

版本号：PSU16GN202309A

航华PSU16电子压力扫描阀使用说明



大连航华科技有限公司

2023年9月8日

目录

1 介绍	4
1.1 功能和用途	4
1.2 特点与优势	4
1.2.1 对比单管及多管压力计	4
1.2.2 对比Scanivavle、TE等进口扫描阀	5
1.2.3 对国内其他厂商的优势	5
1.3 系统组成	6
1.4 运行要求	9
2 起步	10
2.1 软件安装	10
2.2 硬件连接	12
2.3 同步测量	13
2.4 运行软件	16
2.5 软件介绍	16
2.5.1 程序控制区	16
2.5.2 采集操作区	17
2.5.3 扫描阀编号及数据文件设置	18
2.5.4 采集参数设置区	19
2.5.5 采集数据显示	20
2.5.6 统计量显示	20
2.6 软件操作流程	21
2.6.1 单个数据文件	21
2.6.2 多个数据文件	22
2.6.3 每组平均值	24
2.6.4 读取数据文件	25
3 另一条路：Matlab或Python编程	27
3.1 安装数据采集卡驱动及设置软件：NI-DAQmx	27
3.2 Matlab	27
3.2.1 安装Matlab及data acquisition toolbox工具箱	27
3.2.2 利用Matlab程序读取并存储零点电压	27
3.2.3 利用Matlab程序采集并存储实验数据	29
3.3 Python	29
附录	30

目录

A 使用NI MAX	30
B 航华AM2 NACA0018翼型实验模型	31
C 标定文件	32
D 零点温漂	33
E Matlab 读取并记录零点程序	35
F Matlab采集压强程序	37
G Python程序采集零点	39
H Python采集压强程序	40

1 介绍

航华PSU16电子压力扫描阀是一款专门为科研与教学设计的压差测量设备。具有精度高、可靠、使用方便等特点。其内置16支高精度压差传感器与一块数据采集卡，通过数据采集卡进行同步采样的方式“电子扫描”传感器的模拟电压输出获取压差信息。具体优势如下：

1.1 功能和用途

PSU16电子压力扫描阀的主要功能如下：

- a) 测量模型表面压强分布
- b) 与毕托管配合测量流动速度大小
- c) 与多孔气动探针配合测量流速及方向
- d) 用于测量风洞实验段前后压差进而计算风洞实验风速

1.2 特点与优势

1.2.1 对比单管及多管压力计

与教学实验常见的基于液柱高度的压差测量仪相对比，电子压力扫描阀如下优点：

- a) 测量分辨率高

航华PSU系统的分辨率可达0.1Pa，小于基于液柱高度测量的分辨率（约10Pa）。

- b) 电子压力扫描阀时间响应快，具有动态测量能力

液柱压差计只能测量平均压强，PSU系统具有一定的动态测量能力。PSU系统内置压差传感器的动态响应时间约 1 ms^1 。因此，传感器对快速变化的动态信号具备一定的解析能力。有研究表明，当总管长小于0.5m时（对应亥姆霍兹第一模态频率170Hz），扫描阀可有效获取频率为100Hz以下脉动压强的变化。

- c) 采集数据量大

PSU系统可通过长时间采样求取平均值的办法来降低误差。

¹动态响应时间为压强出现阶跃变化后，传感器测量值达到阶跃值60%所需要的时间。

1.2 特点与优势

d) 多通道测量

PSU系统集成了16个压差传感器，可以最多测量空间16个点的压强与参考点压强之差。

e) 结构紧凑

电子压力扫描阀体积远小于基于多管压力计，且不受安装平台限制。可以适应实验室内及车辆等移动平台上使用。

f) 使用方便

探针标定数据被集成在软件包内，用户为传感器归零后点击采集即可开始测量。测量结果直接显示，并以文本文件形式保存。使用便利度有明显优势。

1.2.2 对比Scanivavle、TE等进口扫描阀

Scanivavle与TE（原PSI）多通道压强扫描系统出现较早，在压强扫描测量领域处于领先地位。与这些进口系统相比，航华PSU系统具有如下特点：

a) 投资小，性价比高

航华PSU系统在价格上具有优势。尽管价格相对较低，但航华PSU在测量精度及便利性方面具有相似的品质，整体性价比高。

b) 使用方便

航华PSU专门针对教学与科研进行优化设计，提供适于高校客户的安装及工作方式。软件更符合科研与教育工作者的需求。

c) 更方便与全面的客户维护

我们为客户提供更为方便的产品维护。

1.2.3 对国内其他厂商的优势

电子压力扫描阀的国内厂商较多。相对于友商，航华PSU的优势如下：

- a) 为教学和科研专门设计。不仅提供应用软件，还提供符合科研和教育工作者工作习惯的二次开发包，包括Labview、Matlab、Python 工具包。使用者可方便地嵌入自己的应用程序

1.3 系统组成

表 1: 系统组成

序号	品名	描述	数量
1	PSU16主机, 图1a	内置压强传感器与数据采集卡。通过USB线缆连接电脑	1
2	电源, 图1b	12V, 3A直流电源	1
3	USB数据线, 图1c	连接信号处理器与电脑, 长0.5米	1
4	U盘	内含软件, 可采集、处理、分析并记录数据	1
5	硅胶管, 图1d	内径1mm, 外径2mm, 长10m	1
6	说明书	本文	1
7	外触发线	发送触发信号给数据采集卡, 开始采集数据	选购
8	时钟同步线	连接多个采集卡, 在软件的协助下保证采样时钟与启动时钟同步	选购
9	圆柱实验模型组CM2, 图13	3cm直径, 长度245mm, 光滑, 粗糙圆柱各1支, 内置16个测压孔, 合成树脂	选购
10	翼型实验模型AM2, 图B.1	NACA0018翼型, 弦长100mm, 长度245mm, 内置16个测压孔, 合成树脂	选购
11	5孔探针	测量时均风速矢量, 角度范围 $\pm 45^\circ$	选购
12	7孔探针	测量时均风速矢量, 角度范围 $\pm 60^\circ$	选购

- b) 探针采用一体化设计：扫描阀内集成了传感器、信号放大器、数据采集卡、电源。用户仅需通过USB线将主机与电脑相连，安装软件后就可以开始测量
- c) 同样性能、精度条件下价格更有优势
- d) 完美配合航华空气动力学实验教学方案
- e) 提供软件支持，方便地与航华热线风速仪、航华眼镜蛇探针同步进行联合测量

1.3 系统组成

航华PSU16电子压力扫描阀系统包括主机、电源、数据线、测压管、软件U盘组成，具体见表 1。

- a) 主机，照片见图1a。主机内含传感器、电源整流装置、数据采集卡。主机具体参数见表 2，数据采集卡参数见表 3。



(a) 主机



(b) 电源



(c) USB线



(d) 内径1mm硅胶管

图 1: 眼镜蛇探针系统各个部件开箱照片

表 2: 扫描阀参数

	参数
量程 (kPa)	±1.0 或±2.1
测量精度	优于0.10%F.S.
传感器响应时间(ms)	< 1
通道数	16
接口	Type-C/USB
工作温度(°C)	-20至80
外形尺寸 (长宽高, mm)	213*127*48
重量 (g)	300
直流供电电压 (V)	12
电源功率 (W)	10
测压口外径 (mm)	1.5

表 3: 数采参数 (NI USB6211 www.ni.com/en-us/shop/model/usb-6210.html)

	参数
厂商	National Instrument
型号	USB 6210
模拟量通道数	16 (单端)
采样精度	16 位
最大采样频率	250 kHz
软件功能	采集和初步分析功能,
微机 (选配)	CPU IntelI5, 内存8G, 512G硬盘

1.4 运行要求



(a) 主机正面照片，0端口接参考压，1-16接测点

(b) 主机背面照片。5个接口自左至右分别是电源插口、USB插口、触发口、同步时钟入口、同步时钟出口

图 2: PSU16电子压力扫描阀前、后面板照片

☞ 请注意每台PSU16电子压力扫描阀都有独立编号，通常在主机表面标签上显示，一般为6位数字，比如230004。该编号是软件读取标定文件的重要依据，需要在软件中输入。

- 主机正面(图2a)有17个接口。其中0接口通过软管连接参考压，1-16 接口分别连接模型测压孔。
- 主机背面(图2b)有五个接口，自左至右分别是电源插口（接电源，图1b）、USB插口（USB数据线，图1c）、触发信号接口、同步时钟入口、同步时钟出口。关于同步触发、同步时钟使用方法，详见第2.3节。

1.4 运行要求

我们推荐计算机配置如下：

- Intel I5 及以上CPU；8G及以上内存，500G以上硬盘；
 - Windows 10 或Windows 11，64位系统；
- ☞ 不小于1280*960的显示器分辨率。

2 起步

本节介绍如何使用航华PSU16应用软件来设置电子压力扫描阀。我们还提供了其他方法，比如使用Matlab及Python编程语言直接控制，具体方法见下一节（第3节）。

2.1 软件安装

在系统安装盘内找到setup.exe, 点击运行（图3a），依照提示逐步完成安装航华热线风速仪控制程序（图3b-3d）。程序将安装NI DAQmx 及NI VISA等底层驱动软件（美国National Instrument 公司免费软件），安装过程较长，请耐性等待。

- NI DAQmx驱动包是驱动NI USB6210数据采集卡的低层软件。
- NI VISA是负责串口通讯协议的软件。当软件通过串口驱动航华移测架及温度传感器等设备时需要串口通讯功能
- 最新版软件可从如下网址下载

