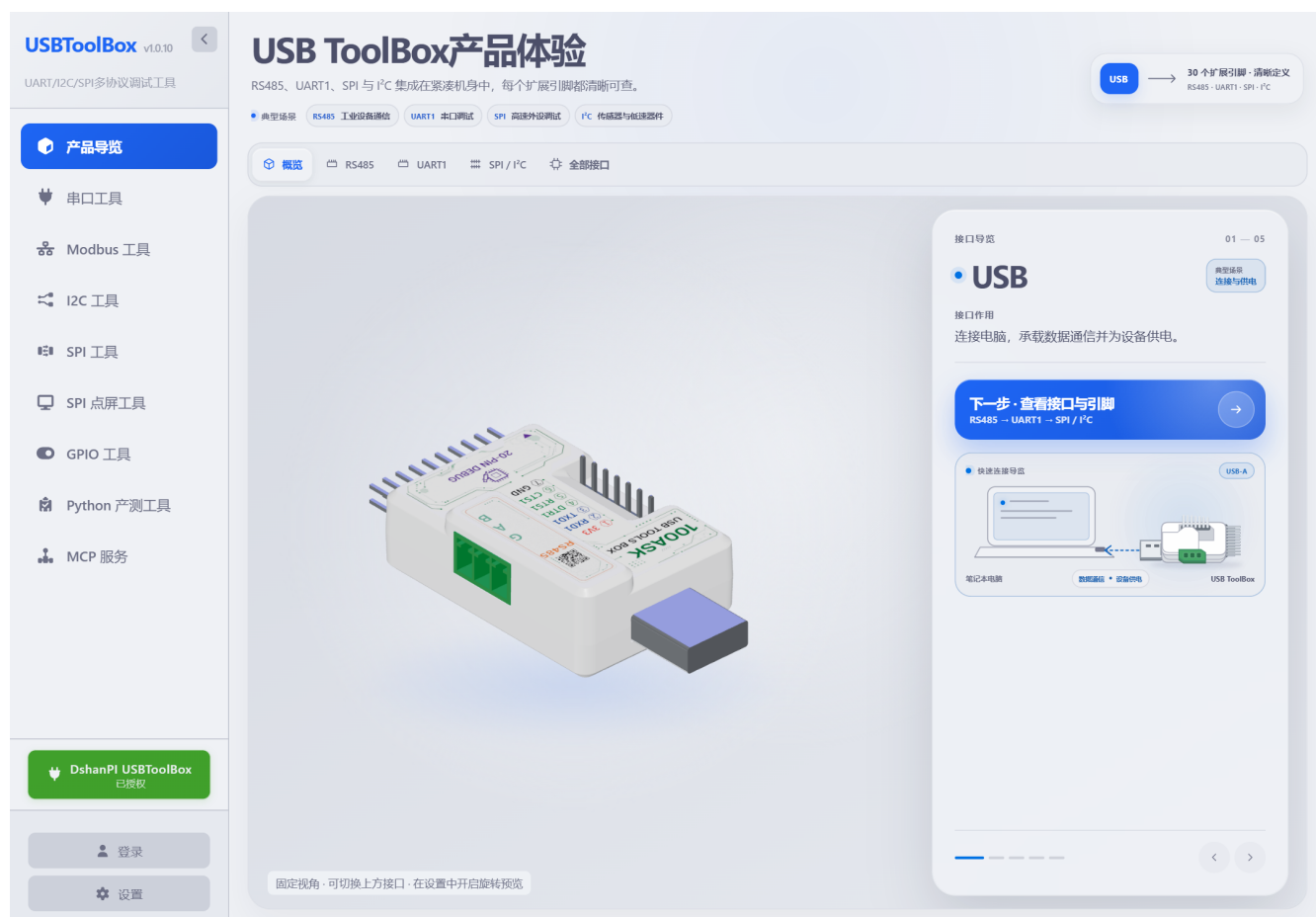


DshanPI USBToolBox

软件与硬件使用指南

从首次接入电脑，到完成通信与产测

v1.0.10 · 中文版 · 2026-10-08



Windows 入门流程 / macOS 与 Linux 安装说明
UART · Modbus · I²C · SPI · GPIO · Python · MCP

目录

1. 准备电脑、设备和连接线	4
2. 安装软件和设备驱动	6
3. 连接设备并区分授权与登录	9
4. 查看引脚示意图并完成接线	10
5. 使用串口工具读取日志和发送数据	14
6. 使用 Modbus 工具读写寄存器	16
7. 使用 I ² C 工具扫描和读写外设	18
8. 使用 SPI 工具验证传输	20
9. 使用 SPI 点屏工具显示文字和图像	22
10. 使用 GPIO 工具读取和输出电平	24
11. 使用 Python 产测工具组织测试	25
12. 登录并使用 USBToolBox 助手	27
13. 通过 MCP 服务接入外部 Agent	29
14. 调整界面、保存设置和查阅帮助	31
15. 常见问题 (FAQ) 与故障排查	32
16. 保存工位验收记录	34

使用本指南

版本: v1.0.10 · 更新日期: 2026-10-08

适用于首次接入 USBToolBox, 以及使用串口、Modbus、I²C、SPI、显示屏、GPIO、Python 产测和 MCP 服务的操作人员。按“接入电脑 → 安装驱动 → 连接设备 → 核对引脚 → 完成一次通信”的顺序开始。

本指南以 Windows 界面为主, 附 macOS 与 Linux 安装说明。软件界面截图来自 v1.0.10 软件组件, 使用隔离的示例设备、端口和账号状态; 用于说明操作位置, 不代表真实外设通信或产测结果。第 15.1 节的设备管理器截图为实际驱动异常示例。实际 COM 号、寄存器值、设备序号以本机为准。

在软件中按 **F1**, 使用系统默认浏览器打开随软件提供的 PDF。部分笔记本需要按 **Fn + F1**。帮助无需登录、无需连接设备, 也无需联网。PDF 支持搜索、缩放和另存; 查看截图细节时可放大到 150% ~ 200%。

1. 准备电脑、设备和连接线

准备一台电脑、DshanPI USBToolBox、与外设匹配的连接线，以及外设的引脚和通信参数说明。仅有 USB-C 接口的电脑还需要支持数据传输的 USB-C 转 USB-A 转接器。

准备项	检查内容
软件安装包	从官方发布入口取得，与系统和 CPU 架构一致
USBToolBox	确认外壳接口标识清楚，插针、端子无弯曲或短路
接线	UART/SPI/I ² C/GPIO 使用合适的跳线；RS485 使用匹配的端子线
外设资料	电源电压、逻辑电平、通信模式、地址、速率、允许读写区域
辅助工具	万用表；必要时使用逻辑分析仪或示波器检查波形

1.1 先连接电脑，再连接外设

1. 暂不连接下游开发板、传感器或屏幕。
2. 将 USBToolBox 的 USB-A 插头接入电脑。首次排查时优先直连电脑，减少延长线和多级集线器。
3. 完成第 2 节驱动安装，并按第 3 节确认软件能识别设备。
4. 阅读第 4 节引脚定义，停止所有通信并断开待测板电源后接线。
5. 核对供电和信号线，再给外设上电，进行对应工具的首次测试。

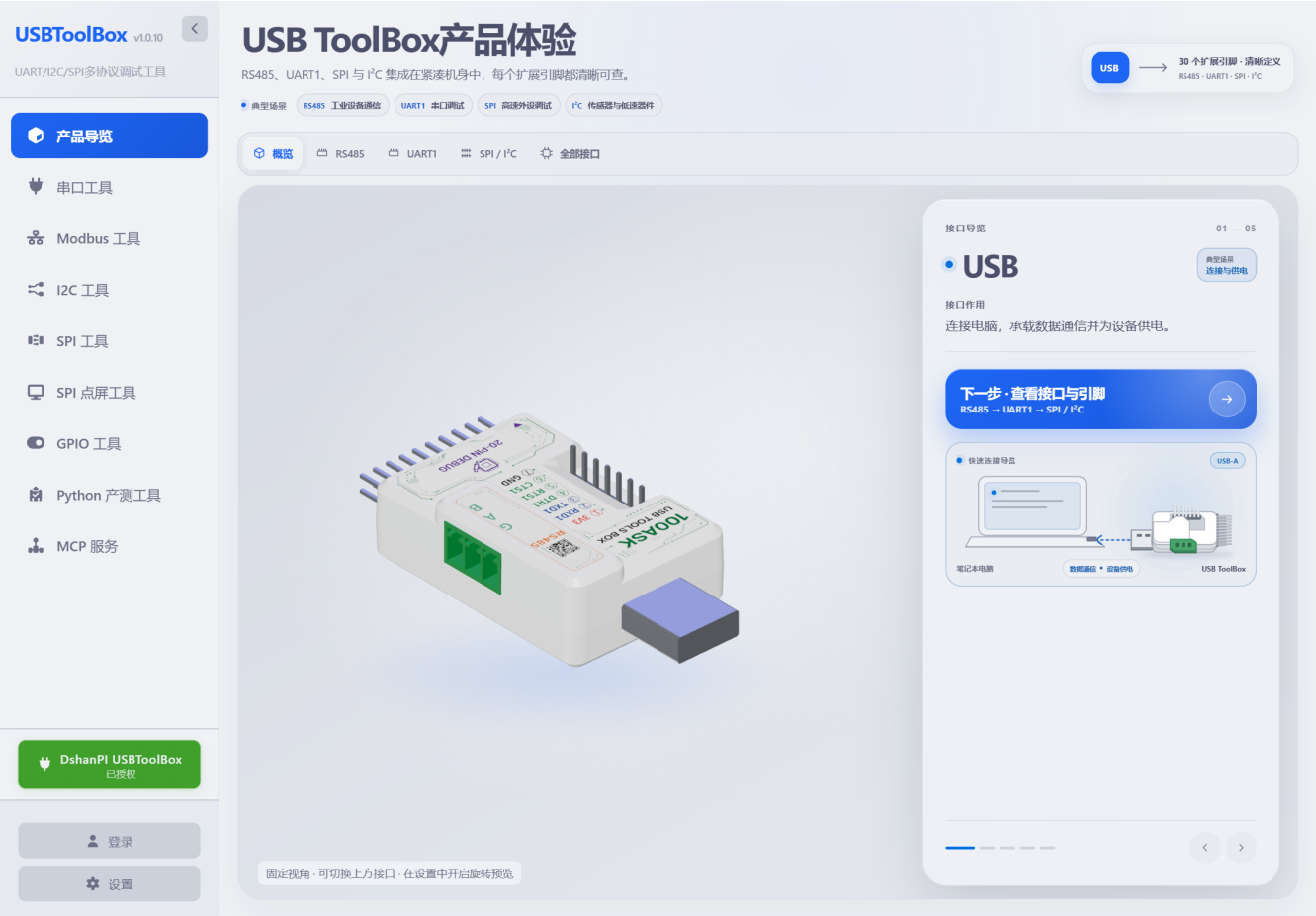


图 1：产品导览中的 USB 接入和接口选择位置

1.2 核对电平与供电

当前软件标注的数字 IO 电平为 **3.3 V**。UART1 是 TTL 串口，不是 RS232；RS485 的 A/B 也不能接到 UART 的 TX/RX。

