

固件升级工具使用说明书

杰理科技

2022 年 7 月 11 日

目录

第 1 章 工具介绍	3
第 2 章 配置项说明	4
1.1 合并固件文件到工具	5
1.2 PC 模式设备名字配置	6
1.3 PID 显示和匹配规则配置	8
1.4 升级选项配置	11
1.5 窗口标题栏文字和图标配置	13
1.6 固件文件选择栏文字和文件类型	14
1.7 自定义版本信息配置	15
1.8 固件信息显示配置	16
1.9 样机信息显示配置	16
1.10 自定描述信息配置	19
1.11 升级和校验按钮文字配置	20
1.12 认证码配置	21

第 1 章 工具介绍

本工具用于产品研发初期，开发者已经将样机送至第三方后，需要更新样机固件的情况。

例如开发者将样机送至第三方后，由于软件功能变更等等其他因素导致需要更新样机固件的情况。

为方便开发者与第三方确认固件程序。本工具提供以下功能：

- a. 升级样机固件
- b. 校验样机固件
- c. 显示样机中固件文件 (FW/UFW) 的校验码
- d. 显示样机中固件是由哪个版本烧写工具烧写的
- e. 将升级的固件文件和工具打包
- f. 工具界面上显示指定的文字信息

以上功能可在配置文件 config.ini 中配置是否使能。config.ini 文件相对 exe 的路径为 data/config.ini。更多细节，请查看本文第二章。

注：由于 PC 升级环境无法对样机进行测试，所以禁止使用此工具进行量产。

第 2 章 配置项说明

配置文件必须用 UTF8 编码保存。

配置文件 config.ini 必须存放在工具同级目录的 data 目录下。需要合并到工具中的文件也必须存放到 data 目录下。如下图：



图 2-1

本工具界面上所有文字都可以通过配置项进行配置。包括中/英语言文字。中/英文文字通过 *text_cn 和 *text_en 配置项配置。

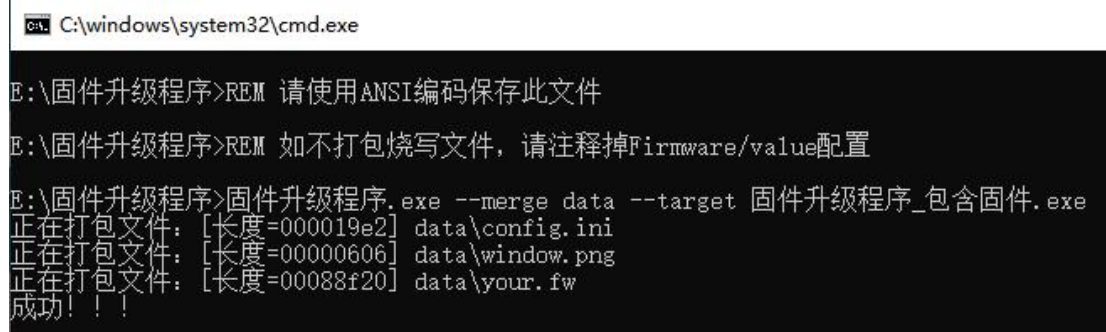
1.1 合并固件文件到工具

此功能详细配置项如下：

```
#####
#
# 需要合并到工具的固件文件
# 升级用的固件文件
# 在固件文件和工具合并后有效
#
#####
[Firmware]
value = your.fw
```

此配置项需要把固件文件和 data/config.ini 文件一起合并到工具中才会生效。合并固件文件时需要将固件文件也放到 data 目录下,并且设置 Firmware/value 的值为合并的固件文件名。参考图 2-1:

文件准备好后,通过命令行执行“固件升级程序.exe --merge data --target 固件升级程序_包含固件.exe”就可以将 data 目录合并到工具中,如下图:



```
C:\windows\system32\cmd.exe

E:\固件升级程序>REM 请使用ANSI编码保存此文件
E:\固件升级程序>REM 如不打包烧写文件,请注释掉Firmware/value配置
E:\固件升级程序>固件升级程序.exe --merge data --target 固件升级程序_包含固件.exe
正在打包文件: [长度=000019e2] data\config.ini
正在打包文件: [长度=00000606] data>window.png
正在打包文件: [长度=00088f20] data\your.fw
成功!!!
```