<http://www.hanghualab.com/nd.jsp?id=120&fromMid=468>

当安装完毕后，

- Windows桌面会出现Scanner应用程序图标（如未出现，请进入安装目录，找到exe文件运行）。
- ☞ 用鼠标右键点击软件图标，选择“以管理员身份运行程序”，这样避免出现权限问题，导致程序无法存储用户设定的测量参数。
- ☞ 尽量避免把安装目录设在C盘，以减少使用中出现权限不足的问题
- 安装成功后，在安装目录下出现一个data目录。该目录下包含以文本文件形式的探针标定信息、传感器零点信息（无风时输出的电压值）。软件会调用该目录内的文件，用户不要直接修改该目录下文件，否则会出现运行错误。

2.1 软件安装

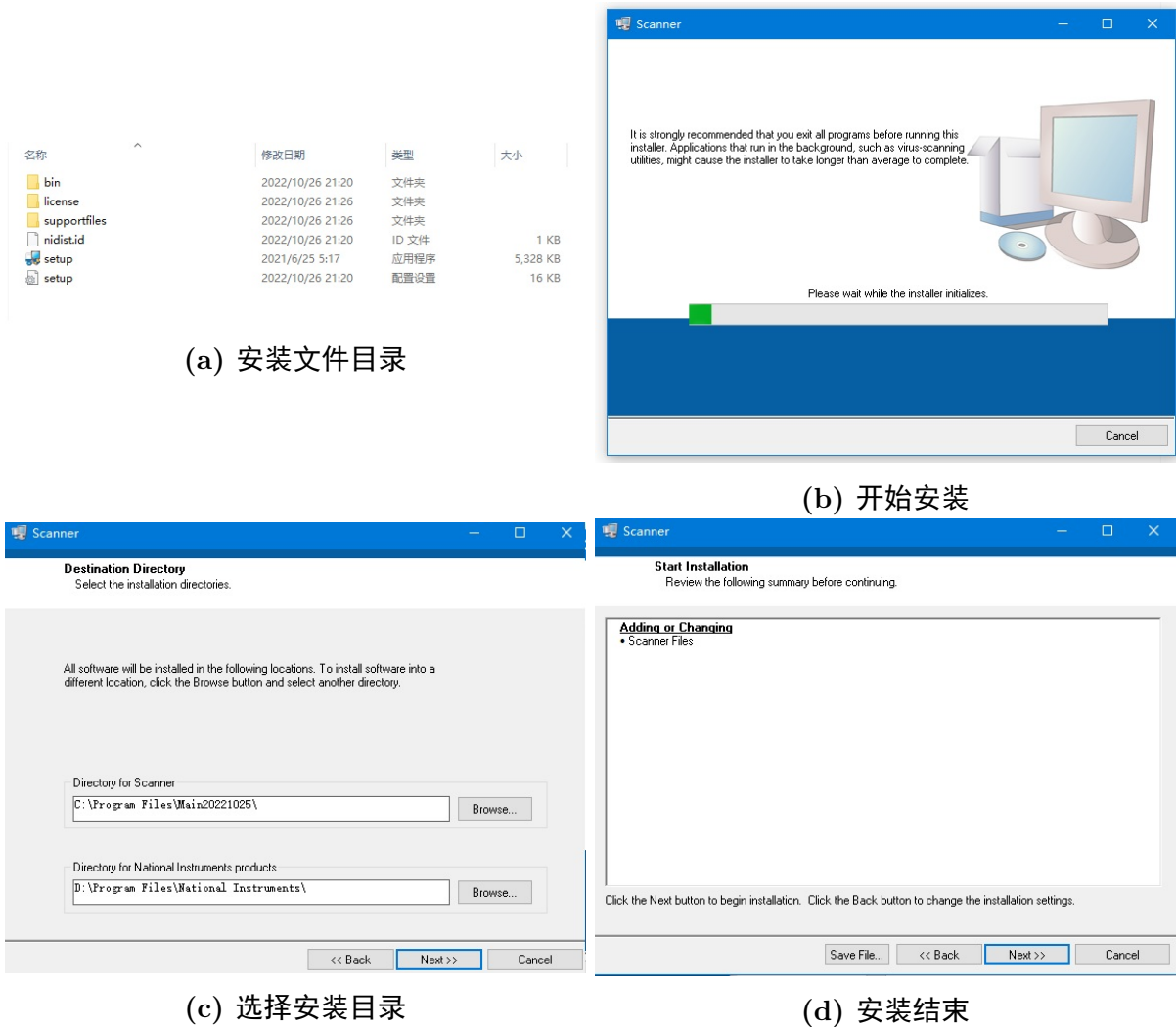


图 3: 软件安装过程示例

2.2 硬件连接



(c) 电源及USB数据线安装方法示意图

图 4: 硬件连接方法示意图

2.2 硬件连接

硬件连接方式参见图 4，其中每个部件照片见图 1。

这里需要注意的是

- 第1步，安装测压软管（图4a）。PSU16前面板1-16金属测压管需要通过软管连接至模型表面。因金属测压管外径1.5 毫米，我们建议用户使用内径为1毫米的硅胶管（淘宝搜：细硅胶管）作为测压软管；
- 第2步，安装参考压软管。PSU16测量的是各个测点与参考压之差。为了正确测量流动带来的压差，可以考虑如下方法：

2.3 同步测量

- 如果测量闭口风洞内模型的风压，可将0端口接在风洞侧壁的静压孔上，使其测量风洞实验段内壁静压，如图（见图 4b）所示。管长不限
- 如果使用开口风洞，或者射流等实验区开放的情况，参考点需要放置在没有风的地方
- 参考压软管内径同样应为1毫米。
- 第4步，使用USB线（图 1c）连接主机及电脑（图 2b），示意图见图4c。
- 第5步，连接主机电源（图 1b与图 2b）。因主机未设置开关，所以连接电源后系统自动上电。
 - ☞ 注意：每次测量以前，都要进行一个小时左右的“热机”：插入电源及探针后，让系统在无风条件下工作一小时。这样可降低传感器的零点漂移带来的影响，关于零点漂移请见附录 D节。
 - ☞ 如果经常使用，也可考虑不关闭扫描阀电源，令其连续工作。
- 第6步，检查指示灯。
 - 当连接好USB数据线后，后面板（见图 2b）绿色指示灯闪烁（位于电源插口和USB插口之间），代表采集卡正在工作。
 - ☞ 注意：绿灯闪烁，仅代表采集卡可以工作。此时如果未连接电源，传感器并未正常工作。一定要接电源上电以后，系统才能正常运行。

2.3 同步测量

在某些情况下，用户有同步测量²的需求。PSU16标配软件中不具备同步功能，但硬件上做好了准备，客户可通过二次开发实现这一功能。用户需要的具体信息包括：

- 方法1：仅保障启动时间同步，忽略各采集卡时钟差异

这种方法使用外触发信号端口命令采集卡开始采集工作。在主机内部，外触发信号端口与采集卡PFI 0 数字端口相连。外触发信号需为TTL 方波信号。当采集卡的采集模式被设为“外触发启动”后，系统会监控PFI 0端口，当该端口电压从0V 阶跃至5V 时，系统会启动采集。该方法假设各采集卡内部时钟差异可以忽略。

²同步测量是多个采集仪协调工作，在相同的时间启动，以相同的时间差去采集传感器数据。

2.3 同步测量

- 方法2：保障启动时间与采样时钟都同步

使用外部触发信号仅能保证多个采集卡同时开始采集。因各个采集卡的时钟可能存在微小差异，仅同时开始不能完全保障采集同步。如果用户对多传感器同步采集的同步性要求较高，比如需要微秒级的同步精度，用户可以考虑将多个采集卡的时钟相连。

具体做法是

- 使用时钟同步线串联多个采集卡，见图5。
 - 指定一个采集卡为主卡，在采集程序中设置其采样时钟为内部采样时钟（Sample Clock）
 - 主卡输出采样时钟（Sample Clock）至PFI 14,PFI 15,PFI 16中的任意一个，比如我们选择PFI 14
 - 主卡输出启动触发（Start Trigger）至PFI 14,PFI 15,PFI 16 中未被上一步选择的、两个中的任意一个。比如，PFI 15
 - 与主卡相连（通过同步线）的卡为第一副卡
 - 指定第一副卡的采样时钟为端口PFI 11。因为同步线将主卡的PFI 14与副卡的PFI 11相连：主卡将自身的采样时钟信号通过导线发送至副卡的PFI 11。经过这一设定以后，副卡不再适用自身的内部时钟，而使用其PFI 11端口的外部时钟信号。从而实现了两个卡的采样时钟同步
 - 指定第一副卡的启动方式为外部触发（Start Digital Edge），端口为PFI 12。并且指定第一副卡先于主卡启动采集。这样副卡启动后等待主卡发送启动触发（Start Trigger）信号给副卡的PFI 12，从而实现两个卡的启动时钟同步。
 - 如有第二副卡，处理方式与第一副卡相同。软件中，直接将主卡分享的两个时钟发送给第二、第三副卡。
- 尽管我们保留了这个功能，但不推荐用户执行此操作，尤其是方法2。扫描阀传感器的响应时间为毫秒级，管长超过1米以后响应时间降为秒级。因此，在传感器自身差异巨大的条件下，片面追求采集卡微秒级的时间同步是没有实际意义的。

2.3 同步测量



图 5: 外触发信号及同步信号内部接线示意图, 同步线将主卡的PFI 11-13 数字端口分别与副卡的PFI 14-16相连。在软件的支持下, 保障采样时钟与启动时钟同步。

