

# 快速入门指引

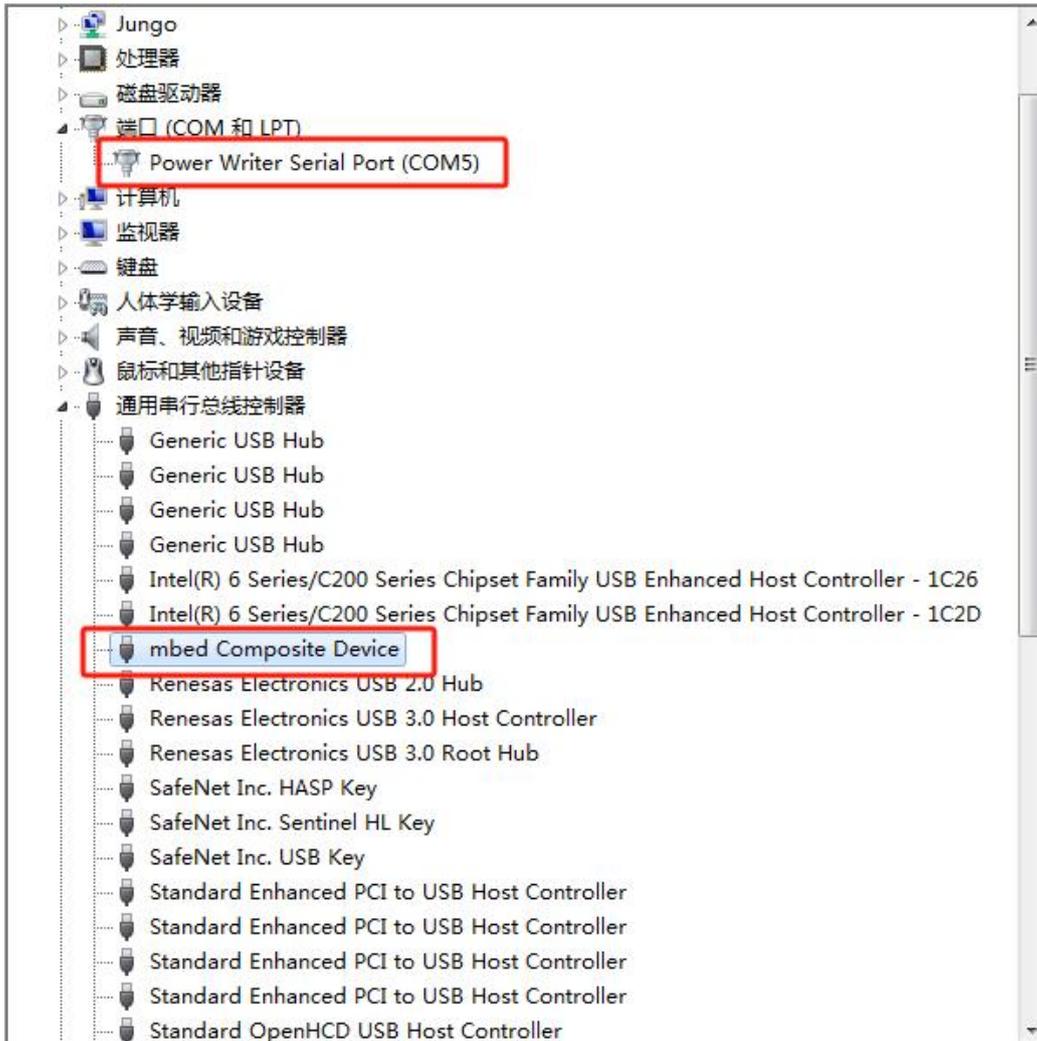
1. 软件和驱动安装.....	2
2. 硬件介绍及信号接口.....	4
3. 软件快速操作流程.....	6
4. 附加软件详解.....	8

## 软件和驱动安装

1. 准备好 Power Writer 系列通用客户端软件，可以从联系客服索要或者从下列地址下载。

<http://www.sz1020.com/article-detail/B3PY1dnN>

**2.Win8/ Win10及以上系统：** 连接 PowerWriter 之后，将会自动安装驱动，用户无需额外安装驱动。



**3. 系统版本低于 Win8的驱动安装方式：** 如果系统版本低于 **Win8**，系统没有兼容的驱动可以直接使用，这个时候启动 PowerWriter软件，客户端软件将自动检测系统是否存在驱动程序。

```
06/25-11:18:32:699> Writer Info: hwVer:1.4 bVer:1.00.06 ifVer:1.01.18  
SN:ECE0CE1B1E4EEEC8735CE4B6C8AEFA8E8 Target:PW200  
06/25-11:18:32:720> [07D4] 当前设备固件类型: A  
06/25-11:18:32:736> Power Writer® 已连接...  
06/25-11:18:32:752> 版本切换:PW200  
06/25-11:18:32:849> 更新烧录器设置完成...  
06/25-11:18:33:030> 更新芯片信息成功...
```

如果系统没有驱动程序，软件将提示如下：



安装完成后，则会提示驱动安装成功，此时如果有连接 PowerWriter 硬件，客户端软件将 会自动连接上 PowerWriter ；

当USB都连接好之后，启动 Power Writer客户端，进入正常的操作流程，见下图所示



## 硬件介绍及信号接口

MP300硬件接口分为： 1.SWIM/SWD烧录接口，2. 烧录机台控制。  
见下图所示：



引脚分类	引脚名称	引脚功能
电源	+5V	电源输出
	GND	电源接地线
SWD 接口	SWDIO	SWD 接口的数据线 SWDIO
	SWCLK	SWD 接口的时钟线 SWCLK
SWIM 接口	RST	SWD 接口的硬件复位线 RESET
	SWIM	SWIM 接口数据线
	VDD (注)	芯片工作电压/ISP外部输入电压
机台控制 接口	KEY	控制烧录(低脉冲启动)
	OK	烧录成功状态输出(输出高)
	NG	烧录失败状态输出(输出高)
	BUSY	烧录忙碌状态输出(输出高)
	TX、RX	通用串口(暂不可用)
引导控制	BOOT	可控制目标芯片的引导方式
其他	LED	烧录状态显示指示灯
	NC	备用

