

Telink

Telink Burning and Debugging Tool (BDT)

用户指南

AN-18101201-C6

Ver1.5.0

2023.08.09

Keyword

BDT

Brief

本文档为 Telink Burning and Debugging Tool (BDT) 的用户指南。

Published by
Telink Semiconductor

Bldg 3, 1500 Zuchongzhi Rd,
Zhangjiang Hi-Tech Park, Shanghai, China

© Telink Semiconductor
All Rights Reserved

Legal Disclaimer

This document is provided as-is. Telink Semiconductor reserves the right to make improvements without further notice to this document or any products herein. This document may contain technical inaccuracies or typographical errors. Telink Semiconductor disclaims any and all liability for any errors, inaccuracies or incompleteness contained herein.

Copyright © 2023 Telink Semiconductor (Shanghai) Co., Ltd.

Information

For further information on the technology, product and business term, please contact Telink Semiconductor Company www.telink-semi.com

For sales or technical support, please send email to the address of:

telinksales@telink-semi.com

telinksupport@telink-semi.com

修订历史

版本	修改内容
V1.0.0	初次发布。
V1.1.0	支持多设备；增加了时间戳；路径空间问题；路径缓冲区溢出等。
V1.2.0	增加工具指南章节；更新 2.4 节 Cache 解析。
V1.3.0	新增 4.4 节 RISC-V Tool。
V1.4.0	新增 4.5 节 EMI Tool；新增 4.6 节 PA Configure Tool；新增 4.7 节 EVK Offline Tool。
V1.5.0	新增 4.8 节 Secure Boot Tool。

Contents

修订历史	3
1 简介	12
1.1 功能概述	12
2 操作指南	13
2.1 下载固件	13
2.1.1 连接硬件	13
2.1.2 连接设备	16
2.1.3 通过“USB”模式将固件下载到“Flash”中	19
2.1.4 通过“EVK”模式将固件下载到“Flash”中	28
2.1.5 将固件下载到“SRAM”或“OTP”中	35
2.1.6 多地址下载	36
2.1.7 OTP 启动信息下载	42
2.2 擦除 Flash 扇区	45
2.3 Flash 去除保护	48
2.4 Cache 解析	49
2.5 Activate MCU	52
2.6 Debug	53
2.6.1 访问存储器	55
2.6.1.1 从内存空间读取数据	55
2.6.1.2 将数据写入内存空间	58
2.6.2 操作变量列表	62
2.6.2.1 更新变量列表	63
2.6.2.2 对变量进行排序	63
2.6.2.3 查看超过 4 字节的变量	64
2.6.2.4 修改变量值	65
2.6.3 Debug MCU	68
2.6.3.1 Run MCU	68
2.6.3.2 Pause MCU	69
2.6.3.3 Step MCU	70
2.6.3.4 Trace PC	75
2.6.3.5 Stall MCU	79
2.6.3.6 Start MCU	80
2.7 单线同步	81
2.8 命令行	82
2.8.1 单命令行	82
2.8.2 多命令行	84
2.9 日志窗口	85
2.10 帮助	88
2.11 默认配置	93
3 升级固件	97
4 工具指南	101
4.1 BQB Tool	101
4.1.1 功能描述 (BQB Tool)	101

4.1.2	界面组成及使用说明 (BQB Tool)	101
4.1.2.1	芯片的选择	102
4.1.2.2	固件的保存和下载	102
4.1.2.3	收发包测试	102
4.2	Jtag Tool	103
4.2.1	功能描述 (Jtag Tool)	103
4.2.2	界面组成及使用说明 (Jtag Tool)	103
4.2.2.1	Jtag 设备的驱动安装	103
4.2.2.2	连接调试设备	105
4.2.2.3	下载功能	106
4.2.2.4	擦除功能	106
4.2.2.5	读写功能	106
4.2.2.6	调试功能	106
4.3	TWS Tool	107
4.3.1	Download Tool	107
4.3.1.1	功能描述 (Download Tool)	107
4.3.1.2	界面介绍	107
4.3.1.3	使用说明	107
4.3.2	EQ Tool	108
4.3.2.1	功能描述 (EQ Tool)	108
4.3.2.2	界面组成	108
4.3.2.3	使用说明	109
4.4	RISC-V Tool	109
4.4.1	读存储器	109
4.4.2	写存储器	110
4.4.3	固件下载	111
4.4.4	VCD 功能	114
4.4.5	命令行说明	117
4.4.5.1	读命令	117
4.4.5.2	写命令	118
4.4.5.3	擦 flash 命令	119
4.4.5.4	保存所读数据	120
4.4.6	日志清除及保存	121
4.5	EMI Tool	123
4.5.1	功能描述 (EMI Tool)	123
4.5.2	界面组成及使用说明 (BQB Tool)	123
4.5.2.1	芯片的选择	127
4.5.2.2	固件的保存和下载	127
4.5.2.3	EMI 测试	128
4.5.2.4	非信令测试	128
4.5.2.5	RF 电流测试	128
4.6	PA Configure Tool	128
4.6.1	功能描述 (PA Configure Tool)	128
4.6.2	界面组成及使用说明 (PA Configure Tool)	128
4.6.2.1	配置生成 PA 工作文件	129
4.7	EVK Offline Tool	129
4.7.1	功能描述 (EVK Offline Tool)	129

4.7.2	界面组成及使用说明 (EVK Offline Tool)	129
4.7.2.1	更新工具固件	131
4.7.2.2	下载用户固件	131
4.8	Secure Boot Tool	131
4.8.1	界面组成及使用说明	131
4.8.1.1	Mode Selection	132
4.8.1.2	Firmware Encryption	132
4.8.1.3	Secure Debug	132
4.8.1.4	Run Description Address	132
4.8.1.5	Pubkey Hash	132
4.8.1.6	Root Key	133
4.8.1.7	Debug Plaintext	134
4.8.1.8	Run Code	136
4.8.1.9	Run Code description	137
4.8.1.10	Download-All	137
4.8.1.11	Download-Flash	137
4.8.1.12	Re-enable Debug-interface	137
4.8.1.13	Read-Info	138
4.8.1.14	Save	139
4.8.2	常用安全机制组合和工具的使用步骤说明	139
4.8.2.1	Secure Boot + Secure Debug	140
4.8.2.2	Firmware Encryption + Secure Debug	140
4.8.3	Secure Boot + Firmware Encryption + Secure Debug	141
5	功能支持列表	143

List of Figures

Figure 1.1	主界面	12
Figure 2.1	直接将目标板与 PC 连接	13
Figure 2.2	将 “Burning EVK” 与 PC 连接	14
Figure 2.3	通过 USB 连接 “Burning EVK” 和目标板	15
Figure 2.4	通过 Swire 连接 “Burning EVK” 和目标板	16
Figure 2.5	烧录和调试工具	17
Figure 2.6	查看可用设备	18
Figure 2.7	选择指定的设备	19
Figure 2.8	选择芯片类型	20
Figure 2.9	选择下载模式为 “USB”	21
Figure 2.10	选择起始偏移地址	22
Figure 2.11	打开文件选择窗口	23
Figure 2.12	选择目标文件并打开新的文件路径	24
Figure 2.13	在 “Reopen” 子菜单中直接选择目标文件路径	25
Figure 2.14	连接状态指示	26
Figure 2.15	通过 USB 模式将固件下载到 Flash 中	27
Figure 2.16	手动复位 MCU	27
Figure 2.17	自动复位 MCU	28
Figure 2.18	选择芯片类型	29
Figure 2.19	选择下载模式为 “EVK”	30
Figure 2.20	打开 “Setting” 对话框	31
Figure 2.21	选择目标文件	31
Figure 2.22	选择 bin 文件	32
Figure 2.23	检查连接状态	32
Figure 2.24	通过 “Burning EVK” 模式将固件下载到 Flash 中	33
Figure 2.25	手动复位 MCU	34
Figure 2.26	自动复位 MCU	35
Figure 2.27	通过 “auto mode” 将固件下载到 “SRAM”	36
Figure 2.28	选择目标空间	37
Figure 2.29	打开 “Multi-address Download” 界面	38
Figure 2.30	添加固件文件并设置目标地址	38
Figure 2.31	添加固件	39
Figure 2.32	显示 bin 文件路径	39
Figure 2.33	设置偏移地址	40
Figure 2.34	显示计算结果	40
Figure 2.35	添加固件 2 的路径	40
Figure 2.36	“Download” 按钮	41
Figure 2.37	开始下载固件 1	41
Figure 2.38	开始下载固件 2	41
Figure 2.39	复位	42
Figure 2.40	Ist 文件	43
Figure 2.41	芯片选择	44
Figure 2.42	OTP 启动信息下载	45
Figure 2.43	选择起始地址	46

Figure 2.44	设置擦除大小	47
Figure 2.45	从 0x000000 起始地址开始擦除 512kB 的 flash 空间	48
Figure 2.46	Flash 去除保护	49
Figure 2.47	Bin 文件选择	50
Figure 2.48	Cache 数据正确	51
Figure 2.49	Cache 数据出错	52
Figure 2.50	Activate MCU	53
Figure 2.51	“Memory Access” 界面	54
Figure 2.52	选择 MCU 类型和通信模式	55
Figure 2.53	设置操作目标和大小	56
Figure 2.54	设置起始地址并开始读取	57
Figure 2.55	保存读取的数据	58
Figure 2.56	内存访问窗口	59
Figure 2.57	设置操作大小	60
Figure 2.58	起始地址的偏移量	61
Figure 2.59	设置数据并开始写入	62
Figure 2.60	情况 1	62
Figure 2.61	情况 2	62
Figure 2.62	更新变量列表	63
Figure 2.63	按名称对变量排序	64
Figure 2.64	读取大于 4 字节的变量值	65
Figure 2.65	双击指定的 “Value” 框	66
Figure 2.66	修改变量值	67
Figure 2.67	将数据写入指定地址	68
Figure 2.68	Run MCU	69
Figure 2.69	Pause MCU	70
Figure 2.70	在 “Single step” 模式下 Step MCU	71
Figure 2.71	在 “continuous” 模式下 Step MCU	72
Figure 2.72	“xxx.lst” 文件	72
Figure 2.73	选择 “interp.lst”	73
Figure 2.74	在 “Single step” 模式下选择 “interp.lst” Step MCU	74
Figure 2.75	在 “continuous” 模式下选择 “interp.lst” Step MCU	75
Figure 2.76	在 “Single step” 模式下 Trace PC	76
Figure 2.77	在 “Single step” 模式下选择 “interp.lst” Trace PC	77
Figure 2.78	将 “Single step” 模式切换到 “continuous” 模式	78
Figure 2.79	在 “continuous” 模式下选择 “interp.lst” Trace PC	79
Figure 2.80	Stall MCU	80
Figure 2.81	Start MCU	81
Figure 2.82	设置 Swire 同步速度	82
Figure 2.83	打开命令行输入窗口	83
Figure 2.84	输入单命令行	84
Figure 2.85	输入多个命令	85
Figure 2.86	打开 usb 日志窗口	86
Figure 2.87	显示输出字符	87
Figure 2.88	退出 “usb log” 模式	88
Figure 2.89	命令行	89
Figure 2.90	显示所有命令	90

Figure 2.91	用户指南	91
Figure 2.92	打开帮助文档	92
Figure 2.93	About Telink	93
Figure 2.94	获取有关 Telink 的信息	93
Figure 2.95	"config"文件夹	94
Figure 2.96	MCU	94
Figure 2.97	通信方式	94
Figure 2.98	Objection	95
Figure 2.99	大小	95
Figure 2.100	下载地址	95
Figure 2.101	擦除地址	95
Figure 2.102	页面显示	95
Figure 2.103	按钮	96
Figure 2.104	路径	96
Figure 3.1	"Upgrade EVK" 窗口	97
Figure 3.2	检查当前固件版本	98
Figure 3.3	加载新固件	99
Figure 3.4	将固件升级到 EVK 中	100
Figure 4.1	BQB 工具固件下载界面	101
Figure 4.2	BQB 工具测试界面	102
Figure 4.3	Jtag 工具界面	103
Figure 4.4	Jtag 设备	104
Figure 4.5	Jtag 设备驱动已安装	104
Figure 4.6	Jtag 设备驱动未安装	105
Figure 4.7	TWS_Tool/Download_Tool 界面	107
Figure 4.8	TWS_Tool/EQ_Tool 界面	108
Figure 4.9	读存储器	110
Figure 4.10	写存储器	111
Figure 4.11	选择固件	112
Figure 4.12	擦除 flash	113
Figure 4.13	下载固件	114
Figure 4.14	选择 header 文件	115
Figure 4.15	启动 VCD 功能	116
Figure 4.16	查看状态	117
Figure 4.17	读命令	118
Figure 4.18	写命令	119
Figure 4.19	擦 flash 命令	120
Figure 4.20	保存所读数据	121
Figure 4.21	清除日志	122
Figure 4.22	保存日志	123
Figure 4.23	EMI 工具固件配置下载界面	124
Figure 4.24	EMI 工具 EMI 测试界面	125
Figure 4.25	EMI 工具非信令测试界面	126
Figure 4.26	EMI 工具 RF 电流测试界面	127
Figure 4.27	PA Configure Tool 界面	129
Figure 4.28	EVK Offline Tool 工具固件界面	130
Figure 4.29	EVK Offline Tool 用户固件界面	130

Figure 4.30	Secure Boot 工具界面	131
Figure 4.31	公钥哈希文件错误	132
Figure 4.32	导入 Root Key 文本	133
Figure 4.33	Root Key 格式错误 1	134
Figure 4.34	Root Key 格式错误 2	134
Figure 4.35	导入 Debug Plaintext 文本	135
Figure 4.36	Debug Plaintext 格式错误 1	136
Figure 4.37	Debug Plaintext 格式错误 2	136
Figure 4.38	重新使能调试接口	138
Figure 4.39	读取信息	139
Figure 4.40	key 文件夹路径	139
Figure 4.41	Secure Boot + Secure Debug 烧录步骤	140
Figure 4.42	Firmware Encryption + Secure Debug 烧录步骤	141
Figure 4.43	Secure Boot + Firmware Encryption + Secure Debug 烧录步骤	142
Figure 5.1	功能支持列表	143

Telink Semiconductor

List of Tables

Table 2.1	Memory Access 介绍	54
-----------	----------------------------	----

Telink Semiconductor

1 简介

“Telink Burning and Debugging Tool (BDT)” 适用于基于 Telink SoC（包括 8267、8269、8266、8232、8233、8366、8368、B85M 系列以及 B91M 系列）的应用开发。

这份文档介绍了如何使用 “Telink Burning and Debugging Tool (BDT)”。

1.1 功能概述

在 SDK 开发过程中，通过使用 “Telink Burning and Debugging Tool (BDT)”，可以通过 USB 模式或者 “Burning EVK”（EVK）将固件直接下载到目标板（例如开发板）。

它主要的功能包括：擦除 flash 扇区，下载固件，通信失败时激活 MCU，访问包括 Flash/Core/Analog/OTP 在内的内存空间，读/写全局变量和查看 USB 日志。

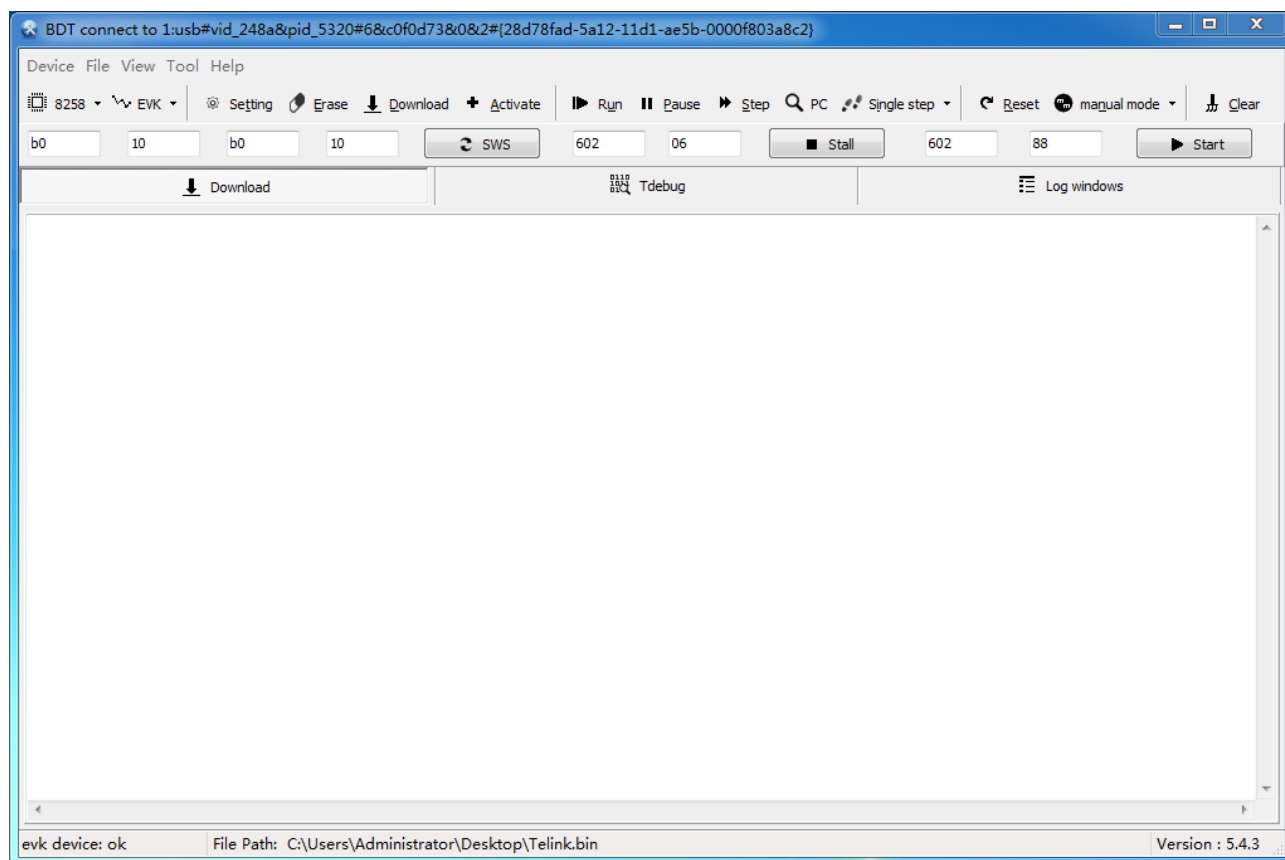


Figure 1.1: 主界面

2 操作指南

2.1 下载固件

2.1.1 连接硬件

在使用“Telink Burning and Debugging Tool (BDT)”之前，需要连接目标板和 PC。

有两种方法连接目标板和 PC，如下所示。

方法 1：通过 USB 直接连接目标板和 PC。

这种方法只适用于具有 USB 接口的目标板和支持 USB 功能的 MCU，例如 dongle 板。



Figure 2.1: 直接将目标板与 PC 连接

方法 2：通过 Telink “Burning EVK” TLSR8266BR56 连接目标板和 PC。

(1) 通过 USB 线将“Burning EVK”与 PC 连接。观察“Burning EVK”的指示灯，指示灯闪烁一次，表明“Burning EVK”与 PC 连接成功。



Figure 2.2: 将 “Burning EVK” 与 PC 连接

(2) 连接 “Burning EVK” 和目标板。

有两种方法连接 “Burning EVK” 和目标板。

- a. 通过 USB 接口直接连接目标板和 “Burning EVK”，如下图所示。这种连接方式只适用于有 USB 接口的目标板和支持 USB 功能的 MCU，如 dongle 板。



Figure 2.3: 通过 USB 连接 “Burning EVK” 和目标板

b. 通过 Swire（单线）接口连接 “Burning EVK” 和目标板，如下图所示。



Figure 2.4: 通过 Swire 连接 “Burning EVK” 和目标板

有关 TLSR8266BR56 的详细信息，请参阅 Telink 文档 “AN_18010500_User Guide for Telink Burning EVK TLSR8266BR56”。

通过 “USB” 方式或 “Burning EVK” 方式将目标板连接到 PC 后，有两种方法可以将固件下载到目标板上，分别对应上述两种硬件连接方式。

2.1.2 连接设备

在将固件下载到目标板之前，请确保 BDT 已找到该设备，如下图所示。

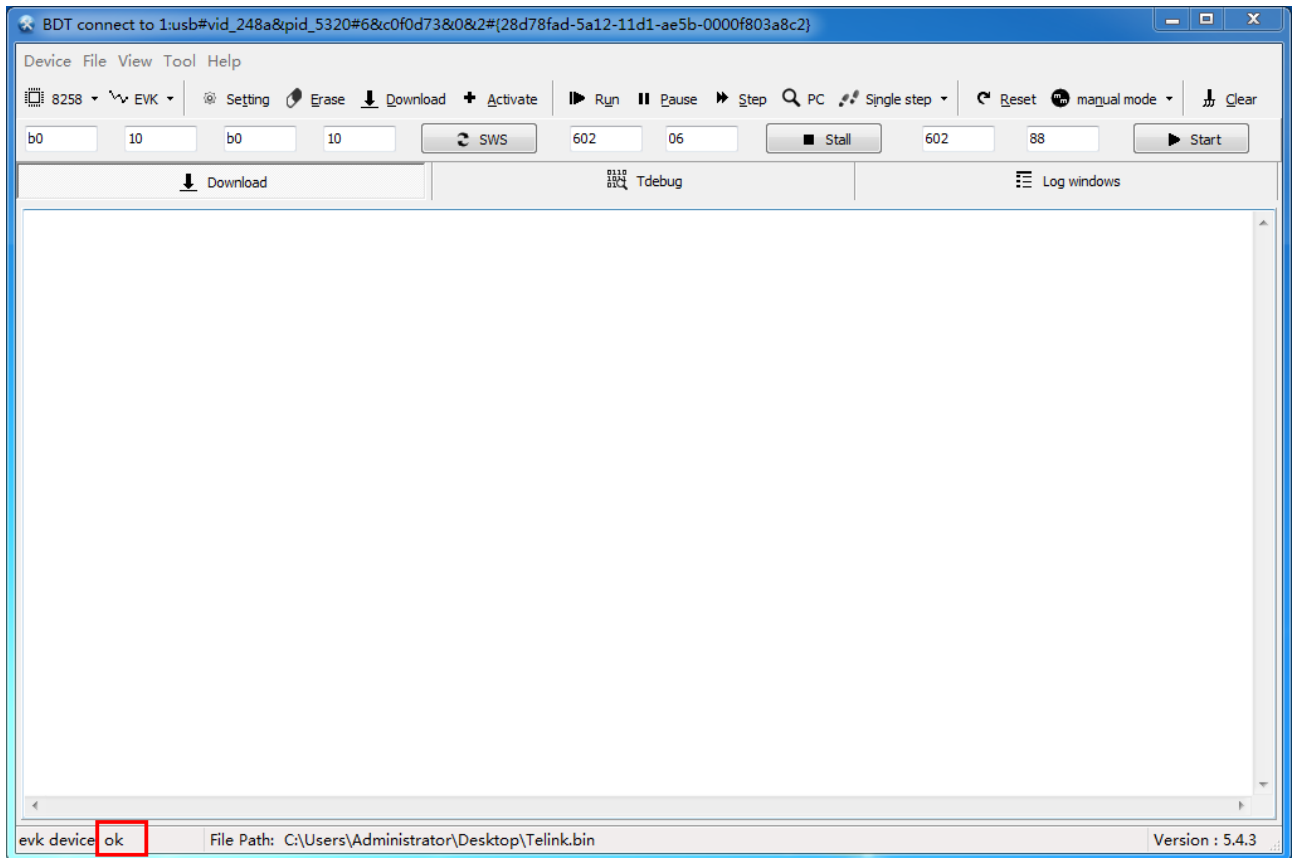


Figure 2.5: 烧录和调试工具

如果 BDT 没有找到设备，如下图所示，可以点击“Refresh”查看可用设备。

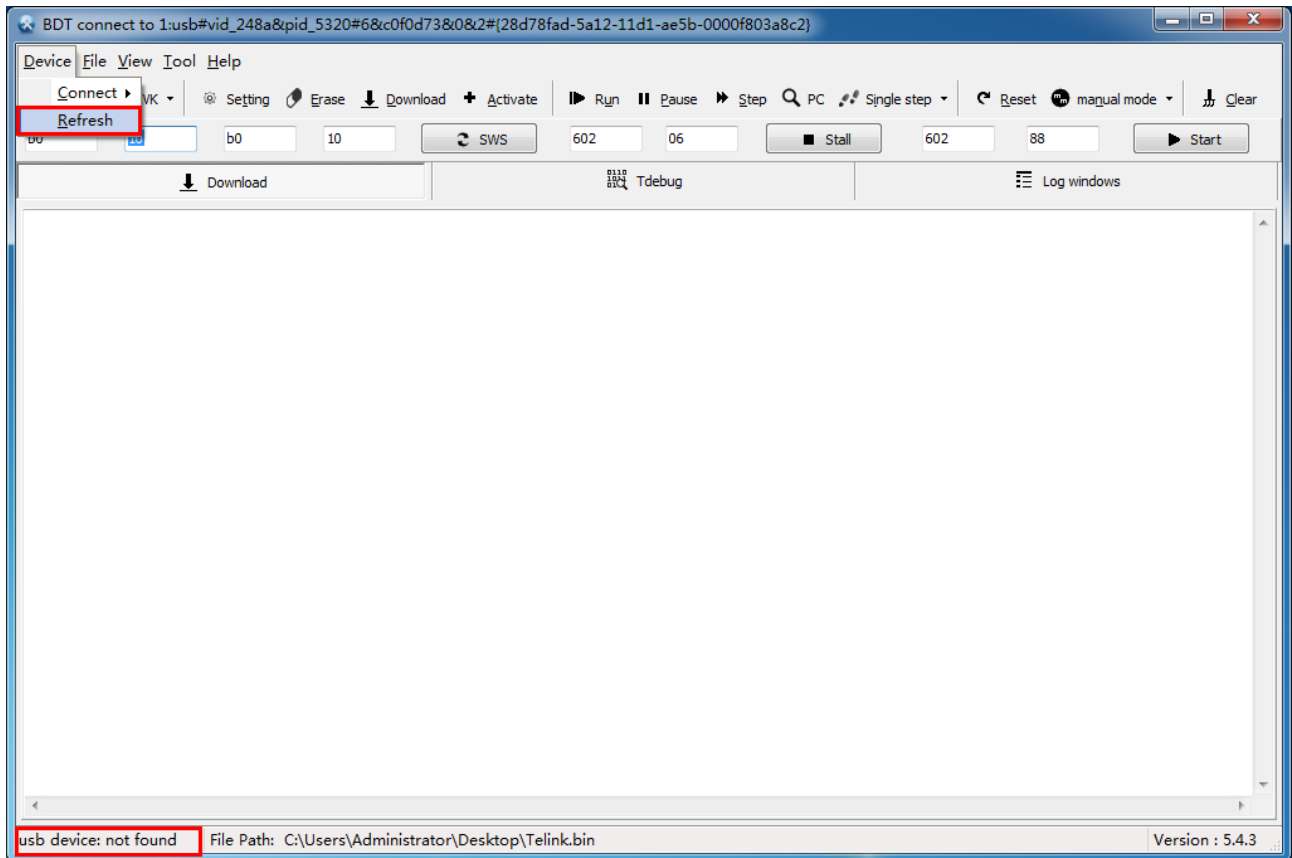


Figure 2.6: 查看可用设备

如果 BDT 发现多个设备，所有设备将会列出，如下图所示。点击“Refresh”按钮后，默认情况下第一个设备将连接到 BDT。如果想要连接其他设备，可以选择指定的设备，如下图所示。

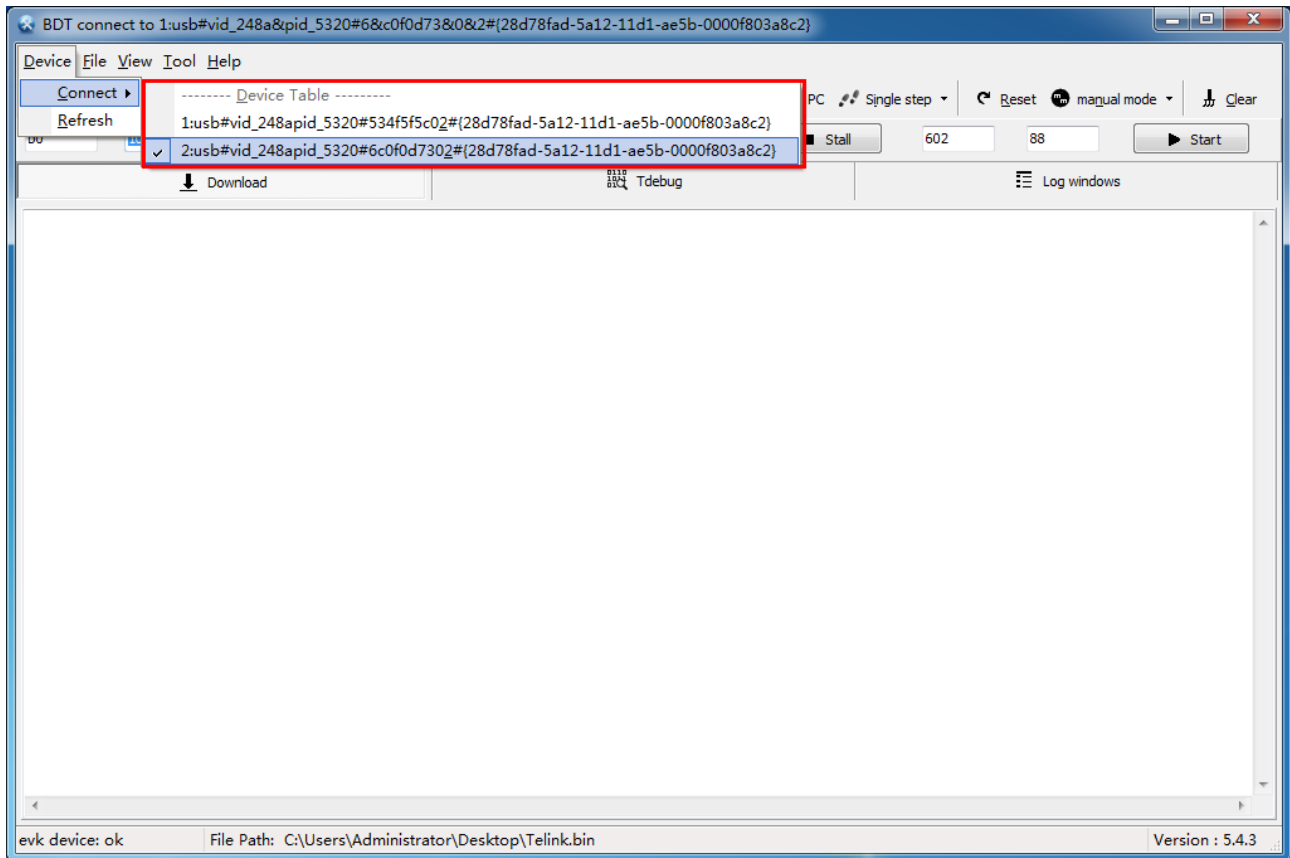


Figure 2.7: 选择指定的设备

2.1.3 通过“USB”模式将固件下载到“Flash”中

在使用“USB”模式下载或调试 MCU 之前，请确保指定的 MCU 支持 USB 功能，并且其 USB 功能可用。详细信息请参阅第 5 章功能支持列表。

接下来，用户可以按照本章指南，通过“USB”模式将固件下载到目标板的特定 flash 空间。

步骤 1: 选择目标板的芯片类型，例如 8258（默认选项），支持以下两种方法。

- 方法 1：使用下拉菜单选择芯片类型。
- 方法 2：按组合快捷键“Ctrl+Q”切换芯片类型。

注意：

所有芯片类型和默认选项都可以在“config.ini”文件中配置。

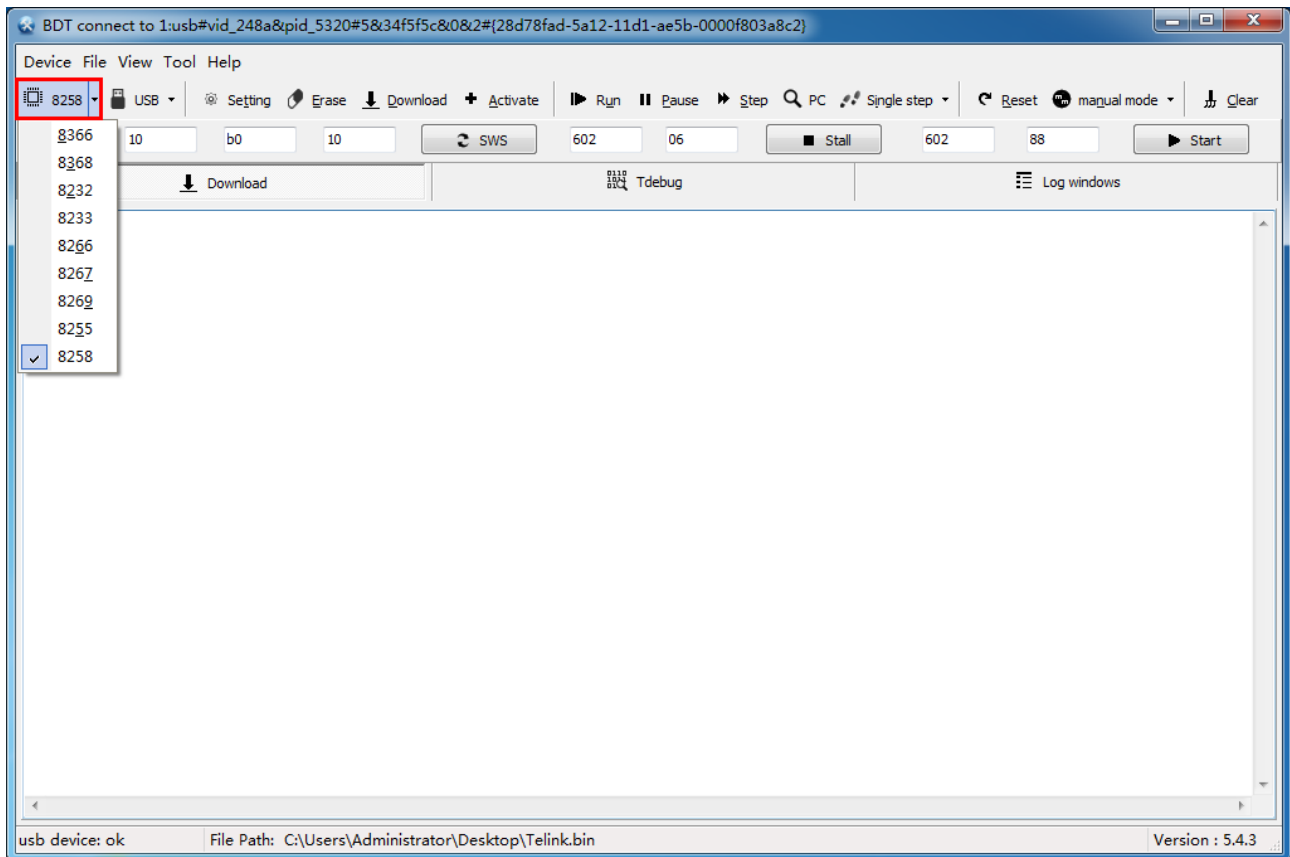


Figure 2.8: 选择芯片类型

步骤 2: 选择下载模式为“USB”，支持以下两种方法。

- 方法 1: 使用下拉菜单选择 USB 下载模式。
- 方法 2: 按组合快捷键“Ctrl+W”切换到 USB 下载模式。每次切换后，下拉菜单将自动显示当前下载模式。

注意:

所有下载模式和默认选项都可以在“config.ini”文件中配置。

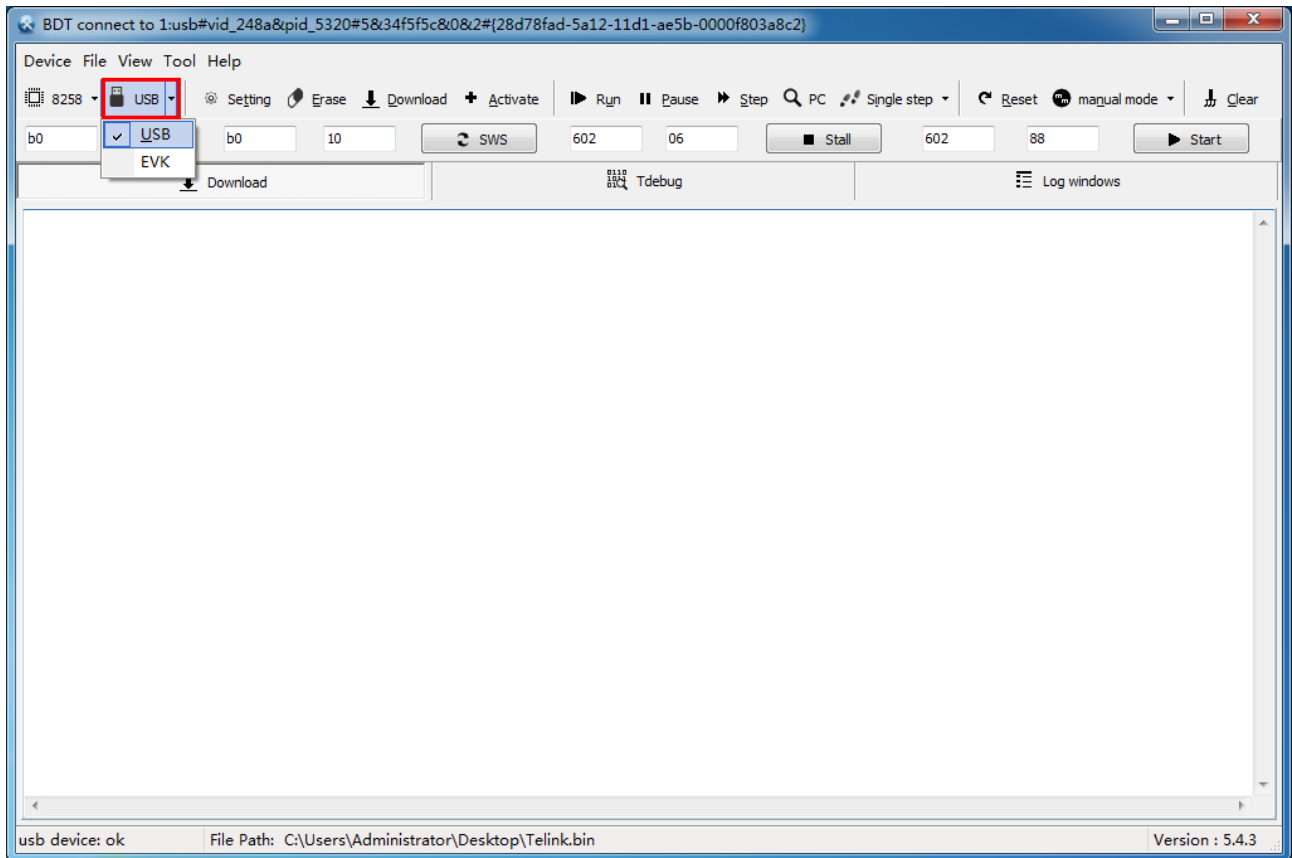


Figure 2.9: 选择下载模式为“USB”

步骤 3: 点击“Setting”按钮，打开设置对话框。选择“Flash”选项。用户可以采用以下任意一种方法，设置目标固件存储的 flash 起始地址偏移量，例如 0x000000（默认选项）。

- 方法 1：使用“Download Addr”的下拉菜单选择可用的偏移选项。
- 方法 2：在“Download Addr”的编辑框中输入偏移地址。

注意：

- 所有设置和默认选项都可以在“config.ini”文件中配置。
- 用户还可以配置“SRAM”/“OTP”设置选项，以便将目标固件下载到 SRAM 或 OTP 中的目标区域。