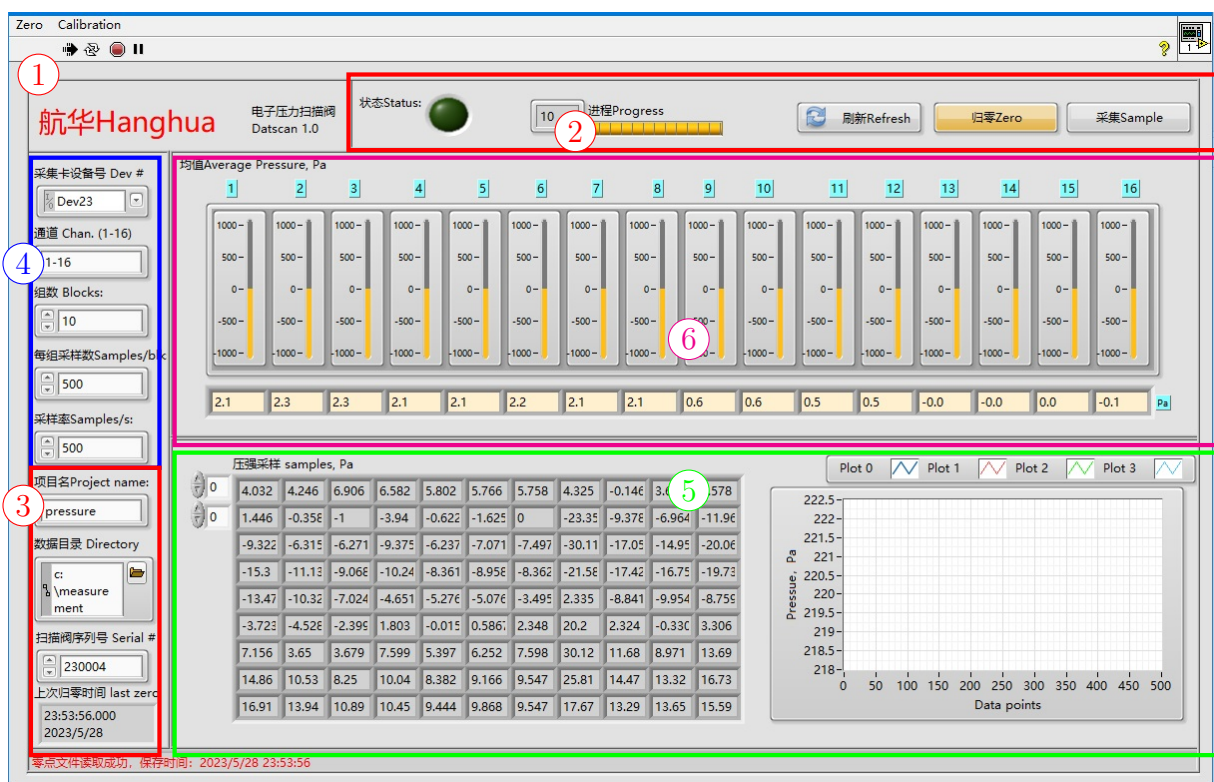


图 6: 软件主界面分为不同的功能区: ①程序控制区、②采集控制区、③文件设置区、④采集参数设置区、⑤样本显示区及⑥时均值显示区。

2.4 运行软件



图 7: 软件功能区1: 程序控制区

2.4 运行软件

在Windows快捷方式、或桌面、或者安装目录下找到并运行可执行文件。具体方法为鼠标右击图标并选择以**管理员身份运行**该程序。程序运行后须先设置参数，具体方法见第2.5节。

2.5 软件介绍

软件界面根据功能可以分为6个区域，见图6。分别是：①程序控制区、②采集控制区、③文件设置区、④采集参数设置区、⑤样本显示区及⑥时均值显示区。

2.5.1 程序控制区

软件界面左上角区域为程序控制区，这里是软件的“开关”：

- 功能区有三个按钮，自左至右分别是：运行（箭头）、连续运行、停止按钮（圆形按钮）。当程序中断后，用户可按“运行”按钮重新运行程序。运行过程中，也可以按“停止”按钮中断程序。
- 程序运行后，按钮状态见图7a；
- 用户可按红色按钮随时终止程序运行，停止后按钮状态见图7b
- 程序出错后系统也会终止运行，用户可点击“运行”重新运行程序
- 程序每次重新运行后都会从硬盘读取以前存储的参数。用户改变参数设置以后如果未按“储存参数”按钮进行保存的话，重新运行时设置结果将被刷新
- 用户完成参数设置以后（且未保存），想恢复原来设置的话，可通过停止运行、重新运行来读取硬盘存储参数

2.5 软件介绍

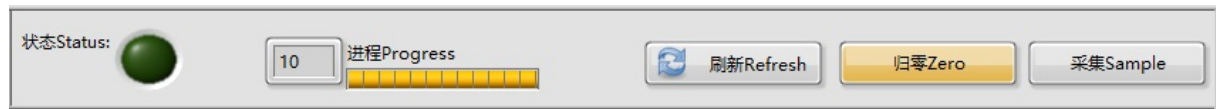
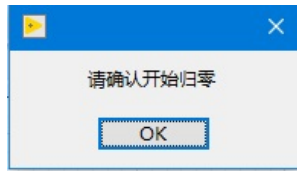
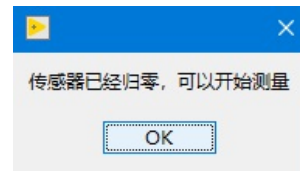


图 8: 软件功能区2: 采集操作区



(a) 提示用户将开始归零



(b) 提示用户归零结束

图 9: 传感器归零提示窗口

2.5.2 采集操作区

采集操作区内包括四个按键（图 8），分别是

- “归零zero” 按键

- 压差传感器的零点（无压差时的电压输出）会受到温度、大气压等多个因素影响而发生微小变化，即零点漂移，见附录 D
- 零点漂移对低速测量会产生一定影响。该影响可通过测量前读取零点值进行修正
- 航华眼镜蛇探针要求用户最长每三小时进行一次归零。用户在无风条件下按“传感器归零”键。系统将弹窗请用户确认，结束后也会弹窗通知用户归零结束，见图9
- 结束后，在安装目录下的data目录内，文本文件zero_cobra.txt会被更新。该文件保存各个传感器零点电压数据。

- “采集Sample” 按键。

用户安下“采集”按键后，系统开始采集。采集前，系统首先读取标定文件及零点文件。如果归零时间距离当前超过3个小时，系统还将提醒用户重新归零，以提高采样质量。

- “刷新Refresh” 按键。

用户每次更改任何设置后，可按“刷新”键将主要参数存盘。当再次运行程序时，程序会自动读取这些参数，用户不必重新设置。

2.5 软件介绍

另外，采集操作区内还包括三个显示项，分别是

- 采集工作显示灯：采集中，该灯亮；反之，则灭；
- 采集组数显示：显示现在正在采集的组数，提示用户采集进程；
- 采集进程条：提示用户采集进程。如果采集组数为100组，目前正在采集第10组，进度条显示为10%。每采集完一组后更新一次进度条。

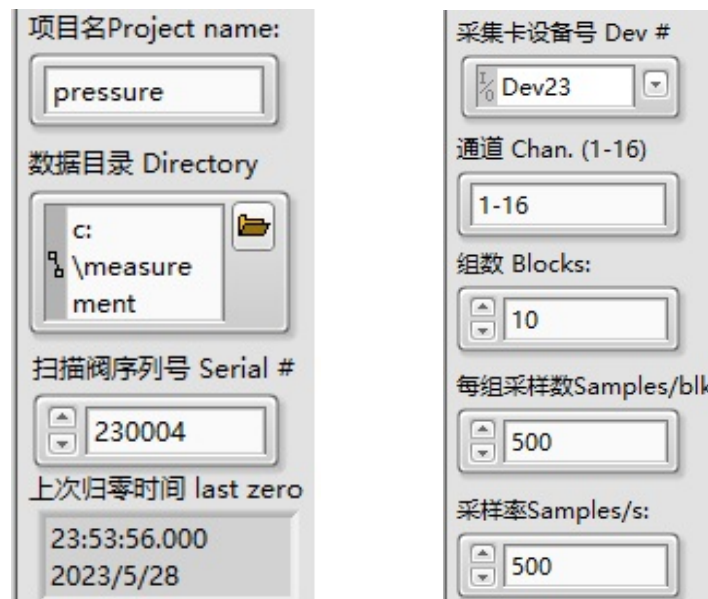
2.5.3 扫描阀编号及数据文件设置

用户在第3功能区设置扫描阀的信息及如何储存数据文件（见图 10a），包括

- 扫描阀编号
 编号通常为6位数字，比如230008。用户须从扫描阀主机标签读取编号，并在软件“扫描阀编号”框内输入该编号。系统运行后将根据探针编号提取标定文件
- 数据目录
 系统将把数据文件存储在用户指定的**数据目录**
- 项目名称
 项目名称为系统保存的数据文件名的前缀。为保证数据文件不会发生覆盖，用户需要设置该前缀，以保证每次采集的数据文件都有不同的名字。
 - 每个数据组独立保存为一个单独的文本文件。比如，如果组数为100。系统将重复100次实验，保存100个数据文件。程序将把采集的压强数据储存在以“项目名+ 采样组号.txt”为文件名的文本文件中。
 - 在每个数据文件中，系统将保存“每组采样数”指定的样本数（每个通道）。该数据文件有N*M个样本组成，其中N为“每组采样数”，M为用户指定的通道个数。
 - 除了数据文件以外，系统还为每组数据求均值，并将该组均值添加在“项目名+ avg.txt”为文件名的文本文件中。
- 填好以上参数后，点击第2功能区内“刷新Refresh”更新系统变量，同时进行保存。再次运行后，系统自动读取以往存储数据。
 - ☞ “储存参数”键将保存所有主要参数，不限于第3功能区参数