“固件升级程序_包含固件.exe”即是合并后包含了固件文件的新工具。

执行合并后的工具,执行“固件升级程序_包含固件.exe”,固件会自动加载到工具中,并且文件选择栏会不可见。如下图:

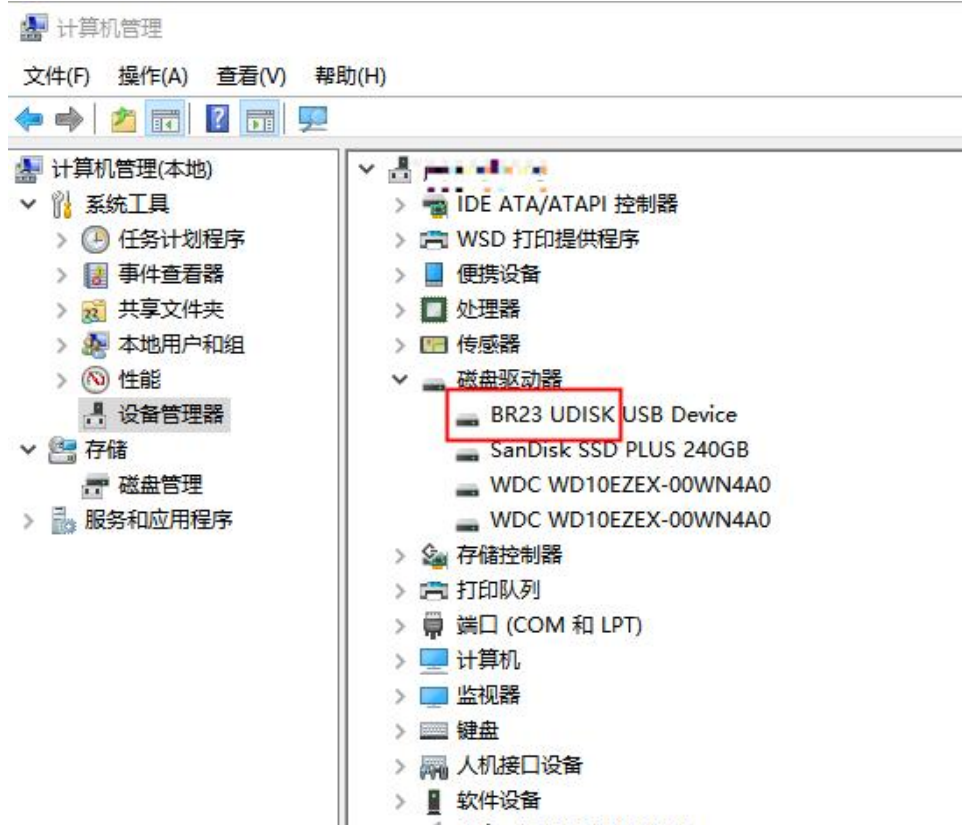


1.2 PC 模式设备名字配置

此功能详细配置项如下：

```
#####
#
#   PC 模式下支持升级样机的名字
#
#####
[PCMode]
name_list = br23udisk           ;支持升级的样机名字列表
wait_time = 1000                ;等待样机从 PC 模式切换到升级模式的时间
```

如果样机不能直接进入升级模式并且有 PC 模式，那么可以通过此配置使样机从 PC 模式切换到升级模式，完成固件升级。以 AC695X 为例，标准 SDK 固件的样机进入 PC 模式后，可以在 WINDOWS 设备管理器中看到设备 BR23 UDISK USB Device。那么 **BR23UDISK** (不包含中间空格) 就是该样机的 PC 模式名字。如下图：