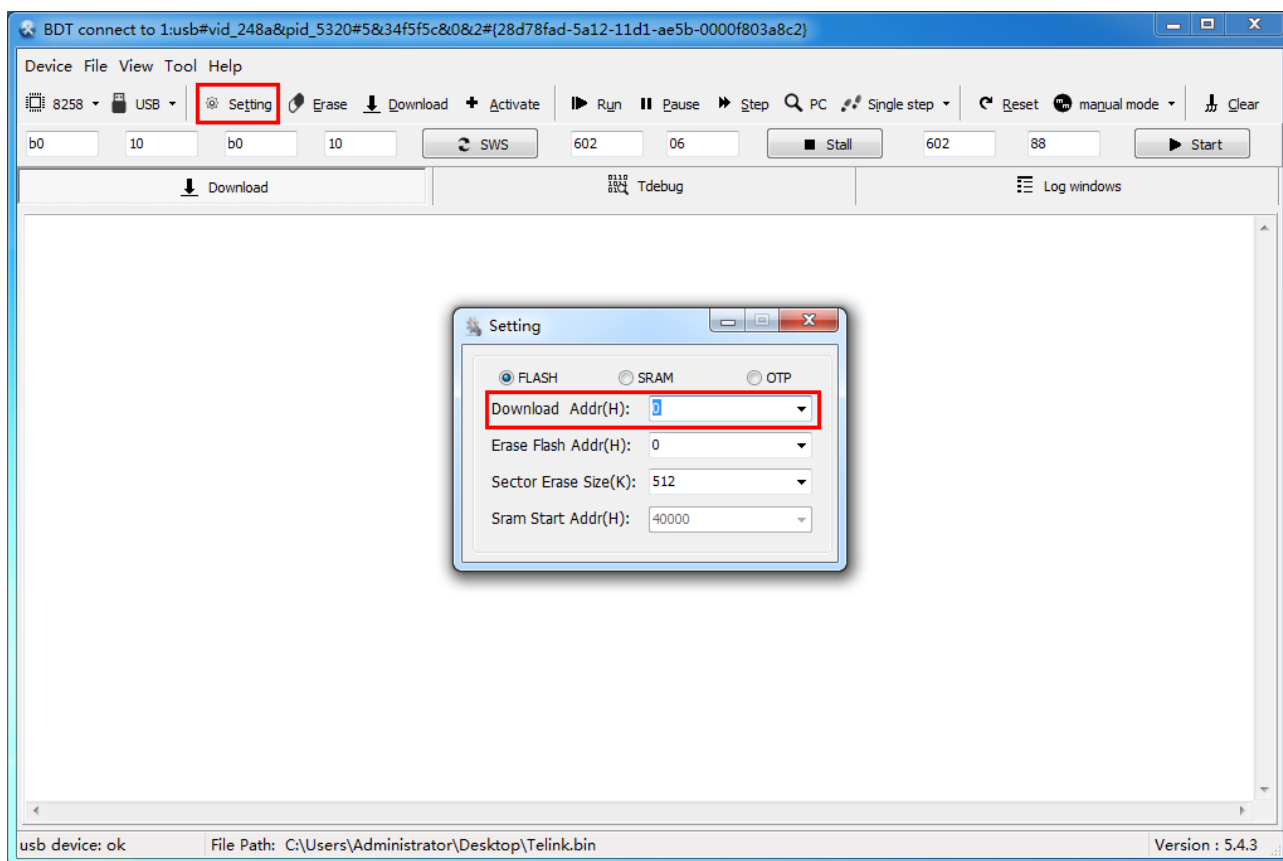


Figure 2.10: 选择起始偏移地址

步骤 4: 选择要下载到目标板的目标固件文件。

(1) 点击 File->Open，打开文件选择窗口。

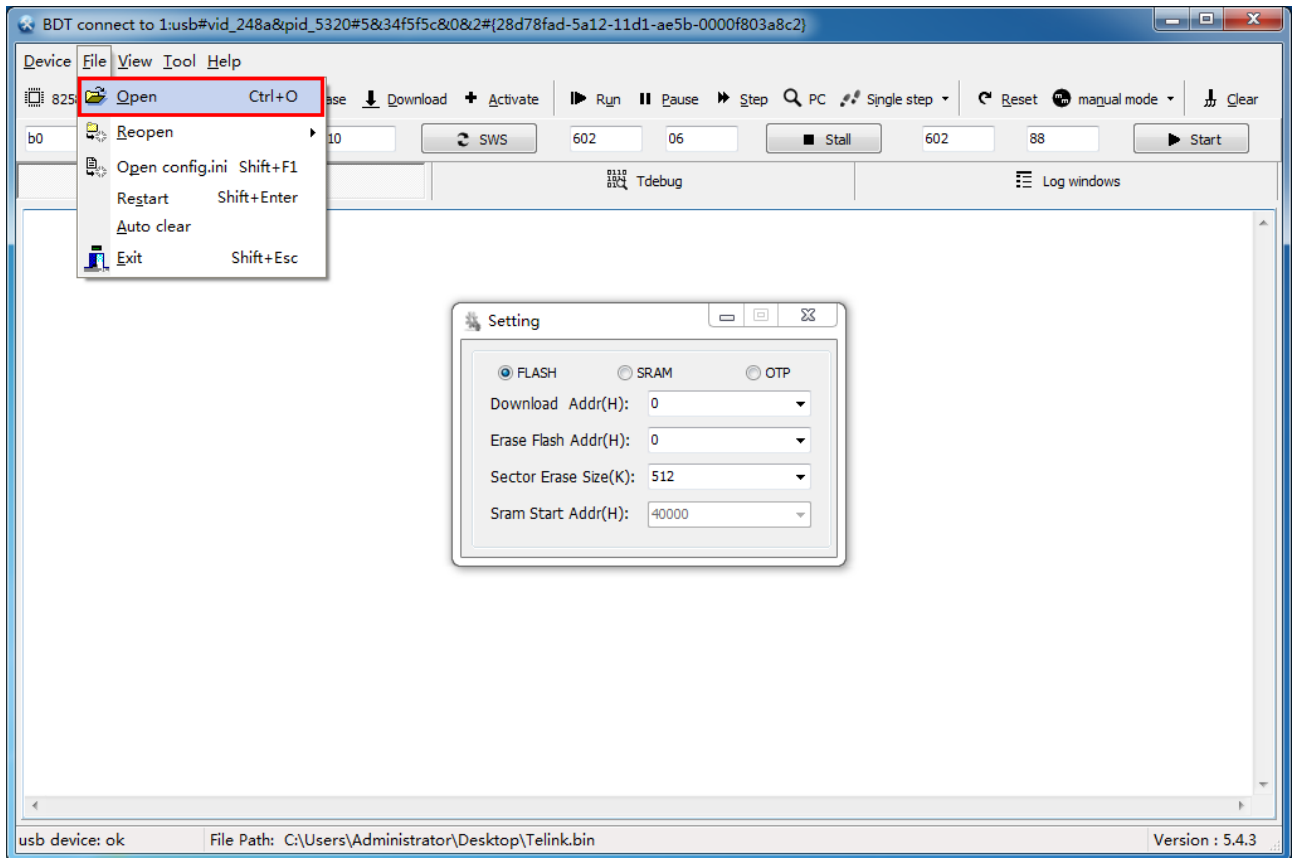


Figure 2.11: 打开文件选择窗口

(2) 在文件选择窗口，选择目标 bin 文件（如 Telink.bin），然后点击“打开”按钮。

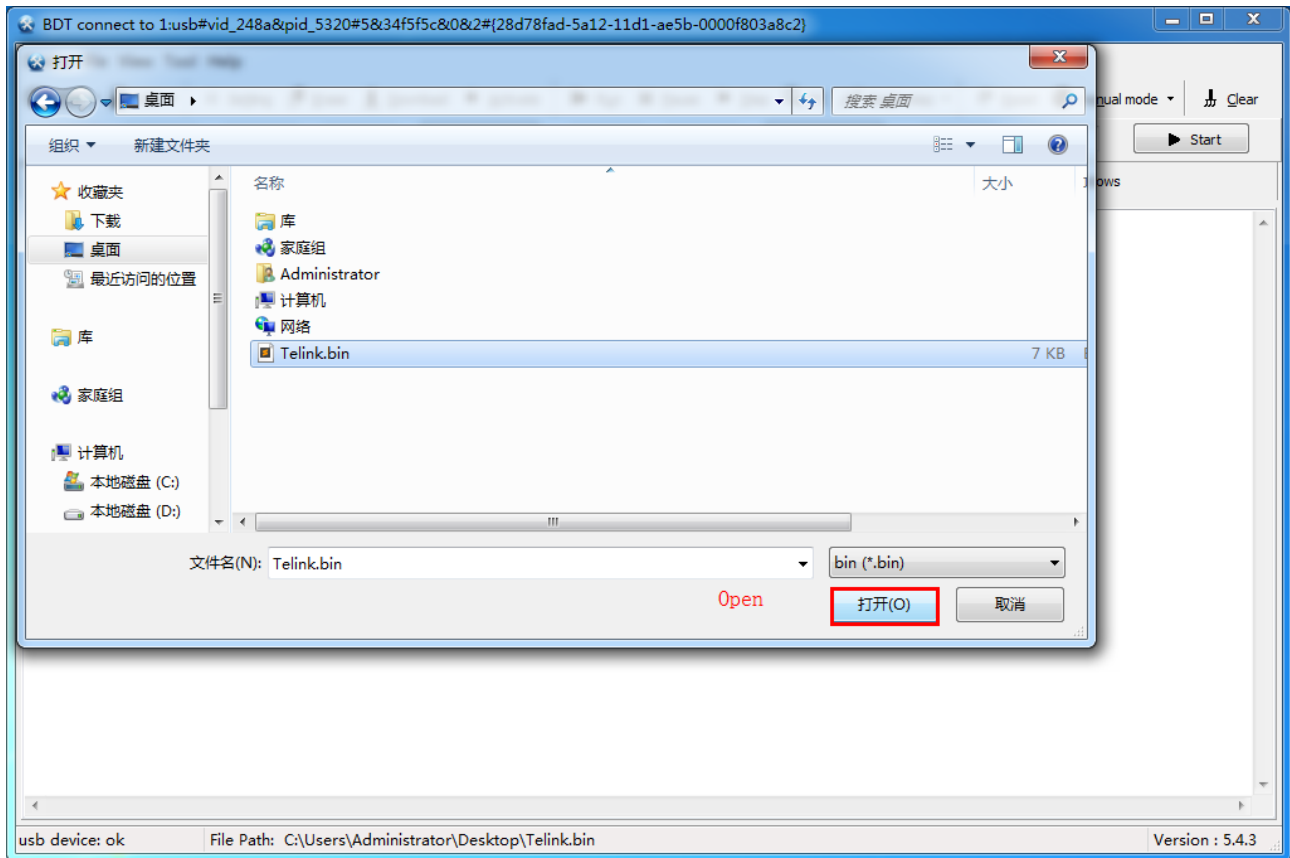


Figure 2.12: 选择目标文件并打开新的文件路径

文件路径将显示在主界面的底部。每次通过文件选择窗口打开新文件路径时，它都会被添加到“File->Reopen”的子菜单中。这样可以简化后续选择固件操作，用户可以直接选择“Reopen”子菜单中可用的目标文件路径。

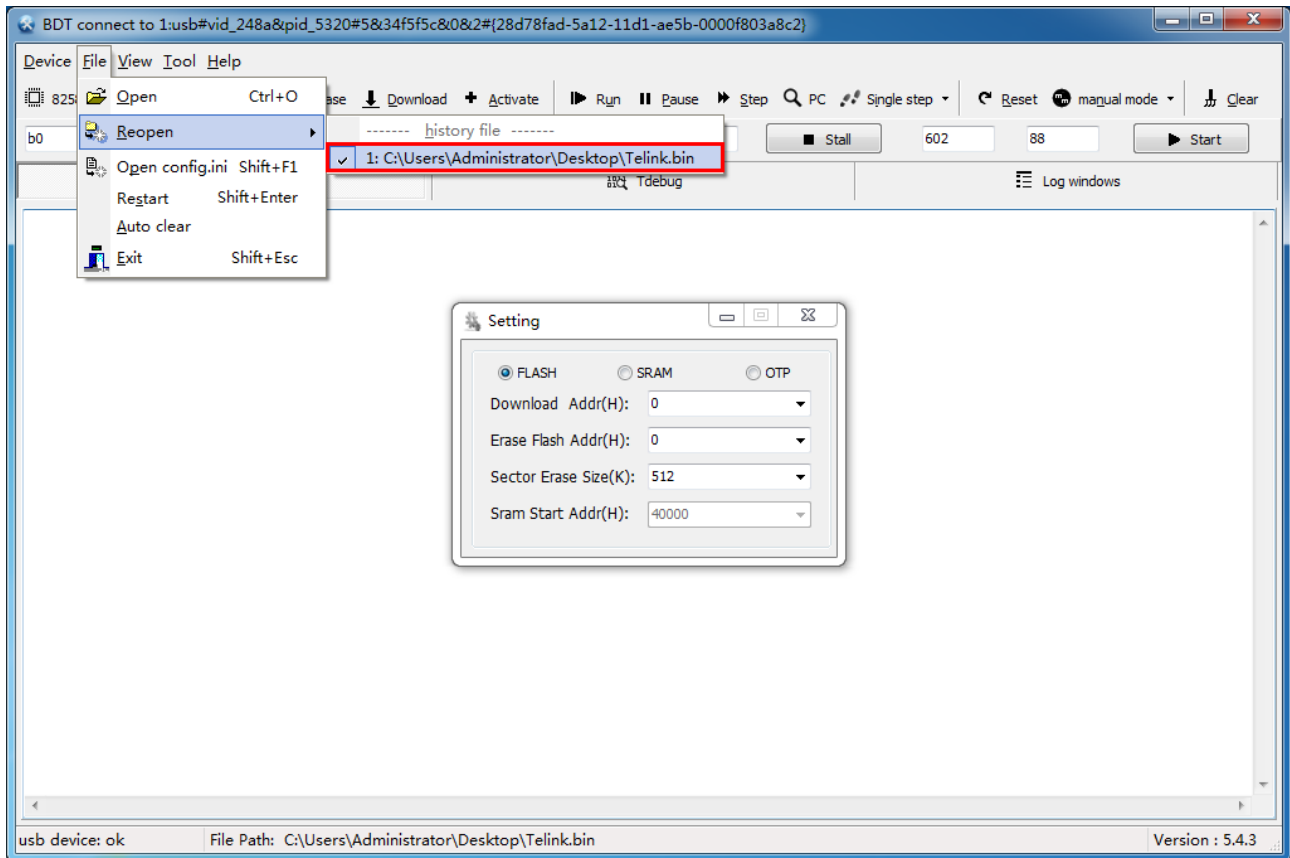


Figure 2.13: 在 “Reopen” 子菜单中直接选择目标文件路径

步骤 5: 检查目标板和 PC 之间的连接状态，确保其正常。

- 如果连接状态正常，主界面左下角将显示 “usb device: ok” 或 “evk device: ok”。
- 如果主界面左下角显示 “usb device: not found”，则表示目标板没有正确连接到电脑上。

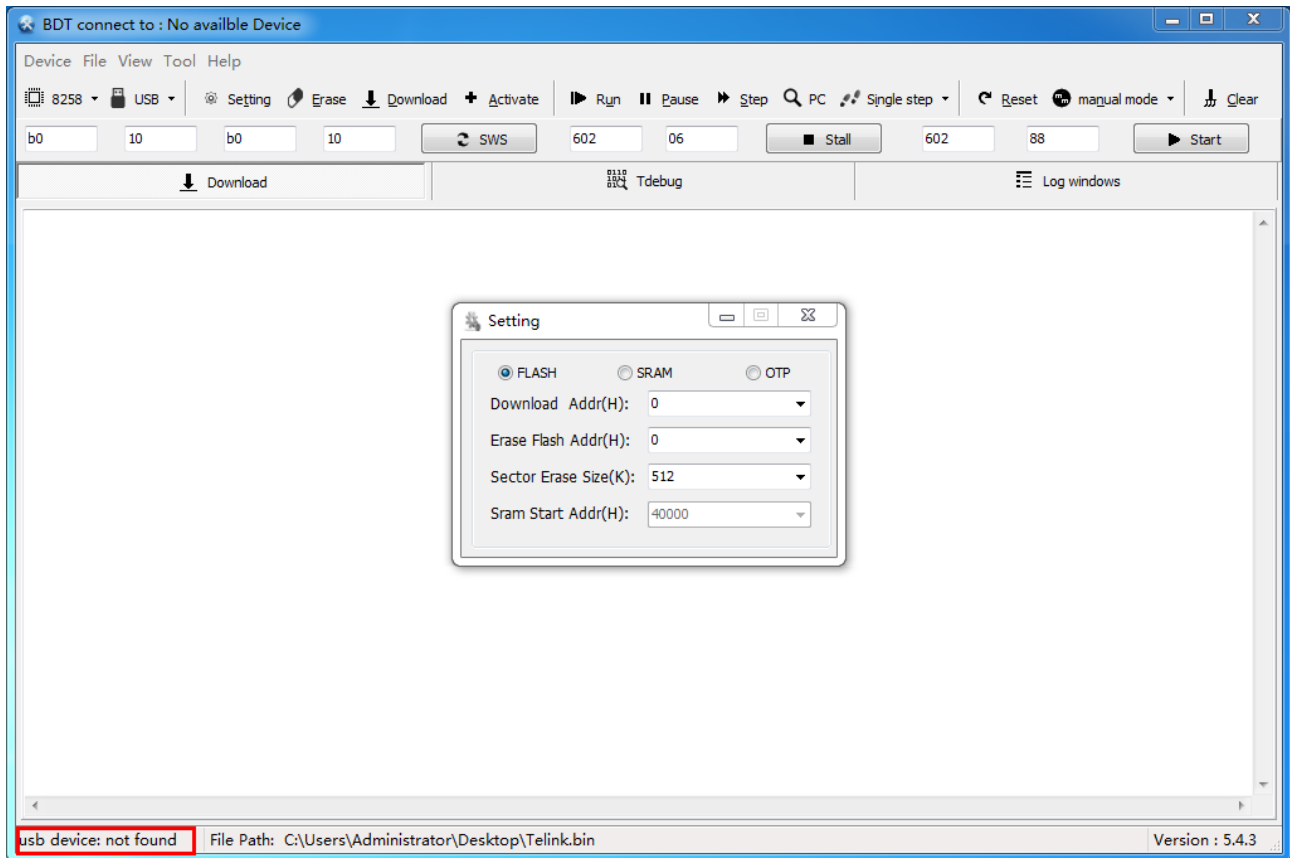


Figure 2.14: 连接状态指示

步骤 6: 将选定的文件下载到目标板。

点击“Download”按钮，所选固件文件将通过 USB 模式下载到目标板的指定 flash 空间。日志窗口将显示相应的日志信息，如下所示。

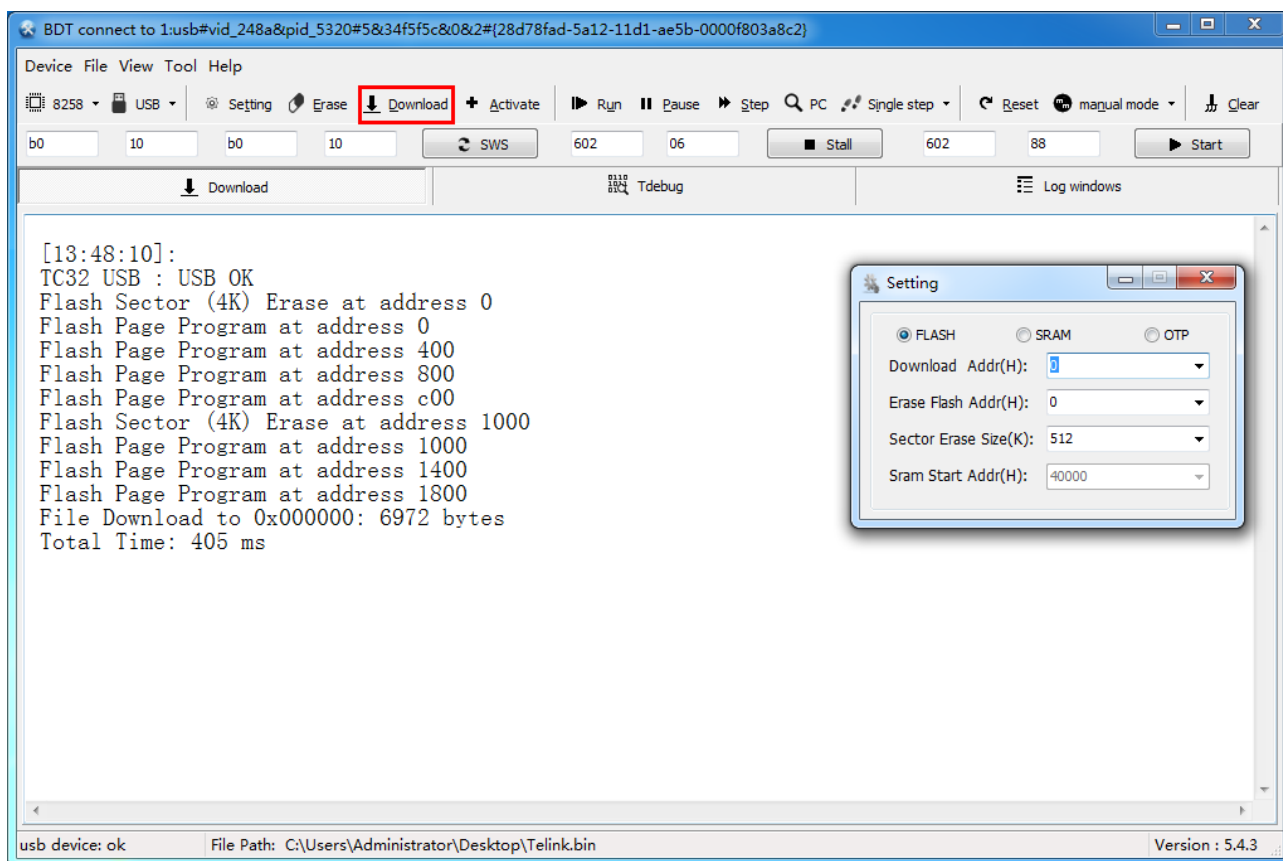


Figure 2.15: 通过 USB 模式将固件下载到 Flash 中

步骤 7: 复位 MCU，使新下载的程序运行，而无需重启 MCU（从 Burning EVK 上拔下/插入），用户可以按照以下任意一种方法启动并运行 MCU。

- 如果通过下拉菜单选择“manual mode”，固件烧录后，用户可以点击工具右上角的“Reset”按钮手动复位 MCU。

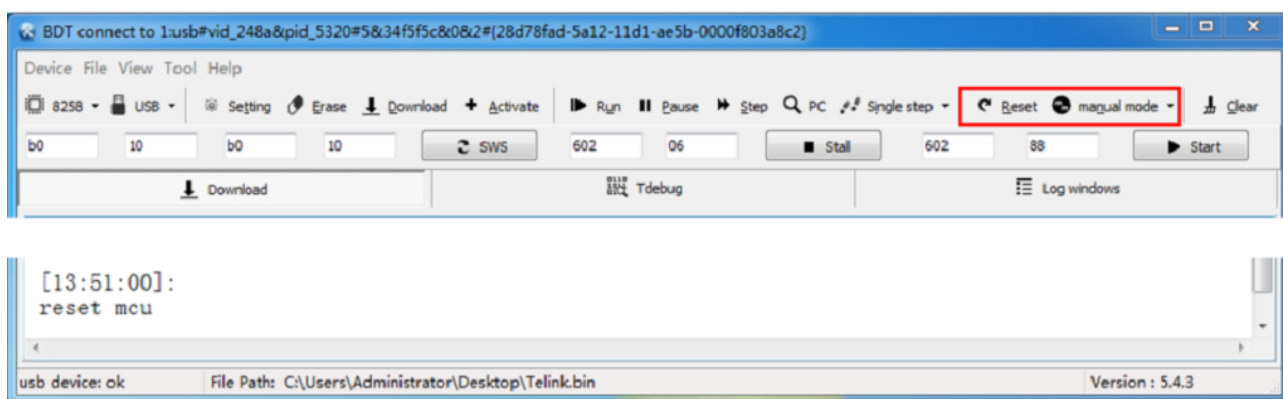


Figure 2.16: 手动复位 MCU

- 用户也可以在固件烧录前通过下拉菜单选择“auto mode”，以便在固件烧录后自动复位。

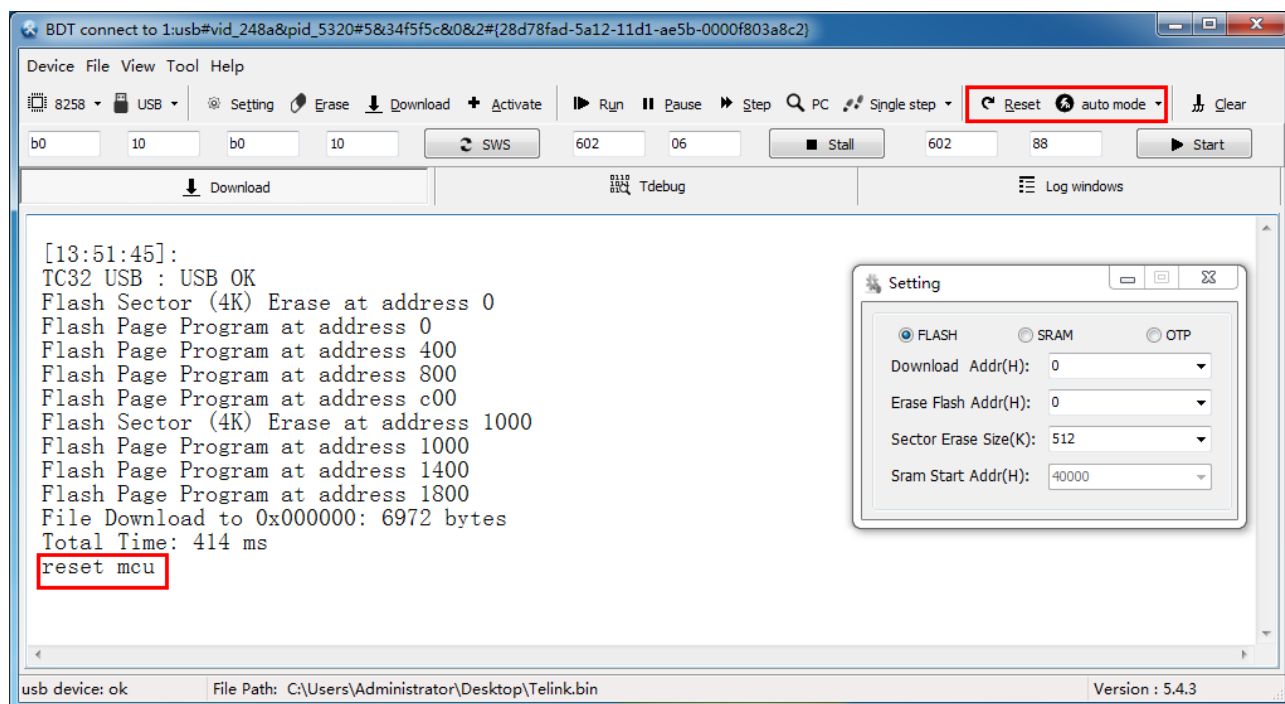


Figure 2.17: 自动复位 MCU

2.1.4 通过“EVK”模式将固件下载到“Flash”中

在使用“EVK”模式下载固件之前，请确保目标板通过“Burning EVK”方式连接到 PC，并且目标板的单线通信已建立。

用户可以按照本章指导，通过“EVK”模式将固件下载到目标板的特定 flash 空间。

步骤 1: 选择目标板的芯片类型，例如 8258（默认选项）。支持以下两种方法。

- 方法 1：使用下拉菜单选择芯片类型。
- 方法 2：按组合快捷键“Ctrl+Q”切换芯片类型，下拉菜单将自动显示每次切换后的当前类型。

注意：

所有芯片类型和默认选项都可以在“config.ini”文件中配置。

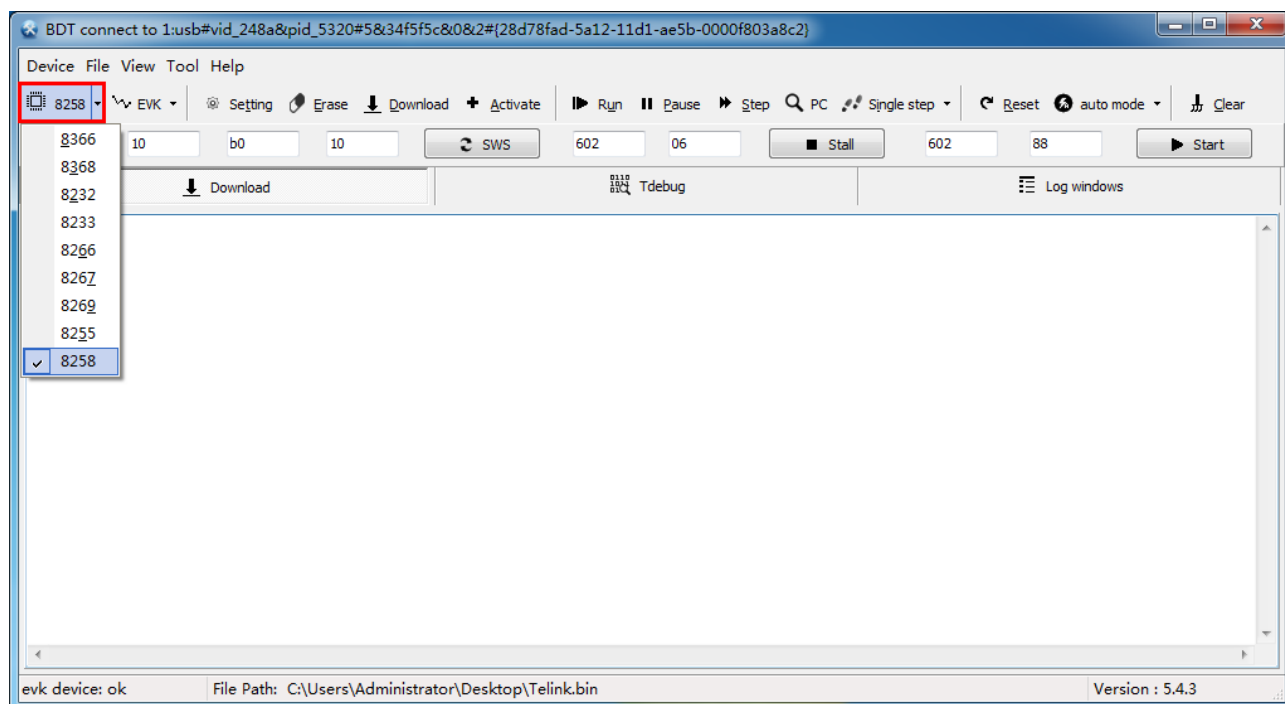


Figure 2.18: 选择芯片类型

步骤 2: 选择下载模式为“EVK”，支持以下两种方法。

- 方法 1: 使用下拉菜单选择 EVK 下载模式。
- 方法 2: 按组合快捷键“Ctrl+W”切换到 EVK 下载模式。每次切换后，下拉菜单将自动显示当前下载模式。

注意:

所有的下载模式和默认选项都可在“config.ini”文件中进行配置。

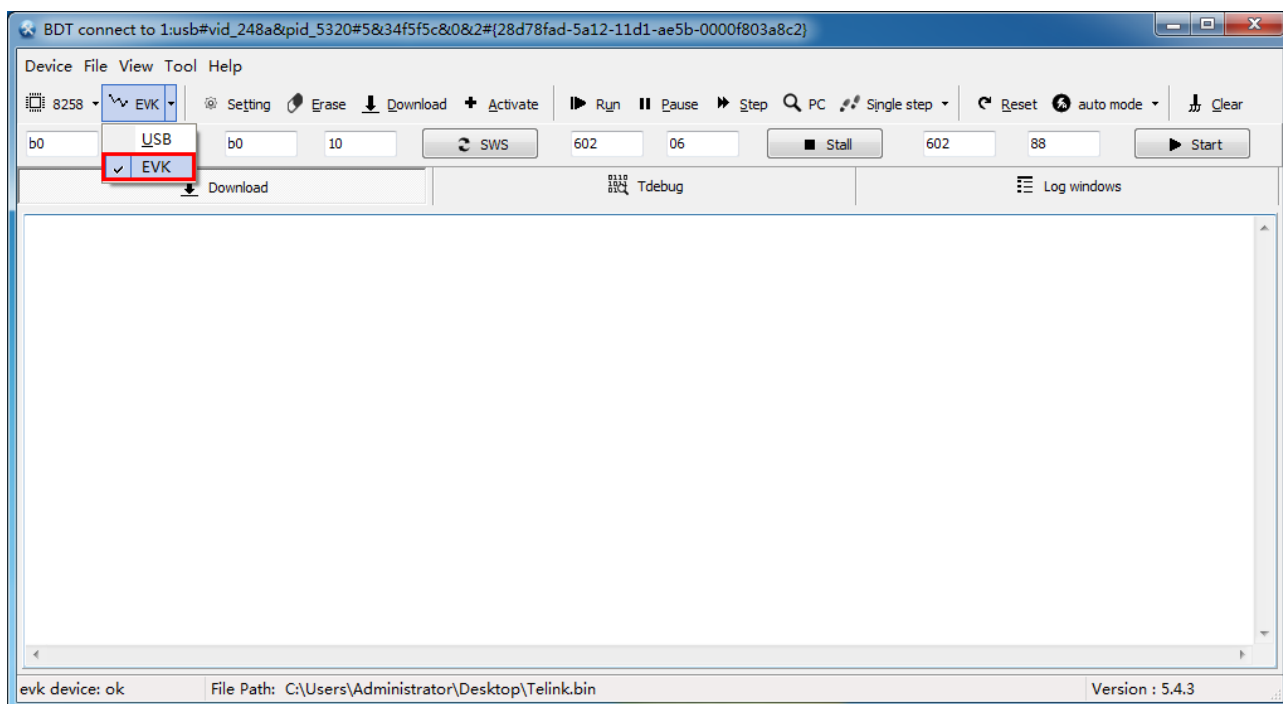


Figure 2.19: 选择下载模式为“EVK”

步骤 3: 点击“Setting”按钮，打开设置对话框。选择“Flash”选项。用户可以采用以下任意一种方法，设置目标固件存储的 flash 起始地址偏移量，例如 0x000000（默认选项）。

- 方法 1：使用“Download Addr”下拉菜单选择可用的偏移选项。
- 方法 2：在“Download Addr”的编辑框中输入偏移地址。

注意：

- 所有设置和默认选项都可以在“config.ini”文件中配置。
- 用户还可以配置“SRAM”/“OTP”设置选项，以便将目标固件下载到 SRAM 或 OTP 中的目标区域。

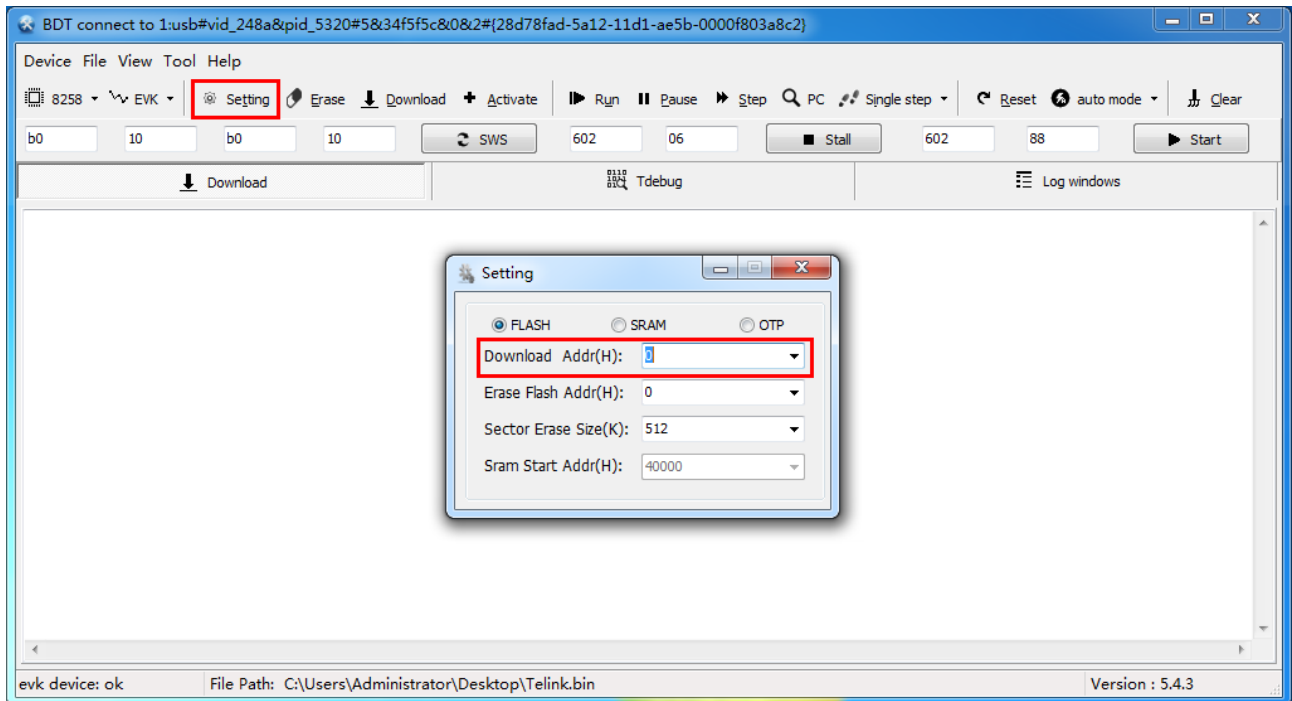


Figure 2.20: 打开“Setting”对话框

步骤 4: 选择要下载到目标板的目标固件文件。

用户可以通过“File -> Open”或“File -> Reopen”（如果在子菜单中可用）选择目标文件，文件路径将在主界面的底部可见。

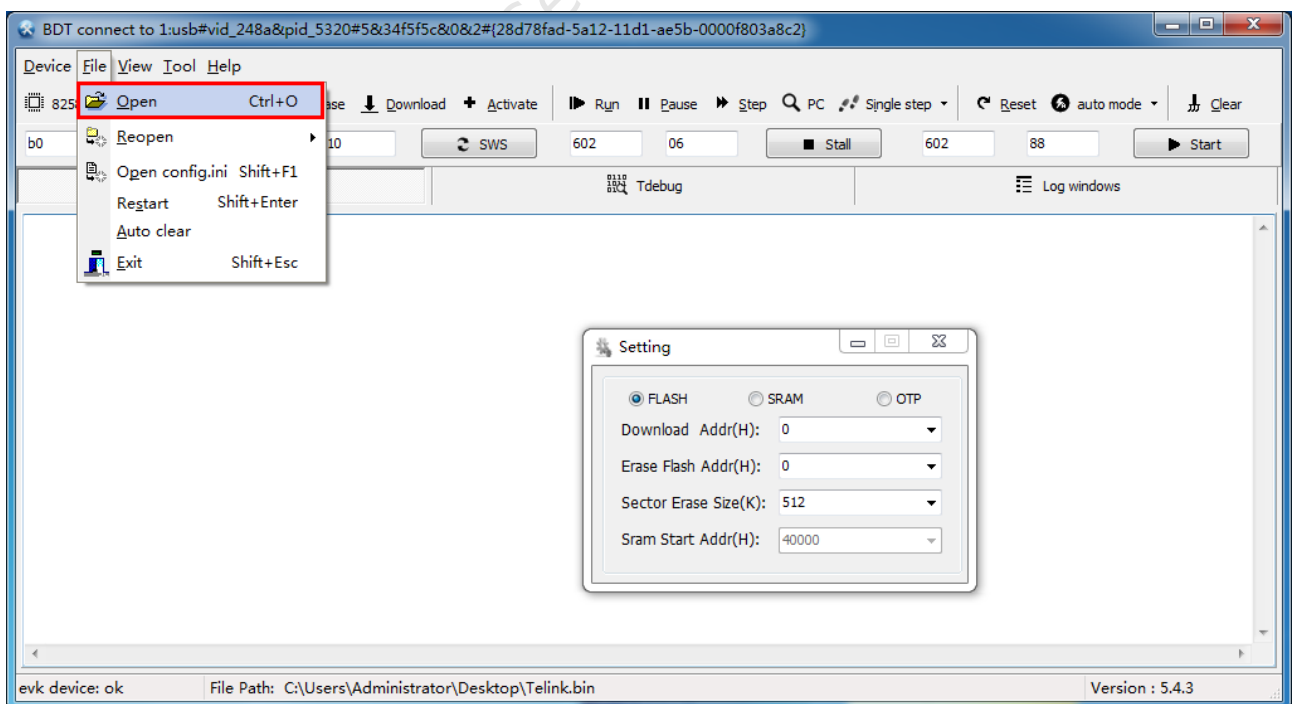


Figure 2.21: 选择目标文件

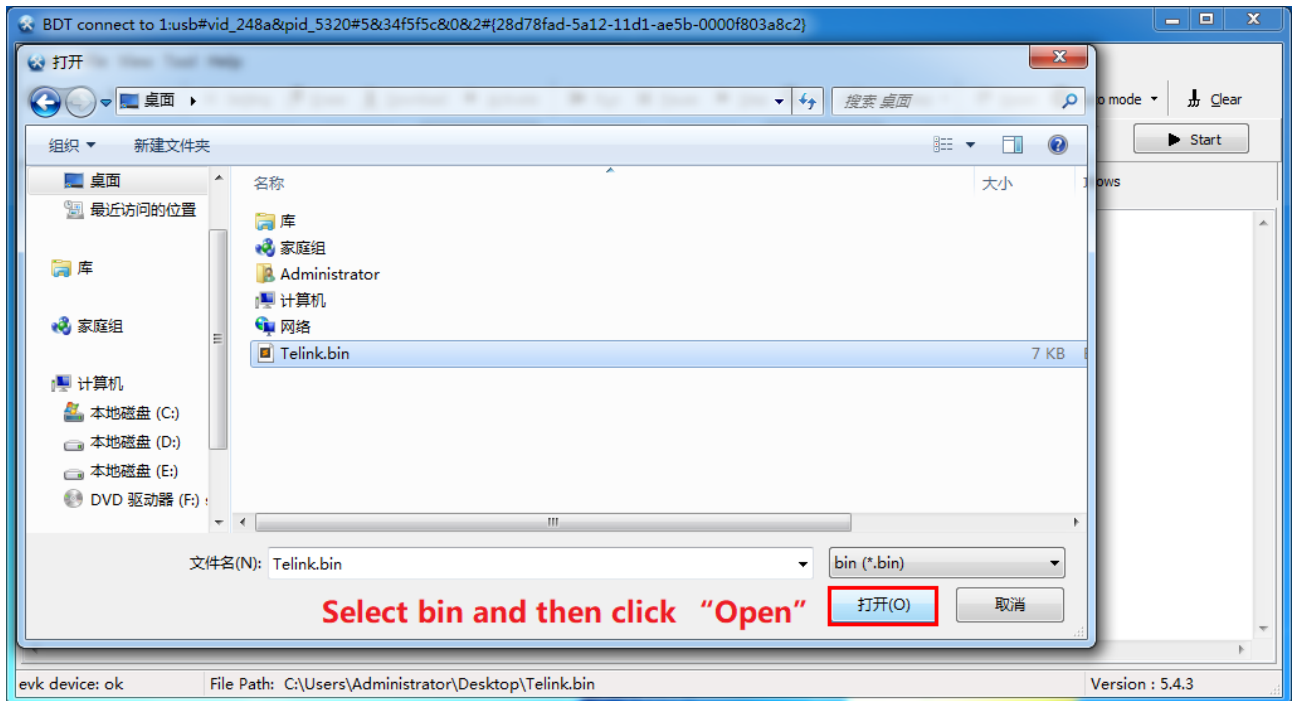


Figure 2.22: 选择 bin 文件

步骤 5: 检查目标板和 PC 之间的连接状态，确保它是正常的。

- 如果连接状态正常，主界面的左下方会显示“usb device: ok”或“evk device: ok”。
- 如果主界面的左下方显示“usb device: not found”，表示目标板没有正确连接到电脑上。