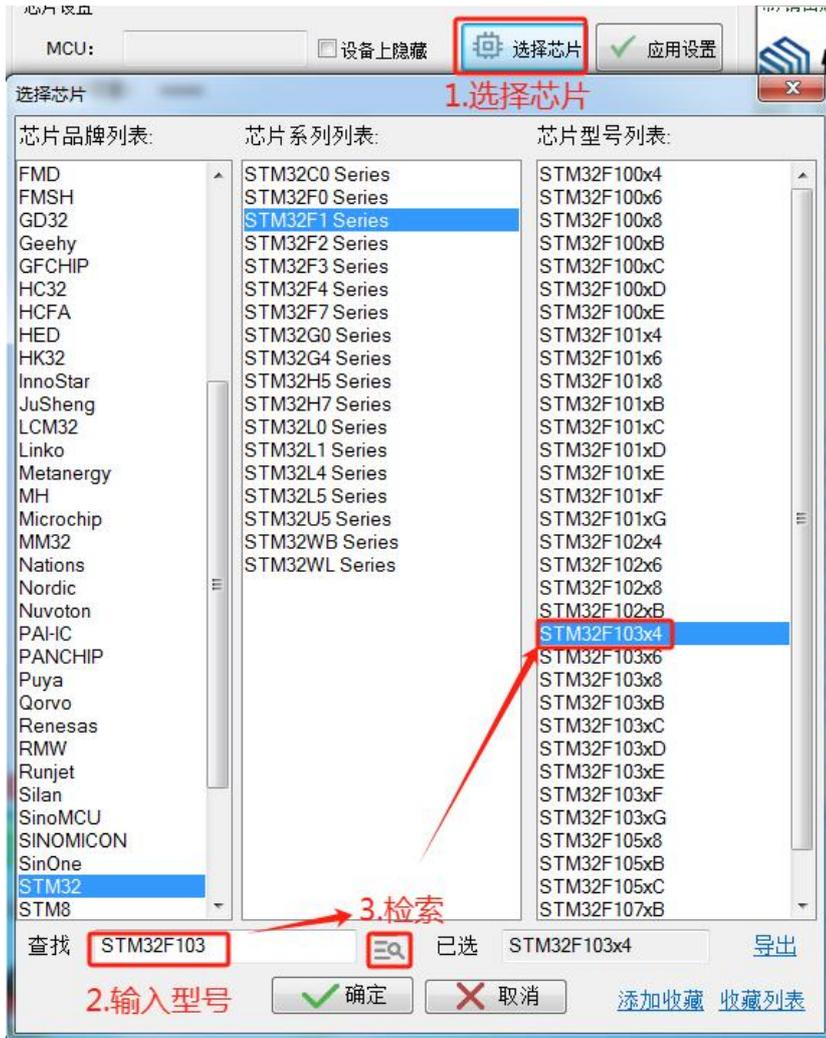
注：VDD 的电压即烧录座中芯片的参考电压。可以通过配置软件设置输出 1.8V、3.3V、5V 三种电压；当使用 ISP 在板烧录时，也可以配置为外部参考电压，参考外部的芯片的工作电压来切换 MF300 编程器的逻辑电平。

## Power Writer 信号定义汇总:

信号名称	信号描述
NG	当 <b>离线</b> 操作: 当读取、擦除、编程等操作失败时, 显示屏NG后面显示“*”计数加1, 直到新的操作到来*号消失
OK	当 <b>离线</b> 操作: 当读取、擦除、编程等操作成功时, 显示屏NG后面显示“*”计数加1, 直到新的操作到来*号消失
蜂鸣器频率定义	PWM 频率为 2.7K Hz (定义如下)
蜂鸣器次数定义	上电无操作时滴一声
	当连接上目标芯片时滴一声
	当烧录成功或者下载离线档案成功时滴两声
	当操作失败时滴三声
	当离线烧录次数为0 时, 滴四声
“开始” 按键	当进行离线烧录时有效(松开触发, 长于 1S 忽略长按)
OK信号	当 <b>离线</b> 操作成功时输出高, 有新操作时清0
NG信号	当 <b>离线</b> 操作失败时输出高, 有新操作时清0
BUSY 信号	当 <b>离线</b> 操作忙碌时输出高, 有新操作时清0
KEY信号	输入 $\geq 40\text{ms}$ && $< 1000\text{ms}$ 的低信号, 启动一次 <b>离线</b> 烧录