2.5 软件介绍

- 软件读取扫描阀编号对应的标定文件后，会在功能区3最下方显示该标定文件的创建时间。



(a) 功能区③：数据文件设置 (b) 区④：采样参数设置

图 10: 软件功能区③与④设置

2.5.4 采集参数设置区

用户在第4功能区设置数据采集参数（见图 10b），具体包括

- 采集卡设备号
 - 设备号为NI数据采集卡的编号。在第一次接入电脑后，电脑赋予该采集卡一个独立的编号。
 - 如果用户仅有一个采集卡，设备号通常为Dev1。
 - 用户可点击采集卡设备号选项右侧下拉符号，再点击Browse 查看系统所有采集设备，并作出选择。
 - 如果用户安装了NI公司的DAQmx软件，可以运行其附带的NI MAX，对已有的采集卡设备号进行修改
- 采集通道

2.5 软件介绍

系统缺省设置为采集并保存所有1-16通道数据。用户可以对此进行更改，具体方法为：使用英文“-”或者“,”来描述通道，语法为”1-16”，“2,3,4-14”，“1-4,7-16”。系统会采集所有通道，但仅保存用户指定通道的数据。

- 采样频率

采样频率为每秒钟采集的样本数，单位为赫兹Hz，即样本数/秒。

- 每点采样数

每点总采样数与采样时长有关：总采样数=采样时长×采样频率。如果采样数为1000个，采样频率为100Hz，采样时间则为1000/100=10 s

- 测点数量

测点数量为系统重复采样时长的次数。比如，测点数量为8，每点采样时长为10秒。系统将重复测量8次，总计采样时间为80秒

2.5.5 采集数据显示

第5功能区内显示的是系统采集的上一组数据（见图 11），每组采集完成后更新。

- 左侧：左侧为采集数据列表，单位为[Pa]。

- 列表行数为“每点采样数”（见功能区3）。
- 列数为16，分别对应16个通道

- 右侧为16通道压强测量结果图

- 横轴为数据点，无单位，最大值为“每点采样数”（见功能区3）。
- 右侧纵轴为压强，单位为[Pa]

2.5.6 统计量显示

第6功能区显示的是各通道每组数据的平均压强（见图 12），共16个数据，自左至右对应每个通道。数据在每组测量后更新，以数字和进度条两种形式显示，单位均为[Pa]。

2.6 软件操作流程

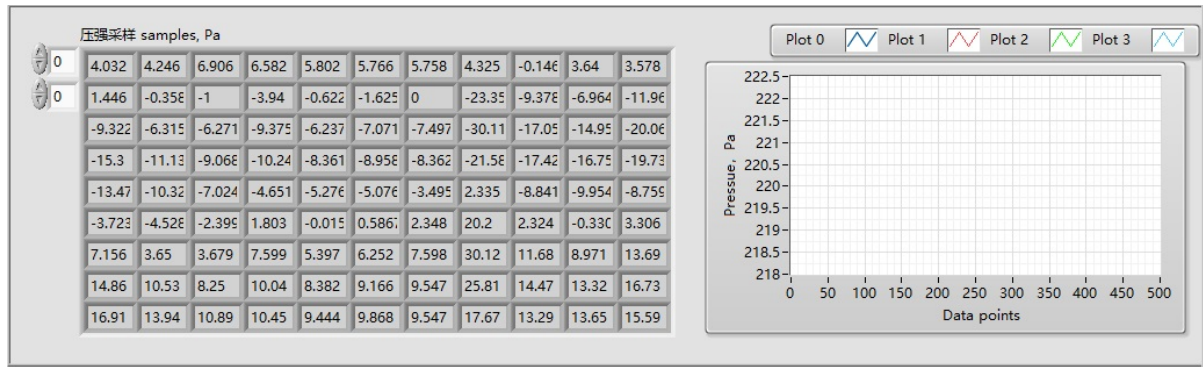


图 11: 软件功能区⑤: 采集结果显示

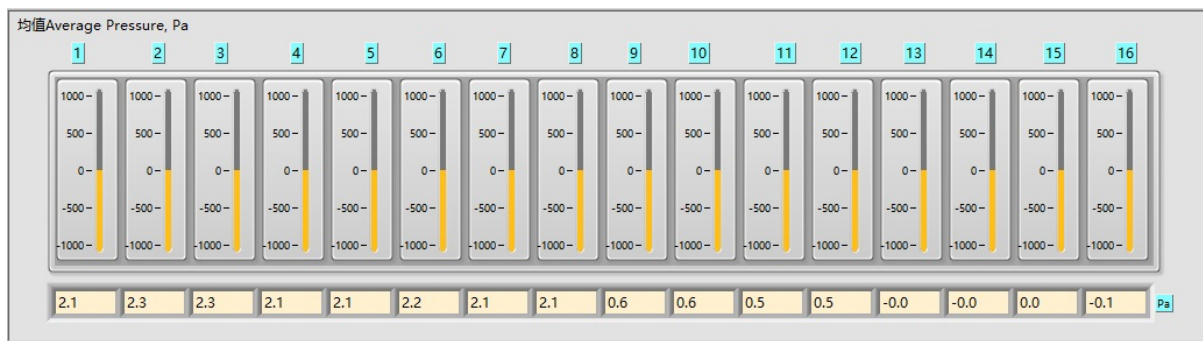


图 12: 软件功能区⑥: 统计量显示

2.6 软件操作流程

本节我们以一个实例介绍如何快速设置软件参数。假设您计划使用1台PSU16电子压力扫描阀（编号230008）采集航华AM2圆柱模型（见图13）16个测压点压强分布。采集频率100Hz，采集总时长60秒。该采集卡编号“Dev1”。您计划把数据保存在“c:\measurement”目录下，所有数据文件名都以“pressure”开头。

- 如果您希望把原始数据³保存为一个大的数据文件，操作方式见2.6.1节；
- 如希望把原始数据分段保存为若干小的数据文件，操作方式见2.6.2节。
- 除了原始数据文件以外，系统还为每组数据求均值，并将该组均值添加在以“pressure”开头、以“avg.txt”结尾的数据文件中，具体见2.6.3节。

2.6.1 单个数据文件

每步骤具体操作方法如下

- a) 连接好主机、导线，检查主机后面板灯是否亮起

³ “原始数据”指采集的所有数据

2.6 软件操作流程

- b) 安装模型，该模型共有16个测压孔。相邻测压孔间距为22.5°。
- c) 连接16个测压软管。扫描阀通道需与模型测压孔标号（见图13）保持一致。
- d) 连接参考压管：扫描阀0端口与风洞实验侧壁测压孔相连
- e) 以管理员身份运行程序
- f) （功能区③，以下省略“功能区”）填入扫描阀编号“230008”
- g) ③选择数据目录：c:\measurement, ④填入项目名称：pressure
- h) ④选择采集卡编号：Dev1
- i) ④填入通道号：1-16
- j) ④填入采样组数：1
- k) ④填入每组采样数：6000
- l) ④填入采样率：100
- m) ②点击“刷新”，保存参数
- n) ②关闭风洞，点击“归零”
- o) ②归零完成后，运行风洞，稳定后点击“采集”，直至实验结束。

上面这个实验，每个通道总采样数是6000个，数据储存在c:\measurement下的pressure1.txt文件中，见图14。该文件内含一个大小为[测点数量, 16]的二维数组。本例为[6000,16]（6000行16列）。数据与图11显示的列表相同。

2.6.2 多个数据文件

如果采集时间过长，推荐用户使用多组分段测量并保存的方式。比如，上一个例子中，我们希望每6秒保存一次数据，总计进行10次测量，总时长同样是60秒。我们进行如下设置：