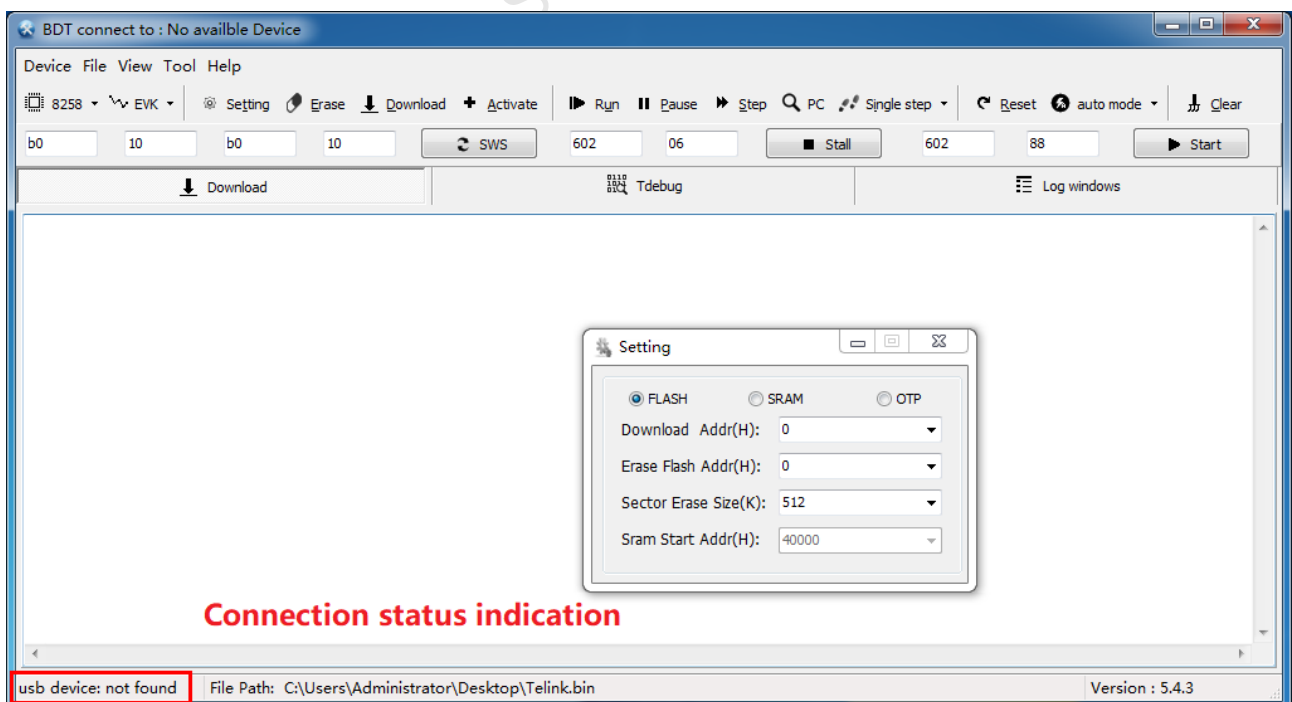


Figure 2.23: 检查连接状态

步骤 6：将选定的文件下载到目标板。

通过点击“Download”按钮，所选的固件文件将通过 EVK 模式下载到目标板的特定 flash 空间。日志窗口将显示相应的日志信息，如下图所示。

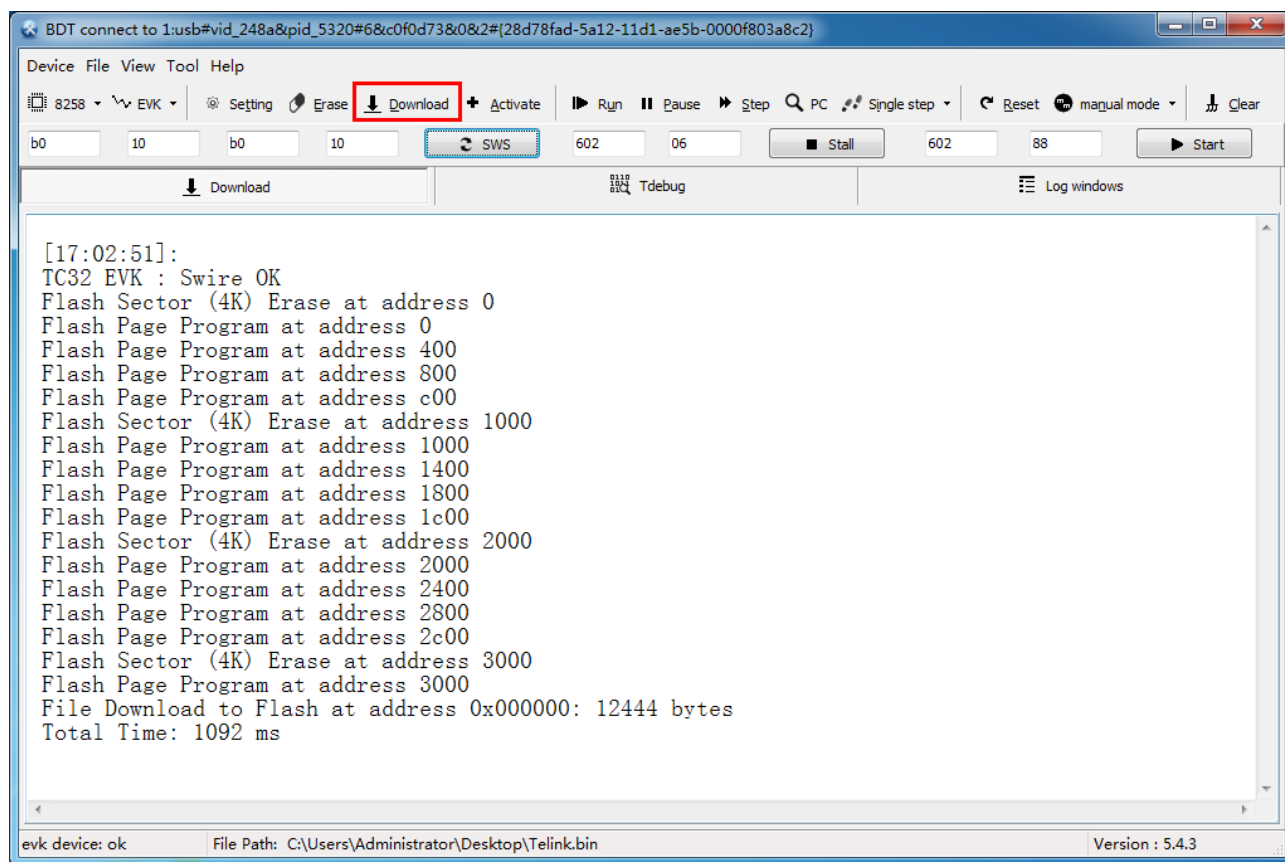


Figure 2.24: 通过“Burning EVK”模式将固件下载到 Flash 中

步骤 7：复位 MCU，使新下载的程序运行，而无需关闭 MCU 电源。用户可以按照以下两种方法之一启动并运行 MCU。

- 如果通过下拉菜单选择“manual mode”，固件烧录后，用户可以点击工具右上角的“Reset”按钮手动复位 MCU。

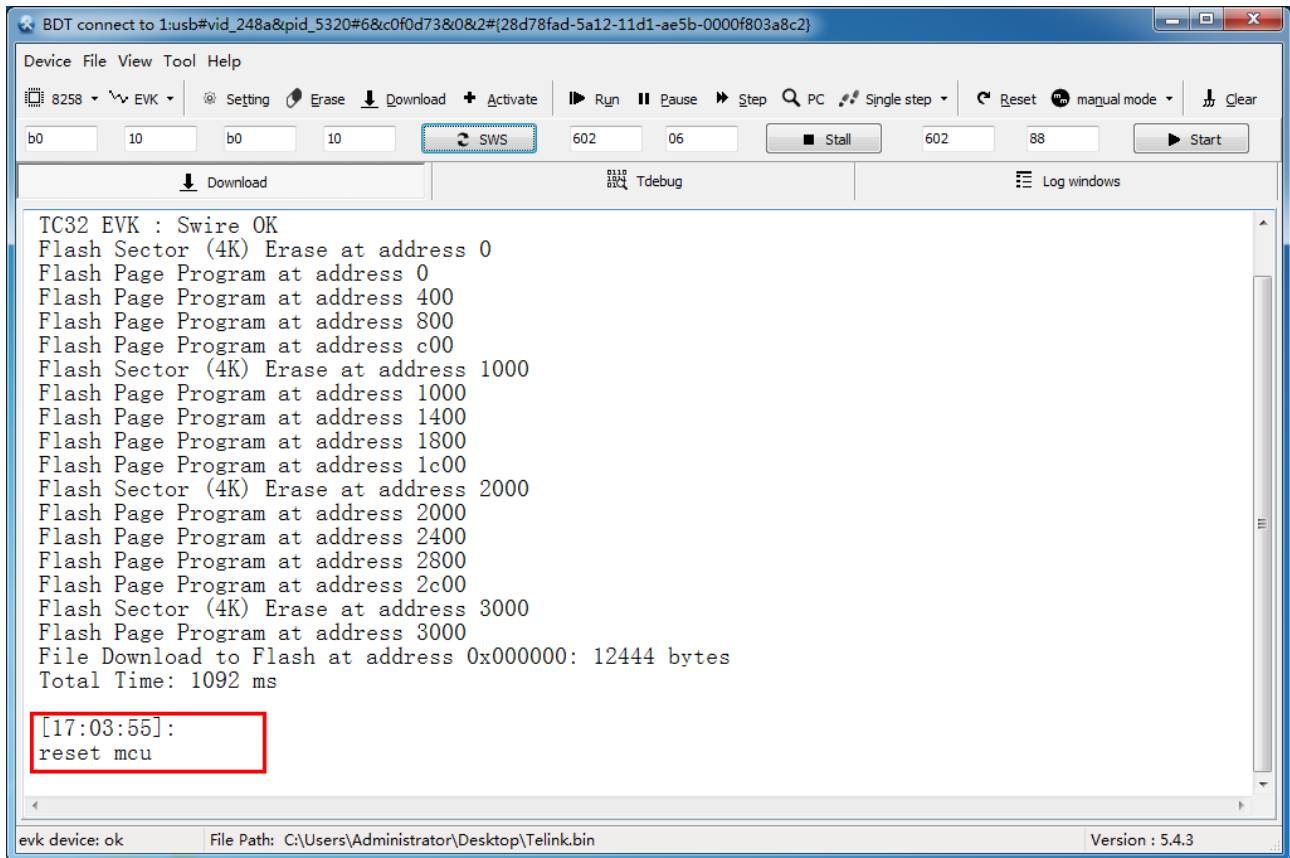


Figure 2.25: 手动复位 MCU

- 用户也可以在固件烧录前，通过下拉菜单选择“auto mode”，以便在固件烧录后自动复位。

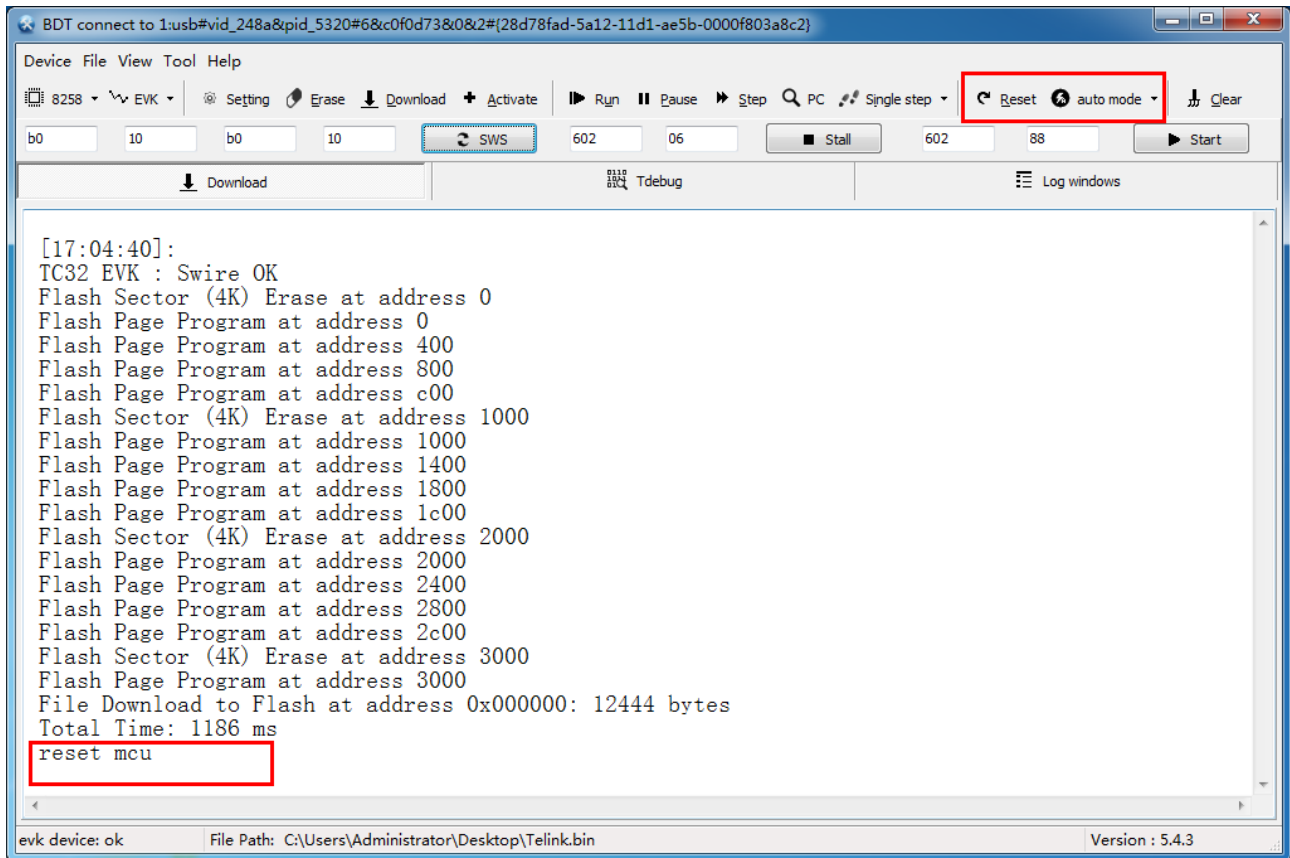


Figure 2.26: 自动复位 MCU

2.1.5 将固件下载到“SRAM”或“OTP”中

默认情况下，“Setting”选项设置为“Flash”，即通过单击“Download”按钮，目标固件将下载到目标板的“Flash”空间。

用户还可以将“Setting”选项更改为“SRAM”或“OTP”。在这种情况下，通过点击“Download”按钮，目标固件将相应地下载到目标板的“SRAM”或“OTP”空间。

对于 TLSR8258，目标固件的 SRAM 起始地址为 0x040000。对于其他 Telink’s SoCs，目标固件的 SRAM 起始地址为 0x008000。当然，用户可以根据需要在编辑框中设置其他地址。

目标文件下载成功后，目标板不会在“manual mode”下自动运行新的固件，因此用户需要点击“Reset”按钮启动程序。用户还可以在下载前通过下拉菜单选择“auto mode”，以启用 MCU 自动复位。

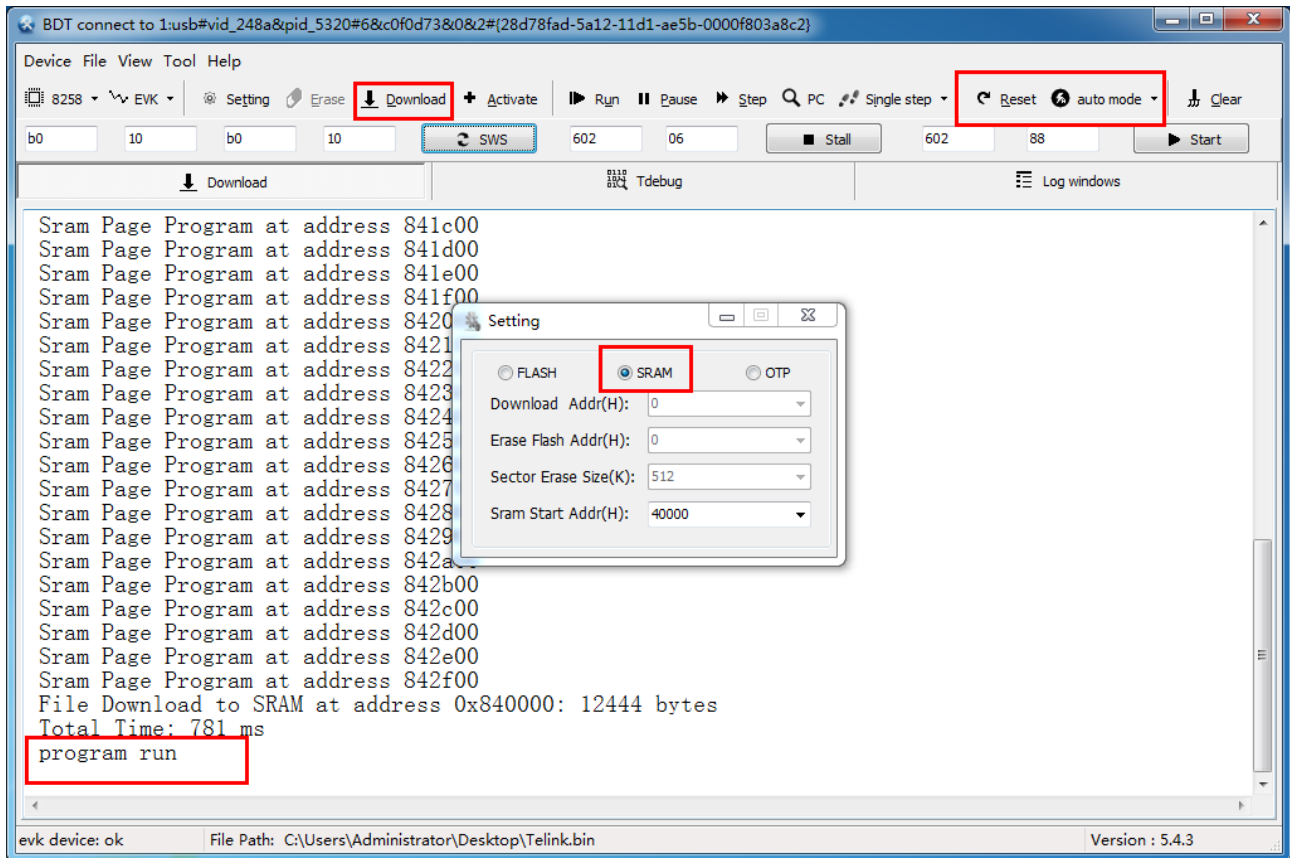


Figure 2.27: 通过“auto mode”将固件下载到“SRAM”

如果目标板支持 OTP 功能，用户可以将固件下载到“OTP”中。详细信息请参阅第 5 章功能支持列表。

2.1.6 多地址下载

有时候，需要将多个固件下载到 Flash、SRAM 或 OTP 的不同地址中。

例如，用户可以按照以下步骤将“Telink.bin”下载到 Flash 的地址“0x000000”和“0x005000”中。

步骤 1: 选择目标内存空间（如 Flash）来存储要下载的固件。

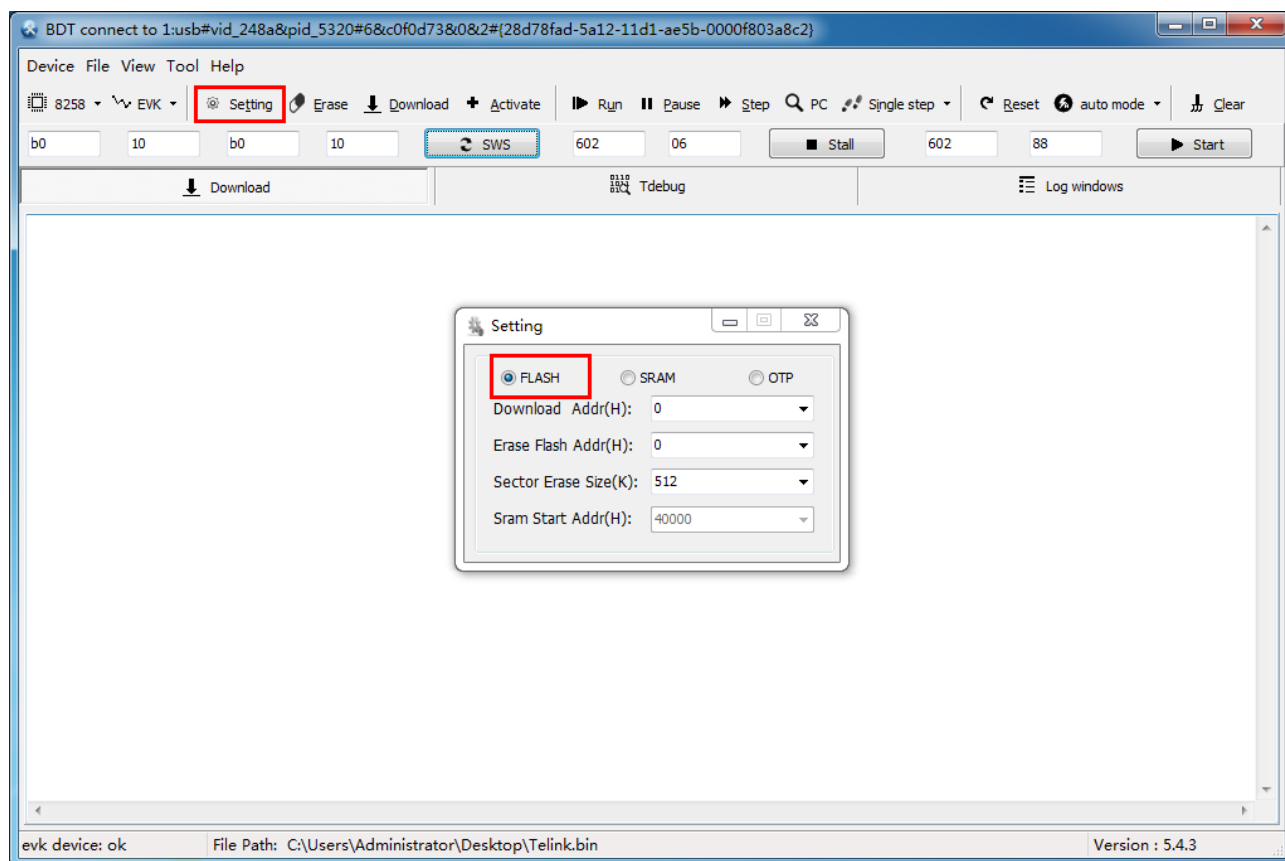


Figure 2.28: 选择目标空间

步骤 2: 点击 “Tool -> Multi-address download” 或按快捷键 “Ctrl+P” ，打开 “Multi-address Download” 界面。

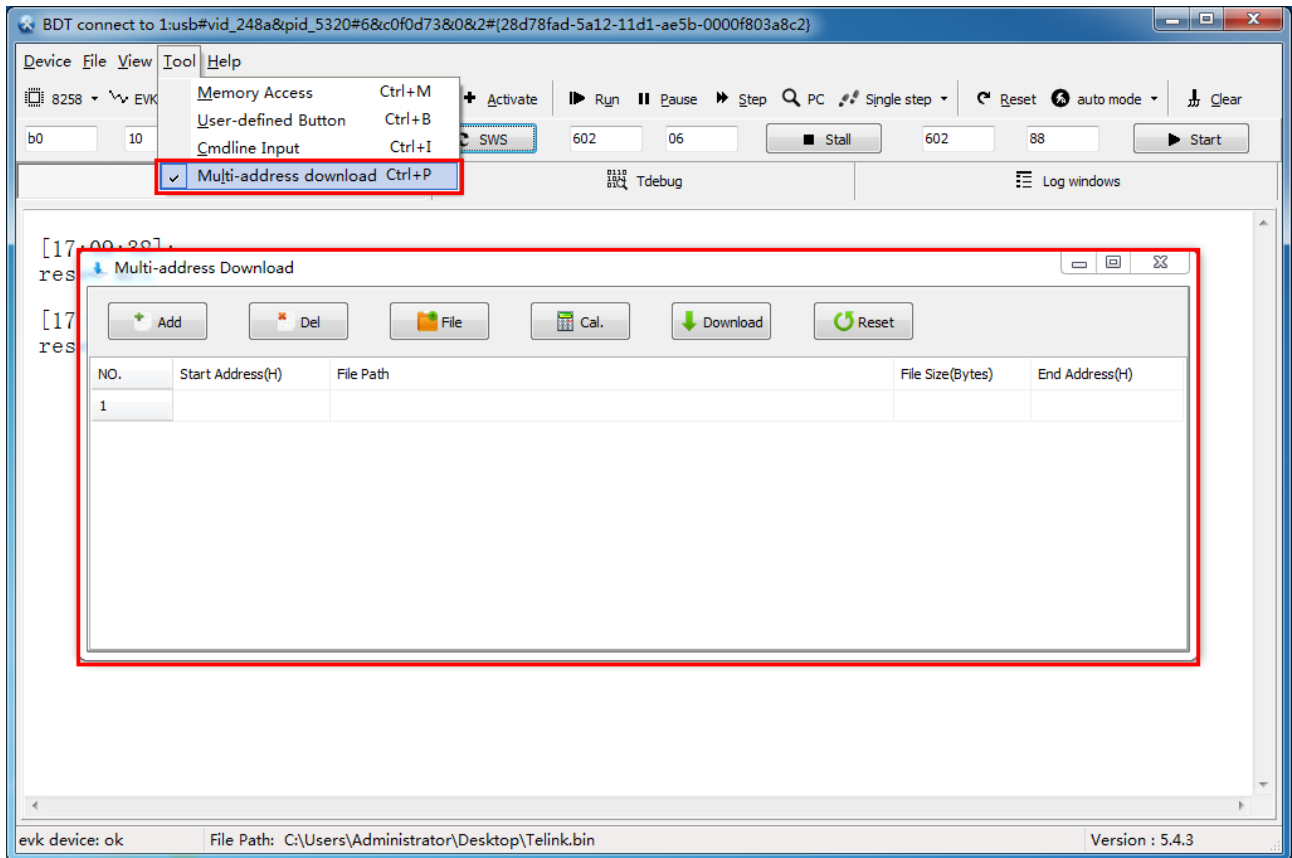


Figure 2.29: 打开 “Multi-address Download” 界面

步骤 3: 点击 “Add” 按钮，添加固件文件并设置目标地址。

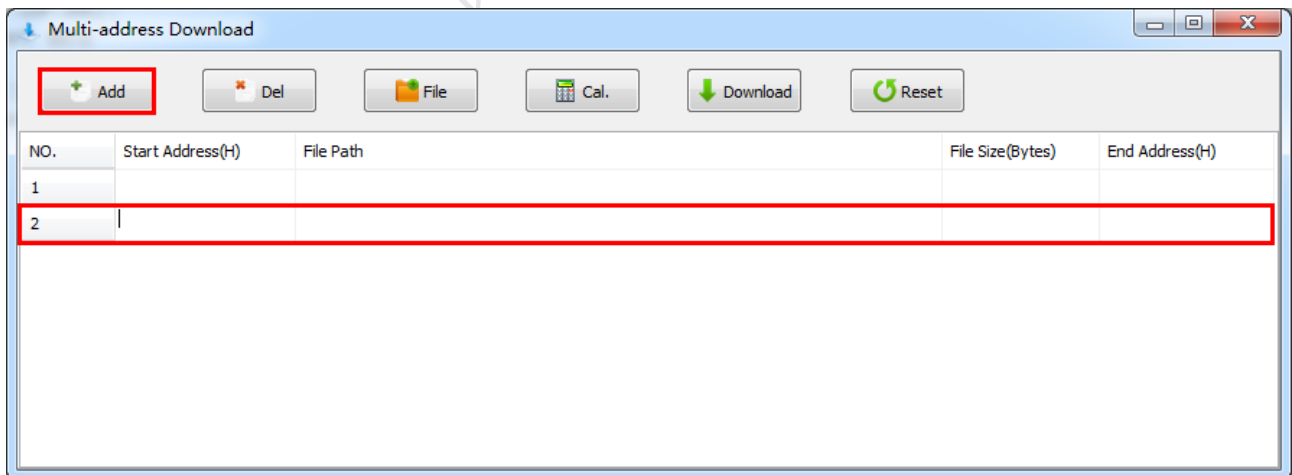


Figure 2.30: 添加固件文件并设置目标地址

步骤 4: 将固件 1 的路径添加到列表中，并设置起始地址的偏移量以存储此固件。

(1) 首先，选择第一行 (No.1)，然后点击 “File” 按钮将目标 bin 文件（例如 “Telink.bin”）添加到列表中，相应的目录将显示在 “File Path” 列中。

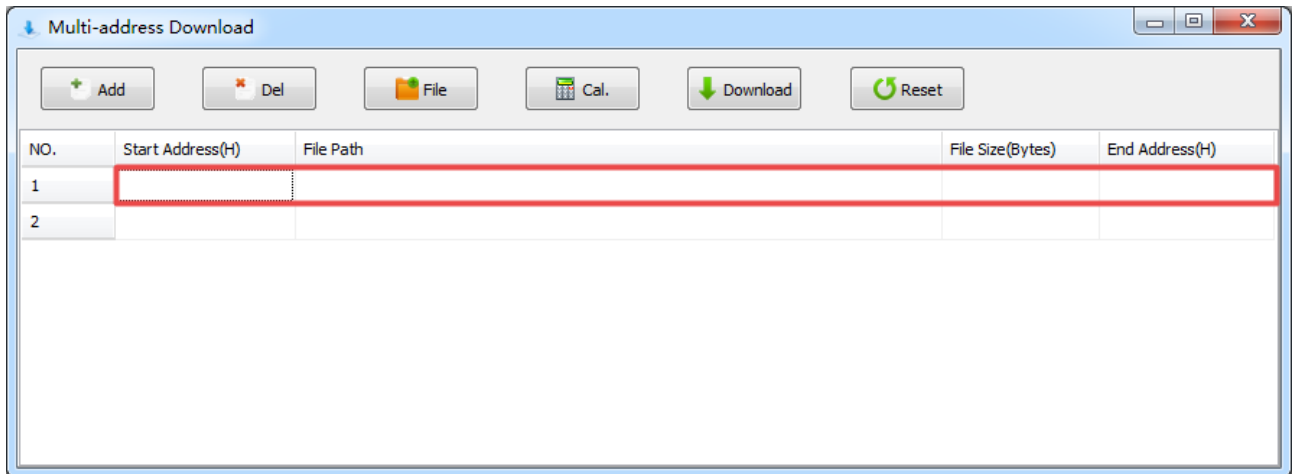


Figure 2.31: 添加固件

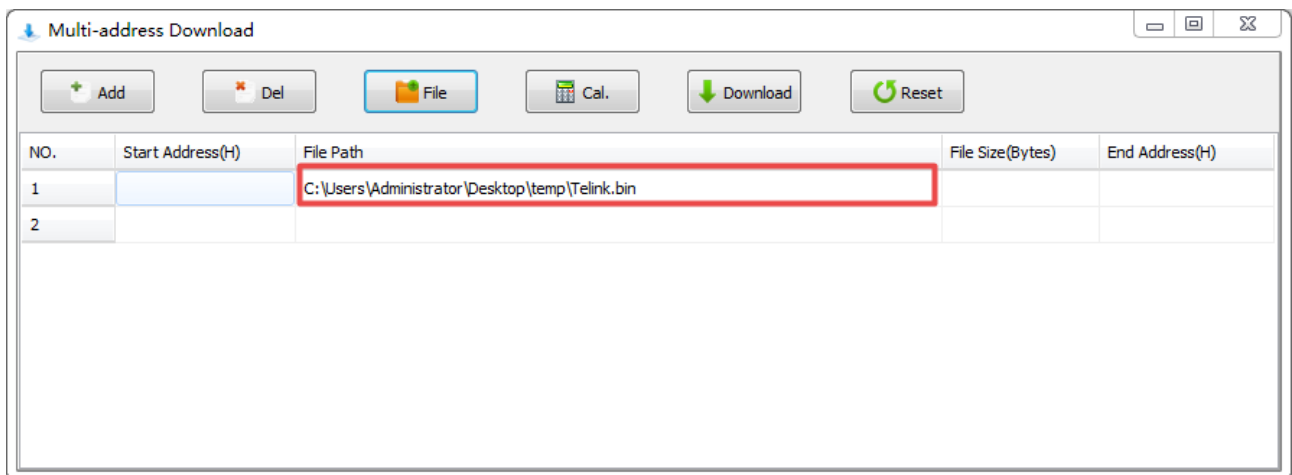


Figure 2.32: 显示 bin 文件路径

(2) 然后设置起始 Flash 地址的偏移量，将该固件存储在 “Start Address(H)” 列中。

为确保固件存储的目标区域有效，该工具将自动计算固件的大小和结束地址，然后在 “File Size(Bytes)” 和 “End Address(H)” 列中显示计算结果。

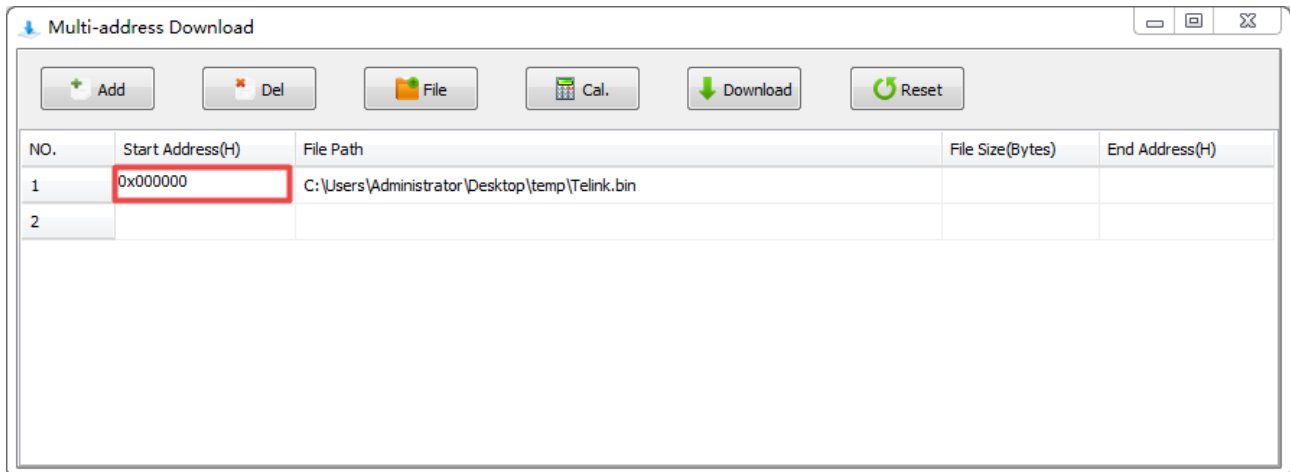


Figure 2.33: 设置偏移地址

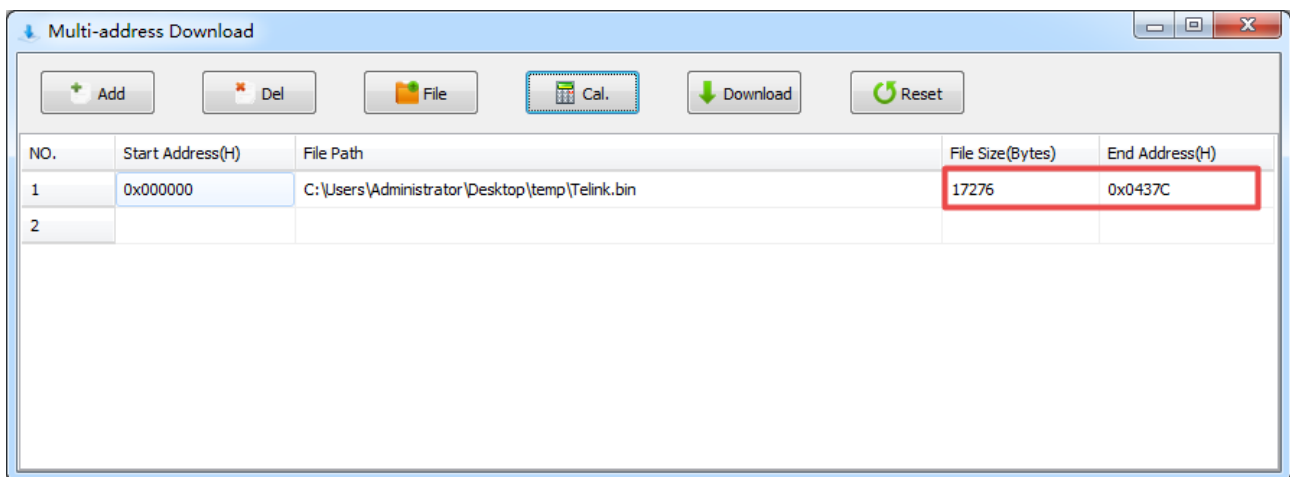


Figure 2.34: 显示计算结果

步骤 5: 请参阅步骤 4，将固件 2 的路径添加到列表中，并设置起始地址的偏移量以存储此固件。

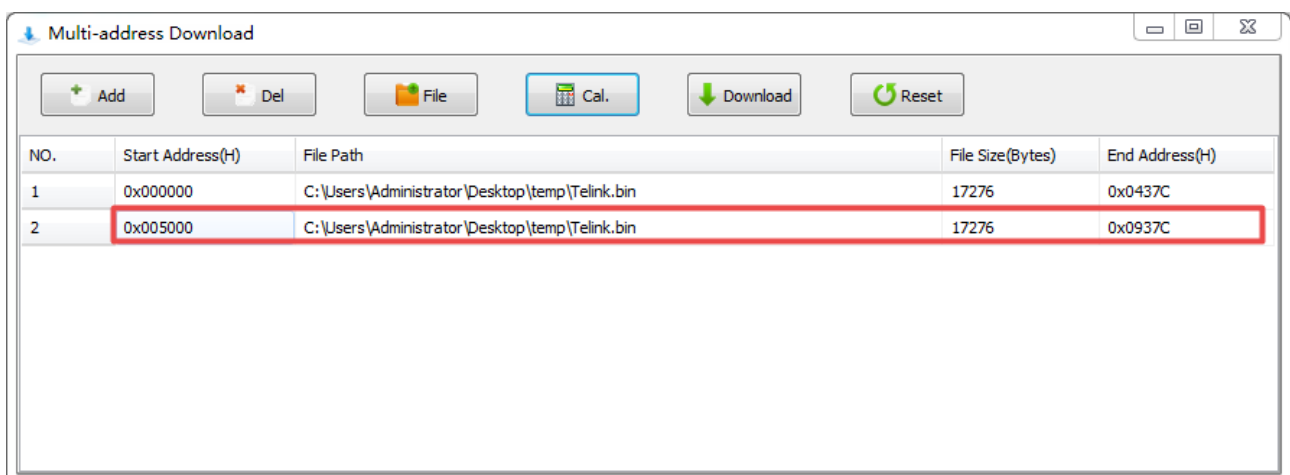


Figure 2.35: 添加固件 2 的路径

步骤 6: 单击 “Download” 按钮，将添加的固件 1 和固件 2 下载到目标板的指定内存位置。

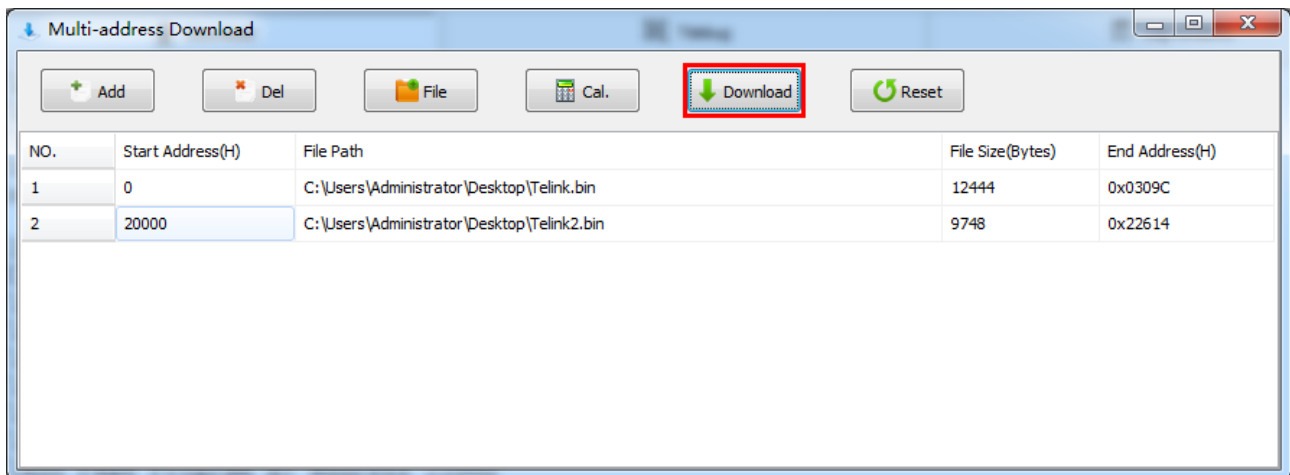


Figure 2.36: “Download” 按钮

```
Flash Page Program at address 1800
Flash Page Program at address 1c00
Flash Sector (4K) Erase at address 2000
Flash Page Program at address 2000
Flash Page Program at address 2400
Flash Page Program at address 2800
Flash Page Program at address 2c00
Flash Sector (4K) Erase at address 3000
Flash Page Program at address 3000
File Download to Flash at address 0x000000: 12444 bytes
Total Time: 1186 ms
```

Figure 2.37: 开始下载固件 1

```
[17:14:39]:
TC32 EVK : Swire OK
Flash Sector (4K) Erase at address 20000
Flash Page Program at address 20000
Flash Page Program at address 20400
Flash Page Program at address 20800
Flash Page Program at address 20c00
Flash Sector (4K) Erase at address 21000
Flash Page Program at address 21000
Flash Page Program at address 21400
Flash Page Program at address 21800
Flash Page Program at address 21c00
Flash Sector (4K) Erase at address 22000
Flash Page Program at address 22000
Flash Page Program at address 22400
File Download to Flash at address 0x020000: 9748 bytes
Total Time: 873 ms
```

Figure 2.38: 开始下载固件 2

步骤 7: 将所有固件下载到目标板后，点击 “Reset” 按钮使程序运行，而无需重启目标板。

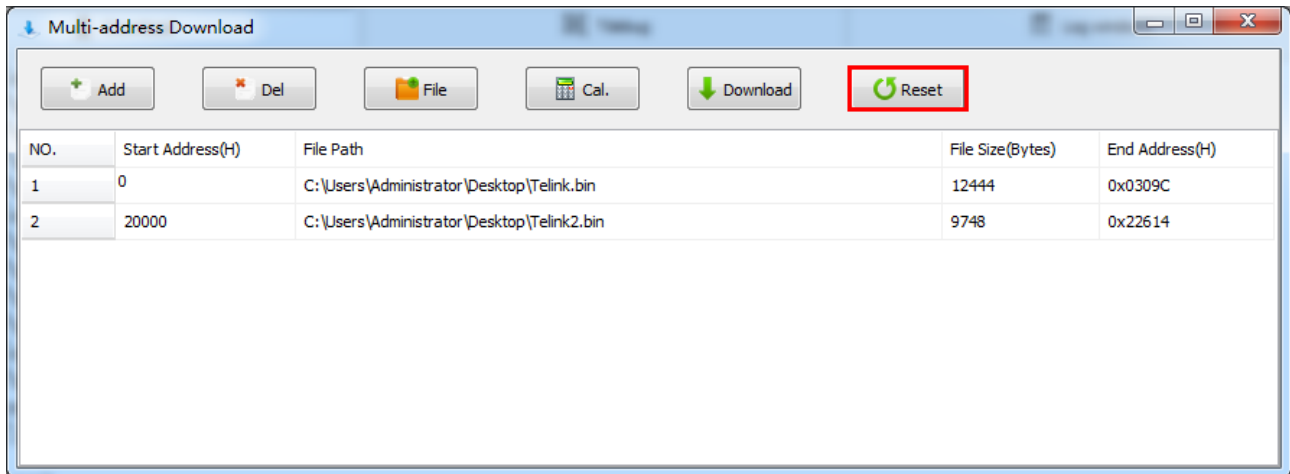


Figure 2.39: 复位

2.1.7 OTP 启动信息下载

- (1) 配置 OTP 时钟，以 8208 芯片为例，otp 地址 0x3ff8 写入数据 0x060301bf。
 - 功能：将 MCU 数字寄存器 0x603 写入 0x01，用于配置 OTP 时钟。
- (2) 启动代码，以 8208 芯片为例，otp 地址 0x3ff0 写入数据 0x060298bf。
 - 功能：将 MCU 数字寄存器 0x602 写入 0x98，MCU 启动代码的执行，程序开始跑起来。
- (3) 搬代码到 RAM 中，以 8208 芯片为例，otp 地址 0x3ff4 写入数据 0x0001f03f。
 - 功能：MCU 将 OTP 中从 0 地址开始的前 7936 个字节的内容搬到 RAM 0 地址处。
 - 数据长度 = $0x1f0 * 16 = 7936$ bytes

OTP 需要搬多少大小程序到 ram 中，可以参照如下计算方法。

OTP 上电第一次需要将 vector 和 ram code 段都搬到 ram 中，这两个段的大小计算公式为：

ram code 的 LMA 的基地址 + ram code size（之所以不用 vector size + ram_code size 来计算，是因为可能会有对齐因素导致大小计算的不对）