## 软件操作流程

步骤一.选择待烧录芯片: 1.选择芯片—>2.输入型号—>3.检索—>4.确认芯片规格



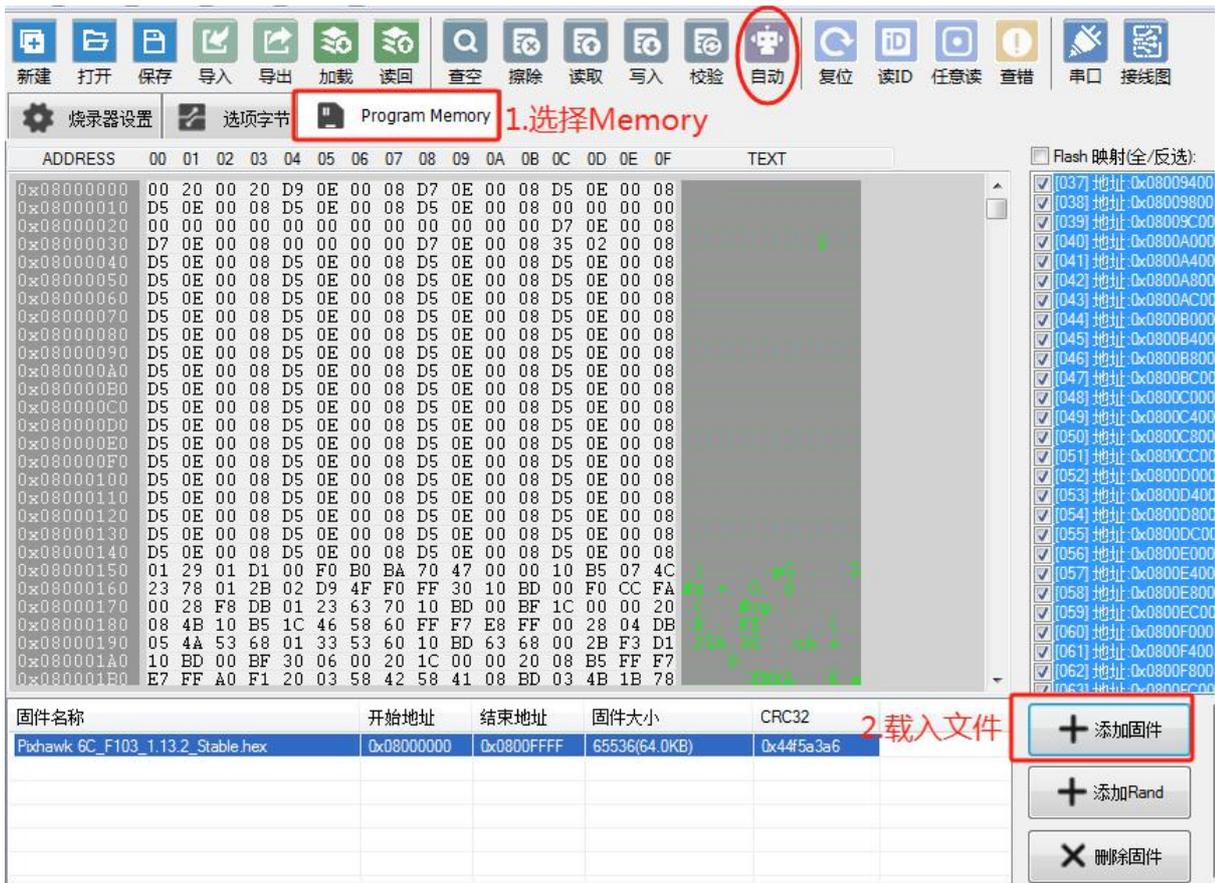
步骤二.更改烧录配置:

擦除方式、芯片工作电压（默认选择3.3V）、编程速度、选项字模式等相关参数，最后并“应用设置”。



步骤三.加载烧录文件并实现联机烧录:

1.选择Program Memory—>2.载入烧录文件—>3.点击“自动”可实现一键烧录



步骤四.通过“加载”工程到编程器，实现脱机烧录：

1.点击“加载” → 2.选择本地路径，并命名文件 → 3.确定



## Power Writer 应用软件详解

本节将详细介绍 Power Writer 应用软件的详细使用方法，用户可以作为参考手册随时来查阅其中的功能，遇到问题时也可以作为工具书来查阅，Power Writer 应用软件的启动界面如下图 3.1.1 所示。



图 3.1.1 Power Writer 应用软件主界面

### “文件”菜单

文件菜单包含保存项目、项目另存为、加载项目、退出四个常用功能



图 3.1.1.2.1 Power Writer 文件菜单

## 保存项目

当用户完成项目设置之后，可以将整个项目打包成一个加密的项目文件

■ 用户第一次点击文件菜单将会弹出首次保存的路径和设置用户密码，见图

3.1.1.2.2 所示：

❖ 密码设置：用户设置项目文件的密码，要求用户输入 16 个字符的密码,注意 密码不能输入太短，否则会提示 **密码设置错误**，如果不想将密码设置成 16 位，可以用一些固定字符代替。

❖ 小眼睛图标 ：默认为不显示密码明文，如果需要查看明文密码，用鼠标 点击此图标。

❖ 文件路径：密码设置完成之后，需要选择保存文件的路径，点击

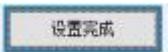
**文件路径** 路径按钮，在弹出的对话框中选择保存的路径，并设置好保存文 件名称，然后点击 设置完成按钮 ，



图 3.1.1.2.2 保存项目文件设置

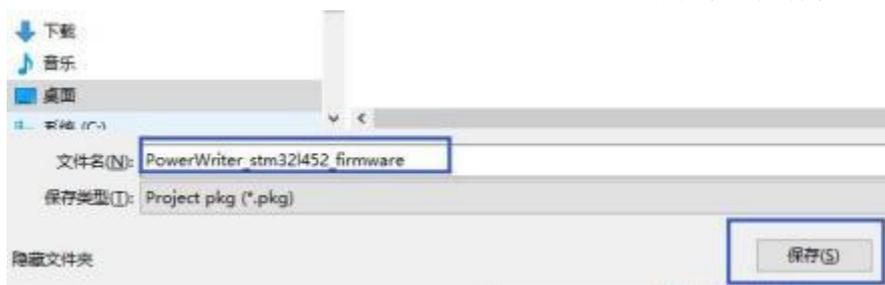


图 3.1.1.2.3 保存文件对话框

保存项目会在状态栏显示项目文件的路径,见图 3.1.1.2.4 所示,同时日志栏 将显示保存的结果33126-11:53:51:7463保存成，如果保存失败将显示失败的提 示信息。