- a) ④填入采样组数：10
- b) ④填入每组采样数：600

2.6 软件操作流程

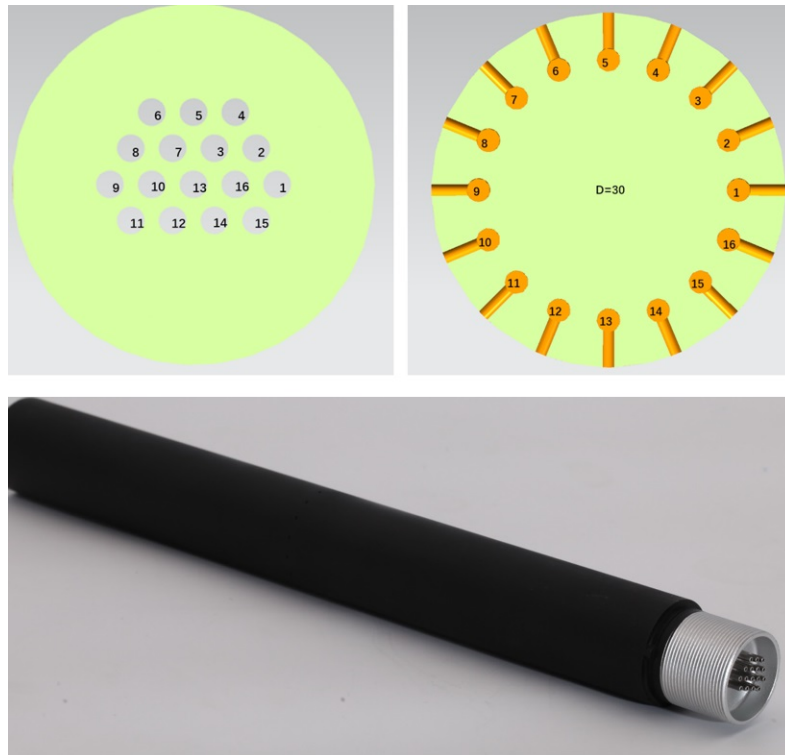


图 13: 航华CM2光滑圆柱模型照片及测压孔位图

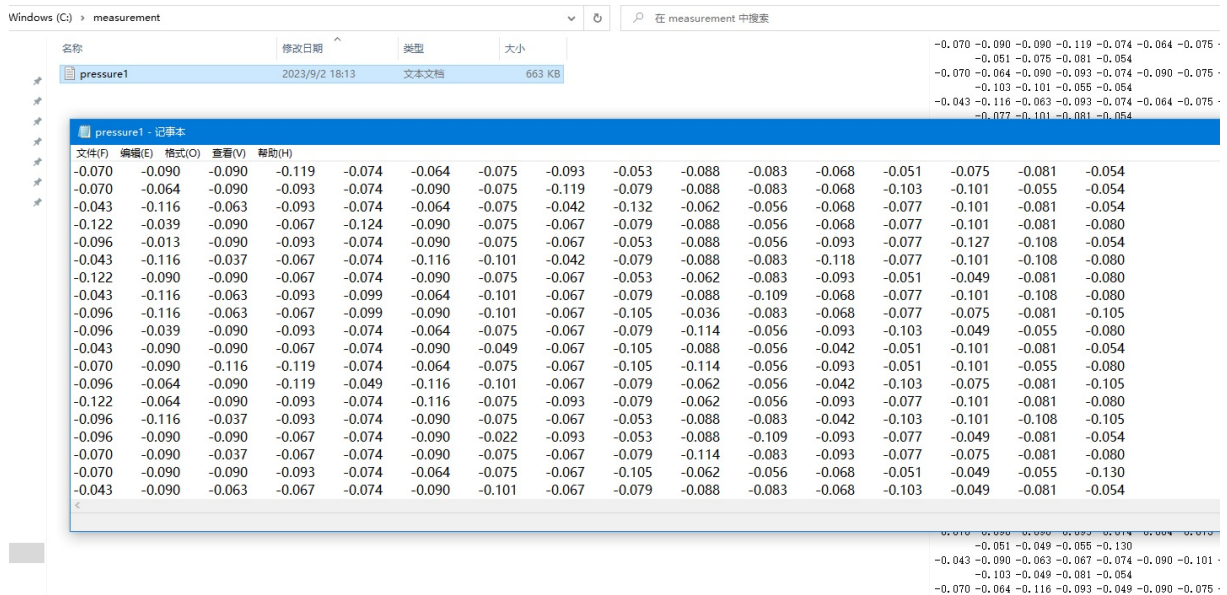


图 14: 全部6000数据点被保存在1个数据文件内。

2.6 软件操作流程

c) ④填入采样率：100

经过上面操作（单支探针在5个不同位置测量）以后，

- a) 数据存储为用户指定的目录c:\measurements下，见图 15。数据文件（文件名=项目名+测点号.txt）
- b) 每个数据内含一个大小为[600,16]的二维数组
- c) 系统没有保存时间文件，如果用户需要，可根据采样频率100Hz自行生成。比如，在matlab 中可参考

$$t=[1:600]*(1/100);$$

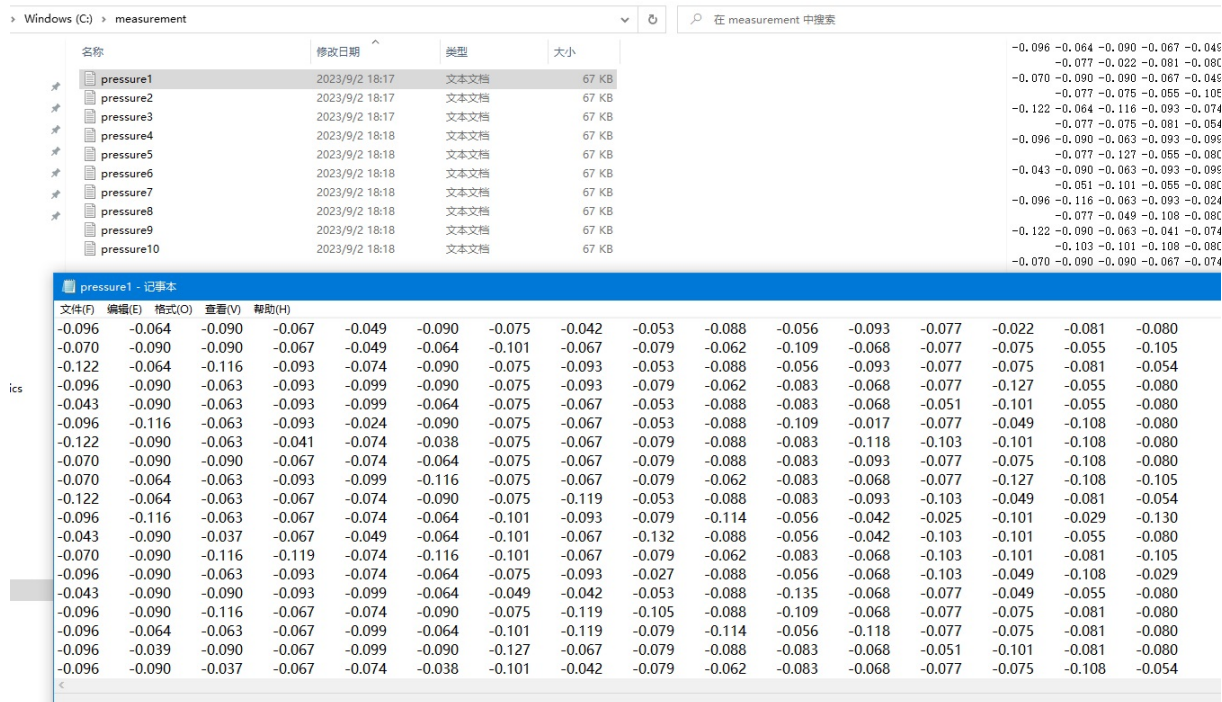
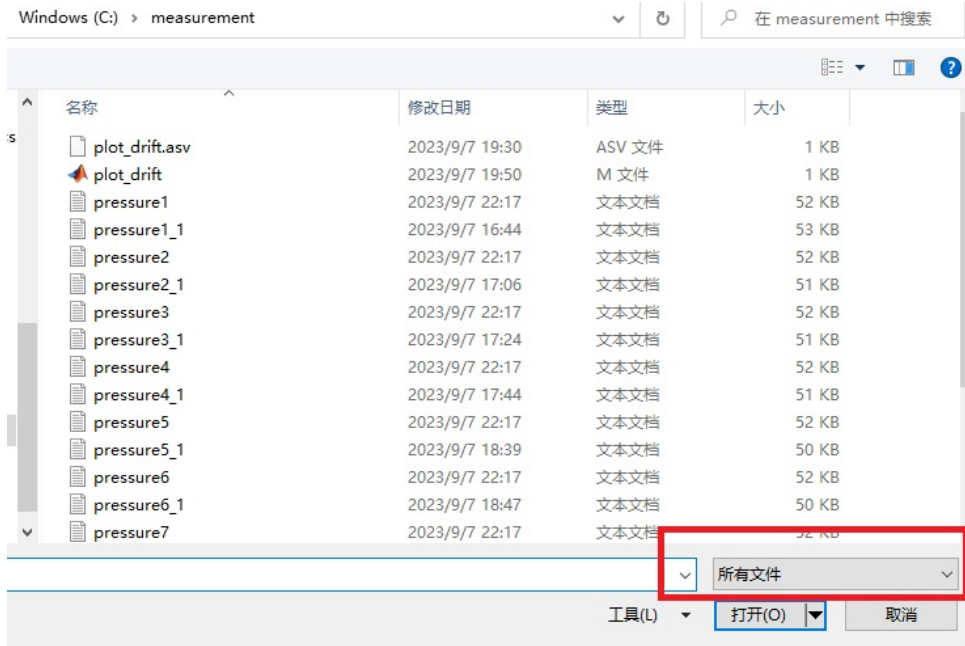


图 15: 数据被分为10段采集后分别保存。每个文件保存600数据点。

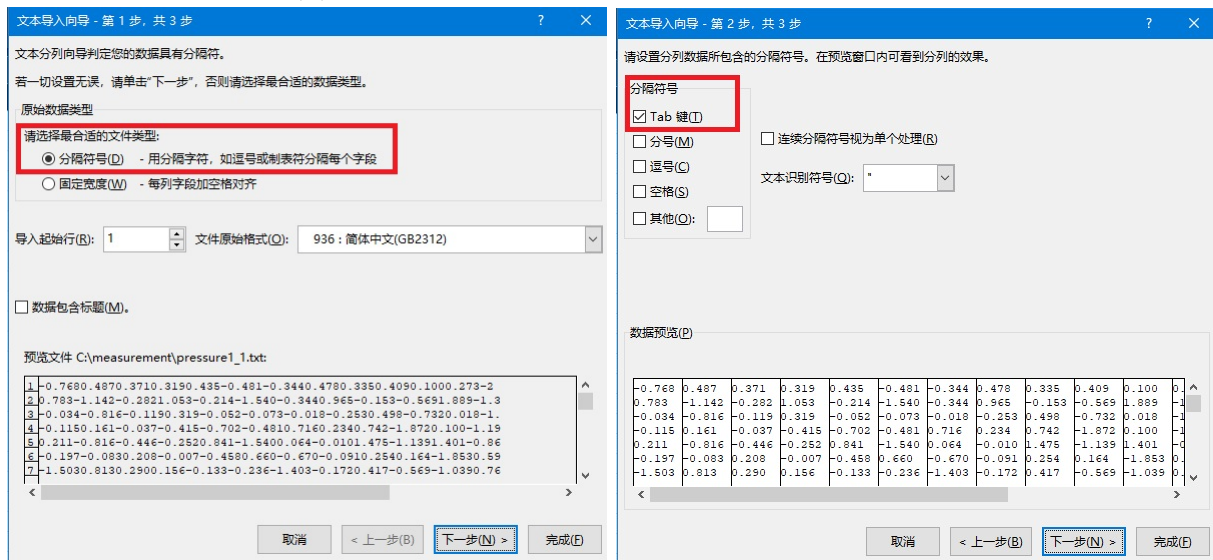
2.6.3 每组平均值

除了原始数据文件以外，程序还计算每组数据的均值，并保存在“项目名+avg.txt”文件内。每组采集以后，程序会将1x16个数据（对应每个通道）添加在该文件最后一行，不会覆盖已有数据。