又因为启动信息中的长度信息是：实际长度/16。所以启动信息中的长度信息的通用的计算公式总结如下：

$((\text{ram code 的 LMA 的基地址} + \text{ram code size}) / 16 \text{ 字节对齐}) / 16$ 。

所以如下 lst 文件的最终计算公式为：0x220+0xAEC，16 字节对齐后的大小为 3344; $3344/16=209(0xd1)$

对应 otp 地址 0x3ff4 写入数据 0x0000d13f 即可。

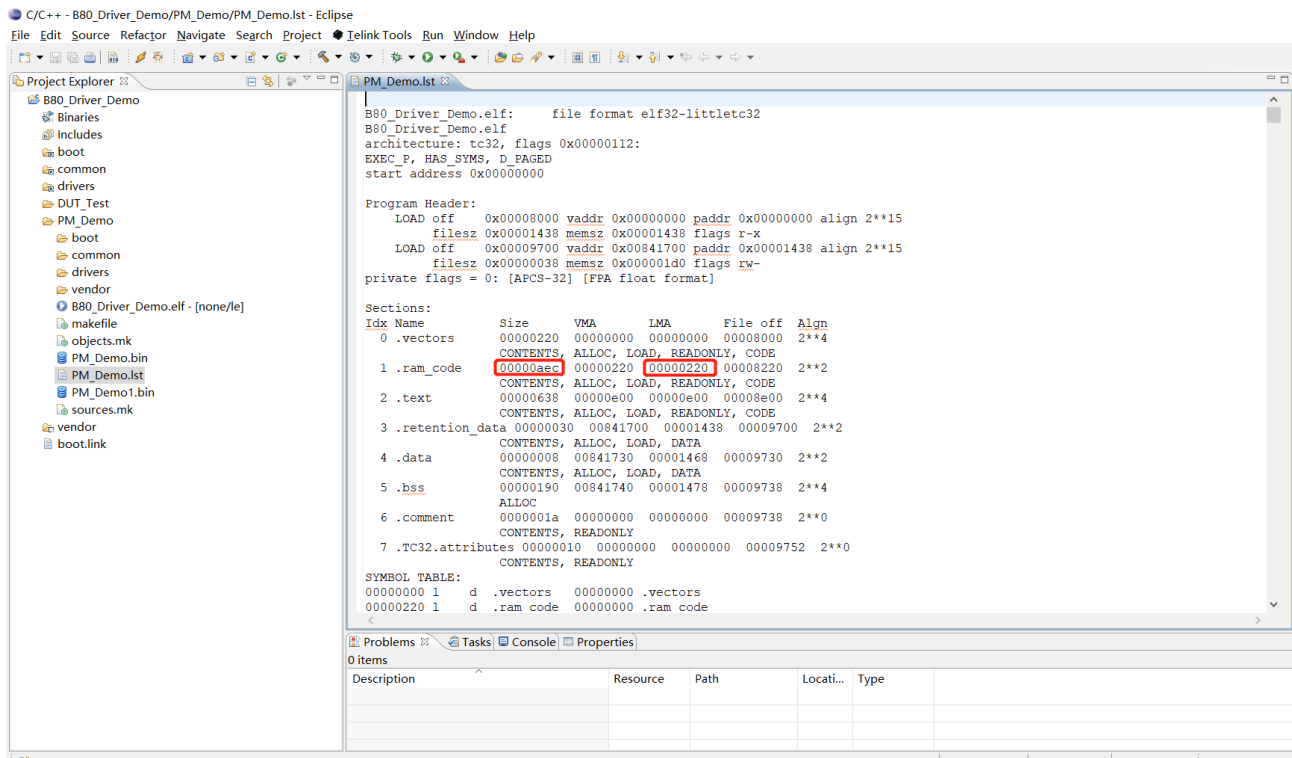


Figure 2.40: lst 文件

以 8208 芯片为例，用户可以按照以下步骤进行 OTP 启动信息的下载。

步骤 1: 打开 BDT 工具，选择 8208 芯片。

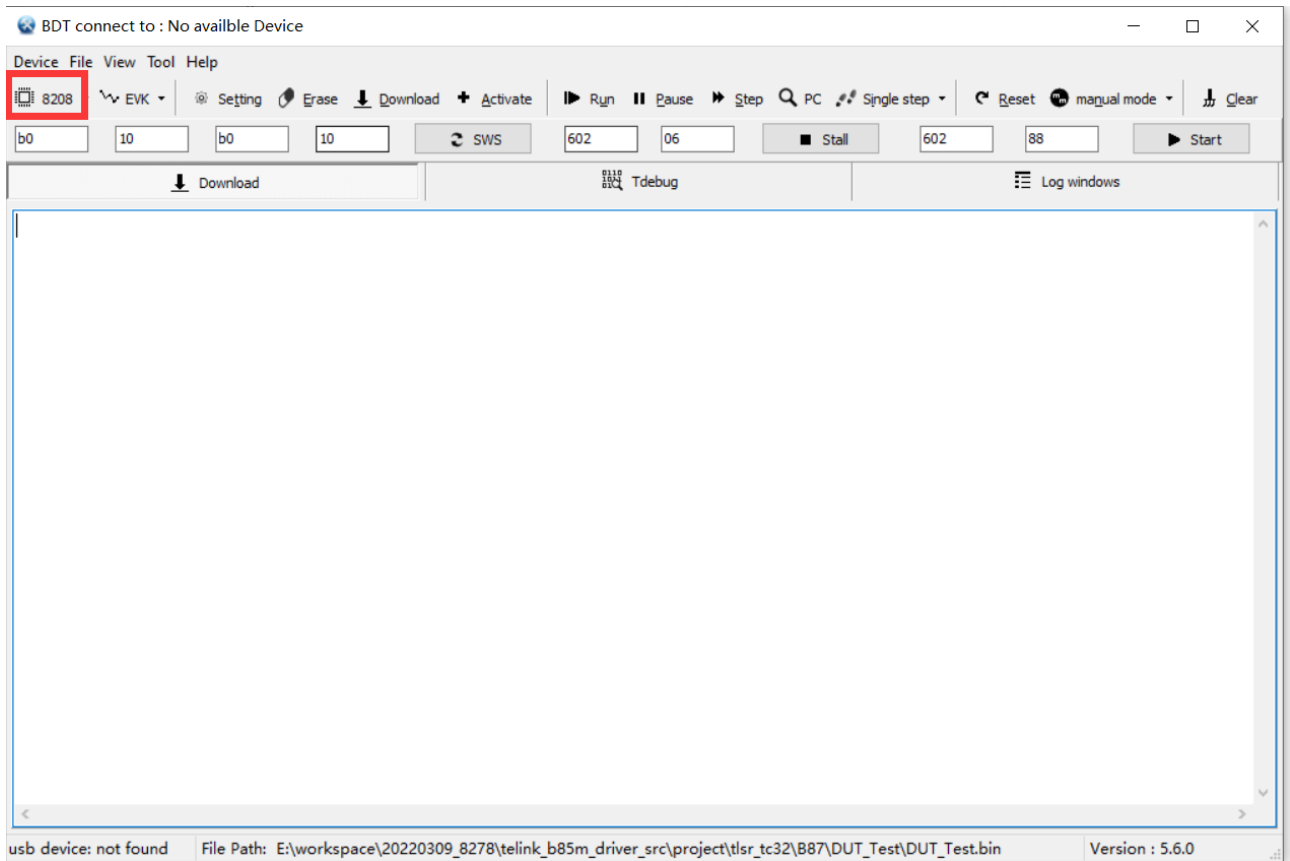


Figure 2.41: 芯片选择

步骤 2: 点击 Tool->Memory Access，输入 OTP 地址，写入启动信息及其长度，并在 data 编辑框按下回车键，对应信息会烧录到 OTP 地址中。

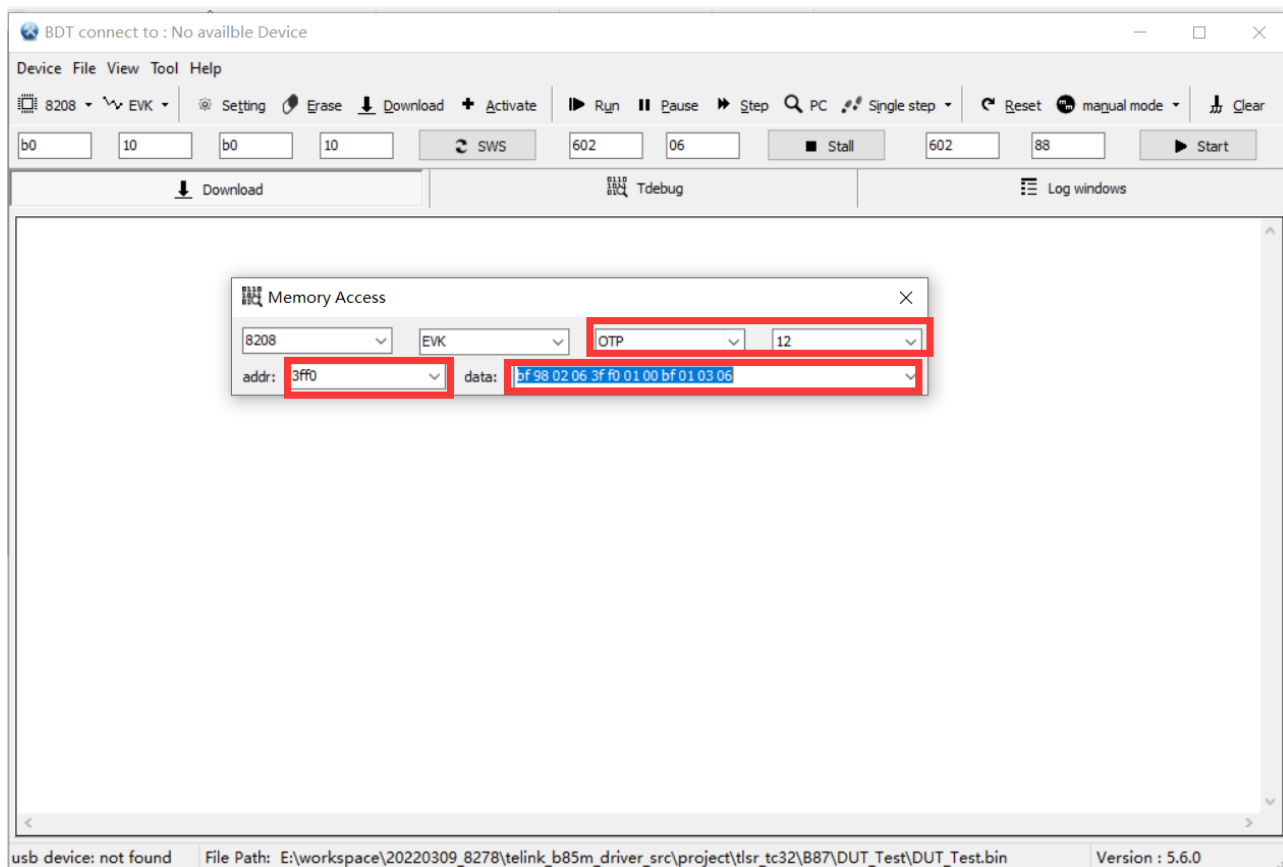


Figure 2.42: OTP 启动信息下载

2.2 擦除 Flash 扇区

“Erase Flash Sector” 功能用于以 4kB 为单位从特定地址开始擦除特定 flash 空间。

例如，要擦除从地址 0x040000 开始的 64kB 的 flash 空间，用户可以按照以下步骤进行。

步骤 1: 点击 “Setting” 按钮打开设置对话框。选择 “FLASH” 设置选项。

设置要擦除 flash 空间的起始地址，例如 0x040000（默认选项：0x000000），用户可以通过 “Erase Flash Addr” 下拉菜单选择偏移选项（如果可用），或直接在编辑框中输入偏移地址。

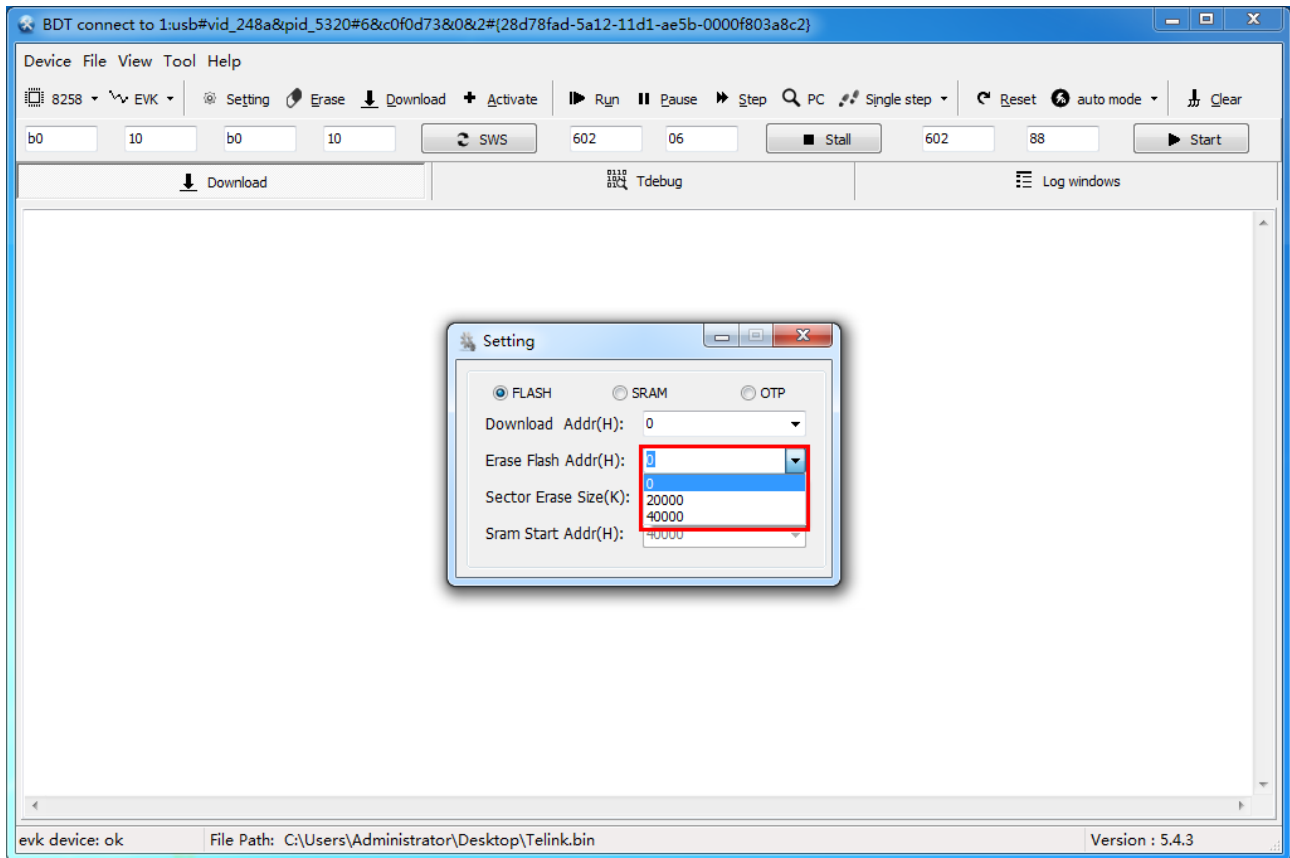


Figure 2.43: 选择起始地址

步骤 2: 以 4kB 为单位设置要擦除的 flash 空间大小，例如 64kB（默认选项：512kB），用户可以通过“Sector Erase Size”下拉菜单选择大小（如果可用），或直接在编辑框中输入大小。

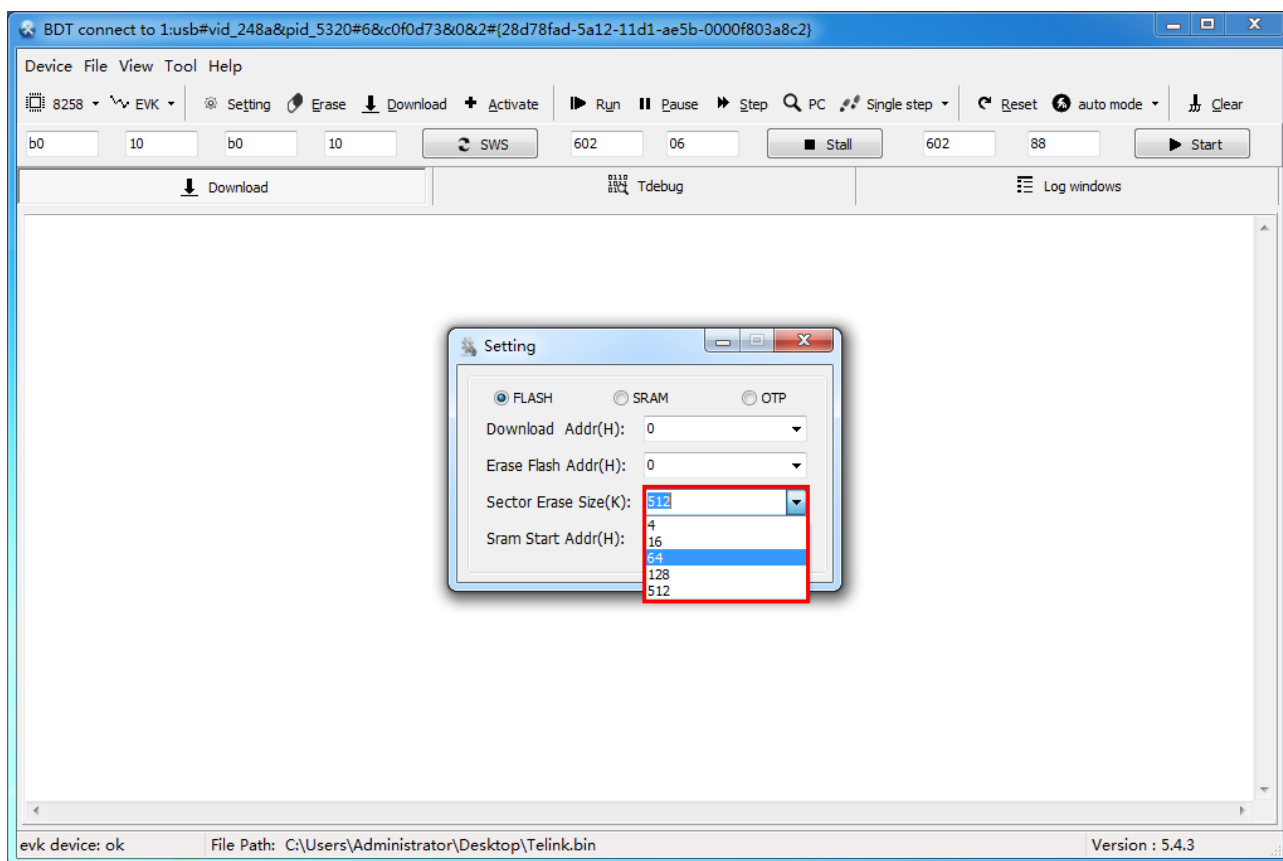


Figure 2.44: 设置擦除大小

步骤 3: 点击“Erase”按钮，开始擦除目标板的指定 flash 空间。

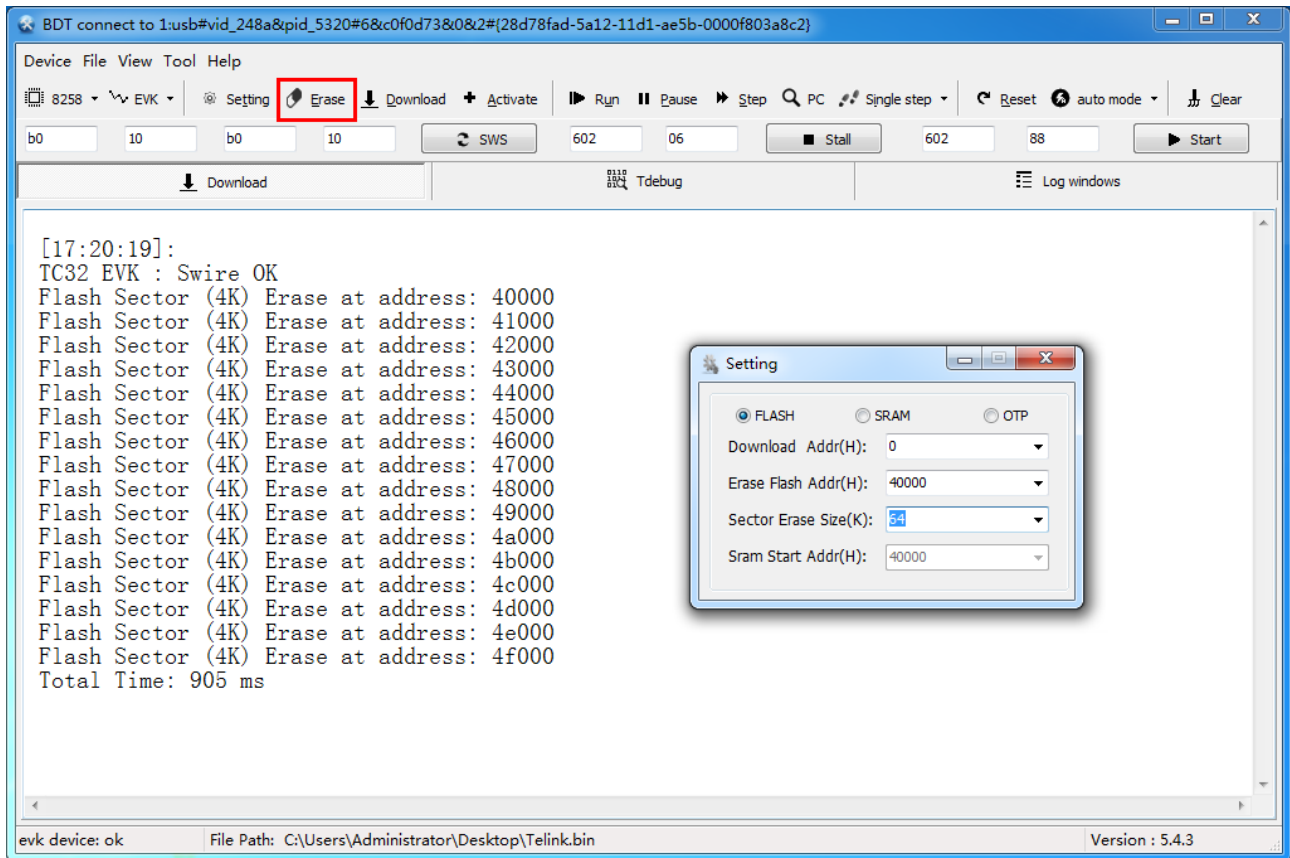


Figure 2.45: 从 0x000000 起始地址开始擦除 512kB 的 flash 空间

2.3 Flash 去除保护

Flash 被写保护之后会导致擦写以及烧录不成功，需要去除保护之后才能对 Flash 进行相应操作。Flash 去除保护操作步骤如下。（该功能从 v5.7.0 版本之后开始支持）

步骤 1：打开 BDT 工具，选择芯片类型。

步骤 2：点击“Unlock”按钮，log 中出现“Flash remove protection successfully!”代表去除保护成功。

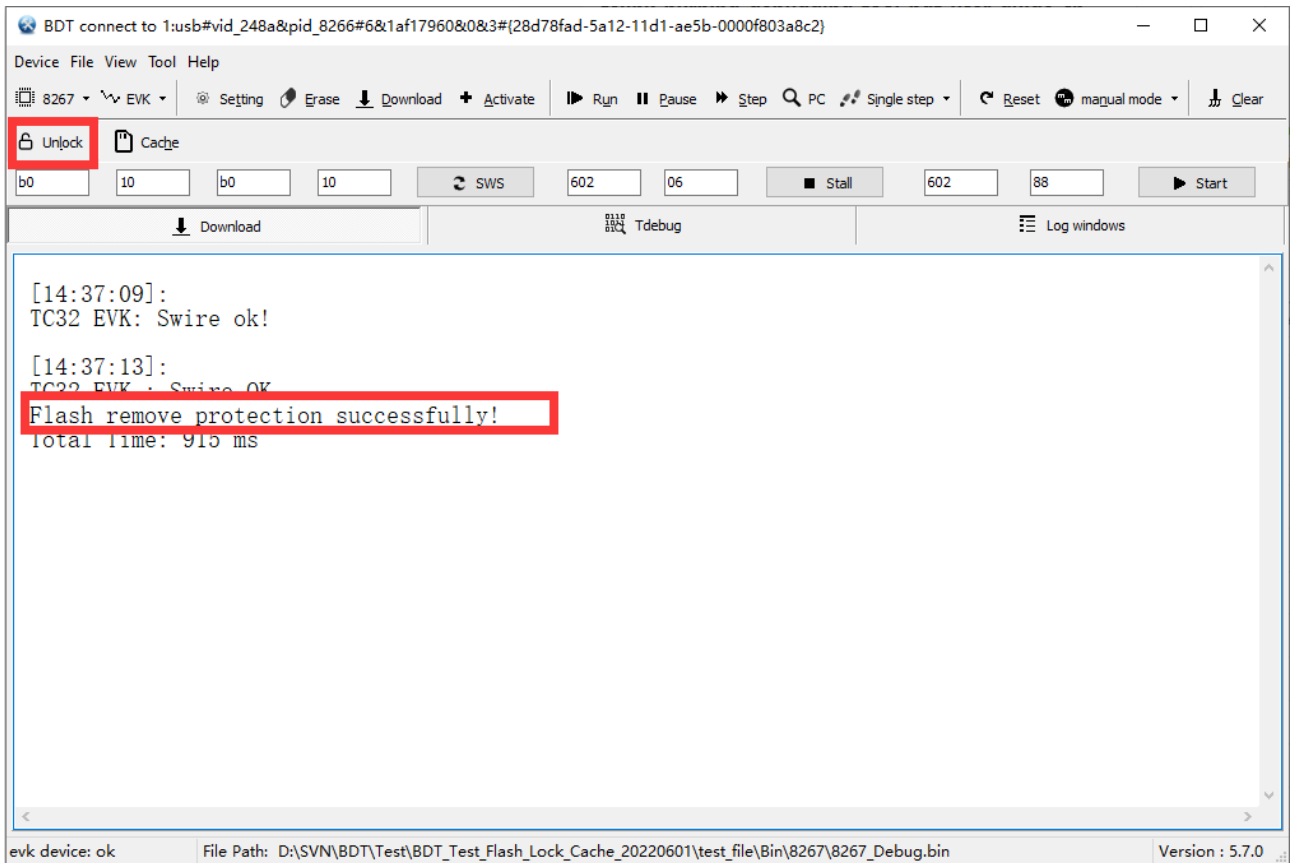


Figure 2.46: Flash 去除保护

2.4 Cache 解析

芯片上电搬运代码，如果即将运行的代码不在 cache 中，则会自动从 Flash 或 OTP 中读取函数到 cache 中。该功能主要是解析 cache 搬运代码是从 flash 哪个地址取数据，以及数据内容是什么。并将解析的数据与原始数据进行比较，来确定 cache 搬运代码是否正确。该功能前提是芯片已下载程序并运行，然后通过如下步骤进行解析。（该功能从 v5.7.0 版本之后开始支持）

步骤 1: 打开 BDT 工具，选择芯片类型，选择芯片需要下载的固件。

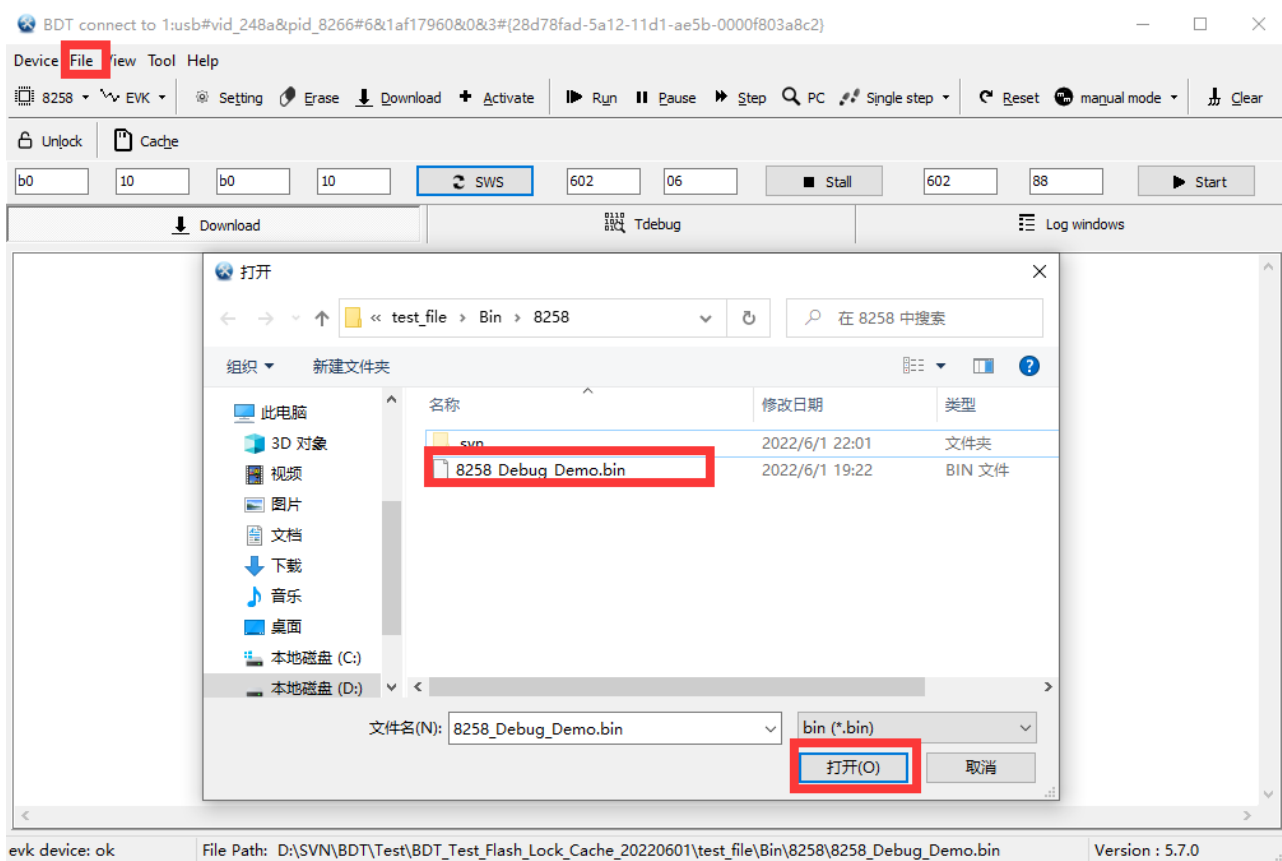


Figure 2.47: Bin 文件选择

步骤 2: 点击“Cache”按钮。下图标记“1”为搬运代码所在的 flash 地址。标记“2”为对应 flash 地址存储的数据。标记“3”代表测试结果。“OK”代表搬运代码数据正确。“FAIL”代表搬运代码数据出错。

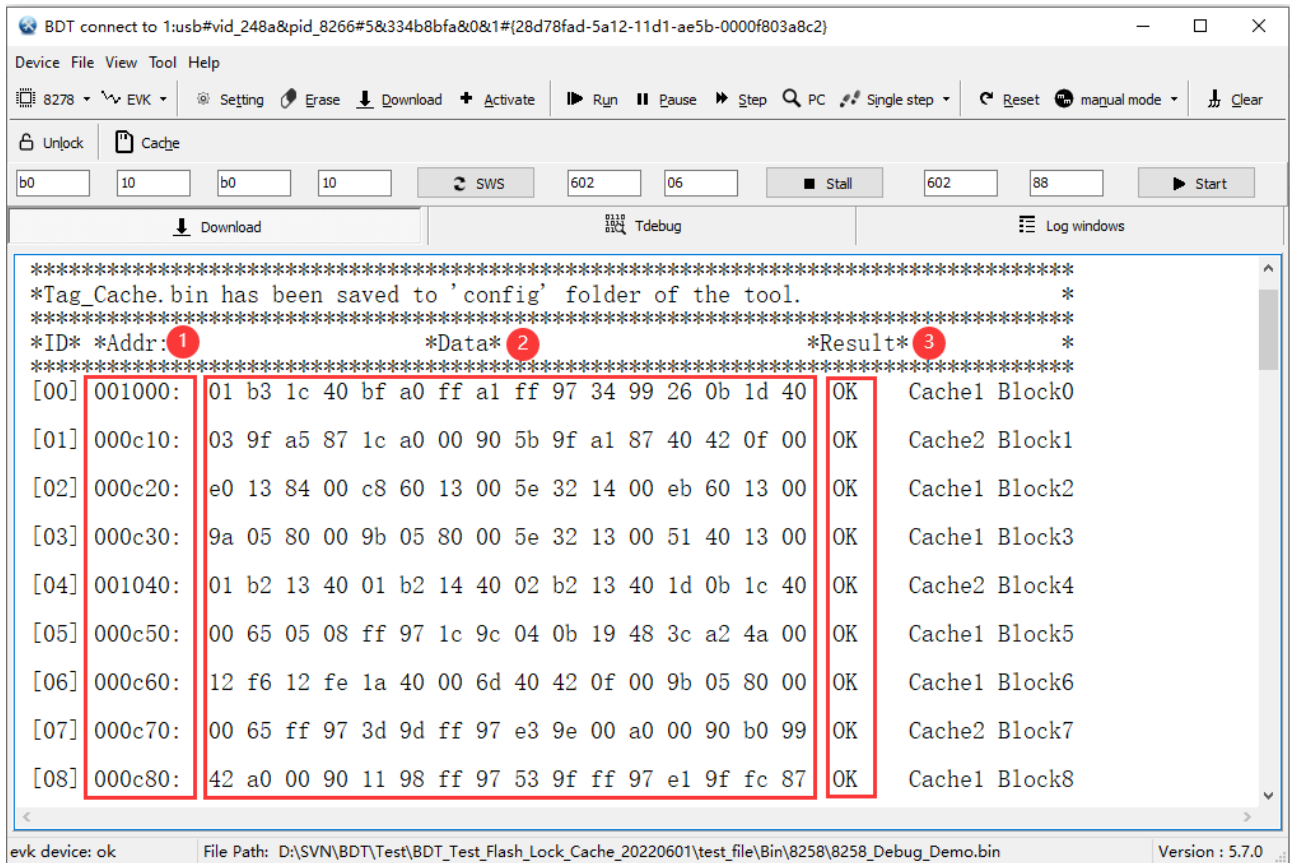


Figure 2.48: Cache 数据正确

Cache 数据出错时，“rawdata”代表 flash 地址对应的正确数据的值。“status”代表“rawdata”和 Cache 中的数据比较的结果。“status”为“T”，二者数据一致。“status”为“F”，则二者数据不一致。

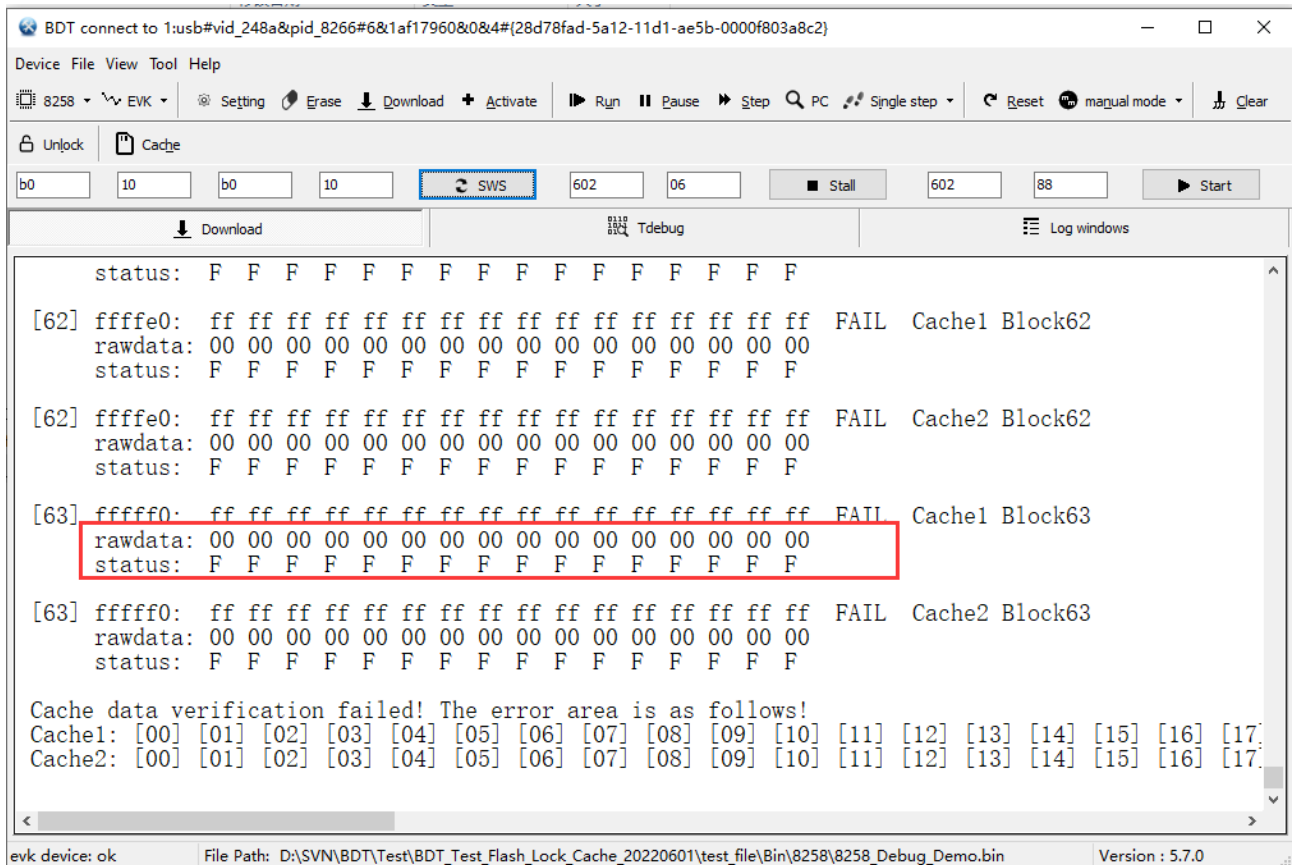


Figure 2.49: Cache 数据出错

2.5 Activate MCU

“与目标板通信失败时激活 MCU” 功能仅适用于 “Burning EVK” 与 “EVK” 模式下目标板之间的 Swire 连接，即不支持 “USB” 模式或 “Burning EVK” 与 “EVK” 模式下目标板之间的 USB 连接。

当固件烧录失败时，请确保目标板通过 Swire 与 “Burning EVK” 连接，然后点击 “Activate” 按钮启用此功能以激活 MCU。

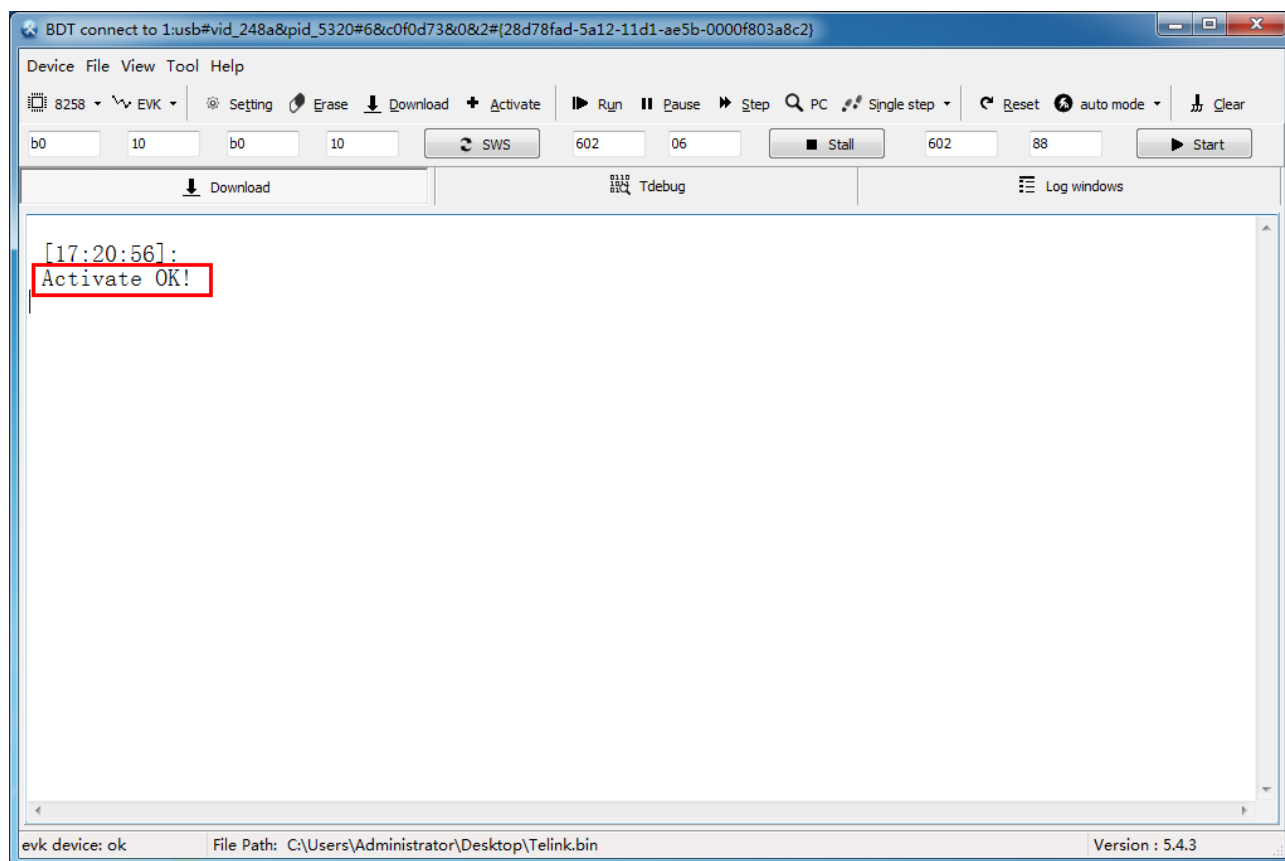


Figure 2.50: Activate MCU

2.6 Debug

MCU 开始运行后，要访问内存空间（Flash/Core/Analog/OTP），用户可以直接按组合快捷键“Ctrl+M”或点击“Tool -> Memory Access”，打开“Memory Access”窗口，如下图所示。

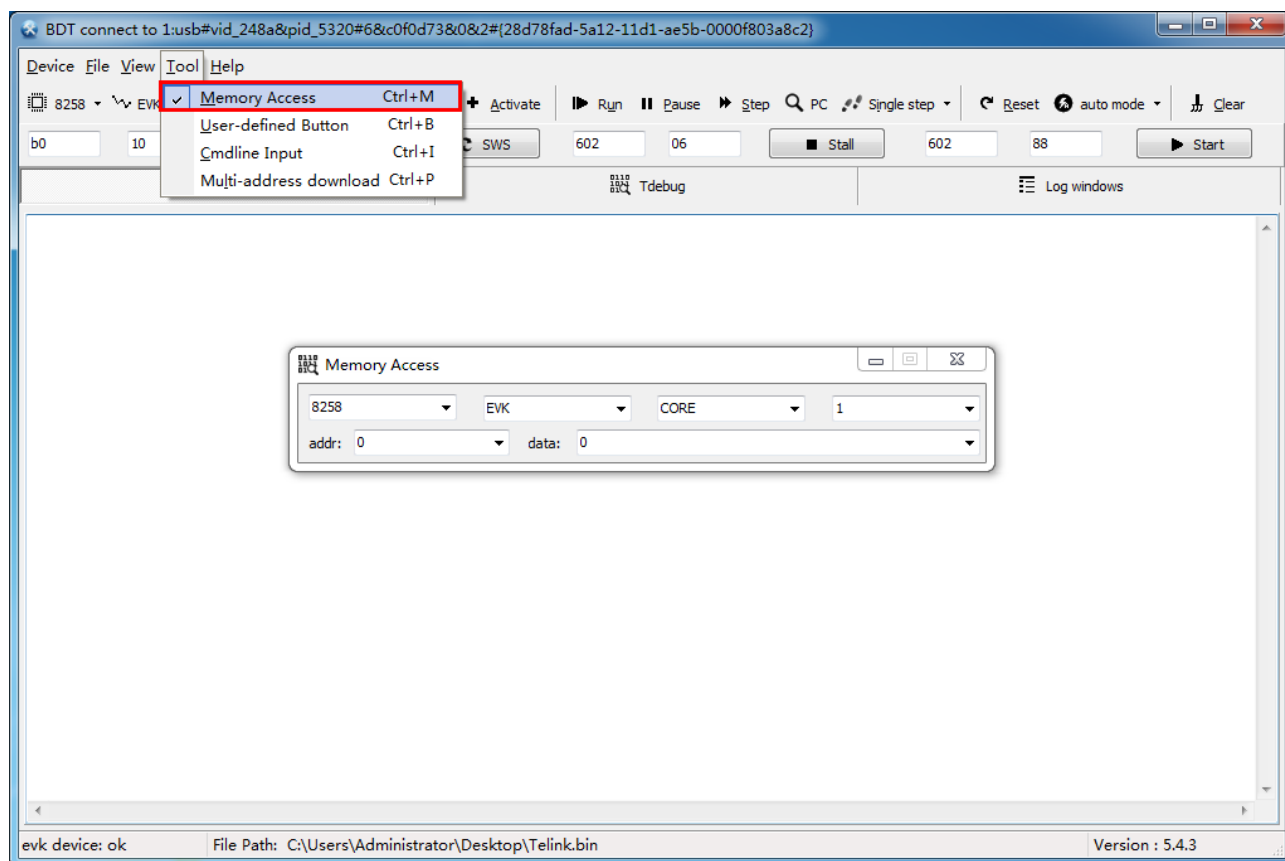


Figure 2.51: “Memory Access” 界面

Table 2.1: Memory Access 介绍

项目	值列表	默认	快捷键
MCU 类型	8366/8368/8266/8267/8269/8232/8233/8x5x	8258	Ctrl+Q
通信方式	EVK(Burning EVK)/USB	EVK	Ctrl+W
操作目标	Flash/Core ¹ /Analog ² /OTP	Core	Ctrl+E
操作大小 (单位: 字节)	1/2/4/8/16	1	Ctrl+R
地址	0	0	Ctrl+A
数据	0	0	Ctrl+D