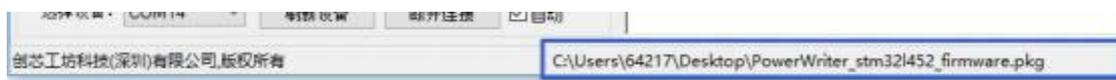


图 3.1.1.2.4 保存项目之后会在状态栏显示保存项目的路径

附加说明:

- 1: 创芯工坊加密文档格式为自研算法,同时具有高强度和良好的性能
- 2: pkg 项目文档内部不保存任何密码的信息,采用内部自验证机制,如果用户忘记了密码, pkg 项目文件将无法解包,创芯工坊官方也只能通过枚举的方法来尝试还原项目文件,所以请用户牢记自己设定的密码。
- 3: 为何设置为强制 16 位长度,经过我们的 Benchmark 测试加密算法性能,过短的密码在 Hacker 通过极限手段获取到了加密算法原理,可采用暴力测试可能会获取到真实数据,而设置成 16 位暴力破解的时间成本将会非常的高,更多安全特性请参考安全特性章节

- 第二次保存将不会再弹出保存文件对话框,直接保存项目到上次设置的信息

### 3.1.1.2.2 项目另存为

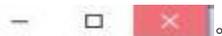
项目另存为的定义和保存项目的定义区别在于:另存为每次都会弹出新的保存项目信息,而不是保存为上次设置的路径和密码,同其他软件定义的另存为功能上完全一致。

### 3.1.1.2.3 加载项目

通过加载项目功能,可以加载之前保存的项目文件到 Power Writer 软件中,操作方式和保存项目完全一致:

- 填写项目的密码,密码错误将无法加载。
- 设置加载项目的路径

### 3.1.1.2.4 退出

退出 Power Writer 软件,功能同系统标题栏的 。

注:系统在操作过程中将执行退出

### 3.1.2.2“执行”菜单

执行菜单包含芯片在线操作的常用功能,见图 3.1.3.1 所示



图 3.1.3.1 执行菜单功能

### 3.1.2.2.1 保存并离线加载

当用户将项目的所有设置完成之后，可以将项目配置到 Power Writer 硬件，用于实际的产品生成，此功能会同时执行 保存项目 并加载项目到 Power Writer 硬件，如果操作成功将会提示

03/26-14:21:11:999> 加载离线数据成功

如图 3.1.3.1 所示：



图 3.1.3.1 保存并离线加载演示

### 3.1.2.2.2 离线读取并保存

如果用户需要将 Power Writer 中已存储的 Project 文件读取回来，可以执行离线读取并保存，此动作将会将项目文件从 Power Writer 读取回，并尝试加载为当前项目，需要用户填写 Project 的密码，如图 3.1.3.2 所示：

注：如果用户密码错误，将无法加载到当前项目，提示



图 3.1.3.3 离线读取并保存设置

### 3.1.2.2.3 读取 Program Memory

用户通过读取 Program Memory 功能读取芯片的 Program Memory 数据到 Power Writer 软件，设置如下图 3.1.3.3.1 所示，读取完成后，数据将会在 Program Memory 选项卡显示，如图 3.1.3.3.2 所示，如果读取失败，则会显示错误信息

- 读取地址：用户可以自由设置读取地址，默认设置为芯片 Program Memory 的首地址
- 读取大小：用户可以设置为 1KB、2KB、4KB、8KB、16KB、32KB、64KB、128KB、256KB、512KB、1MB、2MB、4MB。等
- 整片读取，如果用户需要读取整片的 Program Memory 数据，直接勾选整片读取，默认不勾选



图 3.1.3.3.1 读取 Program Memory 设置

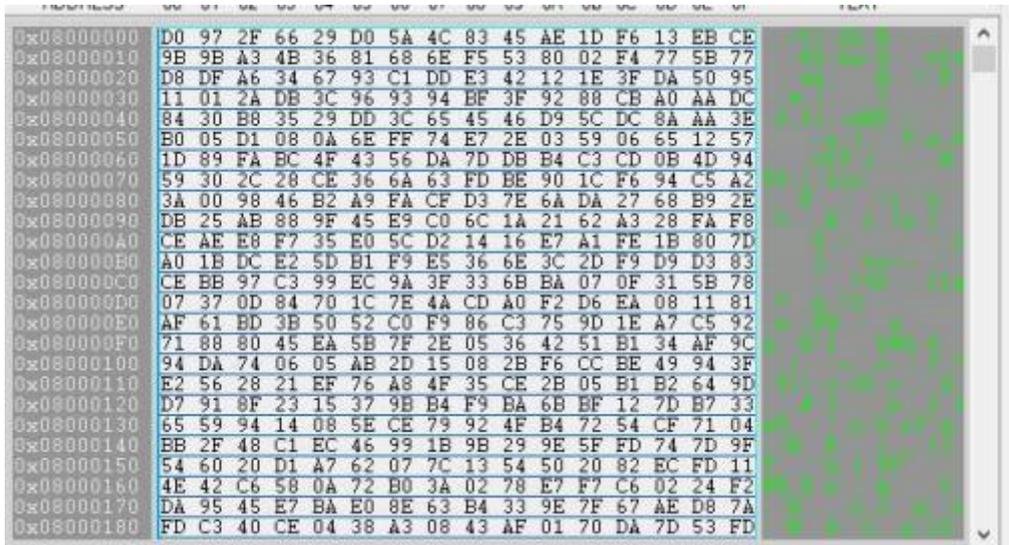


图 3.1.3.3.2 Program Memory 实时显示读取到的数据并高亮

注：

- 1：主流芯片有读保护，则无法读取数据。
- 2：如果设置参数超过了芯片的容量，则会提示读取失败，后段数据无法读取，一般直接勾选整片读取即可
- 3：极少芯片有读保护的情况下，也能读取数据，但是数据是异常的,最好是先读取选项字节，先验证下数据的有效性,大部分芯片都会主动判断有没有读保护

