

TG71XX_Programmer 命令行使用说明

在命令行模式下，可以通过后接指定的参数来运行本程序以进行烧入指定的 hex（或 hexf）文件，或者将几个指定的文件合并成 hexf 文件（参照 2.6 节相关内容）。另外也可以选择既执行合并又执行烧写（实际执行顺序为先合并后烧写）。如果在命令行下不加参数打开本程序则会直接进入 GUI 模式。

本程序所支持的命令行参数如下（注意：参数除了 -a 要跟在 -r 后面外（具体参见 -a 参数的说明），其他参数理论上没有顺序先后的要求，另外每一个参数和它的缩写形式完全等价，可任意进行相互替换）：

参数	缩写	值	说明
--addr	-a	Preserve 擦除保留的地址段	起始地址-结束地址，中间用英文 '-' 分隔开，地址均为 16 进制格式。
--erase	-e	擦除方式选择	<p>主要支持三个擦除方式，type 分别取值位 0, 1, 2。</p> <p>0: 指擦除方式位 512K 方式擦除，就是从 0x2000~flash size，命令格式：-e 0；</p> <p>1: 根据 hexf 的写入内容，进行擦除，命令格式：-e 1；</p> <p>2: 保留地址段擦除方式，该命令需要组合 -a 命令使用，命令格式：-e 2 -a 4000-4fff。</p> <p>起始地址默认为 0x11000000，例：-a 4000-4fff 表示为保留 0x11004000 ~ 0x11004fff 数据区域。</p> <p>请注意所选烧录工具版本。</p> <p>V2.4.a 版本，参数含义有所区别，见如下解释。该参数不是必要参数，若无该参数则默认为 Flash全擦除 如果有该参数，则表示Erase除该区域之外的Flash区域。参数为Offset，起始地址默认为0x11000000</p> <p>例：-e 4000-4fff 表示为保留 0x11004000 ~ 0x11004fff 数据区域</p>
--write	-w	写入的文件 hex 或 hexf 格式	<p>参数为字符串，可以包含路径，路径有空格需要把整个字符串通过双引号包起来，和选项之间通过空格符号分开</p> <p>例如：-w target.hexf</p> <p>注意：在写入开始前程序默认会进行一次擦除操作！</p>
--uarts	-u	更新波特率	<p>该参数不是必要参数，若无该参数，默认波特率为：115200；波特率值设置有：1500000、1000000、500000、250000、115200、76800、38400、9600 等，根据 uart 配置可进行修改（v2.3.8c 目前只支持 TG7100B 在波特率 1500000、1000000 下的烧写）</p> <p>例：-u 500000</p>

--Run	-R	Base run address (1FFF4000--PHY6202, 1FFF4800--PHY6212)	该参数是必要参数，需要修改该值对不同的芯片类型的固件进行烧写；通过该命令进行烧写工具中 configuration—base run address 中的修改，对 hex 文件中的 bin 文件进行烧写的起始地址 例：-R 1FFF4000（对应 6202 的芯片），-R 1FFF4800（对应 6212 的芯片）
--Port	-P	获取的 Port 的 name	多个 uart 与主机相连，获取多个 uart 的 COM 口的名称，对指定的 COM 口的开发板进行固件烧写 例如：-P COM3
--config	-f	配置文件.csv	该参数不是必要参数，主要是用来烧写 MAC 地址，在 csv 文件中设置多行 12 位 mac 地址，通过 4000 和 4004 地址进行 mac 地址值的烧写（备注：有其他配置需求，也可自行添加，获取对应的写入地址 address 和 value） 例如：-f *.csv
--line	-l	配置文件的 line 值设定	该参数不是必要参数，配置文件中可输入多行信息，每次烧写，可以通过改变 line 的值来改变 mac 地址或者其他参数设置 例如：-l 3（这里是英文字符 l（小写 L），不是数字 1）
--help	-h 或 -?	无	显示帮助