- 将 USBToolBox 的 GND 与目标板的信号地连接。RS485 现场接地方式还应遵守设备厂商要求。
- 5V、3V3 是电源引脚名称，不表示所有信号引脚都能承受 5 V。1.8 V 或不兼容电平的外设需使用适当的电平转换。
- 外设已有独立电源时，通常只连接信号和地；未经确认，不把两路电源输出相连。
- 不用 GPIO 直接驱动电机、继电器线圈或大电流负载。需要驱动电路和独立电源。
- 软件没有给出可用于所有外设的供电电流保证。大屏、背光或较大负载应按外设规格独立供电。

检查结果：接口类型、电源电压、逻辑电平和接线方向均有依据后，再执行写入、初始化或输出电平操作。

2. 安装软件和设备驱动

2.1 选择安装包

Windows 的常见 Intel/AMD 电脑选择 x64 包。可在“设置 → 系统 → 系统信息/关于”查看系统类型。本版暂不提供 Windows ARM64 原生安装包。运行下载的安装程序，按向导完成安装，再打开 **DshanPI USBToolBox**。

本版从 [USBToolBox 官方 Release](#) 下载安装包，官网下载站同步暂缓。请展开发布页面的 Assets，选择对应系统的安装程序。不要用更新用的压缩包或签名文件替代首次安装程序。

2.2 在 Windows 安装驱动

1. 首次打开软件，若出现“安装 USBToolBox 设备驱动”，点击“安装驱动”。
2. Windows 出现用户账户控制提示时，核对安装来源并允许安装。普通账户需要管理员协助。
3. 等待安装操作结束。按提示重新插拔设备；如果要求重启，先保存其他工作再重启。
4. 返回软件查看驱动状态，然后检查设备和串口列表。



图 2: Windows 首次启动的驱动安装入口

如果先前选择了“稍后安装”，点击左下方“设置”，向下找到“WINDOWS 驱动安装”，点击“安装驱动”或“重新安装驱动”。不要为日常连接操作点击“卸载驱动”。

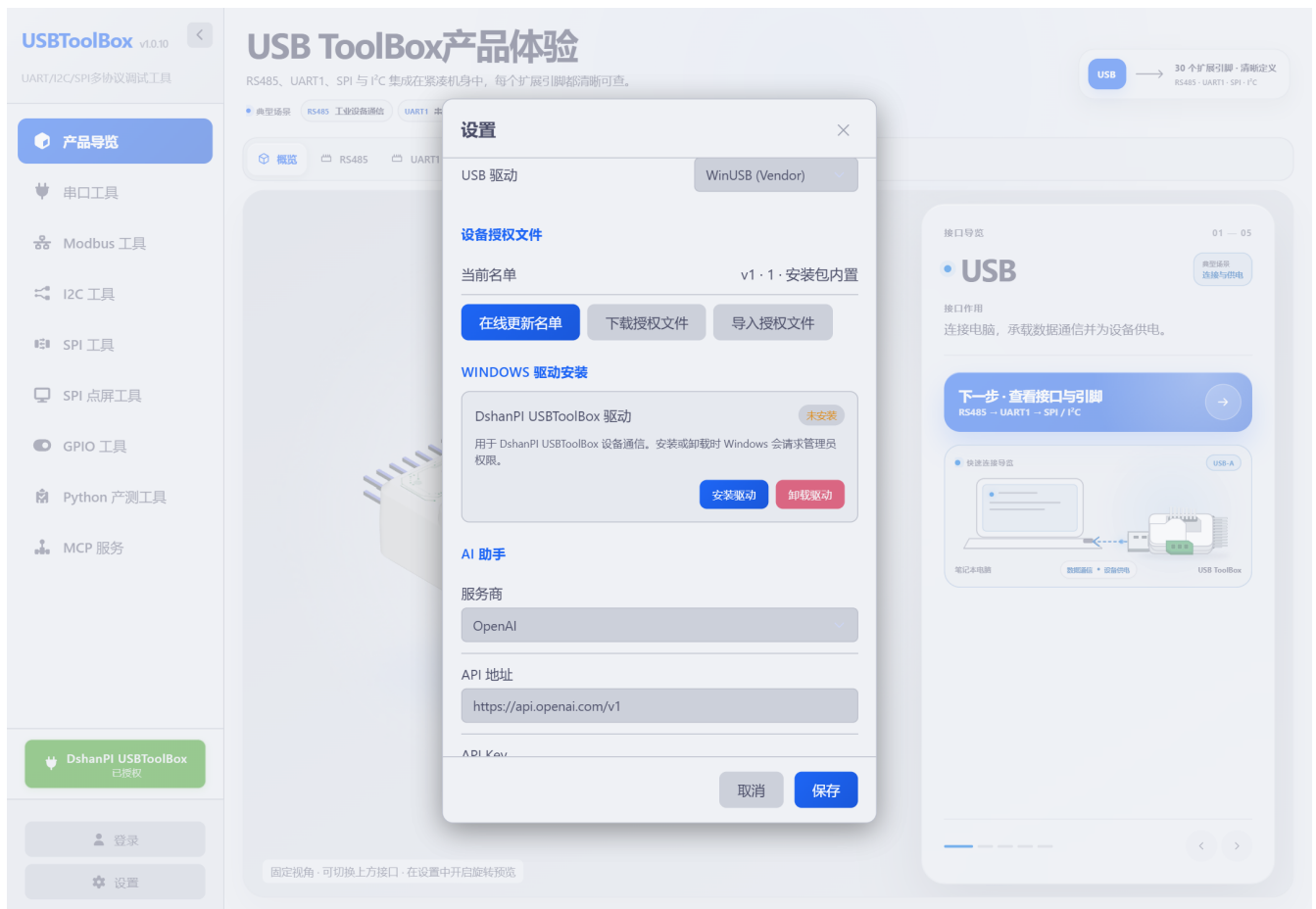


图 3：设置中的设备授权文件与 Windows 驱动安装区域

检查结果：驱动区域显示已安装；重新插拔后，Windows 设备管理器中能看到对应 USB 设备及串口，软件串口列表中出现端口。可通过插拔前后列表变化识别新增 COM 号。电脑若有多个串口，不按“列表第一个”判断 UART1 或 RS485。

“驱动已安装”与“设备已连接且已授权”是两个检查项。若设备列表仍为空，换电脑 USB 口或数据转接器后刷新；若驱动安装失败，确认管理员权限，保留错误信息后再处理。

如果安装后，设备管理器中的 **USB2.0 To Jtag/SPI** 仍带有黄色警告图标，按第 15.1 节卸载该设备，再从软件重新安装驱动。

2.3 在 macOS 安装

v1.0.10 的 macOS 包已完成候选构建和包内运行库检查，真实 Mac 的驱动激活及 USB 硬件验收仍待完成。本节给出当前安装流程，不能替代实际设备验收；正式工位投入前应验证所需接口。

1. 根据 Mac 的芯片选择对应安装包。M 系列使用 Apple Silicon/ARM64 包；Intel 机型仅在发布页面提供对应包时使用 Intel 包。
2. 打开软件 DMG，将 USBToolBox 拖到“应用程序”，从该目录启动。
3. 软件提示安装设备驱动时，点击“安装驱动”，打开包内串口驱动 DMG。
4. 按驱动窗口说明将驱动应用放入“应用程序”，打开驱动应用并点击 **Install**。
5. 按 macOS 提示允许驱动扩展。不同系统版本的入口可能在“登录项与扩展”或“隐私与安全”。可参阅 [Apple 的系统扩展说明](#)。

6. 重新插拔 USBToolBox，返回软件检查驱动状态和串口列表；串口名称通常以 `/dev/cu.` 开头，不使用 Windows 的 COM 号。



图 4: macOS 驱动引导，安装包打开后仍需完成系统批准

串口需要配套驱动；I²C、SPI、GPIO 使用软件内置 USB 接口。打开驱动安装包不等于驱动已经激活。若系统拒绝应用或扩展，先核对包来源和对应系统要求，不通过关闭系统保护来绕过排查。

2.4 在 Linux 安装

选择与发行版、CPU 架构匹配的包。DEB/RPM 包包含 USB 访问规则；安装后重新插拔设备，并确认当前账号有串口访问权限。ApplImage 的 USB 权限取决于主机配置，不能把“程序能启动”视为“USB 权限已配置”。

Linux 不安装 Windows 或 macOS 驱动。遇到权限拒绝时按发行版配置 udev 和串口用户组，优先使用正常用户运行软件。串口路径可能是 `/dev/ttyUSB*` 或 `/dev/ttyACM*`，以本机枚举结果为准。

3. 连接设备并区分授权与登录

3.1 在侧栏连接 USBToolBox

1. 确认 USBToolBox 已插入，点击侧栏底部设备区域的刷新按钮。
2. 单台设备时点击连接按钮；多台设备时按设备标识选择实际接线的那一台。
3. 等待连接和授权检查完成。鼠标停在设备按钮上可查看状态说明。
4. 若仍显示“只读预览”，先处理提示，再尝试读写。进入某工具页面并不表示该工具已获得硬件操作权限。

驱动负责让系统访问设备，设备授权负责开放软件硬件功能，百问网账号登录负责 AI 助手入口。**基础硬件工具不要求账号登录，但要求连接已授权的 USBToolBox。**

3.2 处理设备授权提示

若显示设备未登记或授权名单不可用，进入“设置 → 设备授权文件”。联网时可“在线更新名单”；离线时使用官方提供的授权文件进行“导入授权文件”。导入后按提示重新连接。

不要自行编辑授权文件。状态仍异常时，提供软件版本、设备标识和错误文字给支持人员。账号登录不能代替设备授权。

3.3 分清设备连接和端口打开

侧栏连接成功后，UART 和 RS485 仍需在具体工具中选定串口并点击“打开/连接”。I²C、SPI、GPIO 和点屏使用侧栏选中的 USBToolBox。

同一串口不要同时被两个软件或两个工具打开。切换用途前先关闭原串口；MCP 已接管设备时，先在 MCP 页面结束外部控制，再恢复桌面操作。

4. 查看引脚示意图并完成接线

4.1 在产品导览定位引脚

1. 点击左侧“产品导览”。
2. 选择“UART1”“RS485”或“SPI / I²C”查看局部接口；选择“全部接口”查看总览。
3. 对照机身 USB 插头、UART 排针、RS485 端子和双排扩展口的位置，确定方向。
4. 查看引脚卡片上的 **P 编号和信号名**，再对照实物丝印接线。需要详细说明时选择对应引脚。
5. 在“设置”开启“启用 3D 模型旋转预览”后可旋转查看；关闭时仍可使用固定视图和引脚说明。