### 3.1.2.2.4 查空 Program Memory

用户通过此功能，可以快速检查芯片是不是空片，操作演示如下，测试芯片为 F071CB，测试之前写入了数据，执行查空操作提示如图 3.1.3.4.1 所示：

[p226-14:44.47026]Not Blank Addr:08000000,size:256

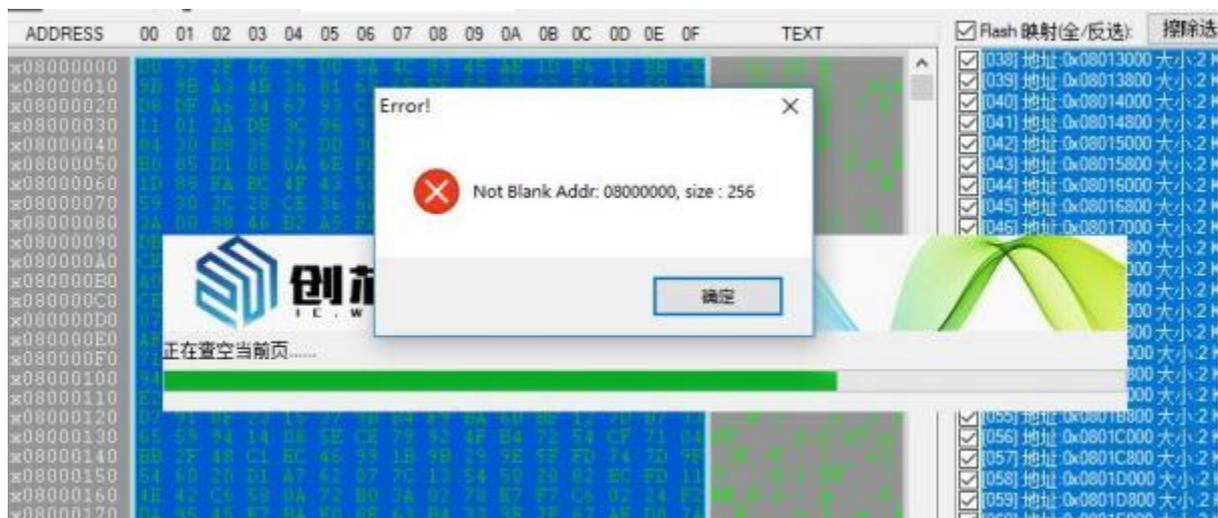


图 3.1.3.4.1 查询芯片是不是空片提示信息

如果芯片为空片，则会在日志栏显示,查空成功,如图 3.1.3.4.2 所示，此演示测试步骤：

- ✓ 擦除目标芯片，提示擦除成功
- ✓ 执行查空操作，提示查空成功



图 3.1.3.4.2 查询芯片是不是空片成功提示

### 3.1.2.2.5 擦除 Program Memory

用户通过 Program Memory 功能可以快速擦除芯片，此功能执行的是整片擦除，如果需要 Sector 擦除，请参考 Program Memory 的扇区分页位置的扇区擦除功能，擦除成功示范如图 3.1.3.5.1 所示：



图 3.1.3.5.1 擦除芯片成功演示

当芯片有读/写保护时，执行擦除操作会失败，如图 3.1.3.5.2 所示

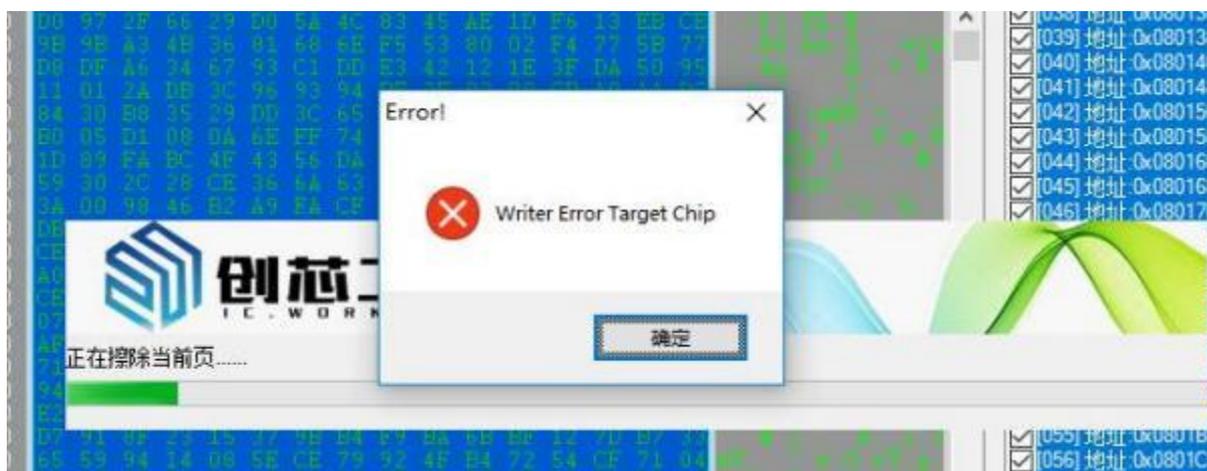


图 3.1.3.5.2 擦除芯片失败

注，如果擦除失败，请按下面步骤进行排查：

- ◆ 执行任意一项操作，或者检查 Power Writer 的状态指示灯是不是长亮的，如果是常亮，说明芯片时在线的，如果是灭的说明线路连接有问题
- ◆ 执行读取选项字节，检查选项字节中的保护设置是否为 0 级，如果不是，则需要修改成 0 级，然后写入。
- ◆ 检查扇区写保护，如果写保护是开启的，去掉，更新选项字节到目标芯片。
- ◆ 当选项字节更新之后，再执行擦除即可正常擦除芯片