此名字可以通过 SDK 的 apps/common/usb/device/msd.c 文中的 SCSIInquiryData 来配置。如下图：

```

59 #if defined(CONFIG_CPU_BR23)
60 #define CPU_NAME 'B','R','2','3'
61 #elif defined(CONFIG_CPU_BR25)
62 #define CPU_NAME 'B','R','2','5'
63 #elif defined(CONFIG_CPU_BR29)
64 #define CPU_NAME 'B','R','2','9'
65 #elif defined(CONFIG_CPU_BR30)
66 #define CPU_NAME 'B','R','3','0'
67 #elif defined(CONFIG_CPU_BR36)
68 #define CPU_NAME 'B','R','3','6'
69 #else
70 #define CPU_NAME 'U','K','N','W'
71 #error "not define cpu name"
72 #endif
73 // *INDENT-ON*
74 static const u8 SCSIInquiryData[] = {
75     0x00, // Peripheral Device Type: direct access devices 0x05, //
76     0x80, // Removable: UFD is removable
77     0x02, // iso ecma ANSI version
78     0x02, // Response Data Format: compliance with UFI
79     0x20, // Additional Length (Number of UINT8s following this one): 31, totally 36 UINT8s
80     0x00, 0x00, 0x00, // reserved
81     CPU_NAME, //-- Vender information start
82     ' ',
83     ' ',
84     ' ',
85     ' ',
86     // " " -- Vend Information end
87     'U', //-- Production Identification start
88     'D',
89     'I',
90     'S',
91     'K',
92     ' ',
93     ' ',
94     ' ',
95     ' ',
96     ' ',
97     ' ',
98     ' ',
99     ' ',
100     ' ',
101     ' ',
102     ' ',
103     ' ',
104     // " " -- Production Identification end
105     0x31, // "1" -- Production Revision Level start
106     0x2e, // "."
107     0x30, // "0"
108     0x30, // "0" -- Production Revision Level end
109 };
110
111 #define MSD_BLOCK_SIZE 1
112 #define MSD_BUFFER_SIZE (MSD_BLOCK_SIZE * 512)
113
114 struct usb_msd_handle {
115     struct usb_scsi_cbw cbw;
116     struct usb_scsi_csw csw;
117     struct msd_info info;
118     u8 *msd_buf;
119     u8 *ep_out_dmabuffer;
120     /* u8 *ep_in_dmabuffer; */
121 };
122
123 struct usb_msd_handle *msd_handle;
124
125 #if USB_MALLOCE_ENABLE
126 #else
127 #if USB_MSD_BULK_DEV_USE_ASYNC

```

1.3 PID 显示和匹配规则配置

在很多情况下，开发者会有多款样机在第三方处进行确认。为避免不同硬件的样机升错固件程序，可以通过 PID 匹配规则来限制样机的升级。此功能详细配置项如下：


```
#####
#
#   PID 显示和匹配规则配置
#
#####

#
#   PID 匹配
#
[PidMatch]
enable                = true        ; 是否匹配 PID, 如果 PID 不匹配那么不允许烧写
pid_mismatch_text_cn = "设备 PID 与固件 PID 不匹配!"
pid_mismatch_text_en = "The device PID does not match the firmware PID!"

#
#   设备 PID 白名单
#   如果样机的 PID 在白名单中
#   升级的时候会忽略样机 PID
#
[PidWhiteList]
value = AC695x_TWS, None

#
#   Pid 映射表, 将实际 Pid 显示为指定 Pid
#
[PidMapping]
OLD PID    = NEW PID
AC695x_TWS = NewPid

#
#   为兼容旧 SDK 增加的参数
#   PID 解析, AC691X 有效, AC692X SDK V1 版有效
#   将配置中的 "PDCTNAME=" 以第一个分隔符为间隔分为两段
#   第一段显示为 PID, 第二段显示为 VID
#
[PidSection]
sep = _          ; 分割符号, 默认为下划线
```

如启动 PID 匹配功能，当固件 PID 和样机 PID 不匹配时，工具会提示在 PidMatch/pid_mismatch_text_??配置中的设置的文字，如下图：



