Tektronix®

使用 tm_devices 和 Python 简化测试自动化

操作指南

目录

许多行业的工程师都使用自动化来扩展测试仪器的功能。许多工程师选择免费编程语言 Python 来实现这一目标。Python 有许多显著优势，是自动化领域的绝佳编程语言：

- 多功能性
- 易教易学
- 代码可读性
- 广泛可用的知识库和模块

自动化主要有两种用途：

- 模仿人类行为的例程，以实现前面板自动化并节省时间，例如自动化符合性测试。每次需要测试一个新部件时，工程师都要开发一个脚本来完成所有这些工作并显示结果，而不是坐在示波器前，添加适当的测量值并写下结果。
- 扩展仪器功能的用途，例如：测量记录、验证或质量保证。自动化使工程师能够执行复杂的测试，而不会出现这些测试固有的许多弊端。操作员无需设置范围和手动记录结果，而且每次都能以相同的方式进行测试。

本操作指南将介绍在 Python 中开始编程控制示波器所需的内容，包括编程接口的基础知识以及如何下载和运行示例。

什么是编程接口?

编程接口（PI）是两个计算系统之间的一个或一组边界，可通过编程执行特定行为。对于我们而言，它是运行泰克每台测试设备的计算机与最终用户编写的应用程序之间的桥梁。更进一步说，它是一组可远程发送到仪器的命令，然后仪器处理这些命令并执行相应的任务。PI 堆栈（**图 1**）显示了从主机控制器到仪器的信息流。最终用户编写的应用代码定义了目标仪器的行为。这通常使用业界流行的开发平台之一来编写，如 Python、MATLAB、LabVIEW 和 C++、或 C#。该应用程序将使用可编程仪器标准命令（SCPI）格式发送数据，这是大多数测试和测量设备都支持的标准。SCPI 命令通常通过虚拟仪器软件架构 (VISA) 层发送，VISA 层通过在通信协议中加入额外的鲁棒性（如错误检查）来促进数据传输。在某些情况下，应用程序可能会调用驱动程序，然后向 VISA 层发送一条或多条 SCPI 命令。

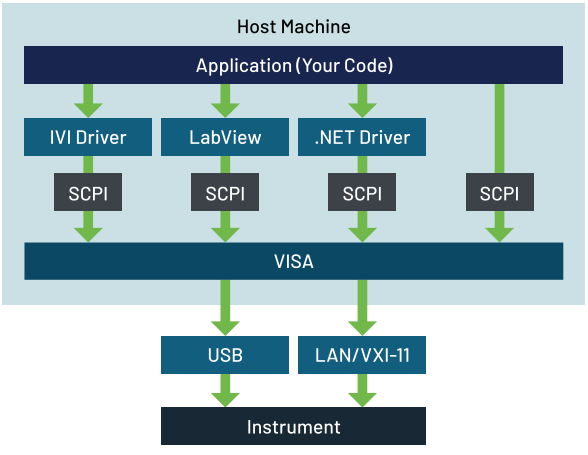


图 1. 编程接口 (PI) 堆栈显示了主机控制器和仪器之间的信息流。

什么是 tm_devices 软件包?

tm_devices 是泰克公司开发的一个设备管理软件包，其中包含大量命令和功能，可帮助用户使用 Python 编程语言轻松实现泰克和吉时利产品的自动化测试。它可以在最流行的 Python 集成开发环境中使用，并支持代码完成辅助工具。无论是否具有编程经验的工程师，都能使用该软件包简单轻松地进行编码和测试自动化。安装也很简单，使用 Python 的软件包管理工具 pip 即可。

设置环境

本节将指导您完成预先设置和安装，为使用 tm_devices 进行开发工作做好准备。本节还包括支持 Python 虚拟环境 (venvs) 的说明，以便使您的这些项目更易于管理和维护，尤其是如果你只是在使用这个软件包之前进行试用。

注：如果您的环境无法直接访问互联网，则必须使用附录中的命令修改您的步骤。如果遇到问题，请随时在**github 讨论区**发帖寻求帮助。

安装和先决条件概述

1. 安装 Python
   a. Python ⩾ 3.8
2. PyCharm - PyCharm 安装、启动项目和安装 tm_devices
3. VSCode - 安装 VSCode、启动项目和安装 tm_devices