如果以上步骤都不能擦除芯片，请联系我们的技术支持团队（cs@icworkshop.com）。

### 3.1.2.2.6 编程 Program Memory

开发人员可以通过 Power Writer 在线对目标芯片进行烧写，烧写时，有以下两点特性需要注意

- 当添加了分段时：执行 Program Memory 操作，写入的是所有的分段固件。分段数量无限限制，如图 3.1.3.6.1 所示：

固件名称	开始地址	结束地址	固件大小	CRC32
STM32F071xB.bin	0x08000000	0x0801FFFF	131072(128.0KB)	0x74e605cd

图 3.1.3.6.1 当使用分段烧录功能时烧录的是分段固件

■ 当没有添加分段时：执行 Program Memory 操作，写入的是 Program Memory 所有数据。如图 3.1.3.6.2 所示：

固件名称	开始地址	结束地址	固件大小	CRC32

图 3.1.3.6.2 当无分段烧录功能时烧录的是 Program Memory 所有数据

执行 Program Memory 操作成功时，将看到图 3.1.3.6.3 所示的提示信息



图 3.1.3.6.3 执行 Program Memory 操作成功提示

执行 Program Memory 操作失败时，将看到图 3.1.3.6.4 所示的提示信息，并提示写入失败的地址空间。

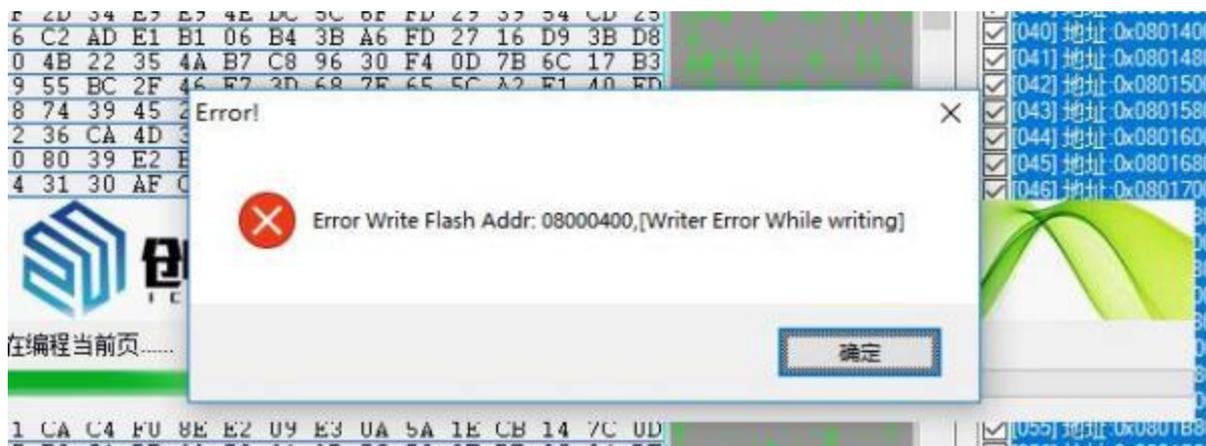


图 3.1.3.6.4 执行 Program Memory 操作失败提示

注：写入失败原因汇总

- Program 区域为非空，处理方法为先擦除芯片，或者擦除制定的扇区
- 芯片保护级别不为 0，处理方法为将芯片保护级别去除，具体选项字节设置
- 芯片的扇区有写保护，处理方法为将扇区写保护去除

### 3.1.2.2.7 校验 Program Memory

用户可以使用校验功能，对比芯片中的数据 and Program Memory 区域的数据是否一致，需要留意的是，校验功能同样是根据是否分段来做自动处理，如果添加了分段固件，校验分段固件本身的数据，如果没有添加分段固件，校验的是整个 Program Memory 数据区域。校验成功时，将会看到如图 3.1.3.7.1 的所示信息。



图 3.1.3.7.1 校验成功时提示信息

当校验失败时，将看到如图 3.1.3.7.2 所示信息，从错误信息中可以看到校验的块的首地址，当前校验的数据长度，默认为每次校验 256 个字节。



图 3.1.3.7.2 校验失败时提示信息

### 3.1.2.2.8 Program Memory 自动编程

Program Memory 自动编程功能是 擦除、写入、校验的组合功能，执行此功能时，将会自动执行 Flash 擦除、写入、校验。操作成功时如图 3.1.3.8.1 所示，如果任何一步失败都将提示对应的是 失败信息。



图 3.1.3.8.1 自动编程成功时提示信息

### 3.1.2.2.9 全功能自动编程

全功能自动编程的作用是将用户所有设定的数据，设置项，以在线的方式一次性写入，包括 SN，和 Matrix 绑定数据(授权方式选择 PowerWriter 内置的前提下)，并会同步更新 SN 的起始序号。相当于用在线的方式执行了一次离线烧录(不含 ICWKEY 的方式)，如图 3.1.2.2.9-1 所示



图 3.1.2.2.9-1 智能自动编程成功时提示信息

### 3.1.2.2.10 其他数据区操作

Power Writer 除了支持 Main Flash 的在下读写之外，同时也支持 EEPROM、OTP、DATA 等数据区的操作，如果目标芯片有这些数据区，PowerWriter 将自动将这些操作命令动态添加到菜单中。如果当前选择的芯片没有其他数据区，则此菜单将变灰色，为不可用的状态。