2.6 软件操作流程



(a) 在Excel中选择“所有文件”以便浏览txt文件



(b) 选择“分隔符号”

(c) 分隔符号为制表符“Tab键”

图 16: 使用Excel读取数据文件过程

2.6.4 读取数据文件

用户可以使用Excel读取数据文件。打开时请依次点击“文件”、“打开”、“浏览”，并选择数据目录。请选择“所有文件”（图16a）以便浏览txt文本文件。选择要打开的数据文件后，在导入向导内选择“分隔符号”以及制表符“Tab键”，点击“完成”，即可读入数据文件。

2.6 软件操作流程

如果用户需要使用Matlab来读取并处理数据，可直接进入数据目录运行

```
cd C:\measurement  
t=[1:600]*(1/100);  
data=load('pressure1.txt');  
figure,plot(t,data(1:1024,1))
```

Python则可运行

```
import numpy as np  
import matplotlib.pyplot as plt  
data = np.loadtxt("C:\measurement\pressure1.txt", delimiter="\t")  
fig, ax1 = plt.subplots(1, 1, sharey=True, figsize=(8, 6))  
ax1.plot(data[range(1,1024),1])
```

3 另一条路：Matlab或Python编程

第2大节介绍了如何使用航华应用软件设置、采集PSU16电子压力扫描阀。用户如果有系统集成等需求，希望开发应用程序，可考虑使用Matlab或Python直接控制扫描阀（注意这里不仅仅是读取文本文件，是在matlab或python环境下通过文本命令直接控制、采集并绘制图表），具体办法如下：

3.1 安装数据采集卡驱动及设置软件：NI-DAQmx

注意，如果已经执行了第2.1节中的PSU16应用软件安装工作，系统已经安装了NI-DAQmx 中的核心部分（不包括NI MAX），用户可以不用再单独下载安装庞大的NI-DAQmx。

NI-DAQmx 是NI公司提供的、驱动NI数据采集卡的免费软件。可从如下网址下载

<https://www.ni.com/en/support/downloads/drivers/download.ni-daq-mx.html#288277>

该网址提供最新的及以往版本。我们建议用户下载18.0，或者更新的版本。建议下载时使用的参数见17。该软件较大，约1.8G，请耐心等待。下载并成功安装以后，重启电脑。

NI-DAQmx软件安装成功以后，采集卡进入工作状态。其驱动程序可被Lview、Matlab、Python、C等语言调用，实现数据采集。安装后，用户可考虑先运行NI-DAQmx附带的NI MAX 软件。该软件是一个非常有利的工具，用户可使用该软件对NI数据采集卡进行状态查询、参数设定、功能调试等。具体见附录内的图A.1 与图A.2。

3.2 Matlab

3.2.1 安装Matlab及data acquisition toolbox工具箱

在安装有Data Acquisition工具箱的Matlab之后，可在命令行输入daqlist命令。该命令会调用NI-DAQmx 提供的低层函数读取采集卡信息。如果出现图18的结果，说明已经安装成功，可以开始使用PSU16电子压力扫描阀了。

3.2.2 利用Matlab程序读取并存储零点电压

使用PSU16进行测量之前，我们需要经常性的（每2个小时）读取一下零点，即0风速下的传感器输出电压。程序见附录第E节。运行该程序前，需将标定文件放置于程序所在目录的data子目录内。关于标定文件见附录第C节。

port/downloads/drivers/download.ni-daq-mx.html#288277

Solutions ▾ Products ▾ Perspectives ▾ Support ▾ Community About Contact Us

HOME / SUPPORT / SOFTWARE AND DRIVER DOWNLOADS / NI DRIVER DOWNLOADS / DOWNLOAD DETAIL PAGE

NI-DAQ™mx

NI-DAQ™mx provides support for customers using NI data acquisition and signal conditioning devices.
[+ Read More](#)

Note: Install programming environments such as NI LabVIEW or Microsoft Visual Studio® before installing this product.

DOWNLOADS

Supported OS: Windows [View Readme](#)

Version: 18.0

Included Editions: Runtime with Configuration Support

Application Bitness: 32-bit and 64-bit

Language: English, French, German, Japanese, Korean, Simplified Chinese

NI-DAQmx 18.0 Runtime with Configuration Support

Release Date: May/17/2018

- > Supported OS
- > Language
- > Checksum

DOWNLOAD

File Size: 1.34 GB

图 17: 在NI公司网站下载NI DAQmx软件驱动NI数据采集卡

```

命令窗口
>> clear all
>> daqlist
>> daqlist

ans =

1x5 table

VendorID DeviceID Description Model DeviceInfo
-----
"ni" "Dev2" "National Instruments(TM) USB-6210" "USB-6210" [1x1 daq.ni.DeviceInfo]

fx >> |
    
```

图 18: 在配有Data Acquisition工具箱的Matlab内运行daqlist命令将显示系统安装的NI采集卡信息

3.3 Python

3.2.3 利用Matlab程序采集并存储实验数据

程序见附录第F节。该程序首先从磁盘读取零点电压文件及标定文件。在采集传感器电压后首先减去零点电压，在根据标定文件进行样条差值获得压强值并输出。

3.3 Python

如果用户使用Python来操作PSU16，我们建议使用Anaconda发行包⁴。用户下载并安装完毕以后，在Anaconda Prompt命令窗内输入如下命令来安装与DAQmx通讯的支持包

```
python -m pip install nidaqmx
```

当看到“Successfully installed nidaqmx-***”字样以后

- 可使用Spyder⁵读取附录第G 节程序并运行，采集PSU16 零点
- 运行附录第H节程序来采集并存储实验数据。该程序首先读取零点及标定文件，采集传感器输出电压并将其转换为压强，最后保存。