1.4 升级选项配置

此功能详细配置项如下：

```
#####  
#  
# 选项配置  
#  
#####  
[EraseAll]  
text_cn = 擦除整个 FLASH  
text_en = Reaset Entire Flash  
checked = true          ; 是否默认勾选  
  
[Reset]  
text_cn = 成功后复位  
text_en = Reset After Success  
checked = false        ; 是否默认勾选  
  
[AutoUpgrade]  
text_cn = 自动升级  
text_en = Auto Upgrade  
visible = true          ; 是否显示  
checked = false         ; 是否默认勾选, 此配置仅在“visible=true”时有效  
  
[DualUboot]            ; 双 UBOOT 烧写, 需要 SDK 支持, 如不确认是否支持, 请勿勾选  
text_cn = 双 UBOOT 烧写  
text_en = Dual Uboot Burning  
visible = true  
checked = false  
  
[DualBackupBurning]    ; 双 UBOOT 烧写, 需要 SDK 支持, 如不确认是否支持, 请勿勾选  
text_cn = 双备份烧写  
text_en = Dual Backup Burning  
visible = true  
checked = false
```

以上配置在工具界面上的效果如下图：



1.5 窗口标题栏文字和图标配置

此功能详细配置项如下：

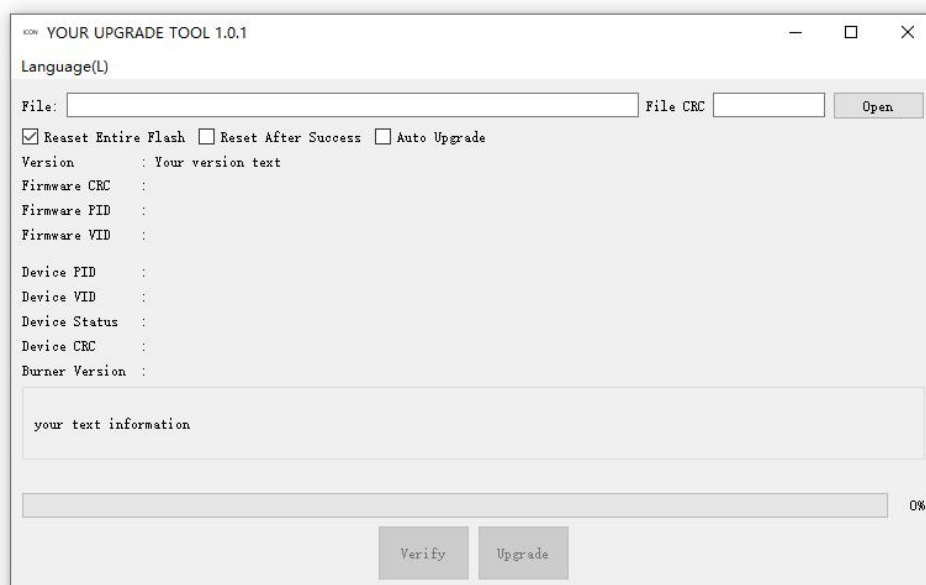
```
#####
#
#   窗口菜单栏信息配置
#
#####

[Title]
text_cn   = 您的升级程序 1.0.1           ;中文标题文字
text_en   = YOUR UPGRADE TOOL 1.0.1      ;英文标题文字
icon_file = window.png                   ;标题栏图标, 支持 png, jpg 格式文件格式

[Menu]
visible = true           ;是否显示顶部菜单栏
language = english       ;默认语言, 仅支持中文和英语 (chinese, english)

[Window]
width    = 750           ;窗口最小宽度
height   = 400           ;窗口最小高度
width_fixed = false      ;是否固定宽度
height_fixed = false     ;是否固定高度
```