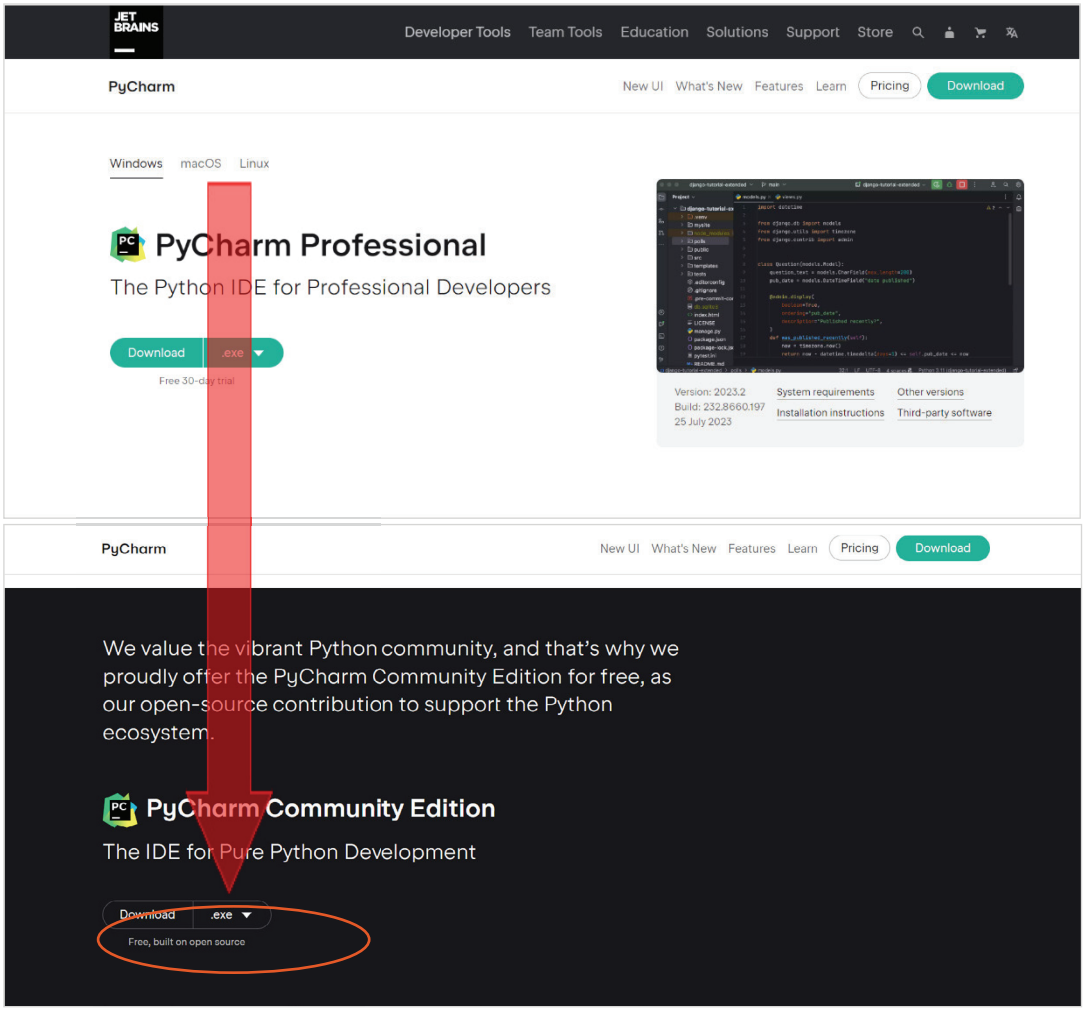
PyCharm 社区（免费）版

PyCharm 是一款流行的 Python 集成开发环境，被各行各业的软件开发人员广泛使用。PyCharm 拥有集成单元测试，允许用户按文件、类、方法或文件夹中的所有测试运行测试。与大多数现代集成开发环境一样，它也有代码自动补全功能，与基本的文本编辑器相比，可大大加快开发速度。

