V）；

若当前采集电流通道，Data 中的值为采集到的电流值，单位为毫安（mA）；

若当前采集电阻通道，Data 中的值为采集到的电阻值，单位为欧姆（ Ω ）（若 Data 值为-1，代表当前接入电阻超出量程）；

若当前采集频率通道，Data 中的值为采集到的频率值，单位为赫兹（Hz）；

若当前采集电平通道，Data 中的值为采集到的电平状态，其值为 0 或 1，分别对应高电平和低电平。

int DeviceNumber 当前操作对应的设备序号。本参数用于设定当前操作所针对的设备序号。只连接一只采集器时，设备序号默认为 0。当有多套 MPS-150601 设备同时连接到计算机时，将按照设备连接到计算机的先后顺序从 0 开始依次分配序号，设定不同的序号值，可针对不同的采集器进行操作。

三、 编程范例

```
int MPS_TEST()
{
    // 声明DLL中的驱动函数
    HINSTANCE hDll; // 打开DLL
    hDll=LoadLibrary("MPS-150601.dll");
    if(NULL==hDll)
    {
        AfxMessageBox("Can't find DLL");
        return 0;
    }

    typedef int(_stdcall *lpMPS_GetData)( int CH, int PGA, int DIOModal, double *Data, int DeviceNumber); // 声明函数
    lpMPS_GetData MPS_GetData = (lpMPS_GetData)GetProcAddress(hDll, "MPS_GetData");
    if(NULL == MPS_GetData)
    {
        AfxMessageBox("Can't find <MPS_GetData> function");
    }
}
```

```
}

// 定义变量
int DeviceNumber = 0;           // 设备序号变量.
int CH = 0;                     // 通道参数变量
int PGA = 0;                     // 增益参数变量
int DIO = 2;                     // DIO模式变量
double Data = 0;                 // 数据变量
int FLAG = 0;                     // 函数执行结果标志变量

// 采集数据
while(1)                         // 测试用循环
{
    FLAG = MPS_GetData(CH, PGA, DIO, &Data, DeviceNumber); // 执行函数，采集一个样点
    if(FLAG == 0)                                           // 函数执行失败报错
    {
        AfxMessageBox("Error");
        return 0;
    }
    else
    {
        // 添加数据处理代码
    }
}
return 1;
}
```

第四章 注意事项

- 拔插设备接线端口请用力适度，以免损害接口。
- 信号输入端只允许接入量程范围内的信号，禁止接入超出量程的信号。
- 当 DIO 设置为输出高电平时，务必防止 DIO 与 GND 短路，否则可能损害计算机 USB 接口。当 DIO 设置为输出低电平时，务必防止 DIO 与外部电源短路，否则可能烧毁采集器。
- 使用中注意电源的开关顺序，尽量先将采集器连接到计算机，后开启信号源电源；关闭时先关信号源电源，后断采集器与计算机连接。
- 将采集卡拔离计算机后，请间隔一定时间再重新插入。拔插过快有可能造成计算机系统配置驱动异常，此时请重新拔插采集卡。
- 本设备属于精密电子仪器，请注意妥善保管，注意防尘防潮与防静电。
- 禁止用户自行打开设备外壳，一经打开即失去保修资格，出现的故障与其它后果由用户自行承担。
- MPS-150601 自出厂之日起一年内，凡用户遵守贮存、运输和使用要求，由于产品质量导致的故障，凭保修卡免费维修。因违反操作规定和使用要求而造成损坏的，需交纳器件维修费。
- MPS 系列信号采集卡由 Morpheus Electronic 提供，更多产品和相关信息请浏览：www.mps-electronic.com.cn, 咨询邮箱 mail@mps-electronic.com.cn。