注意：

- 1.烧写的 TG71XX_Programmer.exe 工具、烧写固件、配置文件.csv 等需要在同一目录下
- 2.log 文件的保存时以追加的形式写入的。多次烧写流程会保存在一个 log 文件中，默认产生的 log 文件名为 zUpgrade.log

基于 V2.4.b 版本的示例

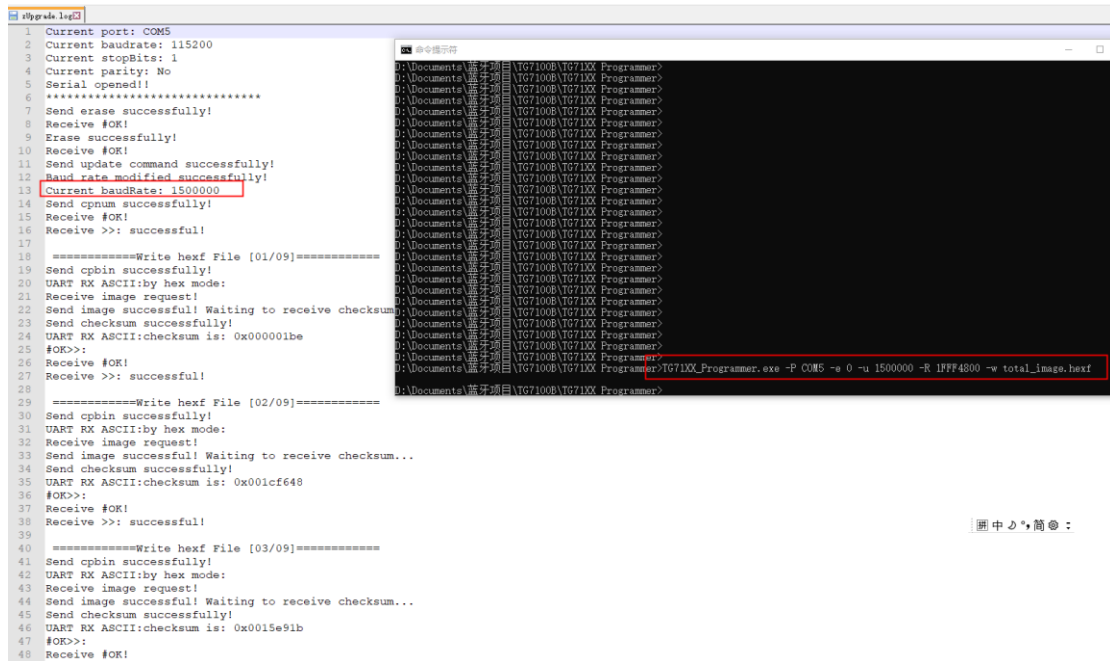
示例 1（全擦模式固件烧录）：

指令：**TG71XX_Programmer.exe -P COM5 -e 0 -u 1500000 -R 1FFF4800 -w**

total_image.hexf

描述：指定 Uart 口为 COM5，设置波特率为 1500000，设置 run address 为 1FFF4800，total_image.hexf（和主程序在同一目录）写入到芯片中（芯片需要通过串口和电脑连接），写入前会自动执行擦除。（注意：TM 拉高，P24 和 P25 拉低）