我们将介绍如何安装 PyCharm 社区版（免费），后在集成开发环境中安装 tm_devices，并设置虚拟环境进行开发。

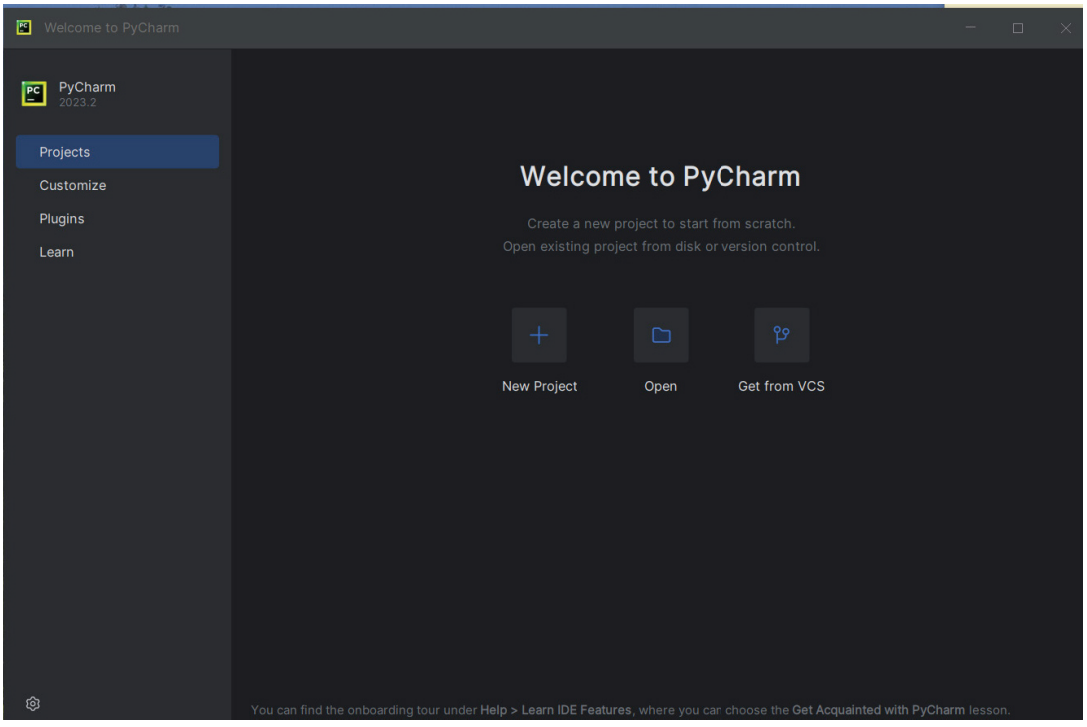
1. 转至 https://www.jetbrains.com/pycharm/

