

伟福 JM3 编程器使用插件功能

插件功能可以利用伟福提供的函数，函数中用户自己写程序参与编程操作过程，让用户在编程操作的每一步（连接、擦除、检空、烧写、校验、加密等步骤）开始前进行一些准备工作或操作结束时进行善后工作。例如让你在编程前根据每个芯片的唯一码（UCID）生成唯一的程序内容，也可以在用编程器对编程后的芯片进行简单测试。插件文件为特殊需要用户提供。我们以 UCID 为例，介绍插件的用法

我们以编程 STM32F1 系列 CPU 为例，利用 CPU 的 UCID（芯片唯一码）改变程序区内某块内容，这样的程序便是唯一不可复制（防破解），以达到加密作用。

在 Keil 中打开 “../Plugin/STM32F1/ STM32F1_UCID.Uv2” 样例项目，（如果用户自建新项目请严格参照样例项目的设置）

一、编写插件程序，插件程序结构如下：

```
01 // 编程 UCID
02 // 所有使用全局变量的函数都要包含此文件
03 #include "../INC/JM3_DEF.H"
04 #include "../INC/JM3_VAR.H"
05
06 struct JMType const vPlugin = {
07     0xFFFF0001, // Plugin ID
08     "STM32F1xx UCID Plugin",
09 };
10
11 // pV: 全局变量指针 pL: 局部变量指针 vPos: 数据区指针 pOption: 编程选项
12 u32 Run(u32 * pV, u32 * pL, u32 vPos, u32 vOption)
13 {
14     if ( vOption == bPGNull ) { // 编程开始时编程程序会调用 Run(vV, pL, bBEFORE, bPGNull)
15         eVP = 1; // 允许调用 Memory 函数
16
17         CVPB = 0x08002F00; // Memory 起始地址
18         CVPE = 0x08002F17; // Memory 结束地址
19     }
20     return(0);
21 }
22
23 // 当编程程序需要 AdrB..AdrE 数据时会调用 Memory
24 // pV: 全局变量指针 pL: 局部变量指针 pBuffer: 数据区指针 AdrB: 开始地址 AdrE: 结束地址
25 u32 Memory(u32 * pV, u32 * pL, u8 * pBuffer, u32 AdrB, u32 AdrE)
26 {
27     u32 i;
28     u8 * pR;
29
30     // 编程程序在连接芯片时会把 12 个字节的 UCID 放在 pUCID 指针指向的12个单元
31     // 先放 12 个字节的 UCID
32     pR = pUCID;
33     for (i=0; i<12; i++) *pBuffer++ = *pR++;
34
35     // 再放 12 个字节的 XOR UCID
36     pR = pUCID;
37     for (i=0; i<12; i++) *pBuffer++ ^= *pR++;
38
39     return(0);
40 }
41 }
```

本插件的 ID 及文字描述,方便使用时识别, ID 前四位固定为 0xFFFF, 表示用户的插件程序

如果选择了插件功能, 则在每步操作之前, 后都会调用 RUN 函数, 调用后由这两个参数决定执行什么功能, vPos 定义编程操作前后, vOption 定义是什么编程操作 (擦除、烧写、校验等)

必须传递这两个参数

eVP 为 1 时表示允许调用下面的 Memory 函数对编程数据进行改写, 被改写数据的地址由 CVPB, CVPE 决定。每个系列 CPU 的程序空间不尽相同, 此处为 STM32F1 系列 CPU 空间。

用芯片唯一码 UCID 对程序空间进行改写

二、对项目进行编译，生成 AXF 文件

```
Build target 'STM32F1_UCID'
compiling STM32F1_UCID.c...
linking...
STM32F1_UCID.axf: Warning: L6305W: Image does not have an entry point. (Not specified or not set due to multiple choices.)
Program Size: Code=100 RO-data=132 RW-data=0 ZI-data=736
"STM32F1_UCID.axf" - 0 Error(s), 1 Warning(s).
```