以上配置在工具界面上的效果如下图：

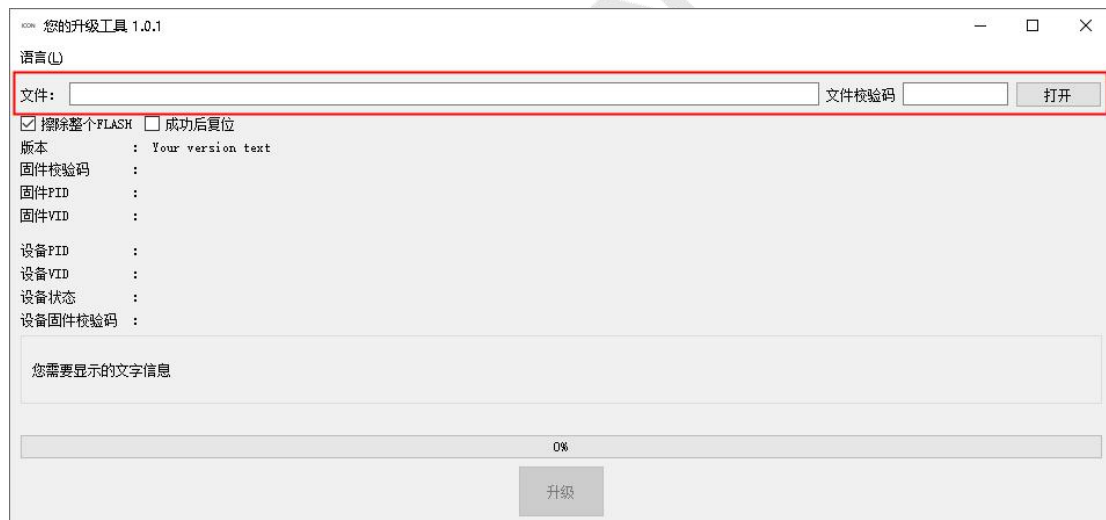


1.6 固件文件选择栏文字和文件类型

工具支持的固件文件类型有 FW 和 UFW 文件，此功能详细配置项如下：

```
#####
#
#   固件文件选择栏配置
#
#####
[FirmwareEdit]
label_text_cn = 文件：
label_text_en = File:
crc_text_cn   = 文件校验码
crc_text_en   = File CRC
open_text_cn  = 打开
open_text_en  = Open
file_type     = fw,u fw    ;支持升级的文件类型，文件类型之间用逗号隔开
```

以上配置在工具界面上的效果如下图：

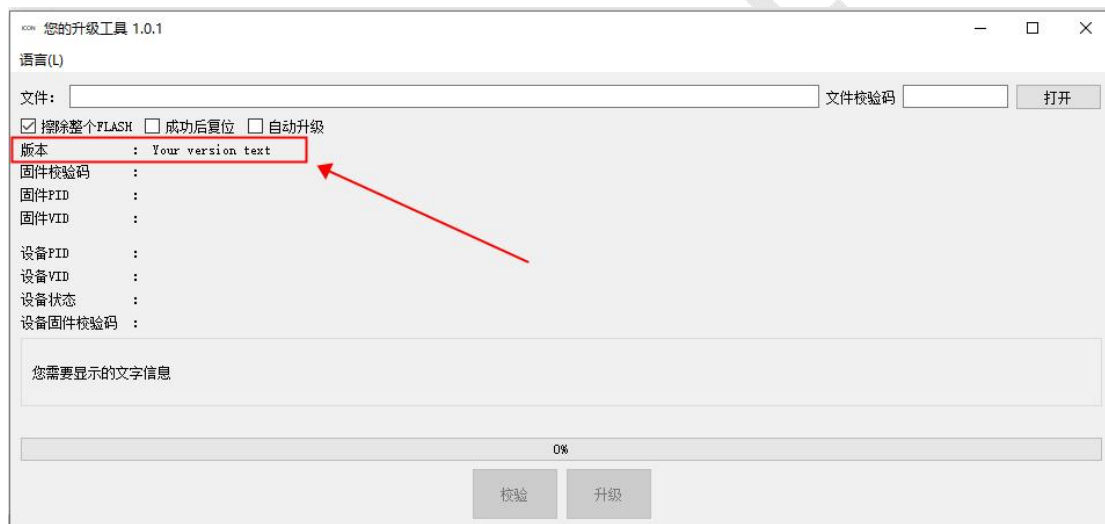


1.7 自定义版本信息配置

此功能详细配置项如下：

```
#####
#
#   需要显示的特定版本文字
#
#####
[ConfigVersion]
text_cn = 版本
text_en = Version
visible = true
value   = Your version text
```