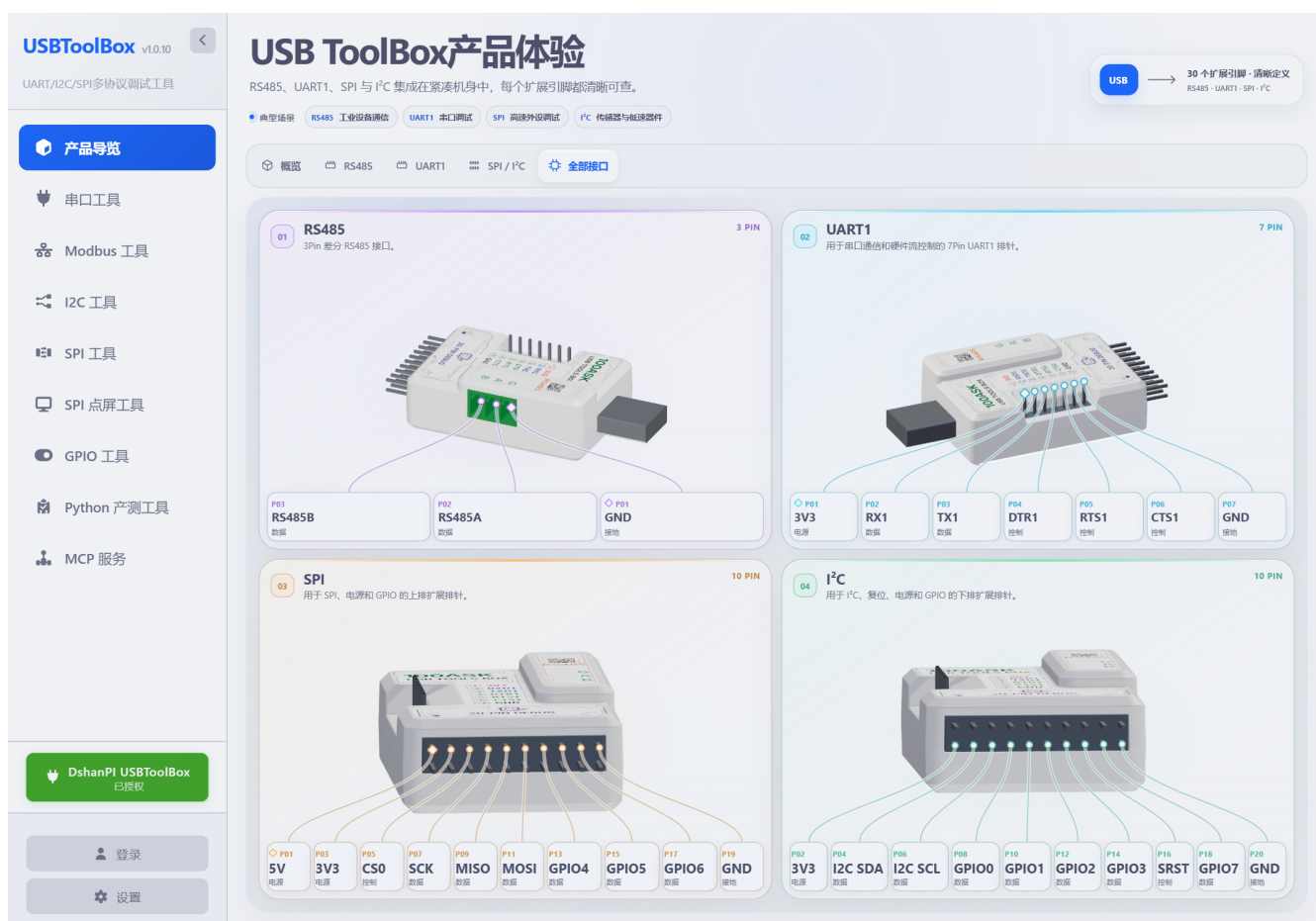


图 5：全部接口示意图，按接口分别核对 P 编号和信号名

4.2 核对 UART1 与 RS485

UART1 和 RS485 的 P1 均靠近 USB 插头。两个接口各自编号，不与扩展口 P1 混用。

UART1 编号	信号	用途
P1	3V3	电源引脚，按外设供电方案决定是否接入
P2	RX1	USBToolBox 接收，连接目标 TX
P3	TX1	USBToolBox 发送，连接目标 RX
P4	DTR1	串口控制信号，按目标要求使用
P5	RTS1	硬件流控输出，按目标要求使用
P6	CTS1	硬件流控输入，按目标要求使用
P7	GND	信号地



图 6：UART1 引脚定位

RS485 编号	信号	用途
P1	GND	信号参考地，按现场接地规范连接
P2	RS485A	差分 A 线
P3	RS485B	差分 B 线



图 7：RS485 引脚定位

4.3 核对双排 SPI / I²C 扩展口

上排 SPI 使用奇数编号，下排 I²C 使用偶数编号。P1/P2 位于靠近 UART1 的一端；以“SPI / I²C”视图方向为准，不按任意照片的左右方向猜测。

上排编号	信号	下排编号	信号
P1	5V	P2	3V3
P3	3V3	P4	I2C SDA
P5	CS0	P6	I2C SCL
P7	SCK	P8	GPIO0
P9	MISO	P10	GPIO1
P11	MOSI	P12	GPIO2
P13	GPIO4	P14	GPIO3
P15	GPIO5	P16	SRST
P17	GPIO6	P18	GPIO7
P19	GND	P20	GND

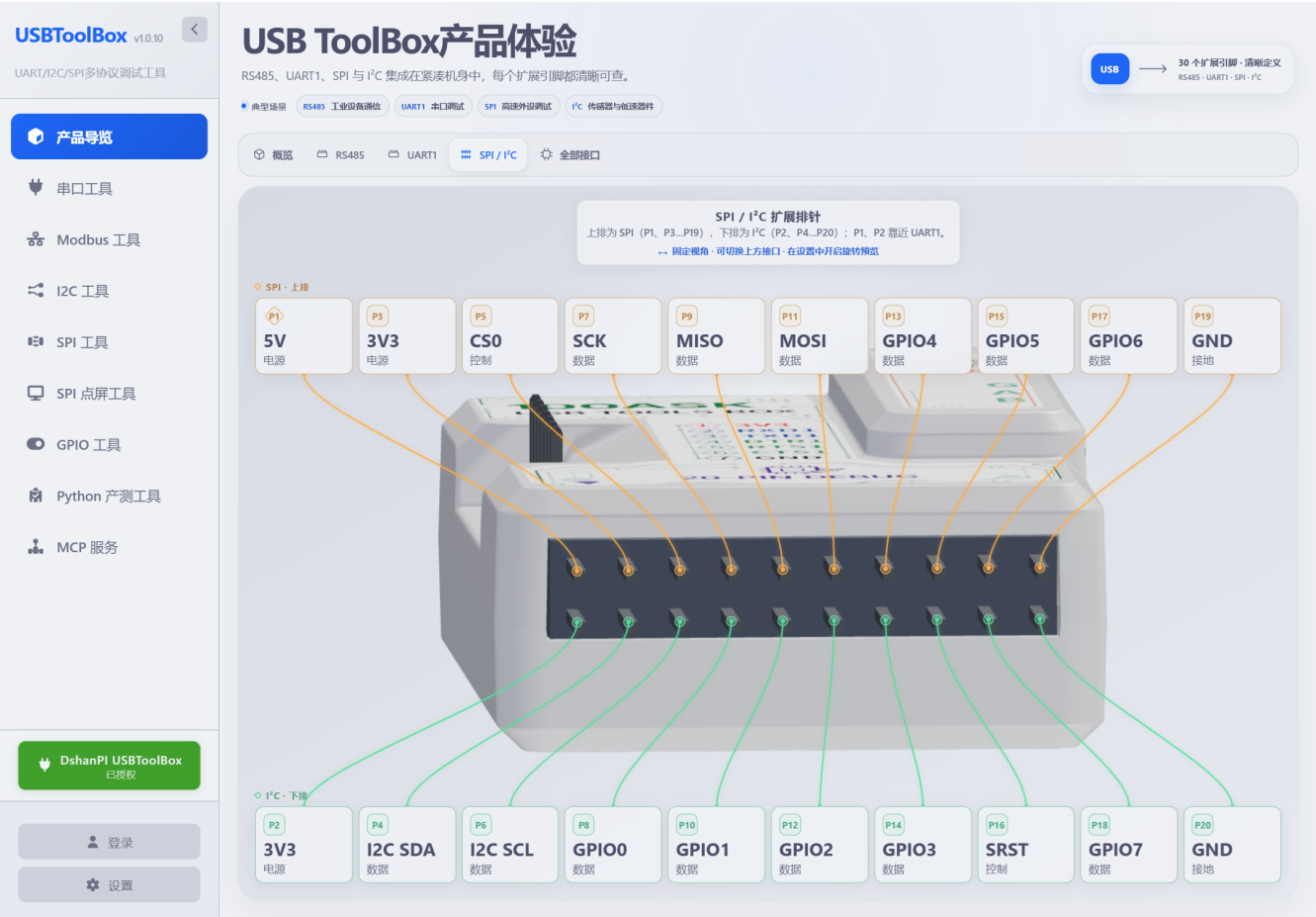


图 8：SPI / I²C 双排扩展口定位

GPIO0 ~ GPIO7 与其他用途共用物理引脚。SPI 点屏常把 GPIO4、GPIO5、GPIO6 分配为 DC、RST、BL，具体以“接线与 IO 映射”为准。SRST 不等同于屏幕使用的 GPIO 复位线，不应仅凭名称替换。

5. 使用串口工具读取日志和发送数据

5.1 连接目标串口

USBToolBox UART1	目标板 TTL 串口
P3 TX1	RX
P2 RX1	TX
P7 GND	GND

目标板独立供电时先使用上述三线连接。没有使用硬件流控时，不接 RTS/CTS，并在软件选择“流控：无”。

5.2 完成第一次收发

1. 点击“串口工具”，在下方选择已确认对应 UART1 的串口。
2. 按目标资料设置波特率、数据位、停止位和校验位。例如目标要求 115200、8 位数据、无校验、1 位停止时，选择 **115200 / 8 / 1 / 无**。
3. 点击“打开”。使目标板输出启动日志，检查接收区域是否出现可读文本。
4. 在发送框输入目标支持的指令，选择文本或 HEX，再点击“发送”。是否添加换行取决于目标协议。
5. 观察响应方向和内容，不把本地回显当成目标应答。

