

MPS 驱动库 V4.0 函数说明

Ver. 4.0.0.x 版本 MPS Driver.dll 支持向下兼容，可直接替换旧版本的 DLL，进行无缝升级。

与旧版本相比，Ver. 4.0.0.x 版本对 MPS_OpenDevice 函数与 MPS_DataIn 函数进行了参数升级，并新增专业参数设置函数 MPS_Configure_Pro。

1、MPS_OpenDevice 函数

本函数功能定义与旧版相同，在旧版基础上增加对 BIN 文件仿真模式的支持。仿真模式通过 DeviceNumber 参数的取值设置。

DeviceNumber 的参数取值新增对-1 至-100 范围的支持。当此参数为负数时，即启用仿真模式。此时驱动不再访问设备硬件，而从 DLL 所在路径文件夹下的<SimFiles>子级文件夹内寻找 BIN 文件进行仿真采集。将<SimFiles>文件夹内的 BIN 文件按文件名进行排序，序号-1 对应排序第一的 BIN 文件，-2 对应排序第二的 BIN 文件，以此类推。若驱动未找到对应的 BIN 文件，则函数返回失败值；若序号设定值 < -100，则函数直接返回失败值。

当函数成功打开 BIN 文件后，将返回仿真设备的句柄，此句柄可以直接兼容后级各项驱动函数，无需修改其他代码即可实现仿真采集。从仿真设备获取的硬件型号是生成 BIN 文件时的设备硬件型号；从仿真设备获取的硬件信息为仿真信息字符串；从仿真设备采集的数据是 BIN 文件内所存储数据的循环播放；对仿真设备的采样率、DIO、PGA 等硬件功能的设置不产生实际效果；但对仿真设备的数字滤波器设置仍然有效。

2、MPS_DataIn 函数

本函数功能定义与旧版相同，在旧版基础上增加了对千万级超大环形缓冲区的支持。本函数的其他参数定义均与旧版相同，仅对 SampleNumber 参数升级。升级包括：（1）不再要求 SampleNumber 参数必须是 128 的整倍数，只须满足是通道个数的整数倍即可，例如四通道的硬件设备，可以支持如 4 点、1000 点、4096 点等任意值；（2）在调用 Start 函数后，数据将自动存入千万点超大缓冲区，支持软件在一段时间（时间 < 一千万 / 采样率 / 通道数）后一次性读出数据，期间数据不会丢失。

3、MPS_Configure_Pro 函数

本版 DLL 在 MPS_Configure 函数之外，新增一个专业版本的 MPS_Configure_Pro 设置函数。

MPS_Configure_Pro 函数集成任意采样率、自动抗混叠滤波器与 FIR 线性相位滤波器（FIR、1024 阶、512 点瞬态）等功能。若需要使用上述功能，可升级程序代码，使用

MPS_Configure_Pro 函数代替 MPS_Configure 函数进行参数设置；若无需使用上述功能，可无需修改程序代码，仍然使用 MPS_Configure 函数做无缝升级。MPS_Configure_Pro 函数具体定义如下：

- `int MPS_Configure_Pro(int SampleRate, int DIOMode, float HighpassCutoffFrequency, float LowpassCutoffFrequency, HANDLE DeviceHandle)`

`int MPS_Configure_Pro` 函数用于专业模式下设置设备的采集参数，函数执行成功后，将对设备的采样率（支持任意采样率值）、IO 口模式以及 FIR 数字滤波器进行设置。函数执行成功返回 1，失败返回 0。

`int SampleRate`：此参数用于设置采样率，设置规则如下：

- ◆ 硬件原生采样率最高为 64K 时，支持 1kHz-128kHz 范围内任意采样率设置
- ◆ 硬件原生采样率最高为 128K 时，支持 1kHz-256kHz 范围内任意采样率设置
- ◆ 若设置值为 64K 或 128K 采样率，直接使用硬件原生采样率与抗混叠滤波器
- ◆ 若设置值为其他采样率，驱动进行采样率重设，并自动匹配抗混叠滤波器

`int DIOMode`：此参数用于设置 IO 口的模式，参数说明见 MPS_Configure 函数。

`float HighpassCutoffFrequency`：此参数用于设置高通滤波器的截止频率，取值范围为 $0 \leq \text{HighpassCutoffFrequency} < \text{采样率} / 2$ ，单位为 Hz。若 HighpassCutoffFrequency 设置值小于 0，驱动自动调整为 0；若 HighpassCutoffFrequency 设置值大于等于采样率 / 2，驱动自动调整为采样率 / 2 - 1。

`float LowpassCutoffFrequency`：此参数用于设置低通滤波器的截止频率，取值范围为 $0 \leq \text{LowpassCutoffFrequency} < \text{采样率} / 2$ ，单位为 Hz。若 LowpassCutoffFrequency 设置值小于 0，驱动自动调整为 0；若 LowpassCutoffFrequency 设置值大于等于采样率 / 2，驱动自动调整为采样率 / 2 - 1。

驱动将根据 HighpassCutoffFrequency 和 LowpassCutoffFrequency 的设置值自动配置滤波器类型：

- ◆ 若 HighpassCutoffFrequency = 0 且 LowpassCutoffFrequency = 0：不启用任何滤波器
- ◆ 若 HighpassCutoffFrequency = 0 且 LowpassCutoffFrequency > 0：启用低通滤波器
- ◆ 若 HighpassCutoffFrequency > 0 且 LowpassCutoffFrequency = 0：启用高通滤波器
- ◆ 若 HighpassCutoffFrequency > 0 且 LowpassCutoffFrequency > 0 且 HighpassCutoffFrequency < LowpassCutoffFrequency：启用带通滤波器
- ◆ 若 HighpassCutoffFrequency > 0 且 LowpassCutoffFrequency > 0 且 HighpassCutoffFrequency > LowpassCutoffFrequency：启用带阻滤波器，并自动交换两个截止频率，LowpassCutoffFrequency 将对应带阻滤波器的低截止频率，HighpassCutoffFrequency 将对应带阻滤波器的高截止频率

`HANDLE DeviceHandle`：此参数为设备句柄，参数说明见 MPS_Configure 函数。