三、伟福编程界面，读入 AXF 转换生成插件。

1、在编程主界面，打勾选中“自动编号”功能，点击右侧“设定”键，打开自动编号设置窗口。

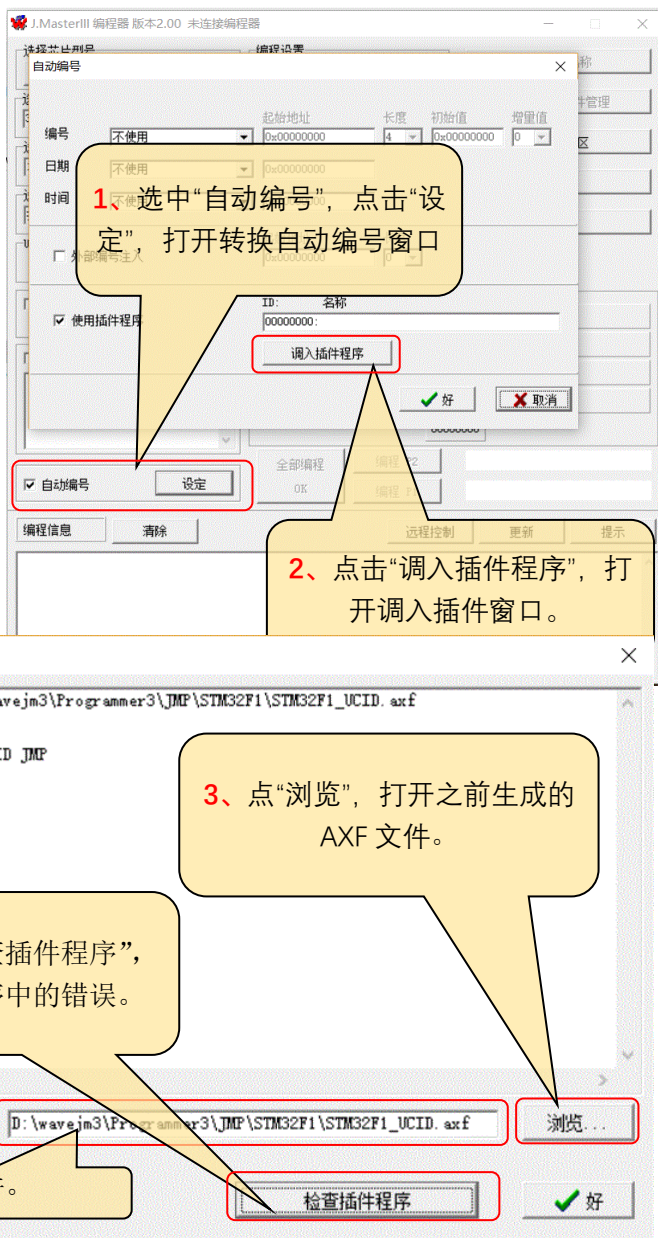
2、在自动编号窗口，点击“调入插件程序”键，打开插件窗口。

3、在新打开的插件窗口中点击“调入 AXF 文件”栏右侧“浏览”键，找到之前生成的 AXF 文件。

4、点窗口下方的“检查插件程序”，以发现程序中的错误。

5、在插件窗口，按“好”关闭插件窗口。

6、在自动编号窗口，按“好”关闭自动编号窗口。



这样设置完成后，在编程时，在连接上芯片后，正式编程前插件会根据程序要求改写烧录内容。保存生成伟福的 JMO 编程文件时，插件程序会一起打包保存。

本例以 UCID 为例说明插件的使用方法，插件程序还可以与被编程的程序配合，在编程完成后，对用户板 CPU 管脚进行 IO 测试，UART 通信测试，SWD 测试等。我们已经提供相应的样例程序，用户可以加以修改以适合自己的需要。

注意：插件文件中对于非法地址的访问和无效的操作是被禁止的。在检查插件程序时如果发现这些错误，我们会指出。