2. 滚动 PyCharm Professional 到 PyCharm Community Edition，点击下载。



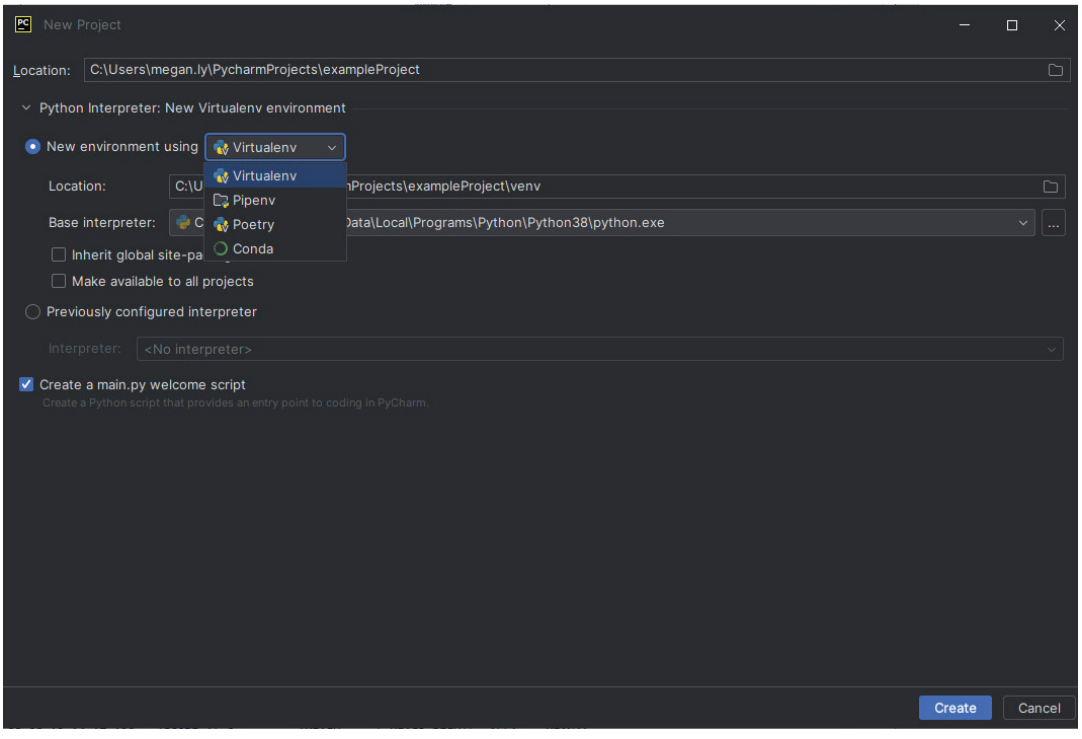
3. 您只需执行默认的安装步骤即可。我们不要求任何特殊的步骤。

4. 欢迎来到 PyCharm！

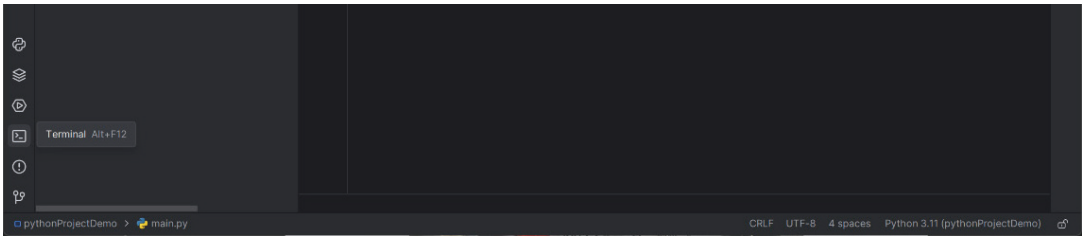


5. 现在，您需要创建一个新项目，并确保设置了虚拟环境。点击 “新建项目”。

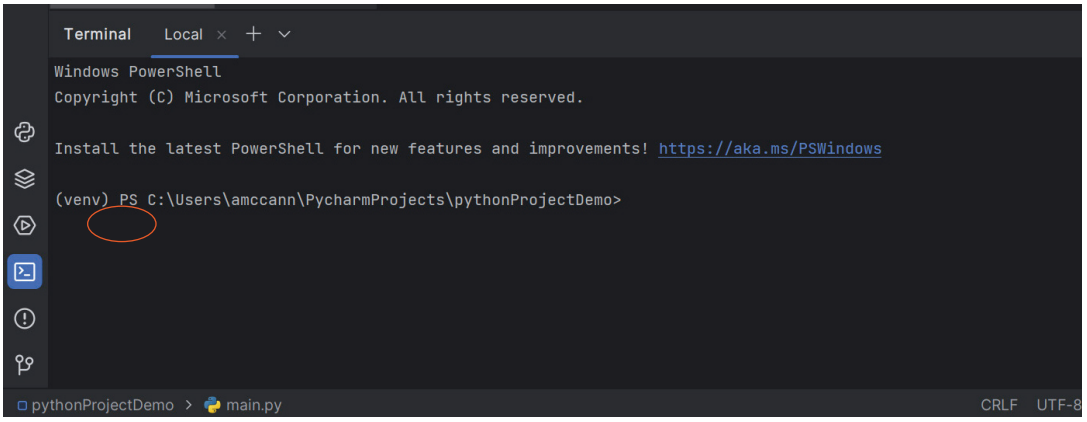
6. 确认项目路径，确保选择了 “Virtualenv”。



7. 打开终端。如果您的视图底部没有标注按钮，请查找该按钮：

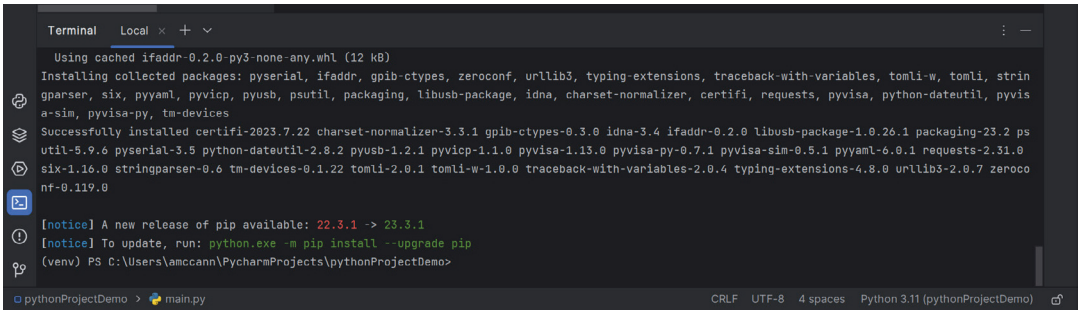


8. 通过检查终端提示符前是否有 (venv) 来确认虚拟环境是否已建立。



9. 从终端安装驱动程序。

键入：`pip install tm_devices`



10. 你的终端应该不会出错了！编程愉快！

Visual Studio 代码

Visual Studio Code 是另一款广受欢迎的免费集成开发环境，各行各业的软件开发人员都在使用它。它适用于大多数语言，并为大多数语言提供了扩展功能，使在该集成开发环境中编码非常方便高效。Visual Studio Code 提供的智能提示（IntelliSense）是开发过程中非常有用的工具，因为它可以帮助完成代码、参数信息以及有关对象和类的其他信息。非常方便的是，tm_devices 支持描述对象和类的命令树的代码自动完成功能。

我们有一份关于安装 Python 和 Visual Studio Code 的操作指南，其中包括虚拟环境设置的信息，请点击**此处**

代码示例

在本节中，我们将逐步介绍一个简单的代码示例，并强调有效使用 tm_ devices 的一些必要组件。

导入模块或库

```
from tm_devices import DeviceManager
from tm_devices.drivers import MSO5B
```