上面两小节提到的Matlab与Python程序请点击下面的链接，进入航华官网后下载PSU16_Matlab_Python.rar文件

<http://www.hanghualab.com/col.jsp?id=128>

⁴<https://www.anaconda.com/>

⁵Anaconda组件之一，已安装好。 <https://anaconda.org/anaconda/spyder>

附录

A 使用NI MAX

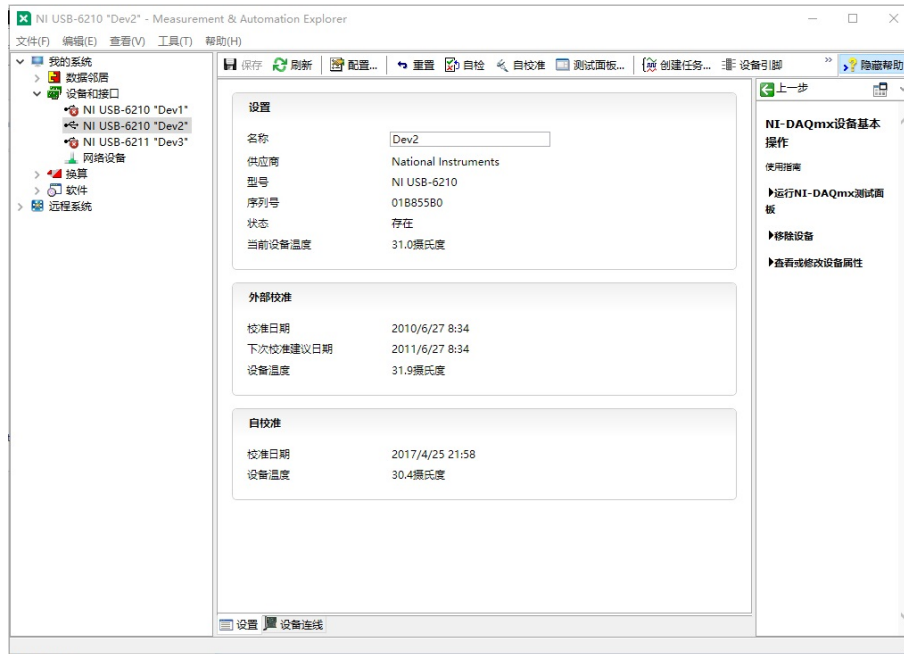


图 A.1: 随DAQmx安装的NI MAX软件，可对NI数据采集卡进行设置

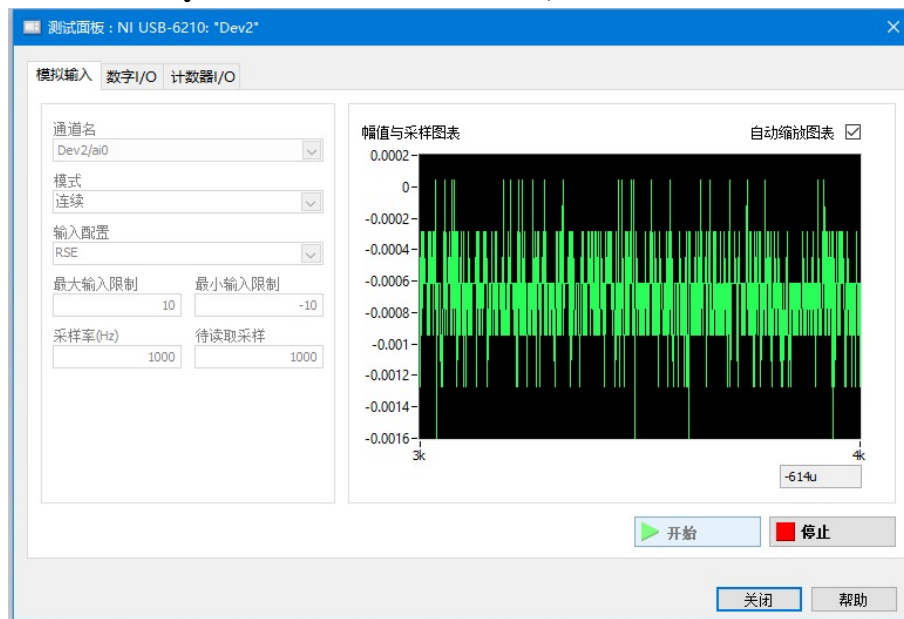


图 A.2: 随DAQmx安装的NI MAX软件，可调试NI数据采集卡

B 航华AM2 NACA0018翼型实验模型

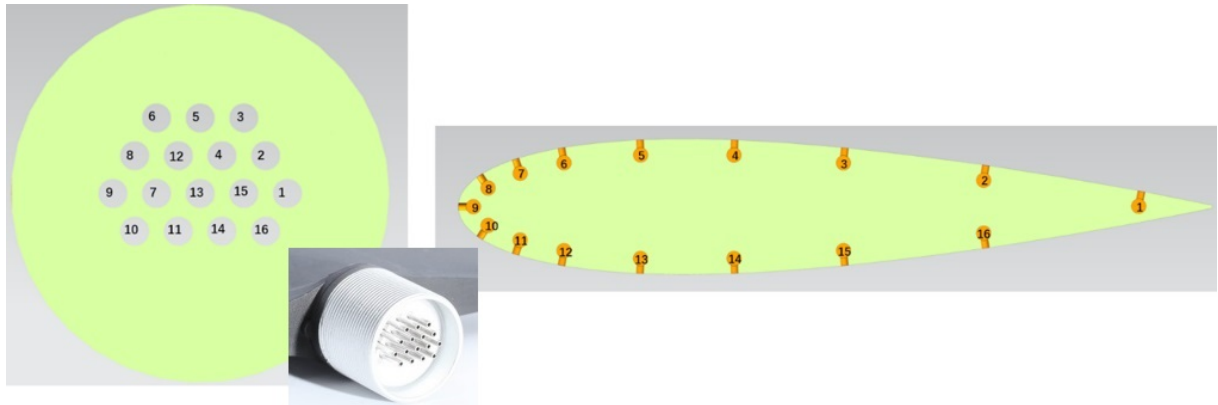


图 B.1: 航华AM2 (NACA0018) 翼型实验模型及孔位图

表 B.1: 航华AF模型孔位 (单位mm) 及方向 (单位度), NACA0018

孔编号	x	y	法向角度
1	90.72	2.04	100.67
2	70.04	5.49	98.31
3	51.32	7.81	95.65
4	36.69	8.86	92.28
5	24.11	8.87	87.39
6	13.69	7.8	80.11
7	7.58	6.33	71.85
8	2.83	4.15	56.45
9	0	0	0
10	2.83	-4.15	-56.45
11	7.58	-6.33	-71.85
12	13.69	-7.8	-80.11
13	24.11	-8.87	-87.39
14	36.69	-8.86	-92.28
15	51.32	-7.81	-95.65
16	70.04	-5.49	-98.31

C 标定文件

每个扫描阀有两个标定文件，一个例子见图C.1。两个文件包括

a) cali_input_pressure + 扫描阀编号 + .txt

该文件保存标定点压强，单位[Pa]。内存一个大小为[1,N]数组。N为标定点的数量，是一个奇数，通常是3、5、7或9等。相邻两个数字以制表符（Tab）分隔。

- 数字从小到大排列（从负到正），覆盖（或几乎覆盖）测量范围
- 中间的数字为0 [Pa]，代表无外加压差的情况

b) cali_output_volt + 扫描阀编号 + .txt

该文件保存传感器在标定点压强作用下的电压输出值，单位为伏特V。

- 该文件内含大小为[N, 16]的数组。N为标定点数量，16为测压通道数
- 第1列的5个电压值，自上至下对应从低到高不同标定压强下的电压输出

c) 软件在读取用户输入的扫描阀编号后，从data子目录下读取两个标定文件。再给根据测量的电压通过样条差值获得压强。

cali_input_pressure230006 - 记事本																
文件(F)	编辑(E)	格式(O)	查看(V)	帮助(H)												
					-2100.000	-1070.000	0.000	1240.000	2145.000							
cali_output_volt230006 - 记事本																
文件(F)	编辑(E)	格式(O)	查看(V)	帮助(H)												
-8.844	-8.809	-8.972	-8.850	-8.835	-8.837	-8.847	-8.843	-8.851	-8.852	-8.975	-8.962	-8.975	-8.962	-8.968	-8.998	
-4.548	-4.517	-4.627	-4.539	-4.530	-4.531	-4.543	-4.534	-4.538	-4.539	-4.604	-4.595	-4.602	-4.595	-4.598	-4.615	
0.000	-0.000	-0.000	-0.000	0.000	-0.000	-0.000	-0.000	-0.001	-0.000	0.000	-0.000	-0.000	-0.000	-0.000	0.000	
5.249	5.227	5.340	5.251	5.241	5.243	5.246	5.246	5.250	5.251	5.317	5.313	5.322	5.313	5.317	5.334	
9.030	9.030	9.168	9.069	9.052	9.058	9.039	9.062	9.071	9.071	9.171	9.176	9.193	9.177	9.183	9.205	

图 C.1: 编号为230006扫描阀在data目录下的标定文件

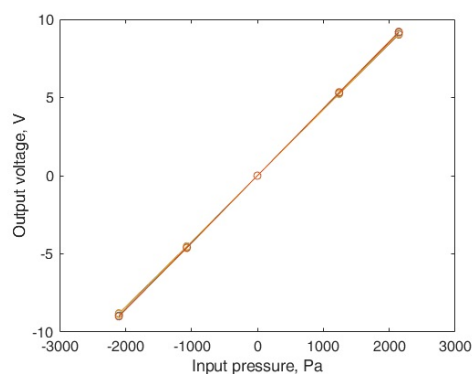


图 C.2: 上图标定文件绘制的压强输入-电压输出图

D 零点温漂

PSU16将数据采集卡与压差传感器集成在一起，这增加便利性，但也造成了热问题：数据采集卡等电子器件产生的热对传感器的零点（无压差时传感器的电压输出值）造成影响，形成零点温漂。图D.1为一个典型的PSU16温漂情况图。在系统上电一个小时以内，最高温漂达到了5Pa，但加电一个小时以后，最大温漂在 $0.25[Pa/h]$ 。具体数字见附表D.1。

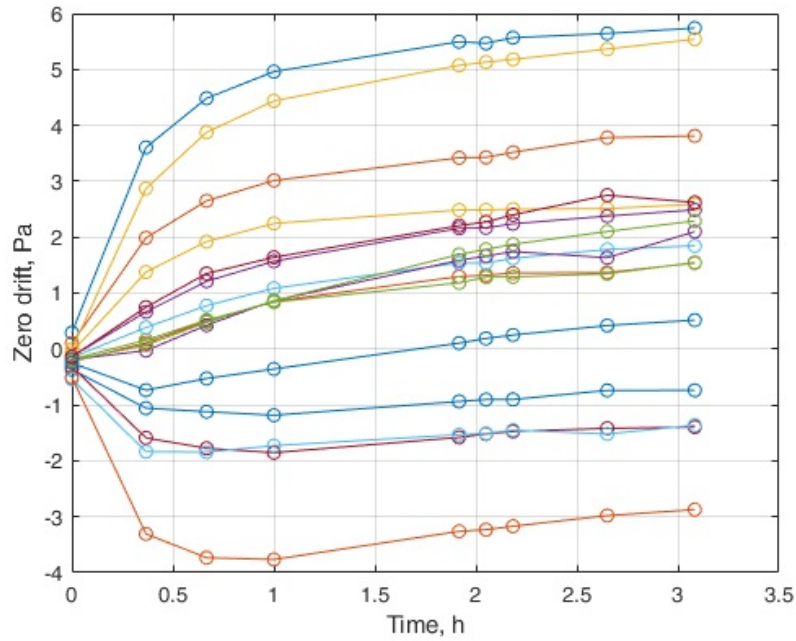


图 D.1: 编号为230006扫描阀零点漂移

表 D.1: 编号为230006扫描阀零点值随时间T变化数据, 单位[Pa]