效果截图

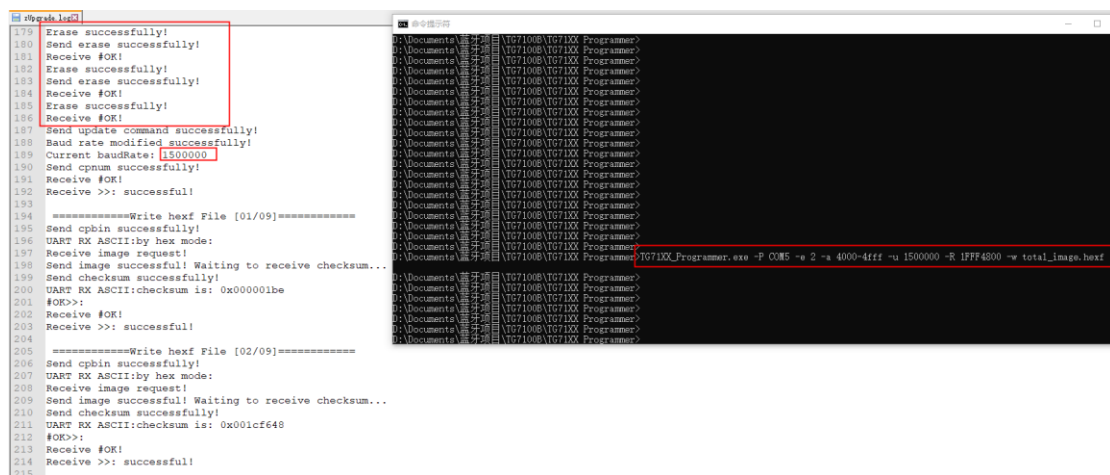


示例 2（保留数据区模式固件烧录）：

指令：**TG71XX_Programmer.exe -P COM5 -e 2 -a 4000-4fff -u 1500000 -R 1FFF4800 -w total_image.hexf**

描述：指定Uart口为COM5，设置波特率为1500000，设置run address为1FFF4800，保留数据区0x11004000~0x11004fff，将total_image.hexf（和主程序在同一目录）写入到芯片中（芯片需要通过串口和电脑连接），写入前会自动执行除保留区外的Flash擦除。

效果截图



注：

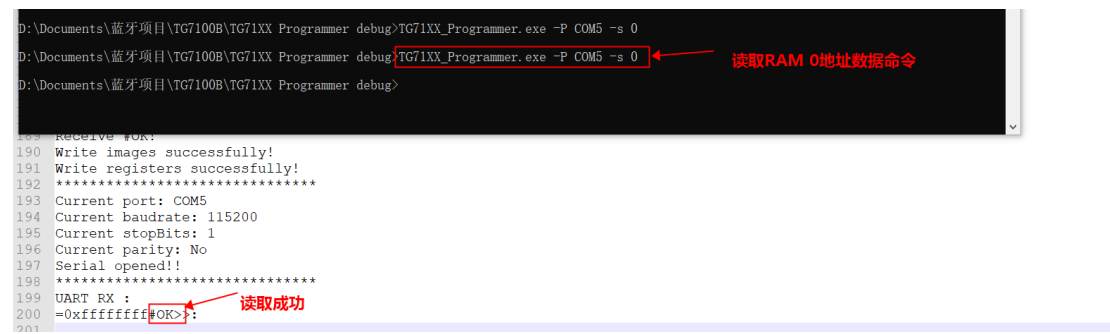
由于PC机串口驱动程序限制，存在不支持波特率为1500000的问题，此时可以降低波特率来适配，例如设置波特率为115200进行烧录。

示例 3（读取 RAM 0 地址的数据）：

指令：**TG71XX_Programmer.exe -P COM5 -s 0**

描述：指定 Uart 口为 COM5，默认波特率为 115200， 读取 RAM 0 地址的值。通常用于判断芯片是否已上电并正常连接。

效果截图



```
D:\Documents\蓝牙项目\TG7100B\TG71XX Programmer debug>TG71XX_Programmer.exe -P COM5 -s 0
D:\Documents\蓝牙项目\TG7100B\TG71XX Programmer debug>TG71XX_Programmer.exe -P COM5 -s 0
D:\Documents\蓝牙项目\TG7100B\TG71XX Programmer debug>

188 Receive OK!
190 Write images successfully!
191 Write registers successfully!
192 *****
193 Current port: COM5
194 Current baudrate: 115200
195 Current stopBits: 1
196 Current parity: No
197 Serial opened!!
198 *****
199 UART RX :
200 =0xffffffff#OK>:
201
```

读取RAM 0地址数据命令

读取成功