



GProgrammer用户手册

版本： 2.0

发布日期： 2021-01-05

版权所有 © 2021 深圳市汇顶科技股份有限公司。保留一切权利。

非经本公司书面许可，任何单位和个人不得对本手册内的任何部分擅自摘抄、复制、修改、翻译、传播，或将其全部或部分用于商业用途。

商标声明

GOODIX 和其他汇顶商标均为深圳市汇顶科技股份有限公司的商标。本文档提及的其他所有商标或注册商标，由各自的所有人持有。

免责声明

本文档中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。

深圳市汇顶科技股份有限公司（以下简称“GOODIX”）对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。GOODIX对因这些信息及使用这些信息而引起的后果不承担任何责任。

未经GOODIX书面批准，不得将GOODIX的产品用作生命维持系统中的关键组件。在GOODIX知识产权保护下，不得暗或以其他方式转让任何许可证。

深圳市汇顶科技股份有限公司

总部地址：深圳市福田区腾飞工业大厦B座2层、13层

电话：+86-755-33338828 传真：+86-755-33338099

网址：www.goodix.com

前言

编写目的

本文档介绍了GProgrammer的安装过程以及各功能模块的操作使用，旨在帮助用户快速入门GProgrammer软件的使用。

读者对象

本文适用于以下读者：

- GR551x、GMF03x用户
- GR551x、GMF03x开发人员
- GR551x、GMF03x测试人员

版本说明

本手册为第6次发布，对应的软件版本为GProgrammer v1.2.14。

修订记录

版本	日期	修订内容
1.5	2020-05-30	首次发布
1.6	2020-06-30	升级软件版本； “3.3.7 芯片配置”章节sectors软件可修改； “3.3.9 命令行”章节GR551x_console.exe命令程序的erase和download命令新增一个参数，新增“GR551x_encrypt_signature.exe”和“用户自定义Windows脚本”章节； “3.3.5 加密加签”章节新增用于验证签名的公钥的Hash文件，变更加密加签后的文件后缀，支持加密加签或者仅加签。
1.7	2020-08-30	升级软件版本； “3.2 芯片选型”章节GR551x系列新增GR5515I0ND芯片； “3.3.3 固件”章节变更“Delete”和“Startup”图标
1.8	2020-09-30	“3.3.3.1 下载固件”章节增加下载固件失败的说明
1.9	2020-11-26	升级软件版本并更新对应软件页面截图
2.0	2021-01-05	升级软件版本并更新对应软件页面截图，更新芯片选型界面截图和固件界面截图

目录

前言.....	I
1 简介.....	1
2 安装指南.....	3
2.1 安装要求.....	3
2.2 安装软件.....	3
3 GProgrammer使用说明.....	5
3.1 硬件连接.....	5
3.2 芯片选型.....	6
3.3 GR551x系列.....	8
3.3.1 主界面.....	8
3.3.2 连接管理.....	9
3.3.3 固件.....	10
3.3.3.1 下载固件.....	11
3.3.3.2 操作序列.....	12
3.3.4 Flash操作.....	13
3.3.4.1 内部Flash.....	14
3.3.4.2 外部Flash.....	18
3.3.5 加密加签.....	20
3.3.5.1 eFuse设置.....	21
3.3.5.2 eFuse下载.....	22
3.3.5.3 固件加密加签.....	23
3.3.6 eFuse展示.....	24
3.3.7 芯片配置.....	25
3.3.7.1 初始化NVDS.....	26
3.3.7.2 读取参数.....	27
3.3.7.3 写入参数.....	28
3.3.7.4 新增USER参数.....	29
3.3.7.5 修改参数.....	30
3.3.7.6 删除USER参数.....	31
3.3.7.7 导入与导出.....	31
3.3.8 设备日志.....	32
3.3.9 命令行.....	33
3.3.9.1 GR551x_console.exe.....	34
3.3.9.2 GR551x_encrypt_signature.exe.....	36
3.3.9.3 用户自定义Windows脚本.....	37
3.4 GMF03x系列.....	37
3.4.1 主界面.....	37

3.4.2 连接管理.....38

3.4.3 内存/文件操作..... 41

 3.4.3.1 读写内存..... 41

 3.4.3.2 展示文件..... 42

3.4.4 Flash操作..... 43

 3.4.4.1 编程Flash.....44

 3.4.4.2 擦除Flash.....44

 3.4.4.3 Flash读/写保护..... 45

3.4.5 用户选项字节配置.....46

3.5 帮助信息..... 47

1 简介

GProgrammer是一款支持Goodix芯片的Flash Programming工具，仅可运行于Windows平台。

GProgrammer的主要特性如下：

- 支持产品选型
 - GR551x系列
 - GMF03x系列
- 支持SWD与串口连接方式
- 根据产品型号自动适配软件界面
- 适配GR551x系列的功能：
 - 下载固件
 - Flash操作
 - 制定产品信息
 - 下载文件至eFuse
 - 查看eFuse存储信息
 - 固件加密加签
 - 配置NVDS参数
 - 显示设备日志
 - 支持命令行操作
- 适配GMF03x系列的功能：
 - 读/写SRAM和Flash
 - 展示bin、hex文件内容
 - Flash编程和擦除
 - Flash读保护和写保护
 - 配置用户选项字节

GProgrammer软件主界面（以GR551x系列为例），如[图 1-1](#)所示。

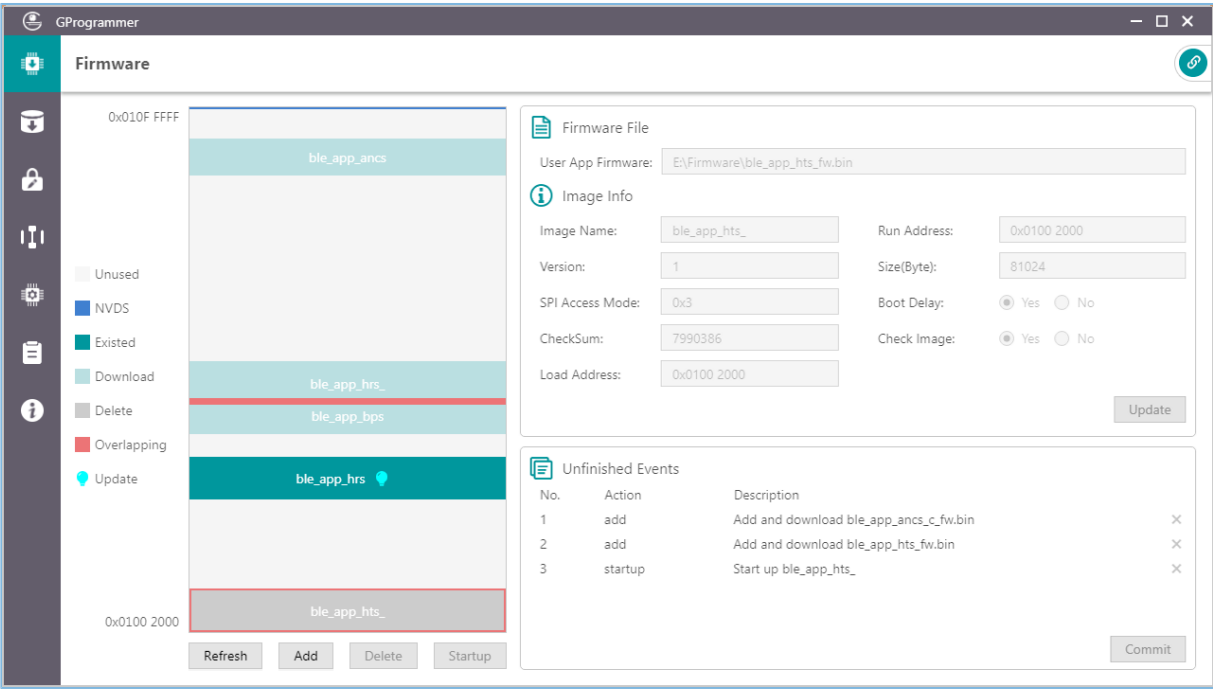


图 1-1 GProgrammer软件主界面GR551x系列

2 安装指南

本章主要介绍GProgrammer安装的环境要求以及安装过程。

2.1 安装要求

- 硬件环境

表 2-1 硬件要求

名称	描述
CPU	1.6 GHz及以上处理器
RAM	1 GB及以上RAM容量

- 操作系统

表 2-2 操作系统

名称	描述
Windows	Windows 7/Windows 10版本（32位/64位）

2.2 安装软件

GProgrammer仅支持在Windows平台下安装使用，其可执行安装程序为*GProgrammer Setup 1.2.14.exe*。

用户可按照以下步骤安装GProgrammer：

1. 双击GProgrammer的可执行安装程序*GProgrammer Setup 1.2.14.exe*，进入GProgrammer安装界面（如图 2-1所示），然后按照安装向导提示逐步完成安装操作。

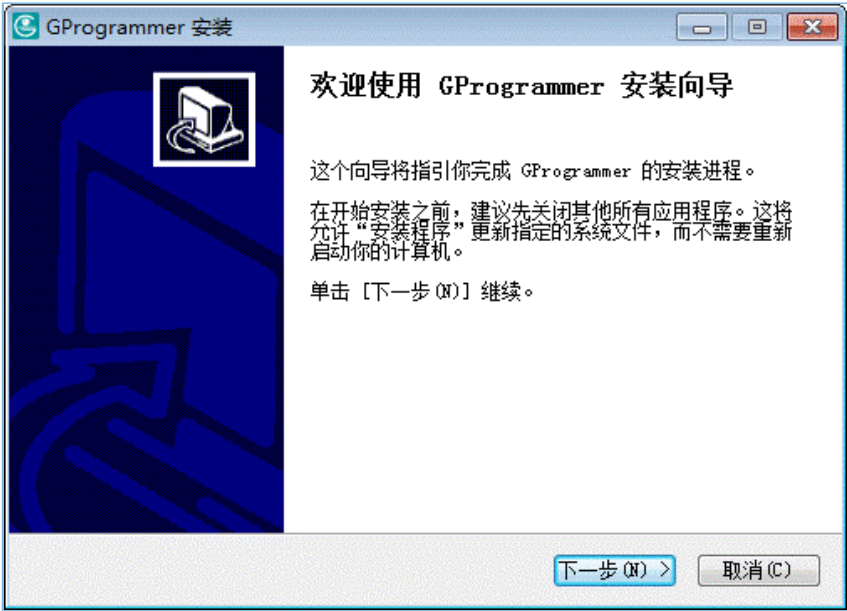


图 2-1 GProgrammer安装界面

2. GProgrammer安装完成后，将提示用户继续安装JLink驱动，如图 2-2所示。



图 2-2 安装JLink提示

说明:

若用户在安装GProgrammer之前已在PC上安装了JLink驱动，则可取消勾选“安装JLink”。

3. JLink安装完成后，用户可通过桌面或开始菜单的快捷方式启动GProgrammer软件。

3 GProgrammer使用说明

本章主要介绍GProgrammer各功能模块的操作使用。

3.1 硬件连接

使用GProgrammer之前，请先完成主机与目标板的硬件连接，可选择SWD连接或串口连接方式。

- SWD连接

选择SWD连接方式时，用户需使用J-Link仿真器来连接PC和目标板。使用USB线将仿真器的一端与PC相连，使用杜邦线将仿真器的另一端与目标板上的芯片管脚相连。

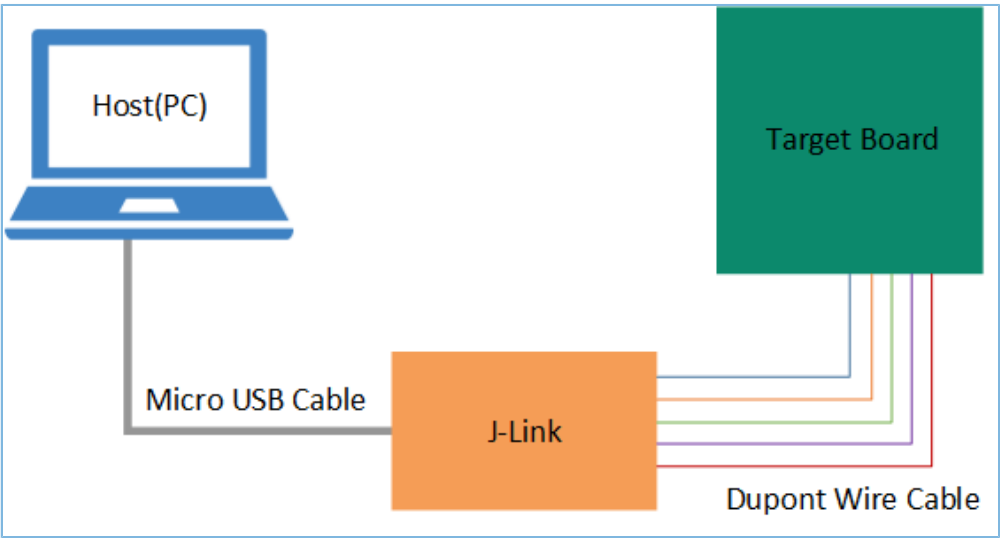


图 3-1 主机与目标板的连接框图

J-Link仿真器与芯片管脚的连接对应关系如下：

表 3-1 J-Link仿真器与芯片的管脚连接对应关系

J-Link仿真器管脚	GR551x芯片管脚	GMF03x芯片管脚
VCC	VCC	VCC
GND	GND	GND
SWDIO	GPIO_1	PA13
SWCLK	GPIO_0	PA14

说明:

如果目标板上集成了J-Link仿真器芯片，用户可直接使用USB线连接PC与目标板。

- 串口连接

选择串口连接方式时，用户则需使用一个USB转串口的转接板来连接主机与目标板。使用USB线将转接板的一端与PC相连，使用杜邦线将转接板的另一端与目标板上的芯片管脚相连。

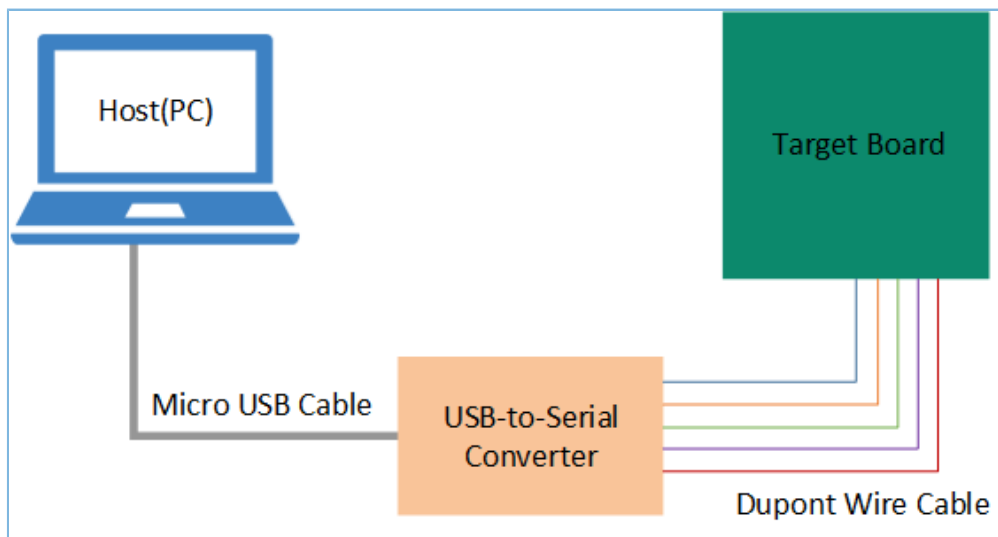


图 3-2 主机与目标板的连接框图

转接板与芯片管脚的连接对应关系如下：

表 3-2 转接板与芯片的管脚连接对应关系

转接板管脚	GR551x芯片管脚	GMF03x芯片管脚
VCC	VCC	VCC
GND	GND	GND
TX	GPIO_1	PA15
RX	GPIO_0	PA14
RTS	CHIP_EN	N/A

另外，GMF03x芯片的BOOT0管脚还需上拉为高电平，详情请参考《GMF03x自举程序说明》“2 自举程序的激活与硬件连接”。

说明:

如果目标板上集成了USB转串口芯片，用户可直接使用USB线连接PC与目标板。

3.2 芯片选型

使用GProgrammer时，必须先选择目标板的主控芯片型号，然后点击右下角的“OK”按钮，才能进入GProgrammer操作主界面。

说明:

启动GProgrammer后，默认进入芯片选型页面。

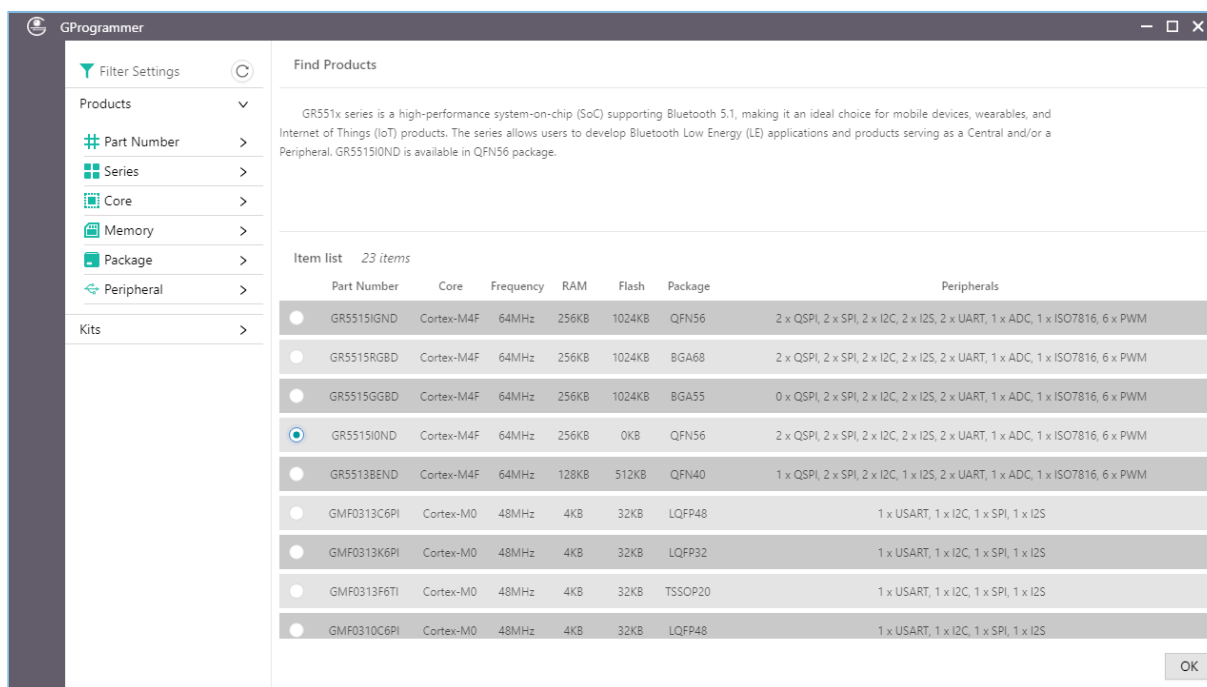


图 3-3 芯片选型界面

在芯片选型页面，左侧为筛选条件的设置面板；右侧为筛选结果的展示面板。用户可按照产品型号、产品系列、内核、内存大小、集成的外设、封装形式等条件进行产品筛选。

通过“Series”选项，可快速筛选出GR551x或GMF03x系列芯片。

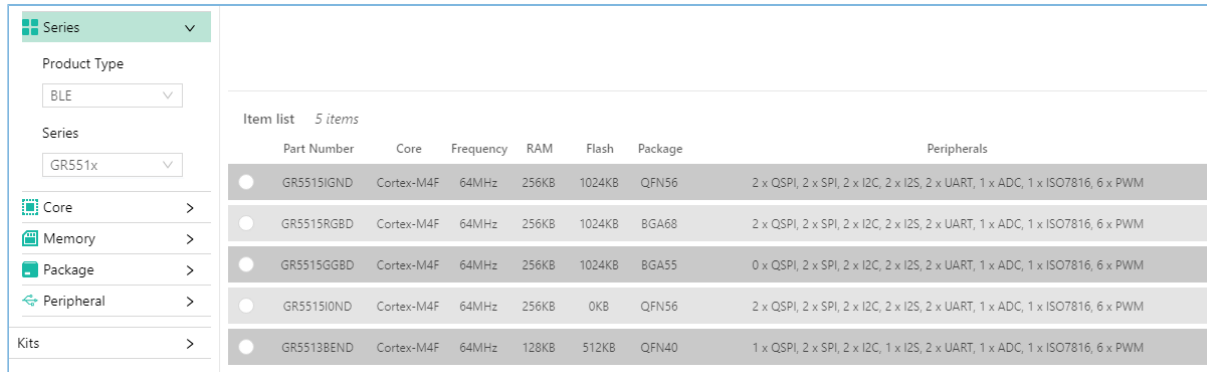


图 3-4 筛选GR551x系列芯片

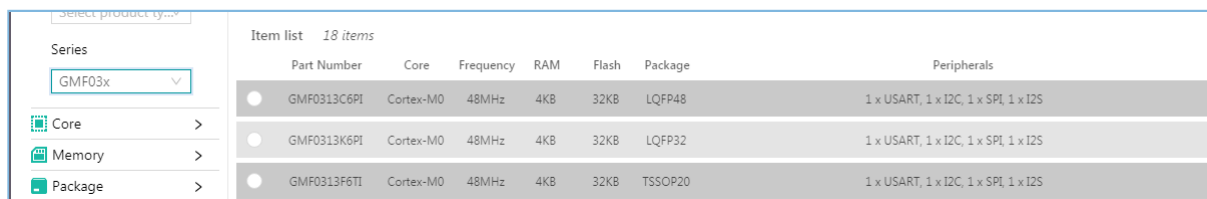


图 3-5 筛选GMF03x系列芯片

目前，GProgrammer支持的产品如下：

- GR551x系列：GR5515IGND、GR5515RGBD等5个产品型号。
- GMF03x系列：GMF0313K6PI、GMF0313C6PI、GMF0313F6TI等18个产品型号。