¹Core 包括数字寄存器和 SRAM。

²Analog 指模拟寄存器。

2.6.1 访问存储器

2.6.1.1 从内存空间读取数据

想要从特定存储空间（Flash/Core/Analog/OTP）读取数据，用户可以执行以下步骤：

步骤 1：在“Memory Access”窗口中选择 MCU 类型（如“8258”）和与目标板的通信模式（如“EVK”）。

注意：

如果选择“USB”模式与目标板通信，请确保所选 MCU 支持 USB 功能且其 USB 功能可用。

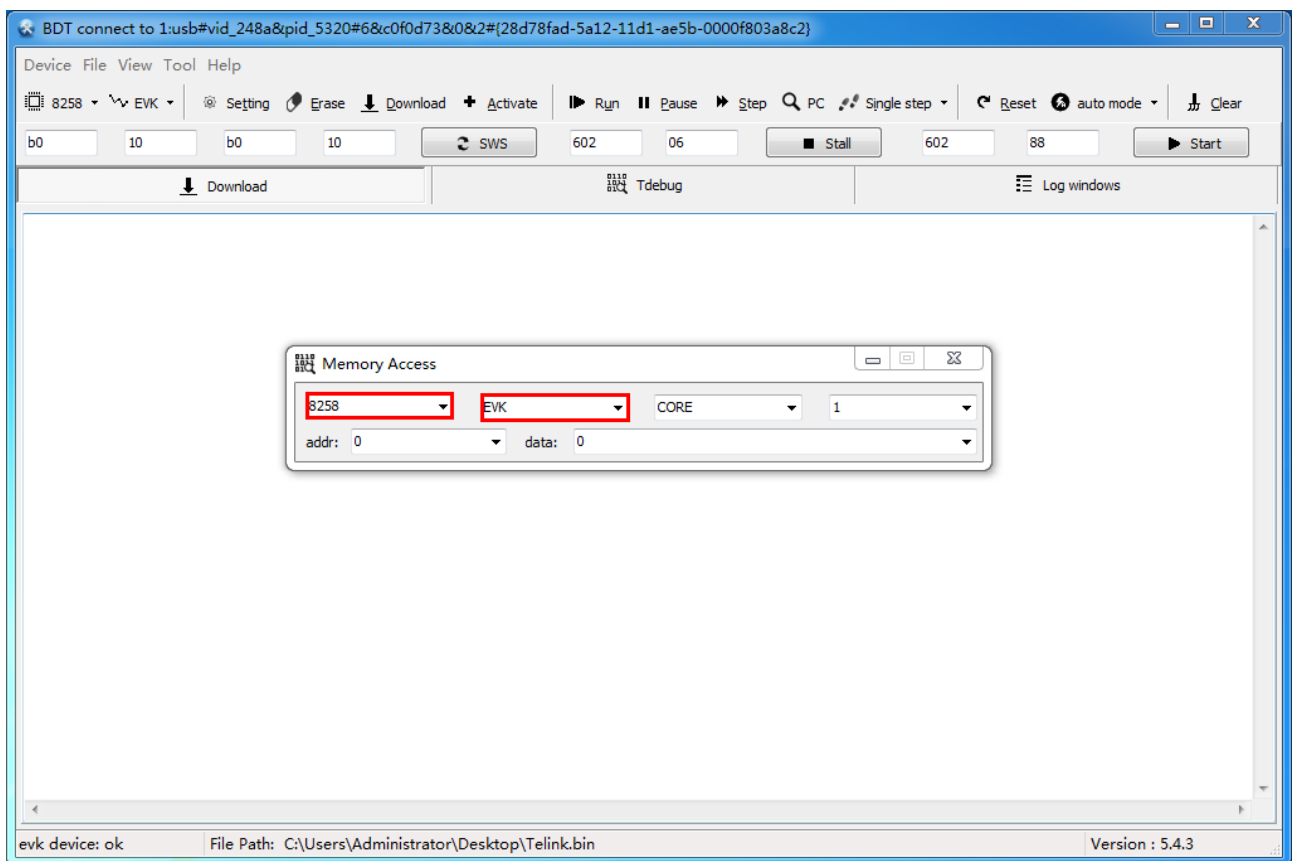


Figure 2.52: 选择 MCU 类型和通信模式

步骤 2：在“Memory Access”窗口中选择操作目标（如“CORE”）并设置操作大小（如“16”）。

注意：

- 如果选择“OTP”作为操作目标，请确保所选 MCU 支持“OTP”功能。
- 想要设置操作大小，用户可以通过下拉菜单选择大小（如果可用），或直接在编辑框中设置大小。
- 最大操作大小为 1MB。

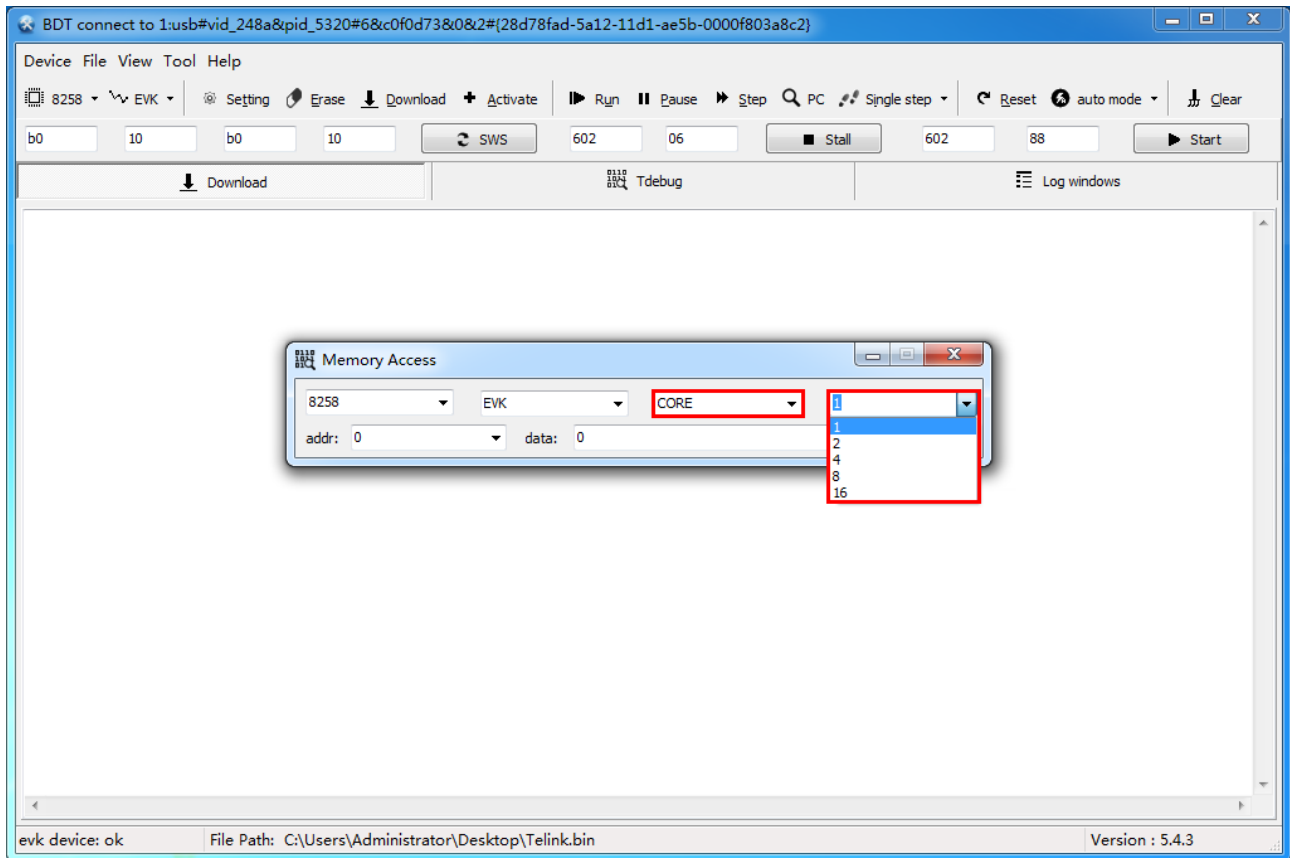


Figure 2.53: 设置操作目标和大小

步骤 3: 将鼠标移动到“addr”框，然后输入要读取的起始地址的偏移量（例如“0x040000”）。将光标保持在“addr”框中，通过单击“Tab”键，用户可以从指定的内存空间读取数据。

注意：

- 每次点击“Tab”键启动内存读取操作时，可编辑框的值将会被保存，以便下次可以通过下拉菜单直接选择它。

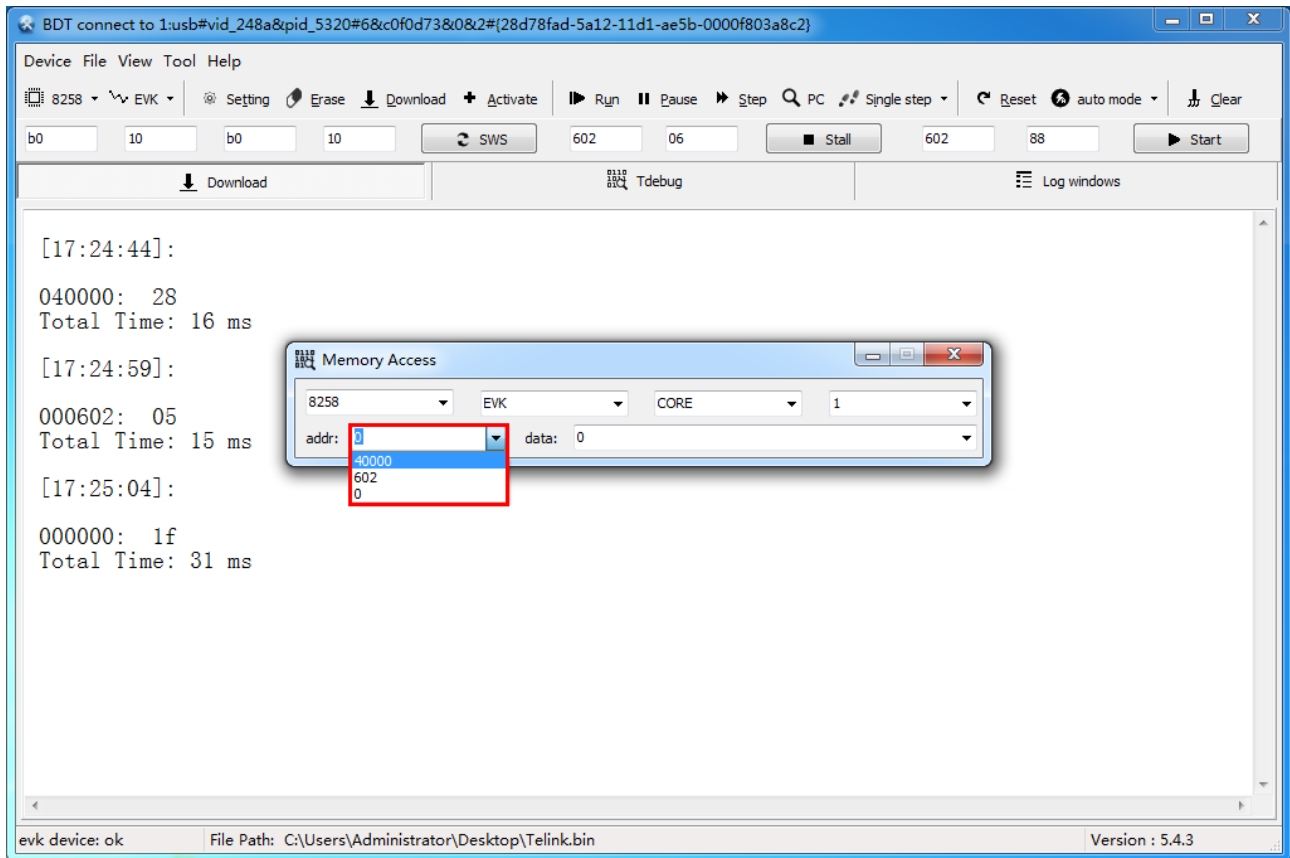


Figure 2.54: 设置起始地址并开始读取

- 当读取大于 1kB 的内存空间 (如 CORE) 时, 读取的数据将自动保存到 “user” 文件夹下名为 “read.bin” 的文件中。用户还可以使用特定的命令行更改文件名以保存读取的数据。详细信息请参阅第 2.8 节命令行。

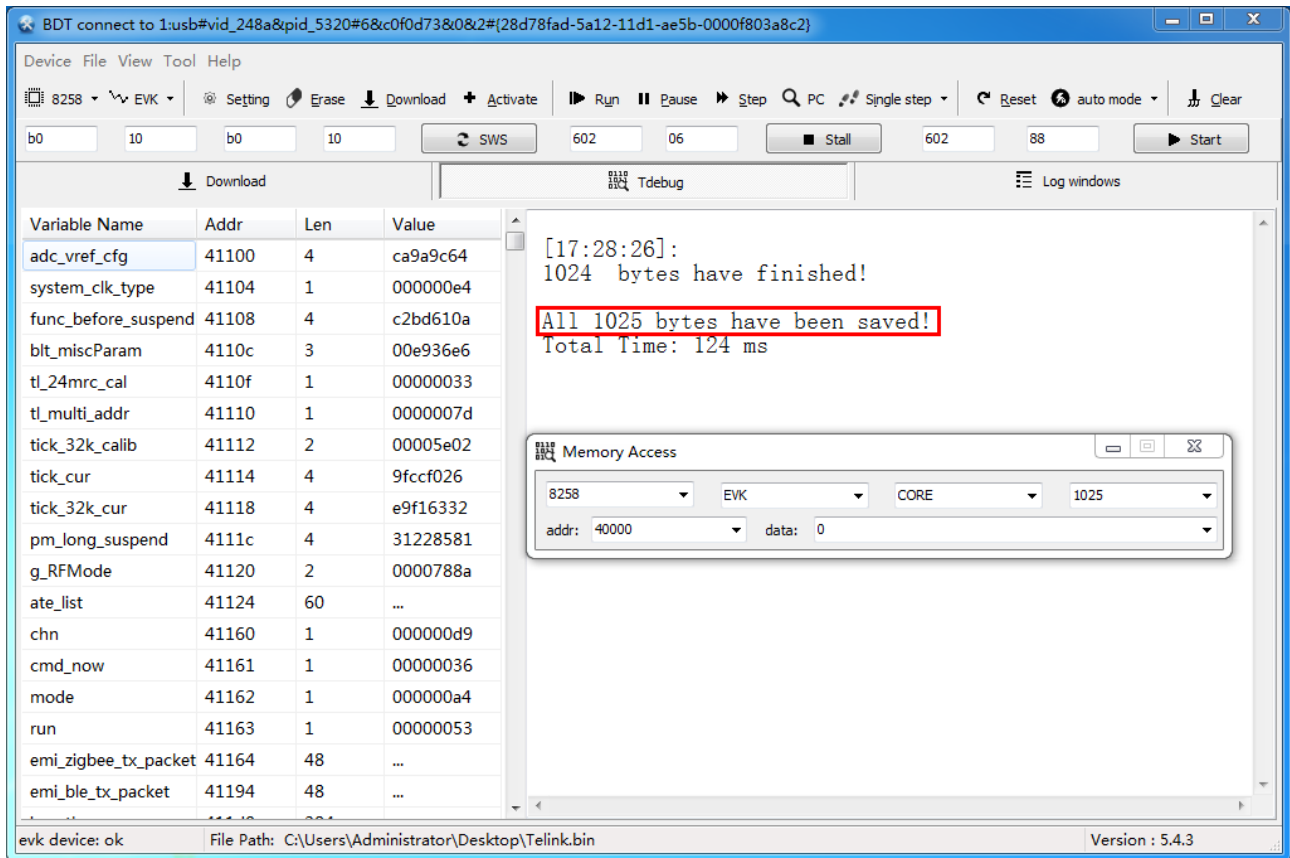


Figure 2.55: 保存读取的数据

2.6.1.2 将数据写入内存空间

将数据写入特定的内存空间，用户可以执行以下步骤：

步骤 1： 在“Memory Access”窗口中选择 MCU 类型（如“8258”）和与目标板的通信模式（如“EVK”）。

注意：

如果选择“USB”模式与目标板通信，请确保所选 MCU 支持 USB 功能且其 USB 功能可用。

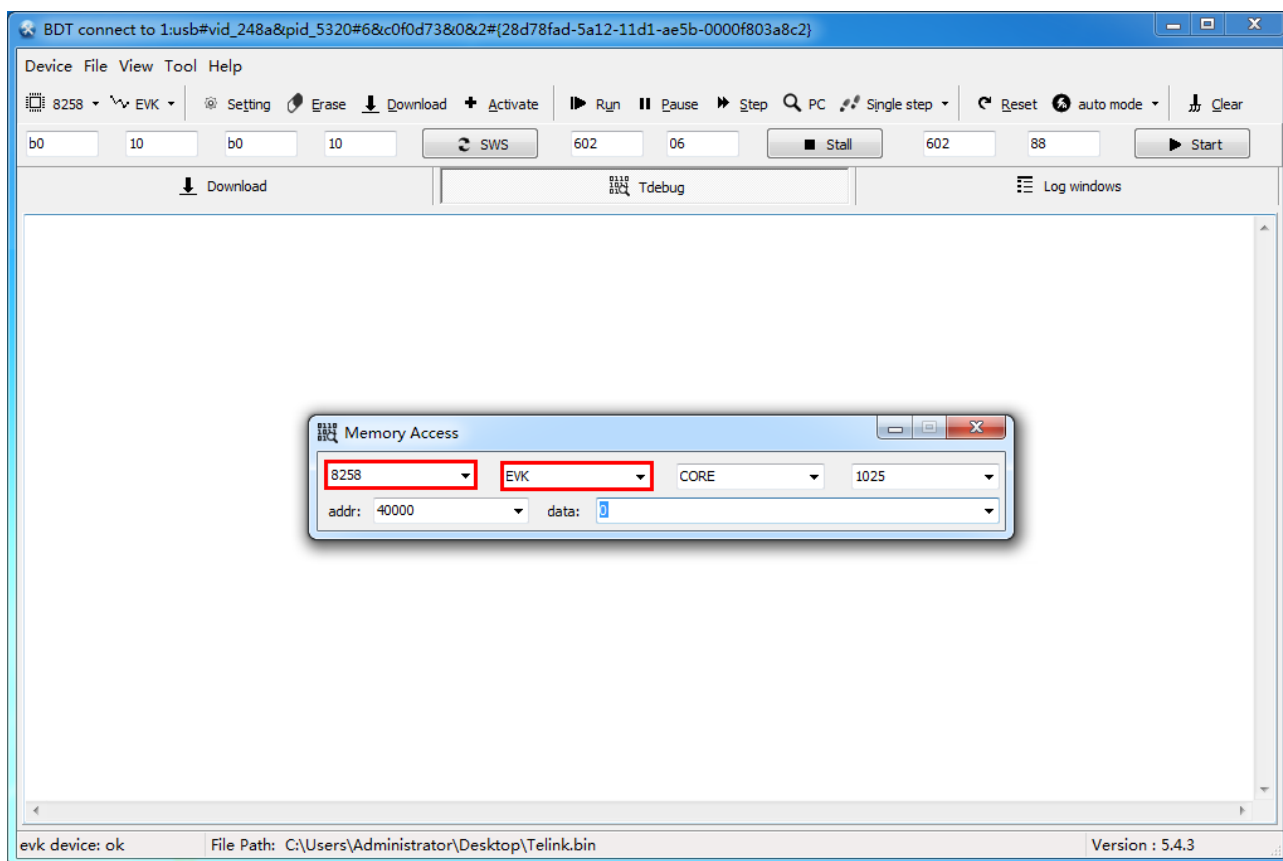


Figure 2.56: 内存访问窗口

步骤 2: 在“Memory Access”窗口中选择操作目标（如“CORE”）并设置操作大小（如“16”）。

注意:

- 如果选择“OTP”作为操作目标，请确保所选 MCU 支持“OTP”功能。
- 想要设置操作大小，用户可以通过下拉菜单选择大小（如果可用），或直接在编辑框中设置大小。
- 最大操作大小为 256 字节。

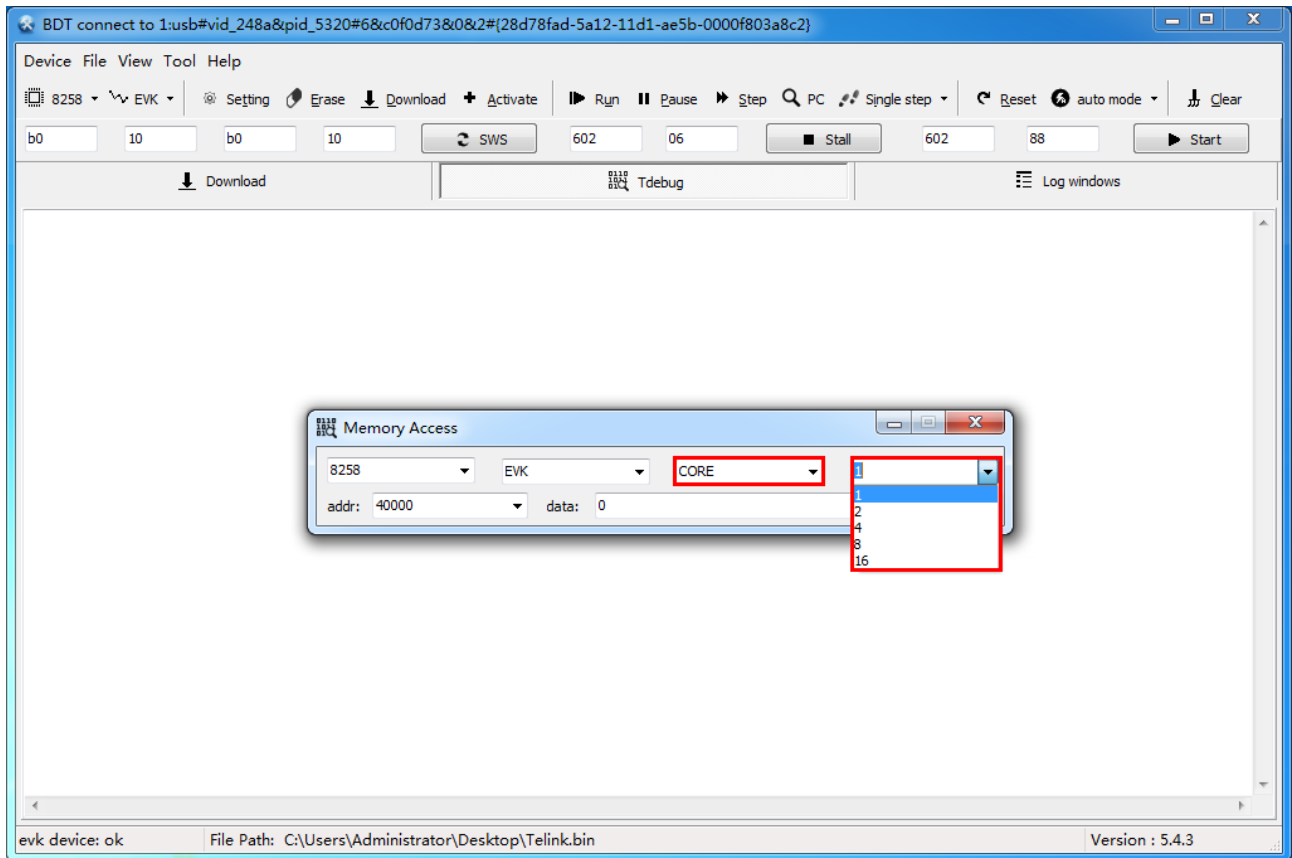


Figure 2.57: 设置操作大小

步骤 3: 将鼠标移动到“addr”框，然后输入要写入的起始地址的偏移量（例如“0x040000”）。

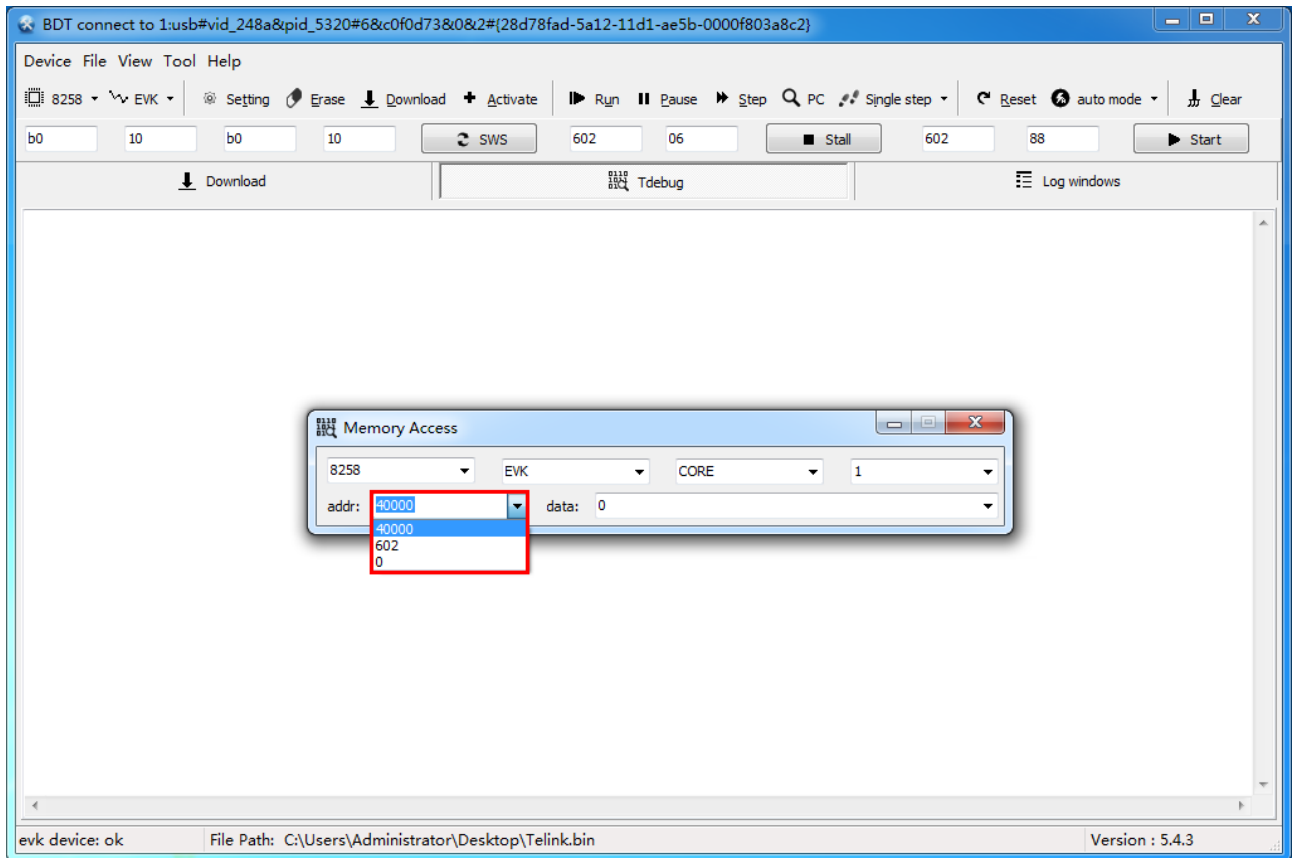


Figure 2.58: 起始地址的偏移量

注意：

当操作目标选择为“CORE”时，如果设置的地址小于“SRAM”的起始地址，则实际访问的目标存储空间为数字寄存器；如果设置的地址大于“SRAM”的起始地址，则实际访问的目标内存空间是 SRAM。

步骤 4：将数据写入指定的地址。

将鼠标移动到“data”框，然后输入要写入的目标数据。将光标保持在“data”框中，通过点击“Enter”键，用户可以将指定的数据写入指定的存储空间。

注意：

当输入数据时，字节之间必须用空格隔开，如下所示：

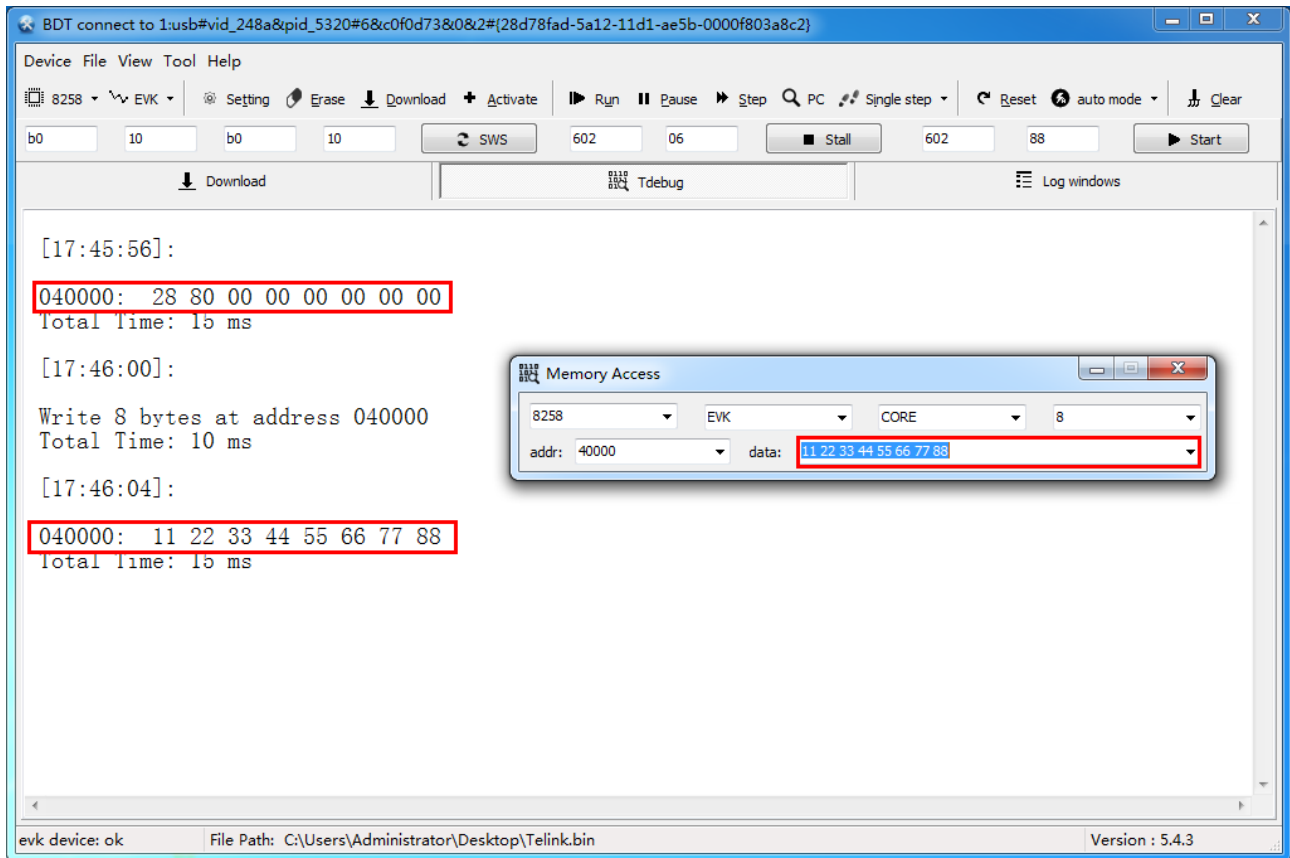


Figure 2.59: 设置数据并开始写入

2.6.2 操作变量列表

在查看所有全局变量之前，请确保“.lst”文件和“.bin”文件位于同一目录下，有可能出现两种情况，如下所示。

情况 1: 如果只有一个“.lst”文件和“.bin”文件位于同一目录下，则会自动选择这个“.lst”文件。

Interp_Cscript.lst	2018/9/11 10:03	LST 文件	317 KB
Telink.bin	2018/9/11 10:03	BIN 文件	17 KB

Figure 2.60: 情况 1

情况 2: 如果有多个“.lst”文件和“.bin”文件放在同一目录下，则自动选择与所选“.bin”文件同名的“.lst”文件。

Interp_Cscript.bin	2018/9/11 10:03	BIN 文件	17 KB
Interp_Cscript.lst	2018/9/11 10:03	LST 文件	317 KB
Telink.bin	2018/9/11 10:03	BIN 文件	17 KB
Telink.lst	2018/9/11 10:03	LST 文件	317 KB

Figure 2.61: 情况 2

例如：如果选择“Telink.bin”文件，将自动选择“Telink.lst”文件。如果没有“Telink.lst”，并且所选“Telink.bin”文件的目录下仅放置了一个“xxx.lst”文件，则将会选择唯一的“xxx.lst”文件。

2.6.2.1 更新变量列表

想要查看所有全局变量，用户可以选择“Tdebug”页面，将鼠标移动到变量列表，然后右键单击鼠标选择“Refresh”或按快捷键“F3”。每次按“F3”，变量列表就会更新一次。

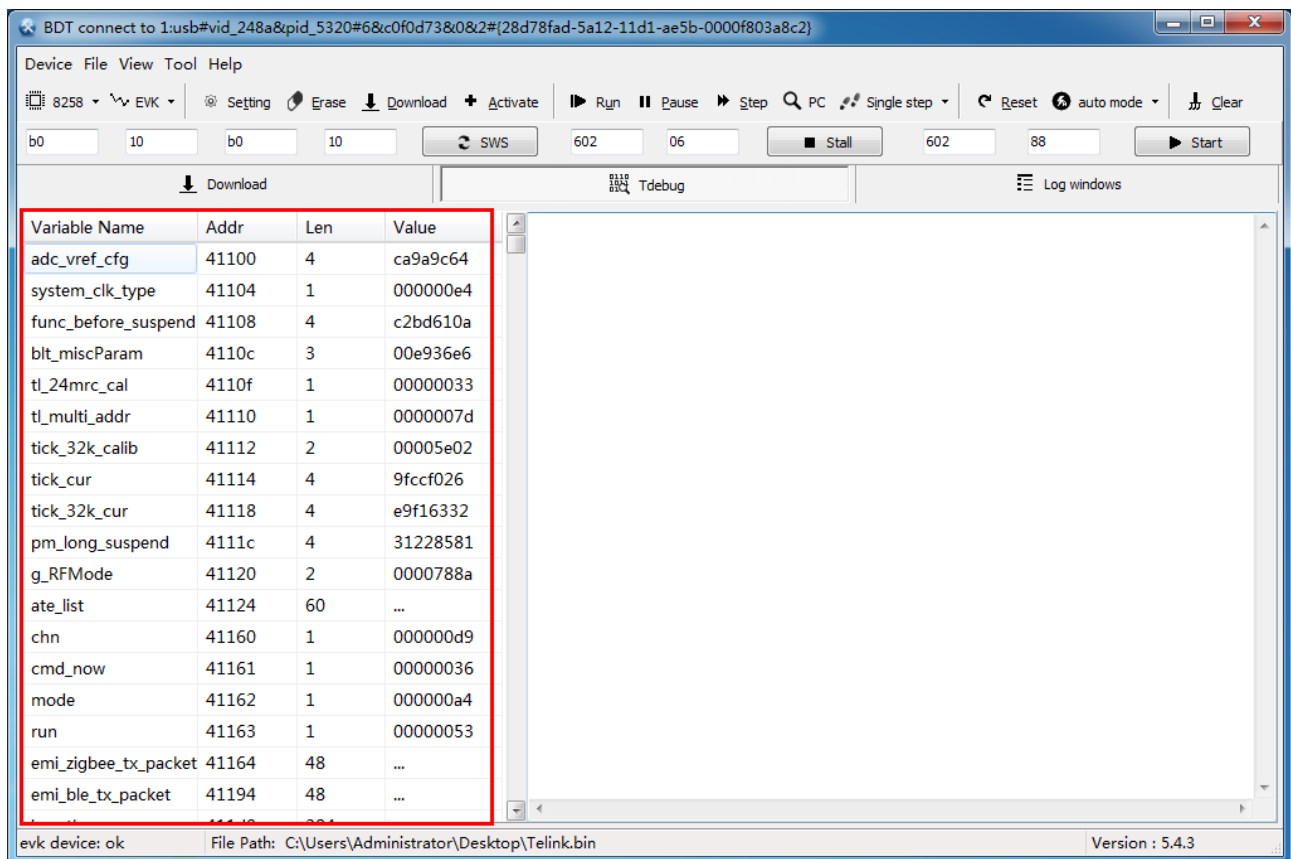


Figure 2.62: 更新变量列表

2.6.2.2 对变量进行排序

根据需要，用户还可以右键单击鼠标，选择“Sort by address”将所有变量按地址从低到高排序（默认），或选择“Sort by name”将所有变量按变量名称的 ASCII 码值从 A 到 z 排序（比较第一个不同的字符）。