时间,h	0.0	0.4	0.7	1.0	1.9	2.1	2.2	2.7	3.1
通道#1	-0.4	-1.1	-1.1	-1.2	-0.9	-0.9	-0.9	-0.7	-0.7
#2	-0.2	0.1	0.5	0.9	1.3	1.3	1.4	1.4	1.5
#3	-0.1	1.4	1.9	2.2	2.5	2.5	2.5	2.5	2.6
#4	-0.1	0.7	1.2	1.6	2.2	2.2	2.2	2.4	2.5
#5	-0.2	0.2	0.5	0.8	1.2	1.3	1.3	1.3	1.5
#6	-0.2	0.4	0.8	1.1	1.5	1.5	1.6	1.8	1.8
#7	-0.3	-1.6	-1.8	-1.9	-1.6	-1.5	-1.5	-1.4	-1.4
#8	0.3	3.6	4.5	5.0	5.5	5.5	5.6	5.6	5.7
#9	0.1	2.0	2.7	3.0	3.4	3.4	3.5	3.8	3.8
#10	0.0	2.9	3.9	4.4	5.1	5.1	5.2	5.4	5.5
#11	-0.2	0.0	0.4	0.9	1.6	1.7	1.7	1.6	2.1
#12	-0.2	0.1	0.5	0.9	1.7	1.8	1.9	2.1	2.3
#13	-0.5	-1.8	-1.8	-1.7	-1.5	-1.5	-1.5	-1.5	-1.4
#14	-0.1	0.7	1.3	1.6	2.2	2.3	2.4	2.8	2.6
#15	-0.3	-0.7	-0.5	-0.4	0.1	0.2	0.3	0.4	0.5
#16	-0.5	-3.3	-3.7	-3.8	-3.3	-3.2	-3.2	-3.0	-2.9

E Matlab 读取并记录零点程序

```
% Hanghua Inc. Dalian China www.hanghualab.com
% coded by N.Gao, gaonum@126.com or nangao@yahoo.com
% work with PSU16 pressure scanner, 16 pressure sensors + a NI USB 6211
% data acquisition card

% make sure you run this program while the wind tunnel is off, so that what
% saved are the zero values

clear all
close all

%%%USER INPUT%%%%%%%%%%%%%%
Channel_IDs=[0:15]; %Channels to acquire data
SamplenNumbers=1000; %Number of samples to acquire
SampleRate=1000; %Sample rate, Samples/second
DaqDeviceName="Dev1"; %Default DAQ device ID. if you dont know your device ID: make sure ther
Scanner_ID="220019"; %ID of the pressure scanner
%%%%%%%%%% USER IMPUT ENDS %%%

TableDev=daqlist("ni"); %List all the devices
temp2=size(TableDev); %read the number of devices:the row number of this array
if temp2(1)==1 % if the row number is 1: there is only 1 device connected
    DaqDeviceName=TableDev.DeviceID; % read its ID name
else
end

dq = daq("ni"); %create a data acquisition object
ch = addinput(dq, DaqDeviceName, Channel_IDs , "Voltage"); %link this object to the device, an
dq.Rate=SampleRate; %set the sample rate, it does not matters for a pressure scanner, which h
for i=1:length(ch)
    ch(i).TerminalConfig = "SingleEnded"; %set every channel
end

dq.Channels % print channel information for user to read on screen
%addinput(dq, "Dev14", "ai1", "Voltage")
%addoutput(dq, "Dev14", "ao0", "Voltage")
```

```
%%% acquire data %%%  
full_data = read(dq, SampleNumbers); % reads a certain amount of data  
t=full_data.Time; %put time information to variable t  
E=full_data.Variables; %put voltage information, an array of [SampleNumbers*  
  
%display the voltages  
figure  
plot(t,E);  
ylabel("Voltage (V)")  
xlabel("Time (s)")  
  
E_zero=mean(E); %mean value of the voltages  
%save the mean values, for ther program to load  
save(['data/press_scanner_zeros' num2str(Scanner_ID) '.txt'], '-ascii', 'E_zero');
```

F Matlab采集压强程序

```
% Hanghua Inc. Dalian China www.hanghualab.com
% coded by N.Gao, gaonum@126.com or nangao@yahoo.com
% work with PSU16 pressure scanner, 16 pressure sensors + a NI USB 6211
% data acquisition card

clear all
close all

%%%USER INPUT%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Channel_IDs=[0:15]; %Channels to acquire data
SamplenNumbers=1000; %Number of samples to acquire
SampleRate=1000; %Sample rate, Samples/second
DaqDeviceName="Dev1"; %Default device ID. if you dont know your device ID: make sure there is
Scanner_ID='220019'; %ID of the pressure scanner
ProjectName='Press_'; %data file will be saved with file names beginning with this word
DataDir=''; %directory to store data files
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% USER IMPUT ENDS %%%%

zeros=load(['data/press_scanner_zeros' num2str(Scanner_ID) '.txt']);
Press_Cali=load(['data/cali_input_pressure' num2str(Scanner_ID) '.txt']);
output_volt=load(['data/cali_output_volt' num2str(Scanner_ID) '.txt']);

TableDev=daqlist("ni"); %List all the devices
temp2=size(TableDev); %read the number of devices:the row number of this array
if temp2(1)==1 % if the row number is 1: there is only 1 device connected
    DaqDeviceName=TableDev.DeviceID; % read its ID name
else
end

dq = daq("ni"); %create a data acquisition object
ch = addinput(dq, DaqDeviceName, Channel_IDs , "Voltage"); %link this object to the device, an
dq.Rate=SampleRate; %set the sample rate, it does not matters for a pressure scanner, which h
for i=1:length(ch)
    ch(i).TerminalConfig = "SingleEnded"; %set every channel
end
```

```

for loopid=1:1 %acquire data in different blocks

    %%%% acquire data %%%%
    full_data = read(dq, SamplenNumbers); % reads a certain amount of data
    t=full_data.Time; %put time information to variable t
    E=full_data.Variables; %put voltage information, an array of [SampleNumb

    for i=1:length(Channel_IDs)
        Volt_Cali=output_volt(:,Channel_IDs(i)+1);
        P(:,i)=interp1(Volt_Cali,Press_Cali,E(:,i)-zeros(Channel_IDs(i)+1),'spline');
    end

    figure
    bar(Channel_IDs,mean(P));
    ylabel("Pressure (Pa)")
    xlabel("Channels")
    title(["Mean values of the data block " num2str(loopid)])

    %save data file
    save([DataDir ProjectName 'block' num2str(loopid) '.txt'], '-ascii', 'P');
end
    
```

G Python程序采集零点

此程序可点击下面的链接下载PSU16_Matlab_Python.rar

```
#http://www.hanghualab.com/col.jsp?id=128
@author: Nan Gao, Hanghua Inc. www.hanghualab.com
import nidaqmx
import matplotlib.pyplot as plt
import numpy as np
from nidaqmx.constants import LineGrouping
from nidaqmx.constants import Edge
from nidaqmx.constants import AcquisitionType
chans=np.linspace(0,15,16)

sampling_rate = 1000
sample_numbers= 1000
scanner_ID='220019'
Dev_ID="Dev14"

with nidaqmx.Task() as task:
    task.ai_channels.add_ai_voltage_chan(Dev_ID+"/ai0:15")
    task.timing.cfg_samp_clk_timing(sampling_rate, source="", active_edge=Edge.RISING, sample_
    a=task.read(sample_numbers)

b=np.asarray(a)
zeros=np.mean(a,axis=1)

fsize=10
fig, ax1 = plt.subplots(1, 1, sharey=True, figsize=(8, 6))
ax1.bar(chans,zeros)
ax1.set_xlabel(r'$Channels$',fontsize=fsize+2)
ax1.set_ylabel(r'$Zero voltages, V$',fontsize=fsize+2)
plt.title('ouput at 0 Pa')
plt.show()

np.savetxt('data/press_scanner_zeros'+str(scanner_ID)+'.txt', zeros, delimiter=' ') # X is a
```

H Python采集压强程序

此程序可点击下面的链接下载PSU16_Matlab_Python.rar

```
#http://www.hanghualab.com/col.jsp?id=128
@author: Nan Gao, Hanghua Inc. www.hanghualab.com
"""

import nidaqmx
import matplotlib.pyplot as plt
import numpy as np

from nidaqmx.constants import LineGrouping
from nidaqmx.constants import Edge
from nidaqmx.constants import AcquisitionType

#####User input area#####
sampling_rate = 1000
sample_numbers= 1000
scanner_ID='220019'
number_chans=16
number_blocks=2
project_name='Press_'
Dev_ID="Dev14"
#####

chans=np.linspace(0,number_chans-1,number_chans)
zeros = np.loadtxt('data/press_scanner_zeros'+str(scanner_ID)+'.txt', comments="#", delimiter=
PressCali= np.loadtxt('data/cali_input_pressure'+str(scanner_ID)+'.txt', comments="#", delimit
output_volt= np.loadtxt('data/cali_output_volt'+str(scanner_ID)+'.txt', comments="#", delimit
press_signal=np.zeros((number_chans,sample_numbers), dtype=float)

for j in range(0,number_blocks,1):

    with nidaqmx.Task() as task:
        task.ai_channels.add_ai_voltage_chan(Dev_ID+"/ai0:"+str(number_chans-1))
        task.timing.cfg_samp_clk_timing(sampling_rate, source="", active_edge=Edge.RISING, sam
        a_list=task.read(sample_numbers)
```

```
signal_in_volts=np.asarray(a_list)

for i in range(0,number_chans-1,1):
    VoltCali=output_volt[:,i]
    press_signal[i,:]=np.interp(signal_in_volts[i,:], VoltCali, PressCali)

P=np.mean(press_signal,axis=1)

fsize=10
fig, ax1 = plt.subplots(1, 1, sharey=True, figsize=(8, 6))
ax1.bar(chans,P)
ax1.set_xlabel(r'$Channels$',fontsize=fsize+2)
ax1.set_ylabel(r'$Pressure, Pa$',fontsize=fsize+2)
plt.title('pressure measurement')
plt.show()

np.savetxt(project_name+'block'+str(j)+'.txt', press_signal, delimiter=' ')
```