以上配置在工具界面上的效果如下图：



1.8 固件信息显示配置

此功能详细配置项如下：

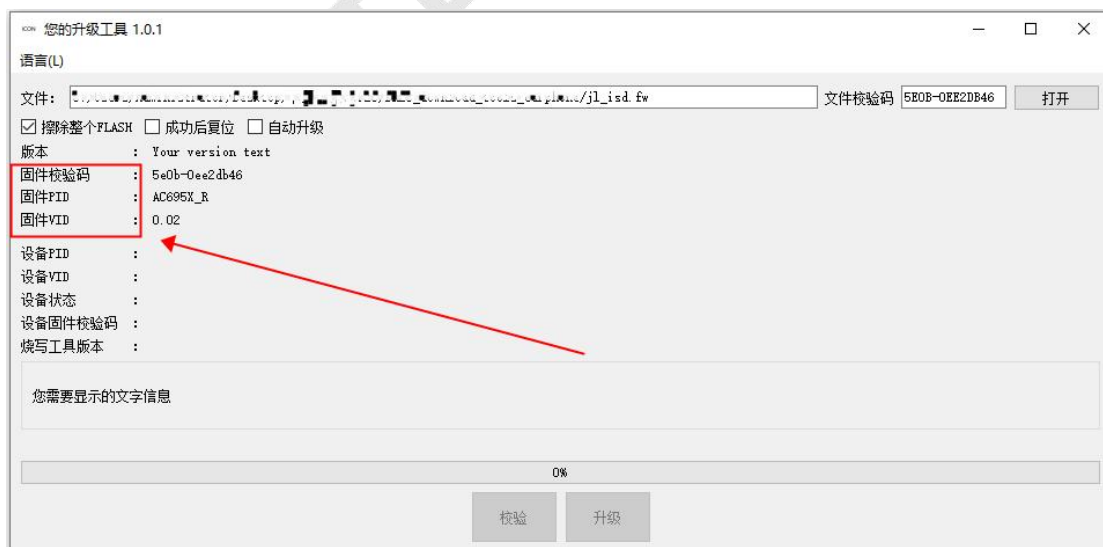
```
#####
#
#   固件信息显示配置
#
#####

[FirmwareCrc]
text_cn = 固件校验码
text_en = Firmware CRC
visible = true           ;是否显示固件程序校验码

[FirmwarePid]
text_cn = 固件 PID
text_en = Firmware PID
visible = true           ;是否显示固件文件中配置的 PID

[FirmwareVid]
text_cn = 固件 VID
text_en = Firmware VID
visible = true           ;是否显示固件文件中配置的 VID
```