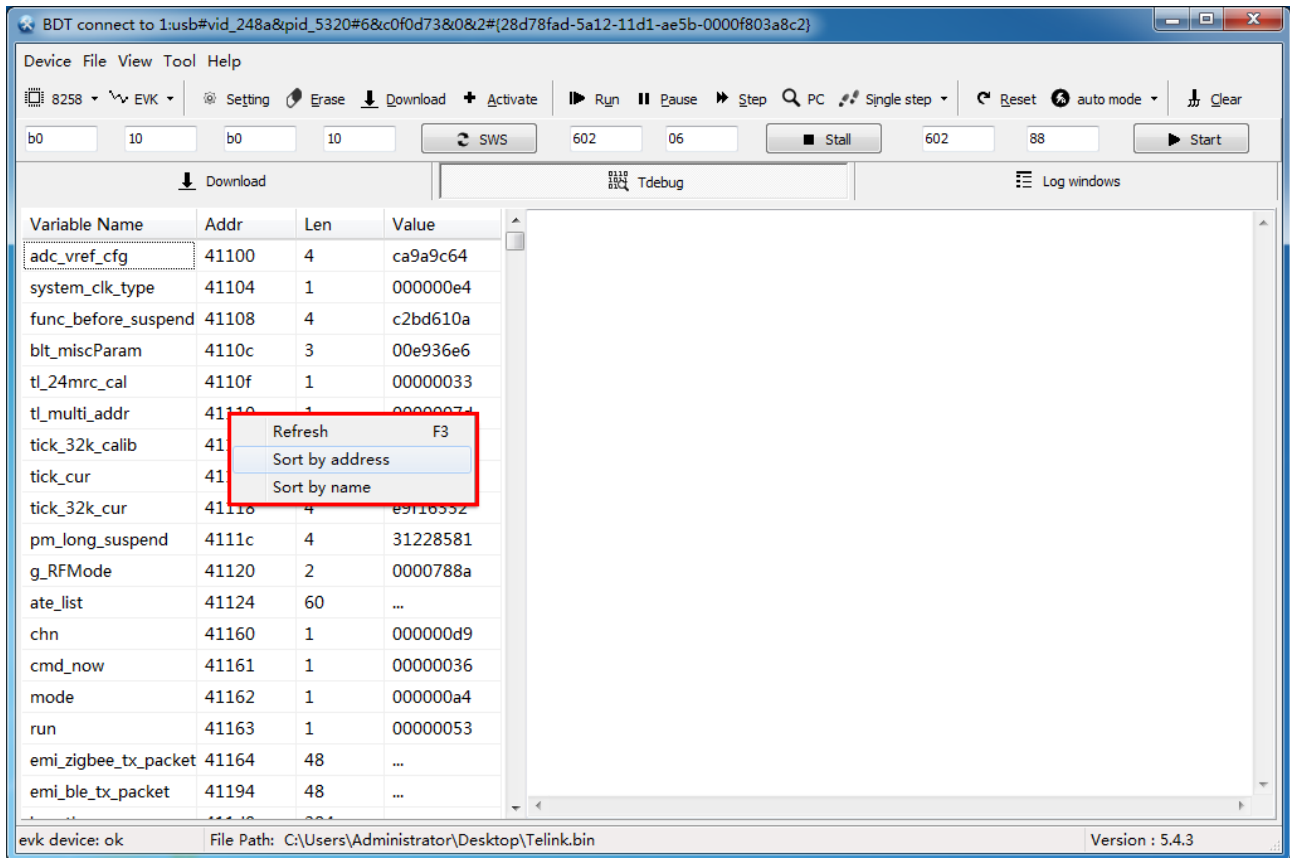


Figure 2.63: 按名称对变量排序

2.6.2.3 查看超过 4 字节的变量

如果变量的长度大于 4 字节，通过点击“variable list”的“Value”列中的“...”，其值将显示在右侧的日志窗口中。

如果变量的长度大于 1024 字节，则该变量的所有值都将保存在名为“Read.bin”的文件中。

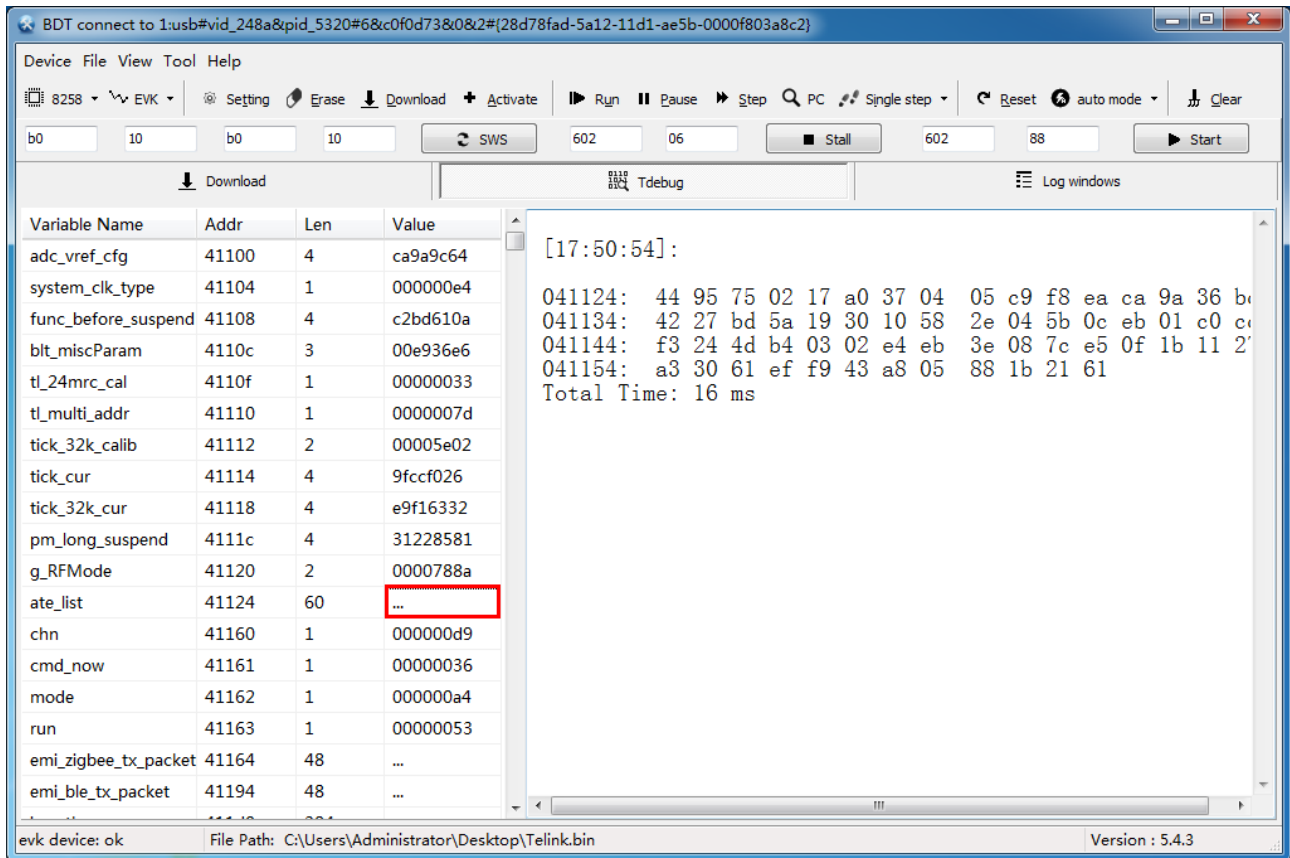


Figure 2.64: 读取大于 4 字节的变量值

2.6.2.4 修改变量值

长度不超过 4 字节的变量支持直接写入操作，双击“Value”列中相应的框可以直接修改其值。

例如：要将地址“0x41100”中变量“adc_vref_cfg”的值修改为“0x00000004”，用户可以执行以下步骤：

步骤 1：双击指定变量对应“Value”列中的框（例如“adc_vref_cfg”），使该框可编辑。

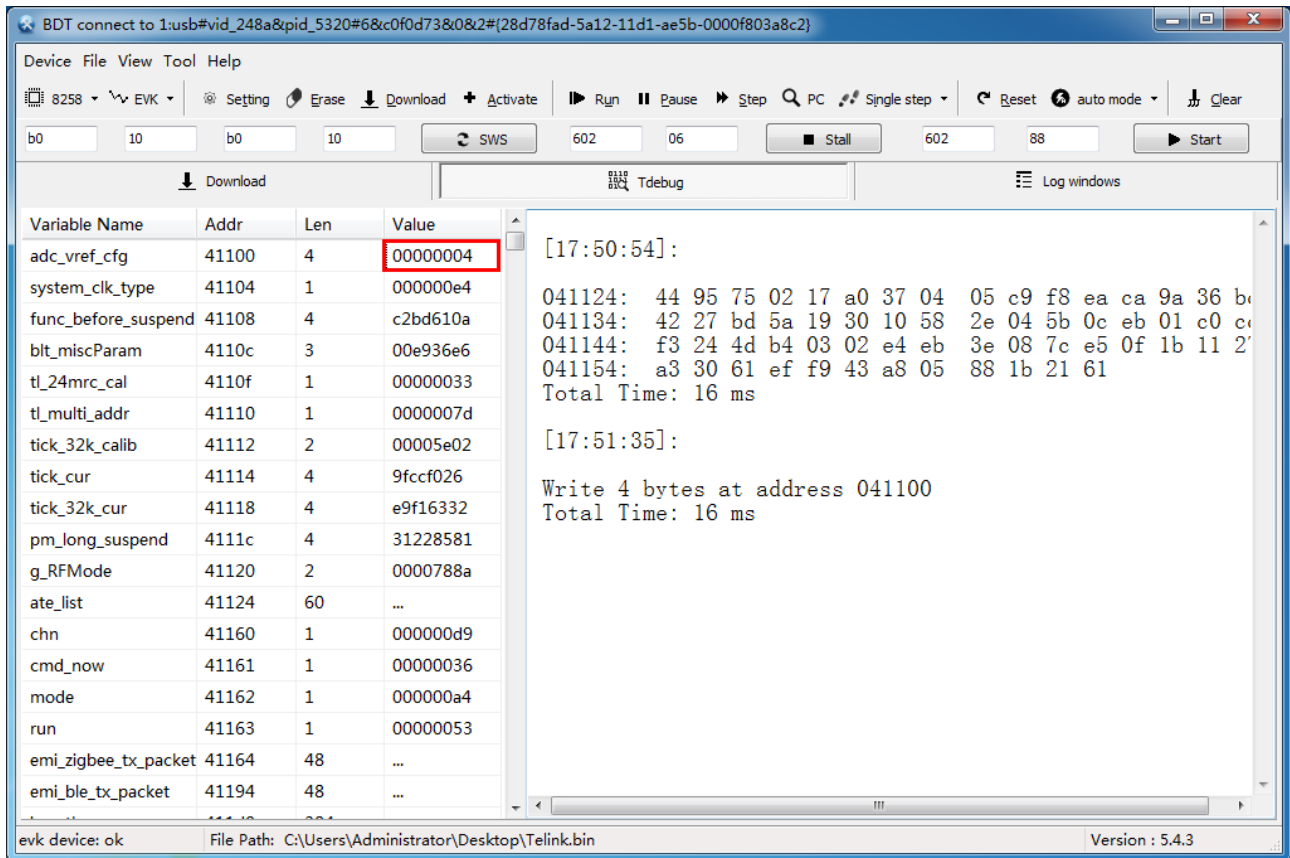


Figure 2.65: 双击指定的“Value”框

步骤 2: 输入新的变量值（例如“0x00000025”）来替换旧的变量值（“0x00000004”）。

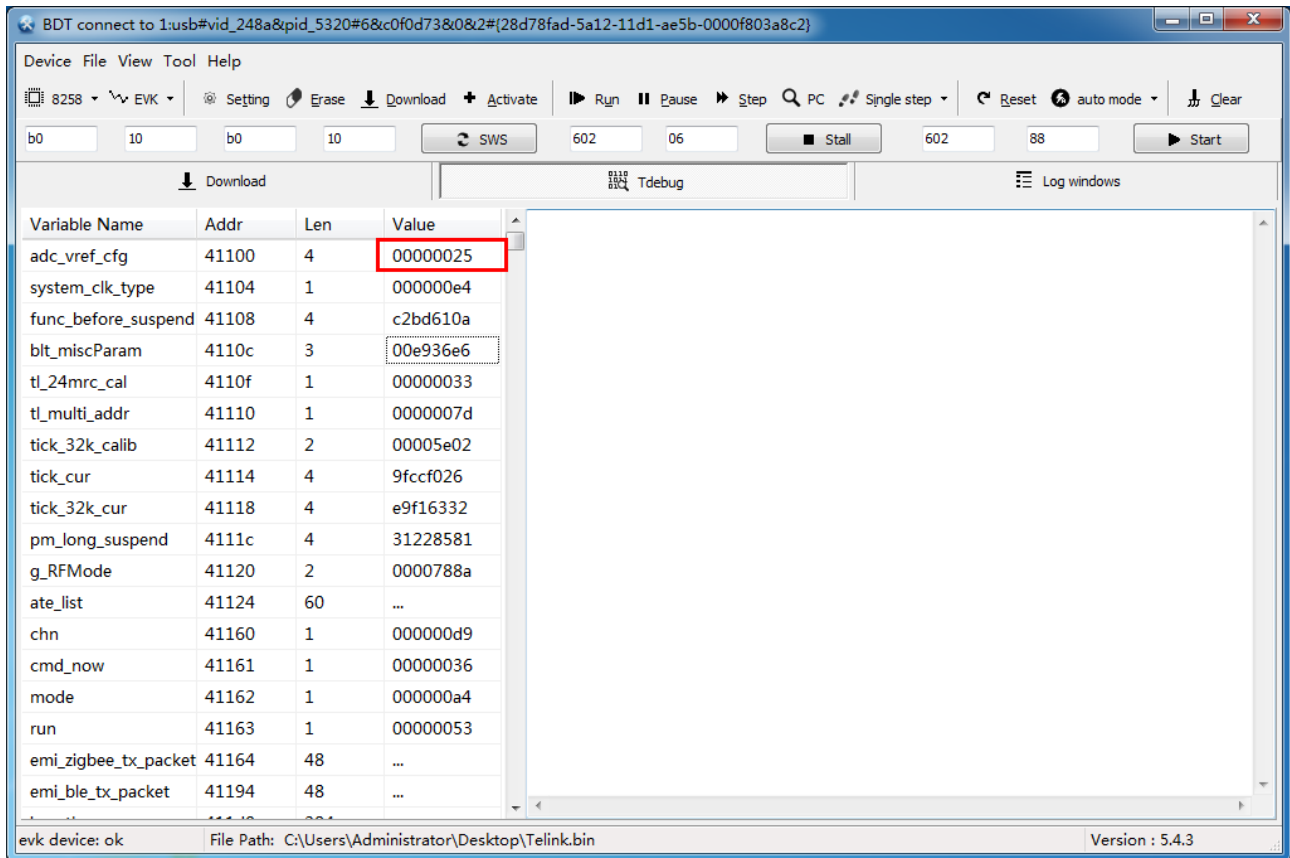


Figure 2.66: 修改变量值

步骤 3: 按 “Enter” 键将新的值写入指定变量的地址中。

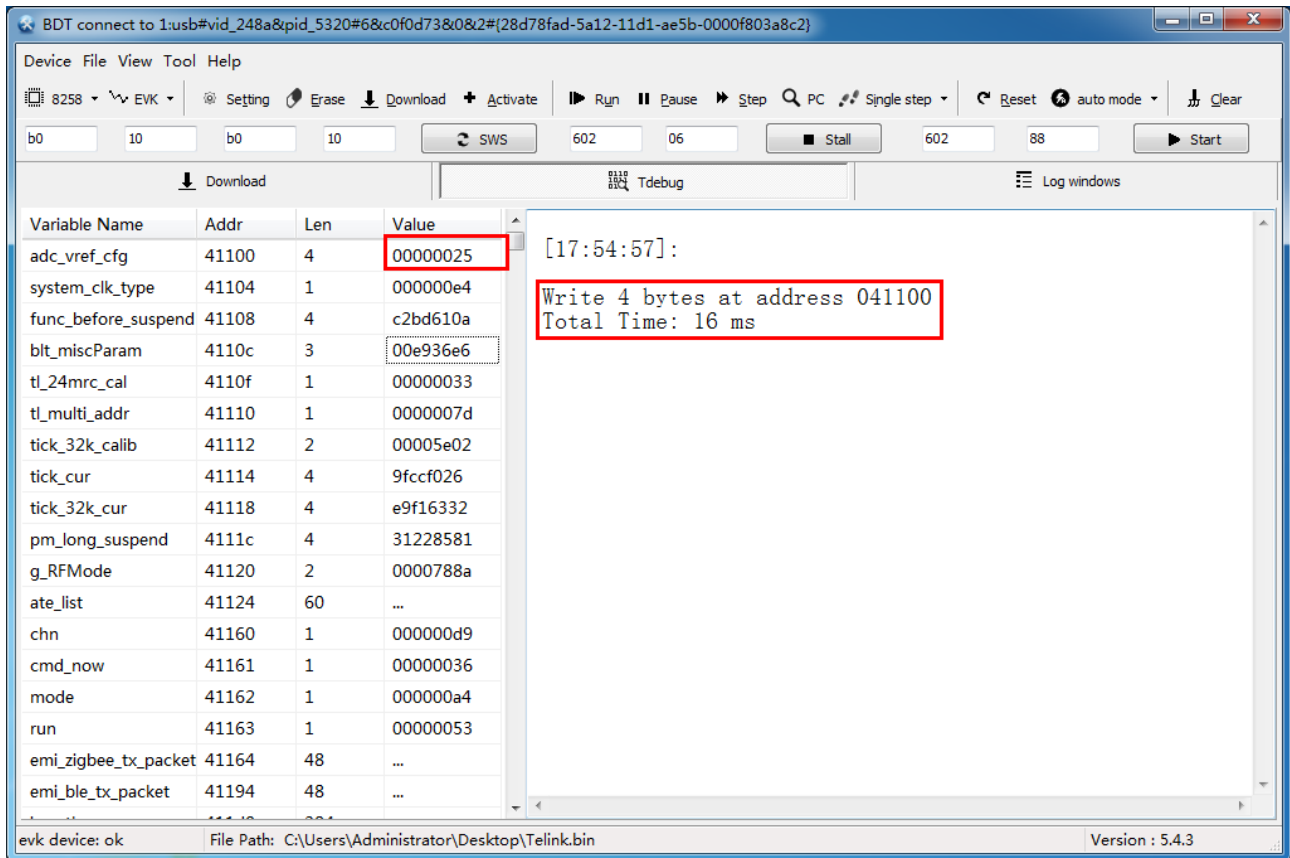


Figure 2.67: 将数据写入指定地址

2.6.3 Debug MCU

在使用“Run”、“Pause”、“Step” and “PC”等按钮调试 MCU 之前，请确保所选 MCU 支持此功能。有关详细信息，请参阅第 5 章功能支持列表。

在下面的小节中，以 MCU “8258” 为例介绍该功能。

2.6.3.1 Run MCU

点击“Pause”或“Step”按钮后，用户可以通过点击“Run”按钮使 MCU 从当前位置继续运行。请点击“PC”按钮查看 MCU 状态，读取 PC 并确保程序再次运行。

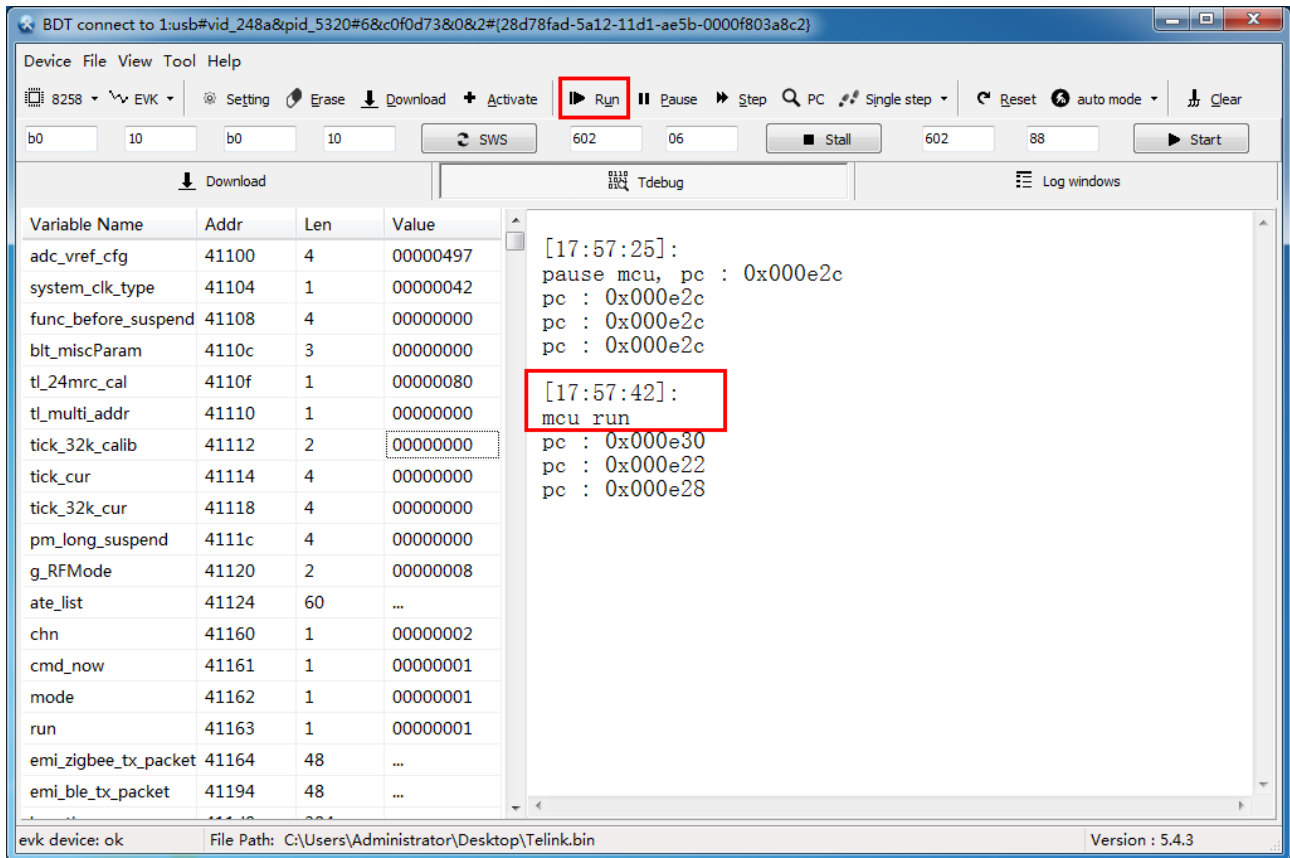


Figure 2.68: Run MCU

2.6.3.2 Pause MCU

想要查看 MCU 状态的详细信息，用户可以点击“Pause”按钮暂停 MCU。点击“Run”按钮，MCU 将从当前位置继续运行。

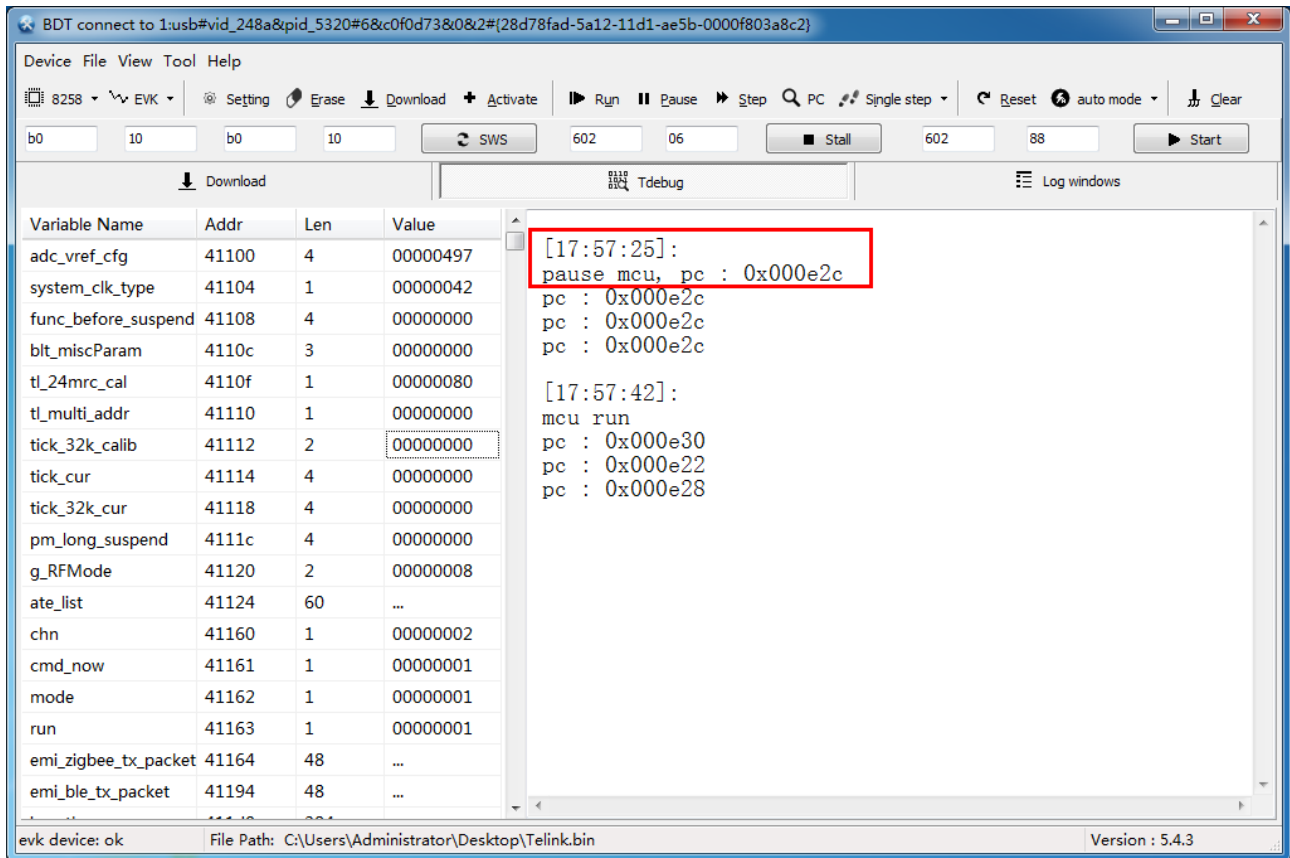


Figure 2.69: Pause MCU

2.6.3.3 Step MCU

如果可以与目标板通信，用户可以点击“Step”按钮逐步查看指令的当前位置。

“Trace MCU”和“Step MCU”都支持以下两种模式。

- (1) 如果选择了“Single step”模式，MCU 将进入单步模式，在点击“PC”或“Step”按钮后，将实施一次 PC 追踪。

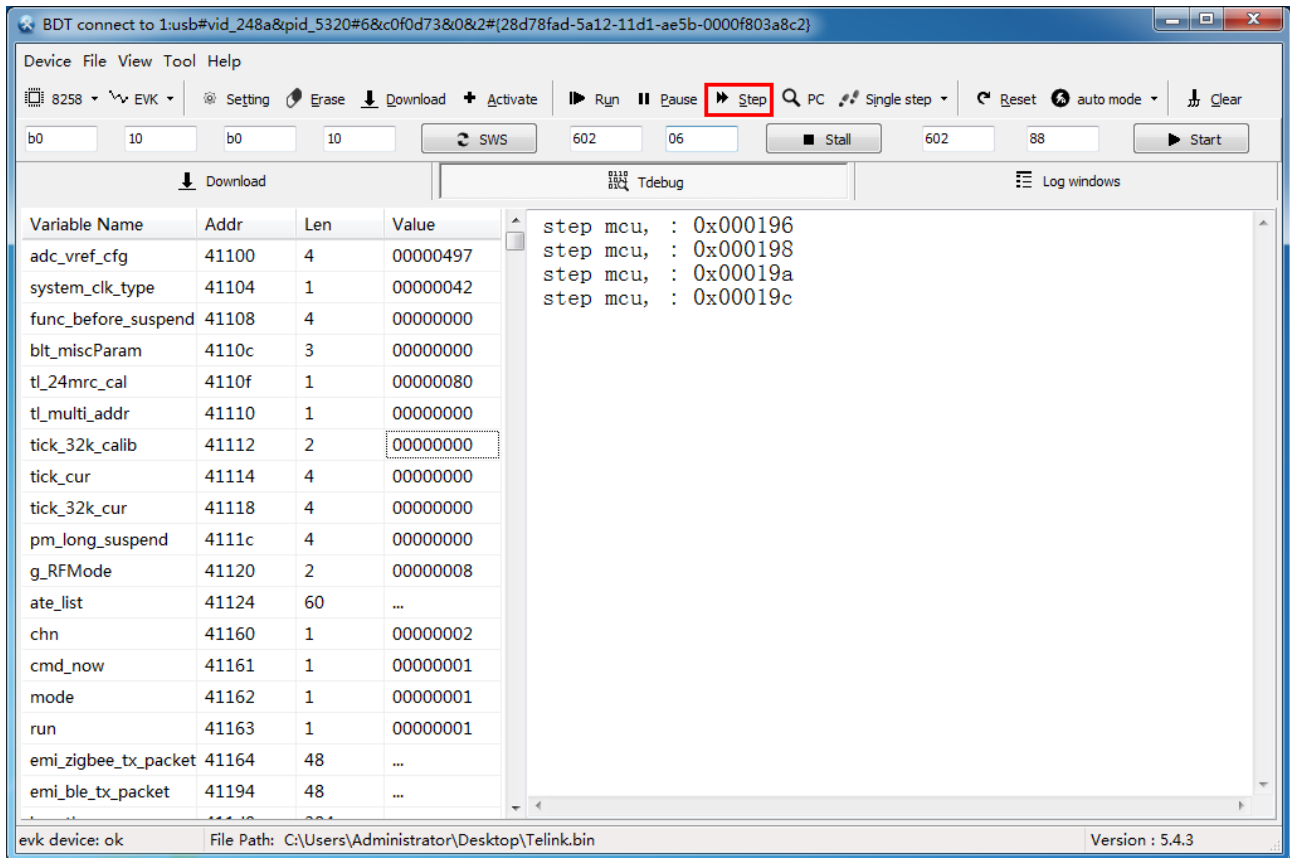


Figure 2.70: 在“Single step”模式下 Step MCU

(2) 如果选择“continuous”模式，MCU 将进入连续模式，实施连续的 PC 追踪或使 MCU 连续步进。

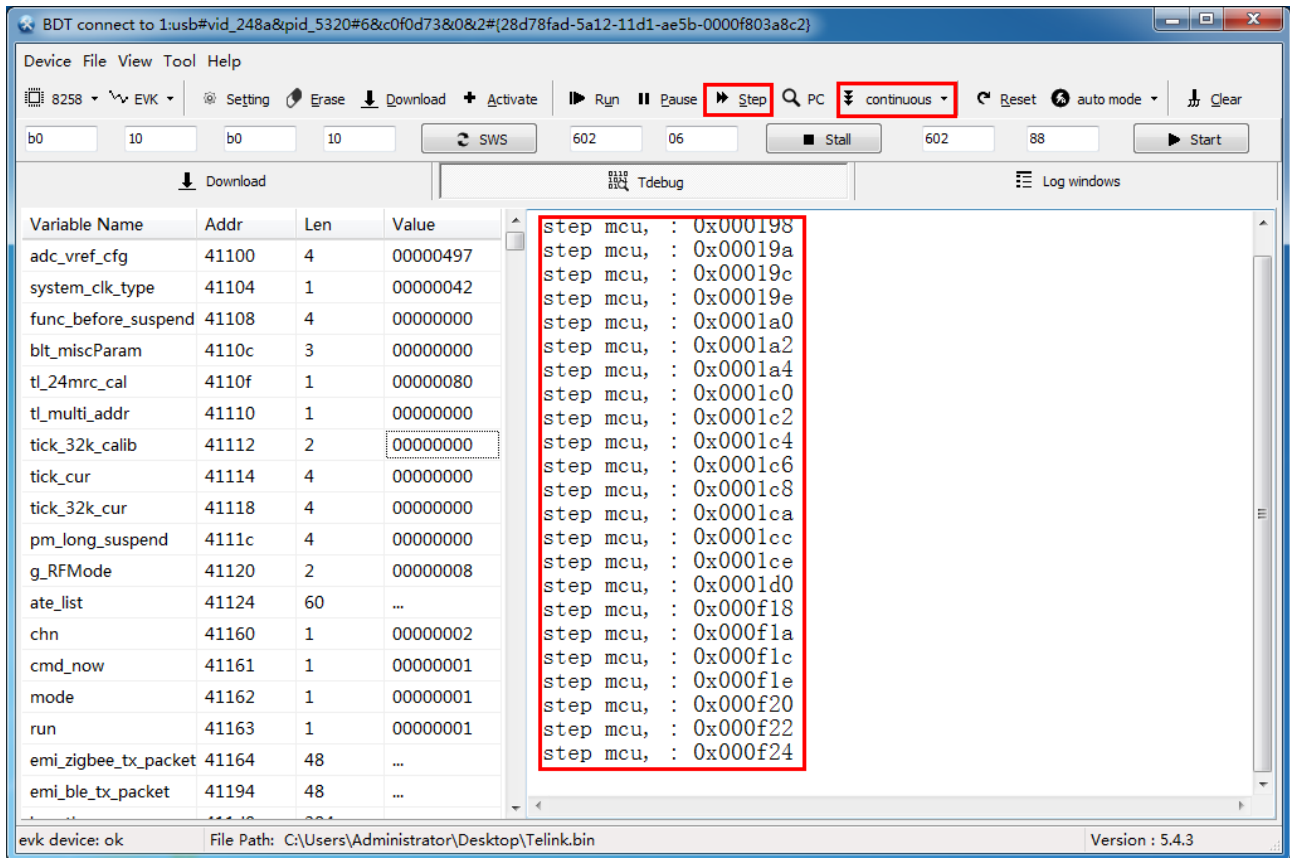


Figure 2.71: 在“continuous”模式下 Step MCU

想要取消“continuous”模式，用户可以将模式切换到“Single step”模式，或者点击“Run”/“Pause”按钮。

如果目标固件（bin 文件）和“xxx.lst”文件位于同一目录下，用户可以通过选择“View -> interp .lst”并点击“PC”按钮解析“xxx.lst”文件来查看更多详细信息，如下所示。参考第 2.6.2 节操作变量列表。

	Telink.bin	2018/3/1 17:55	BIN 文件	3 KB
	Telink.lst	2018/3/1 17:55	LST 文件	44 KB

Figure 2.72: “xxx.lst”文件

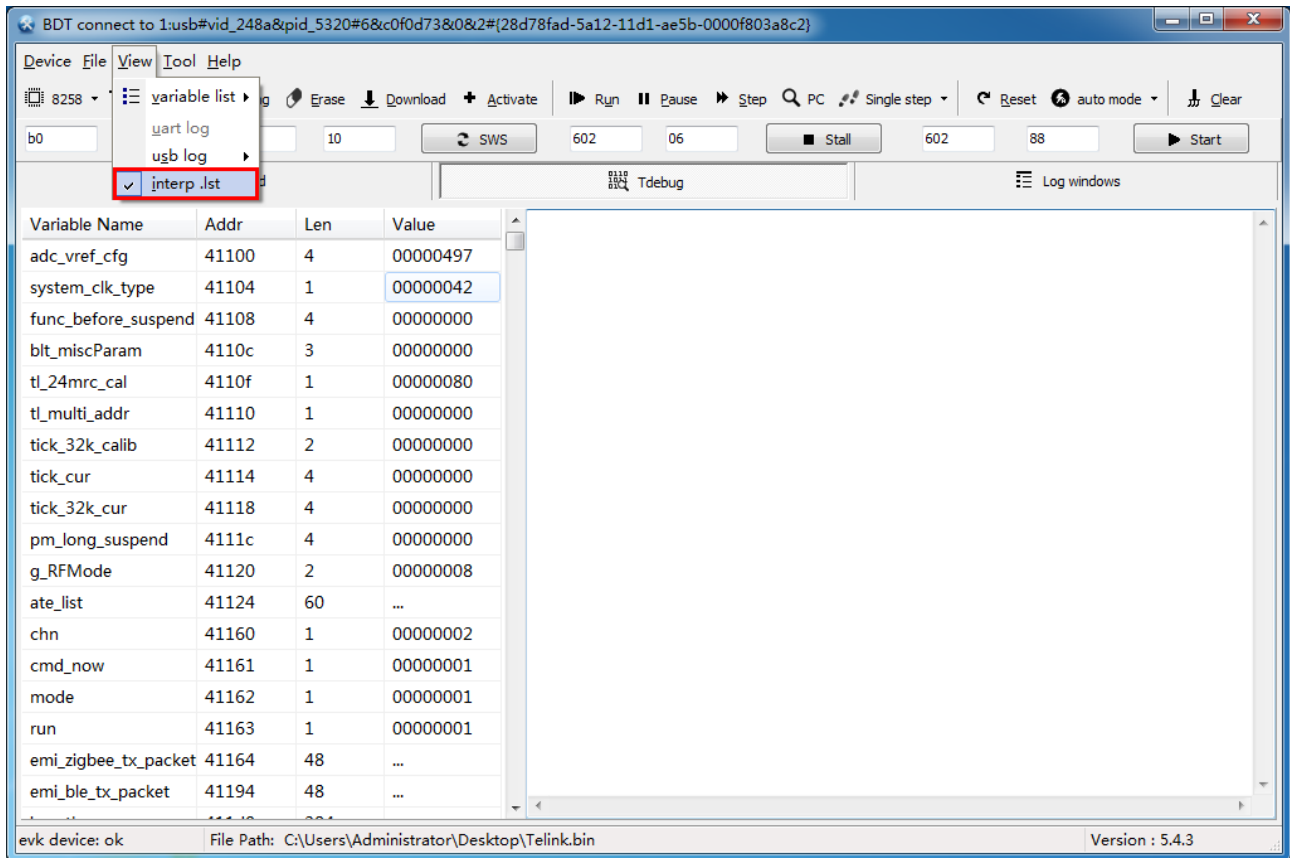


Figure 2.73: 选择 “interp.lst”