这两行对于有效使用 tm_devices 至关重要。在第一行中，我们导入了 DeviceManager。它将处理多个设备类的模块连接和断开。

在第二行中，我们导入一个特定的驱动程序，在本例中是 MSO5B。

我们使用设备管理器（Device Manager）设置文本管理器：

```
with DeviceManager(verbose=True) as device_manager:
```

然后，当我们同时使用设备管理器和驱动程序时：

```
    scope:MSO5B = device_manager.add_scope("192.168.1.1")
```

我们可以将仪器实例化，并根据其型号设置特定的命令集。只需输入仪器的 IP 地址（其它 VISA 地址也可以）。完成这四行后，我们就可以开始为 MSO5B 编写有意义的特定自动化程序了！

代码片段

让我们来看看几个简单的操作 -

将触发器类型设置为 Edge

```
    # Setting Trigger A to Edge
    scope.commands.trigger.a.type.write("EDGE")
```

以下是在 CH1 上添加和查询峰峰值测量的方法：

```
    # Specifying source as Channel 1
    scope.commands.display.select.source.write("CH1")
    # Identifying pk2pk as the measurement we would like to make
    scope.commands.measurement.addmeas.write('PK2Pk')
    # Make sure the operation is complete using the opc command
    scope.commands.opc.query()
    # Store the value locally before we print
    ch1pk2pk = float(scope.commands.measurement.meas[1].results.allacqs.mean.query())
    # Printing the value onto the console
    print(f'Channel 1 pk2pk: {ch1pk2pk}')
```

如果要测量 CH2 的幅度：

```
# Specifying source as channel 2
scope.commands.display.select.source .write("CH2")
# Identifying amplitude as the measurement we would like to make
scope.commands.measurement.addmeas .write('AMPLITUDE')
# Make sure the operation is complete using the opc command
scope.commands.opc.query()
# Store the value locally before we print
ch2amp = float(scope.commands.measurement.meas[2].results.allacqs.mean.query())
# Print the value onto the console
print(f'amplitude: {ch2amp}')
```

使用智能提示 / 代码补齐

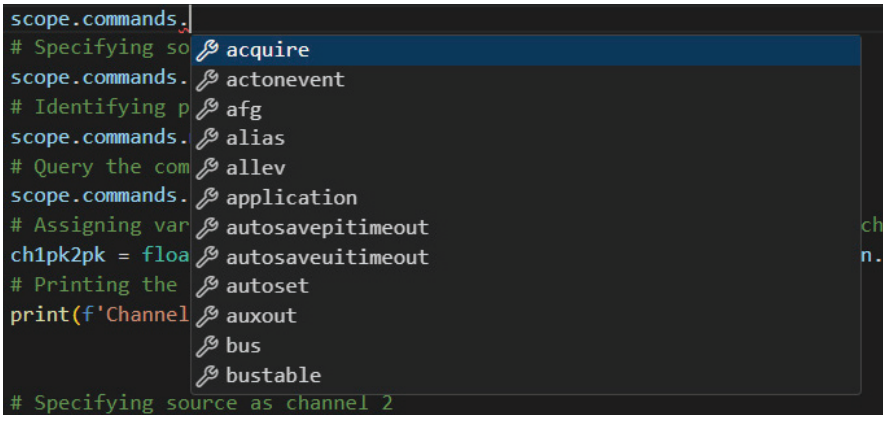
IntelliSense - 微软对代码自动完成的称呼，是集成开发环境中一项非常强大的功能，我们一直在尽可能地加以利用。测试和测量设备自动化的核心障碍之一是 SCPI 命令集。

它是一种老套的结构，其语法在开发界未得到广泛支持。

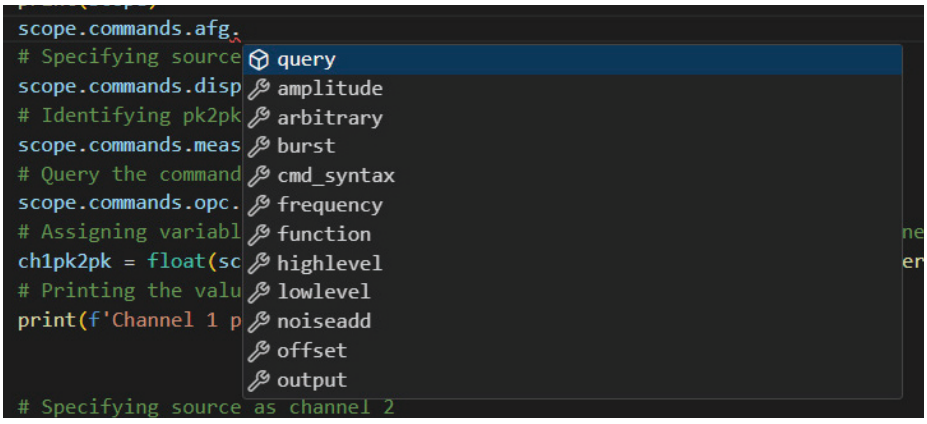
在 tm_devices 中，我们为每条 SCPI 命令创建了一组 Python 命令。这样，我们就可以根据现有的命令语法生成 Python 代码，从而避免手动开发驱动程序，并创建现有 SCPI 用户熟悉的结构。它还可以映射到在创建程序时可能需要有意调试的低级代码。Python 命令的结构模仿 SCPI（或在某些情况下，使用吉时利的 TSP）命令结构，因此如果您熟悉 SCPI，就会熟悉这些命令。