以上配置在工具界面上的效果如下图：

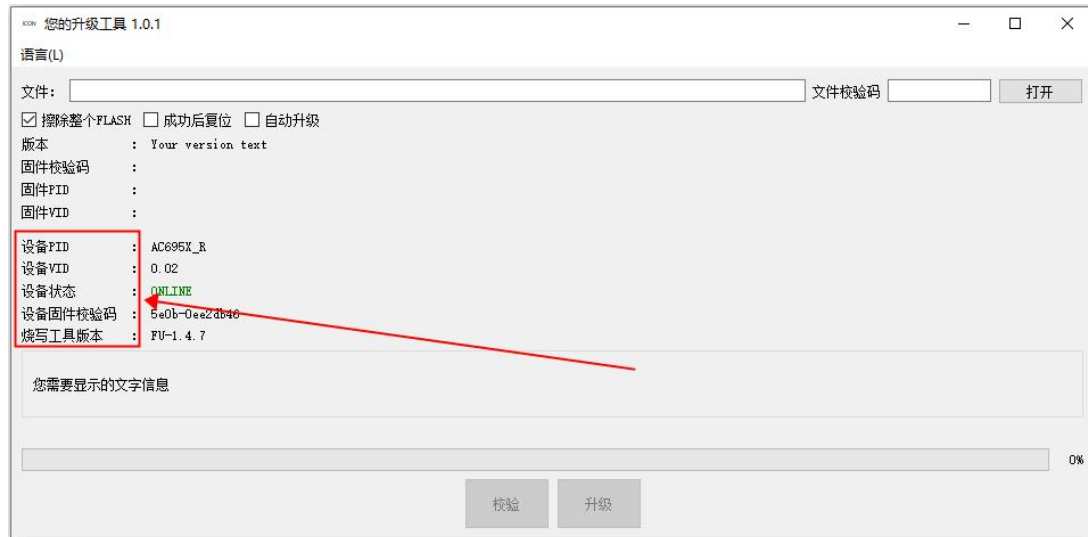


1.9 样机信息显示配置

此功能详细配置项如下：


```
#####  
#  
#   样机信息显示配置  
#  
#####  
[DevicePid]  
text_cn = 设备 PID  
text_en = Device PID  
visible = true           ;是否显示样机中的 PID  
  
[DeviceVid]  
text_cn = 设备 VID  
text_en = Device VID  
visible = true           ;是否显示样机中的 VID  
  
[DeviceCrc]  
text_cn = 设备固件校验码  
text_en = Device CRC  
visible = true           ;是否显示设备中的固件校验码  
  
[DeviceStatus]  
text_cn = 设备状态  
text_en = Device Status  
  
[BurnerVersion]  
text_cn = 烧写工具版本  
text_en = Burner Version  
visible = true           ;是否显示设备中固件是被哪个烧写工具烧写的  
  
[ChipCfgInfo]  
text_cn = 芯片配置信息  
text_en = Chip CFG Info  
visible = false          ;是否显示芯片配置信息，目前仅支持 AC104N, AD14N, AD15N
```

以上配置在工具界面上的效果如下图：

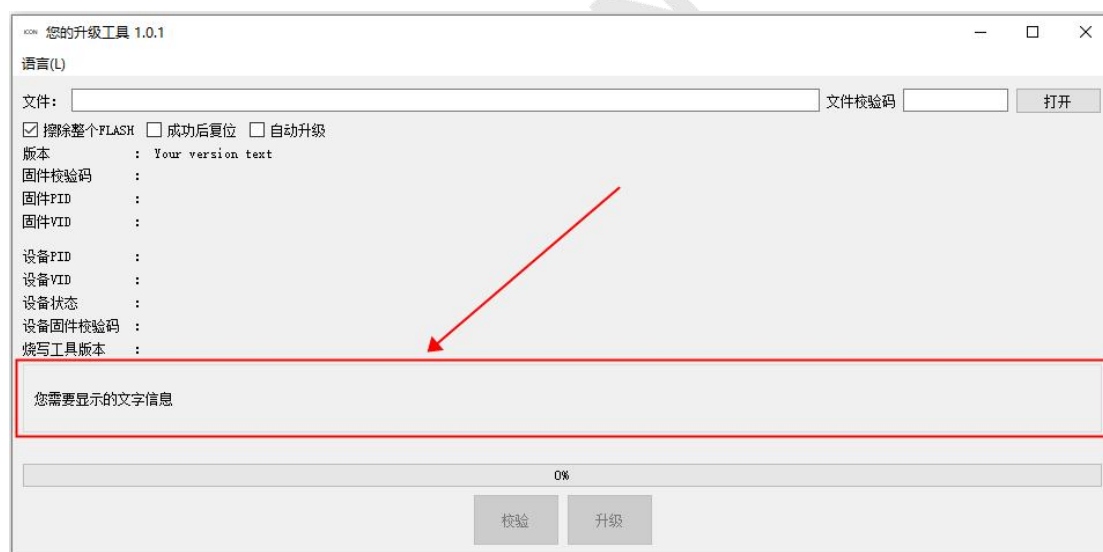


1.10 自定描述信息配置

此功能详细配置项如下：

```
#####
#
#   自定义描述信息
#
#####
[Description]
visible = true
text_cn = "
您需要显示的文字信息
"
text_en = "
your text information
"
```