#### 3.1.2.2.10.1 查空EEPROM(OTP、Data 等)

功能和 Program Memory Flash (主程序区) 操作方法一致，具体参考 Program Memory Flash 的操作方法。

#### 3.1.2.2.10.2 擦除EEPROM(OTP、Data 等)

功能和 Program Memory Flash (主程序区) 操作方法一致, 具体参考 Program Memory Flash 的操作方法。

### 3.1.2.2.10.3 写入 EEPROM(OTP、Data 等)

功能和 Program Memory Flash (主程序区) 操作方法一致, 具体参考 Program Memory Flash 的操作方法。

### 3.1.2.2.10.4 读取EEPROM(OTP、Data 等)

功能和 Program Memory Flash (主程序区) 操作方法一致, 具体参考 Program Memory Flash 的操作方法。

### 3.1.2.2.10.5 校验 EEPROM(OTP、Data 等)

功能和 Program Memory Flash (主程序区) 操作方法一致, 具体参考 Program Memory Flash 的操作方法。

### 3.1.2.2.11 复位目标芯片

复位目标芯片功能, 可以执行一次目标芯片的复位动作。此操作将同时触发:

- 执行一次软复位动作。
- RST 引脚输出一次外部复位信号

操作成功将得到成功提示信息, 如图 3.1.3.10.1 所示:



图 3.1.3.10.1 复位成功提示信息

### 3.1.2.2.12 读取选项字节

用户可以使用读取选项字节功能, 读取目标芯片的所有选项字节, 如图: 3.1.3.11.1 所示:



图：3.1.3.11.1 读取选项字节成功提示

如果用户选择的芯片和目标芯片不匹配，则无法读取选项字节，提示信息如下图 3.1.3.11.2 所示：



图：3.1.3.11.2 读取选项字节失败提示

### 3.1.2.2.13 写入选项字节

当用户需要更新芯片的选项字节时，可以通过此功能，写入自定义的选项字节，操作成功如下图 3.1.3.12.1 所示，选项字节跟官方发布的数据手册保持一致。

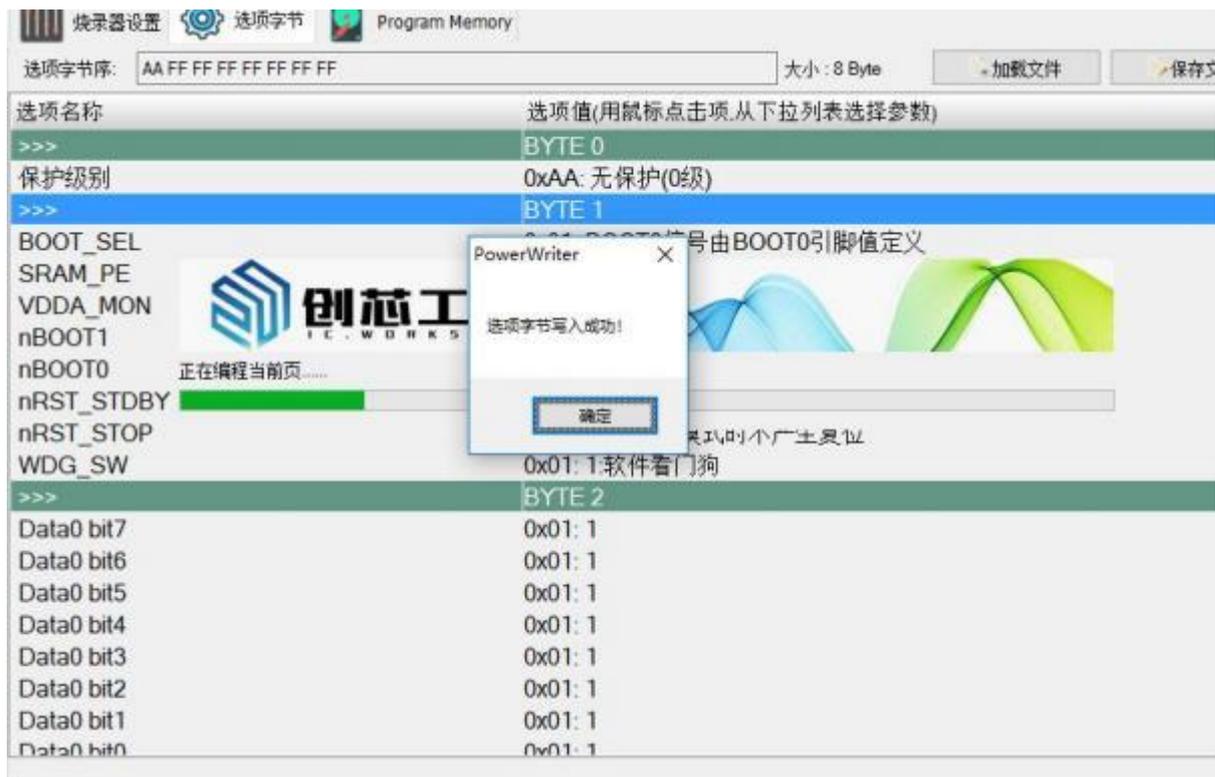


图 3.1.3.12.1 选项字节写入成功

注：选项字节写入后，会立即生效，无须用户将芯片断电，Power Writer 内部已经做了自动处理。

### 3.1.2.2.14 读取CID

读取 CID 功能可以读取芯片的 CID，连接上芯片之后直接读取即可在日志栏查看芯片的 CID，如图 3.1.3.13.1 所示



图 3.1.3.13.1 读取芯片 CID

注：部分芯片的 CID 是非连续的，Power Writer 显示的 CID 是将不连续的 CID 连续显示，在 Power Writer 内部自动做了拼接处理。