这是 IntelliSense 显示先前键入的命令，所有可用命令的示例如下：

在 scope. 后出现的可滚动列表中，我们可以看到按字母顺序排列的命令类别列表：



选择 afg，我们就能看到 AFG 类别目录：

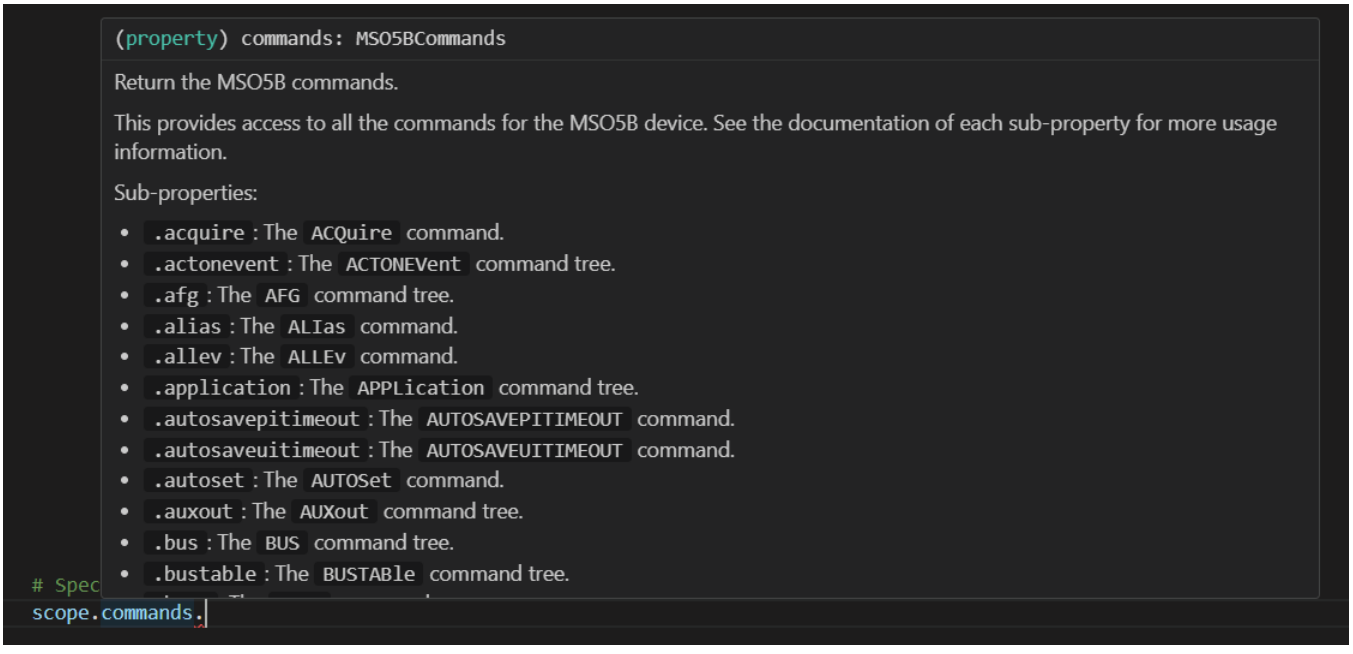


借助 IntelliSense 编写的最终命令：

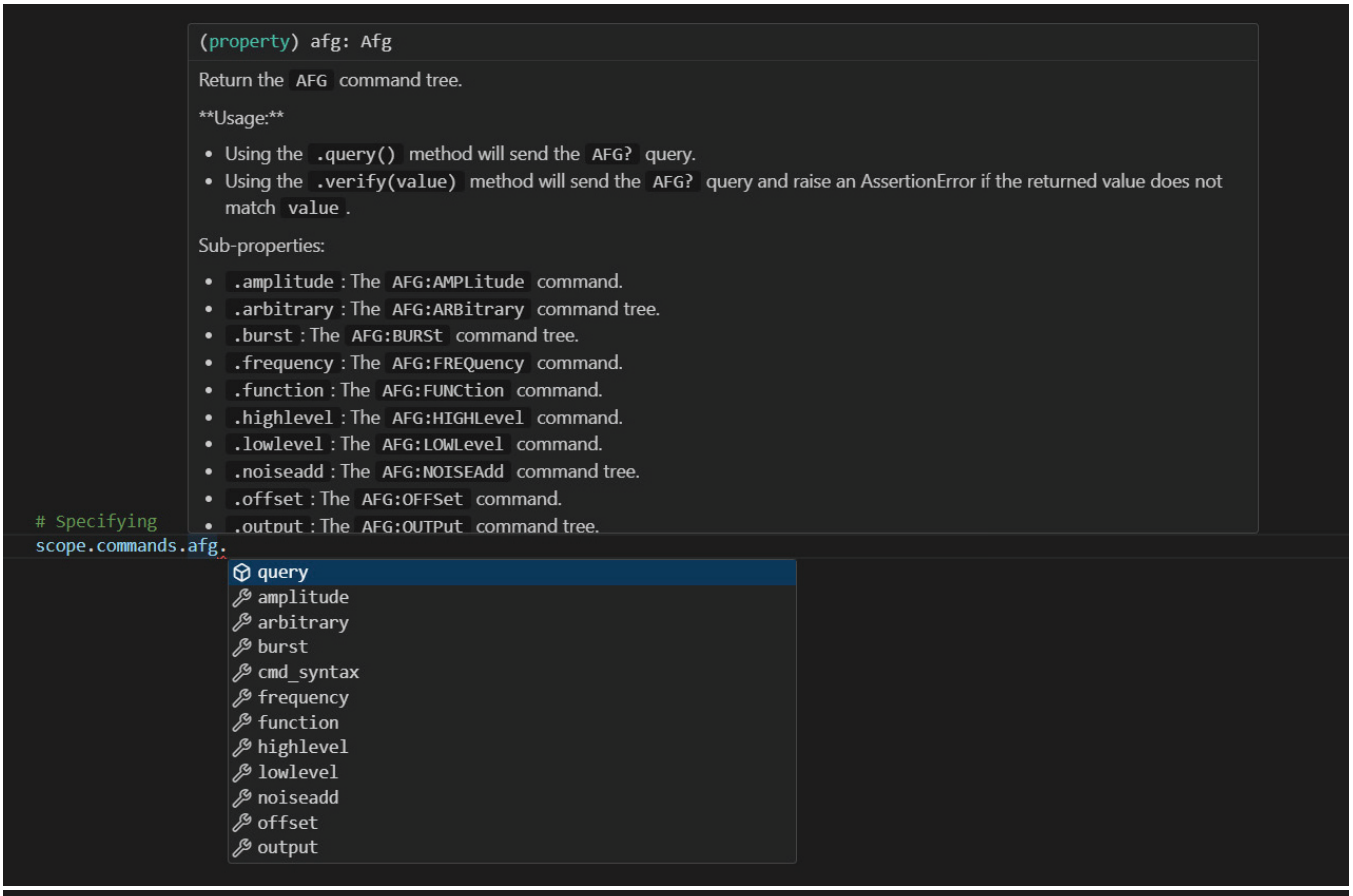
```
scope.commands.afg.amplitude.write(10e6)
```