说明:

芯片选型页面上所展示的芯片外设，并非芯片的所有外设。如需获取芯片的全面信息，请参考该系列芯片的数据手册。

3.3 GR551x系列

本章主要介绍GR551x系列芯片的GProgrammer功能操作。

3.3.1 主界面

选择GR551x系列芯片后，将进入GR551x芯片对应的软件主界面，如图 3-6所示。

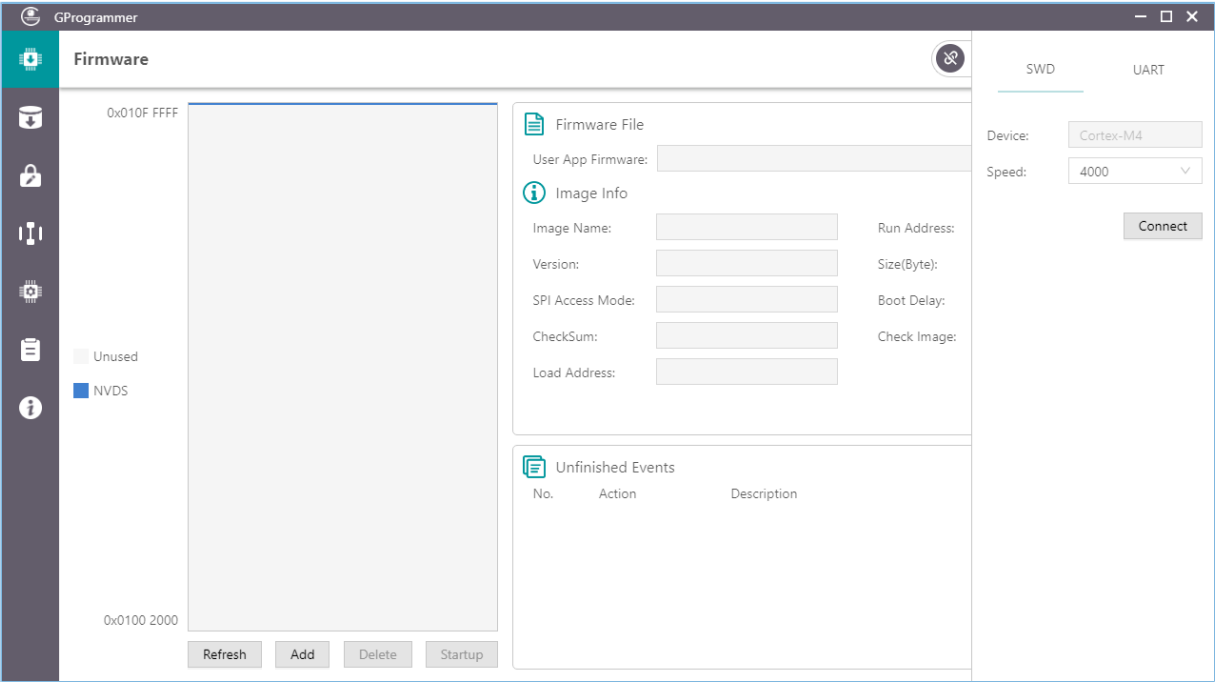


图 3-6 GProgrammer软件主界面-GR551x系列

软件界面的左侧为功能导航栏（具体描述见表 3-3），右侧为功能操作区域。

表 3-3 功能导航栏选项

图标	功能名称	描述
	Firmware	点击该图标，进入“固件”操作页面
	Flash	点击该图标，进入“Flash”操作页面
	Encrypt & Sign	点击该图标，进入“加密加签”操作页面
	eFuse Layout	点击该图标，进入“eFuse展示”操作页面
	Chip Configuration	点击该图标，进入“芯片配置”操作页面
	Device Log	点击该图标，进入“设备日志”操作页面
	Help	点击该图标，进入“帮助”操作页面

3.3.2 连接管理

用户可管理控制目标板与主机之间的连接。

点击软件界面右上角的 ，可展开连接管理面板；再次点击 ，可收起并隐藏该面板。

GProgrammer支持SWD和串口两种连接方式。

- SWD连接

在SWD连接方式下，用户只需配置传输速率“Speed”，即可点击“Connect”按钮连接目标板。

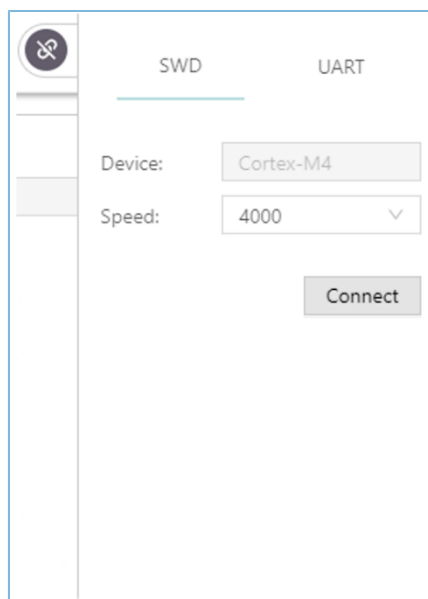


图 3-7 SWD连接

- 串口（UART）连接

在串口（UART）连接方式下，用户可根据实际情况配置串口号“Port”（需点击“Refresh”按钮获取串口号列表，再选择正确的串口号）和波特率，其他参数默认配置不可修改。

参数配置完成后，点击“Connect”按钮连接目标板。

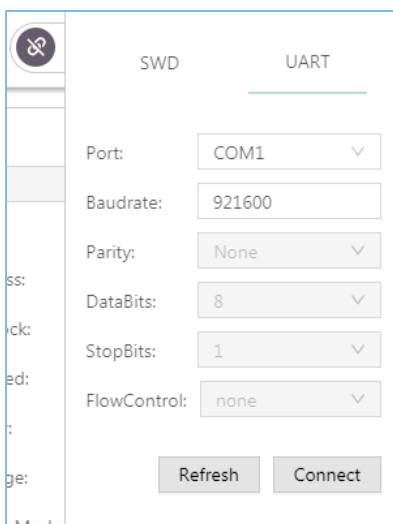


图 3-8 UART连接

目标板连接成功后，连接管理面板将自动收起并隐藏。同时，按钮  将变成 ，表示当前的连接状态为“已连接”。

如需断开连接，再次点击 ，打开连接管理面板，然后点击“Disconnect”按钮即可。

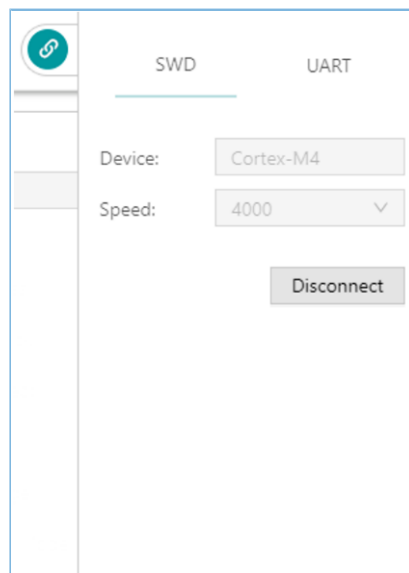


图 3-9 断开连接

3.3.3 固件

点击左侧功能导航栏的图标 ，进入“Firmware”（固件操作）页面。

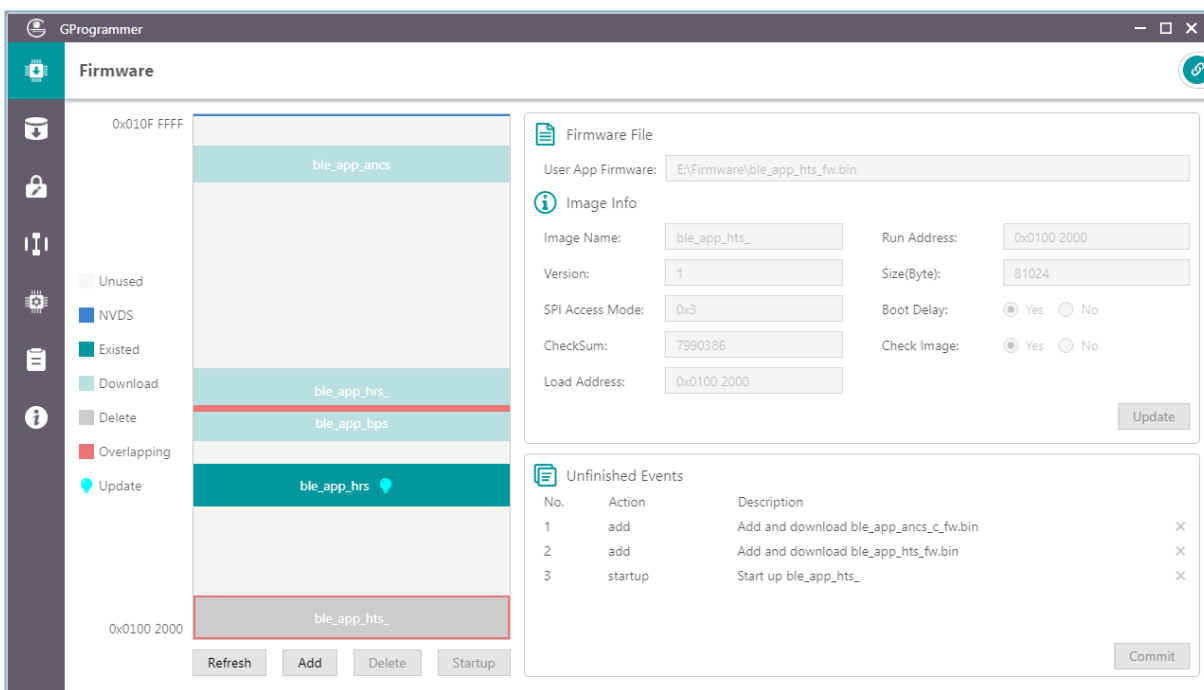


图 3-10 “Firmware” 页面

用户可将 User APP 固件下载至 Flash 中 0x01002000 ~ 0x010FFFFF 的连续空间。

3.3.3.1 下载固件

GProgrammer以图形化方式展示Flash Firmware Layout（如图 3-11所示），可帮助用户直观了解Flash中固件区域的占用情况。

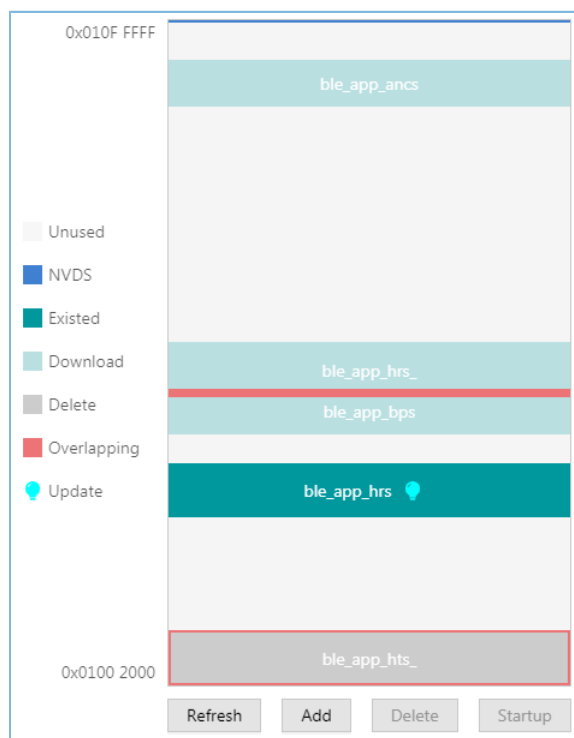


图 3-11 Flash Firmware Layout 示意图

- （浅灰色）：可下载的Flash空间。
- （深蓝色）：软件默认配置的NVDS区域。固件不能下载至该区域。
- （深灰色）：待删除的固件，如ble_app_ancs。
- （浅绿色）：待下载的固件，如ble_app_hrs。
- （深绿色）：已存在于Flash中的固件，如ble_app_bps。
- （红色）：两个固件的占用空间的重叠区域，如ble_app_T3u和ble_app_hrs。

用户下载固件时，请按以下步骤操作：

1. 点击“Add”按钮，从本地添加需下载的固件文件。固件文件添加后，可查看到该固件的详细信息，例如固件文件路径、固件Image Info信息。
2. 点击“Commit”按钮，将固件文件下载到Flash。

下载完成后，若Layout示意图中的固件由“浅绿色”变为“深绿色”，则表示固件下载成功。

说明:

- 1. 连接成功后，将自动从连接的目标板的Flash中读取已存在的固件。
- 2. 下载固件时，如果连接JLink失败，GR5515 Starter Kit开发板无法连接或者正常下载固件。GR551x芯片当前可能处于睡眠状态（即开启了睡眠模式的固件正在运行），可先按下GR551x Starter Kit开发板的R“ESET”键，并间隔1秒左右重新下载固件。如果仍然无法下载固件，可擦除Flash后，重新下载固件。

3.3.3.2 操作序列

用户可一次性执行多个操作，例如：一次性下载多个固件，并将某个固件设置为启动项。用户执行的操作均为“预操作”，需点击“Commit”按钮后才会真正地执行，并且操作序列将在“Unfinished Events”栏中显示，如图 3-12所示。

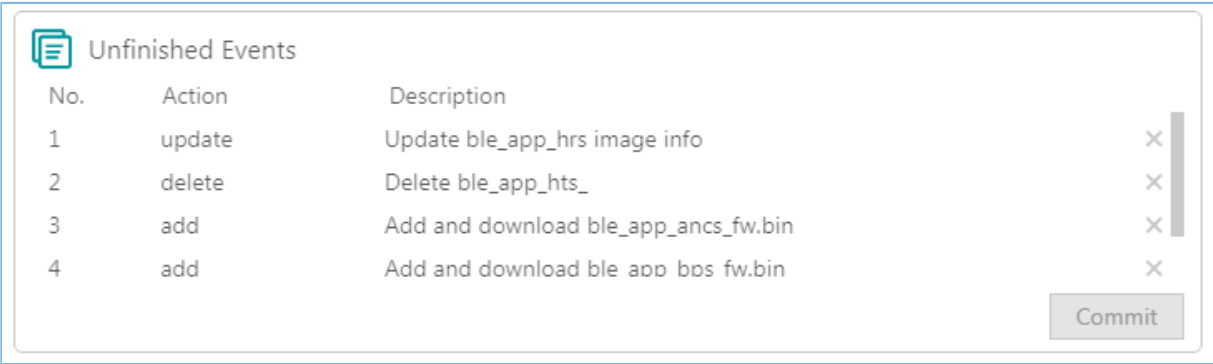


图 3-12 操作列表

用户可执行的固件操作主要包括：

表 3-4 固件操作

操作名称	操作按钮	描述
添加固件	Add	点击“Add”按钮，从本地浏览添加固件文件。
刷新固件	Refresh	指获取当前目标板Flash中的固件信息。点击“Refresh”后，会被直接执行。 如果“Unfinished Events”有其他未执行操作，如“开始运行”或“更新固件”，那么这些未执行操作将被撤回，已修改的参数也会被重置。
删除固件	Delete	指删除目标板Flash中已存在的固件。在Flash Layout示意图中选中待删除固件，然后点击“Delete”按钮，该固件会变成深灰色，且“Unfinished Events”中将增加一条delete操作。
设置为启动项	Startup	指将固件设为启动项，即下载完成后立即运行该固件。在Flash Layout示意图中选中固件，然后点击“Startup”按钮，该固件的右侧将显示🔥图标，且“Unfinished Events”中将增加一条startup操作。固件运行后，连接将会自动断开。

操作名称	操作按钮	描述
更新固件	Update	指更新目标板Flash中已存在的固件的信息。在Flash Layout示意图中选中待更新固件，然后修改固件信息（修改后的参数信息将变成红色），最后点击“Update”按钮，该固件的右侧将显示  图标，且“Unfinished Events”中将增加一条update操作。 执行update操作后，所有的参数选项都将被锁定，不允许再编辑。如需重新修改，则必须先撤回之前的update操作。

说明:

- 在操作序列中，点击操作项右侧的 , 可撤销该操作。
- 对于有关联关系的两项操作，撤销其中一项可能会自动撤销另一项。例如：添加一个固件文件并将其设置为启动项，即执行“添加”和“设置为启动项”两项操作。如果撤销“添加”操作，那么“设置为启动项”操作将被自动撤销。

另外，如果固件区域里存在重叠区域，“Commit”按钮将置灰不可用，冲突解决后才可提交并执行操作序列。

说明:

若两个固件完全重叠，单击可选中其中一个固件，再双击可选中另一个固件。

3.3.4 Flash操作

点击左侧功能导航栏的图标 , 进入“Flash”（Flash操作）页面。

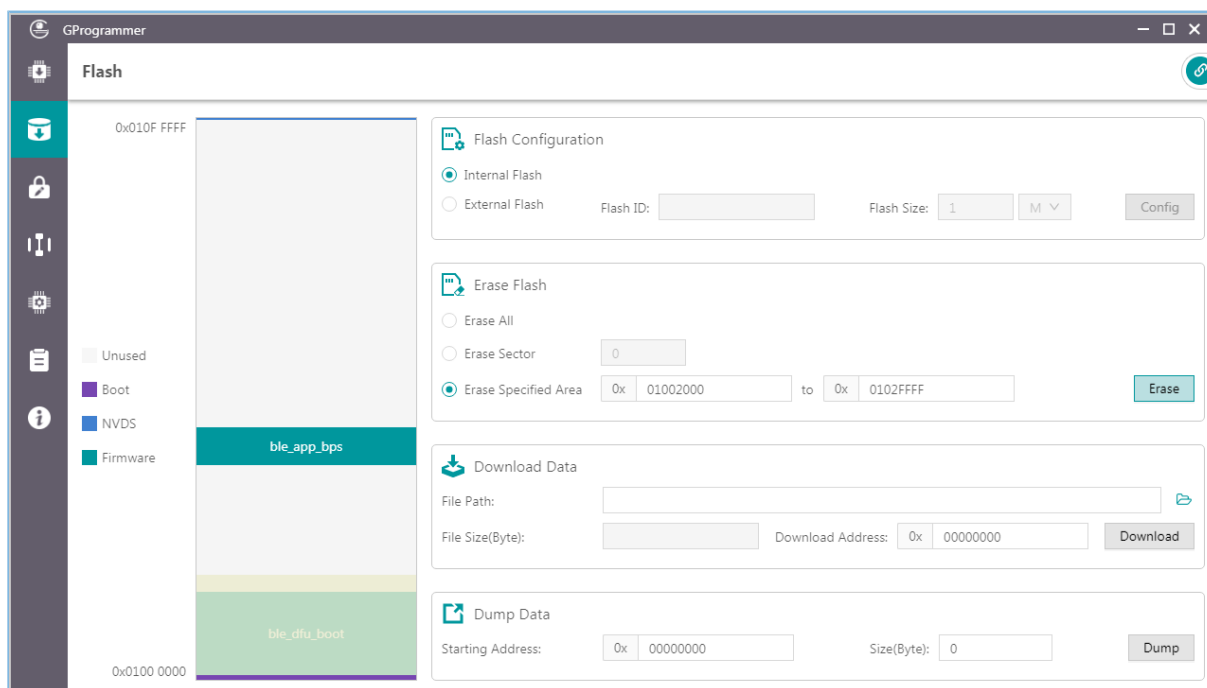


图 3-13 “Flash” 页面

用户可对GR551x内部Flash和外部Flash进行操作，包括擦除Flash、下载数据至Flash、从Flash中Dump数据。

与固件模块相似，该模块也采用图形化的方式展示Flash Layout信息。

- （浅灰色）：可用的Flash空间。
- （深蓝色）：配置的NVDS区域。
- （深紫色）：Boot Info区域（0x01000000 ~ 0x01002000），选择使用内部Flash时将自动加载并显示Boot Info区域。
- （深绿色）：已存在Flash中的固件，如ble_app_bps。
- （淡黄色）：待操作区域，如待擦除的Flash区域。

3.3.4.1 内部Flash

3.3.4.1.1 配置Flash

当用户操作内部Flash时，首先需在“Flash Configuration”栏选择“Internal Flash”。

页面左侧的Flash Layout示意图将同步“Firmware”模块的Firmware Layout信息，从而获取当前Flash中的固件区域以及NVDS区域，同时加载Boot Info区域。

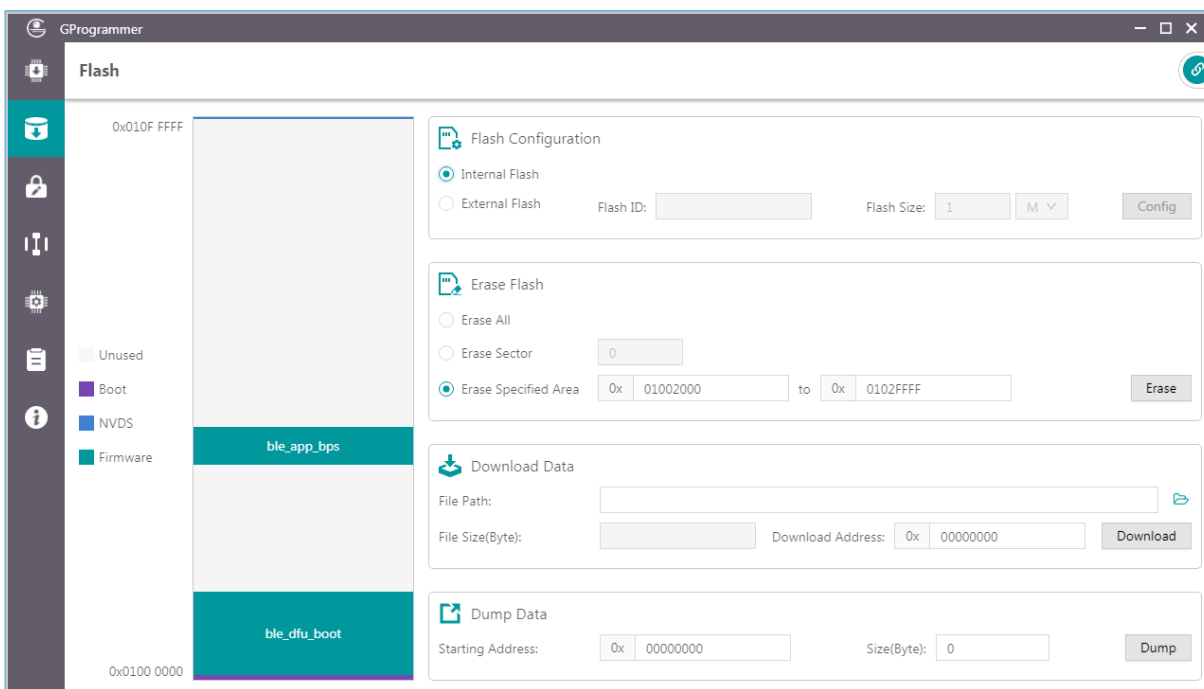


图 3-14 选择内部Flash

3.3.4.1.2 擦除Flash

GProgrammer支持三种Flash擦除方式：全片擦除、Sector擦除和区域擦除。

- 全片擦除（Erase All）

全片擦除，指整体擦除Flash空间。

擦除后，内部Flash中的Boot和NVDS区域内容将被清空，所有Firmware将被删除。

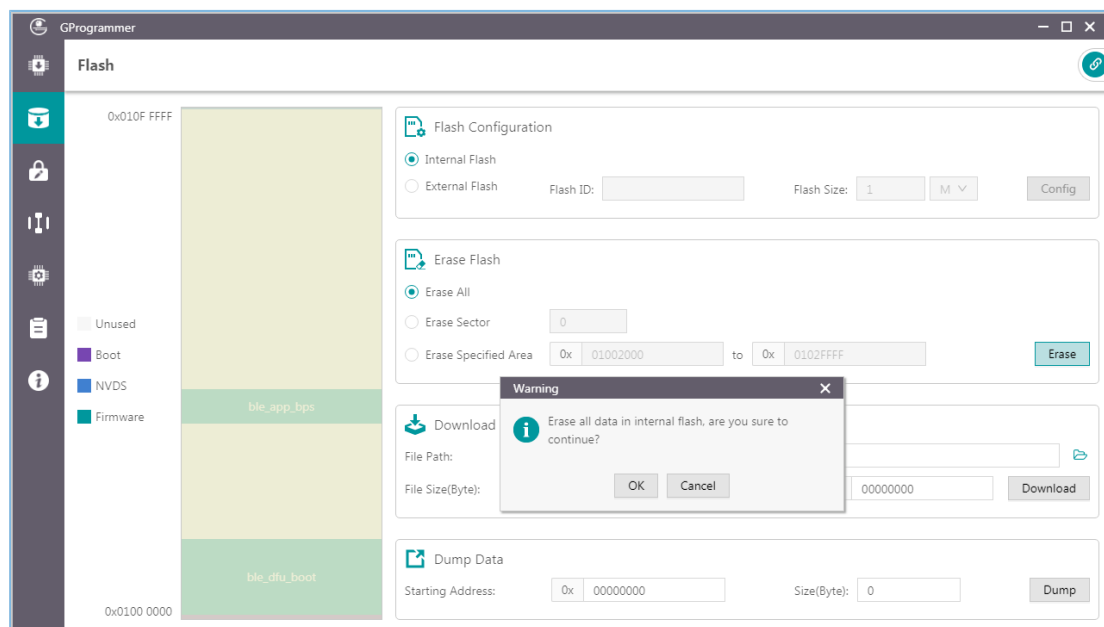


图 3-15 全片擦除

- Sector擦除（Erase Sector）

Sector擦除，指擦除指定的Sector区域（4 KB大小）。

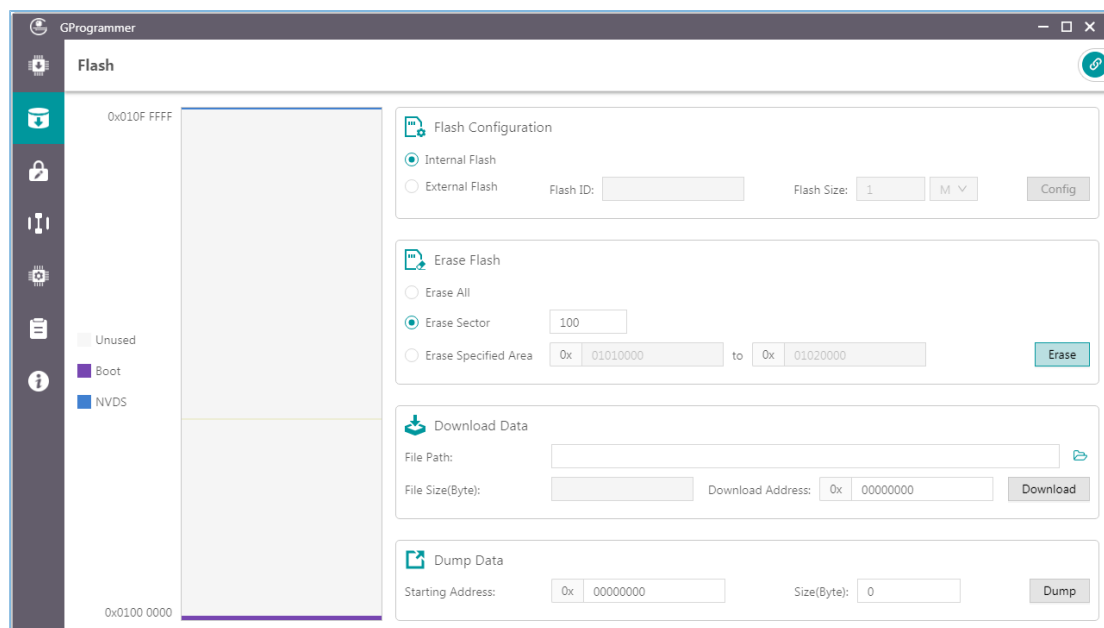


图 3-16 Sector擦除

- 区域擦除（Erase Specified Area）

区域擦除，指在指定的地址范围内按Sector逐一擦除。

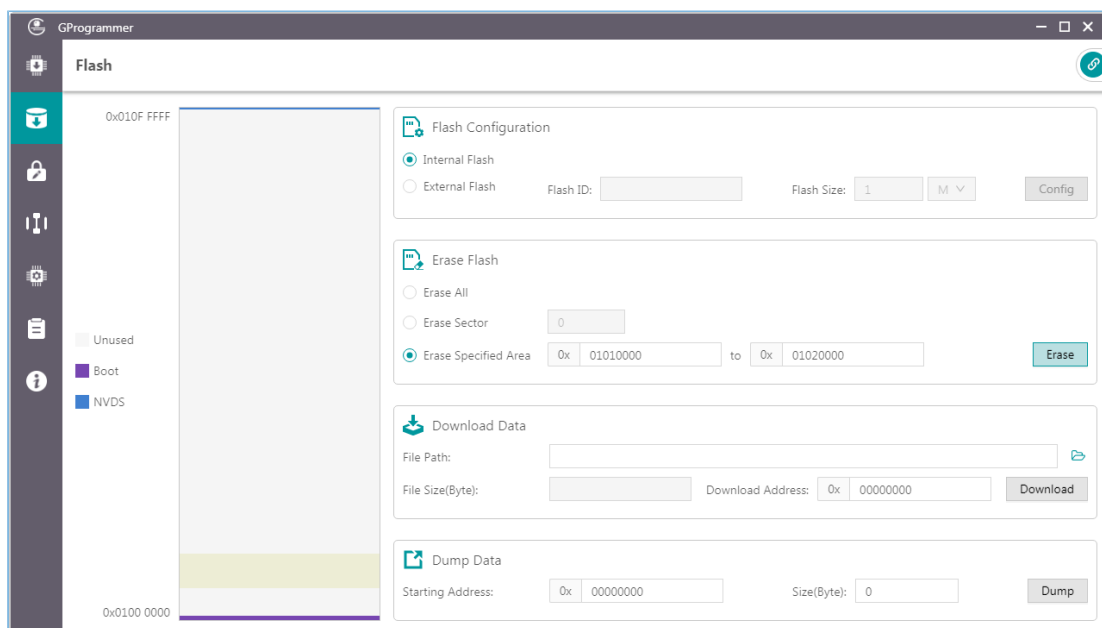


图 3-17 浏览选择数据文件

3.3.4.1.3 下载数据

下载数据至Flash时，用户只需浏览添加数据bin文件，然后设置下载起始地址即可。

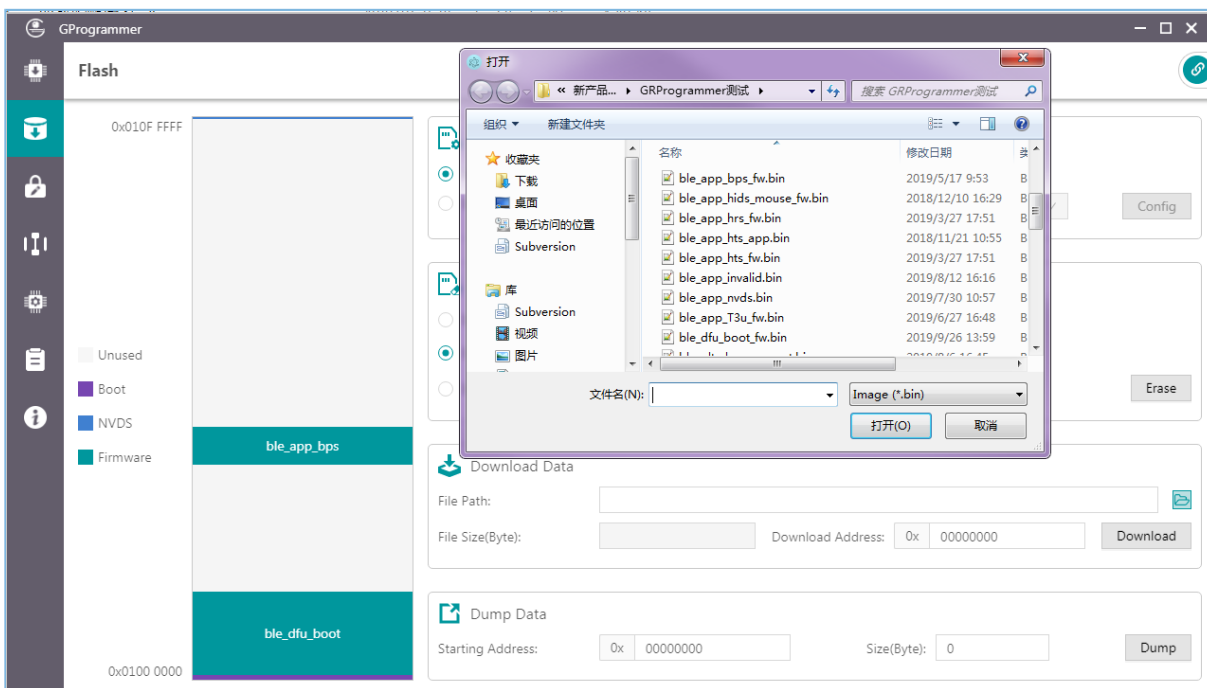


图 3-18 浏览选择数据文件

当下载的数据文件过大或设置的下载起始地址超出范围时，将导致Flash溢出报错。

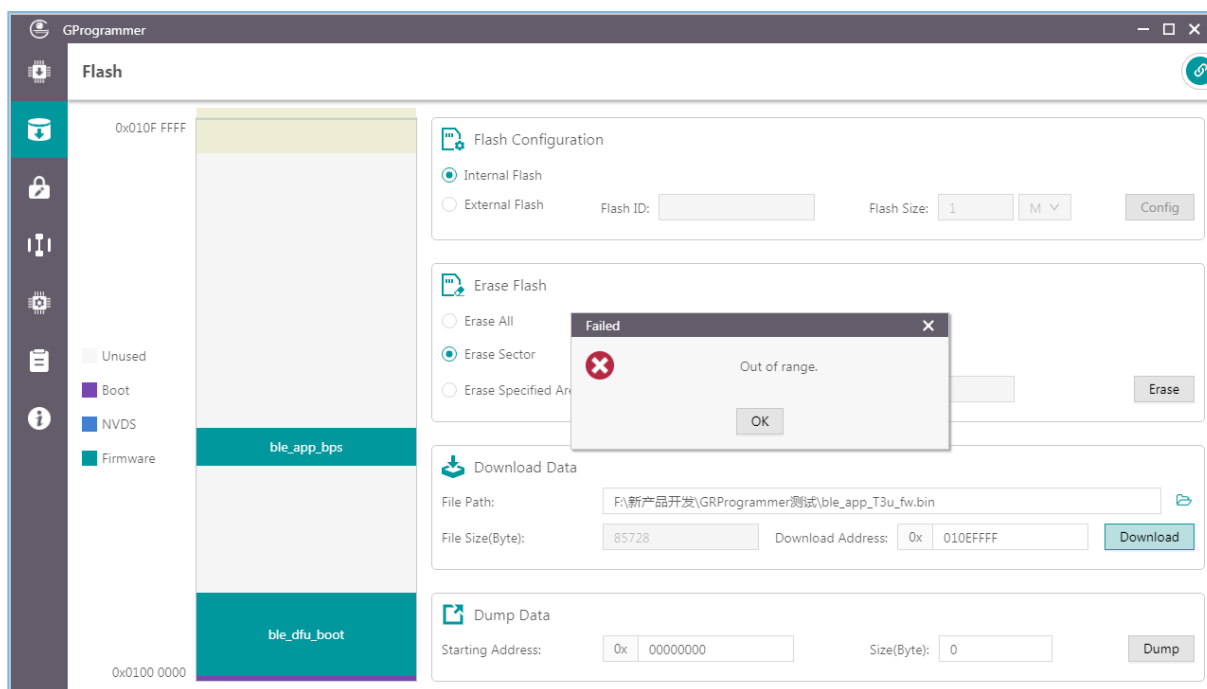


图 3-19 Flash 溢出

说明:

如果使用SWD方式连接目标板，那么用户可以强制下载数据到Boot区域。如果使用串口方式连接目标板，那么用户无法强制下载数据到Boot区域。

3.3.4.1.4 Dump数据

用户可通过指定起始地址和数据大小，将Flash中的任意数据Dump到本地保存为文件。

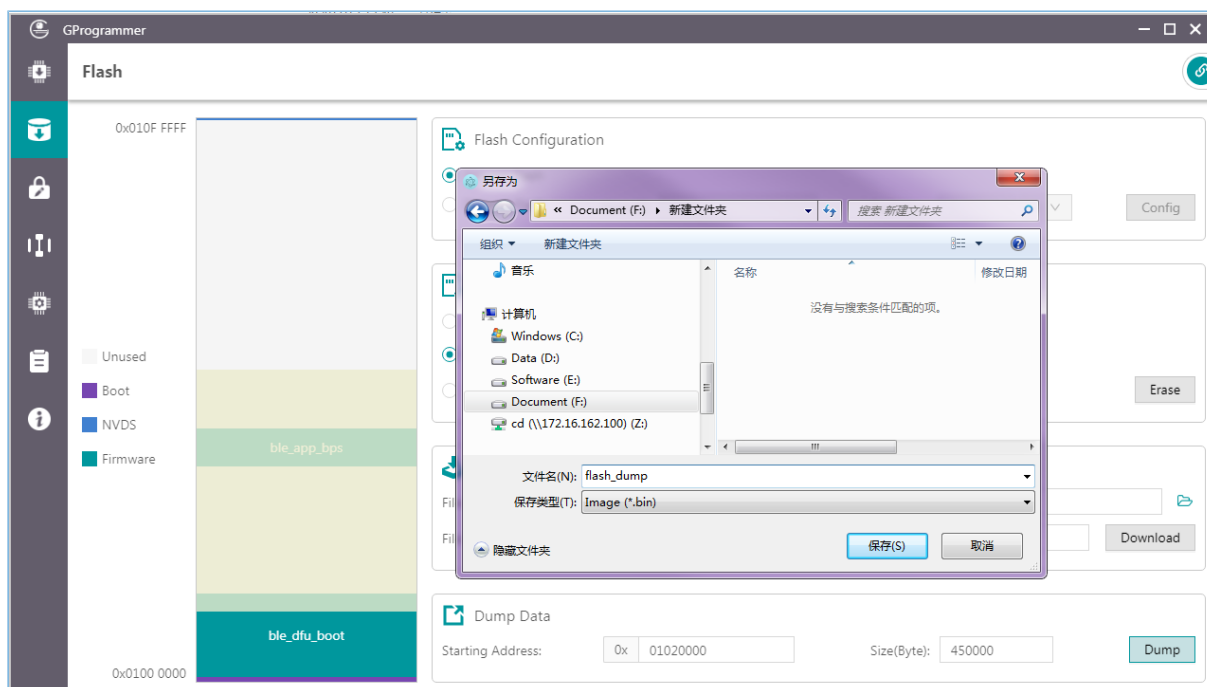


图 3-20 Dump数据

3.3.4.2 外部Flash

3.3.4.2.1 配置Flash

当用户操作外部Flash时，首先需在“Flash Configuration”栏选择“External Flash”；然后点击“Config”按钮，打开外部Flash配置窗口。用户可根据实际情况配置PIN脚。

配置完成后，点击“Apply”按钮应用配置信息。

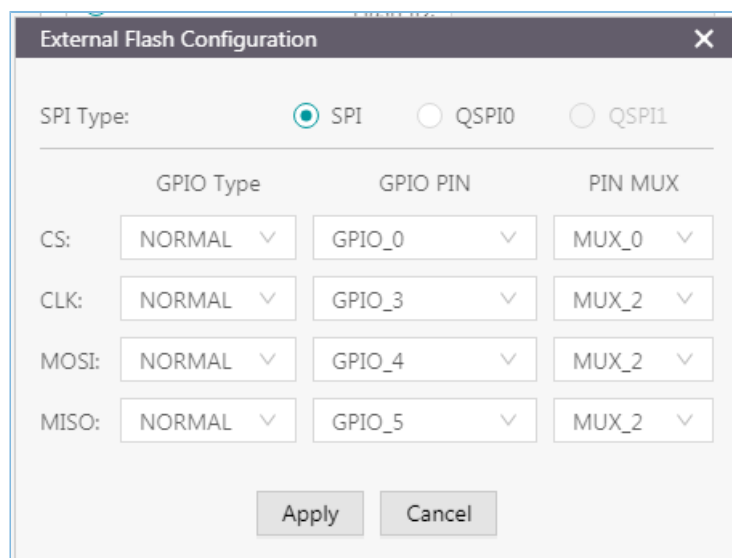


图 3-21 SPI配置项

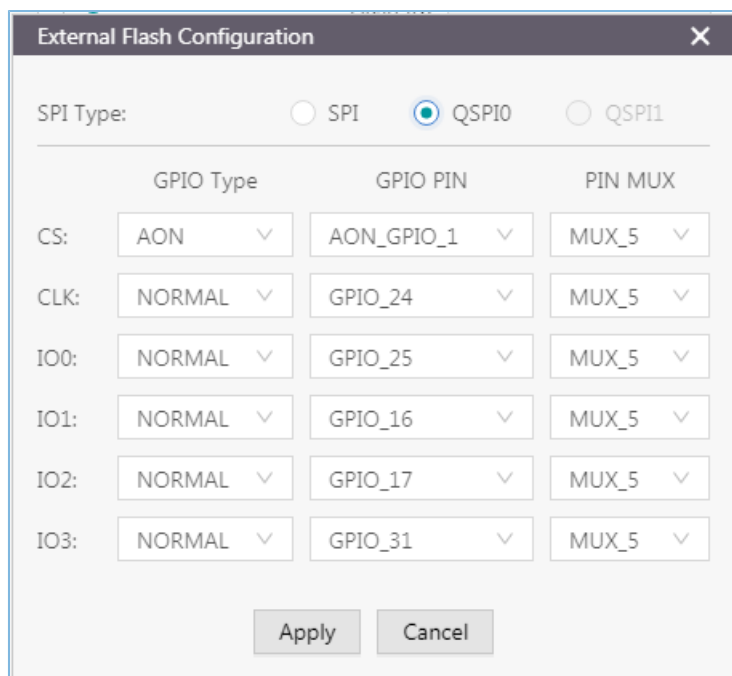


图 3-22 QSPI0配置项

- Flash Size 配置

用户完成PIN脚配置并应用后，软件将读取并显示外部Flash ID，并根据Flash ID自动配置Flash Size。

说明:

点击“Apply”按钮之前，必须确保外部Flash按照引脚配置与目标板正确连接。否则，外部Flash无法与目标板正常通信。

若无法判定Flash Size，则用户需自行选择设置Flash Size。

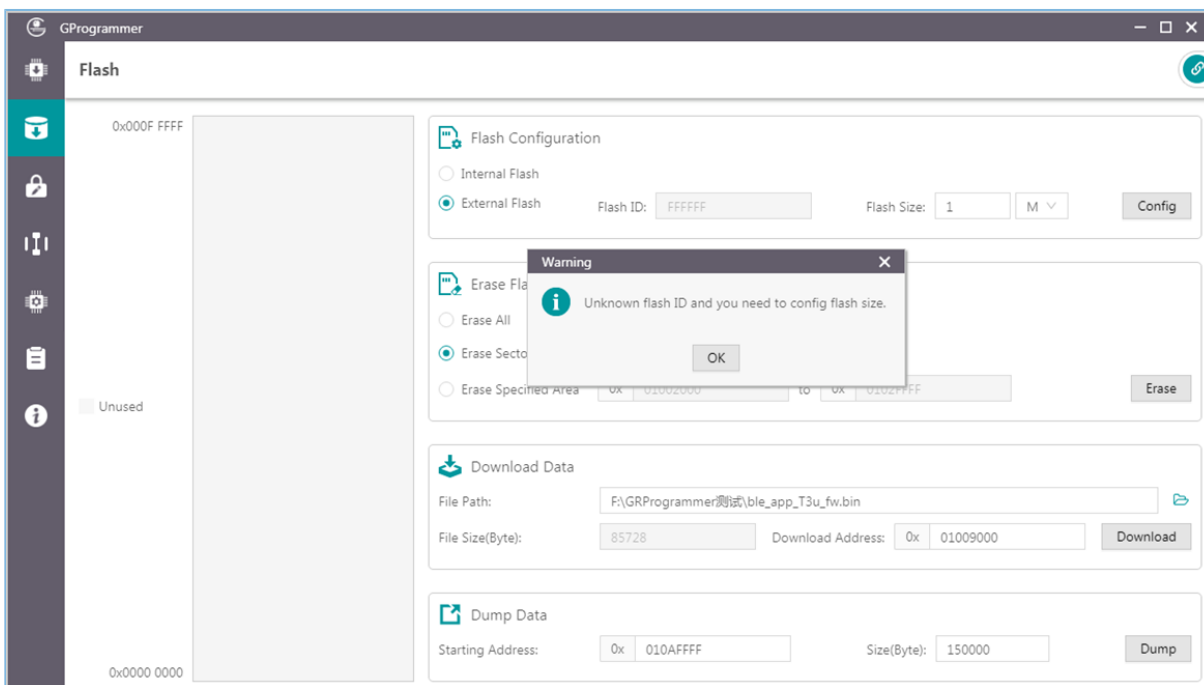


图 3-23 未知Flash

3.3.4.2.2 Flash操作

用户操作外部Flash（擦除Flash、下载数据、Dump数据）时，只要在有效地址区域之内，可任意操作，无特殊限制。

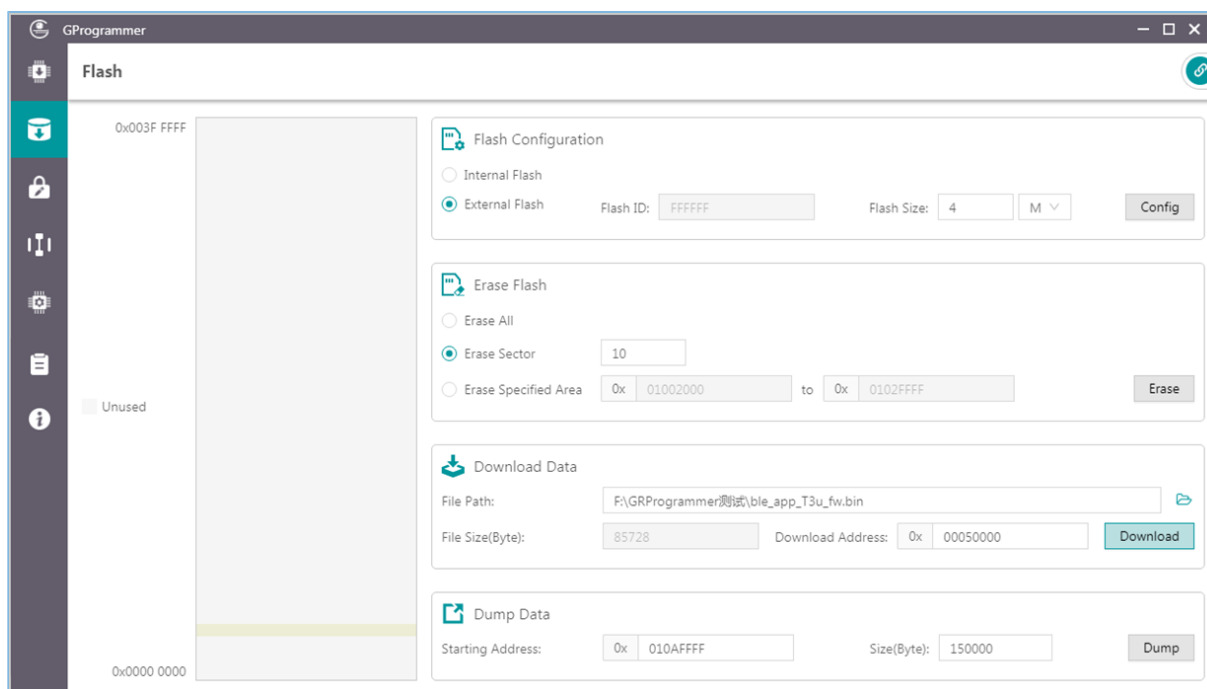


图 3-24 下载数据

说明:

完成PIN脚配置前，无法进行任何针对外部Flash的操作。

3.3.5 加密加签

点击左侧功能导航栏的图标 ，进入“Encrypt & Sign”（加密加签）页面。

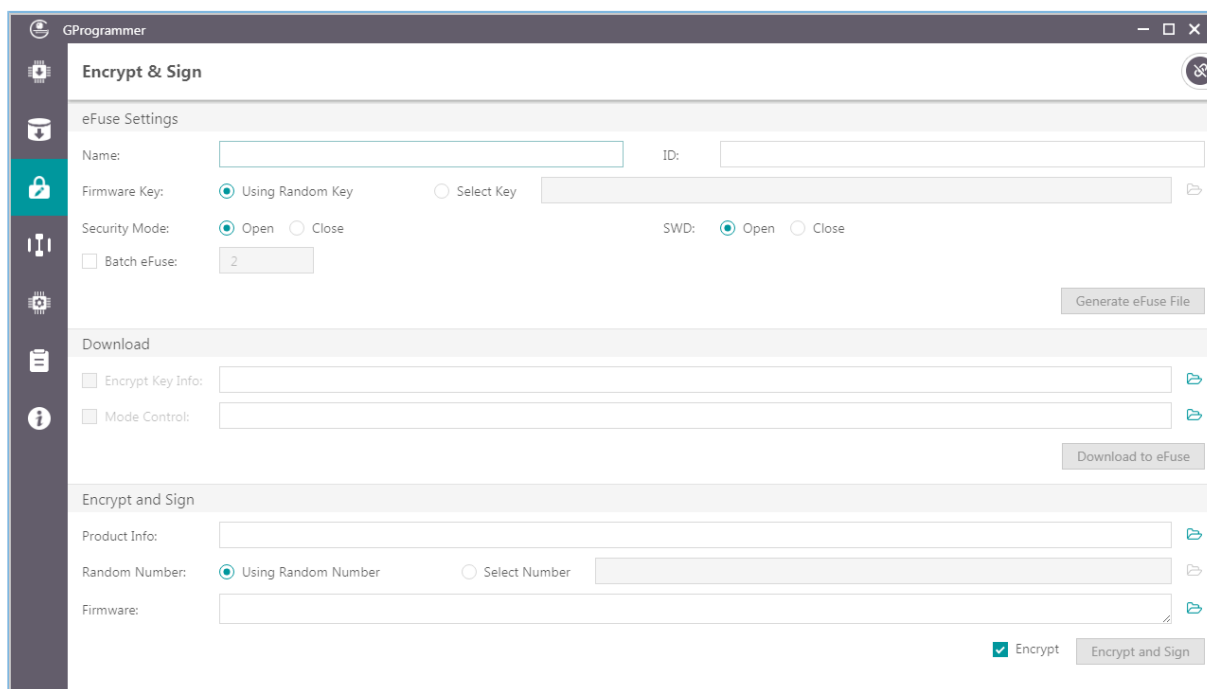


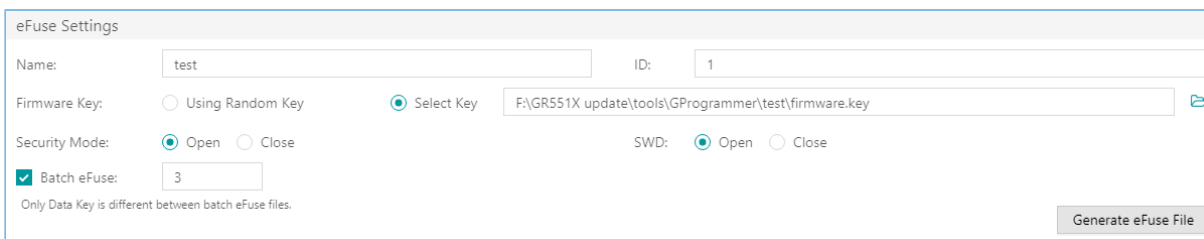
图 3-25 “Encrypt & Sign” 页面

GR551x系列芯片支持安全模式和非安全模式，由写入eFuse的产品信息中的Security Mode决定。在安全模式开启的情况下，固件需加密加签后才可下载至Flash运行。

3.3.5.1 eFuse设置

eFuse是GR551x芯片内部的一个具有随机访问接口的一次性可编程存储器。eFuse中存放着产品配置信息、安全模式控制信息以及用于加密加签的各种密钥信息等。

用户可通过指定产品的Name、ID、Firmware Key、Security Mode和SWD接口生成存储在eFuse中的相关文件。



The image shows the 'eFuse Settings' dialog box. It contains the following fields and options:

- Name: test
- ID: 1
- Firmware Key: ☐ Using Random Key, ☒ Select Key (with a file path: F:\GR551X update\tools\GProgrammer\test\firmware.key)
- Security Mode: ☒ Open, ☐ Close
- SWD: ☒ Open, ☐ Close
- Batch eFuse: ☒ (with a value of 3 in the adjacent text box)
- A note: Only Data Key is different between batch eFuse files.
- A button: Generate eFuse File

图 3-26 设置参数

说明:

- Firmware Key可使用软件自动生成的随机Key，也可使用用户自己选择添加的Key文件。
- 开启安全模式后，用户可选择是否关闭SWD接口。

另外，GProgrammer还支持通过设置“Batch eFuse”选项批量生成多个*Encrypt_key_info.bin*文件，且生成的各文件具有唯一性，以保证“一机一密”。例如，勾选“Batch eFuse”选项并在文本框中设置文件数量为“3”，则会生成3个*Encrypt_key_info.bin*文件：*Encrypt_key_info.bin*，*2_Encrypt_key_info.bin*，*3_Encrypt_key_info.bin*。

生成的文件包括：

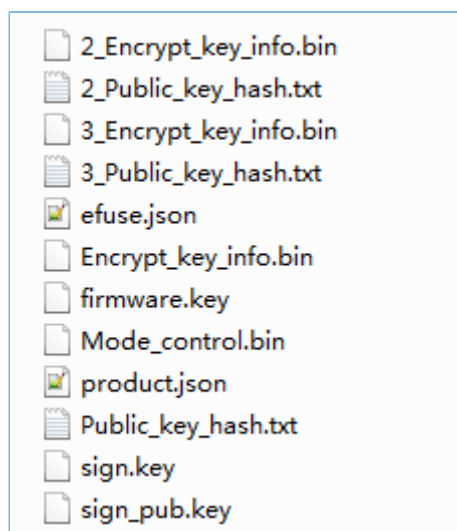


图 3-27 生成的文件

- *efuse.json*: 临时文件。

- *Encrypt_key_info.bin*、*2_Encrypt_key_info.bin*、*3_Encrypt_key_info.bin*: eFuse下载文件，包含产品及加密加签信息。此文件需下载至eFuse。
- *firmware.key*: 用于加密固件的私钥。
- *Mode_control.bin*: eFuse下载文件，包含Security Mode和SWD信息。此文件需下载至eFuse。
- *product.json*: 产品信息文件。当固件加密加签时，需导入此文件。
- *sign.key*: 用于生成签名的私钥。
- *sign_pub.key*: 用于验证签名的公钥。
- *Public_key_hash.txt*、*2_Public_key_hash.txt*、*3_Public_key_hash.txt*: 用于验证签名的公钥Hash。

为了方便用户下载文件至eFuse或加密加签固件，生成的*Encrypt_key_info.bin*和*Mode_control.bin*文件的路径将自动被添加到“Download”区域，产品信息文件*product.json*的路径将自动被添加到“Encrypt and Sign”的“Product Info”栏中，如图 3-28所示。

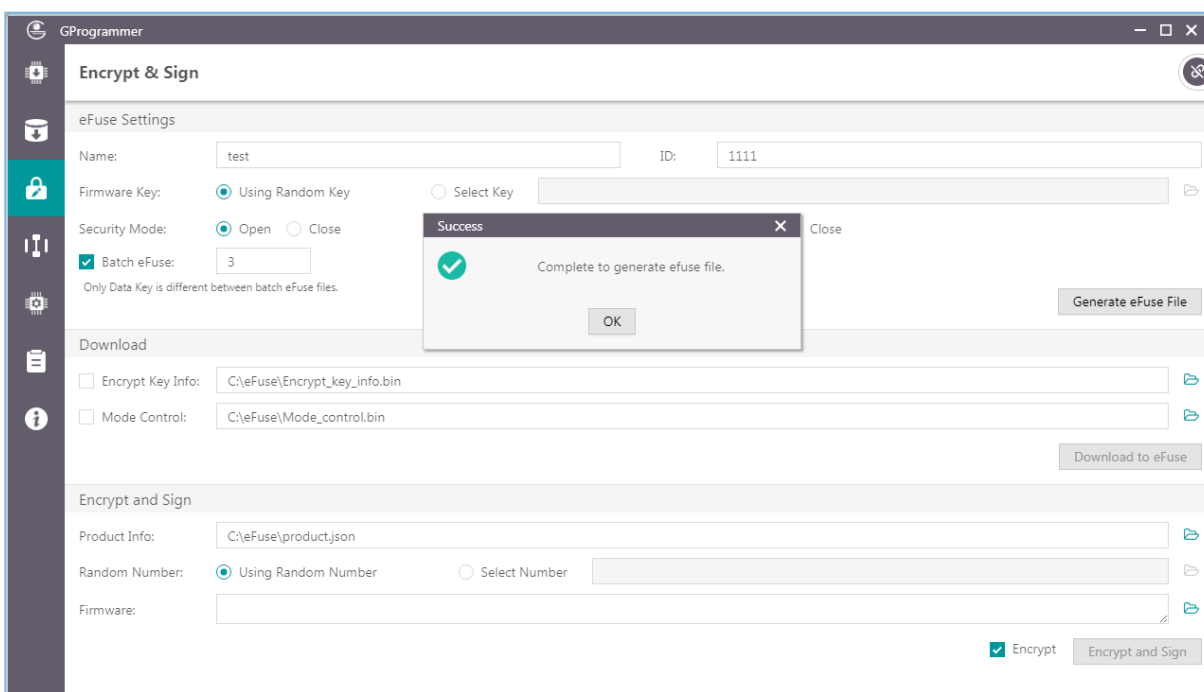


图 3-28 自动加载文件路径

3.3.5.2 eFuse下载

若用户先执行“Generate eFuse File”操作生成了*Encrypt_key_info.bin*和*Mode_control.bin*文件，则可直接勾选“Encrypt Key Info”和“Mode Control”选项，点击“Download to eFuse”按钮将文件下载至eFuse。

否则，用户需先手动选择添加*Encrypt_key_info.bin*和*Mode_control.bin*文件，然后才可下载。

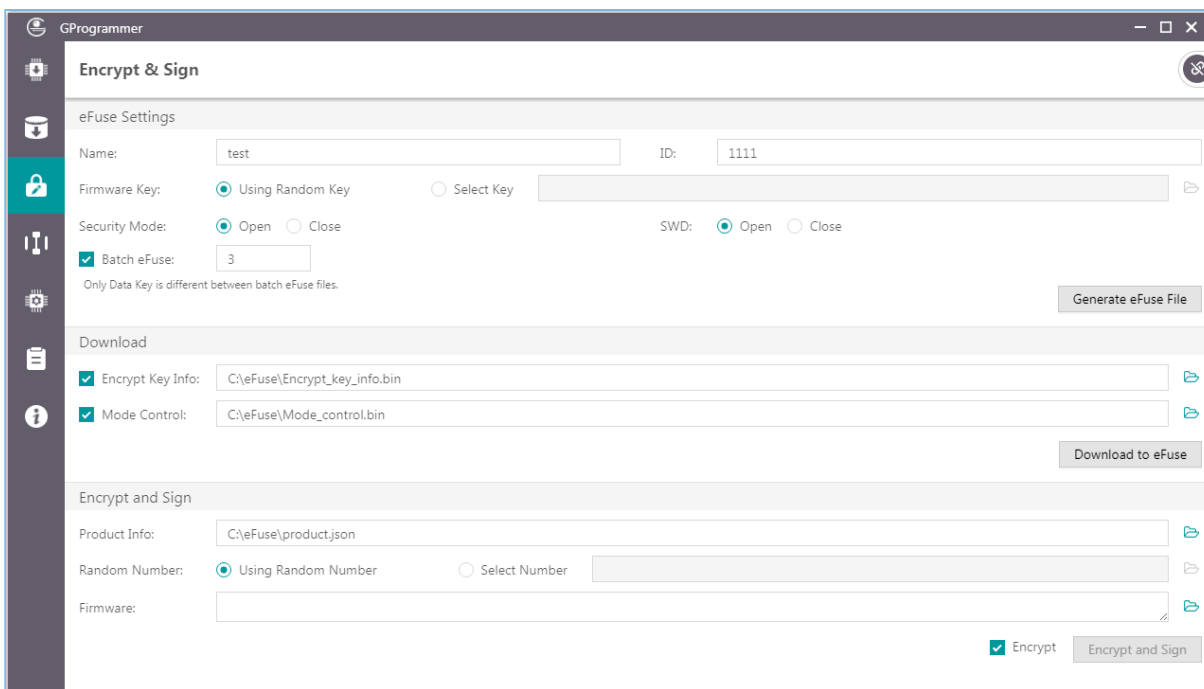


图 3-29 下载文件到eFuse

说明:

eFuse不能重复下载。

3.3.5.3 固件加密加签

在安全模式开启的情况下，固件需加密加签后才可以下载至Flash运行。GProgrammer可对固件进行加密加签或者仅加签的操作，允许用户使用同一套产品信息和同一个Random Number操作一个或者多个固件文件。

Random Number可使用用户选择添加的随机数文件，或者使用软件自动生成的随机数。

添加多个固件时，各固件的路径之间用“;”隔开，如图 3-30所示。

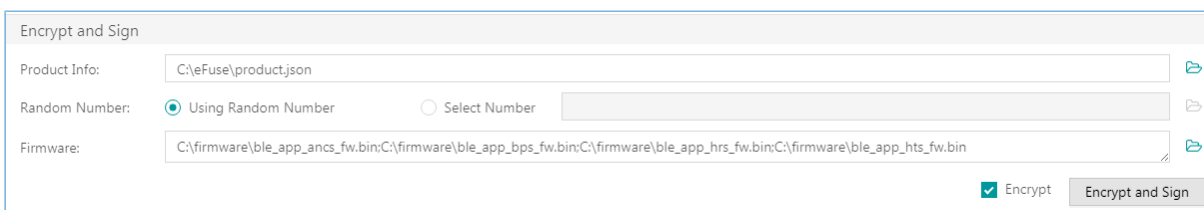
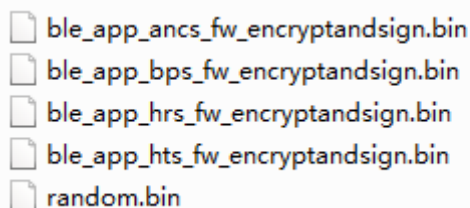


图 3-30 添加多个固件

如果勾选“Encrypt”复选框，则操作按钮变成“Encrypt and Sign”；如果不勾选“Encrypt”复选框，则操作按钮变成“Sign”。点击操作按钮，选择路径后，即可生成加密加签或者仅加签的固件。

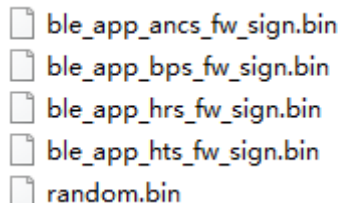
加密加签后生成的文件包括：



```
ble_app_ancs_fw_encryptandsign.bin
ble_app_bps_fw_encryptandsign.bin
ble_app_hrs_fw_encryptandsign.bin
ble_app_hts_fw_encryptandsign.bin
random.bin
```

图 3-31 加密加签生成的文件

仅加签后生成的文件包括：



```
ble_app_ancs_fw_sign.bin
ble_app_bps_fw_sign.bin
ble_app_hrs_fw_sign.bin
ble_app_hts_fw_sign.bin
random.bin
```

图 3-32 仅加签生成的文件

说明:

软件随机生成的Random Number是一个用于加密算法的随机数。执行加密加签操作后，随机数文件（*random.bin*）会与加密加签后的固件文件保存在同一目录下。若用户需再次使用该随机数加密加签，则可浏览添加之前保存的随机数文件。

3.3.6 eFuse展示

点击左侧功能导航栏的图标 ，进入“eFuse Layout”（eFuse展示）页面。

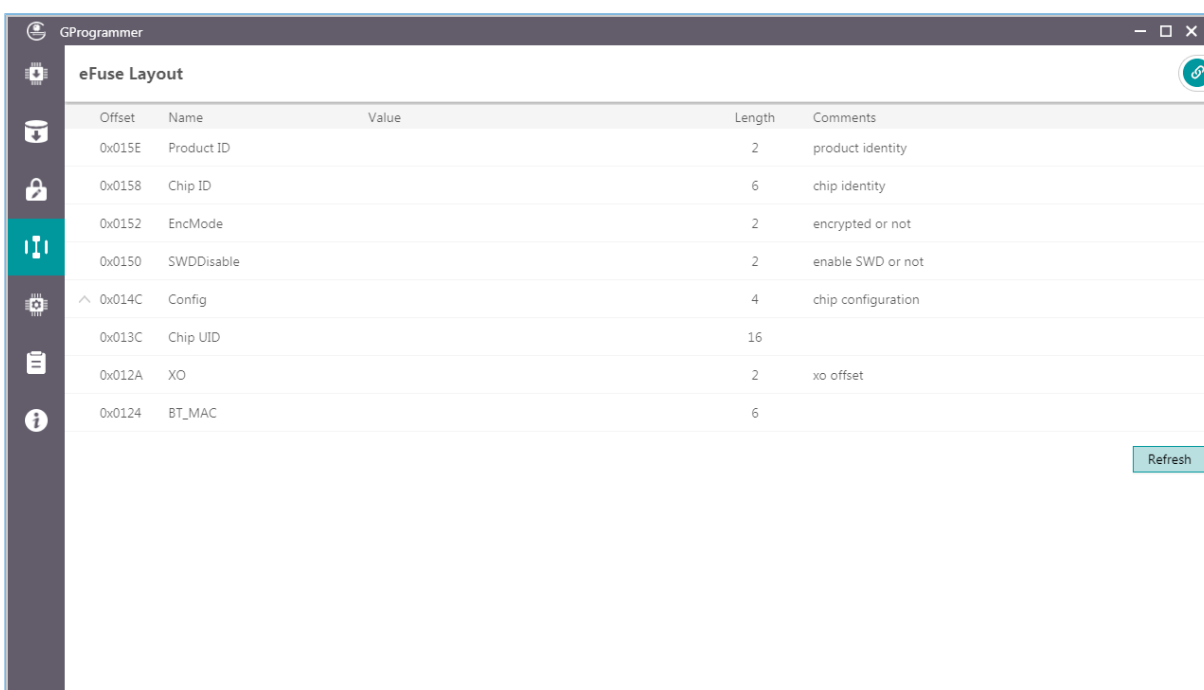
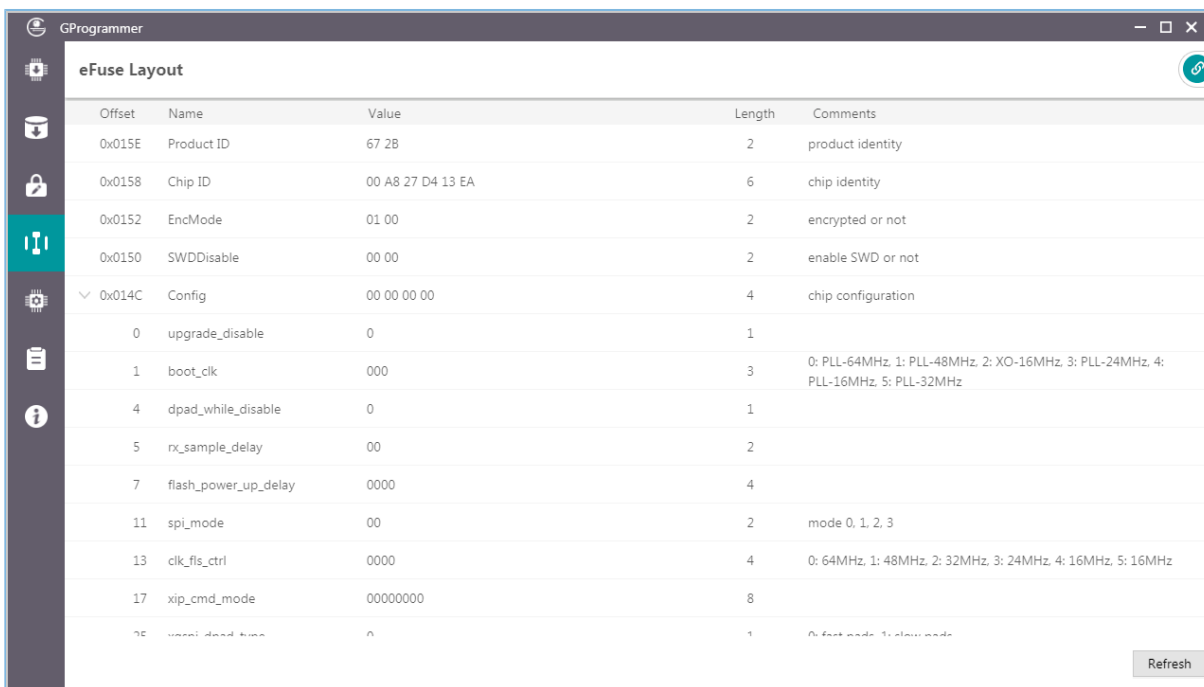


图 3-33 “eFuse Layout” 页面

用户可查看eFuse的存储布局信息，如：产品唯一标识符（“Product ID”）、芯片唯一标识符（“Chip ID”）、安全模式（“EncMode”）、SWD接口（“SWDDisable”）、配置信息（“Config”）字段的偏移地址、值、长度等信息。其中，“Config”字段还包含多个位域。

各字段/位域的值信息，需点击“Refresh”按钮读取。

点击“Config”字段的“Offset”前的 ^ 图标，可展开其位域信息，如图 3-34 所示。如需收起位域信息，请点击 v 图标或直接双击“Config”条目行。



Offset	Name	Value	Length	Comments
0x015E	Product ID	67 2B	2	product identity
0x0158	Chip ID	00 A8 27 D4 13 EA	6	chip identity
0x0152	EncMode	01 00	2	encrypted or not
0x0150	SWDDisable	00 00	2	enable SWD or not
^ 0x014C	Config	00 00 00 00	4	chip configuration
0	upgrade_disable	0	1	
1	boot_clk	000	3	0: PLL-64MHz, 1: PLL-48MHz, 2: XO-16MHz, 3: PLL-24MHz, 4: PLL-16MHz, 5: PLL-32MHz
4	dpad_while_disable	0	1	
5	rx_sample_delay	00	2	
7	flash_power_up_delay	0000	4	
11	spi_mode	00	2	mode 0, 1, 2, 3
13	clk_fls_ctrl	0000	4	0: 64MHz, 1: 48MHz, 2: 32MHz, 3: 24MHz, 4: 16MHz, 5: 16MHz
17	xip_cmd_mode	00000000	8	
1E	user_defined_data	0	1	0: fast mode, 1: slow mode

Refresh

图 3-34 展开位域

说明:

列表中展示的字段和位域信息，均来自config文件夹下的efuse_config.json文件，并非芯片eFuse中存储的全部信息。

3.3.7 芯片配置

点击左侧功能导航栏的图标 ，进入“Chip Configuration”（芯片配置）页面。

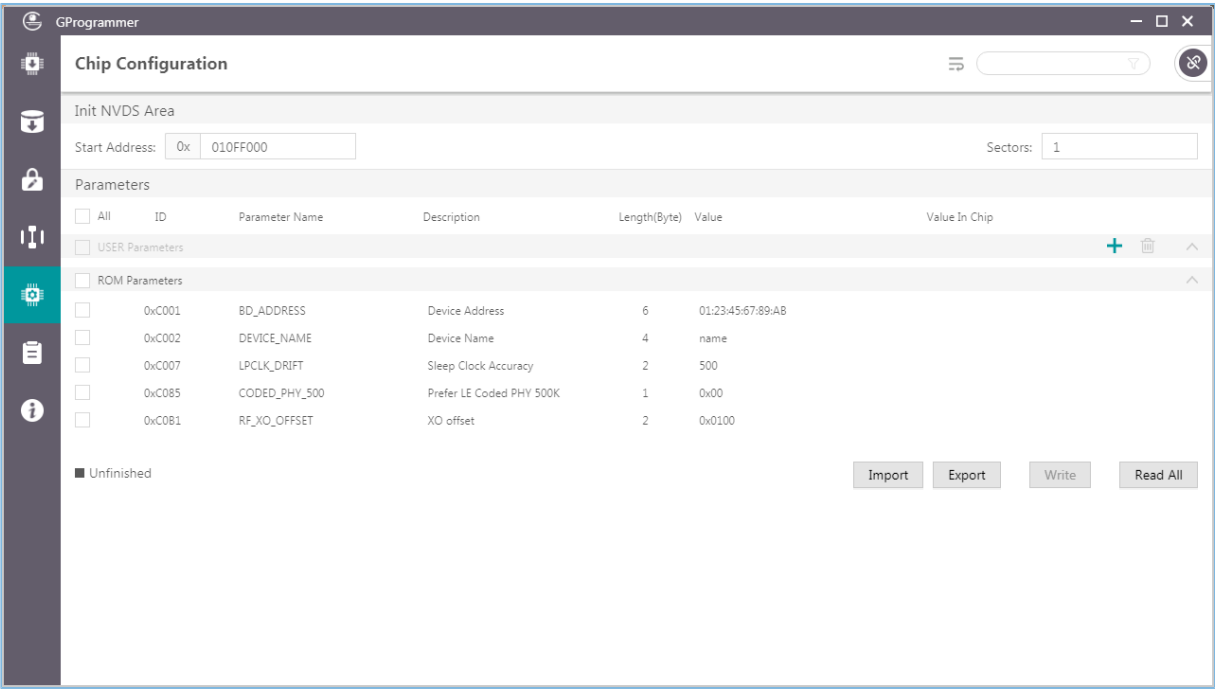


图 3-35 “Chip Configuration” 页面

用户可配置存储在芯片NVDS区域的参数，包括USER参数和ROM参数。

- **USER参数：**用户自定义的参数，允许用户新增、删除以及修改参数。
- **ROM参数：**芯片ROM中的参数，只允许用户修改参数值，不能新增或删除。

 说明:

- 列表中默认显示的ROM参数，均来自config文件夹下的nvds_config.json文件，并非从芯片NVDS中实时读取。ROM参数的详细说明，参见表 3-5。
- 点击界面右上方的  按钮，可打开“自动换行”功能，列表将换行显示。
- 利用界面右上方的筛选功能，可快速查找参数。

表 3-5 NVDS ROM参数

ID	参数名称	描述
0xC001	BD_ADDRESS	设置蓝牙设备地址。
0xC002	DEVICE_NAME	设置设备名称。
0xC007	LPCLK_DRIFT	设置系统睡眠时钟精度等级SCA，取值范围为10 ~ 500，单位：ppm。
0xC085	CODED_PHY_500	设置默认Coded PHY，0为125 kbps，1为500 kbps。
0xC0B1	RF_XO_OFFSET	设置时钟校准参数，范围为0x000 ~ 0x1FF。

3.3.7.1 初始化NVDS

配置NVDS参数之前，用户需指定NVDS区域的起始地址（以4 KB大小对齐）和占用的Sector数量。

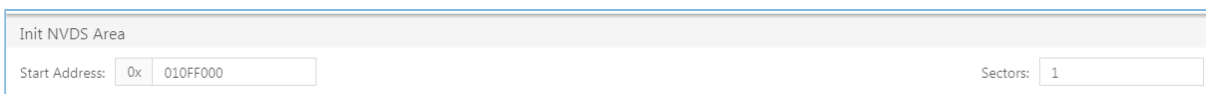


图 3-36 设置NVDS起始地址

若配置的NVDS区域与已存在的Firmware区域重叠，将无法完成NVDS初始化。

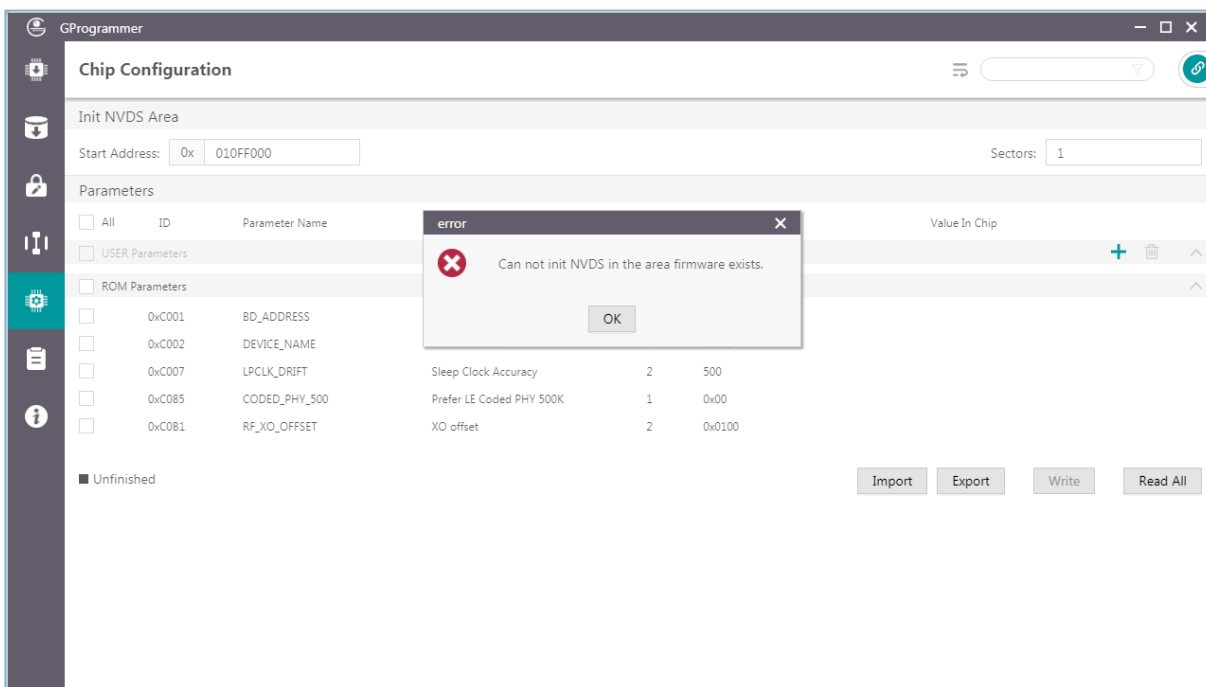


图 3-37 NVDS区域与固件区域重叠

3.3.7.2 读取参数

GProgrammer支持一次性读取当前芯片NVDS中的所有参数，并将读取的参数信息在“Parameters”列表中展示。

为了防止将芯片NVDS中已存在的参数错误覆盖，导致User APP无法正常运行，建议用户完成连接操作后，首先执行“Read All”操作，以确认参数是否已存在NVDS中。

GProgrammer以“Unfinished”、“Same”、“Different”表示参数状态，提示用户列表中的参数是否已存在NVDS中。

- “Unfinished”状态：字体为黑色，表示从NVDS中读取到列表中没有的参数（如图 3-38中ID为0x4000的参数），或者NVDS中已有此参数但长度不相等（如图 3-38中ID为0x4001的参数）。
- “Same”状态：字体为绿色，表示NVDS中有已有此参数，并且长度相等、值相同（如图 3-38中ID为0x4002的参数）。
- “Different”状态：字体为橙色，表示NVDS中已有此参数，长度相等但值不相同（如图 3-38中ID为0x4003的参数）。

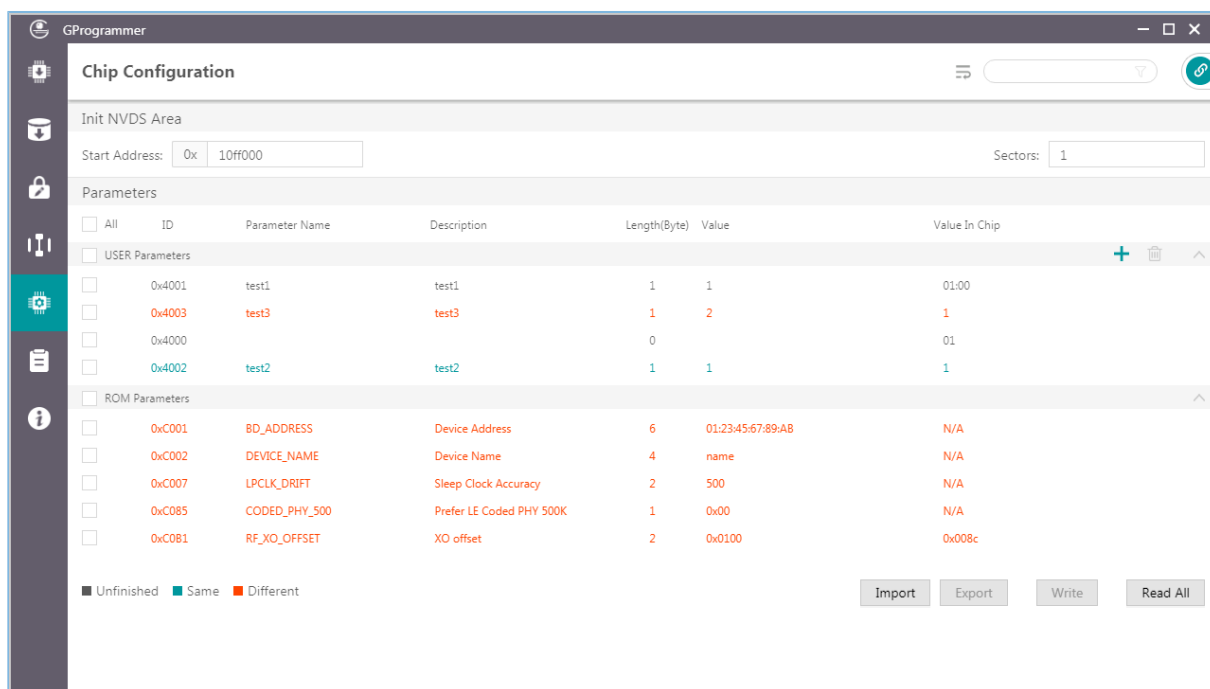


图 3-38 读取参数

3.3.7.3 写入参数

写入参数时，需先勾选待写入的参数项，然后点击“Write”按钮，将参数值写入NVDS。

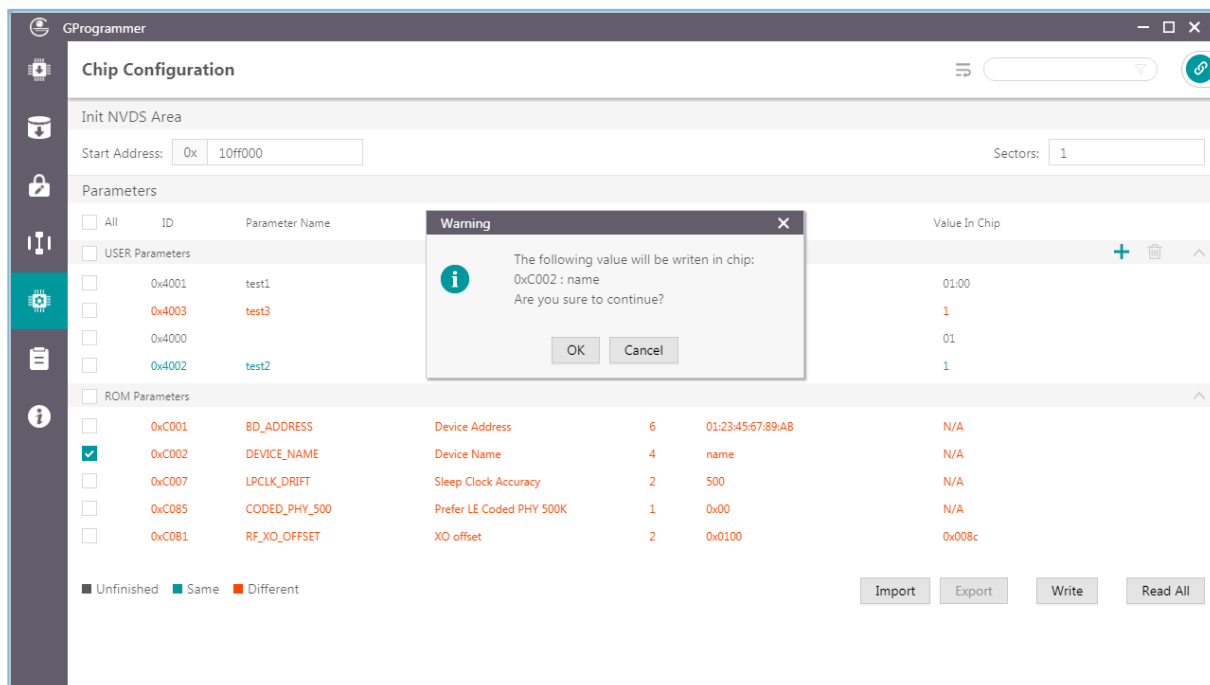


图 3-39 写入参数

说明:

- 若参数状态为“Unfinished”，则不允许直接写入芯片。
- 勾选多个参数，然后点击“Write”按钮，可实现参数的批量写入。
- 若勾选的多个参数中包含“Unfinished”状态的参数，则“Write”按钮将被禁用，不允许写入。

3.3.7.4 新增USER参数

新增USER参数时，请按以下步骤操作：

1. 点击 **+** 按钮，打开新增USER参数窗口。
2. 指定参数ID、名称、描述、类型、长度、值以及数据显示格式。

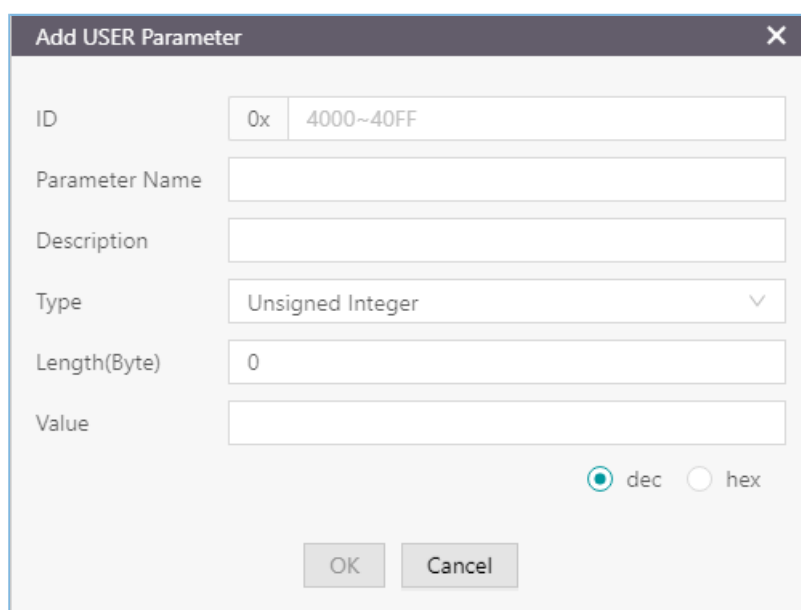
The image shows a dialog box titled "Add USER Parameter". It contains several input fields: "ID" with a dropdown set to "0x" and a text field containing "4000~40FF"; "Parameter Name" (empty); "Description" (empty); "Type" (dropdown set to "Unsigned Integer"); "Length(Byte)" (text field containing "0"); and "Value" (empty). At the bottom right, there are two radio buttons for "dec" (selected) and "hex". At the bottom center, there are "OK" and "Cancel" buttons.

图 3-40 新增USER参数

3. 点击“OK”按钮确认，即完成新增参数。

说明:

- 新增参数的ID不能与列表中已有的参数ID相同。若相同，则将弹出警告提示信息，如图 3-41所示。
- 如果NVDS中无相同ID的参数，新增的参数被直接写入NVDS。
- 如果NVDS中有相同ID的参数，且参数长度相同，新增的参数值可被写入NVDS。
- 如果NVDS中有相同ID的参数，但参数长度不同，新增的参数值不会被写入NVDS，且参数状态显示为“Unfinished”。需修改参数长度，才能写入NVDS。

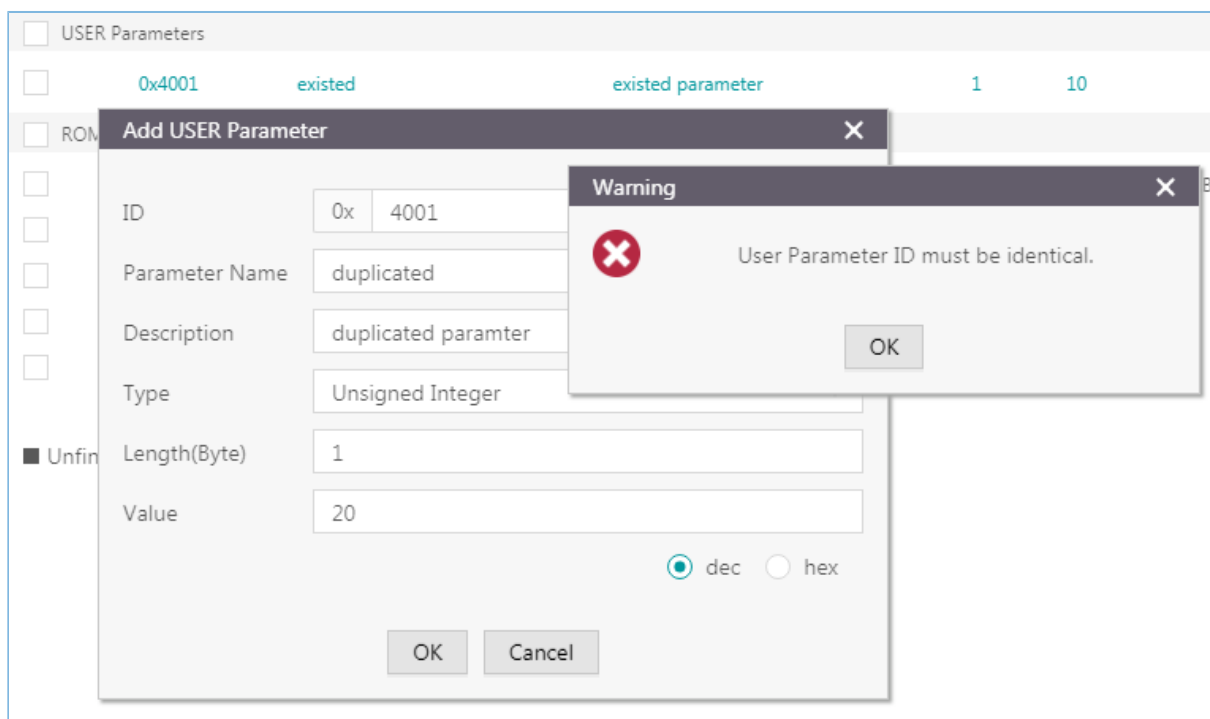


图 3-41 参数ID重复

3.3.7.5 修改参数

用户可修改NVDS中的USER参数和ROM参数。

ROM参数可修改参数名称、描述以及参数值，并且参数值的修改不会导致长度的变化（变长字符串类型除外）。

USER参数状态为“Same”或“Different”时，可修改的内容与ROM参数一致；USER参数状态为“Unfinished”时，还可修改参数类型和参数长度。

修改参数时，首先双击参数打开参数编辑窗口，编辑参数信息，然后点击“OK”按钮将修改后的参数信息写入NVDS中。

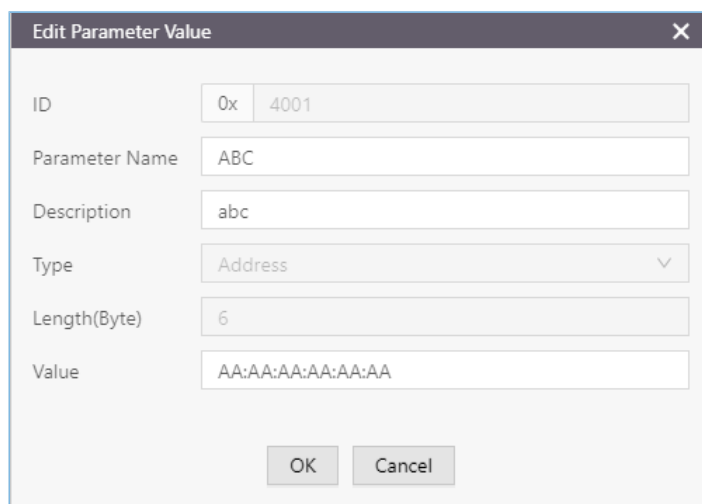


图 3-42 参数编辑窗口

说明:

若修改的参数状态为“Unfinished”，且修改后的长度与NVDS中的参数长度不一致，则执行写入操作后，参数仍将保持“Unfinished”状态，不会被写入NVDS。

3.3.7.6 删除USER参数

用户只可删除NVDS中的USER参数，不能删除ROM参数。

删除参数时，先勾选待删除参数，然后点击 **Delete** 图标，即可实现从NVDS中删除参数。

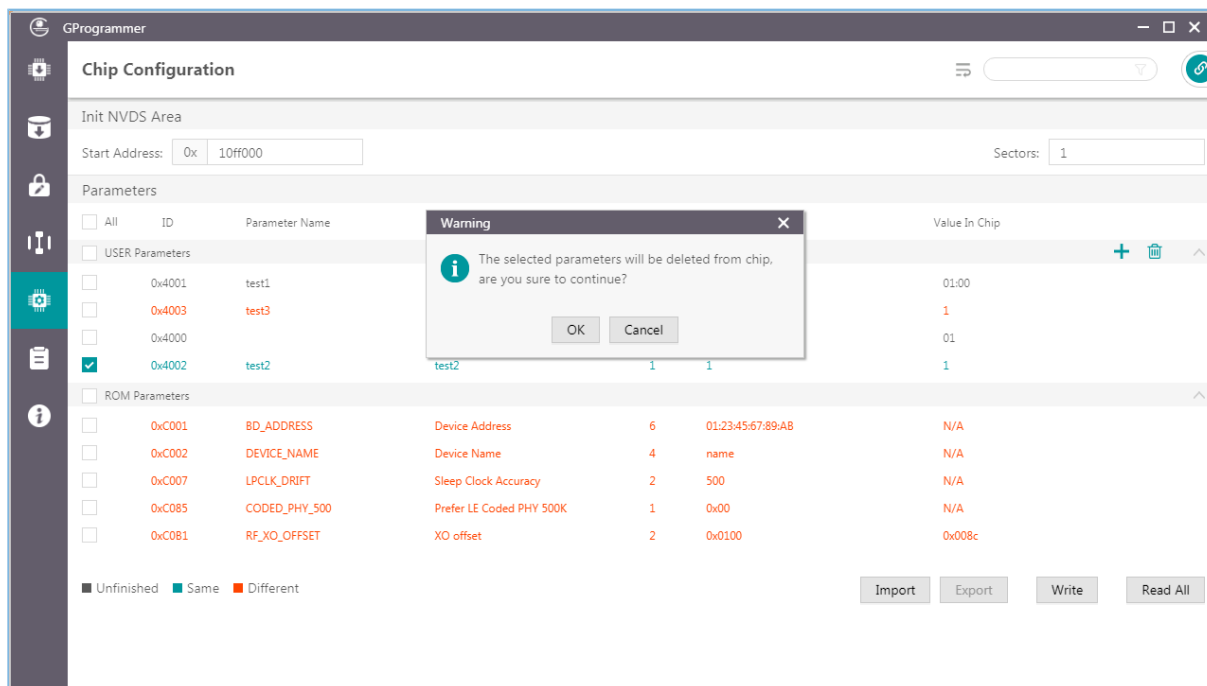


图 3-43 删除User参数

说明:

- 勾选多个User参数，然后点击 **Delete** 图标，可实现USER参数的批量删除。
- 如果勾选的多个参数中包含ROM参数，**Delete** 图标将被置灰（即“Remove”功能被禁用），不允许删除。

3.3.7.7 导入与导出

GProgrammer支持将参数列表中的数据（参数名称、描述、长度及值）导出为json格式的配置文件，以及从本地导入配置文件加载到列表显示。

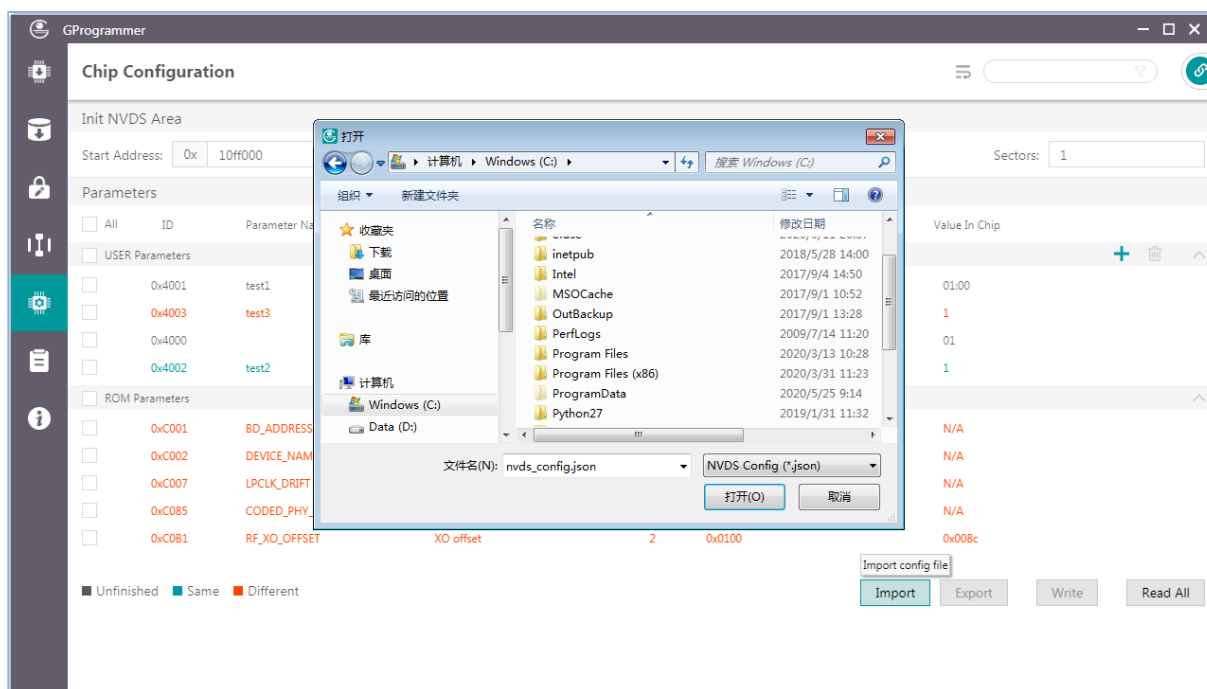


图 3-44 导入文件

说明:

- 导入配置文件后，文件中的参数会覆盖当前列表中的参数。
- 修改参数后，请及时导出配置文件并保存，以避免重复修改。

3.3.8 设备日志

点击左侧功能导航栏的图标 , 进入“Device Log”（设备日志）页面。

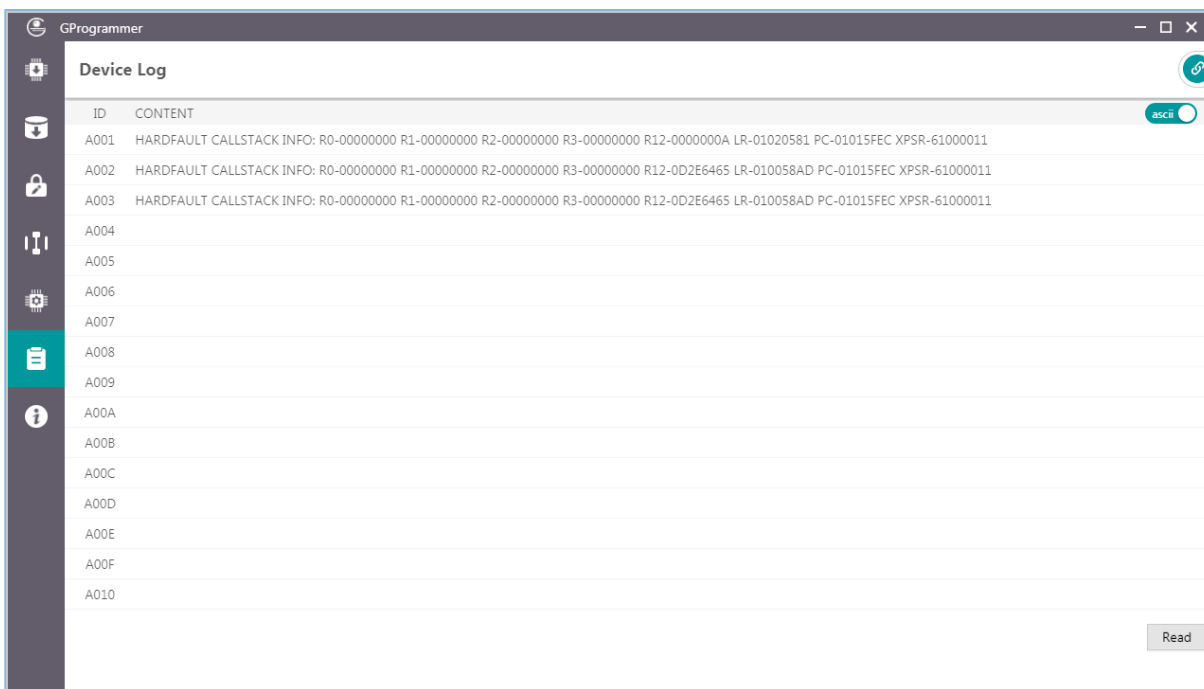


图 3-45 “Device Log” 页面

用户可查看设备日志信息，一般为芯片运行中产生的错误信息。点击“Read”按钮，即可从芯片读取设备日志信息。

另外，通过切换页面右上角的开关按钮，可选择日志信息的显示方式。

- ：开关按钮切换至“ascii”，表示以字符形式显示日志信息，如图 3-46 所示。
- ：开关按钮切换至“stream”，表示以字节流形式显示日志信息，如图 3-47 所示。

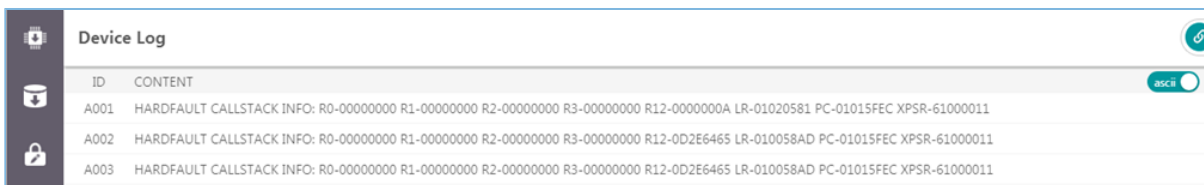


图 3-46 以字符形式显示日志

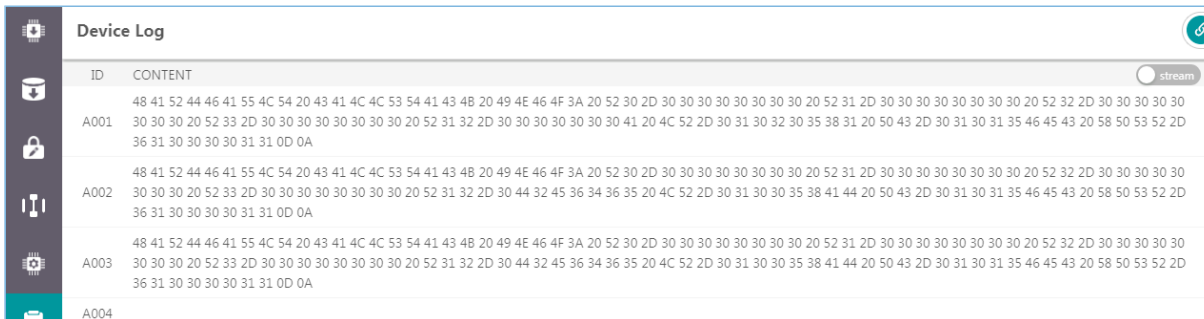


图 3-47 以字节流形式显示日志

3.3.9 命令行

GProgrammer提供2个命令程序，GR551x console.exe和GR551x encrypt signature.exe。

- `GR551x_console.exe`可方便用户直接使用命令行完成固件下载和Flash擦除等基本功能操作。
- `GR551x_encrypt_signature.exe`可方便用户直接使用命令行完成固件的加密加签或仅加签操作。

3.3.9.1 GR551x_console.exe

命令行程序`GR551x_console.exe`的使用步骤如下：

1. 选择“开始菜单 > 命令提示符”，或者在运行对话框中输入“cmd”再回车，打开“命令提示符”窗口。
2. 使用cd命令进入GProgrammer安装目录。
3. 输入`GR551x_console.exe command`命令行完成相应操作。关于“command”具体描述，参见表 3-6。

表 3-6 `GR551x_console.exe`支持的命令

command	功能描述	命令格式及参数说明
program	下载固件文件至芯片内部Flash	program <firmware file path> <run immediately:y n> 参数说明： • <firmware file path>: 待下载固件文件路径 • <run immediately:y n>: 下载完成后是否立即运行固件
erase	擦除指定地址范围内的芯片内部Flash数据	erase <start address<hex>> <end address<hex>><force erase when conflict with firmware/bootinfo/nvds:y n> 参数说明： • <start address<hex>>: 擦除区域的起始地址（16进制） • <end address<hex>>: 擦除区域的结束地址（16进制） • <force erase when conflict with firmware/bootinfo/nvds:y n>当和固件、bootinfo、nvds区域冲突之后，是否强制擦除
eraseall	擦除全部芯片内部Flash数据	eraseall
download	下载数据文件至芯片内部Flash	download <data file path> <start address<hex>><force download when conflict with firmware/bootinfo/nvds:y n>

command	功能描述	命令格式及参数说明
		参数说明： • <data file path>: 待下载数据文件路径 • <start address<hex>>: 下载区域的起始地址（16进制） • <force download when conflict with firmware/bootinfo/nvds:y n>: 当和固件、bootinfo、nvds区域冲突之后，是否强制下载
wrotefuse	将Encrypt Key Info和Mode Control文件写入eFuse	wrotefuse <Encrypt Key Info file Path> <Mode Control file Path> 参数说明： • <Encrypt Key Info file Path>: Encrypt Key Info文件路径 • <Mode Control file Path>: Mode Control文件路径
reset	复位芯片系统	reset
help	查看所有命令帮助信息	help

以使用program命令下载一个固件文件，并设置下载完成后立即运行该固件为例，输入的命令为：

```
GR551x_console.exe program "C:\ble_app_hrs_fw.bin" y
```

program命令执行过程中会实时显示下载进度，如图 3-48所示。

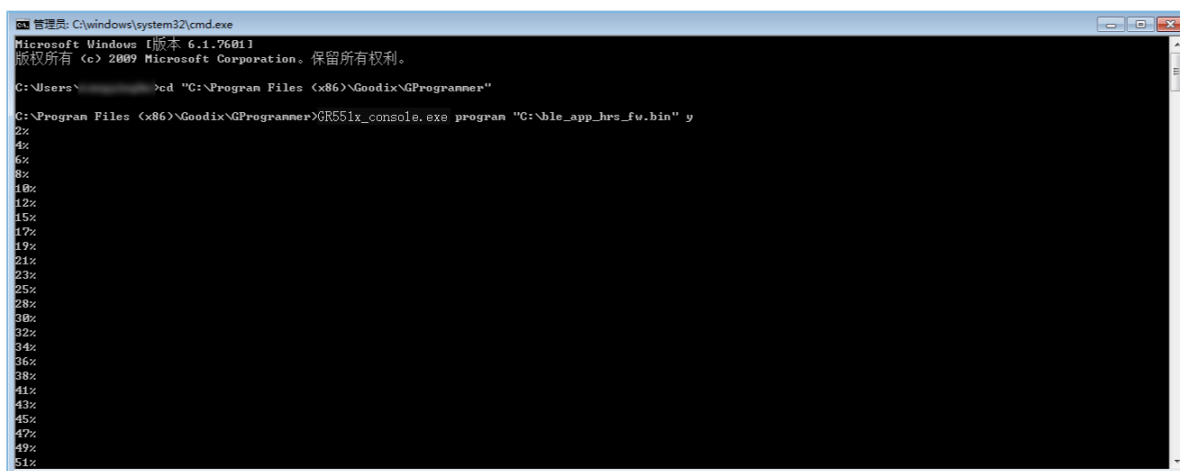


图 3-48 执行program命令过程

说明：

GProgrammer命令行程序GR551x_console.exe和界面程序无法同时使用。

3.3.9.2 GR551x_encrypt_signature.exe

命令行程序GR551x_encrypt_signature.exe的使用步骤如下：

1. 选择“开始菜单 > 命令提示符”，或者在运行对话框中输入“cmd”后回车，打开“命令提示符”窗口。
2. 使用cd命令进入GProgrammer安装目录。
3. 输入GR551x_encrypt_signature.exe --parameter完成操作。关于“parameter”具体描述，参见表 3-7。表格中只列出常用的参数，全部参数可使用GR551x_encrypt_signature.exe --help命令进行查看。

表 3-7 GR551x_encrypt_signature.exe的常用参数列表

参数	描述
operation	加密加签或者仅加签操作，参数值可选项为encryptandsign或者sign
firmware_key	用于加密加签或者仅加签的firmware.key的路径
signature_key	用于加密加签或者仅加签的sign.key的路径
signature_pub_key	用于加密加签或者仅加签的sign_pub.key的路径
product_json_path	用于加密加签或者仅加签的product.json的路径
rand_number	用于加密加签或者仅加签的random.bin的路径
ori_firmware	待加密加签或者仅加签的原始固件路径
output	加密加签后或者仅加签的固件保存路径
random_output	加密加签或者仅加签使用的随机数的保存路径
help	显示帮助信息

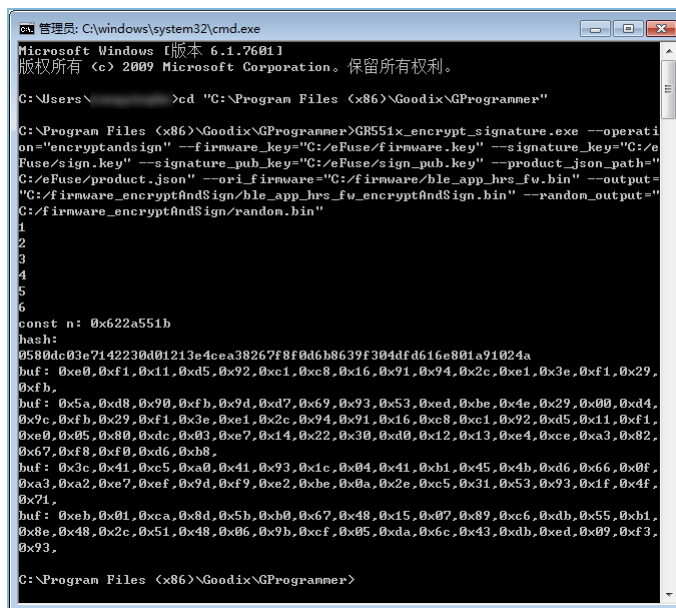
以加密加签操作为例，输入的命令为：

```
GR551x_encrypt_signature.exe
--operation="encryptandsign"
--firmware_key="C:/eFuse/firmware.key"
--signature_key="C:/eFuse/sign.key"
--signature_pub_key="C:/eFuse/sign_pub.key"
--product_json_path="C:/eFuse/product.json"
--ori_firmware="C:/firmware/ble_app_hrs_fw.bin"
--output="C:/firmware_encryptAndSign/ble_app_hrs_fw_encryptAndSign.bin"
--random_output="C:/firmware_encryptAndSign/random.bin"
```

其中参数值路径“C:/eFuse/”为3.3.5.1 eFuse设置中，点击“Generate eFuse File”按钮后用户选择的文件保存路径。

- --ori_firmware="C:/firmware/ble_app_hrs_fw.bin"：原始固件的路径。
- --output="C:/firmware_encryptAndSign/ble_app_hrs_fw_encryptAndSign.bin"：保存加密加签后固件的路径。

加密加签命令执行过程如图 3-49 所示。



```
管理员: C:\windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\>cd "C:\Program Files (x86)\Goodix\GProgrammer"

C:\Program Files (x86)\Goodix\GProgrammer>GR551x_encrypt_signature.exe --operati
on="encryptandsign" --firmware_key="C:/eFuse/firmware.key" --signature_key="C:/e
Fuse/sign.key" --signature_pub_key="C:/eFuse/sign_pub.key" --product_json_path="
C:/eFuse/product.json" --ori_firmware="C:/firmware/ble_app_hrs_fw.bin" --output=
"C:/firmware_encryptAndSign/ble_app_hrs_fw_encryptAndSign.bin" --random_output="
C:/firmware_encryptAndSign/random.bin"

1
2
3
4
5
6
const n: 0x622a551b
hash:
0580dc03e7142230d01213e4cea38267f8f0d6b8639f304dfd616e801a91024a
buf: 0xe0,0xf1,0x11,0xd5,0x72,0xc1,0xc8,0x16,0x91,0x94,0x2c,0xe1,0x3e,0xf1,0x29,
0xfb,
buf: 0x5a,0xd8,0x90,0xf0,0xd7,0x69,0x93,0x53,0xed,0xbe,0x4e,0x29,0x00,0xd4,
0x9c,0xf0,0x29,0xf1,0x3e,0xe1,0x2c,0x94,0x91,0x16,0xc8,0xc1,0x92,0xd5,0x11,0xf1,
0xe0,0x05,0x80,0xdc,0x03,0xe7,0x14,0x22,0x30,0xd0,0x12,0x13,0xe4,0xce,0xa3,0x82,
0x67,0xf8,0xf0,0xd6,0xb8,
buf: 0x3c,0x41,0xc5,0xa0,0x41,0x93,0xc1,0x04,0x41,0xb1,0x45,0x4b,0xd6,0x66,0x0f,
0xa3,0xa2,0xe7,0xef,0x9d,0xf9,0xe2,0xbe,0x0a,0x2e,0xc5,0x31,0x53,0x93,0x1f,0x4f,
0x71,
buf: 0xeb,0x01,0xca,0x8d,0x5b,0xb0,0x67,0x48,0x15,0xb7,0x89,0xc6,0xd0,0x55,0xb1,
0x8e,0x48,0x2c,0x51,0x48,0x06,0x9b,0xcf,0x05,0xda,0x6c,0x43,0xdb,0xed,0x09,0xf3,
0x93,
C:\Program Files (x86)\Goodix\GProgrammer>
```

图 3-49 执行加密加签命令过程

3.3.9.3 用户自定义Windows脚本

用户可自定义编写Windows脚本文件，来调用命令执行程序。在GProgrammer的安装目录下的GR551x_script目录中，存放了2个示例脚本文件。

*encryptAndSignatureFirmware.bat*脚本使用同级目录中*firmware_origin.bin*文件和eFuse目录，生成加密加签的固件保存到*firmware_encryptAndSign\firmware_encryptAndSign.bin*路径中。

*program_Firmware_EncryptAndSign.bat*脚本先擦除全部内部Flash，然后下载*firmware_encryptAndSign\firmware_encryptAndSign.bin*路径中的固件到内部Flash中。

3.4 GMF03x系列

本章主要介绍GMF03x系列芯片的GProgrammer功能操作。

3.4.1 主界面

选择GMF03x系列芯片后，将进入GMF03x芯片对应的软件主界面。

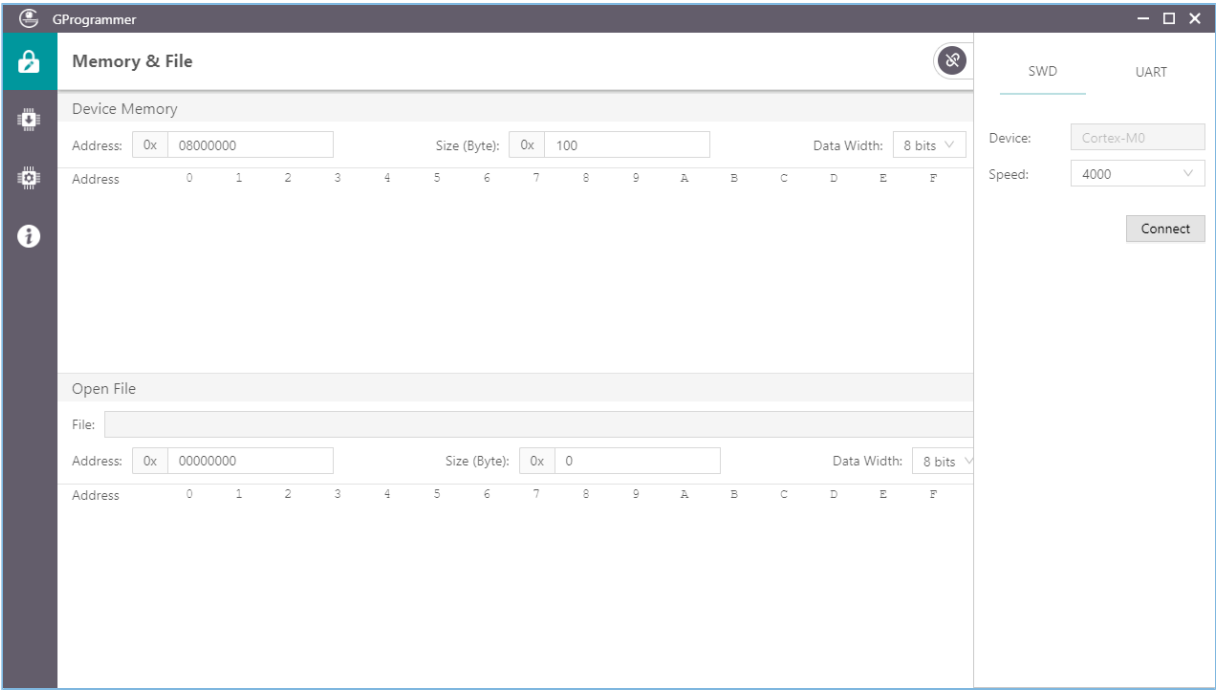


图 3-50 GProgrammer软件主界面-GMF03x系列

软件界面的左侧为功能导航栏（具体描述见[表 3-8](#)），右侧为功能操作区域。

表 3-8 功能导航栏选项

图标	功能名称	描述
	Memory & File	点击该图标，进入“内存/文件”操作页面
	Programming & Erasing	点击该图标，进入“编程/擦除”操作页面
	User Option Bytes	点击该图标，进入“用户选项字节配置”操作页面
	Help	点击该图标，进入“帮助”操作页面

3.4.2 连接管理

用户可管理控制目标板与主机之间的连接。

点击软件界面右上角的 ，可展开连接管理面板；再次点击 ，可收起并隐藏该面板。

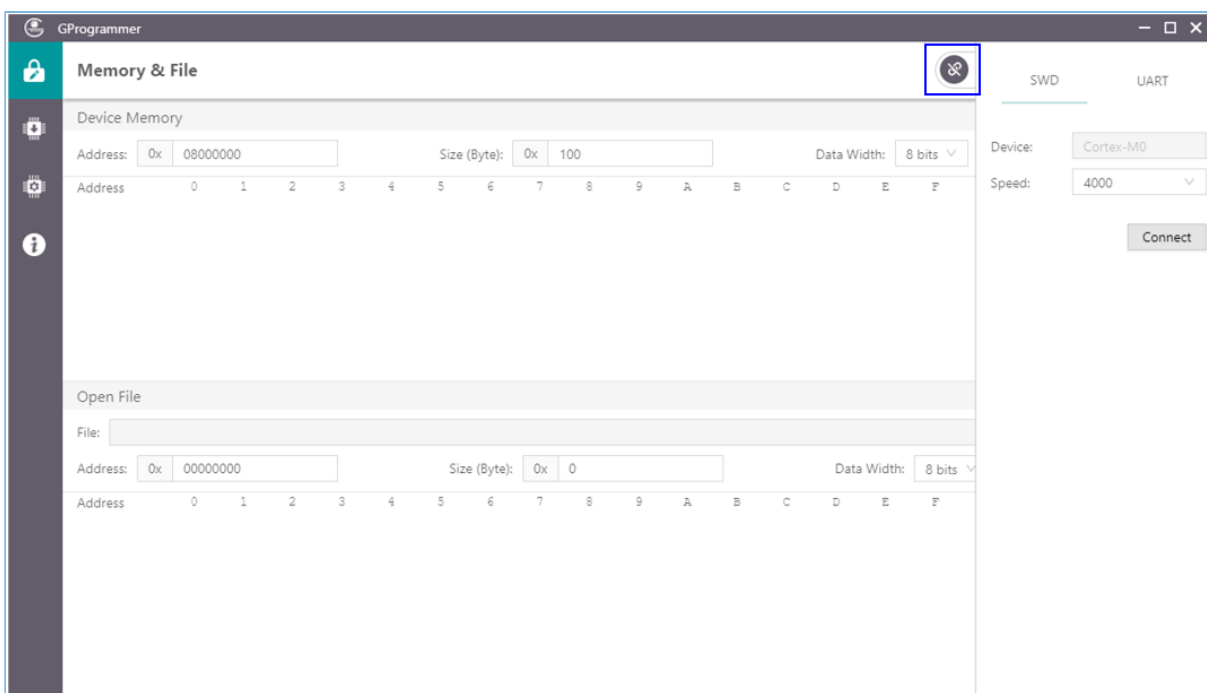


图 3-51 打开连接管理面板

GProgrammer支持SWD和串口两种连接方式。

- SWD连接

在SWD连接方式下，用户只需配置传输速率“Speed”，即可点击“Connect”按钮连接目标板。

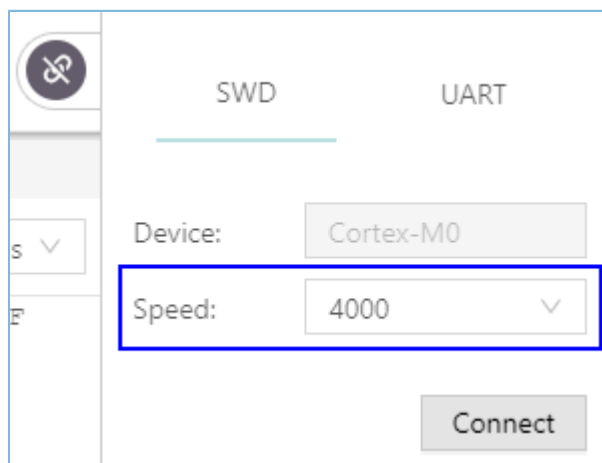


图 3-52 SWD连接

- 串口（UART）连接

在串口（UART）连接方式下，用户可根据实际情况配置串口号“Port”（需先点击“Refresh”按钮获取串口号列表，再选择正确的串口号）、波特率“BaudRate”和奇偶校验“Parity”，其他参数默认配置不可修改。

参数配置完成后，点击“Connect”按钮连接目标板。

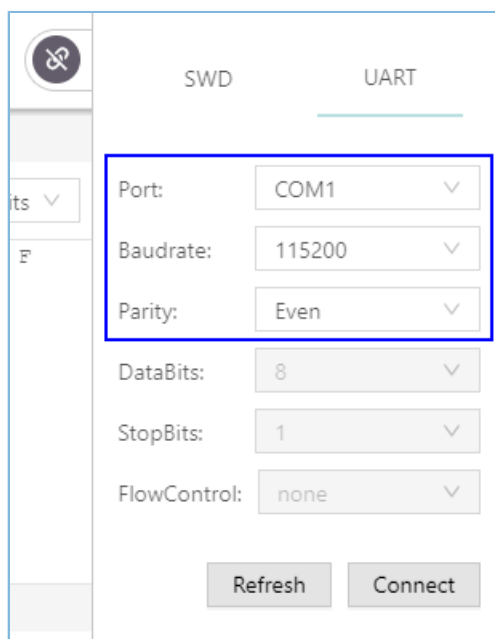


图 3-53 UART连接

目标板连接成功后，连接管理面板将自动收起并隐藏，按钮将变成，表示当前的连接状态为“已连接”。同时，将弹出提示对话框，告知实际连接的目标板的芯片型号。

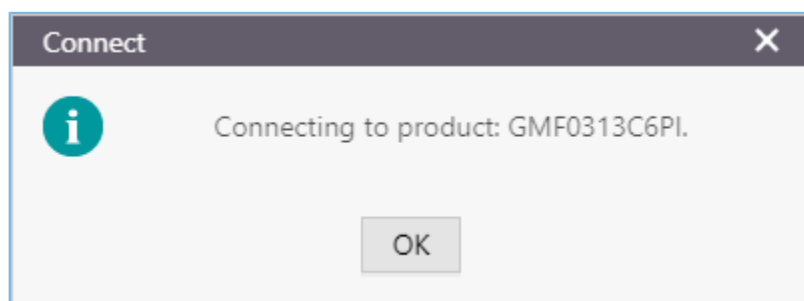


图 3-54 连接目标板的芯片型号

如需断开连接，再次点击打开连接管理面板，然后点击“Disconnect”按钮即可。

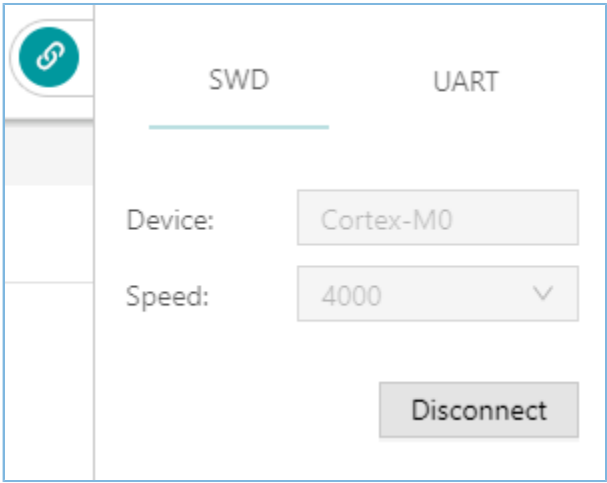


图 3-55 断开连接

3.4.3 内存/文件操作

点击左侧功能导航栏的图标, 进入 “Memory & File” （内存/文件操作）页面。

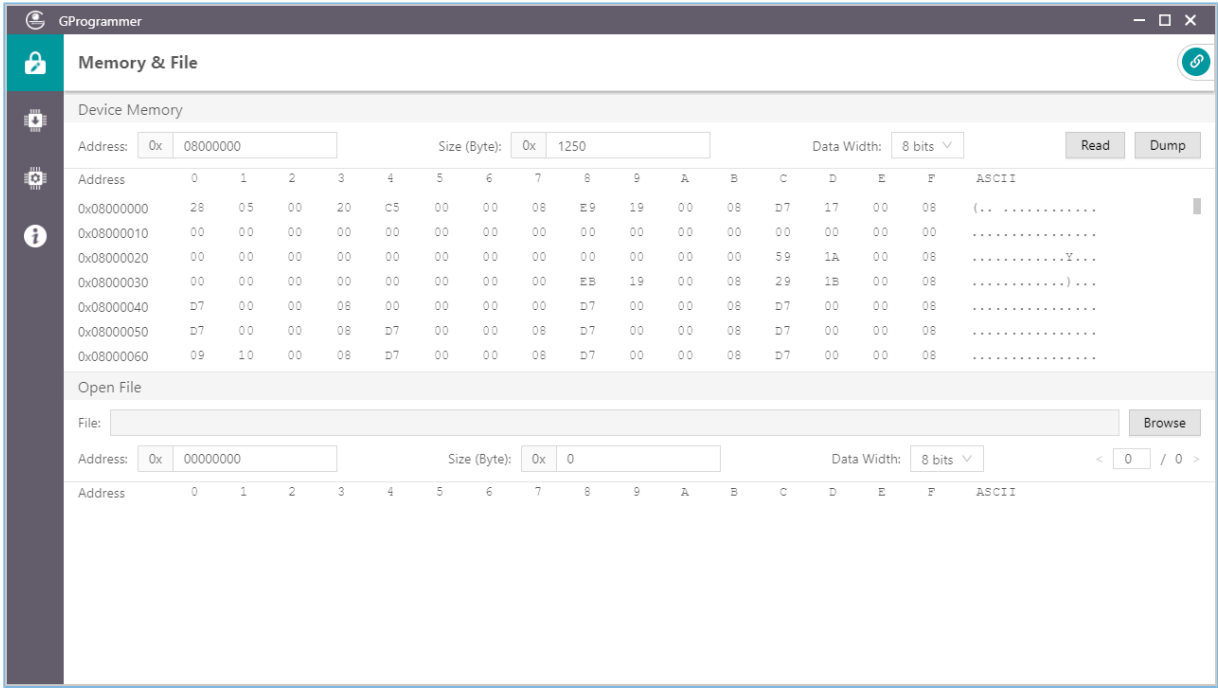


图 3-56 “Memory & File” 页面

用户可对芯片的内存进行读写操作，同时还可查看bin和hex文件内容。

3.4.3.1 读写内存

用户可对内存进行读、写操作，即从内存读取数据及向内存写入数据。

• 读内存

从内存读取数据时，请先设置读取数据的起始地址“Address”和数据长度“Size(Byte)”，再点击“Read”按钮。

若操作成功，读取的数据将在列表中显示。用户还可点击“Data Width”下拉框选择按8 bits（字节）、16 bits（半字）或32 bits（字）方式显示数据。

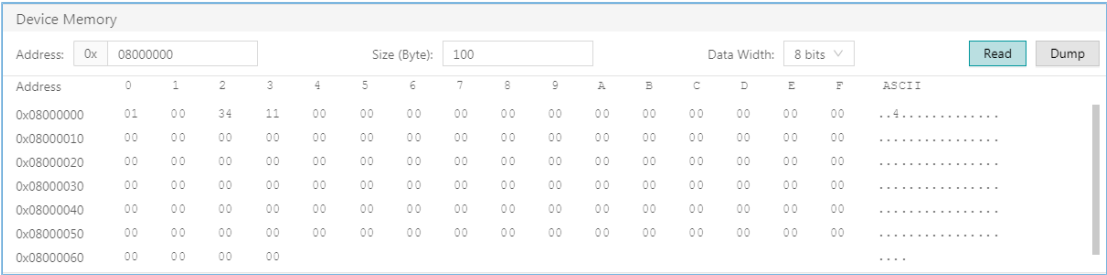


图 3-57 从内存读取数据

另外，点击“Dump”按钮，可将读取的数据保存为本地bin文件。

说明:

起始地址“Address”必须4字节对齐。

• 写内存

用户可修改内存中的数据，即向内存写入数据。

在列表中，双击待修改的数据，进入文本框编辑模式；然后重新输入数据值；最后回车确定，将数据写入内存。

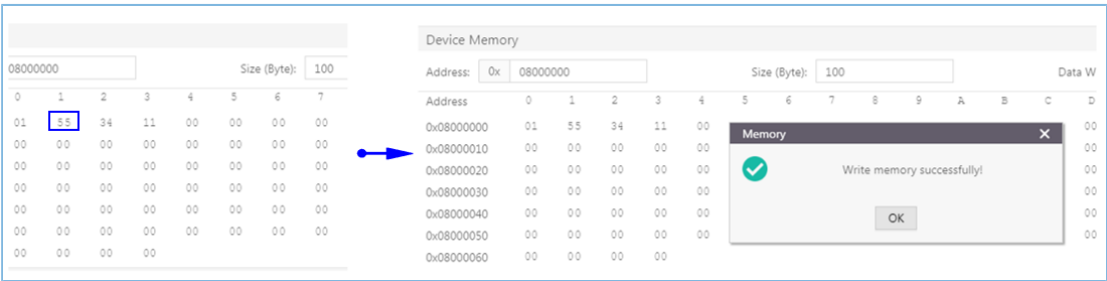


图 3-58 向内存写入数据

说明:

双击“Size (Byte)”输入框，可切换10进制与16进制的输入、显示。

3.4.3.2 展示文件

用户还可查看bin、hex文件中的数据内容。

点击“Browse”按钮，从本地添加文件；然后点击“Data Width”下拉框，选择数据的展示方式（8 bits、16 bits或32 bits）。另外，文件大小“Size(Byte)”也由软件自动计算并加以展示。

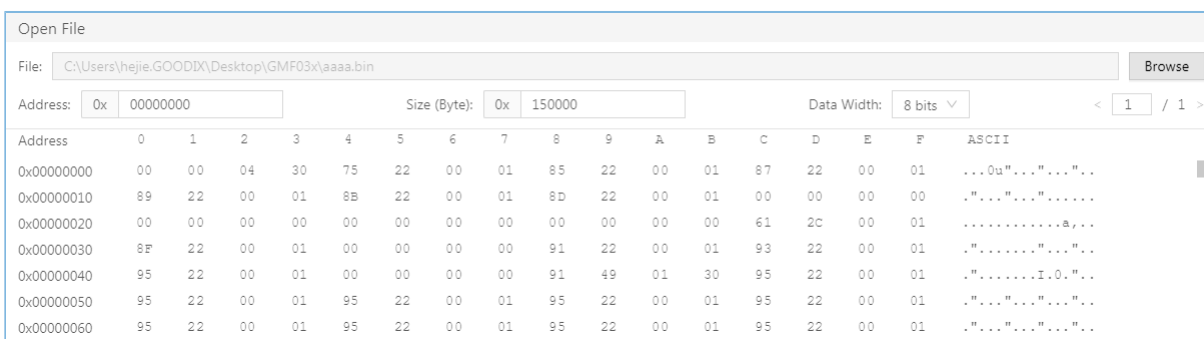


图 3-59 展示bin文件内容

如果打开的文件为hex文件，可使用“Browse”按钮下方的翻页按钮，以快速浏览hex文件各段的内容。

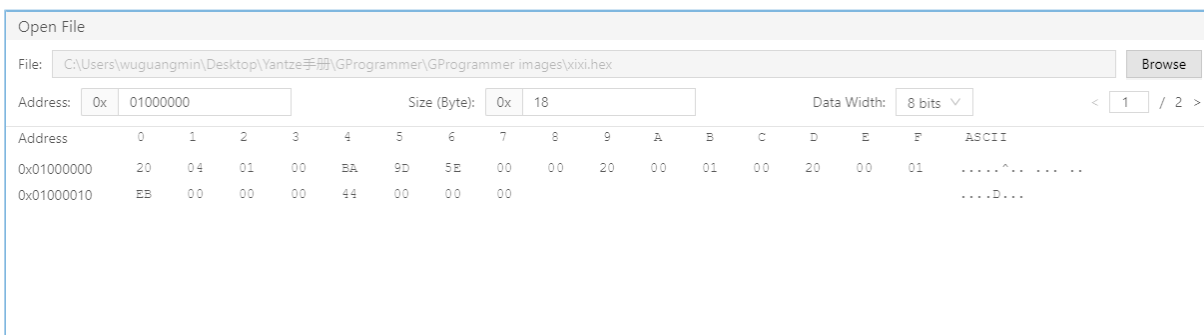


图 3-60 hex文件分页显示

3.4.4 Flash操作

点击左侧功能导航栏的图标, 进入“Programming & Erase”（Flash操作）页面。

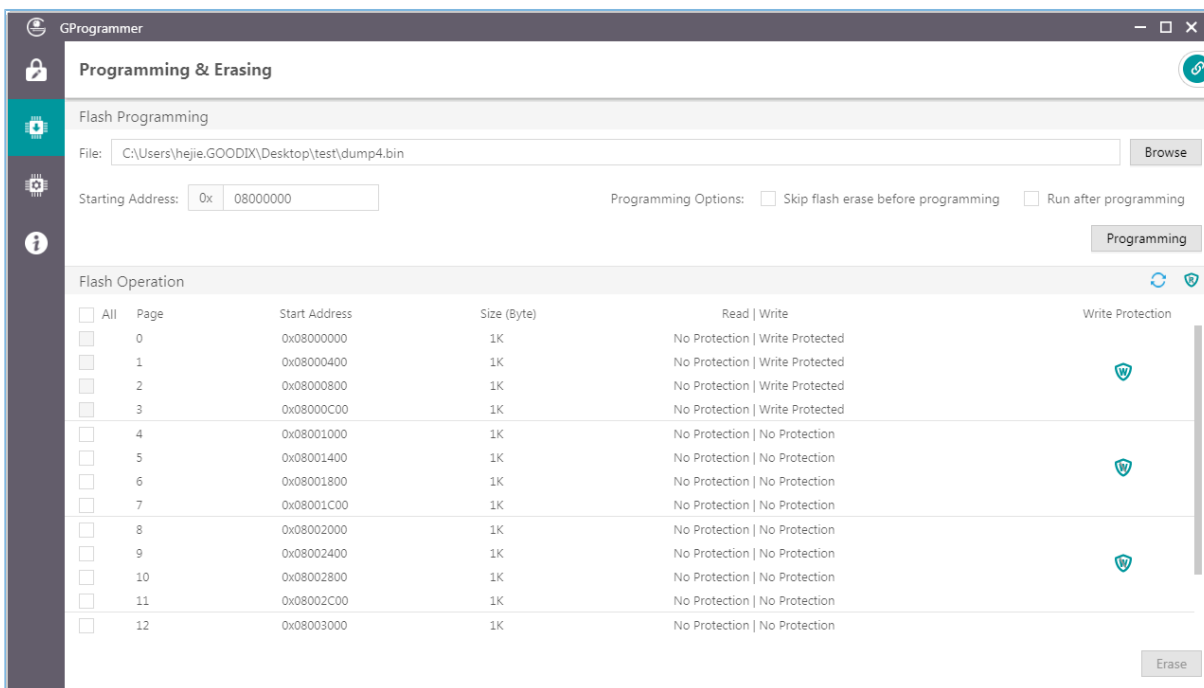


图 3-61 “Programming & Erase” 页面

用户可进行Flash编程、擦除以及读/写保护操作。

3.4.4.1 编程Flash

编程Flash的操作步骤如下：

- 1. 点击“Browse”选择待编程文件（支持bin、hex文件）。
- 2. 设置编程起始地址。
- 3. 配置编程选项（编程前是否擦除Flash、编程后是否立即运行程序）。
- 4. 点击“Programming”按钮进行Flash编程。

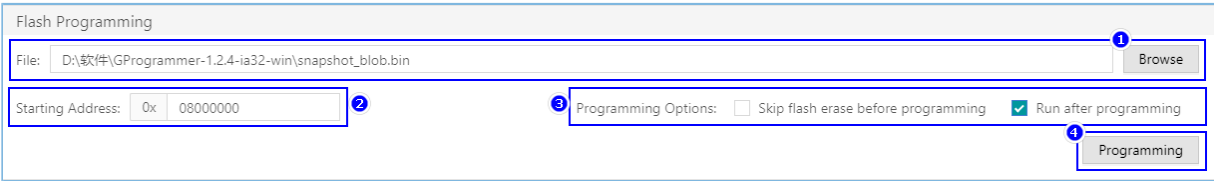


图 3-62 编程Flash

说明:

- 若下载的固件文件为hex文件，则无需设置起始地址“Starting Address”和编程配置选项“Starting Address”。
- 起始地址“Starting Address”必须4字节对齐。

3.4.4.2 擦除Flash

GProgrammer支持按页（Page）擦除Flash主存储区（Flash Memory）。操作时，需先选中待擦除的Page，再点击“Erase”按钮。

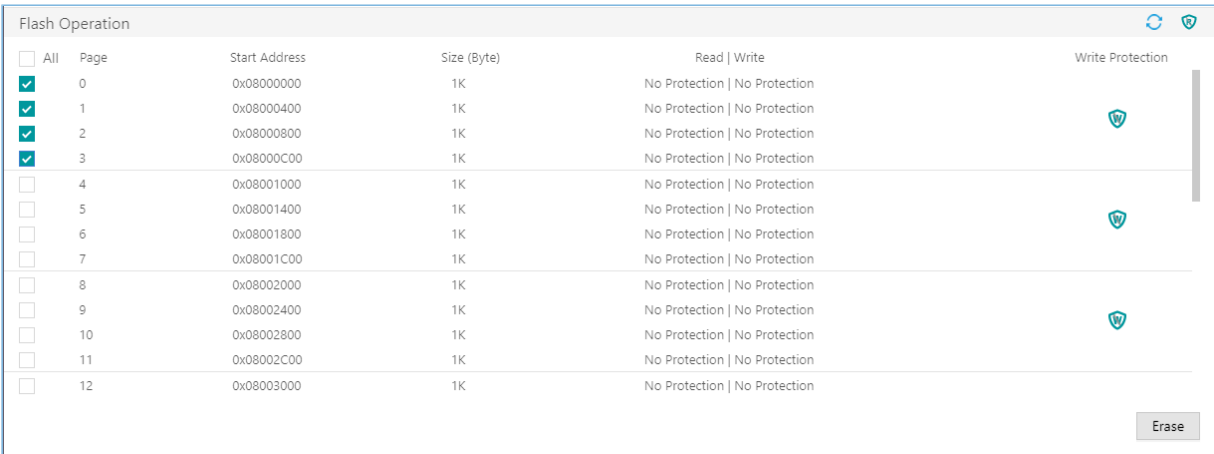


图 3-63 按页擦除

说明:

若Page处于写保护状态，则不能进行擦除操作。

另外，勾选“**All**”复选框，将一次性选中所有**Page**（除了写保护状态的**Page**），再点击“**Erase**”按钮可一次性全部擦除所有**Page**。

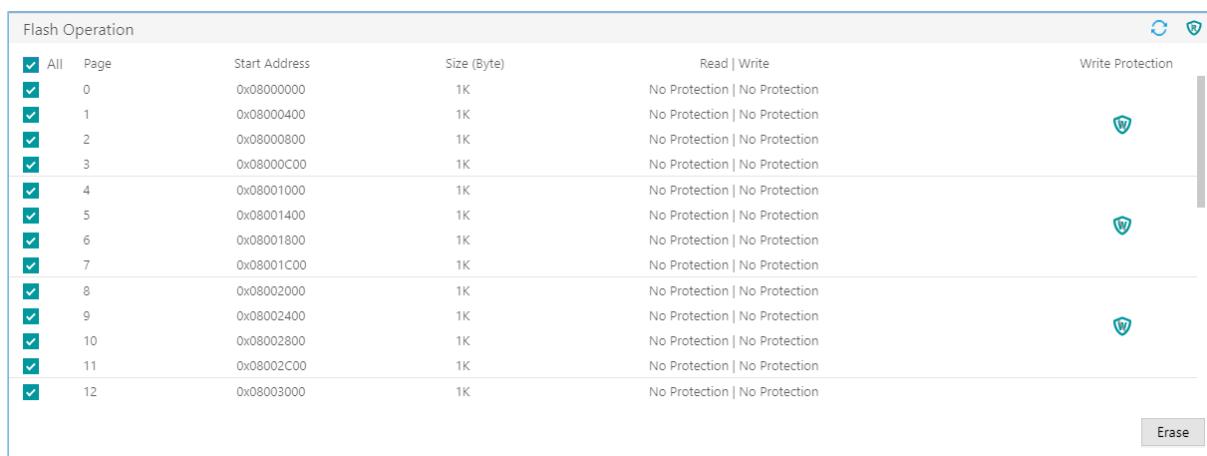


图 3-64 全部擦除

3.4.4.3 Flash读/写保护

GProgrammer支持Flash读/写保护设置。

- 读保护

设置读保护后，将不能对Flash区域进行读操作，且不同连接方式下的限制区域不同。在SWD连接方式下，将不能对Flash主存储区进行读操作；在UART连接方式下，所有Flash区域都不能进行读操作。

说明:

若芯片已设置读保护，则无法再使用UART连接方式。

点击“Flash Operation”标题栏右侧的 按钮，可设置读保护。

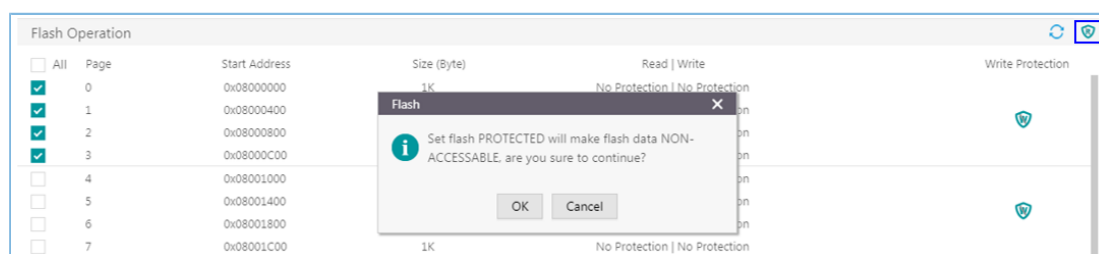


图 3-65 设置读保护

设置读保护成功后， 按钮将变成 。点击 可取消读保护，同时将擦除Flash主存储区。

- 写保护

设置写保护后，指定的Flash区域将不能进行写操作，且写保护只能以扇区（Sector，4个Page）为单位进行操作。

点击列表中“Write Protection”列的  按钮，可设置写保护。

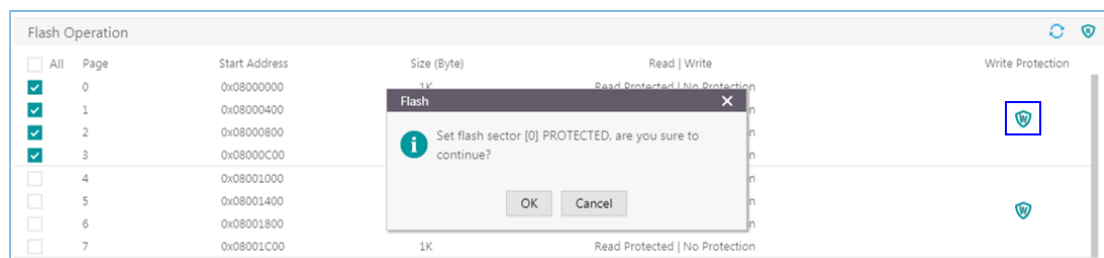


图 3-66 设置写保护

设置写保护成功后， 按钮将变成 ，同时Sector对应的各Page的复选框将不可勾选。点击  可取消写保护。

3.4.5 用户选项字节配置

点击左侧功能导航栏的图标 ，进入“User Option Bytes”（用户选项字节配置）页面。

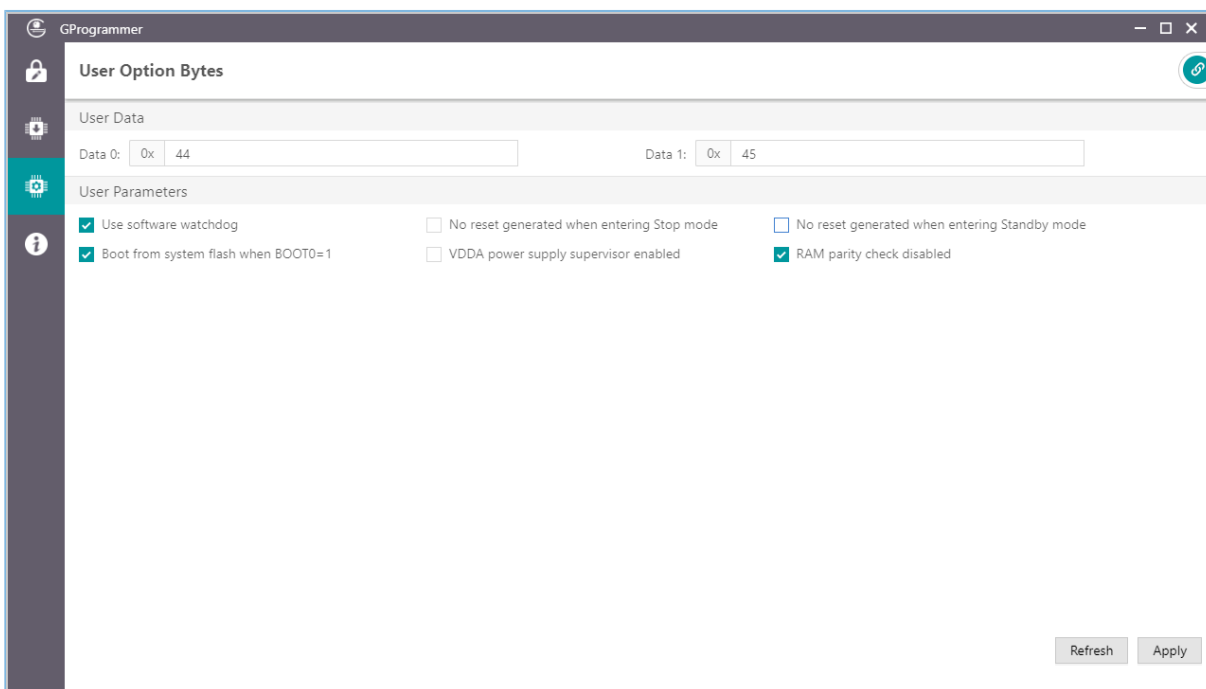


图 3-67 “User Option Bytes” 页面

用户可修改芯片中的用户选项字节（User Option Bytes）参数信息，如：是否使用软件看门狗、是否从Flash启动、RAM奇偶性校验等。

参数设置完成后，点击“Apply”按钮，将用户选项字节的配置信息下载至芯片。

另外，点击“Refresh”按钮，可获取当前芯片中的用户选项字节的配置信息。

3.5 帮助信息

点击左侧功能导航栏的图标 ，进入“Help”（帮助信息）页面。

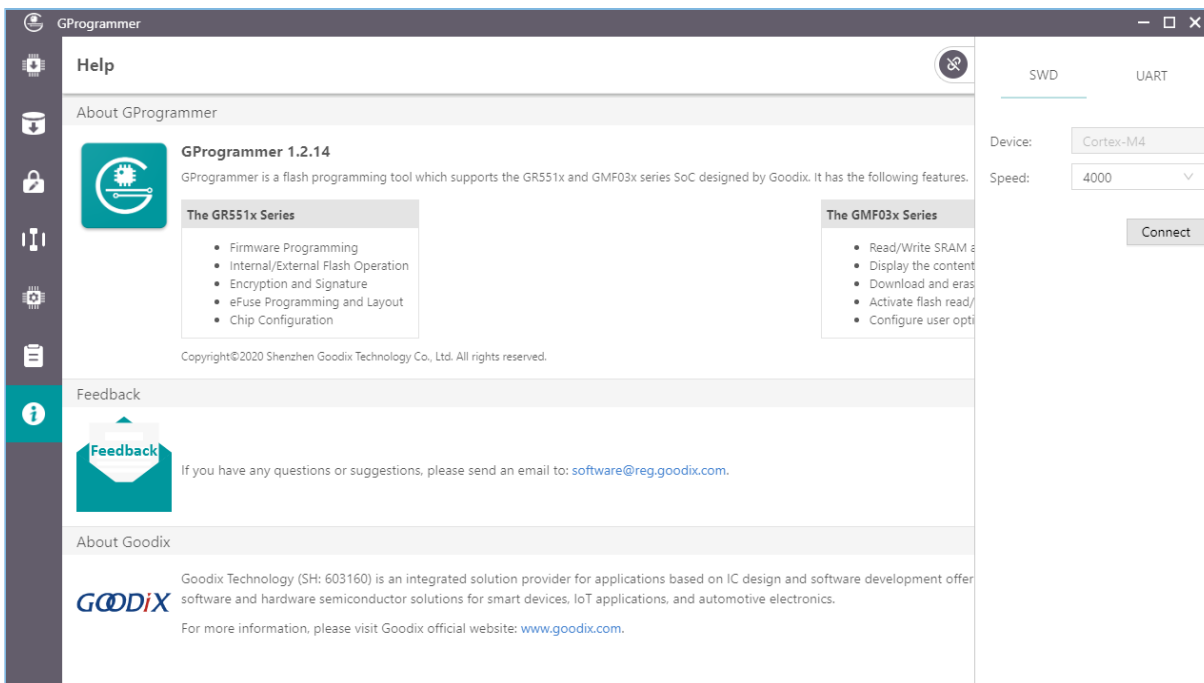


图 3-68 帮助页面

GProgrammer为用户提供的帮助支持，包括：

- **About GProgrammer**
查看GProgrammer软件版本信息以及功能特性。
- **Feedback**
如对GProgrammer软件有任何问题，可发送邮件至software@reg.goodix.com。
- **About Goodix**
如需了解更多Goodix的产品信息，可访问Goodix官网www.goodix.com。