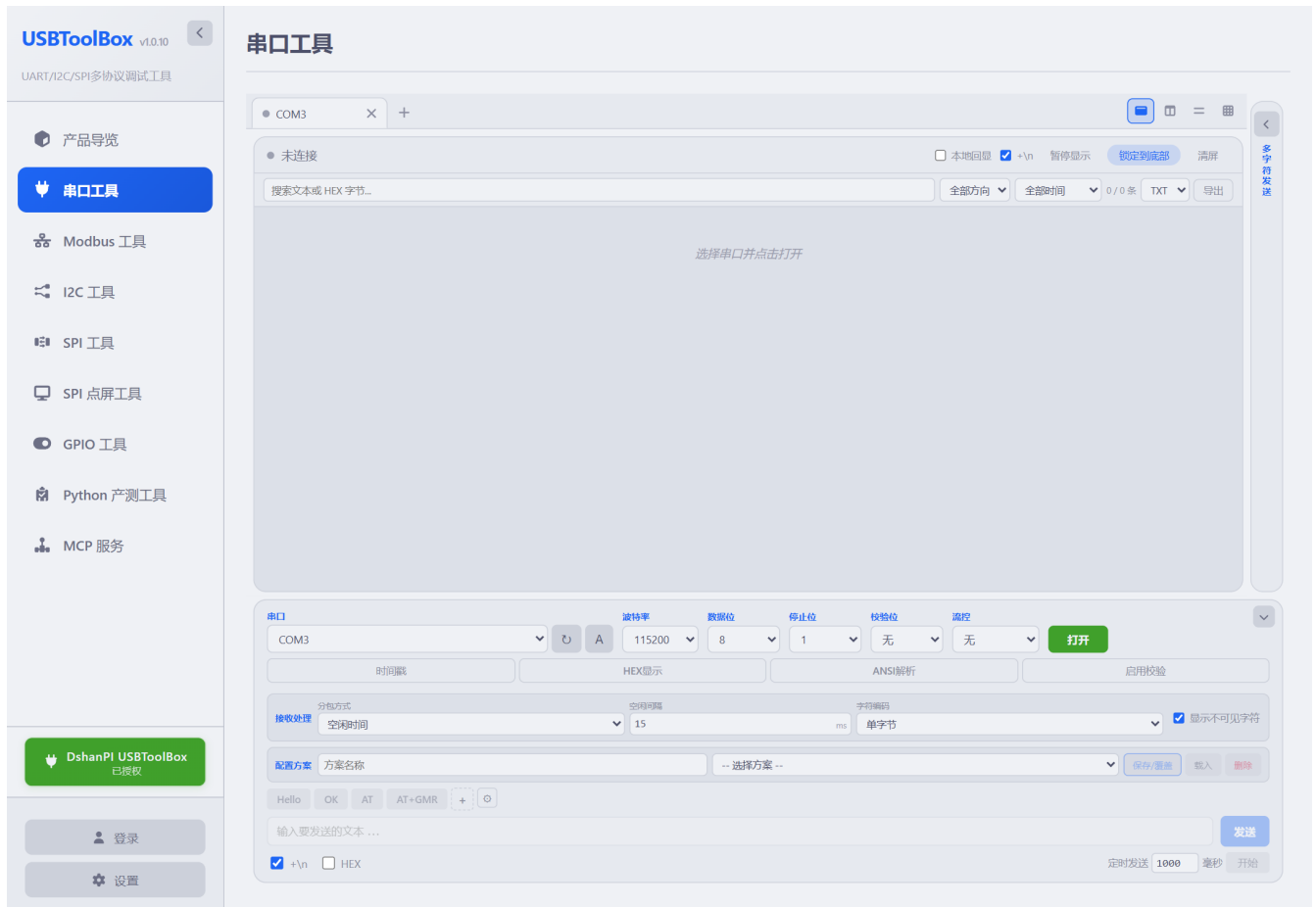


图 9：串口工具，底部配置和发送，上方查看接收内容

5.3 用回环测试检查基础链路

断开目标板，只将 UART1 的 TX1 与 RX1 短接。选择对应端口并打开，关闭“本地回显”，发送 **hello**。接收区应出现对应文本。完成后关闭端口、移除短接线，再连接目标板。

回环通过只能证明该串口的基本收发链路，不证明目标波特率、协议或接线正确。若无回显，先检查 COM 号、端口是否打开和短接引脚。

5.4 使用显示、导出和多串口功能

- “时间戳” “HEX 显示” “ANSI 解析” 控制接收内容的呈现；HEX 发送与 HEX 显示是不同设置。
- 二进制协议可使用 HEX 发送，按字节输入，如 **01 02 A0**；按协议决定是否启用校验，避免重复添加已有校验。
- 搜索、方向/时间筛选用于定位记录；导出可选择 TXT、HEX 或 CSV。“清屏”只清理显示内容，不会清除外设数据。
- 右侧“多字符发送”用于保存多条指令；定时发送前先完成一次手动通信，并确认发送间隔。
- 标签旁的“+”可新增串口会话，右上角布局按钮可改变排列。每个会话单独选择端口；关闭前停止定时发送。出现乱码时先核对波特率、校验和字符编码；始终无数据时检查 TX/RX 交叉连接、共地和目标是否实际发送。

6. 使用 Modbus 工具读写寄存器

6.1 连接 RS485 从机

将 RS485A 接目标 A，RS485B 接目标 B，信号地按设备说明连接。A/B 命名在不同厂家之间可能存在差异，先核对目标端子定义。终端电阻、偏置、屏蔽和接地按总线拓扑及外设说明配置，不给每个节点随意添加电阻。

1. 点击“Modbus 工具”，选择“主机（Master）”。
2. 底部协议选“Modbus RTU（串口）”，选择对应 RS485 端口。
3. 按从机设置波特率和校验格式，例如 9600、8N1；点击“连接”。
4. 在“读写定义”填写从机 ID、功能码、起始地址和数量。

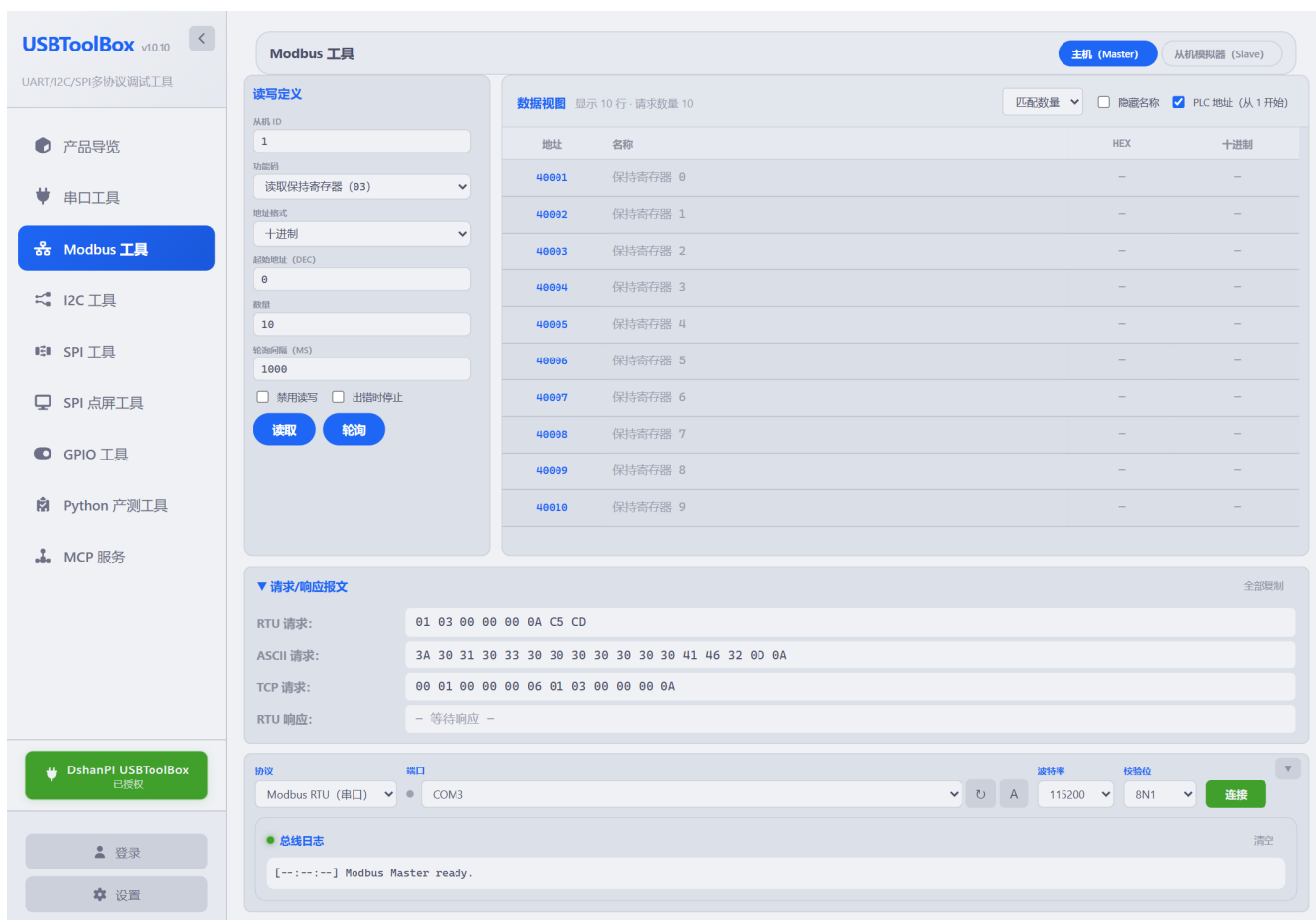


图 10: Modbus 主机的请求定义、数据视图与底部连接区

6.2 读取一组已知寄存器

先从目标说明中找到允许读取的寄存器。例如目标明确使用从机 ID 1、保持寄存器协议地址 0、数量 2：选择“读取保持寄存器 (03)”，起始地址填 0，数量填 2，点击“读取”。

检查结果：请求/响应区域出现响应报文，数据视图显示值，总线日志没有超时或协议异常。具体数值应与目标状态相符，不能仅凭收到任意字节判断成功。

Modbus 报文中的寄存器地址从 0 开始；资料中的 PLC 显示地址 40001 常对应保持寄存器协议地址 0。界面的“PLC 地址（从 1 开始）”属于显示方式，填写前仍应核对目标地址定义。[Modbus 官方协议说明](#)

6.3 使用轮询、写入和 TCP

- 单次读取稳定后设置轮询间隔，再点击“轮询”。首次调试可从 1000 ms 开始，并按设备响应能力调整；结束时停止轮询。
 - 写入前核对数据类型和地址。功能码 05/06 写单个线圈/寄存器，15/16 写多个。先选可恢复的测试项，记录原值，写后读回。
 - 多寄存器数值的倍率、符号和字序按目标协议解释，不能默认两字节就是实际温度或电压。
 - Modbus TCP 使用电脑网络连接，填写目标 IP 和端口后连接；USBToolBox 的 RS485 端子不是网口。软件的设备授权前提仍有效。
 - “从机模拟器 (Slave)” 用于对接另一台主机，配置从机 ID 与数据区后联调；使用独立串口或连接，不与主机页争用同一端口。
- 超时先检查端口、A/B、从机 ID 和通信格式；协议异常响应应按目标支持的功能码、地址范围和数量排查。

7. 使用 I²C 工具扫描和读写外设

7.1 连接 SDA、SCL 和地

USBToolBox 扩展口	I ² C 外设
P4 I2C SDA	SDA
P6 I2C SCL	SCL
P20 GND	GND
P2 3V3 (按供电方案选用)	支持 3.3 V 供电的 VCC

I²C 需要合适的上拉。确认模块是否自带上拉，以及上拉连接的电源电压；不要把信号上拉到不兼容的电压。首次测试用短线、单个外设，并从 100 kHz 开始。

7.2 扫描并读取已知寄存器

1. 点击 “I2C 工具”，确认侧栏设备已连接。
2. 选择 I2C 速率，点击 “扫描”，记录应答地址。
3. 在 “从机地址 (7-bit Hex)” 填写目标地址。若器件资料给出写地址 A0、读地址 A1，对应的 7 位地址通常为 **50**，应先确认资料使用的地址格式。
4. 按外设选择 8 位或 16 位寄存器地址，填写寄存器地址和读取长度。
5. 点击 “读取数据”，检查接收内容和日志。按需 “保存 BIN” 保留原始字节。

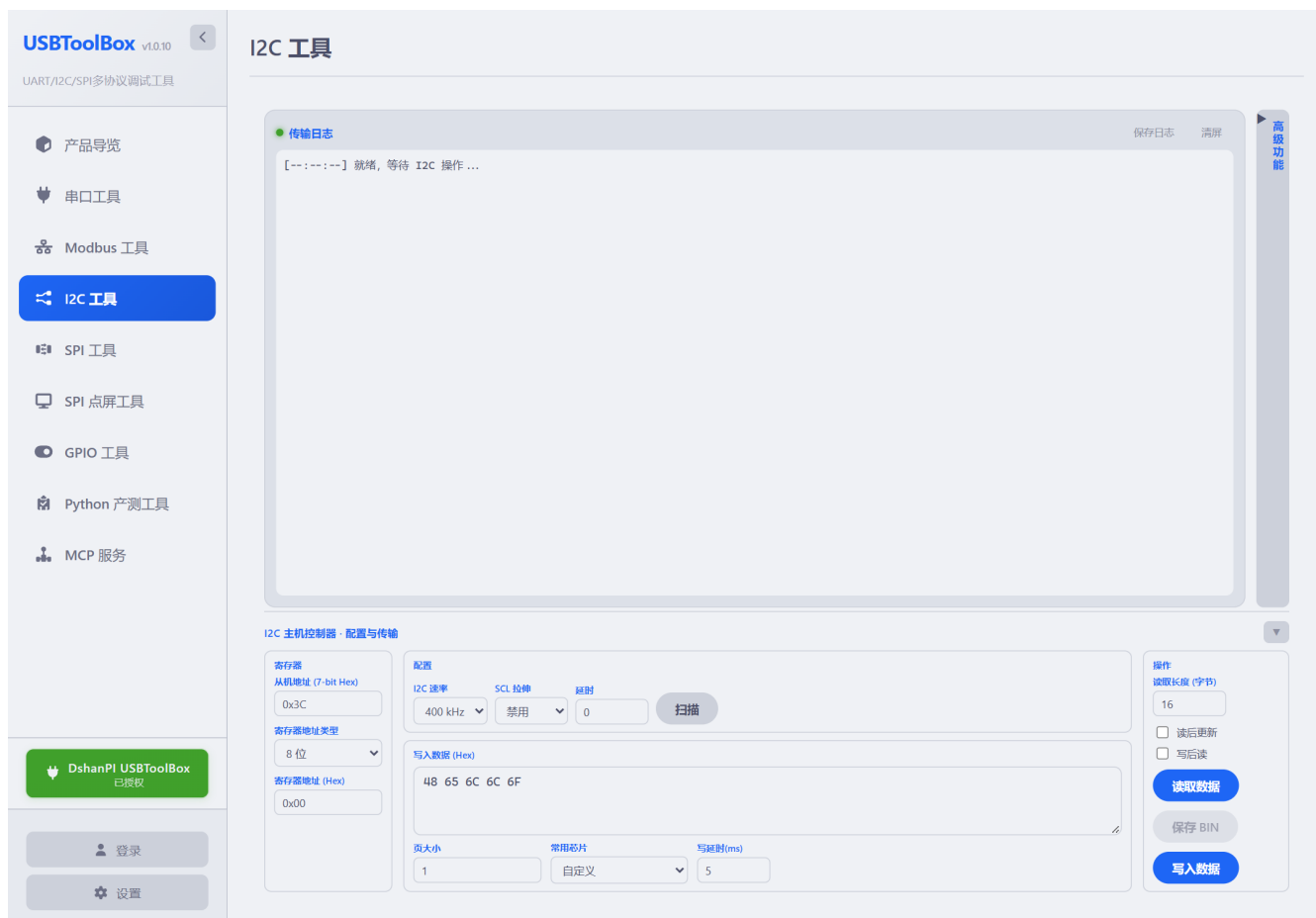


图 11: I2C 地址、寄存器、速率和读写区域

示例：对于地址脚已配置为 0x50 的 24C02，读取地址 0x00 的 16 字节时，填写从机地址 0x50、8 位寄存器地址 0x00、读取长度 16。先读取并备份，再考虑写入；芯片型号或地址脚不同不能照抄此例。

扫描有应答只说明该地址存在响应，不等于芯片型号已识别。部分命令式外设不使用常规寄存器读写格式，应按外设协议选择合适的操作方式。

7.3 写入并读回检查

1. 在“写入数据（Hex）”输入字节，先选一小段允许改写的区域。
2. EEPROM 写入需核对实际型号的页大小和写入完成时间。页面提供常用参数，但具体容量、地址方式和跨页行为以芯片资料为准。
3. 选择适当页大小与写延时，点击“写入数据”。传感器寄存器写入按协议选择不分页。
4. 使用“写后读”或再次读取相同区域，逐字节比较。
5. “高级功能”可组织更复杂的工作流；先验证单次步骤再批量执行。

读取全 FF/全 00 时不要立即认定器件损坏，先检查地址、寄存器宽度、供电、上拉、接线和器件当前数据。写入失败后先读回确认，不连续重发可能改变设备状态的命令。

8. 使用 SPI 工具验证传输

8.1 连接 SPI 四线和地

USBToolBox 扩展口	SPI 外设
P7 SCK	SCK / CLK
P11 MOSI	MOSI / SDI / DIN
P9 MISO	MISO / SDO / DOUT（需要读回时）
P5 CS0	CS / CS#
P19 GND	GND

供电另按外设要求连接。MOSI 是主机发往外设，MISO 是外设返回主机；不要按 UART 的方式交叉 MOSI/MISO。无读回接口的屏幕可以不接 MISO。

8.2 初始化并执行单次传输

1. 点击“SPI 工具”。按外设资料设置 Mode 0 ~ 3、速率、CS0、数据位宽和位序。
2. 首次测试选择较低速率，如 468.75 kHz 或 937.5 kHz；设置位宽 8、MSB 的前提是外设明确支持。
3. 展开“接线与 IO 映射”，核对 SCK、MOSI、MISO、CS 和辅助 IO。
4. 点击“初始化 SPI”。在“单次传输”输入数据，按任务选择“写入”“读取”或“全双工”。
5. 查看 SPI 日志，按需保存接收数据为 BIN。

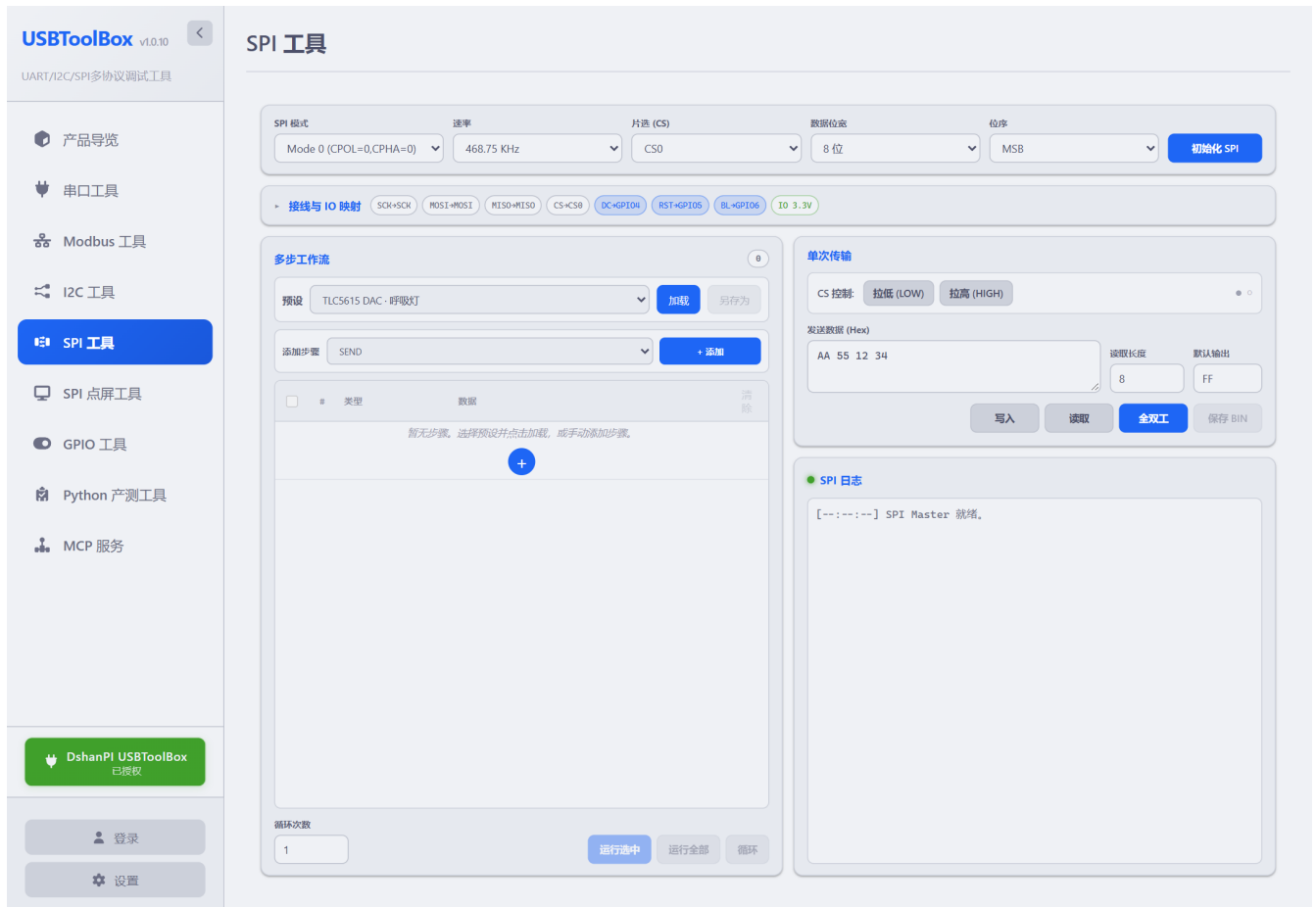


图 12: SPI 配置、多步工作流和单次传输

对于明确支持 0x9F JEDEC ID 命令的 SPI NOR Flash，可按其要求配置模式后，使用全双工发送 9F FF FF FF，查看命令之后返回的识别字节，并与该芯片资料比较。此命令不是所有 SPI 外设的通用测试命令。

“写入”过程中收到的状态不能替代读回内容；“全双工”会在发送时同时采样接收。命令、空周期、数据顺序和 CS 边界由外设协议决定。

8.3 组织多步工作流

需要保持 CS 或穿插 DC/RST 时，在左侧添加 SEND、DUPLEX、CS LOW/HIGH、辅助 IO 和 DELAY 步骤。先运行单个选中步骤，再运行完整序列；循环前确认每一轮都能结束到预期状态。

停止循环后仍需核对目标状态。停止软件任务不会撤销已经发出的写入，也不能保证目标立即回到复位状态。

SPI 全 00/全 FF 或不稳定时，依次核对模式、CS、MISO、位序和时钟；降低速率并缩短线长。需要确认时序时使用逻辑分析仪，不仅看发送成功提示。

9. 使用 SPI 点屏工具显示文字和图像

9.1 连接屏幕并选择参数

先确认屏幕是本工具支持的 SPI 接口类型，核对控制器型号、分辨率、供电和背光要求。I²C 屏不能仅凭外形相同接到 SPI 点屏接口。

软件信号	常见默认映射	屏幕侧名称示例
SCK / MOSI	P7 / P11	SCL 或 CLK / SDA 或 DIN
CS	P5 CS0	CS
DC	P13 GPIO4	DC / A0
RST	P15 GPIO5	RES / RST
BL	P17 GPIO6	背光控制端（仅在电气条件允许时）
GND	P19 或 P20	GND

屏幕上标注 SDA/SCL 不一定表示 I²C，应以模块接口模式和资料为准。BL 仅是控制映射，不能据此假定 GPIO 可以直接给背光供电。

9.2 初始化并发送文字

1. 点击“SPI 点屏工具”，核对“接线与 IO 映射”。
2. 在“屏幕预设”选择实际控制器与分辨率匹配的型号，点击“加载”。没有匹配项时按屏幕资料配置，不任意套用相近型号。
3. 检查 SPI 模式、频率、分辨率、偏移和初始化命令；按需要复位，再点击“发送初始化”。
4. 在“文本”页输入短文本，设置字号和颜色，先用全屏刷新发送。
5. 对比实际屏幕与“画面预览”，确认方向、边界、颜色和字符形状。

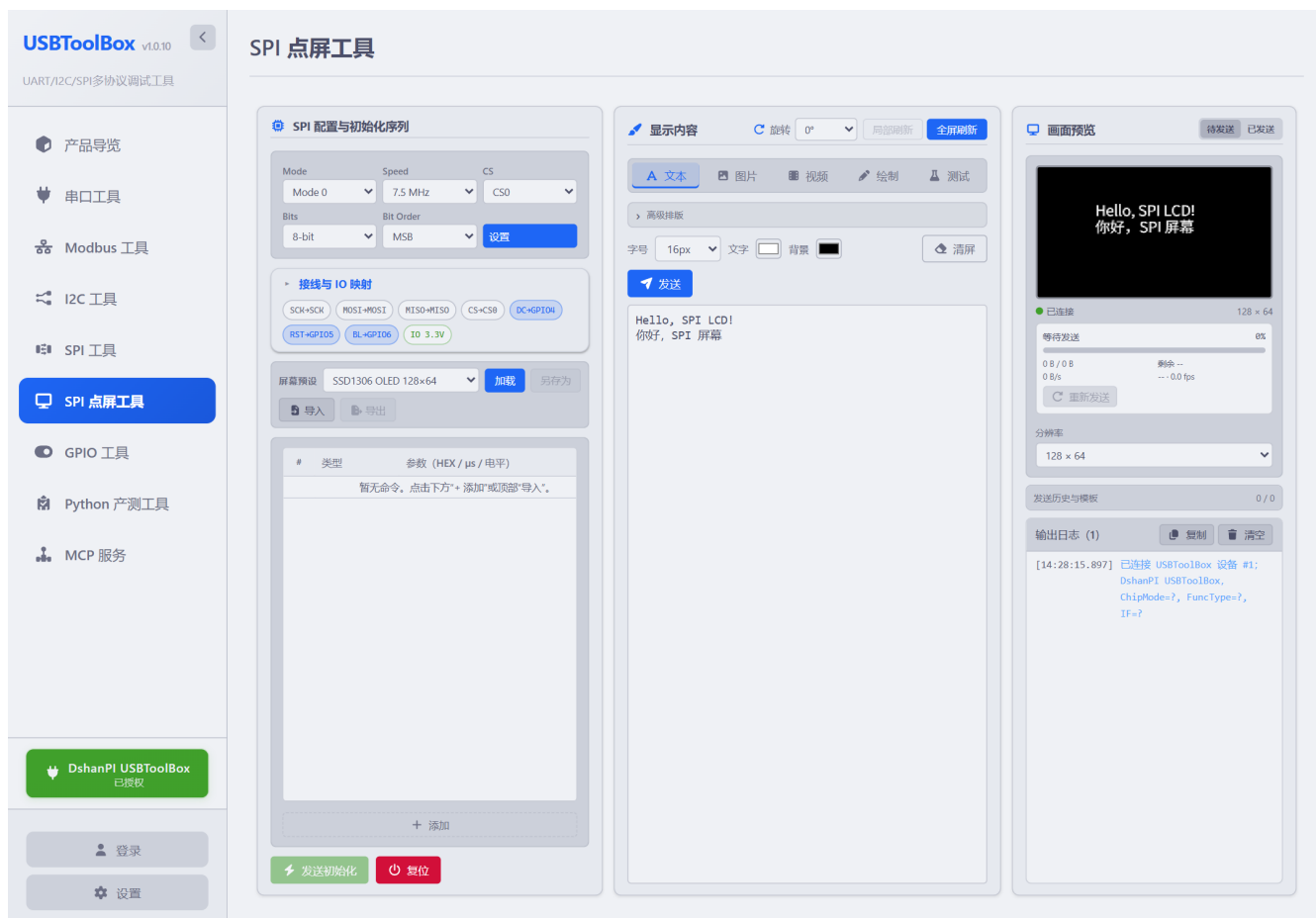


图 13：点屏参数、显示内容、待发送与已发送预览

9.3 调整图像和刷新

文本正常后再尝试“图片”“绘制”“测试”，最后使用视频或局部刷新。RGB565 屏还应核对颜色字节序、行列偏移和屏幕方向；单色屏则核对页格式与分辨率。

“待发送”表示编辑后的预览，“已发送”表示软件记录的发送内容，都不是摄像头采集的屏幕实况。进度完成后仍要检查物理屏幕。

白屏或黑屏时先查供电、背光、复位和 DC/CS，再核对初始化；图像偏移查分辨率与行列偏移，颜色异常查像素格式和字节序。多次随机初始化不能代替确认屏幕型号。

10. 使用 GPIO 工具读取和输出电平

10.1 先读取输入

1. 根据第 4 节找到目标 GPIO，引脚序号与软件行中的 GPIO 名称对应。
2. 将外设信号接到目标 GPIO，连接共地，确认输入电平兼容。
3. 点击“GPIO 工具”，选择该行“设为输入”，点击“读取状态”。输入模式会自动刷新状态。
4. 改变一个已知输入信号，观察 HIGH/LOW 是否对应变化。悬空输入不适合作为稳定性判断依据。

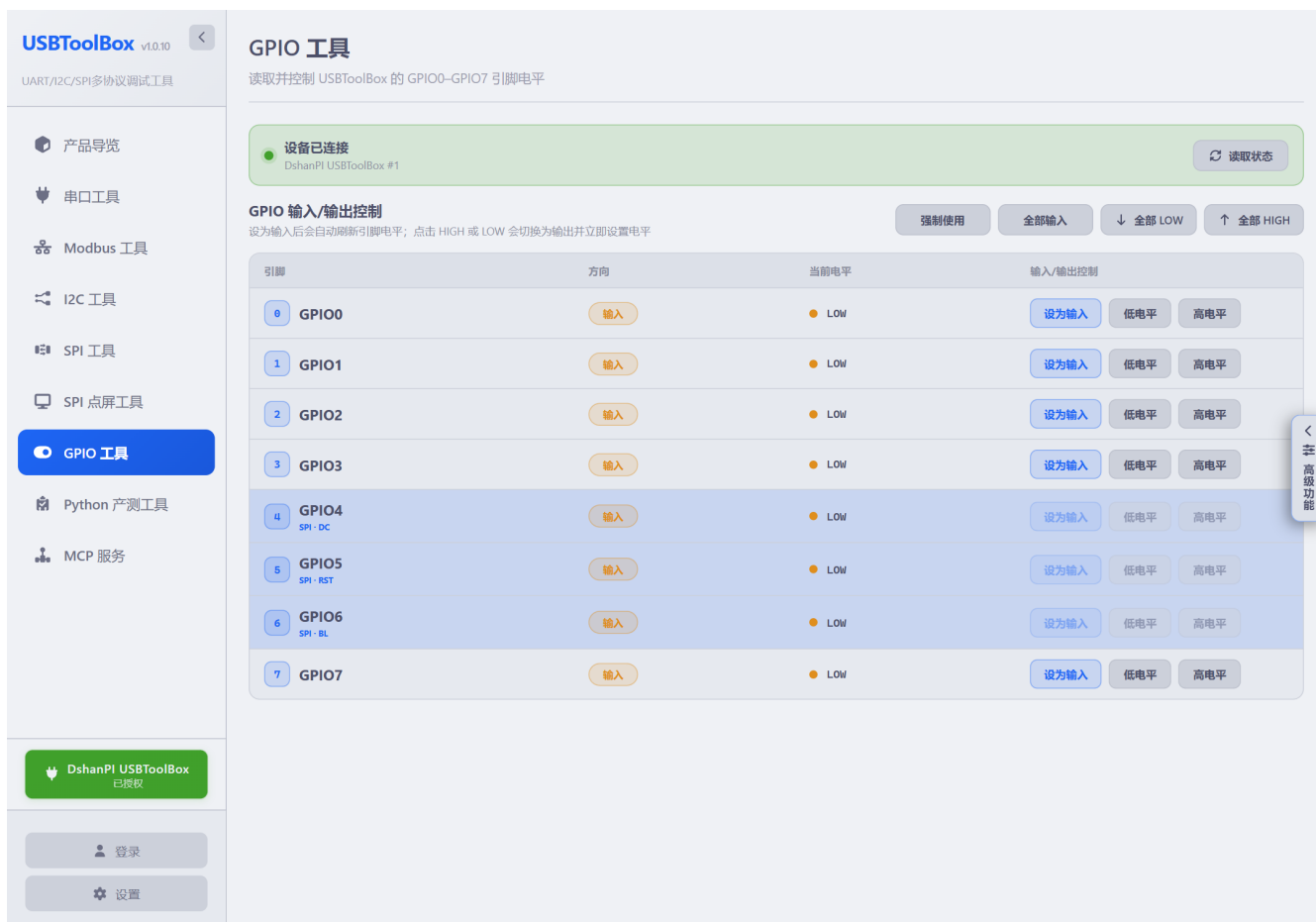


图 14：GPIO 输入、输出和 SPI 辅助信号占用标识

10.2 输出电平并验证

先接高阻测量仪器或具有限流电路的测试负载。点击该行“低电平/高电平”会立即切换为输出并设置电平，使用万用表或示波器验证。

GPIO4/5/6 等行可能显示 SPI 的 DC/RST/BL 占用。“强制使用”会允许对共用引脚操作，使用前停止对应 SPI/点屏任务并确认不会同时驱动。全局“全部输入/LOW/HIGH”操作影响范围更大，首次调试优先逐引脚操作。

“高级功能”用于定时或序列输出。先用静态电平确认接线，再配置时序，并通过波形测量确认实际输出；软件调用间隔不能当作高精度波形测量值。

11. 使用 Python 产测工具组织测试

11.1 运行第一个脚本

1. 连接已授权 USBToolBox，点击“Python 产测工具”。内置运行环境无需另装 Python。
2. 保持“手动运行”，使用初始脚本 `print("hello python")`，点击“运行”。
3. 输出区域应出现文本，脚本随后结束。此步骤不读写硬件，用于检查运行环境。
4. 点击“+”新建脚本，或“打开”已有 .py 文件。通过标签切换编辑内容；窗口最大化后代码区会随可用空间扩展。

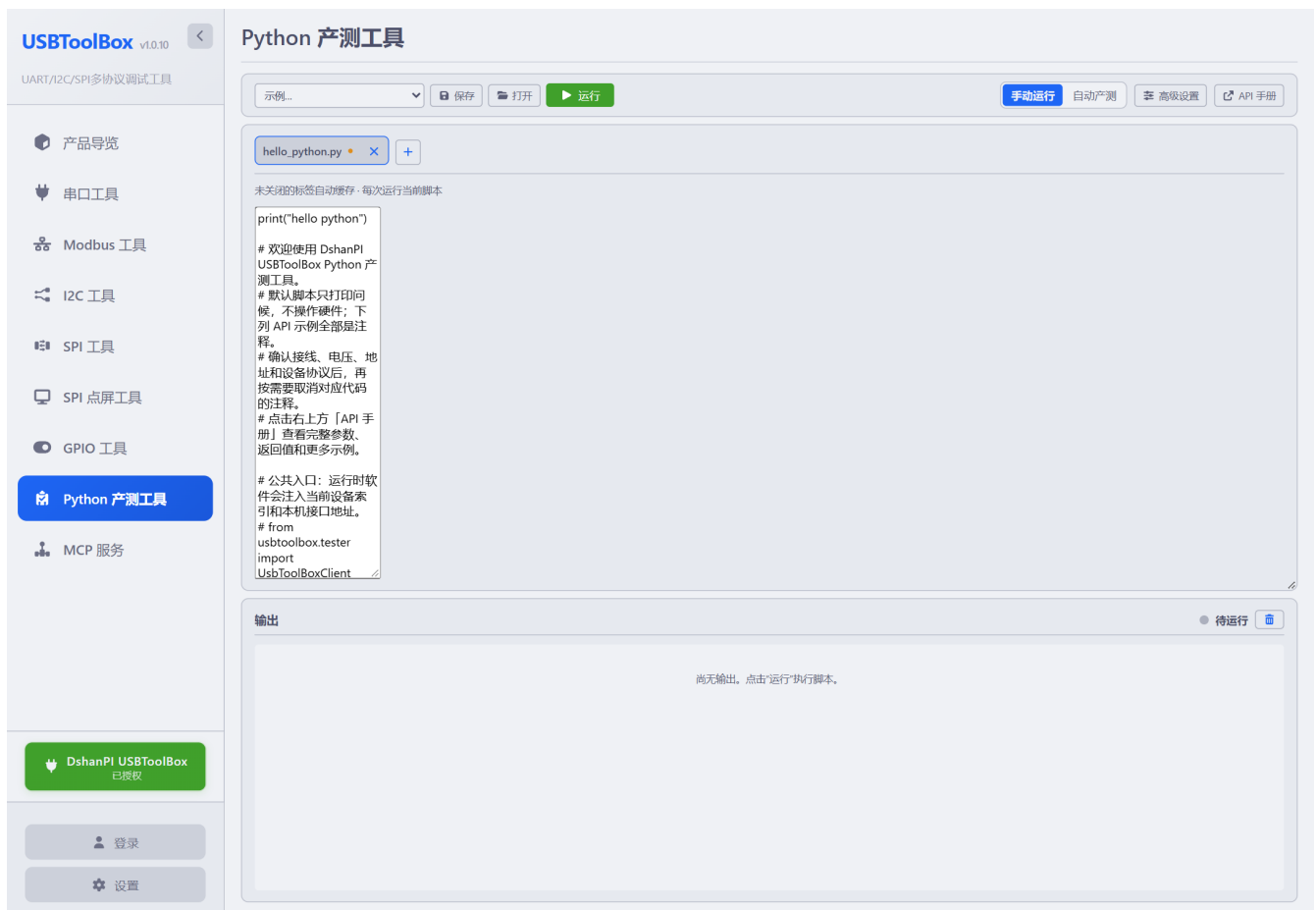


图 15: Python 多脚本工作区与运行输出

未关闭的标签会自动缓存，重启后恢复内容；“保存”才会生成正式 .py 文件。关闭未保存标签时可选择保存、取消或“不保存关闭”。选择不保存会删除该标签的缓存；新标签复用当前空闲的最小 `untitled_N.py` 编号，不会一直累计。

11.2 将硬件操作写成脚本

“API 手册”打开离线接口参考。先照第 5 ~ 10 节用图形界面验证外设，再把已确认的参数迁移到脚本。

```
from usbtoolbox.test import UsbToolBoxClient
```

```
client = UsbToolBoxClient()
index = client.device_index()
print("本轮设备索引:", index)
print("脚本环境检查完成")
```

使用 `client.device_index()` 获取当前设备，不写死设备索引。串口脚本在“高级设置”选择正确的脚本串口，再用 `client.serial_port()` 读取；选择框留空时由脚本明确指定，不假定第一个端口就是当前工位。

软件内运行会准备本机接口；通常不必手动启动外部 API 服务。外部脚本调用才需要核对高级设置中的地址与端口。脚本没有交互式标准输入面板，不用 `input()` 等待现场输入；参数应预先写入脚本或配置。

测试失败应返回非零退出码，例如 `raise SystemExit(1)`。仅打印 **FAIL** 后正常退出，仍会被视为脚本正常结束。正式工位还需由脚本保存产品编号、时间、测试值及合格结论，界面日志不是产测数据库。

11.3 开启自动产测

1. 先手动运行并验证当前脚本，保存正式版本。
2. 点击“自动产测”。选择模式本身不会运行当前已经接入的设备。
3. 拔出 USBToolBox，再接入一台待测 USBToolBox；枚举和授权通过后自动运行一次。
4. 等待该轮结束，检查输出和结果，再更换下一台。
5. 结束工位工作时停止自动产测。离开 Python 页面会关闭自动模式并停止当时的自动测试；重启软件后也不会自动恢复执行。

这里的触发源是 **USBToolBox 本体的 USB 插拔**。USBToolBox 一直连着电脑、只更换它后面的待测板，不会触发新一轮。此类工位应由脚本检测握手、GPIO 到位信号或治具按钮，再执行单板测试。

自动模式和运行过程中，脚本编辑、切换等操作会受限，防止测试途中修改内容。多台设备同时接入、设备未授权、串口缺失或连接失败时，先按提示排查，不让脚本猜测目标或无限重试写入。

12. 登录并使用 USBToolBox 助手

12.1 配置 AI 服务

1. 连接并授权 USBToolBox，点击侧栏“登录”，按登录窗口流程完成百问网账号登录。
2. 打开“设置 → AI 助手”，选择服务商或自定义兼容接口。
3. 填写所选服务商的 API 地址、API Key 和模型，保存设置。账号登录不替代 AI 服务商的 Key。
4. 返回工具页。具备设备权限且已登录时，侧栏登录按钮上方显示“USBToolBox 助手”。未登录时该入口隐藏，与各工具保持一致。



图 16：设置 AI 服务商、接口地址、Key 和模型

请求会发送到配置的 AI 服务商。不要在提示词、代码或附件中放入无关密钥和客户敏感数据；是否允许发送技术资料按所在单位要求执行。

12.2 请求解释、生成配置或生成脚本

描述接口类型、外设型号、接线、通信参数、已观察到的错误和希望完成的动作。先让助手解释或生成草稿，再检查地址、数据和作用范围。

串口、Modbus、I²C、SPI、GPIO、点屏工具的助手用于解释与填入配置草稿。填入配置不等于硬件已执行，连接、发送和运行仍需在工具中确认。

Python 页面提供脚本助手，默认展开，可通过同一侧栏入口收起；普通收起保留草稿。检查生成代码后应用到编辑器，再手动运行验证。退出登录会关闭 AI 界面并停止请求，脚本区内容保留。

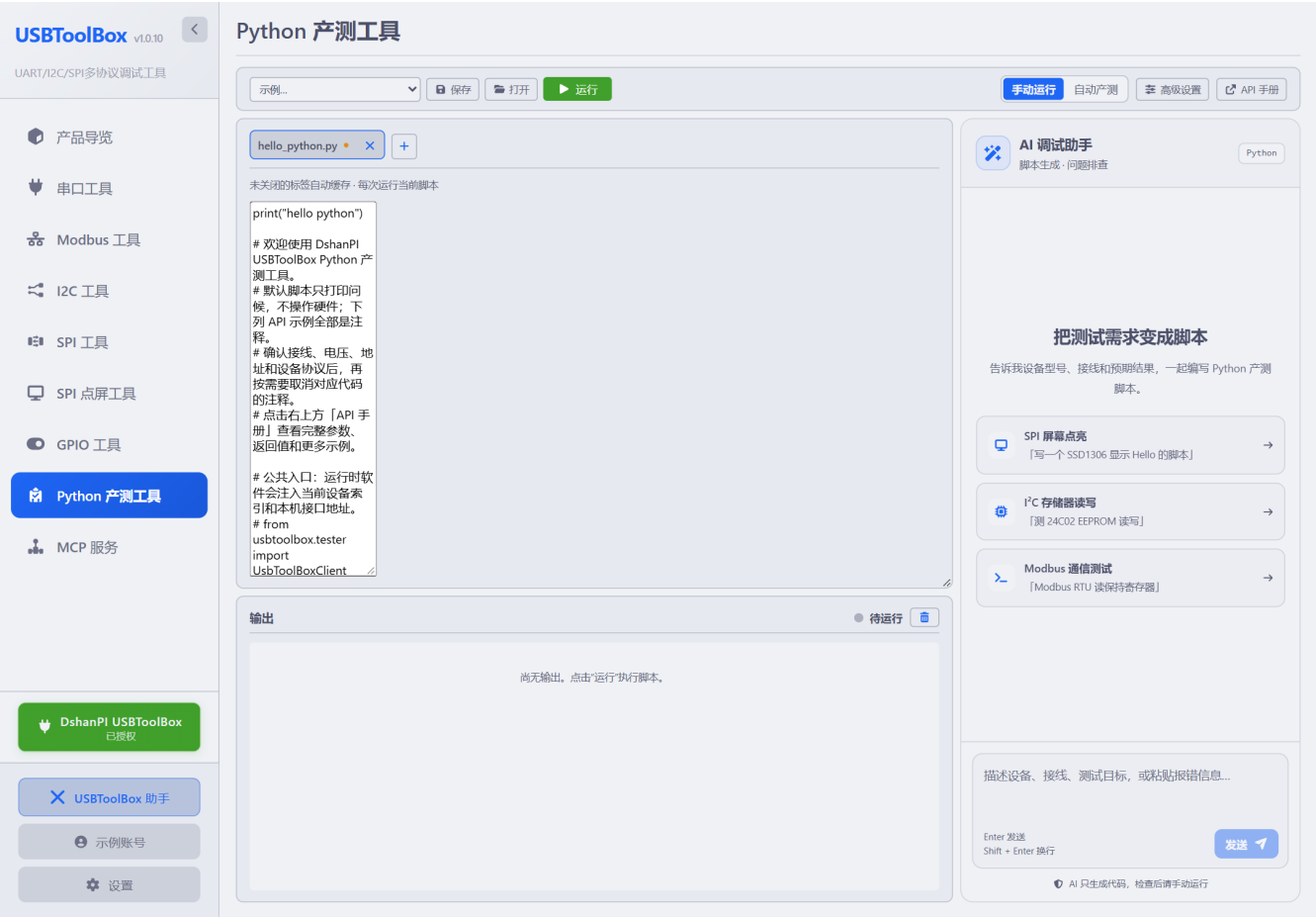


图 17：Python 脚本助手与侧栏入口（示例登录状态）

13. 通过 MCP 服务接入外部 Agent

MCP 服务允许外部 Agent 在授权范围内实际操作硬件，与只生成配置草稿的内置聊天助手不同。外部 Agent 必须能访问运行 USBToolBox 的这台电脑；云端 Agent 的 127.0.0.1 不指向本机。

13.1 设置接口范围并启动服务

1. 先停止正在运行的本地脚本、串口和工作流，连接已授权 USBToolBox。
2. 打开“MCP 服务”，核对本机地址。默认是 `http://127.0.0.1:8766/mcp`，与 Python API 的端口不同。
3. 选择需要允许的 UART/RS485、I²C、SPI、GPIO 接口。只启用本次需要的接口。
4. 启用串口接口时，明确选择 UART1 与 RS485 对应端口，两者不能选同一个端口。
5. 首次使用选择“请求批准”，设置接入方式和空闲释放时间。
6. 核对客户端、接线和所选操作范围，勾选确认，再点击“启用 MCP 服务”。

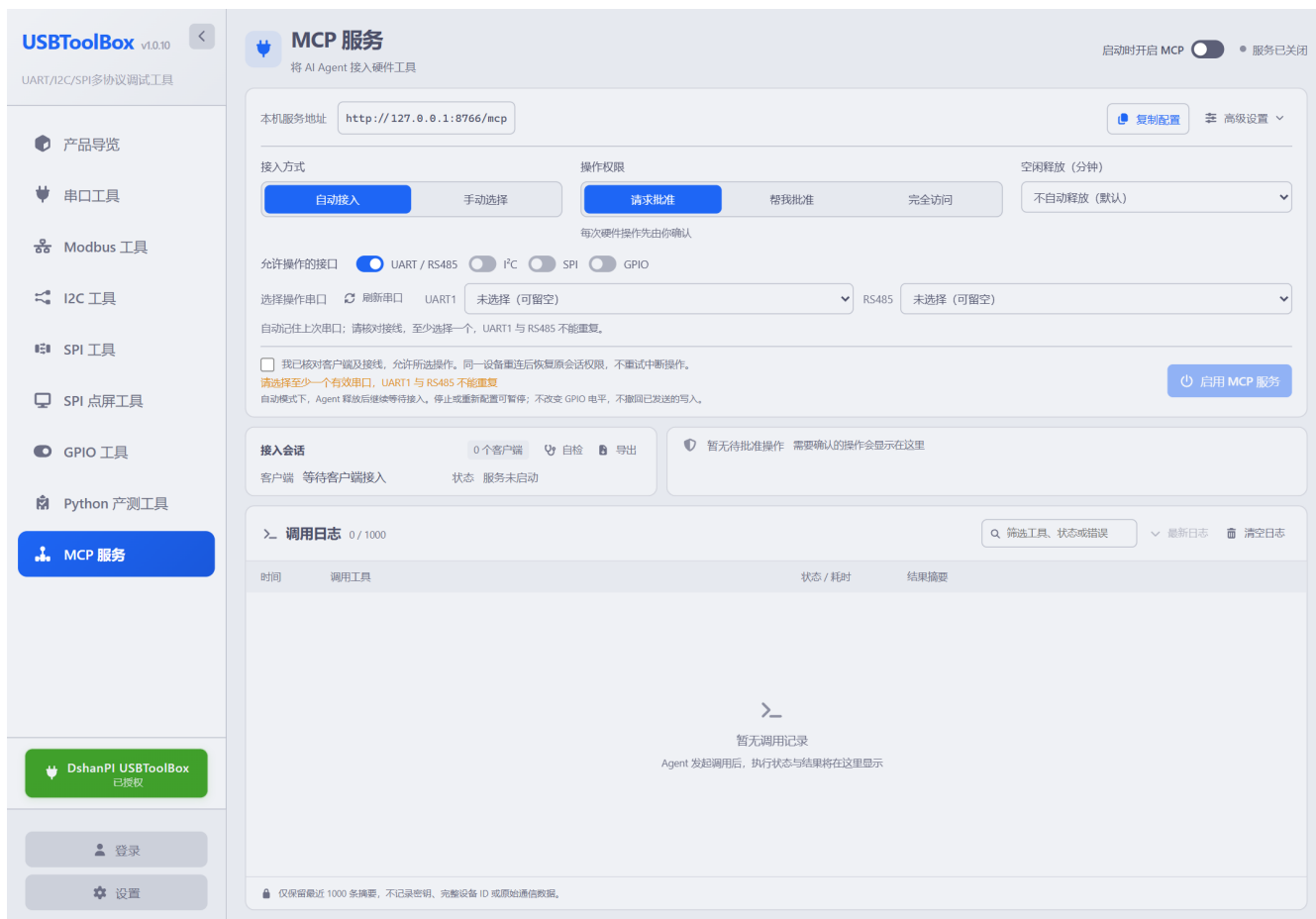


图 18：MCP 接口范围、权限、串口选择与客户端状态

13.2 配置客户端并批准请求

点击“复制配置”，按软件提供的客户端格式粘贴到客户端。配置包含访问密钥，应作为凭据保管，不贴入公开聊天、截图或日志。

自动接入适合只有一个明确客户端的场景；多个客户端时使用手动选择。客户端连接后，检查“接入会话”和待批准区域，再批准预期操作。先从状态查询或已确认的读取任务开始。

权限	使用方式
请求批准	每次硬件操作先由人工确认，适合首次接入
帮我批准	自动放行允许的查询类操作，其他操作仍需确认
完全访问	在已选接口和串口范围内自动执行；只在明确理解任务后开启

13.3 查看日志并结束外部控制

查看调用日志中的工具名、状态和结果摘要；“自检”用于检查接入状态，“导出”用于保留诊断信息。日志最多保留最近 1000 条摘要，不是完整总线抓包。

结束时点击“停止 MCP 服务”，确认外部控制已释放后再操作桌面工具。停止不会撤回已发送的写入，也不会自动把 GPIO 改回某个电平；按工位要求检查实际设备状态。

“启动时开启 MCP”指启动本软件时按确认过的配置启动服务，不是 Windows 开机启动，也不会恢复旧会话或重放中断命令。先验证手动接入流程，再考虑启用。

14. 调整界面、保存设置和查阅帮助



图 19：设置中的侧栏、页面恢复和 3D 预览开关

- “侧边栏默认折叠”控制默认布局，折叠后可点击侧栏顶部按钮展开。
- “启动时恢复上次使用的工具”保存后生效；恢复页面，不恢复运行脚本或硬件操作。MCP 的启动配置独立生效。
- “启用 3D 模型旋转预览”控制产品导览的旋转模式，关闭后仍可查阅固定视图和引脚。
- 修改设置后点击“保存”；主题、语言等选项按界面提供的行为预览和应用。
- 按 **F1** 打开本指南。浏览器标签保留时，退出软件后当前已加载内容通常仍可阅读，但重新载入本地地址需要软件运行；长期留存请使用浏览器另存 PDF。
- 浏览器若设置为下载 PDF 而非内嵌显示，会按该浏览器偏好处理；调整浏览器 PDF 设置后重试 F1。
- 更新前保存脚本和重要配置，停止通信、自动产测和 MCP 控制，再在“设置 → 软件更新”检查更新。

15. 常见问题（FAQ）与故障排查

15.1 处理驱动安装后的黄色警告图标

问题：安装驱动后，设备管理器中的 USB2.0 To Jtag/SPI 仍显示黄色警告图标，如何处理？

在 Windows 设备管理器中展开“外部接口”，如果看到以下情况，先卸载该设备，再返回 USBToolBox 软件重新安装驱动。

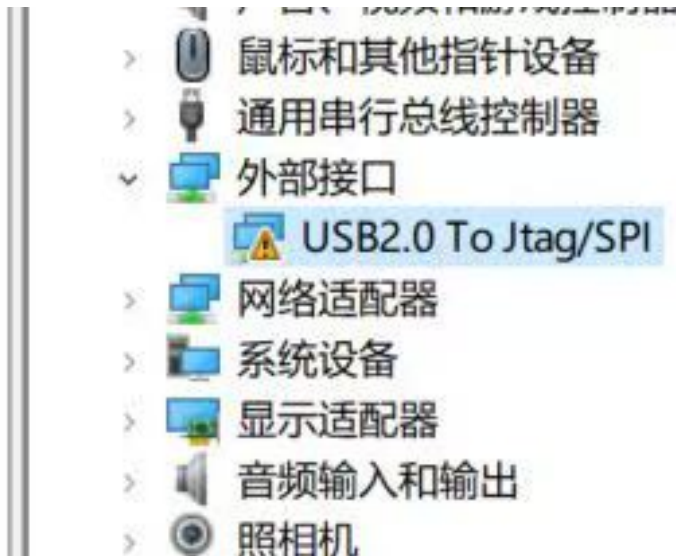


图 20：实际异常示例，USB2.0 To Jtag/SPI 左侧显示黄色警告图标

1. 保持 USBToolBox 与电脑连接。右键点击 Windows “开始” 按钮，打开“设备管理器”。
2. 展开“外部接口”，找到带黄色警告图标的 **USB2.0 To Jtag/SPI**。右键点击该设备，选择“卸载设备”。
3. 在确认窗口中确认卸载，等待操作完成。此处的“卸载设备”在 Windows 设备管理器中操作。
4. 返回 USBToolBox，打开左下方“设置”→“WINDOWS 驱动安装”，再次点击“安装驱动”。如果按钮显示“重新安装驱动”，点击该按钮。按 Windows 提示允许管理员操作，并等待安装完成。
5. 按安装提示重新插拔 USBToolBox；如果要求重启，保存工作后重启。返回软件刷新设备和串口列表。

检查结果：设备管理器中对应设备的黄色警告图标消失，软件能够识别 USBToolBox 和所需串口，再按第 3 节连接设备。

如果警告图标仍然存在，右键点击该设备，打开“属性”，记录“设备状态”中的错误信息和错误代码，连同软件驱动安装提示提交排查。

15.2 按现象快速排查

现象	先检查	下一步
安装驱动后设备带黄色警告图标	设备管理器中是否为 USB2.0 To Jtag/SPI	按第 15.1 节卸载该设备，再从软件重新安装驱动
软件设备列表为空	USB 接触、数据转接器、系统是否枚举	换电脑接口、重新插拔并刷新；再检查驱动
驱动已安装但仍只读	侧栏是否连接、设备授权状态	更新或导入官方授权名单，按提示重连
没有助手按钮	是否登录、设备是否授权、是否被 MCP 控制	登录并恢复桌面操作权限
串口打不开	COM 号是否存在、其他软件是否占用	关闭占用端口的会话，重新选择端口
串口乱码或不回应	波特率、校验、TX/RX、共地、换行	先做断开外设的回环测试，再验证目标协议
Modbus 超时	端口、A/B、从机 ID、通信格式	单次读取已知有效地址，暂停轮询后排查
I ² C 无应答	供电、SDA/SCL、上拉、7 位地址	单外设短线降速测试
SPI 读回固定值	模式、CS、MISO、命令及空周期	降速并测量波形，不连续尝试写命令
屏幕不显示	电源、背光、DC/RST/CS、控制器型号	核对预设和初始化，先发送简单测试图
GPIO 电平与预期不同	输入/输出模式、引脚复用、外部驱动	停止其他工具，逐引脚测量
Python 显示 FAIL 但正常结束	脚本退出码	失败路径返回非零退出码
自动产测不触发	是否插拔 USBToolBox 本体、是否仍在当前页面	单台设备、手动验证后重新启用
缓存不可用	空间或本机存储错误	先逐个保存重要脚本，不直接清理数据
MCP 无法接入	地址、端口、密钥、接口范围、客户端位置	用本机客户端重新复制配置并检查自检结果
F1 无法打开	是否使用包含本指南的新安装包、默认浏览器	重启新版软件；笔记本尝试 Fn + F1

提交问题时记录软件版本、操作系统、所用接口、接线照片、通信参数、操作步骤及错误日志。截取必要范围，并遮挡 API Key、MCP 密钥和账号敏感信息。

16. 保存工位验收记录

首次在新电脑、转接器或外设上使用时，按下表逐项记录。示例截图不能代替本机验收。

检查项	合格判据
软件与驱动	软件正常启动；设备和所需串口可枚举
授权与目标选择	选择的设备与实际接线一致，硬件操作可用
引脚与供电	信号地、电平、供电和接口方向已核对
首次通信	收到符合外设协议的响应，数据意义可解释
写入或输出	在批准区域测试，并有读回或物理测量结果
停止与断连	能停止循环/脚本，断连后不会继续向旧目标执行
产测判定	合格/不合格有明确阈值与退出码，结果已保存
帮助	未登录、未接设备时，F1 能打开本地 PDF

资料入口

更详细的 Python 接口、参数和示例在软件“Python 产测工具 → API 手册”中。硬件的电气限制、模块初始化序列及寄存器定义以对应外设资料为准。

软件入口：[百问网官网](#)；协议地址定义：[Modbus 官方规范](#)；Mac 扩展批准：[Apple 支持说明](#)。