以上配置在工具界面上的效果如下图：



1.11 升级和校验按钮文字配置

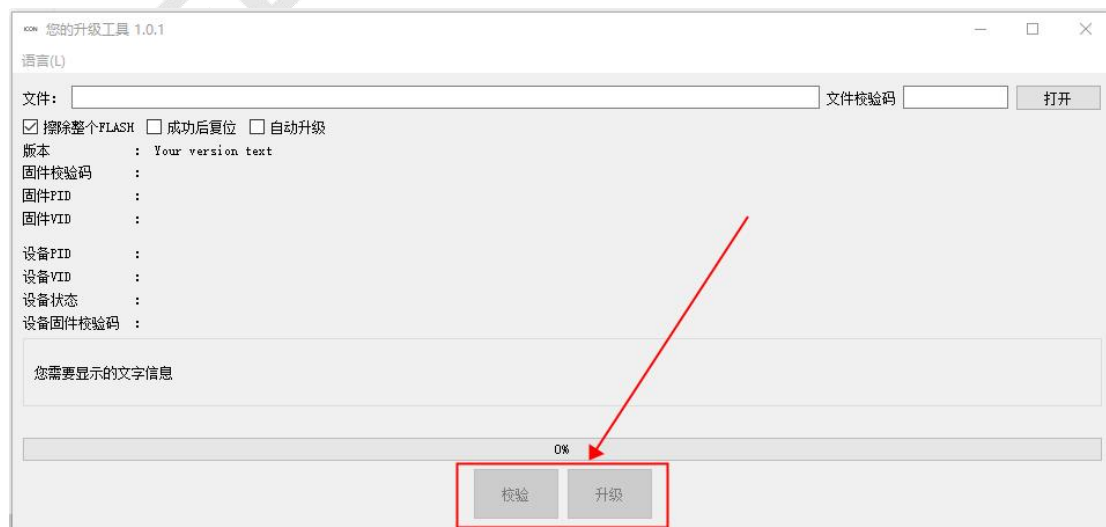
升级按钮和校验按钮只有在加载了固件文件和有可用设备时才会可用。此功能详细配置项如下：

```
#####
#
#   按钮配置
#
#####

#
#   升级按钮配置
#
[UpdateButton]
text_cn  = 升级
text_en  = Upgrade
visible  = false           ;是否显示升级按钮，如果升级按钮不显示，工具不会自动升级

#
#   校验按钮配置
#
[VerifyButton]
text_cn  = 校验
text_en  = Verify
visible  = true           ;是否显示校验按钮
```

以上配置在工具界面上的效果如下图：



1.12 认证码配置

如果在样机功能确认过程中，需要烧写序列号等第三方数据，那么可以通过开启认证码烧写功能，在升级样机固件的同时升级将序列号等第三方数据写入样机中。

此功能详细配置项如下：

```
#####
#
#   认证码文件选择栏配置
#
#####
[AuthCodeFileEdit]
label_text_cn = 认证码文件：
label_text_en = Auth Code File:
crc_text_cn   = 文件校验码
crc_text_en   = File CRC
open_text_cn  = 打开
open_text_en  = Open
file_type     = csv,hex           ;认证码文件格式，文件类型之间用逗号隔开（仅支持
csv 文件和 hex 文件）
visible       = true             ;是否显示认证码文件选择栏

[AuthCodeInfo]
text_cn = 认证码信息
text_en = Auth Code Info
visible = true                   ;是否显示认证码详细信息

[AuthCodeTotal]
text_cn = 总个数
text_en = Total Number
visible = true                   ;是否显示认证码总数

[AuthCodeUsed]
text_cn = 已使用个数
text_en = Number Of Used
visible = true                   ;是否显示已使用个数

[AuthCodeAvailable]
text_cn = 剩余的个数
text_en = Number Of Available
visible = true                   ;是否显示剩余个数

#
```

重写认证码选项仅在 AuthCodeFileEdit/visible=true 时有效

#

[OverwriteAuthCode]

text_cn = 重写认证码

text_en = Overwrite Auth Code

visible = true ; 是否显示

checked = false ; 是否默认勾选, 此配置仅在“visible=true”时有效

以上配置在工具界面上的效果如下图:

