



专注于高端实时控制芯片设计

GS32-DSP 在线烧录工具 GS32Prog 用户手册

2026 年 1 月 14 日

格见构知（上海）半导体有限公司 | 深圳格见构知半导体有限公司（简称“格见半导体”）是一家专注于高端实时控制芯片设计公司，在芯片产品定义、设计研发、量产导入、销售运营等领域都具备丰富经验。

GS32-DSP 是格见半导体自主研发的实时控制微处理器系列产品。公司致力于为数字能源、数字电源、工业自动化、智能汽车、机器人、高端家电等领域提供芯片解决方案。

更多产品资料：sales@gejian-semi.com; support@gejian-semi.com

更多详情访问：www.gejian-semi.com



更多官方资讯

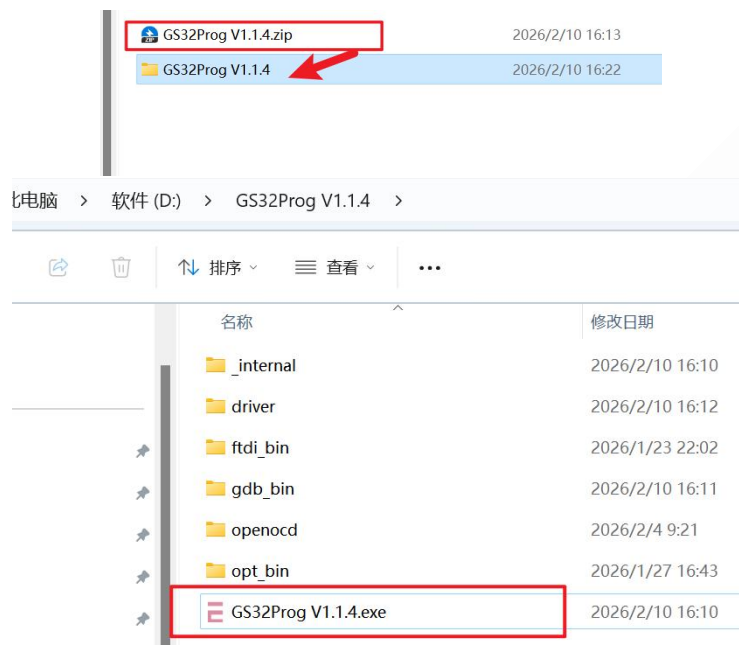
目录

0 快速启动	2
第一步：解压文件	2
第二步：连接目标芯片	2
第三步：选择文件烧录	2
第四步：执行烧录	3
第五步：检查烧录结果	3
1 GS32Prog 简介	4
1.1 说明	4
1.2 目录内容	4
1.3 安装与启动	4
1.4 支持型号	4
1.5 GS32Prog 功能概述	5
2 OpenOCD 页签	6
2.1 说明	6
2.2 烧录程序	6
2.3 连接测试	7
2.4 连接选项	8
2.5 读取芯片内存	8
2.6 全片擦除	9
2.7 工程选择和工程保存	9
2.8 查看烧录文件，连接以及烧录记录	9
3 OptionBytes 页签	11
3.1 说明	11
3.2 Option Bytes 设置	11
3.3 配置项	12
4 BootROM 页签	15
4.1 说明	15
4.2 下载	15
4.3 SCI 模式下载	15
4.4 CAN 模式下载	19
5 Bootloader 页签	19
5.1 说明	19
5.2 镜像头部的添加	20
5.3 镜像文件的烧录	21
6 Image Tools 页签	23
6.1 说明	23
6.2 格式转换	23
6.3 数据修改	24
6.4 文件合并	24
6.5 CRC 校验	25
7 修订历史	25

0 快速启动

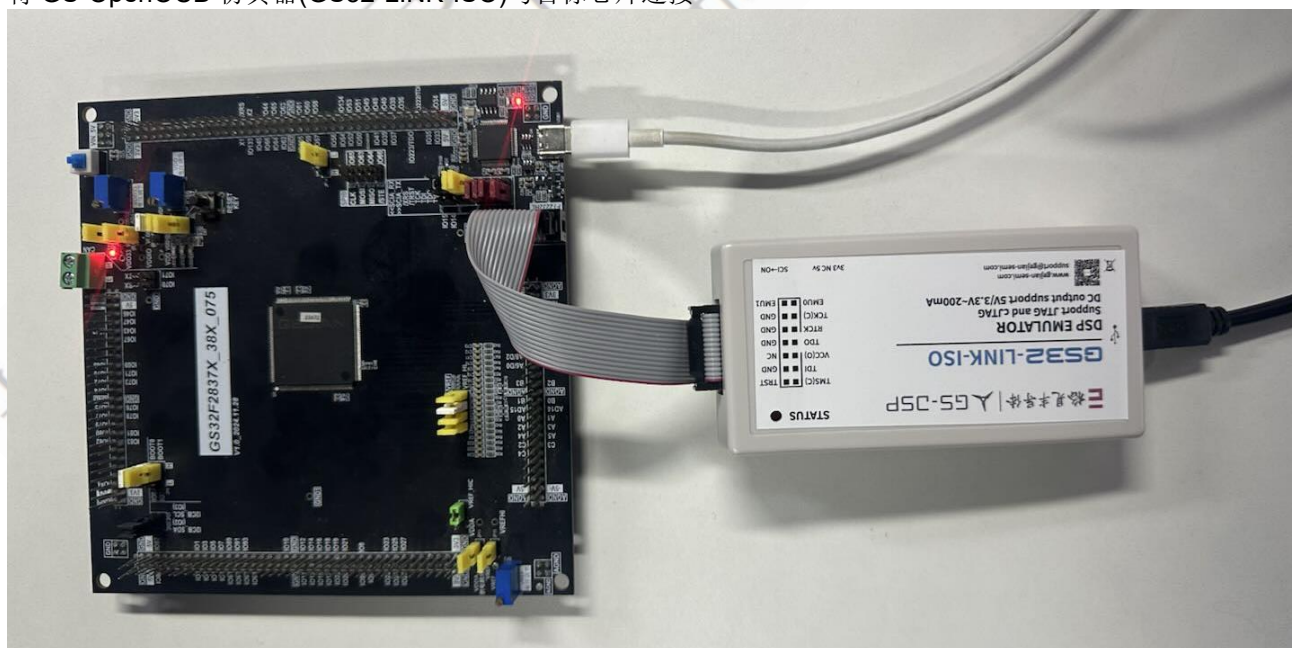
第一步：解压文件

将文件解压后进入软件目录，双击 GS32Prog V1.1.4.exe 启动软件



第二步：连接目标芯片

将 GS-OpenOCD 仿真器(GS32-LINK-ISO)与目标芯片连接



第三步：选择文件烧录

- (1) 选择芯片型号
- (2) “浏览” 选择烧录文件

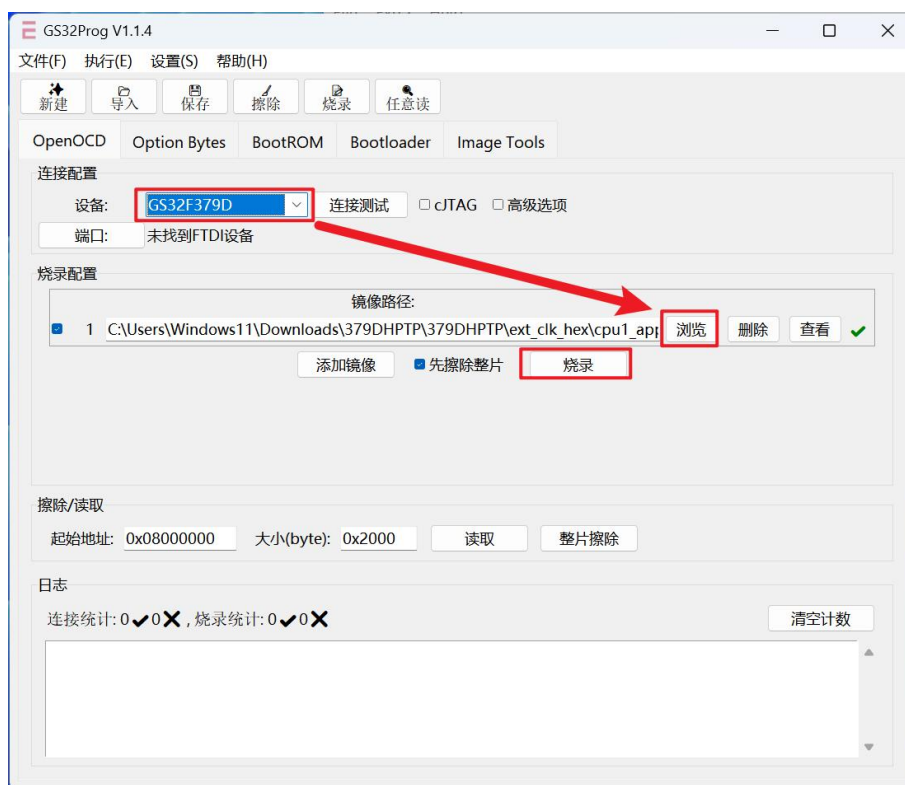


图 0.1-3 选择芯片型号与烧录文件

第四步：执行烧录

点击“烧录”按钮执行烧录



图 0.1-4 执行烧录

第五步：检查烧录结果

进度条完成，且“日志”中显示“烧录并验证成功”，“烧录操作完成”。



图 0.1-5 烧录结果

1 GS32Prog 简介

1.1 说明

GS32Prog 为用户提供 GS32-DSP 芯片的在线烧录功能，可以通过 JTAG/cJTAG 接口、SCI BootROM、SCI BootLoader 等进行芯片 Flash 的擦写操作。还包含对镜像文件(HEX 文件/BIN 文件)修改与添加校验和等操作；

1.2 目录内容

文件	说明
openocd	可执行文件 gs-openocd 存放位置
driver	USB 工具存放位置
docs	GS32Prog 用户手册
GS32Prog.exe	界面启动程序

1.3 安装与启动

GS32Prog 为绿色软件，解压之后，运行根目录下的 GS32Prog.exe 即可启动。

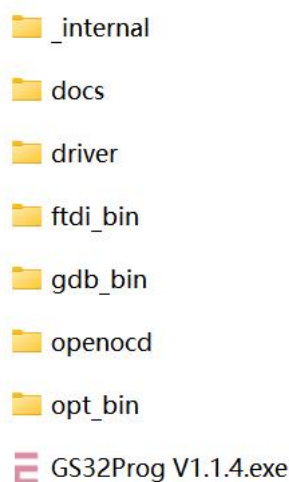


图 1.3-1 GS32Prog 安装目录的内容

1.4 支持型号

序号	芯片型号
1	GS32F0039, GS32F0039H, GS32F0039H1, GS32F0039P, GS32F0038-Q, GS32F0039-Q, GS32F0039H-Q, GS32F0049, GS32F0049H, GS32F0049H1, GS32F0025A, GS32FMT5000A, GS32FMT5800
2	GS32F0025, GS32F0025H, GS32F035, GS32F035H, GS32F035P, GS32F00137, GS32F00137H, GS32F00157, GS32F00157H, GS32F00157-Q, GS32F0039A, GS32F0049A, GS32FP55A, GS32FMT4000, GS32FMT5000, GS32FPW3300, GS32FPW3600

序号	芯片型号
3	GS32F075, GS32F075D, GS32F374S, GS32F374SX, GS32F374SH, GS32F374SH1, GS32F374D, GS32F374DX, GS32F374DH, GS32F374DH1, GS32F377S, GS32F377SX, GS32F377SH, GS32F377SH1, GS32F377D, GS32F377DH, GS32F377DH1, GS32F379S, GS32F379SX, GS32F379SH, GS32F379SH1, GS32F379D, GS32F379DX, GS32F379DH, GS32F379DH1, GS32F388DA, GS32F388SX, GS32FP65SK, GS32FP65SKX, GS32FP65SKH, GS32FP65SKH1, GS32FP65DK, GS32FP65DKX, GS32FP65DKH, GS32FP65DKH1,

1.5 GS32Prog 功能概述

GS32Prog 分为 5 个页签:

- OpenOCD, 通过 JTAG/cJTAG 接口与 GS-OpenOCD 仿真器(GS32-LINK-ISO), 实现在线烧录功能。
- Option Bytes, 设置 RDP 等级, 配置 option byte 选项。
- BootROM, 通过 SCI/CAN 接口与芯片内部出厂自带的 BootROM, 实现在线烧录功能。
- Bootloader, 通过 SCI/CAN 接口与用户写入的 BootLoader, 实现在线烧录功能。
- Image Tools, 镜像文件的格式转换, 数据修改, 文件合并, CRC 校验等操作。

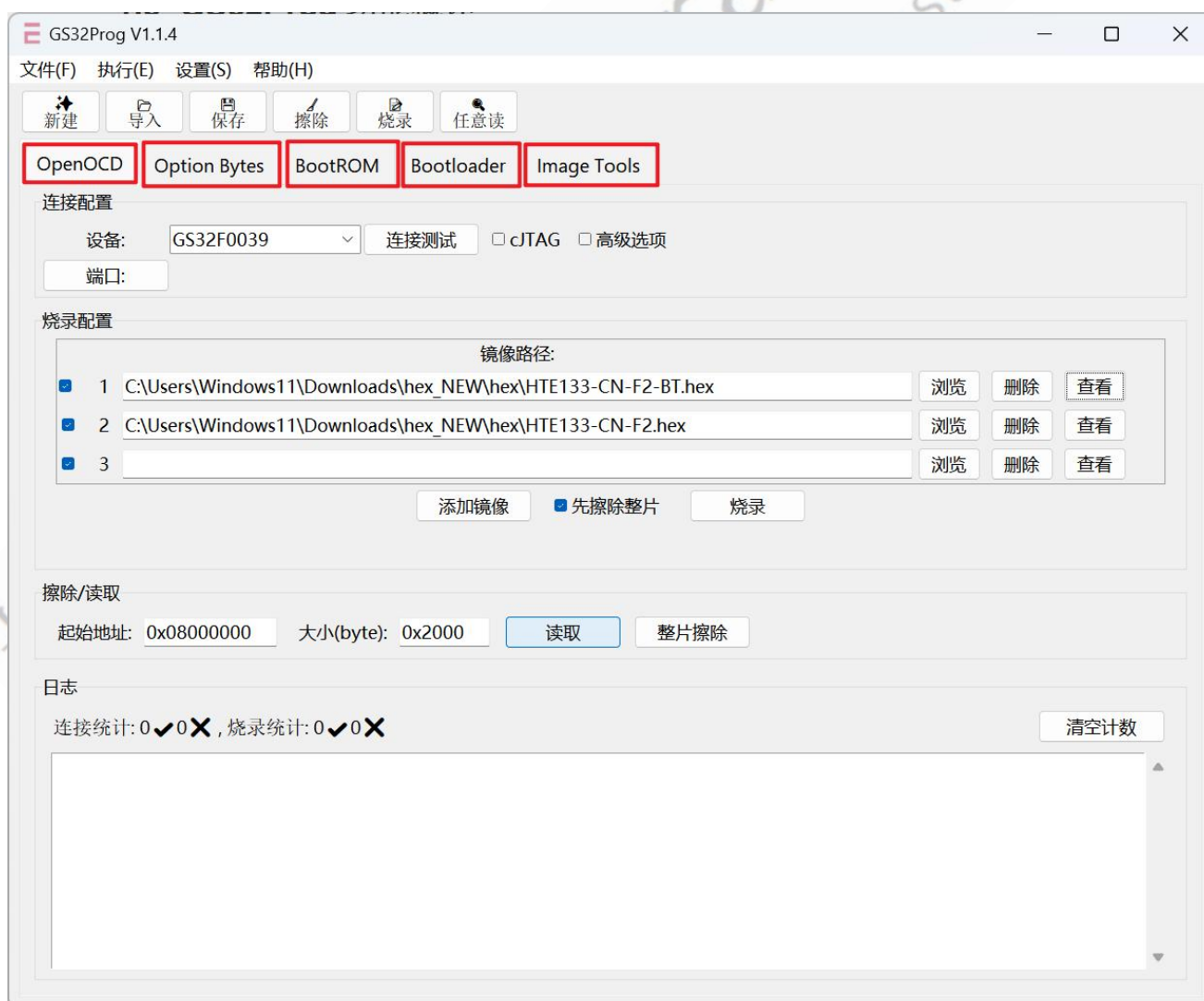


图 1.4-1 GS32Prog 界面

2 OpenOCD 页签

2.1 说明

本页签的功能是通过 JTAG/cJTAG 接口与 GS32-LINK-ISO 仿真器对 GS32-DSP 进行在线程序烧录（Flash 擦写）。使用之前，需确保芯片正常上电，且仿真器与目标芯片正确连接。

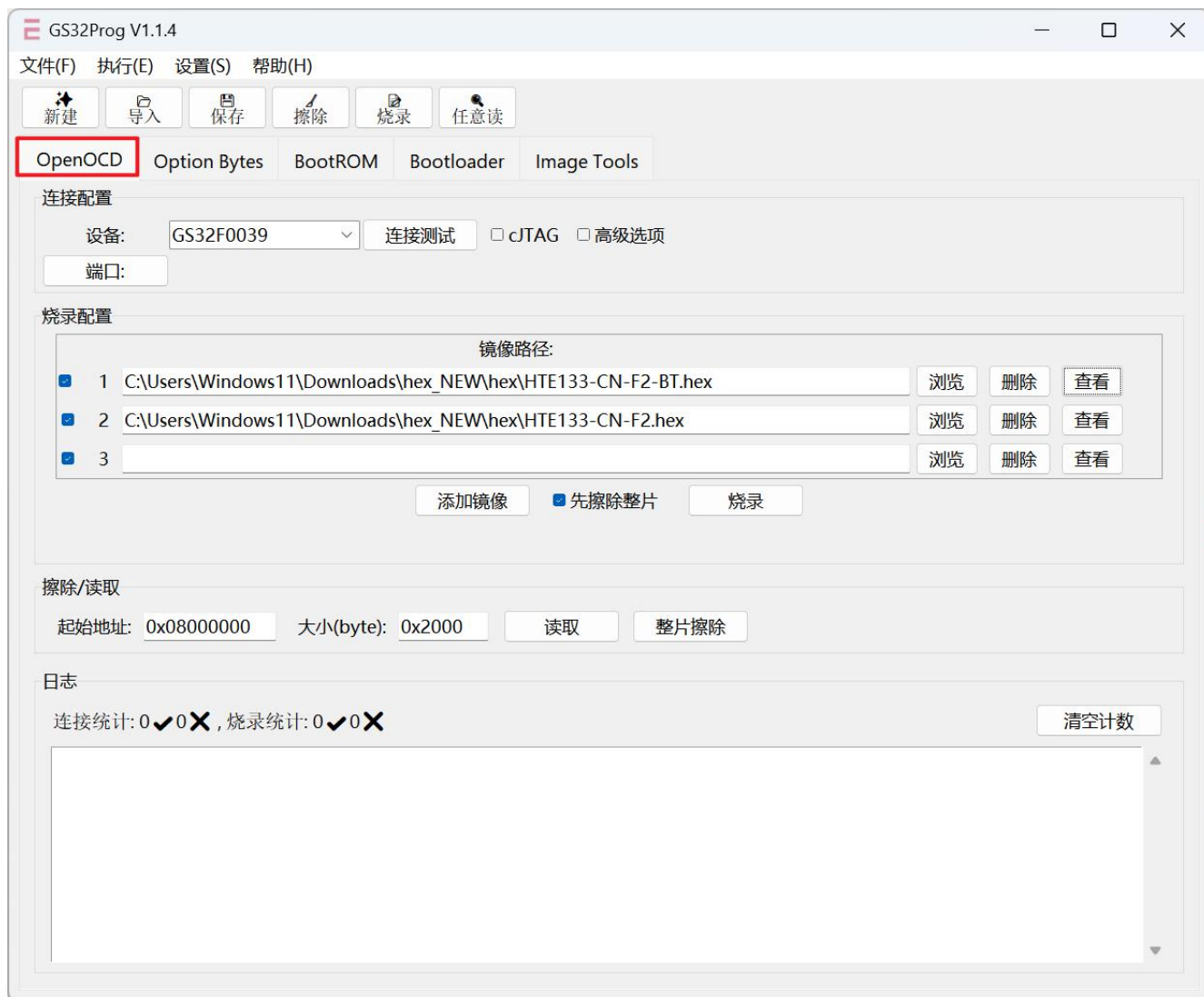


图 2.1-1 OpenOCD 界面

2.2 烧录程序

可以选择 bin、hex、elf 进行烧录，可以选择多个镜像文件顺序烧录。步骤如下：

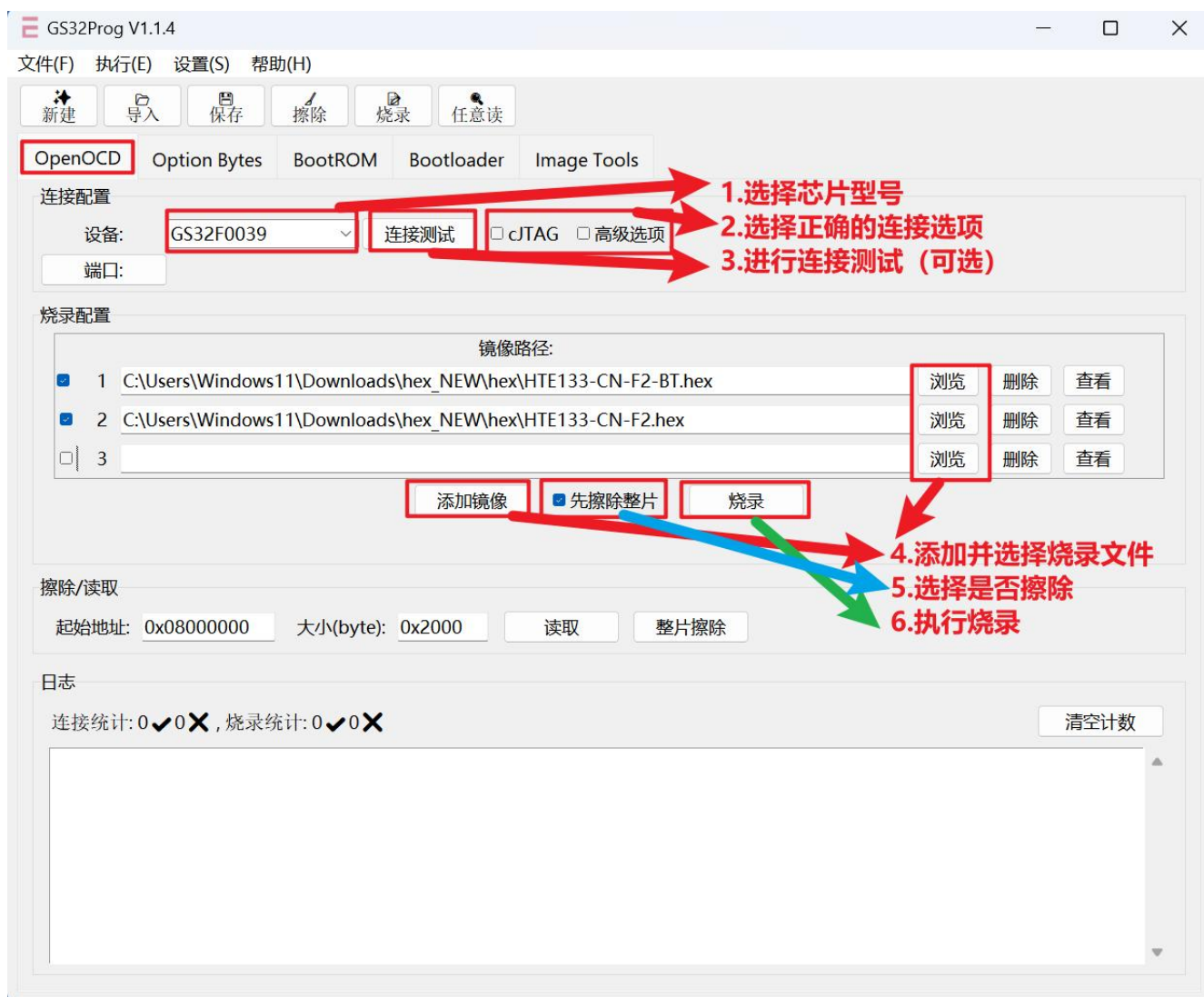


图 2.2-1 烧录步骤

注意：1.如果选择 bin 进行烧录，弹出起始地址输入界面，输入需要的起始地址后点击确定即可

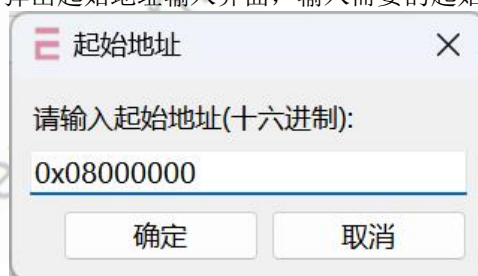


图 2.2-2 bin 起始地址选择输入

2. 烧录内容大小不可以超过所选芯片的规格

2.3 连接测试

连接测试用于测试当前的调试器与硬件连接是否正常。硬件方面需要将 GS32-LINK-ISO 仿真器与芯片正确连接。然后在软件上进行如下操作：

- (1) 选择芯片型号
- (2) 点击连接测试

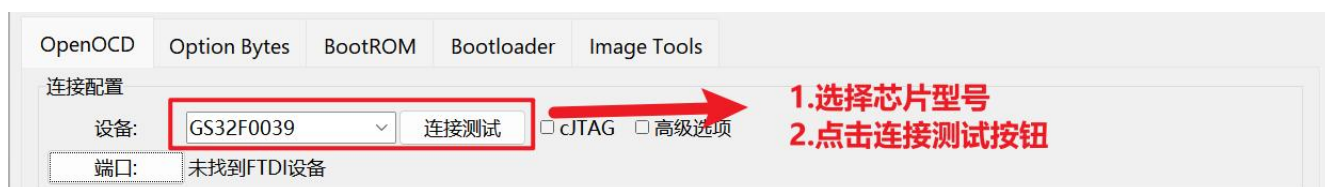


图 2.3-1 连接测试

2.4 连接选项

【cJTAG】：选择 2 线 TCKC 和 TMSC 的接口方式

【高级选项】：显示与连接频率有关的选项

【JTAG 频率】：设置调试器与目标设备通信的接口速度，直接影响调试效率和稳定性，单位 KHZ

【芯片主频】：设置目标芯片的时钟频率，单位 HZ

【端口】：多个调试器连接后，选择调试器使用

选项包括 cJTAG、JTAG 频率(KHZ)、芯片主频(HZ)、需要连接的端口，其中 JTAG 频率(KHZ)和芯片主频(HZ)如果不选择，则使用默认设置

【端口】按钮点击后如果显示相关的选项，则说明满足端口选择的条件，可以指定需要连接到端口

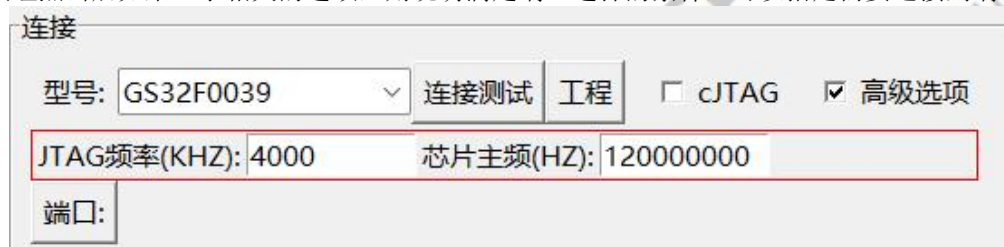


图 2.4-1 连接选项

2.5 读取芯片内存

设置的起始位置和需要读取的大小，执行读取任务。

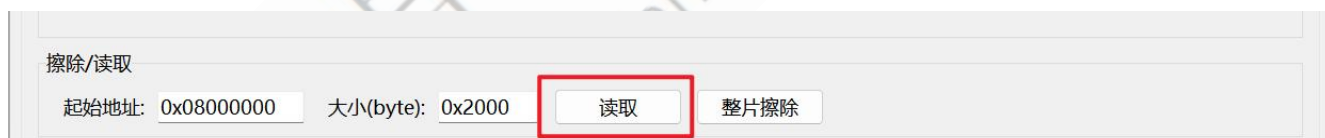


图 2.5-1 设置和执行读取芯片 hex

显示指定内存地址的数据，并可以保存为 HEX 文件或 BIN 文件。

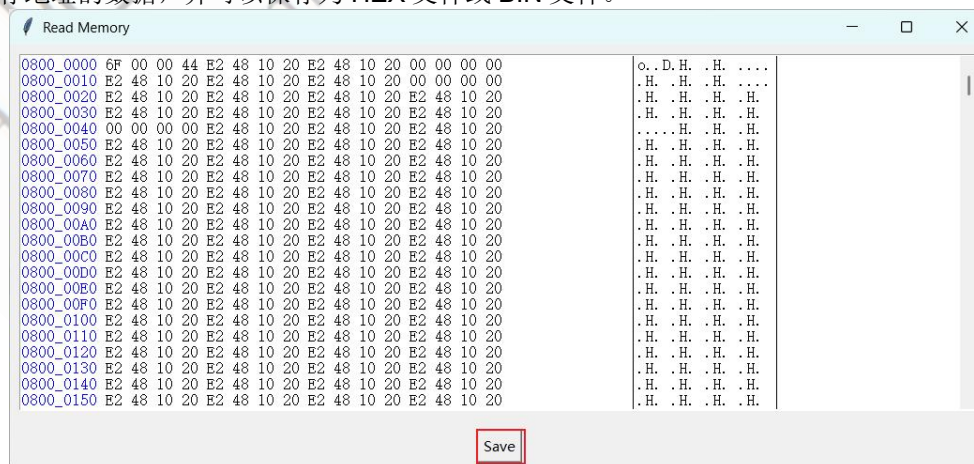


图 2.5-2 显示 hex

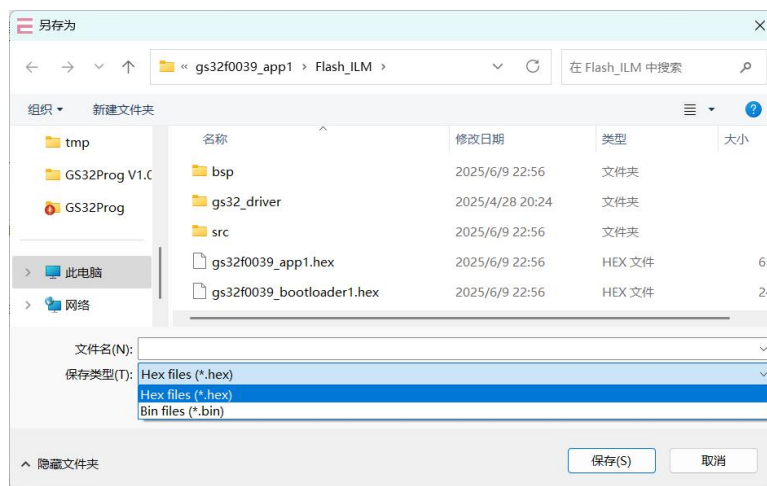


图 2.5-3 保存界面

2.6 全片擦除

执行全片擦除任务。



图 2.6-1 执行全片擦除

2.7 工程选择和工程保存

工程管理“新建”，“导入”，“保存”可以将当前的烧录配置保存在一个工程中，便于下次烧录时快速载入烧录配置：

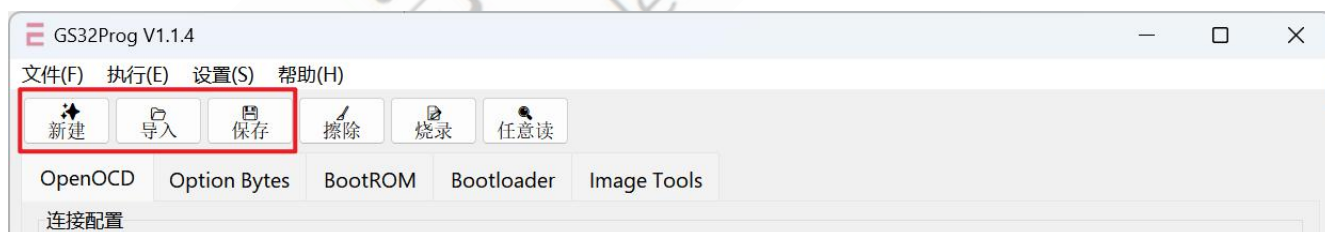


图 2.7-1 工程管理界面

2.8 查看烧录文件，连接以及烧录记录

点击每个文件后的“查看”按钮可以查看此文件的内容

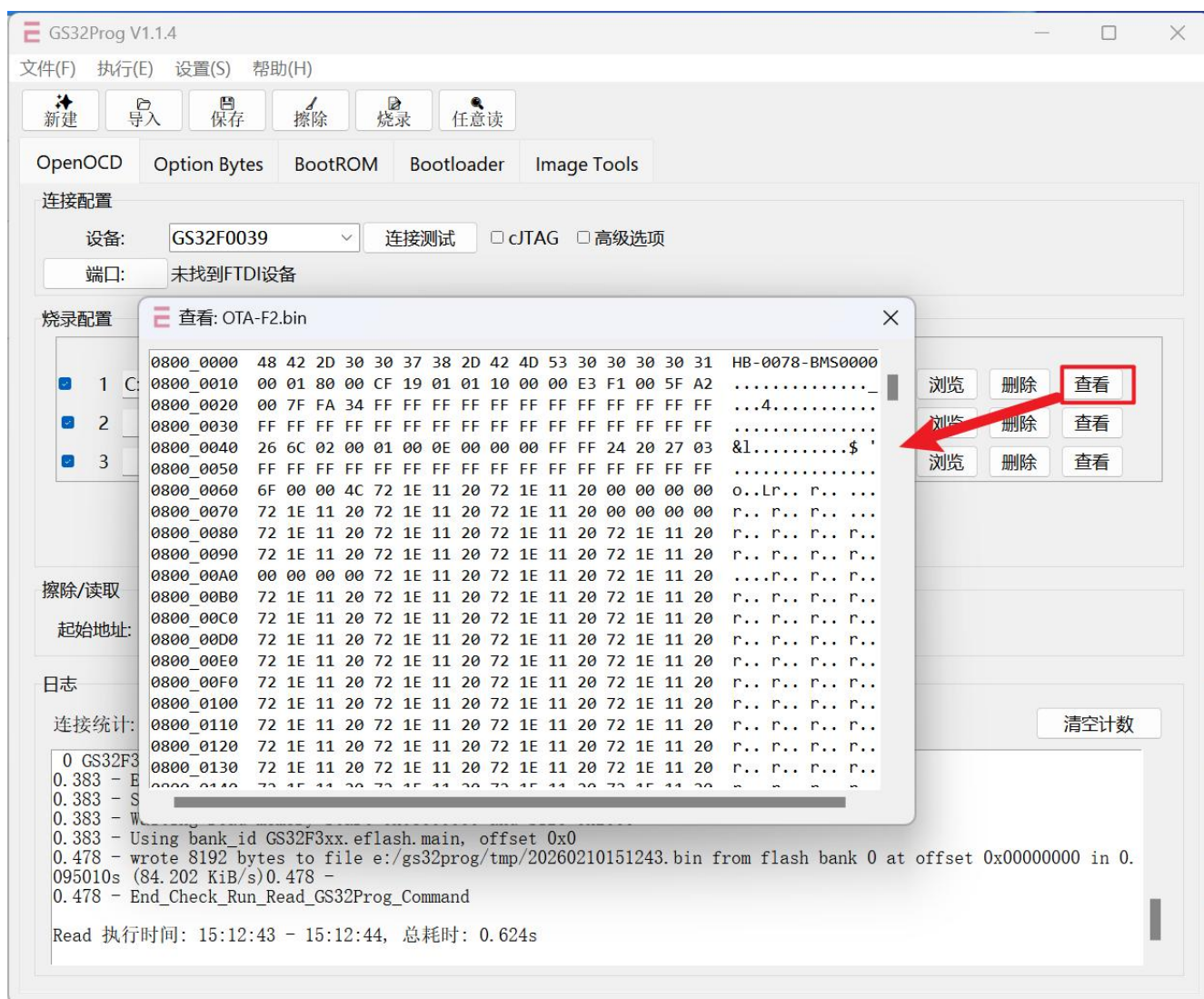


图 2.8-1 烧录程序 hex 显示

底部日志信息，记录使用过程中的成功与失败，且有相关的日志进行分析，日志位置在当前程序的 Logs 文件夹中



图 2.8-2 成功失败记录

3 OptionBytes 页签

3.1 说明

此页面主要用于修改 Flash 中的特定功能选项

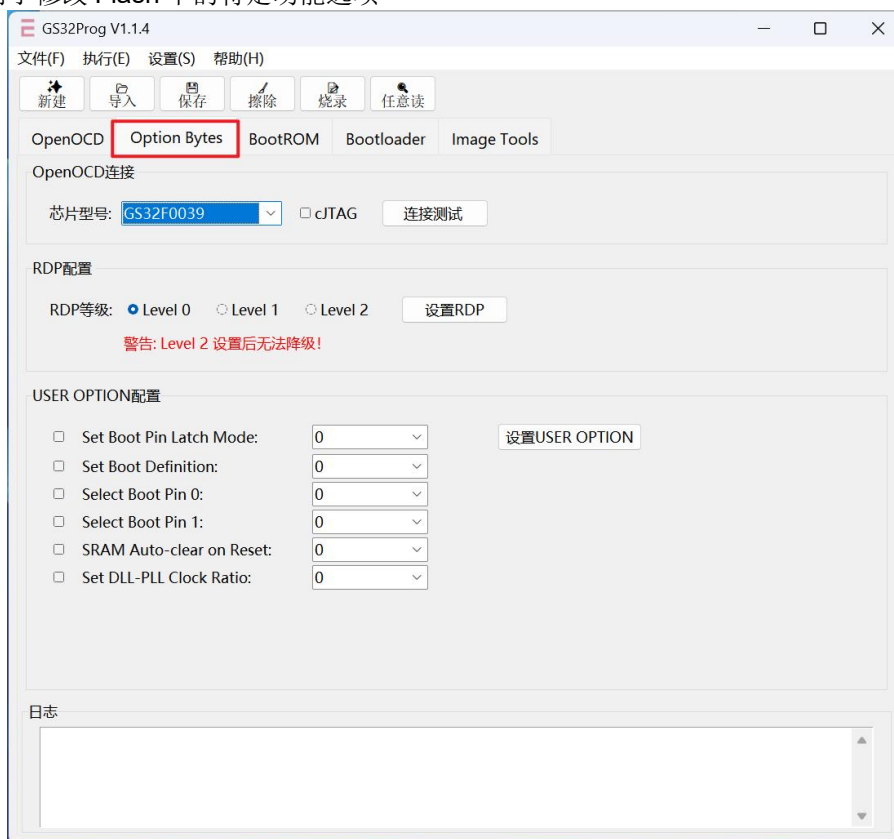


图 3.1-1 OptionBytes 界面

3.2 Option Bytes 设置

设置选项：1.设置 RDP 等级
2.设置 USER OPTION

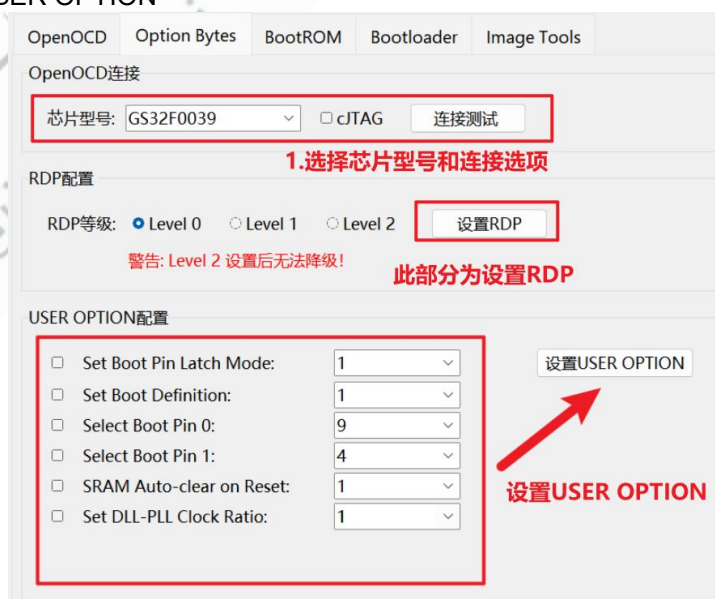


图 3.2-1 Option Bytes 设置详细说明

3.3 配置项

序号	OPT 名称	选项说明
1	RDP	Level 0: 拥有读写权限 Level 1: 不可读不可烧录, 可降级处理, 降级时将擦除固件 Level 2: 不可读不可烧录, 无法降级
2	Set Boot Pin Latch Mode	0: XRS 复位后会重新锁存 Boot Pin 1: 仅上电锁存 Boot Pin
3	Set Boot Definition	设置自定义启动引导模式 (表 3.3.1 Boot_DEF 设置表)
4	Select Boot Pin 0	设置引导引脚 0 的 GPIO 编号 (0-255)
5	Select Boot Pin 1	设置引导引脚 1 的 GPIO 编号 (0-255)
6	SRAM Auto-clear on Reset	0: SRAM 上电复位不自清除 1: SRAM 上电复位自清除
7	Set DLL-PLL Clock Ratio	0: 低速模式 (1:1) 1: 高速模式 (1:2)
8	Set Single/Dual BANK Mode	0: single bank mode 1: dual bank mode
9	Enable ILM/DLM Shadow	0: 不使能 (128KB ILM ; 32KB DLM) 1: 使能 ILM/DLM Shadow (64KB ILM + 64KB ILM Shadow, 32KB DLM + 32KB DLM Shadow)
10	Set Boot Pin Latch Follow Software	设置启动引脚上电锁存模式跟随软件配置

注意: 橙色区域并非所有芯片型号都支持

序号	OPT 名称	支持此设置的芯片型号
7	Set DLL-PLL Clock Ratio	GS32F0039, GS32F0039H, GS32F0039H1, GS32F0039P, GS32F0038-Q, GS32F0039-Q, GS32F0039H-Q, GS32F0049, GS32F0049H, GS32F0049H1, GS32F0025A, GS32FMT5000A, GS32FMT5800
8	Set Single/Dual BANK Mode	GS32F0025, GS32F0025H, GS32F035, GS32F035H, GS32F035P, GS32F00137, GS32F00137H, GS32F00157, GS32F00157H, GS32F00157-Q, GS32F0039A, GS32F0049A, GS32FP55A, GS32FMT4000, GS32FMT5000, GS32FPW3300, GS32FPW3600
9	Enable ILM/DLM Shadow	
10	Set Boot Pin Latch Follow Software	

自定义启动引导模式参照表:

序号	组合配置	BOOT_DEF
1	00:flash、01:flash、10:flash、11:flash	0
2	00:ram、01:flash、10:flash、11:flash	1
3	00:system、01:flash、10:flash、11:flash	2
4	00:flash、01:ram、10:flash、11:flash	4
5	00:ram、01:ram、10:flash、11:flash	5
6	00:system、01:ram、10:flash、11:flash	6
7	00:flash、01:system、10:flash、11:flash	8

序号	组合配置	BOOT_DEF
8	00:ram、01:system、10:flash、11:flash	9
9	00:system、01:system、10:flash、11:flash	10
10	00:flash、01:flash、10:ram、11:flash	16
11	00:ram、01:flash、10:ram、11:flash	17
12	00:system、01:flash、10:ram、11:flash	18
13	00:flash、01:ram、10:ram、11:flash	20
14	00:ram、01:ram、10:ram、11:flash	21
15	00:system、01:ram、10:ram、11:flash	22
16	00:flash、01:system、10:ram、11:flash	24
17	00:ram、01:system、10:ram、11:flash	25
18	00:system、01:system、10:ram、11:flash	26
19	00:flash、01:flash、10:system、11:flash	32
20	00:ram、01:flash、10:system、11:flash	33
21	00:system、01:flash、10:system、11:flash	34
22	00:flash、01:ram、10:system、11:flash	36
23	00:ram、01:ram、10:system、11:flash	37
24	00:system、01:ram、10:system、11:flash	38
25	00:flash、01:system、10:system、11:flash	40
26	00:ram、01:system、10:system、11:flash	41
27	00:system、01:system、10:system、11:flash	42
28	00:flash、01:flash、10:flash、11:ram	64
29	00:ram、01:flash、10:flash、11:ram	65
30	00:system、01:flash、10:flash、11:ram	66
31	00:flash、01:ram、10:flash、11:ram	68
32	00:ram、01:ram、10:flash、11:ram	69
33	00:system、01:ram、10:flash、11:ram	70
34	00:flash、01:system、10:flash、11:ram	72
35	00:ram、01:system、10:flash、11:ram	73
36	00:system、01:system、10:flash、11:ram	74
37	00:flash、01:flash、10:ram、11:ram	80
38	00:ram、01:flash、10:ram、11:ram	81
39	00:system、01:flash、10:ram、11:ram	82
40	00:ram、01:ram、10:ram、11:ram	85
41	00:system、01:ram、10:ram、11:ram	86
42	00:flash、01:system、10:ram、11:ram	88
43	00:ram、01:system、10:ram、11:ram	89
44	00:system、01:system、10:ram、11:ram	90
45	00:flash、01:flash、10:system、11:ram	96
46	00:ram、01:flash、10:system、11:ram	97
47	00:system、01:flash、10:system、11:ram	98
48	00:flash、01:ram、10:system、11:ram	100
49	00:ram、01:ram、10:system、11:ram	101

序号	组合配置	BOOT_DEF
50	00:system、01:ram、10:system、11:ram	102
51	00:flash、01:system、10:system、11:ram	104
52	00:ram、01:system、10:system、11:ram	105
53	00:system、01:system、10:system、11:ram	106
54	00:flash、01:flash、10:flash、11:system	128
55	00:ram、01:flash、10:flash、11:system	129
56	00:system、01:flash、10:flash、11:system	130
57	00:flash、01:ram、10:flash、11:system	132
58	00:ram、01:ram、10:flash、11:system	133
59	00:system、01:ram、10:flash、11:system	134
60	00:flash、01:system、10:flash、11:system	136
61	00:ram、01:system、10:flash、11:system	137
62	00:system、01:system、10:flash、11:system	138
63	00:flash、01:flash、10:ram、11:system	144
64	00:ram、01:flash、10:ram、11:system	145
65	00:system、01:flash、10:ram、11:system	146
66	00:flash、01:ram、10:ram、11:system	148
67	00:ram、01:ram、10:ram、11:system	149
68	00:system、01:ram、10:ram、11:system	150
69	00:flash、01:system、10:ram、11:system	152
70	00:ram、01:system、10:ram、11:system	153
71	00:system、01:system、10:ram、11:system	154
72	00:flash、01:flash、10:system、11:system	160
73	00:ram、01:flash、10:system、11:system	161
74	00:system、01:flash、10:system、11:system	162
75	00:flash、01:ram、10:system、11:system	164
76	00:ram、01:ram、10:system、11:system	165
77	00:system、01:ram、10:system、11:system	166
78	00:flash、01:system、10:system、11:system	168
79	00:ram、01:system、10:system、11:system	169
80	00:system、01:system、10:system、11:system	170

表 3.3.1 Boot_DEF 设置表

4 BootROM 页签

4.1 说明

本页签的功能通过 SCI/CAN 接口与芯片内部出厂自带的 bootrom，实现在线烧录功能。

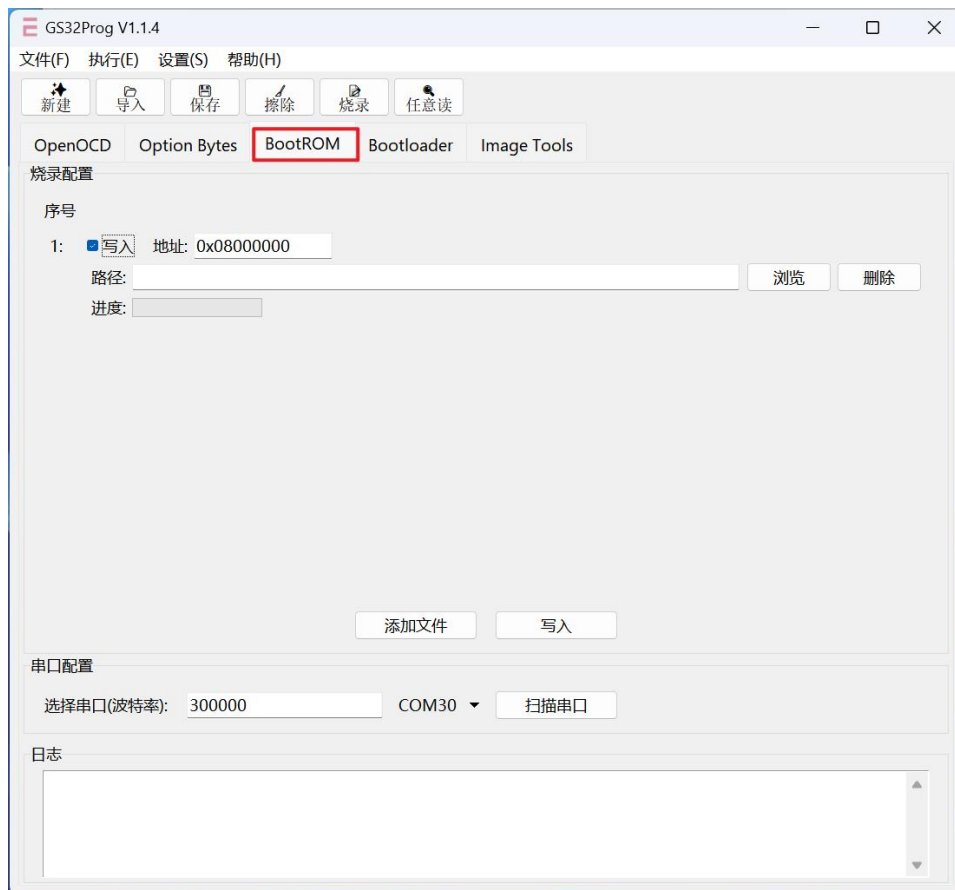


图 4.1-1 BootROM 界面

4.2 下载

此功能要求下位机带有 bootrom，控制 APP 镜像文件的烧写，该功能通过 UART 串口实现。用户可以在互不冲突的起始地址烧录多个 APP 文件。点击“添加文件”->“浏览”，添加需要下载的文件，若确定此次烧录，勾选左边“写入”小框，若需要删除行，点击每行对应的“删除”。

该功能支持 Hex 文件和 bin 文件，加载若为 Hex 文件会自动转换成 bin 文件，保存在自动生成的“output”文件夹中。

加载好对应的起始地址和图像路径后选择想要的串口，点击“写入”开始烧录。（在此之前，确认负责接收镜像的 BOOTROM 已开始运行）。每一行最后的进度条显示该文件的烧录的进度。

可以选择合适的频率以及端口进行烧录。

4.3 SCI 模式下载

下面以 GS32F379D 为例，通过 bootrom 同时升级 3 个 bin 文件，

- (1) Bootloader: 0x08000000;
- (2) Cpu1 app: 0x08008000;
- (3) Cpu2 app: 0x08100000。

1. 将生态板或者 EVB 板的 bootpin，即 BOOT0（GPIO84），BOOT1（GPIO72）跳线帽切换至 0b01/0b10 状态，让芯片进入 BootRAM 模式，注意，在该芯片上，由于 GPIO84 同时作为 bootpin 和 SCIA-TX，需先断开 TX 接线，芯片上电锁存后，再连接 TX，不然 DSP 无法进 BootRAM 模式。



图 4.3-1 开发板或 socket 板 bootpin 跳线配置状态

确认芯片进入 BootRAM 模式的方法，通过 jlink command 连接执行 halt

若 PC 指针为 0x1FFFxxxx, 表示进入 BootRAM 模式,

若 PC 指针为 0x1FFF0000, 则 bootrom 程序本身有问题或者芯片没有 BootRAM 程序

若 PC 指针为其它值, 则表示锁存进入 bootrom 失败, 这时候有可能进入到 main flash 模式
由于执行了 halt, 测试时, 需要 reset 重新进行锁存进 BootRAM 模式, 然后在线升级。

```

DataBuf: 4 entries
autoexec[0] implemented: Yes
Detected: RV32 core
Temp. halting CPU for for feature detection...
HW instruction/data BPs: 8
Support set/clr BPs while running: No
HW data BPs trigger before execution of inst
CSR access via abs. commands: Yes
Compressed instruction support: Yes
Feature detection done. Restarting core...
BG memory access support: Via SBA
Memory zones:
Zone: "Default" Description: Default access mode
RISC-V identified.
J-Link>halt
pc = 1FFF1DF2 sp = 1084FFA0 ra = 1FFF0DBA
gp = 10840000 tp = 10840210 fp = 1084FFEC
t0 = 00000000 t1 = 0000000F t2 = 00000000
t3 = 671F0FFD t4 = 07000000 t5 = 00000000 t6 = 00000000
a0 = 108402A0 a1 = 10840214 a2 = 00001000 a3 = 1084FF80
a4 = 400C1000 a5 = 00000000 a6 = 400C0000 a7 = 00000000
s1 = 00000000 s2 = 108402A0 s3 = 00000000 s4 = 00000000
s5 = 00000000 s6 = 00000000 s7 = 00000000 s8 = 00000000
s9 = 00000000 s10 = 00000000 s11 = 00000000
J-Link>
    
```

图 4.3-2 成功进入 BOOTROOM 模式确认

选择固件镜像进行下载，串口速率 300000bps。如图：

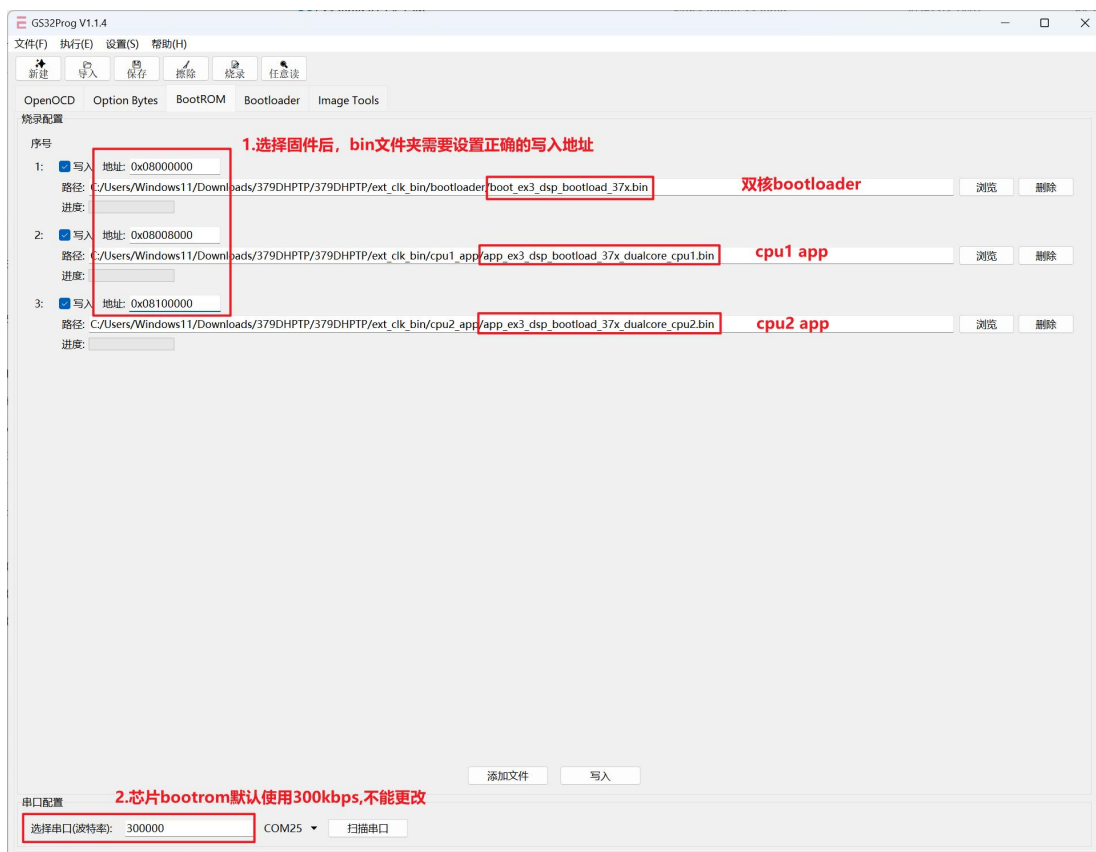


图 4.3-3 BOOTROOM 升级 GS32F379D 界面

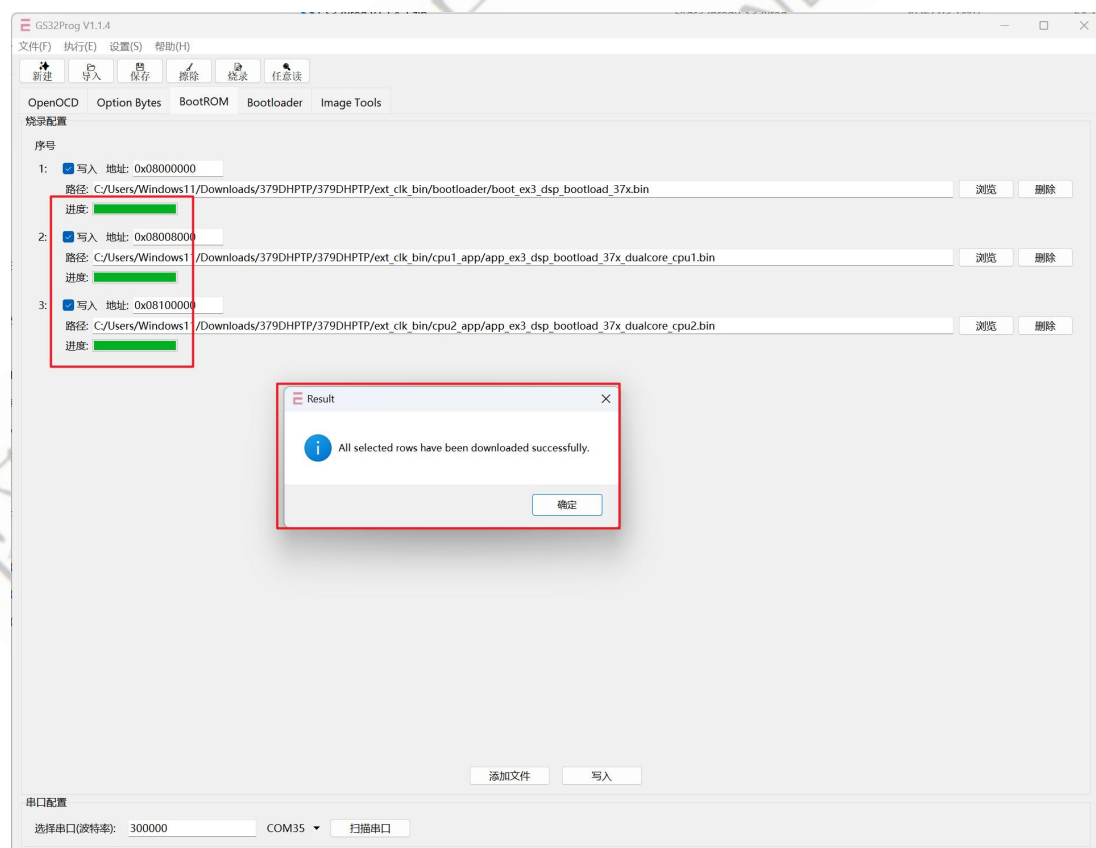


图 4.3-4 BOOTROOM 升级 GS32F379D 完成界面

将 bootpin 切换至 0b11, 进入 main flash 启动模式。



图 4.3-5 开发板或 socket 板 bootpin 跳线配置成 main flash 启动状态

通过 J-Flash 工具/OpenOCD 页签回读以上 3 个文件,并且跟 ide 编译出来的 raw binary 文件进行二进制比对,有效内容必须完全相等

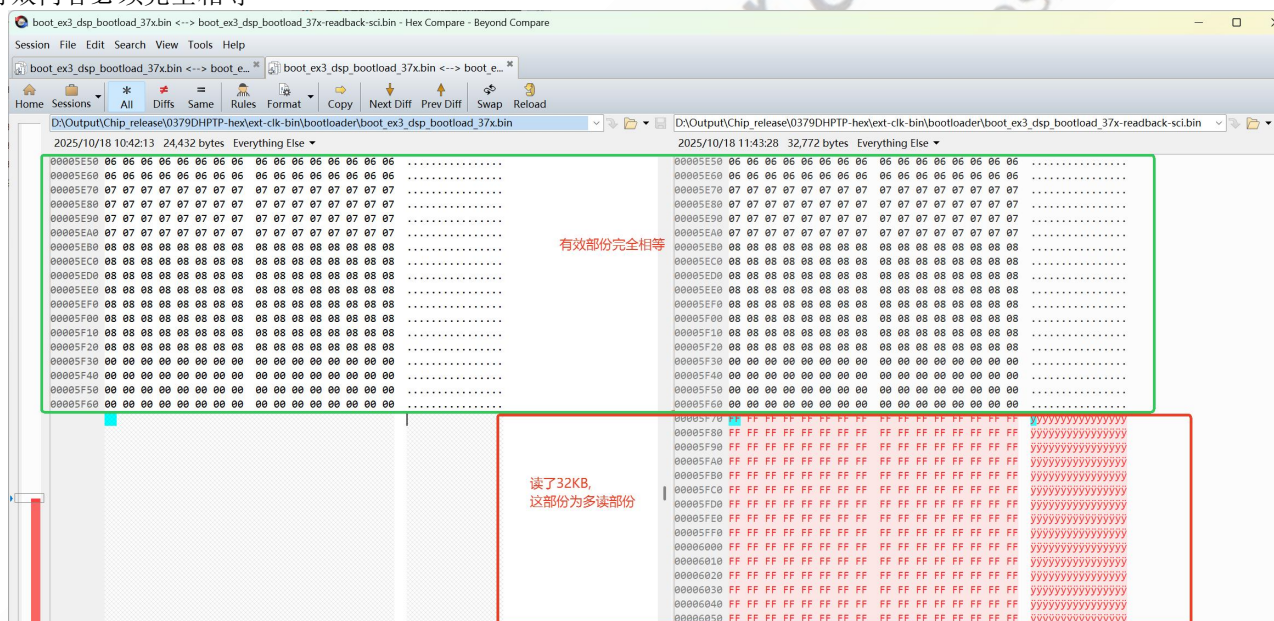


图 4.3-6 二进制比对 bootloader binary 回读文件状态

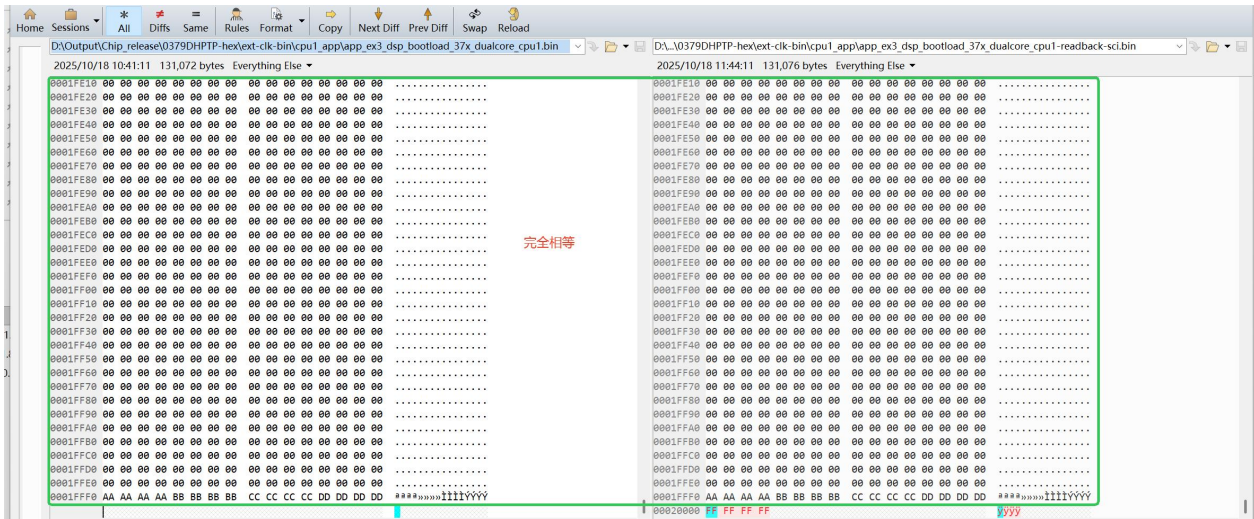


图 4.3-7 二进制比对 cpu1 app binary 回读文件状态



图 4.3-8 二进制比对 cpu2 app binary 回读文件状态

4.4 CAN 模式下载

GS32Prog 不支持 CAN，只支持 COM 口下载，若需要采用 CAN 下载，请使用 USB 转 COM-CAN。

5 Bootloader 页签

5.1 说明

此页面主要提供 APP 镜像头部的添加和镜像烧写功能。烧写功能要求下位机带有 bootloader

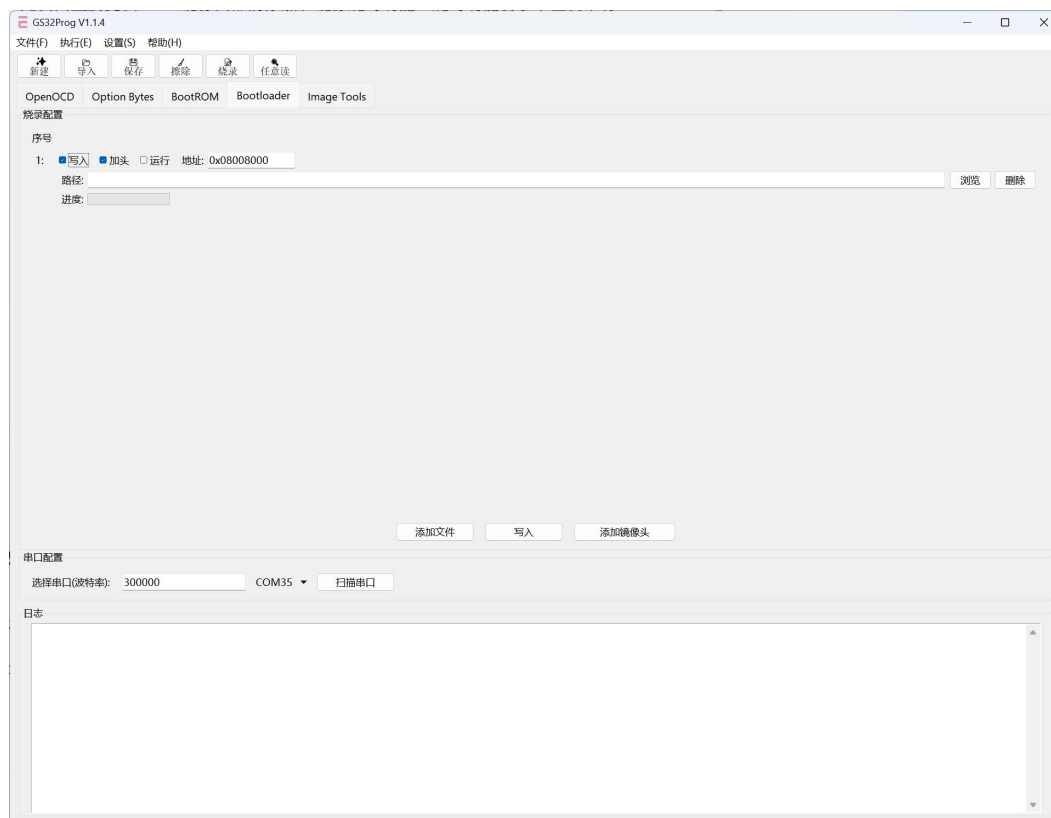


图 5.1-1 Bootloader 界面

5.2 镜像头部的添加

需要点击“添加文件”按钮添加镜像文件（bin 文件）同时勾选对应行的“加头”勾选框（默认勾选）。点击下方“添加镜像头”按钮，上位机开始为所有勾选过“加头”的行添加镜像头部。同时日志输出到 log 框中。

添加头部完成后，对应行会出现‘H’标记，标记此镜像已添加头部选项，生成的已添加头部文件位于 output 目录下。

文件名: 原始文件名.bin -> 原始文件名_with_header.bin

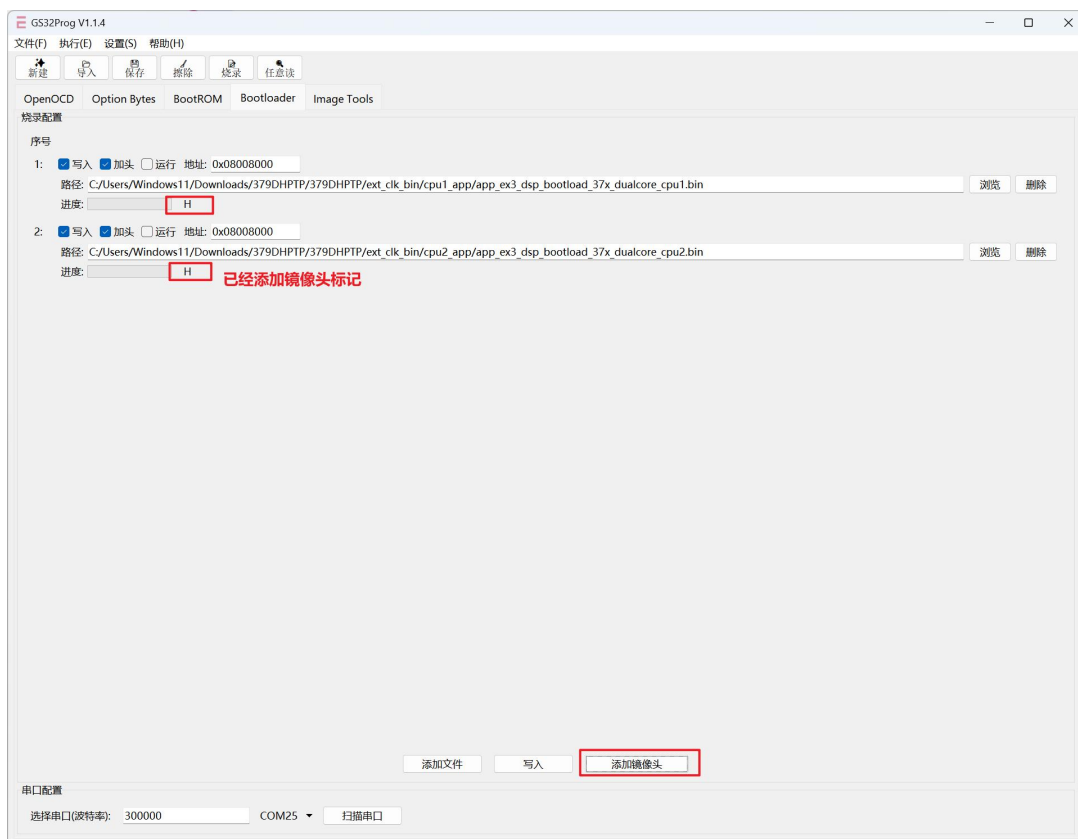


图 5.2-1 头部添加

5.3 镜像文件的烧录

烧录功能通过 UART 串口实现，要求下位机带有 **bootloader**。用户可以烧录多个 **APP** 镜像。点击“添加文件”，增加行，若确定此次烧录，勾选左边的“写入”勾选框，若需要删除行，点击每行对应的“删除”。输入合法地址后下方点击“写入”按钮烧写。上位机开始烧写每行勾选过“写入”选项的对应镜像。烧写镜像时若未添加头部（右侧无‘H’标记），则会烧写原所选文件，若已添加头部，则实际烧写 **output** 文件夹下对应生成的已添加头部的镜像文件。

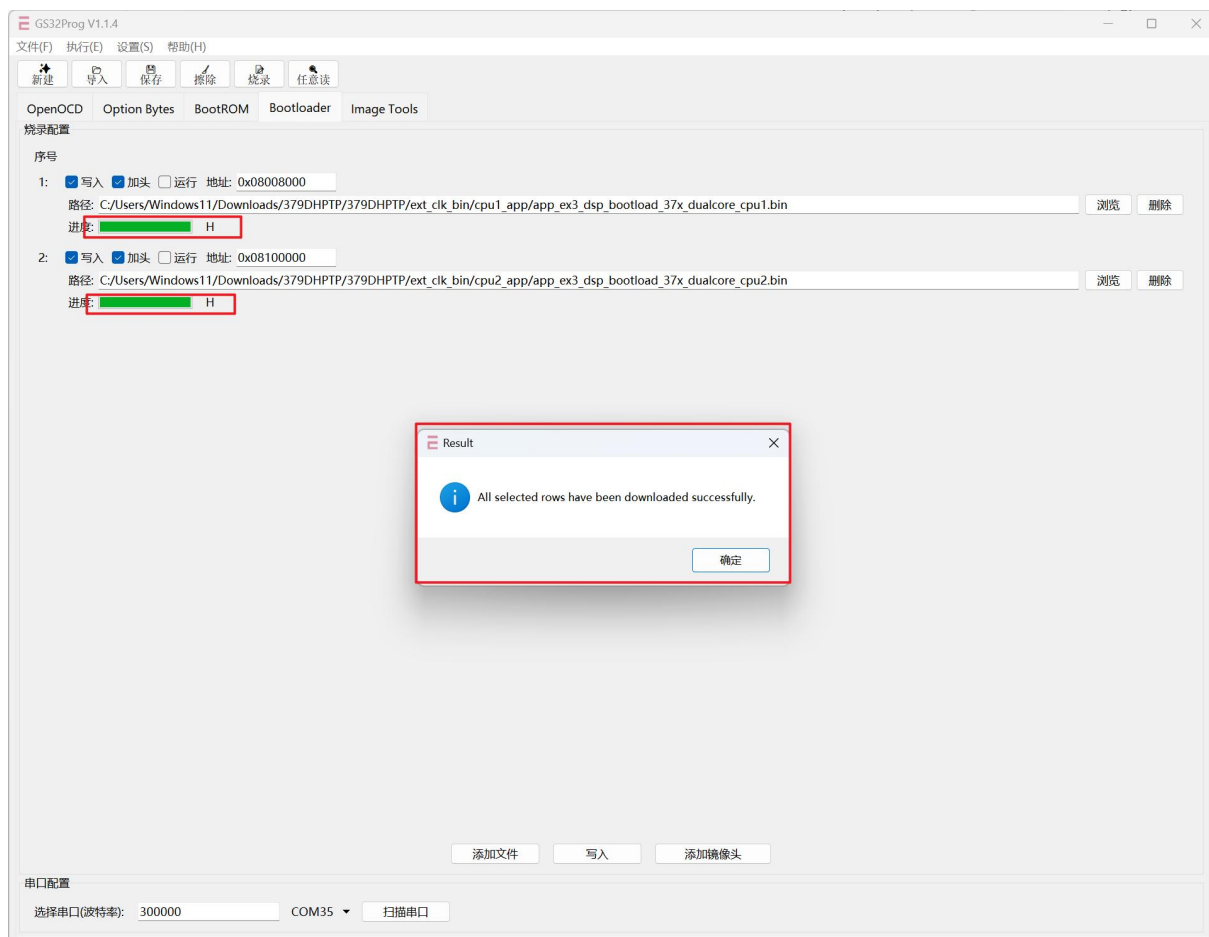


图 5.3-1 烧录

6 Image Tools 页签

6.1 说明

此页签包含对镜像文件的格式转换处理，更改文件中地址的数据内容，文件合并以及 CRC 校验。

6.2 格式转换

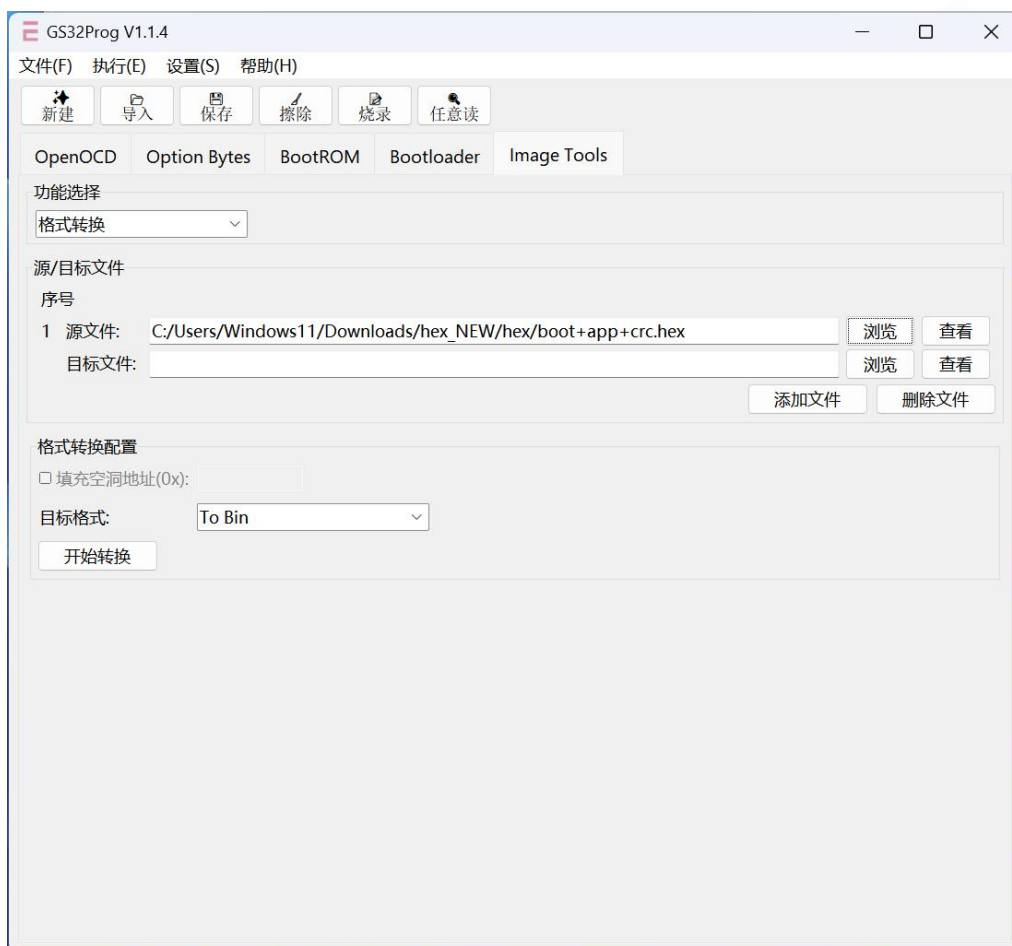


图 6.2-1 格式转换界面

镜像格式转换功能允许用户在 BIN、HEX 和 ELF 格式之间进行互转，将 Intel HEX（16-byte 数据记录）转换为 32-Byte Hex（每行合并为 32 字节数据记录）。

关于地址空洞填充：在所有转换模式中，GS32Prog 均允许用户可选地指定往地址不连续处填充值。

启用填充：如果勾选了“填充空洞地址(0x)”，GS32Prog 会将指定的填充值（如 0xFF）填充不连续的空洞（间隙）地址。这在 HEX/ELF 转换为 BIN 时尤为重要，以确保数据在 BIN 文件中的偏移位置正确，并符合 Flash 擦除后的状态。

6.3 数据修改

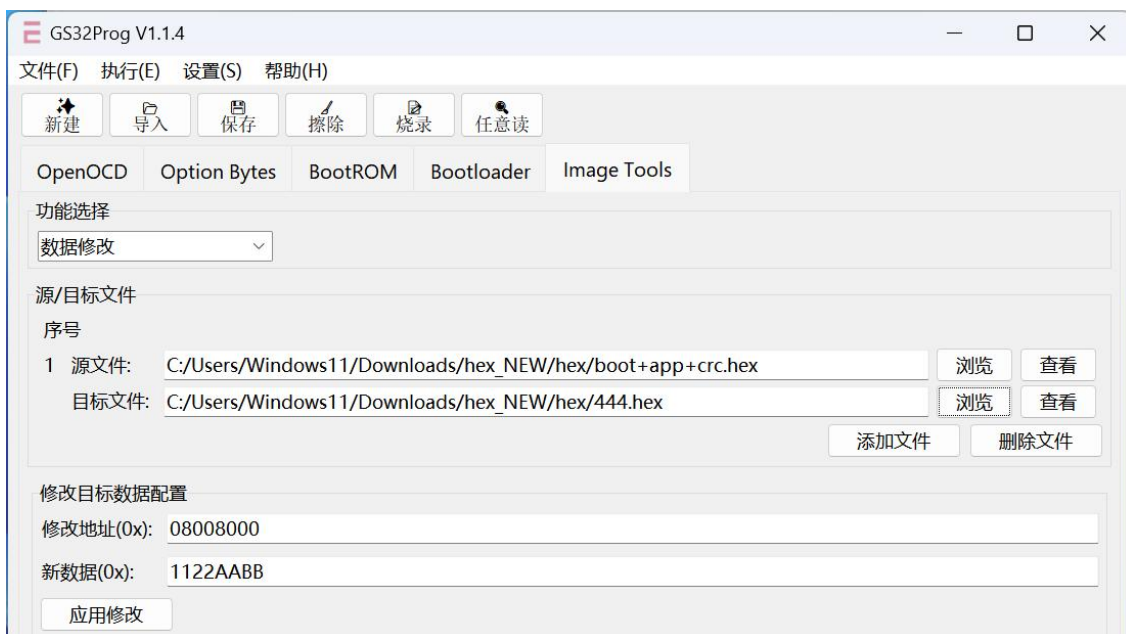


图 6.3-1 数据修改界面

数据格式: 新数据必须是十六进制字符串（可带 0x 前缀），长度必须为偶数（按字节写入）。

6.4 文件合并

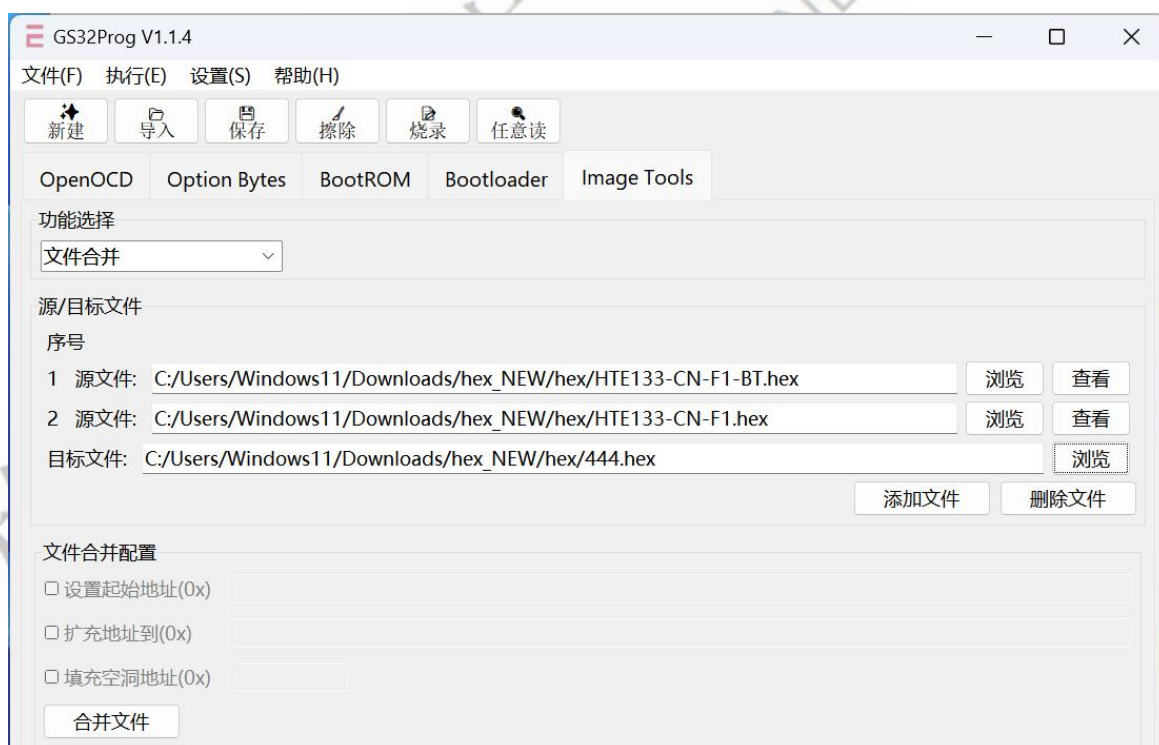


图 6.4-1 文件合并界面

文件合并仅支持 HEX 文件，至少需要两个源文件。

以下是增加的可选功能，如不选用，正常对 HEX 文件进行合并。

可选功能：生成地址连续的 HEX 文件，空洞内容被填充。

1. 设置起始地址：即设置合并后 hex 文件的起始地址，并不是移动文件内容到此地址，是从此地址开始填充直到源文件有内容的地址
2. 扩充地址到：即设置合并后 hex 文件的结束地址
3. 填充空洞地址：自定义填充的值，生成连续地址的 HEX

6.5 CRC 校验

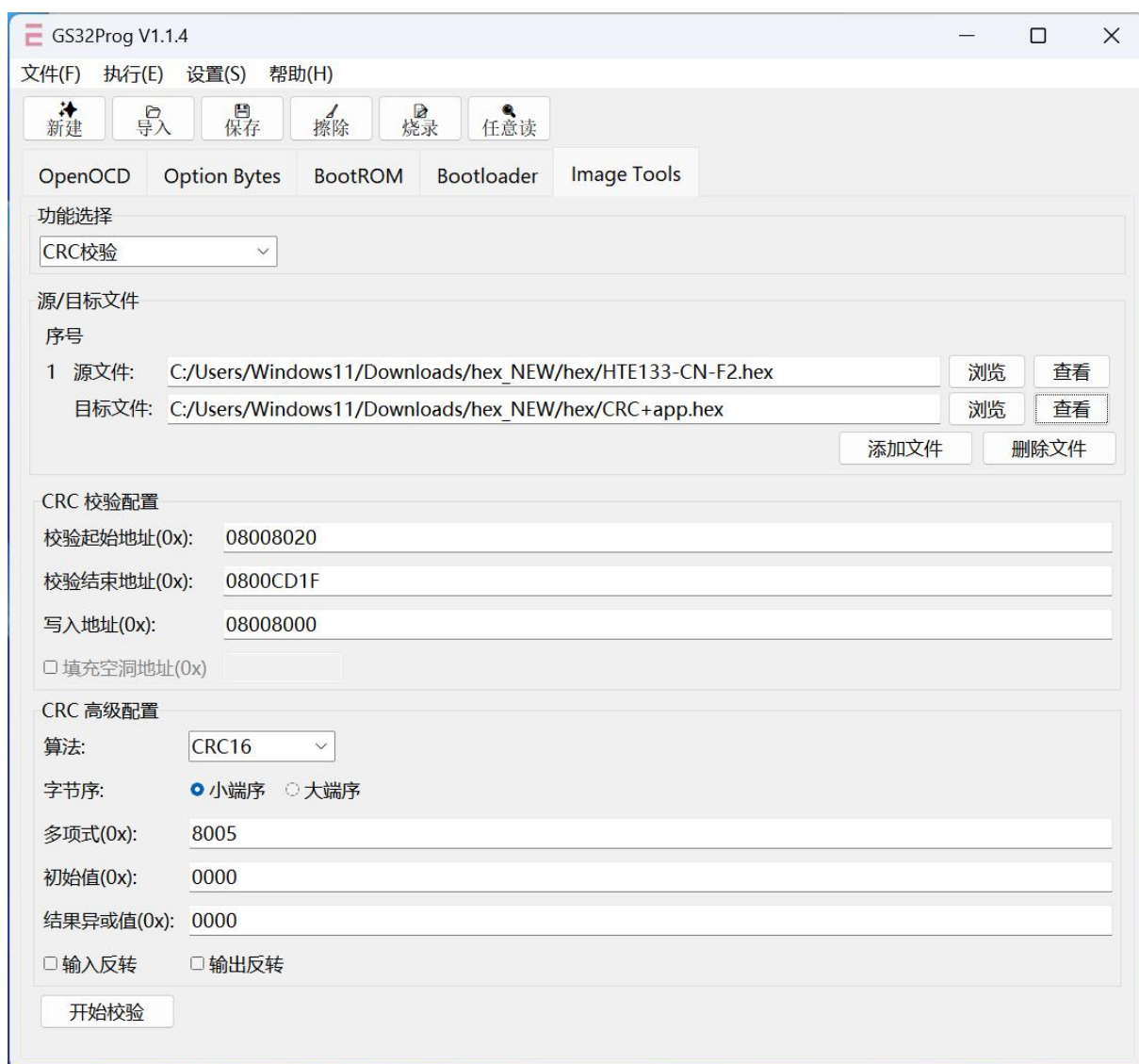


图 6.5-1 CRC 校验

对文件指定区域的内容计算出 CRC 校验和，再写入指定地址。

7 修订历史

版本	时间	内容描述	审核人	批准人
v0.1	2024.7.6	初始版本，依据公司规定格式重新编写	Jason	
v0.2	2024.7.10	增加 OptionBytes 说明	Jason	
V0.3	2024.8.2	增加 Image Tools、增加命令行	Jason	
V0.4	2024.8.22	修改烧录信息、增加命令行	Jason	
V1.0	2025.12.5	更新截图、修改 openocd 章节顺序、增加 bootrom 例程内容、更新 option bytes 章节、修改命令行的说明增加可读性	Jason	
V1.1.4	2026.1.14	修改 option bytes 章节选项，将原“Hex Modification”删除,将此页功能与“Image Tools”功能合并，调整各项功能后重新修订此手册	Jason	