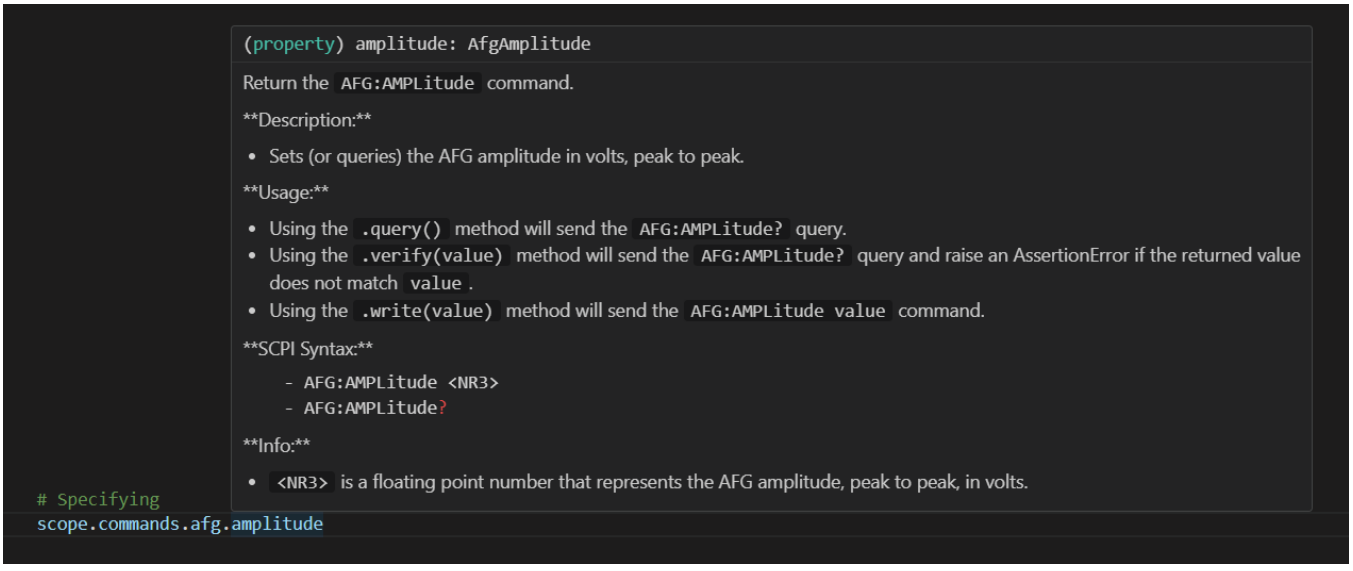
文档字符串帮助

在你编写代码或阅读别人的代码时，你可以将鼠标悬停在语法的不同地方，以获取该级别的具体帮助文档。越接近完整的命令语法，帮助文档就越具体。



根据集成开发环境的条件，您可以同时显示 IntelliSense 和 docstring 帮助。





通过本指南，您已经了解了 Tek 的 python 驱动程序软件包 tm_devices 的一些优势，可以开始您的自动化之旅了。通过简易设置、代码自动补全和内置帮助，您无需离开集成开发环境就能学习，加快开发时间，并更有信心地编写代码。

如果你想改进软件包，Github 软件仓库中有贡献指南。文档和示例文件夹中的软件包内容中都有大量更高级的示例。

其它资源

tm_devices - PyPI - 软件包驱动程序下载和信息

tm_devices Github - 源代码、问题跟踪、贡献

tm_devices Github - 在线文档

故障排除

升级 pip 通常是排除故障的第一步：

在终端键入：`Python.exe -m pip install -upgrade pip`

错误：whl 看起来像文件名，但文件不存在，或者 .whl 不是此平台支持的

```
(.venv) PS C:\pythonProject> python -m pip install pythonProject/tm_devices/tm_devices-0.1.0-py3-non-any.whl
WARNING: Requirement 'pythonProject/tm_devices/tm_devices-0.1.0-py3-non-any.whl' looks like a filename, but the file does not exist
ERROR: tm_devices-0.1.0-py3-non-any.whl is not a supported wheel on this platform.
(.venv) PS C:\pythonProject>
```

解决方案：用 Pip 安装 wheel，使其能识别文件格式。

在终端键入： `pip install wheel`

如果需要离线安装 wheel，可以按照附录 A 的类似说明进行操作，但需要下载 tar.gz 文件，而不是 .whl 文件。

附录 A - tm_devices 的离线安装

1. 在有互联网的计算机上，使用 " 下载 " 功能将软件包和所有依赖项下载到指定路径位置：

   `pip download --dest <目的地路径>wheel setuptools tm_devices`

2. 如果该计算机不支持访问 Internet，将文件复制到计算机上，

3. 接下来，按照您使用的集成开发环境的主要指南中的说明进行操作，但将安装命令换成以下命令：

   `pip install --no-index --find-links <新文件路径> tm_devices`

泰克官方微信

如需所有最新配套资料，请立即与泰克本地代表联系！

或登录泰克公司中文网站：www.tek.com.cn

泰克中国客户服务中心全国热线：400–820–5835

泰克科技(中国)有限公司
上海市浦东新区川桥路1227号
邮编：201206
电话：(86 21) 5031 2000
传真：(86 21) 5899 3156

泰克北京办事处
北京市朝阳区酒仙桥路6号院
电子城 • 国际电子总部二期
七号楼2层203单元
邮编：100015
电话：(86 10) 5795 0700
传真：(86 10) 6235 1236

泰克上海办事处
上海市长宁区福泉北路518号
9座5楼
邮编：200335
电话：(86 21) 3397 0800
传真：(86 21) 6289 7267

泰克深圳办事处
深圳市深南东路5002号
信兴广场地王商业大厦3001-3002室
邮编：518008
电话：(86 755) 8246 0909
传真：(86 755) 8246 1539

泰克成都办事处
成都市锦江区三色路38号
博瑞创意成都B座1604
邮编：610063
电话：(86 28) 8620 3028
传真：(86 28) 8527 0053

泰克武汉办事处
武汉市洪山区文化大道555号
融创智谷二期B1栋7层05室
邮编：430072
电话：(86 27) 8781 2760

泰克香港办事处
香港九龙尖沙咀弥敦道132号
美丽华大厦808-809室
电话：(852) 3168 6695
传真：(852) 2598 6260

在 TEK.COM.CN 上查找更多有价值的资源

052124 SBG 46C-74037-1