### 3.1.2.2.15 任意地址读数据

任意地址读数据是一个非常强大且实用的功能，开发者可以指定任意的芯片内部存储空间地址，包括但不限于：

- 读取芯片 RAM 数据并显示
- 读取芯片 Flash 数据并显示

■ 读取芯片所有寄存器数据并显示

如：读取芯片 RAM 数据，从地址 0x20000000 读取 1K 的 RAM 数据并显示，结果如下：

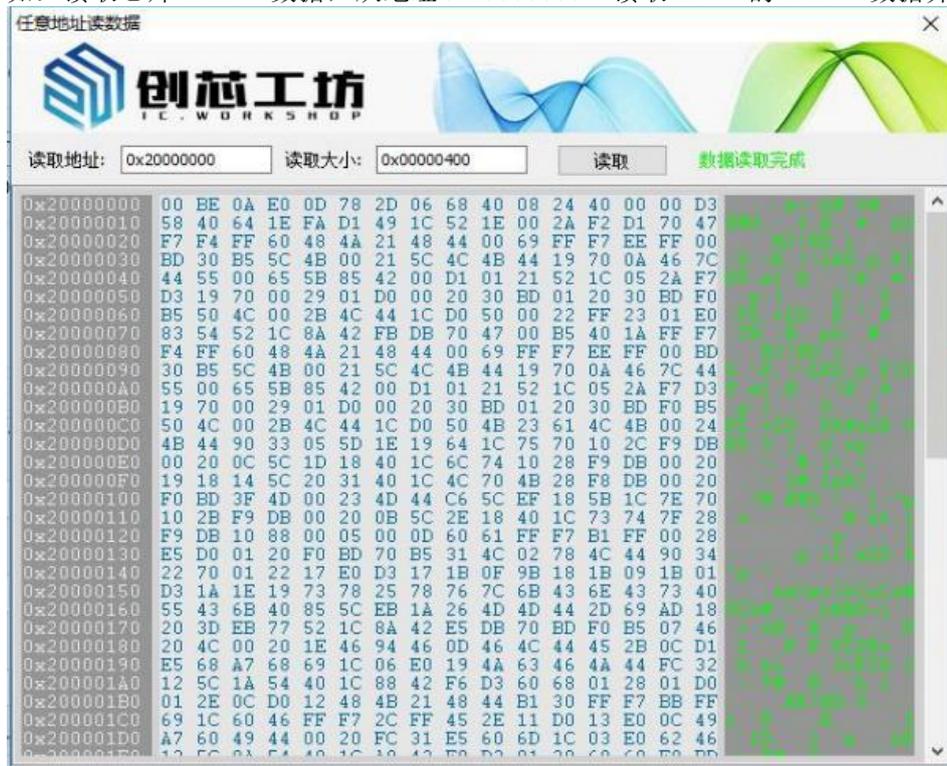


图 3.1.3.14 读取 RAM 数据示范

也可以读取任意的 Flash 地址数据，如读取 0x08000000 1K 大小的数据，结果显示为图 3.1.3.14.2 所示：



图 3.1.3.14.2

读取 Flash 数据示范， 也可以读取芯片内部的 CODE 数， 结果显示为图

### 3.1.3.14.3

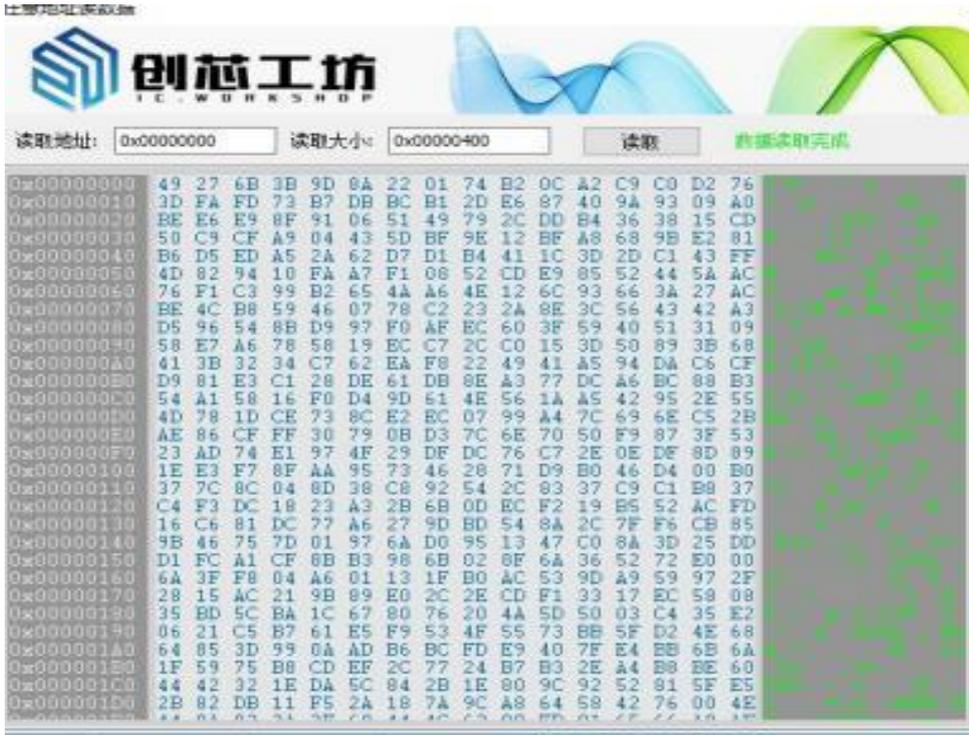


图 3.1.3.14.3 读取芯片内部 CODE 空间数据示范

注：任意地址读数据依赖于你对当前使用芯片的熟练程度，读取芯片内部的数据时，需要设定好需要读取的虚拟地址，以及读取的大小；

读数据可能因为权限问题，或者是由于跨模块时读取数据会导致失败；此功能跟 ST Programmer 的任意地址读数据功能一致。