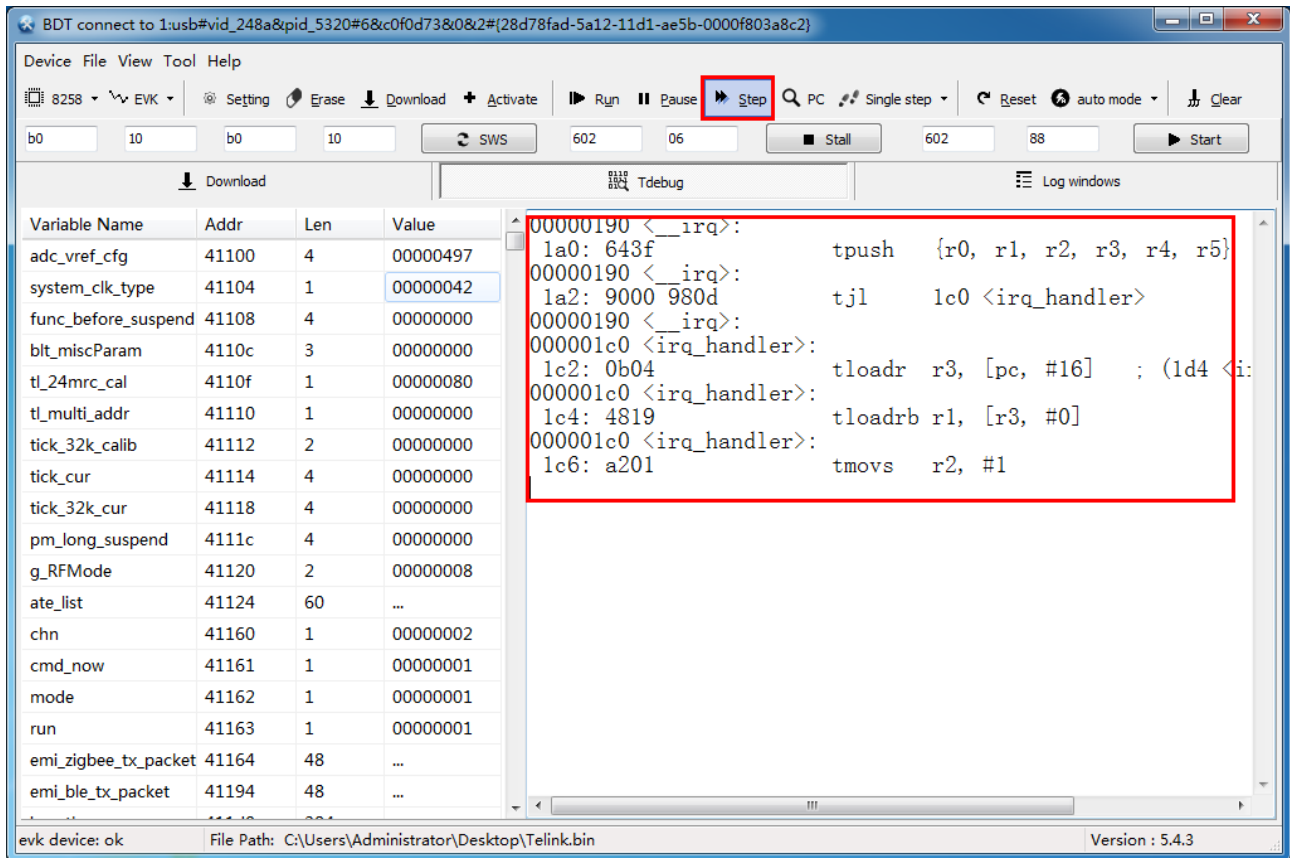


Figure 2.74: 在“Single step”模式下选择“inter.lst” Step MCU

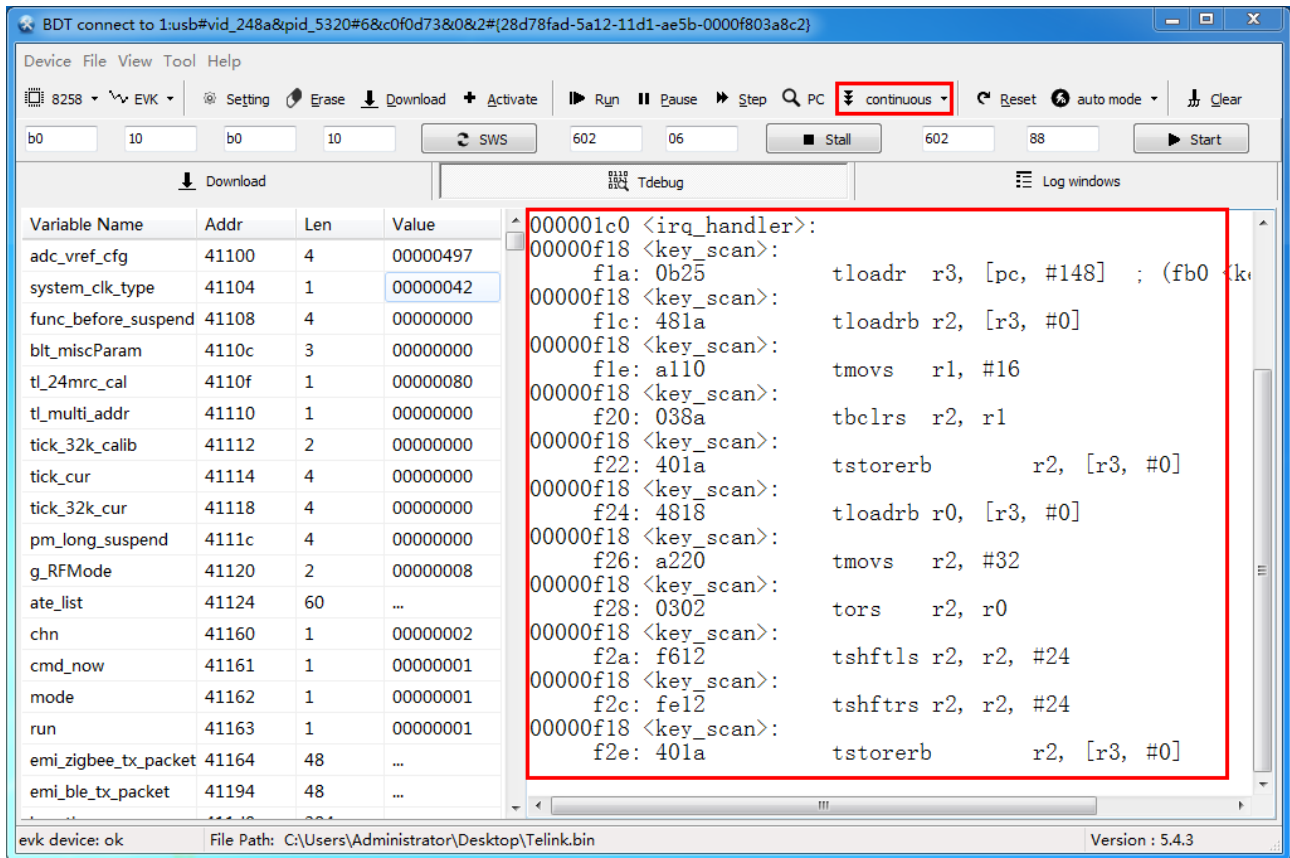


Figure 2.75: 在“continuous”模式下选择“interp.lst” Step MCU

2.6.3.4 Trace PC

如果可以与目标板通信，用户可以点击“PC”按钮查看指令的当前位置，如下所示：

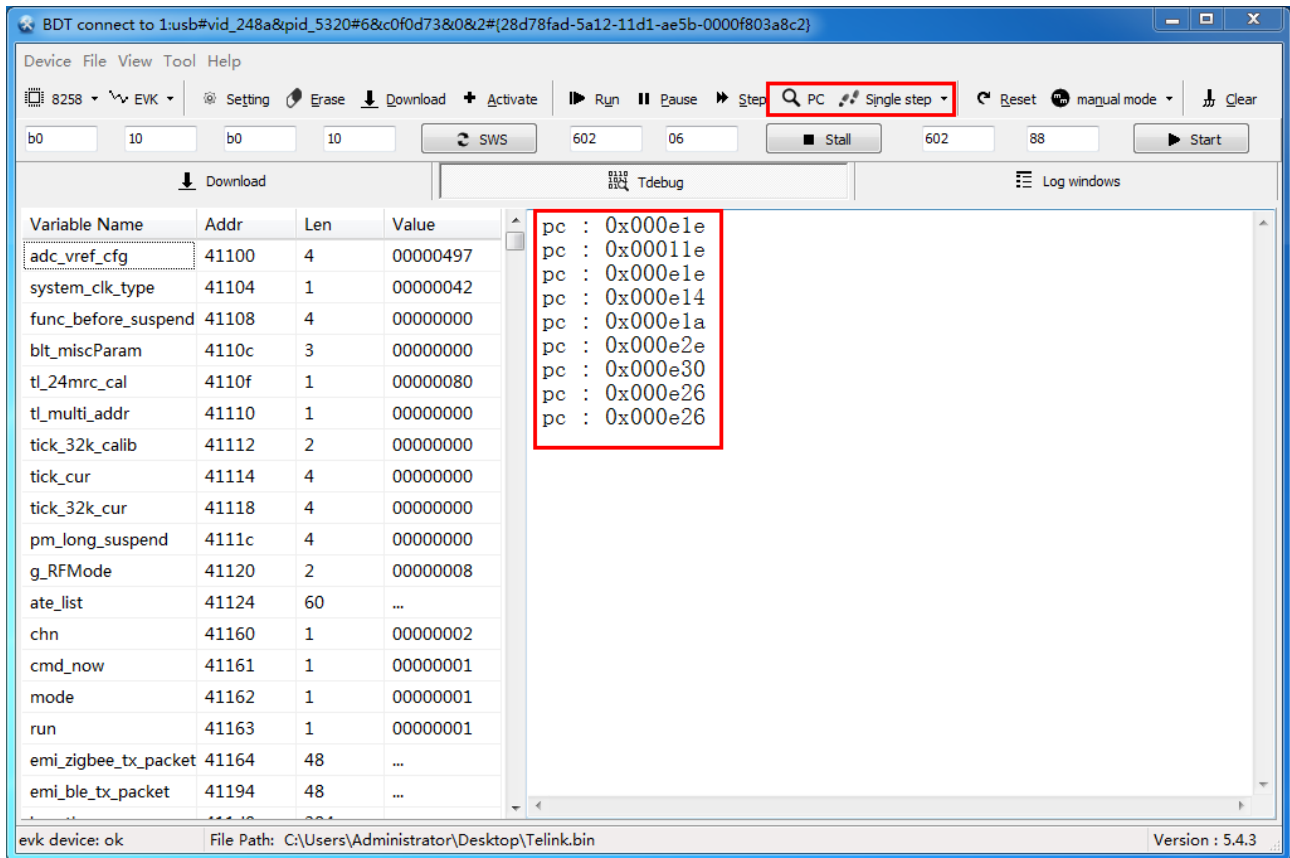


Figure 2.76: 在 “Single step” 模式下 Trace PC

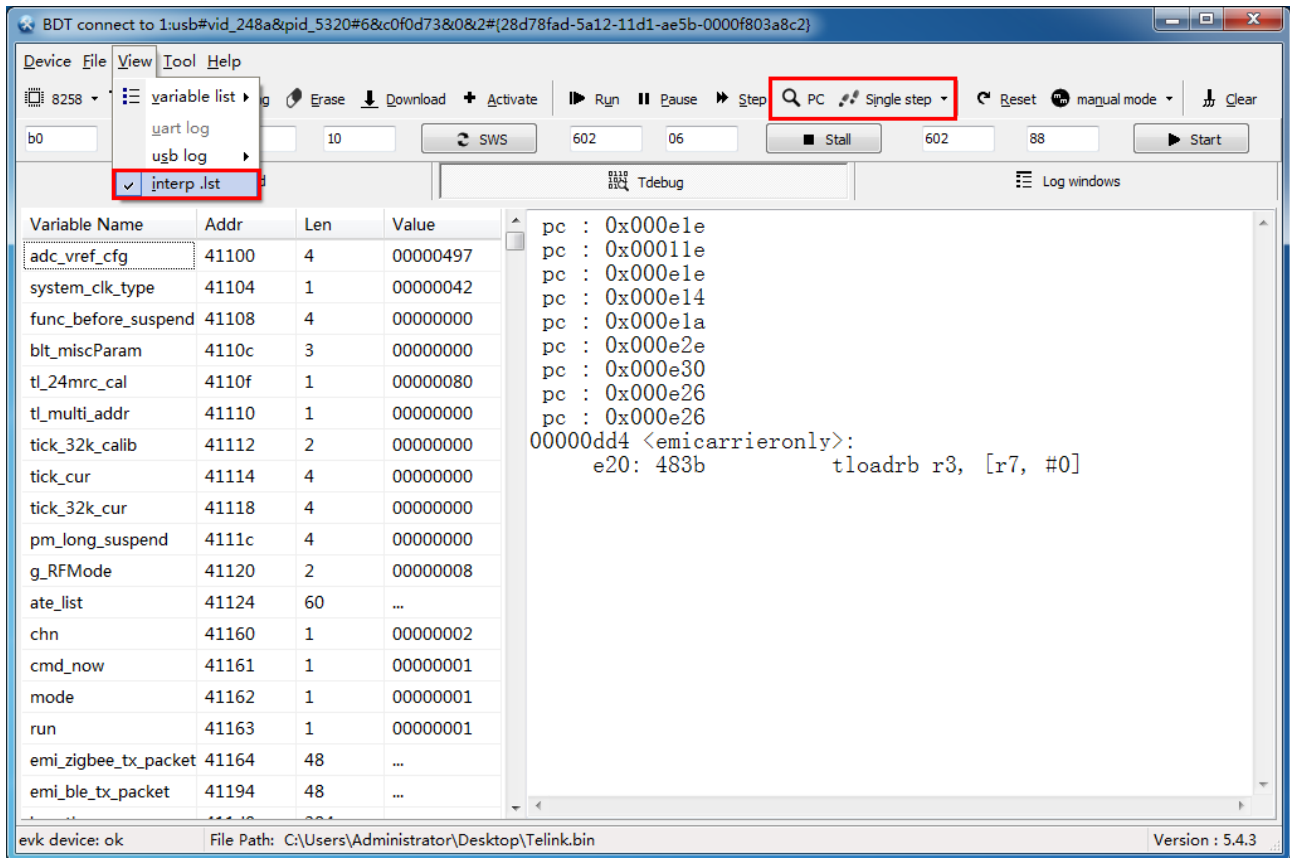


Figure 2.77: 在“Single step”模式下选择“interp.lst” Trace PC

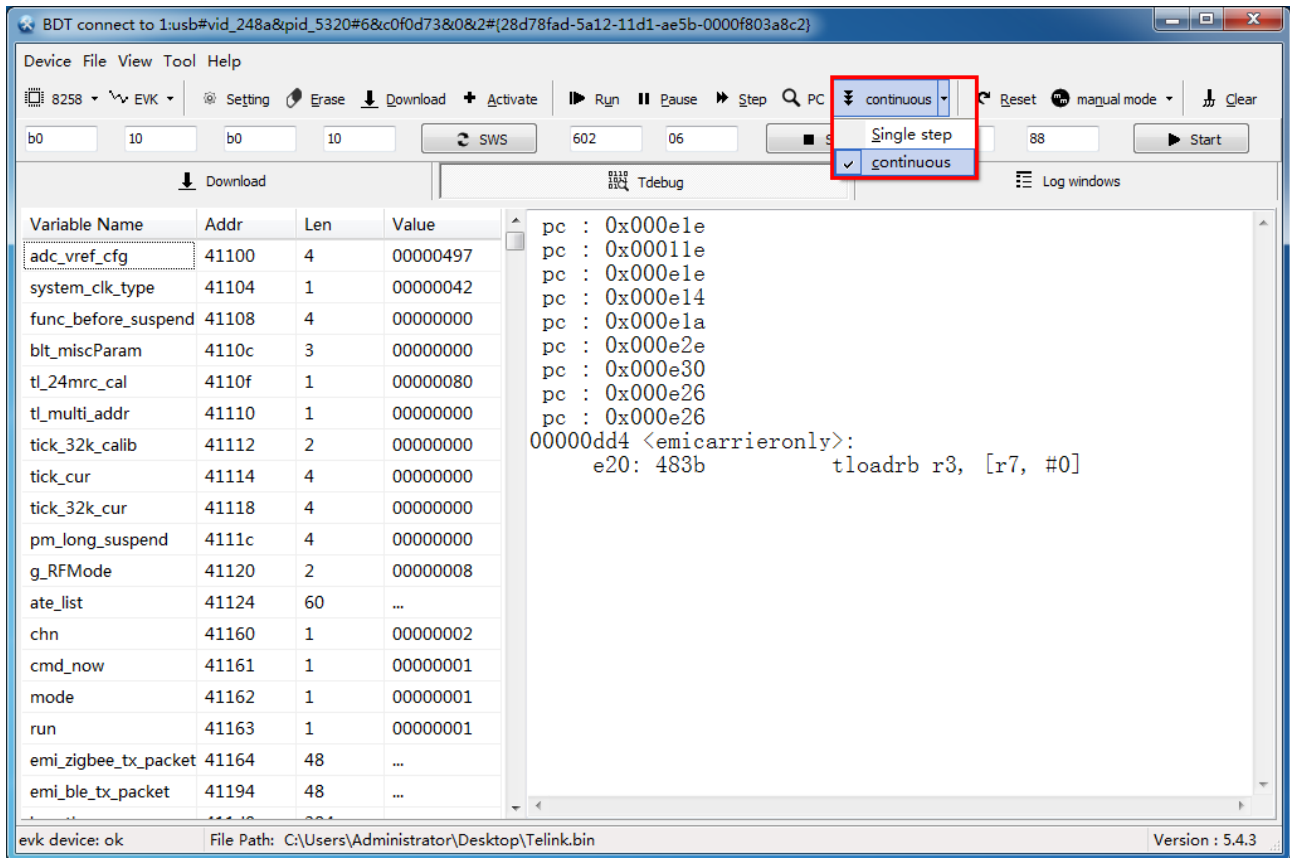


Figure 2.78: 将“Single step”模式切换到“continuous”模式

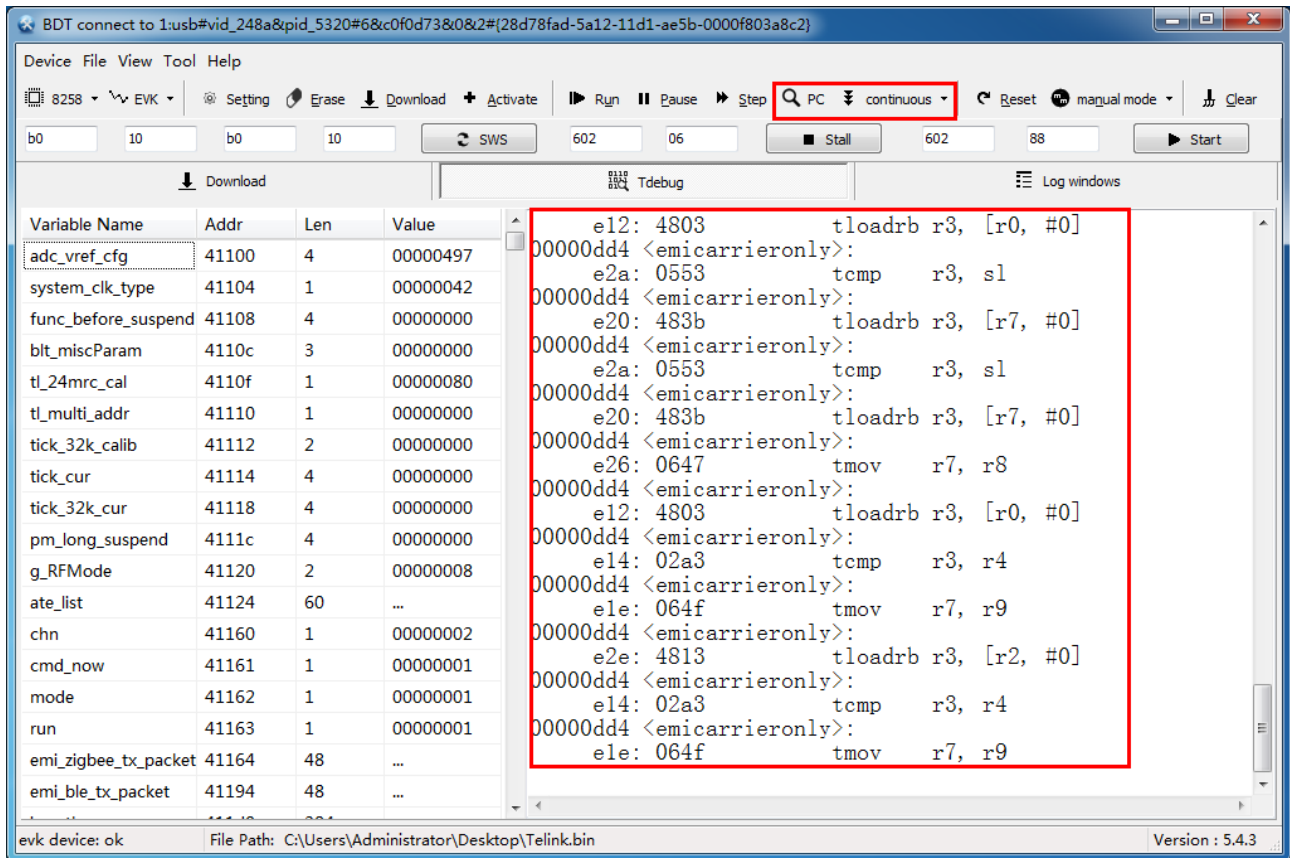


Figure 2.79: 在“continuous”模式下选择“interp.lst” Trace PC

2.6.3.5 Stall MCU

当没有足够的时间查看 MCU 的状态时，用户可以点击“stall”按钮暂停 MCU 并查看 MCU 的状态或更改 MCU 的配置。

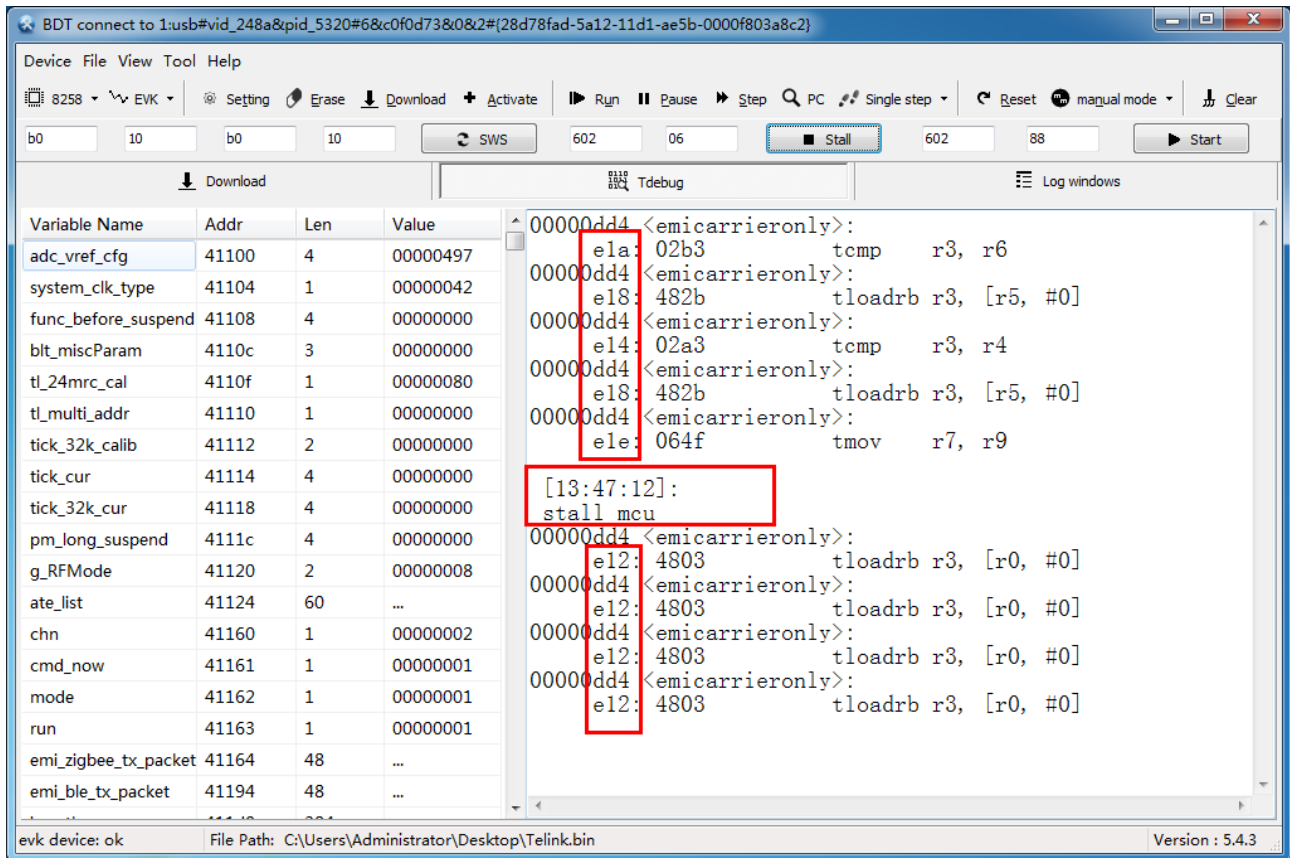


Figure 2.80: Stall MCU

注意:

在 EVK 模式下，只有使用 “Firmware_v3.2.bin”（或更高版本）烧录的 EVK 支持此功能。

2.6.3.6 Start MCU

暂停 MCU 后，点击 “start” 按钮启动 MCU，使 MCU 从 SRAM 开始运行。

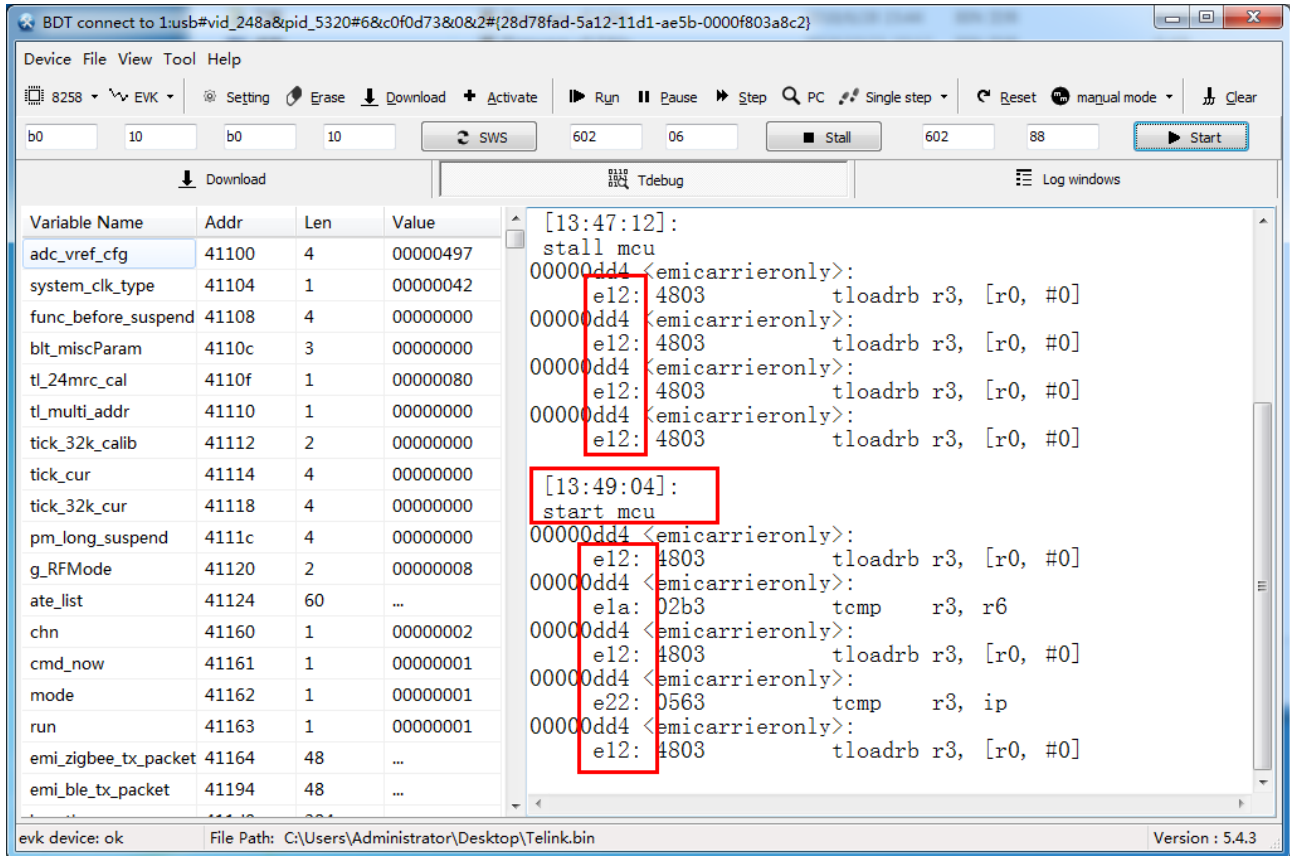


Figure 2.81: Start MCU

2.7 单线同步

在设置单线同步速度之前，请确保以下各项：

- (1) 电源正常；
- (2) MCU 未处于“低功耗”模式；
- (3) MCU 的单线功能可用；
- (4) 系统时钟正常。

当“Burning EVK”和目标板之间无法建立连接时，用户可以尝试设置 Swire（单线）同步速度来改善通信。**注意：**Swire 寄存器地址可能因芯片类型而异。

- (1) 在四个可编辑框中更改主速度和从速度，如下所示。**注意：**Swire 寄存器地址可能因芯片类型而异。
- (2) 点击“SWS”按钮以实现 Swire 同步。

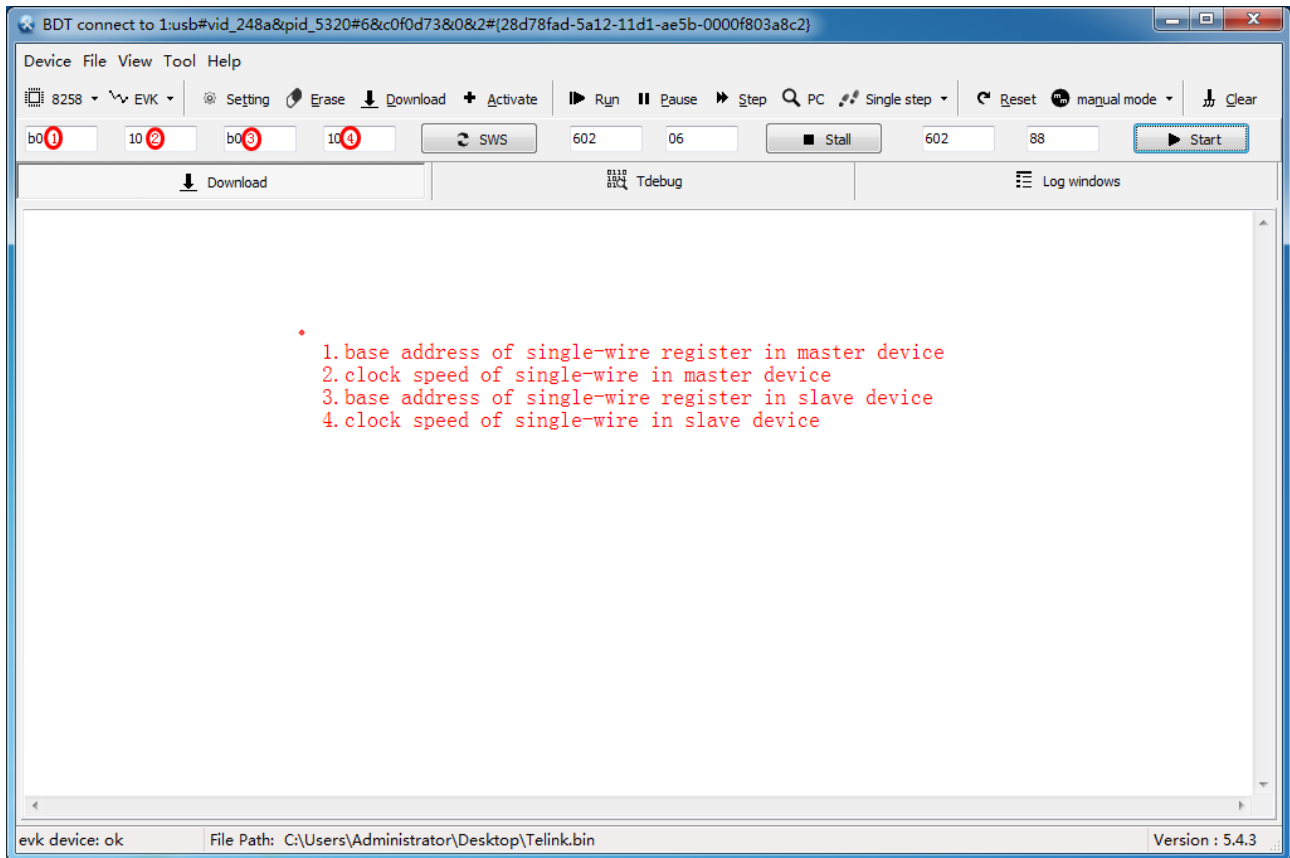


Figure 2.82: 设置 Swire 同步速度

在将固件下载到目标板或调试 MCU 之前，建议每次打开 MCU 电源时，用户应执行一次 Swire 同步，以检查与目标板的通信是否正常。

如果与目标板的通信状态错误，用户可以按照第 2.5 节开头提到的方法解决问题。

2.8 命令行

用户还可以使用命令行访问内存或将文件下载到目标板。在下面的小节中，以“read core”为例。

2.8.1 单命令行

步骤 1：通过选择“Tool->Cmdline Input”或按组合快捷键“Ctrl+I”打开命令行输入窗口。

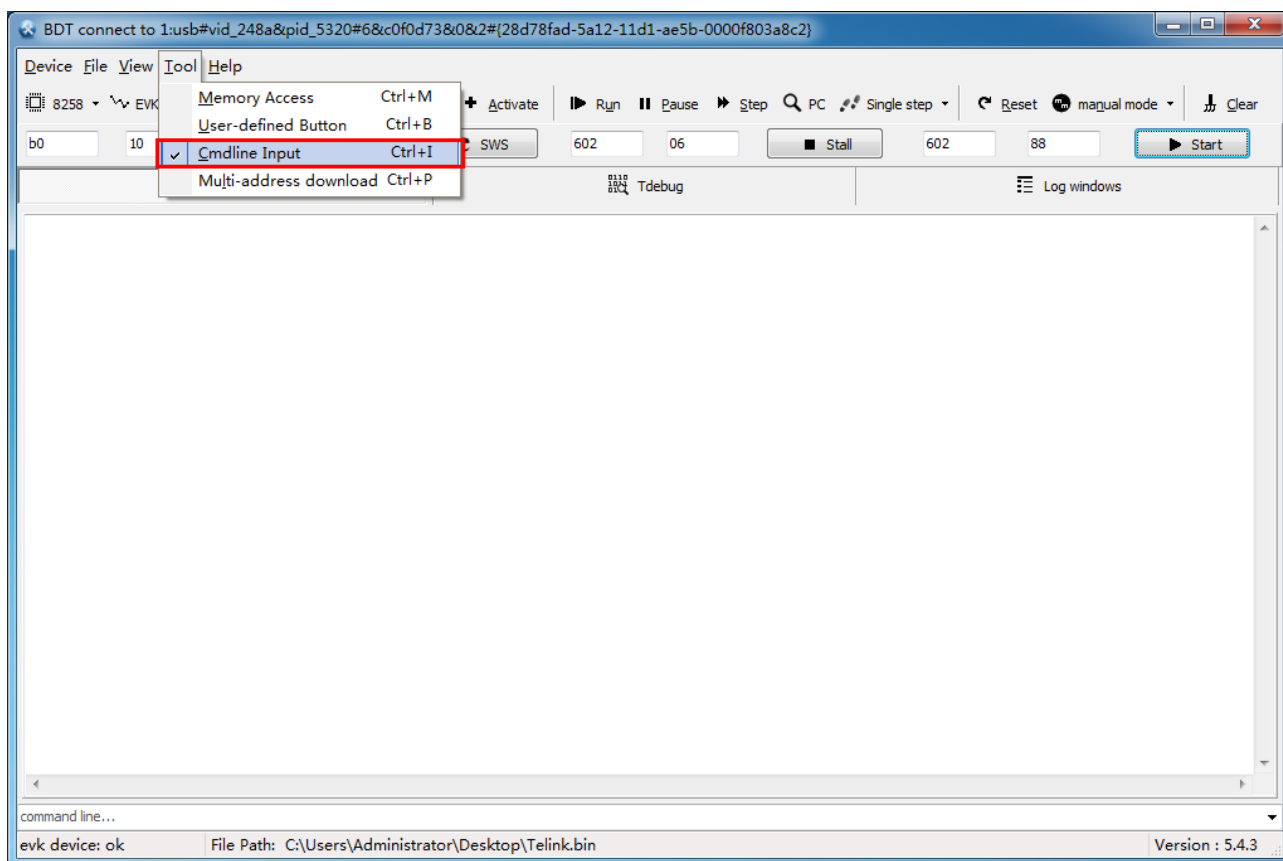


Figure 2.83: 打开命令行输入窗口

步骤 2: 按如下所示输入所需的单个命令，然后按“Enter”键。具有指定名称的文件将在指定位置可用。

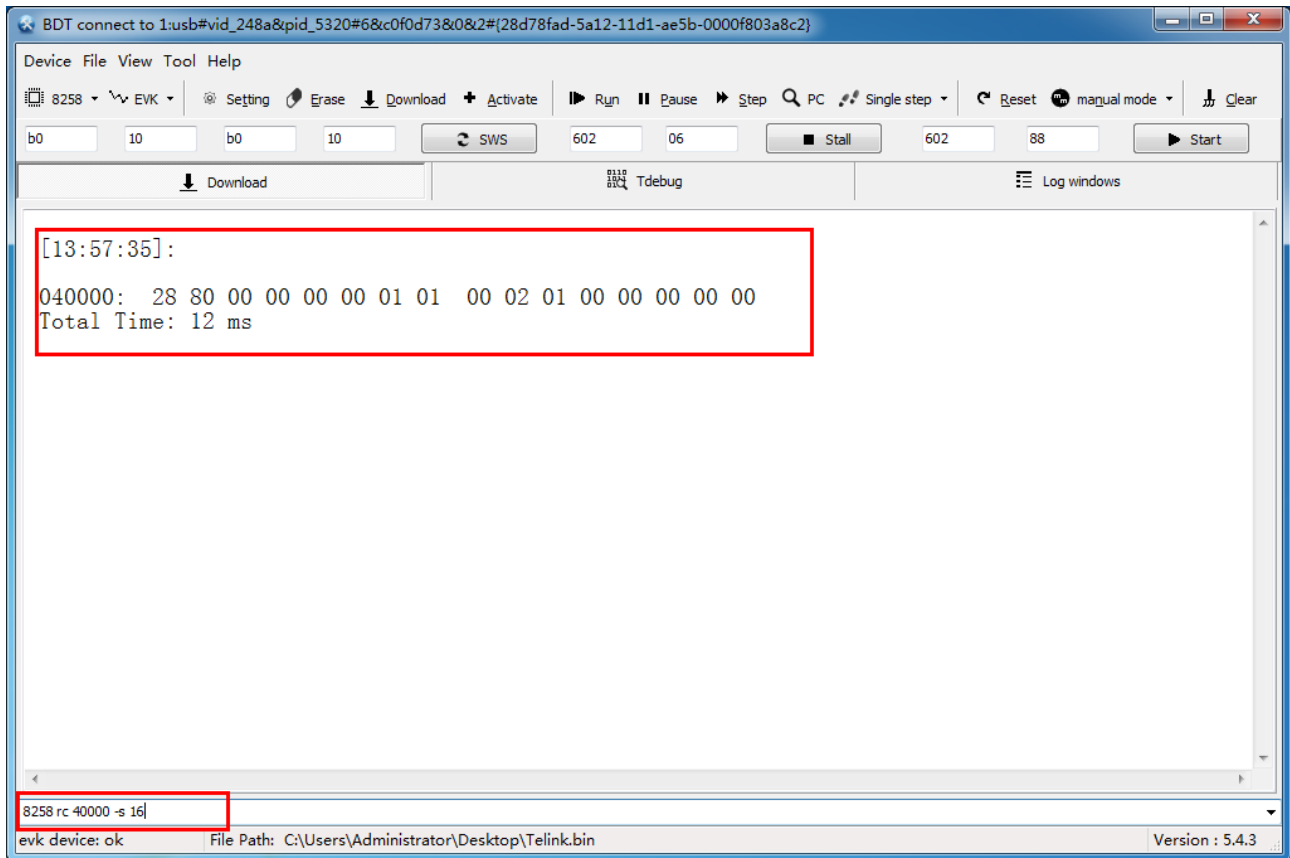


Figure 2.84: 输入单命令行

2.8.2 多命令行

该工具支持多命令行操作。

所有命令都需要用分号隔开，如下所示。

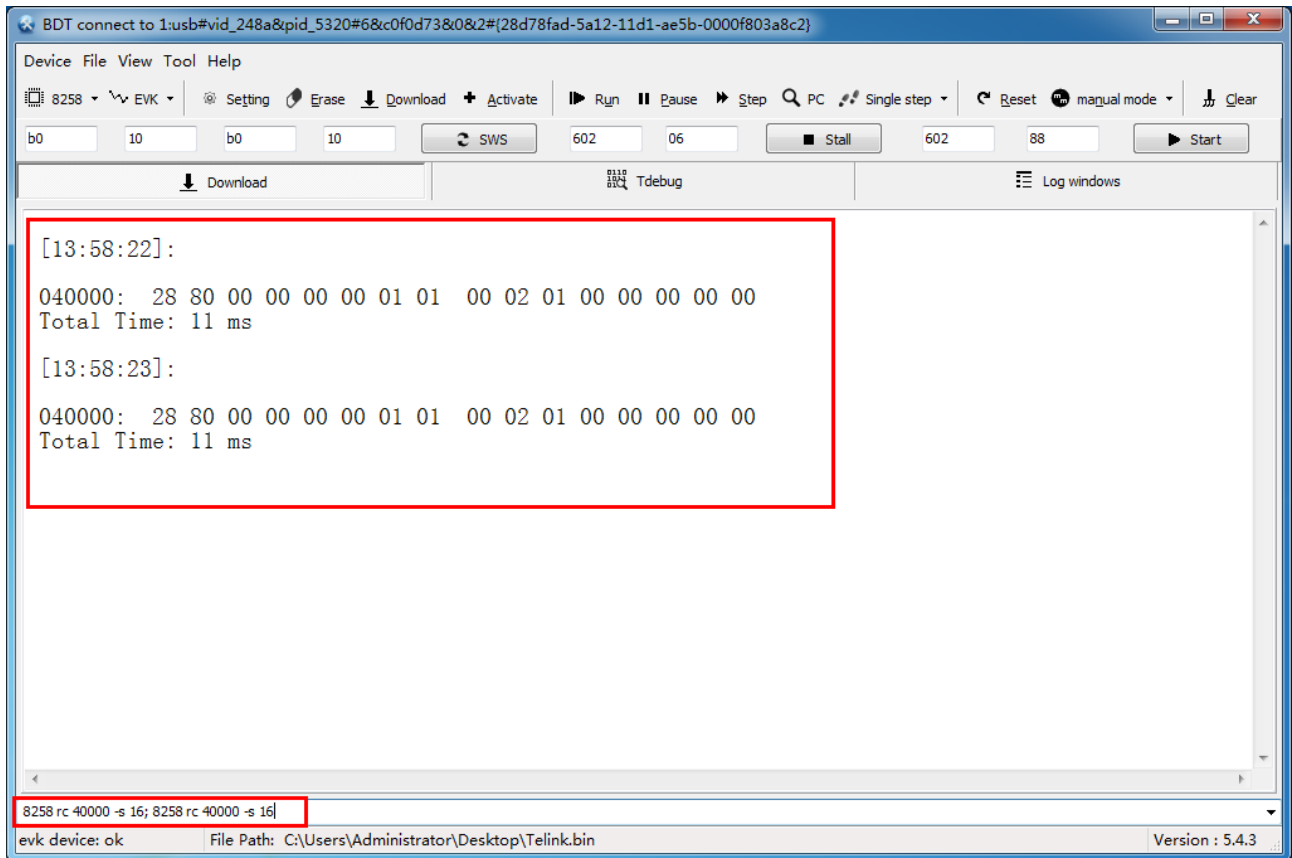


Figure 2.85: 输入多个命令

2.9 日志窗口

该工具支持 USB 打印功能，因此具有 USB 功能的 MCU 可以使用 USB 打印功能，在 PC 端的日志窗口中输出信息。

步骤 1: 点击“View -> usb log”打开 usb 日志窗口。

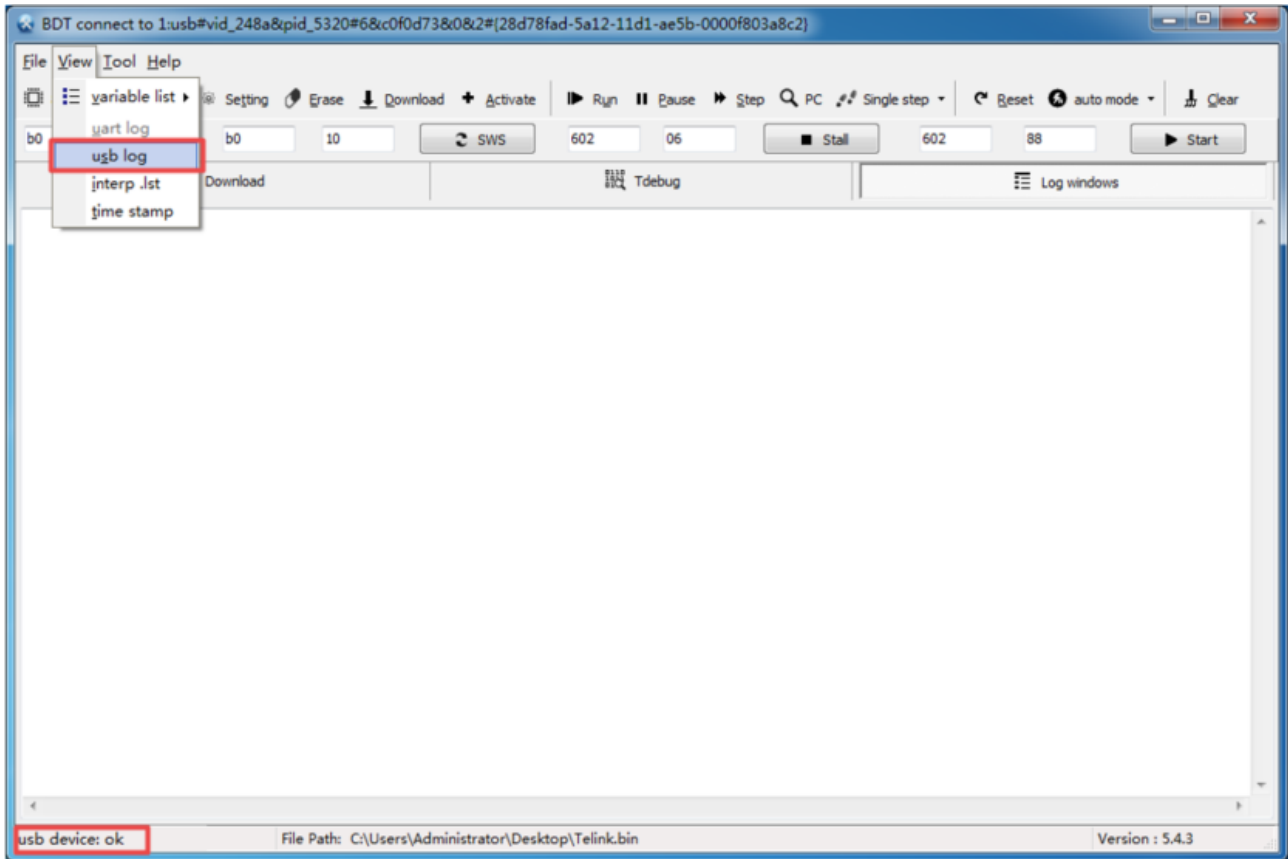


Figure 2.86: 打开 usb 日志窗口

步骤 2: 在 “usb log” 模式下，所有输出的字符将以 “Ascii” 形式显示。

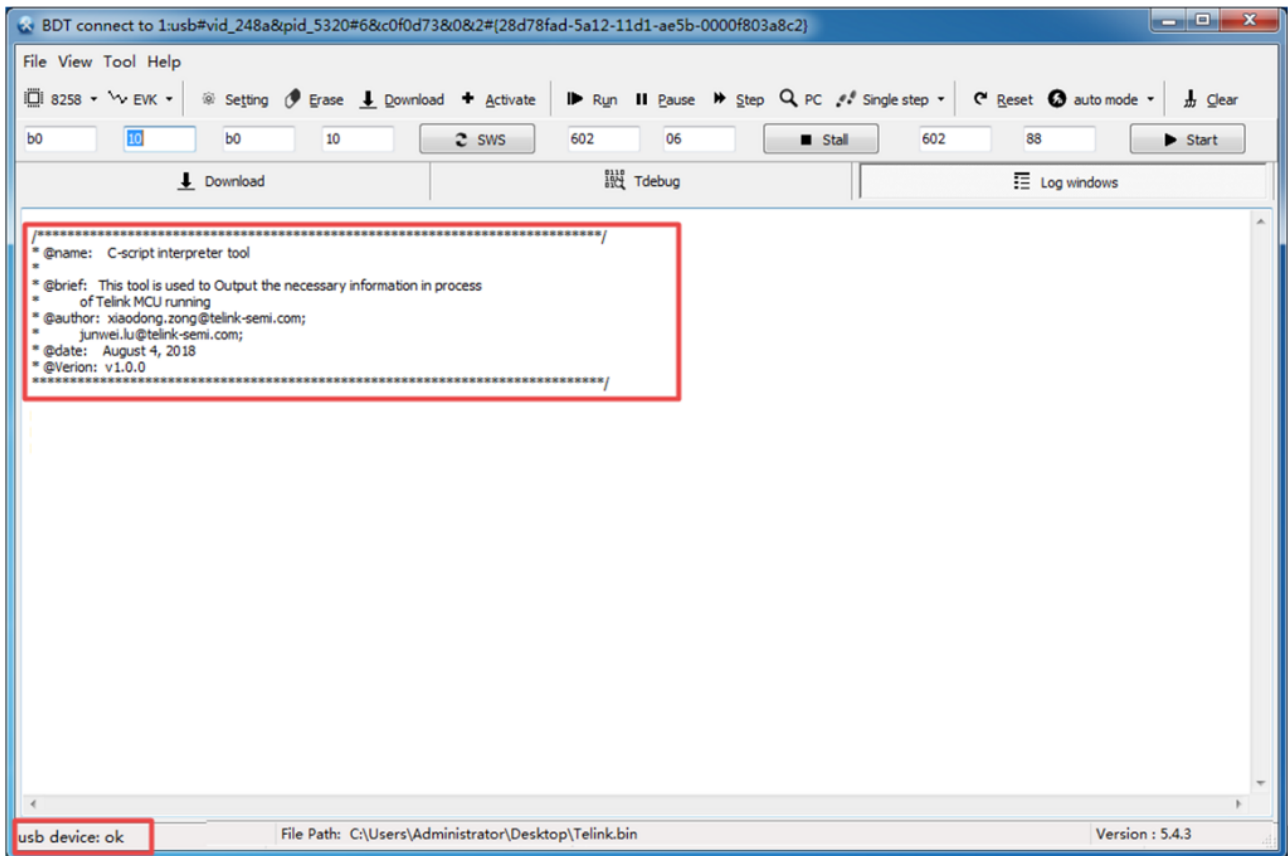


Figure 2.87: 显示输出字符

步骤 3: 再次点击 “View -> usb log” 退出 “usb log” 模式。

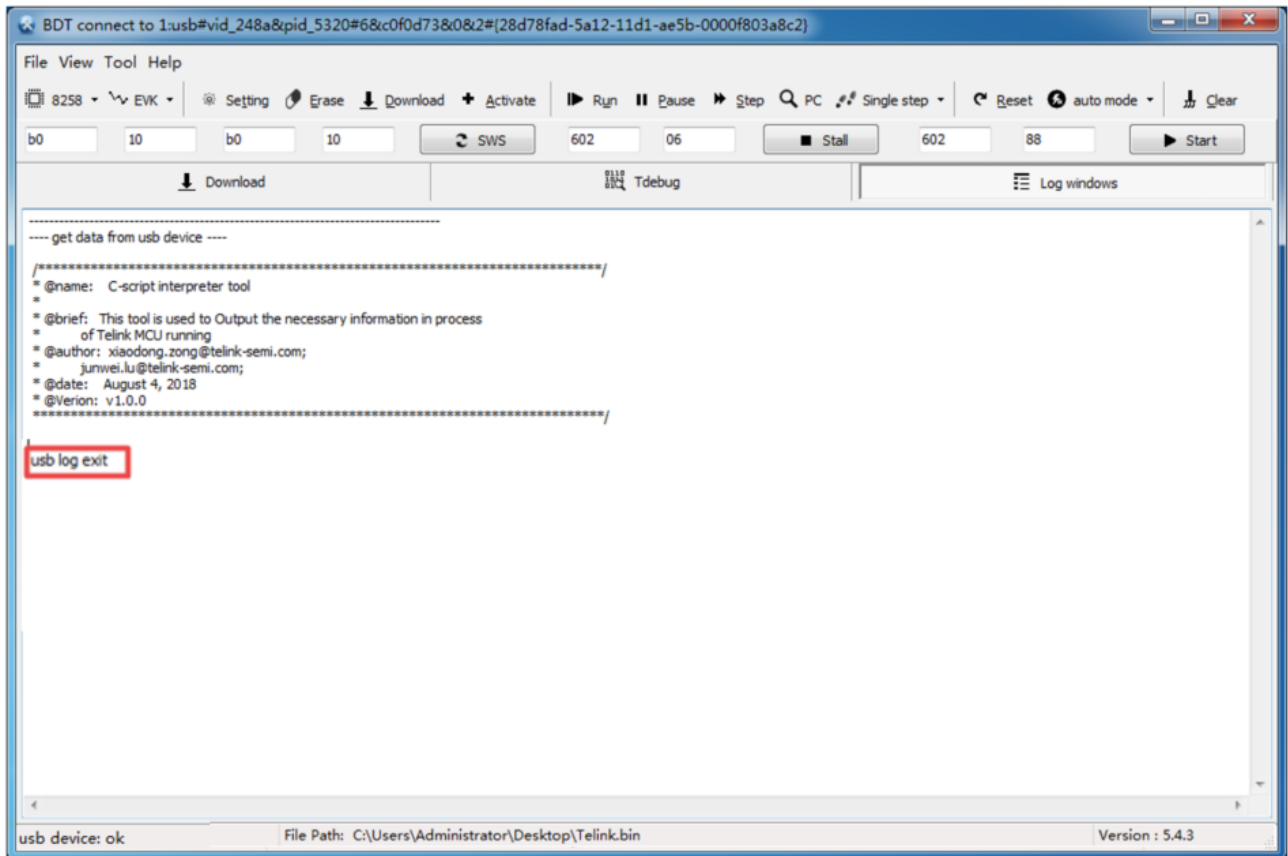


Figure 2.88: 退出“usb log”模式

2.10 帮助

点击“Help -> Command line”，所有命令将显示在“Download”或“Tdebug”页面的日志窗口中。

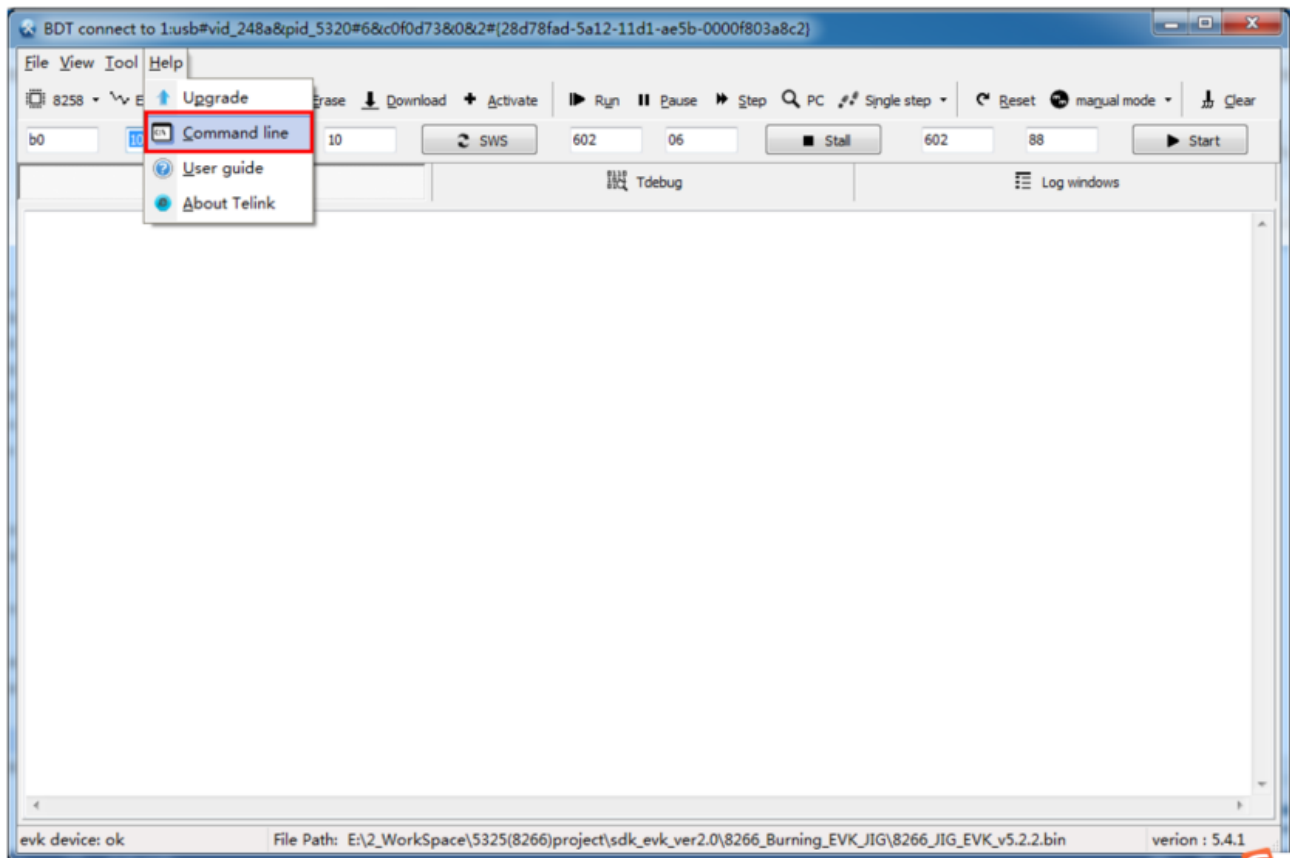


Figure 2.89: 命令行

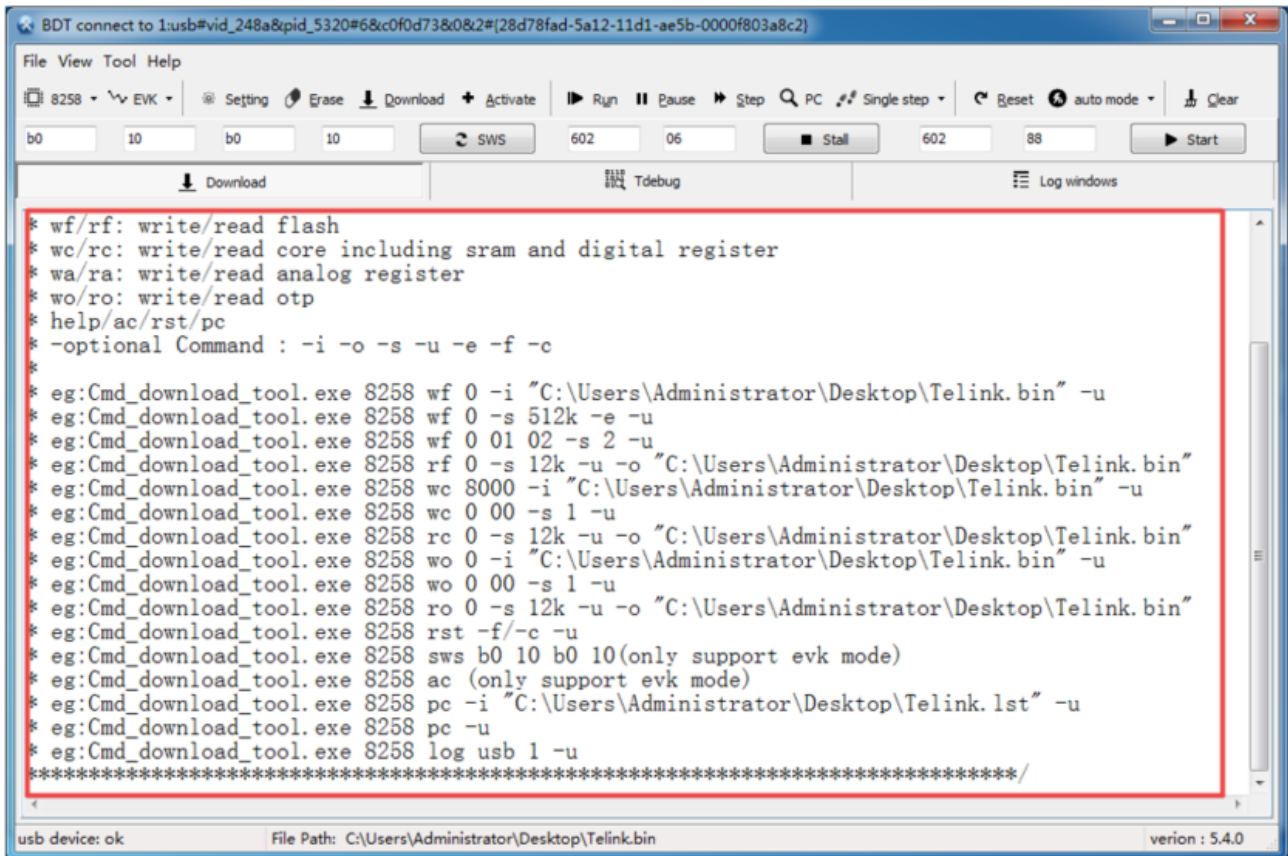


Figure 2.90: 显示所有命令

点击 "Help -> User guide", 将打开嵌入的 "Help" 文档 "Telink Burning and Debugging Tool (BDT) User Guide", 如下所示。

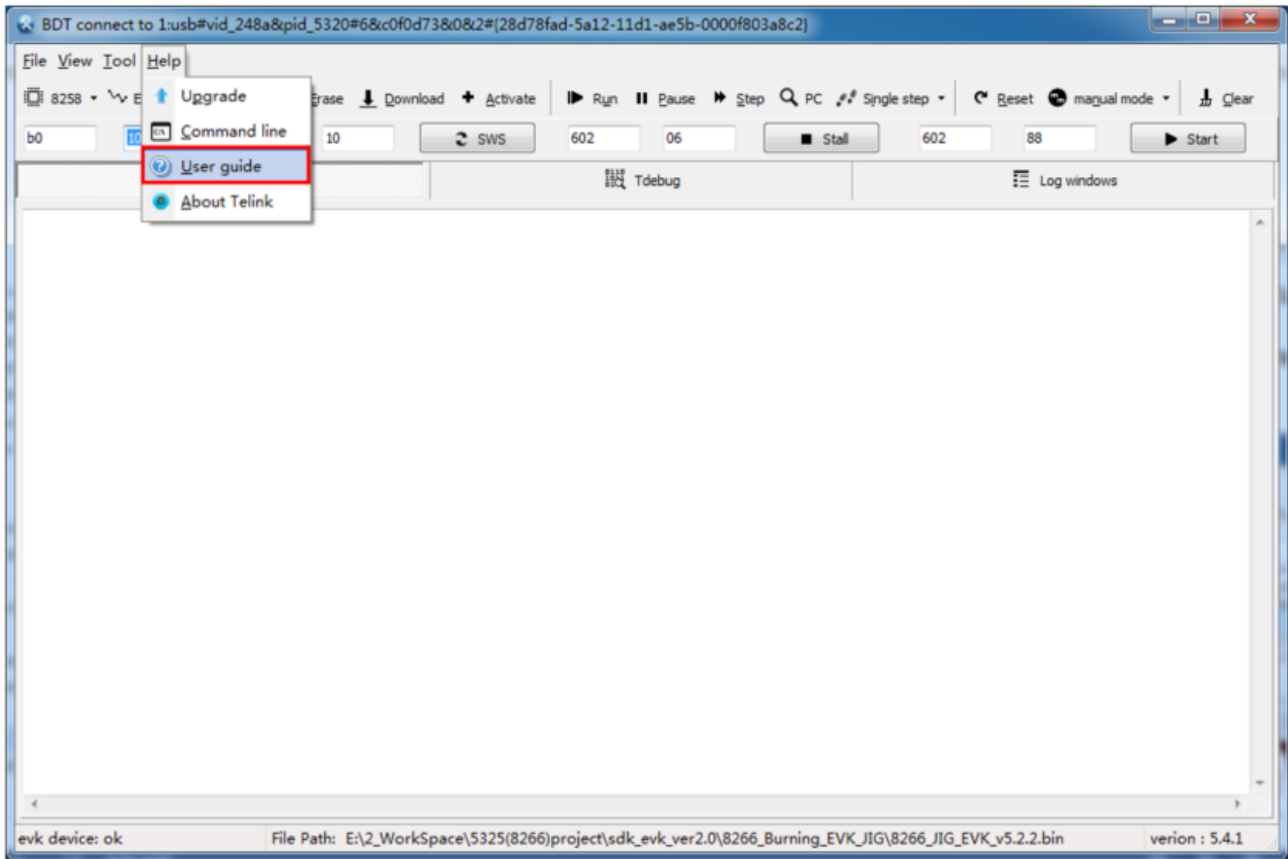


Figure 2.91: 用户指南

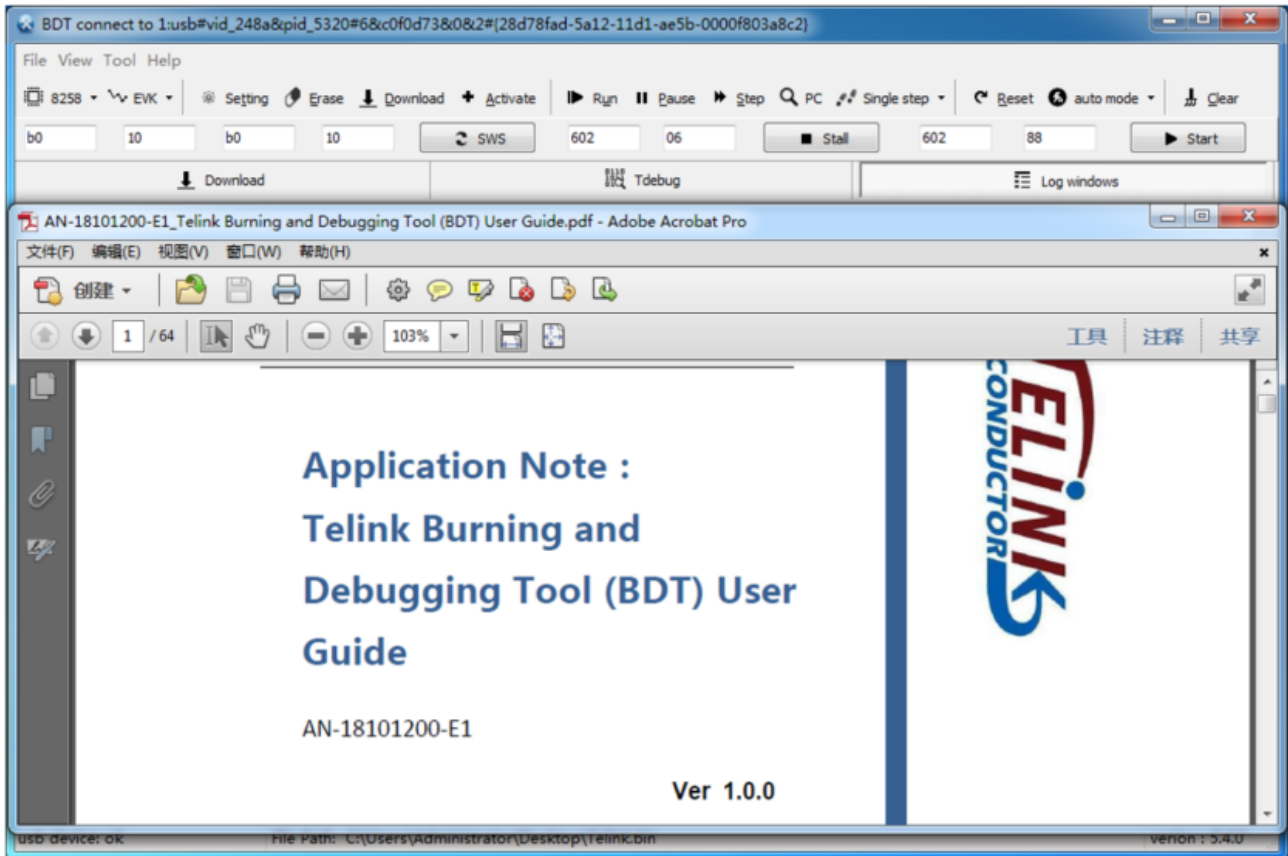


Figure 2.92: 打开帮助文档

要获取有关 Telink 的更多信息，用户可以点击“Help -> About Telink”。

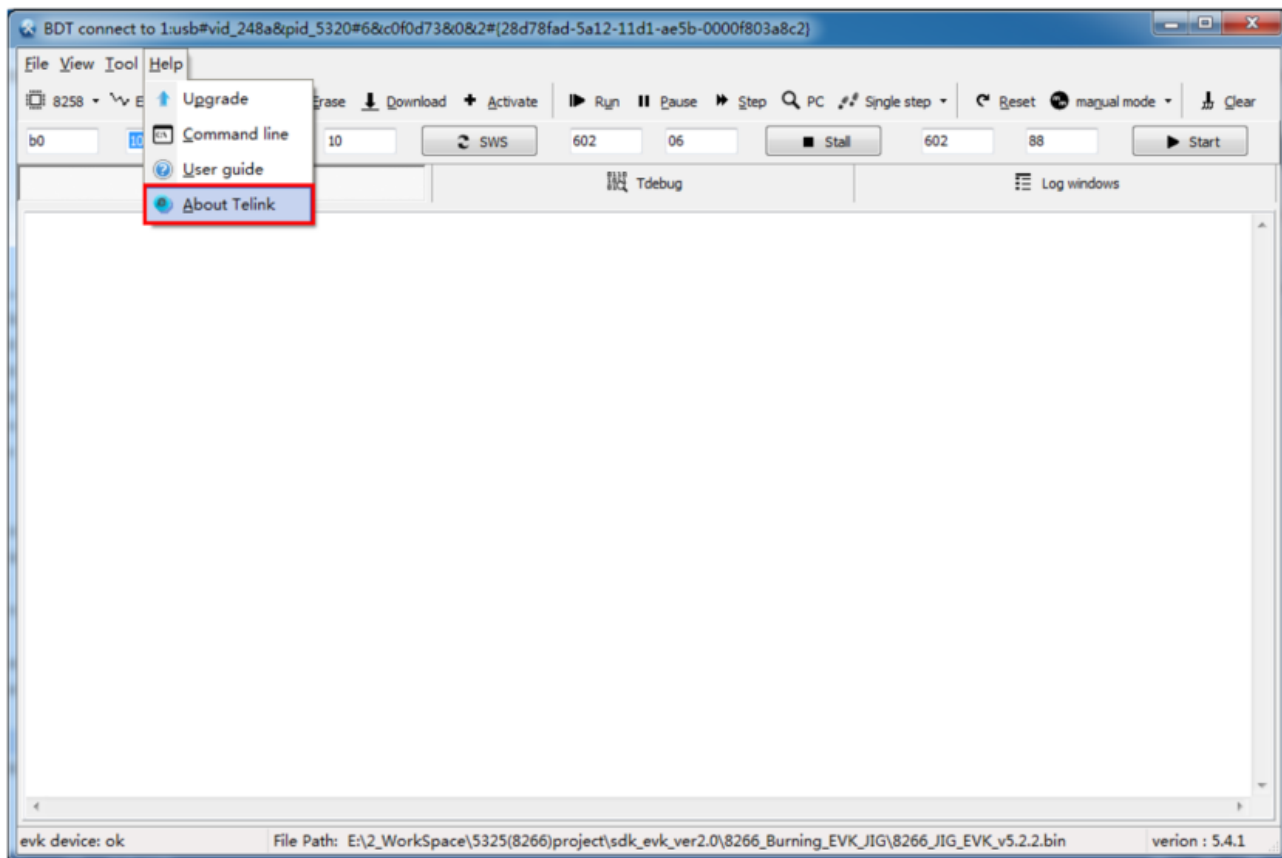


Figure 2.93: About Telink

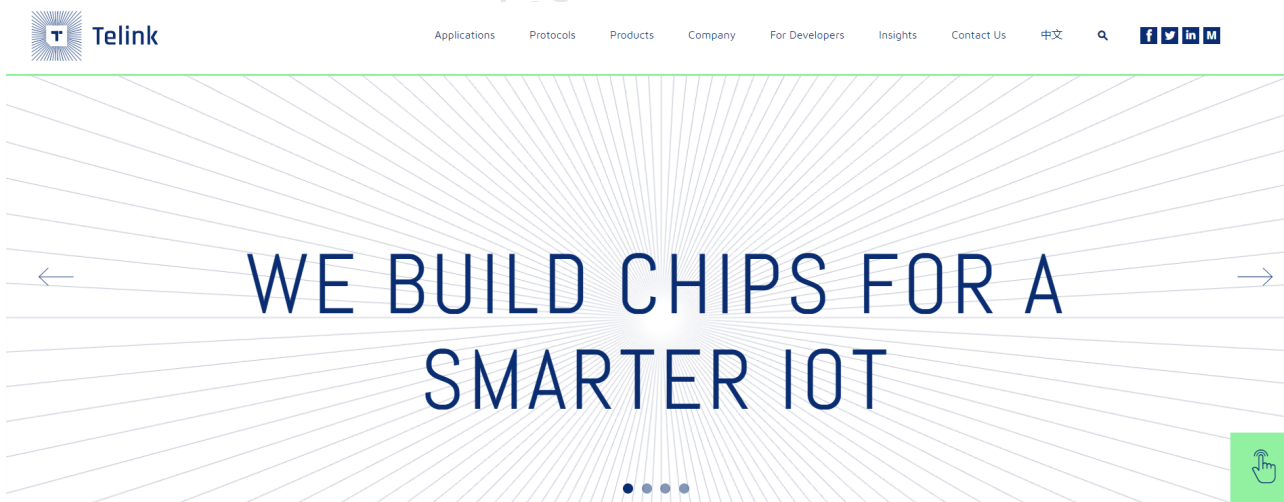


Figure 2.94: 获取有关 Telink 的信息

2.11 默认配置

在这个工具的根目录中打开“config”文件夹，如下所示。

名称	修改日期	类型	大小
bin	2018/10/9 15:17	文件夹	
user	2018/10/9 9:31	文件夹	
Cmd_download_tool.exe	2018/10/10 10:24	应用程序	2,197 KB
config.ini	2018/10/9 18:07	INI 文件	3 KB
tl_usb2.0.dll	2017/12/1 14:02	DLL 文件	100 KB

Figure 2.95: “config”文件夹

通过修改 “config.ini” 文件，可以更改此工具中的所有默认配置。下面是详细说明。

```
[MCU]
TYPE 1 0x01 - 8258 \bin\dut_8368_flash_v0244.bin 8000 b010b010 \bin\dut_8366_otp_v0005.bin \bin\dut_8366_otp_normal_v0002.bin End
TYPE 1 0x02 - 8258 \bin\dut_8368_flash_v0246.bin 8000 b010b010 \bin\dut_8368_otp_v0005.bin \bin\dut_8368_otp_normal_v0002.bin End
TYPE 2 0x03 - 8232 \bin\dut_8232_flash_v0002.bin 8000 b010b010 End
TYPE 2 0x04 - 8233 \bin\dut_8233_flash_v0002.bin 8000 b010b010 End
TYPE 2 0x05 - 8266 \bin\dut_8266_flash_v0244.bin 8000 b010b010 End
TYPE 2 0x06 - 8267 \bin\dut_8267_flash_v0002.bin 8000 b010b010 End
TYPE 3 0x07 - 8255 \bin\dut_8255_flash_v0002.bin 40000 b010b010 End
TYPE 3 0x08 - 8258 \bin\dut_8258_flash_v0003.bin 40000 b010b010 End
def 8258 End
```

Figure 2.96: MCU

(1) MCU 类型：8368

(2) 固件路径：\bin\dut_8368_flash_v0246.bin

该固件用于操作 flash，例如：将指定固件下载到 flash、读/写 flash、擦除 flash 扇区，并在下载固件到 flash 或擦除 flash 的某些扇区时检查 flash 数据。

(3) SRAM 起始地址：0x8000, 例如 8258: 0x40000;

(4) 单线通信：

- 主设备的 Swire 寄存器基址：0xb0
- 主设备的 Swire 时钟：0x10
- 从设备的 Swire 寄存器基址：0xb0
- 从设备的 Swire 时钟：0x10

(5) 固件路径：

固件 1: \bin\dut_8368_otp_v0005.bin，用于操作 OTP，例如将固件下载到 OTP，读取/写入 OTP，在将数据写入 OTP 或将固件下载到 OTP 期间检查 OTP 数据。

固件 2: \bin\dut_8368_otp_normal_v0002.bin，用于将数据写入 OTP 或将固件下载到 OTP 期间检查 OTP 数据。

(6) 默认：8258

```
[MODE]
MODE 0x01 - 0x01 End
MODE 0x04 - 0x04 End
default - 0x01 End
```

Figure 2.97: 通信方式

- (1) 通信方式: USB
- (2) 默认: EVK (Burning EVK)

```
[OBJ]
OBJ 0x01 = FLASH End
OBJ 0x02 = CORE End
OBJ 0x03 = ANALOG End
OBJ 0x04 = GPP End
default = CORE End
```

Figure 2.98: Objection

- (1) 访问 Objection: Flash
- (2) 默认: Core

```
[SIZE]
SIZE 0x01 = 1 End
SIZE 0x02 = 1 End
SIZE 0x03 = 1 End
SIZE 0x04 = 8 End
SIZE 0x05 = 16 End
default = 2 End
```

Figure 2.99: 大小

- (1) 读/写数据的大小: 2
- (2) 默认: 1

```
[DOWNLOAD_ADDR]
DOWNLOAD 0x01 = 0 End
DOWNLOAD 0x02 = 20000 End
DOWNLOAD 0x03 = 40000 End
default = 2 End
```

Figure 2.100: 下载地址

- (1) 下载地址: 0x20000
- (2) 默认: 0

```
[ERASE_ADDR]
ERASE 0x01 = 0 End
ERASE 0x02 = 20000 End
ERASE 0x03 = 40000 End
default = 2 End
```

Figure 2.101: 擦除地址

- (1) 擦除地址: 0x20000
- (2) 默认: 0

```
[PAGE_DISPLAY]
PAGE 0x01 = 1 Download End
PAGE 0x02 = 1 Debug End
PAGE 0x03 = 2 Log windows End
default = 2 End
```

Figure 2.102: 页面显示

- (1) 页面显示：1 - Tdebug
- (2) 默认：0 - Download

```
[BUTTON]
Button 1 = example1 external_exe E:\1_wtcd\tcdb.exe uart -1 -u End
Button 2 = example2 internal_exe 8266 wf 34000 -e 48k ;8266 rc 00 -s 10 -u; End
Button 3 = example3 external_file E:\1_wtcd\prj\kite\doc\TS5565_AFEpin_description_v1.0.xlsx End
/But4 End
/But5 End
/But6 End
/But7 End
/But8 End
/But9 End
/But10 End
/But11 End
/But12 End
/But13 End
/But14 End
```

Figure 2.103: 按钮

- (1) 按钮 1 的名字：example1
- (2) 按钮 1 的属性：external executable application
- (3) 外部可执行应用程序的路径：E:\1_wtcd\tcdb.exe uart -1 -u
- (4) 未定义按钮：Button4

按钮的属性包括三种类型：

- a. external_exe：外部可执行应用程序
- b. internal_exe：内部可执行应用程序，也就是这个工具。
- c. external_file：默认情况下打开任意外部文件。

```
[PATH]
PATH 0x01 = C:\Users\Administrator\Desktop\Telink.bin End
default = 0x2 End
```

Figure 2.104: 路径

- (1) 固件路径：“C:\Users\Administrator\Desktop\Telink.bin”

将此固件下载到目标板以更新固件，当文件路径过多时，可以直接删除这些路径。但是，无法删除第一条路径。

- (2) 默认文件路径：“C:\Users\Administrator\Desktop\Telink.bin”

3 升级固件

用户可以按照以下步骤升级 “Burning EVK” 的固件。

步骤 1： 点击 “Help -> Upgrade” 打开 “Upgrade EVK” 对话框。

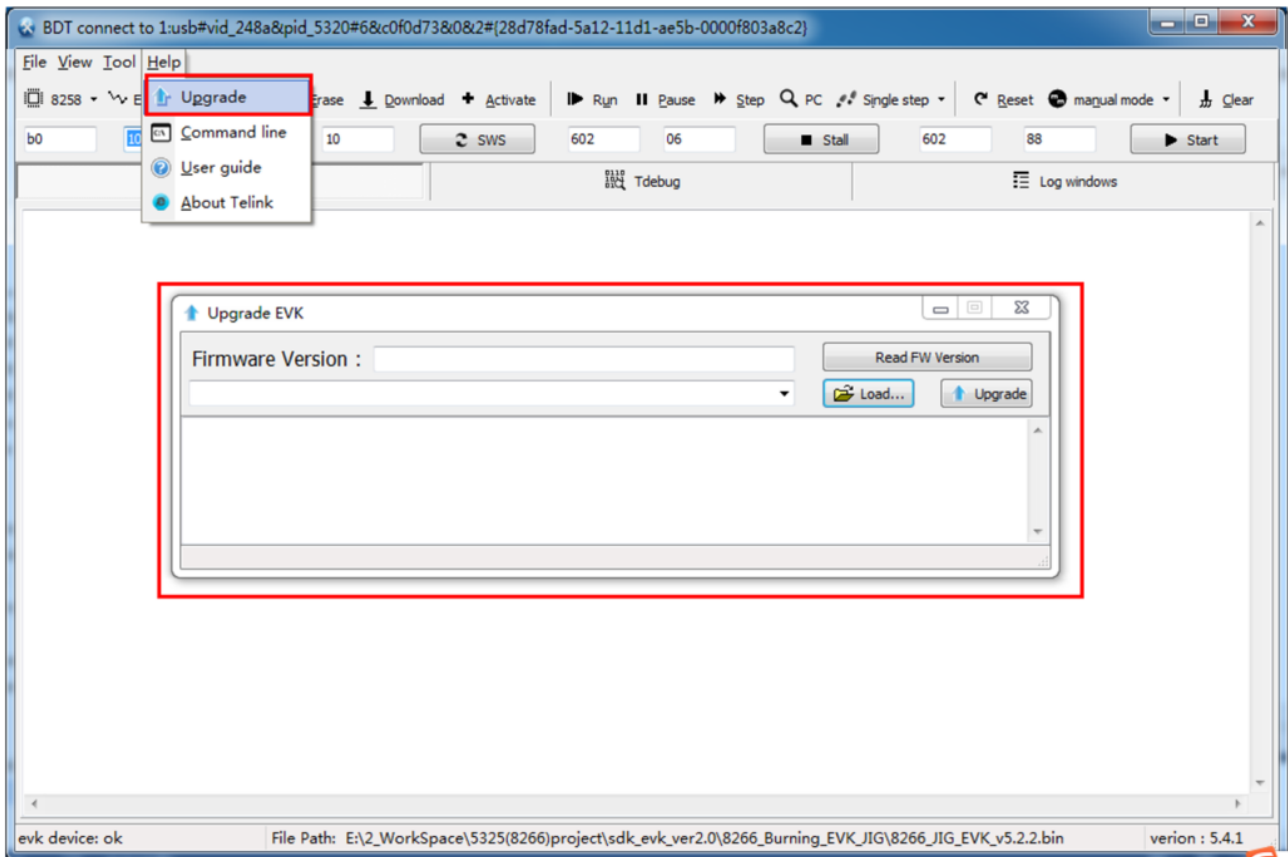


Figure 3.1: “Upgrade EVK” 窗口

步骤 2： 点击 “Read FW Version” 按钮检查当前固件版本。

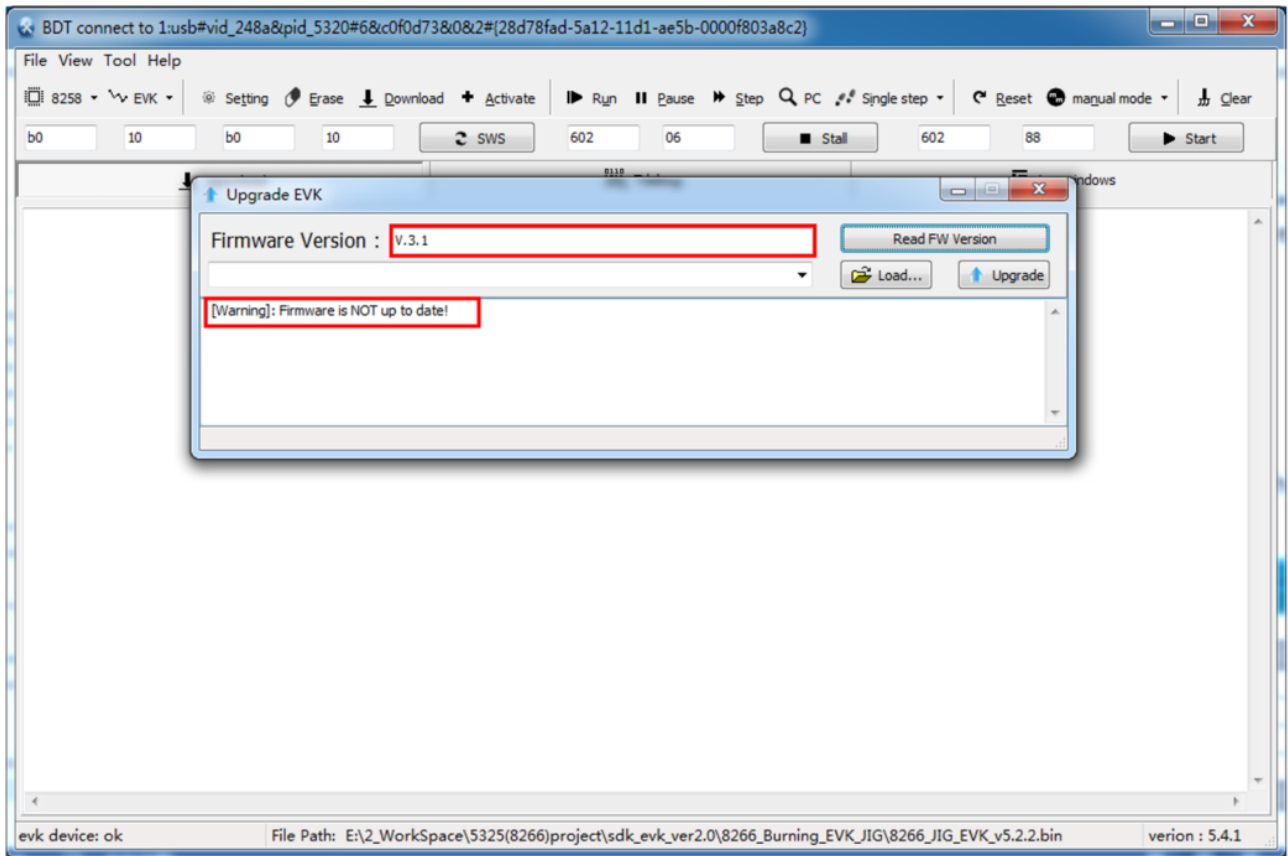


Figure 3.2: 检查当前固件版本

步骤 3: 单击 “Load...” 按钮加载新固件。

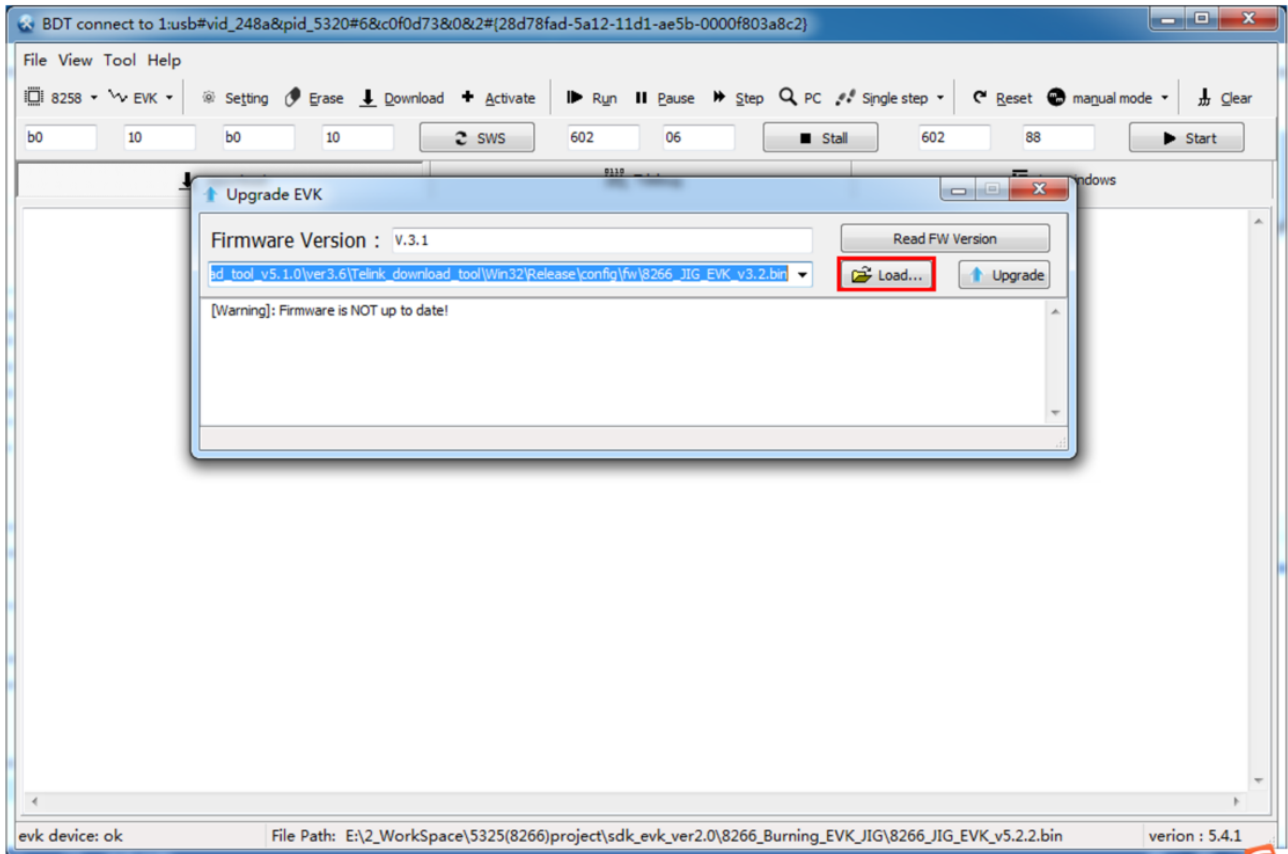


Figure 3.3: 加载新固件

步骤 4: 点击 “Upgrade” 按钮将新固件升级到 “Burning EVK” 中。注意：用户必须在使用 EVK 之前对它进行重启。

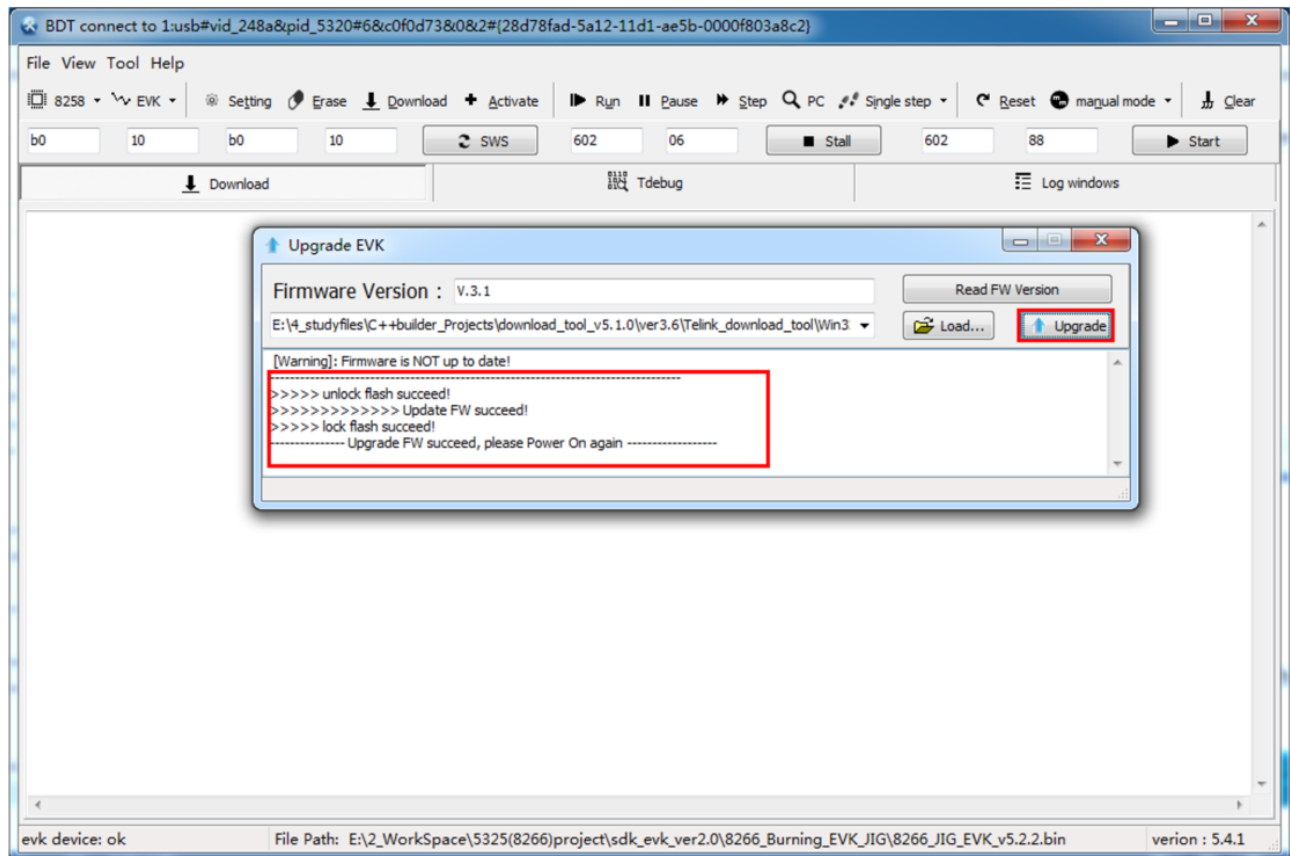


Figure 3.4: 将固件升级到 EVK 中

4 工具指南

本章节主要介绍 BDT 的内置工具，内容包括工具的功能及其使用方法。

4.1 BQB Tool

4.1.1 功能描述（BQB Tool）

该工具有两个主要的功能：

- 根据需求配置生成用于 BQB 测试的固件（Flash 和 SRAM 两种类型的 bin 文件），其中可配置的选项包括：UART 引脚、PA 控制引脚、TX 能量、内外部电容、内部供电方式以及校准参数的位置。另外，生成的固件可以通过工具直接下载到 DUT 中（通过 Swire 的方式）。**注意：**上述 BQB 测试固件为 uart 两线测试程序，串口通信的波特率为 115200。
- 收发包测试

4.1.2 界面组成及使用说明（BQB Tool）

该工具主要由固件下载界面和测试界面两个部分组成，具体如下图所示。

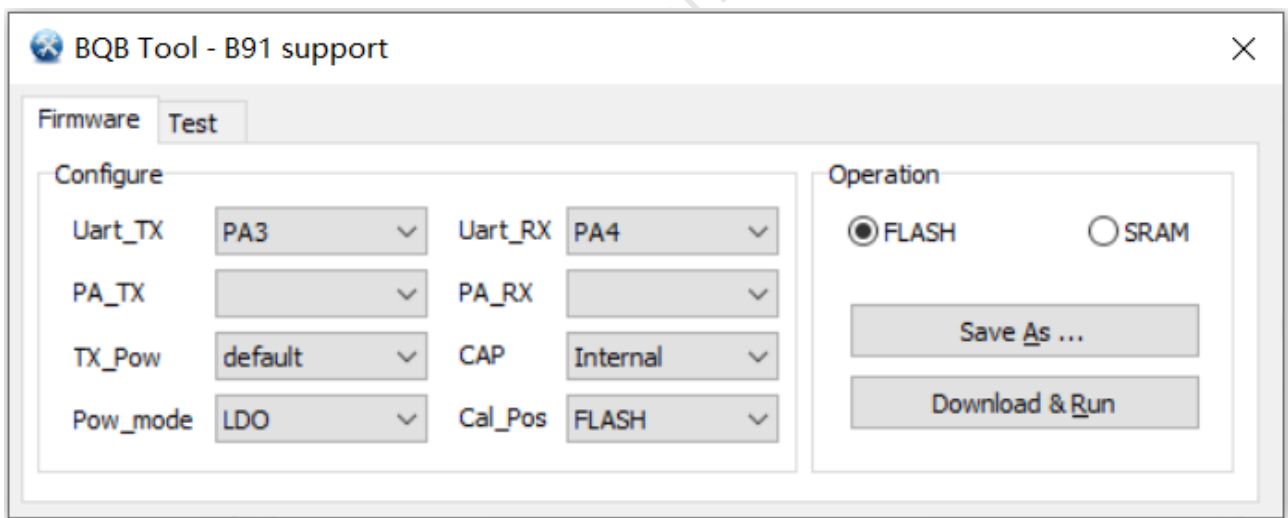


Figure 4.1: BQB 工具固件下载界面

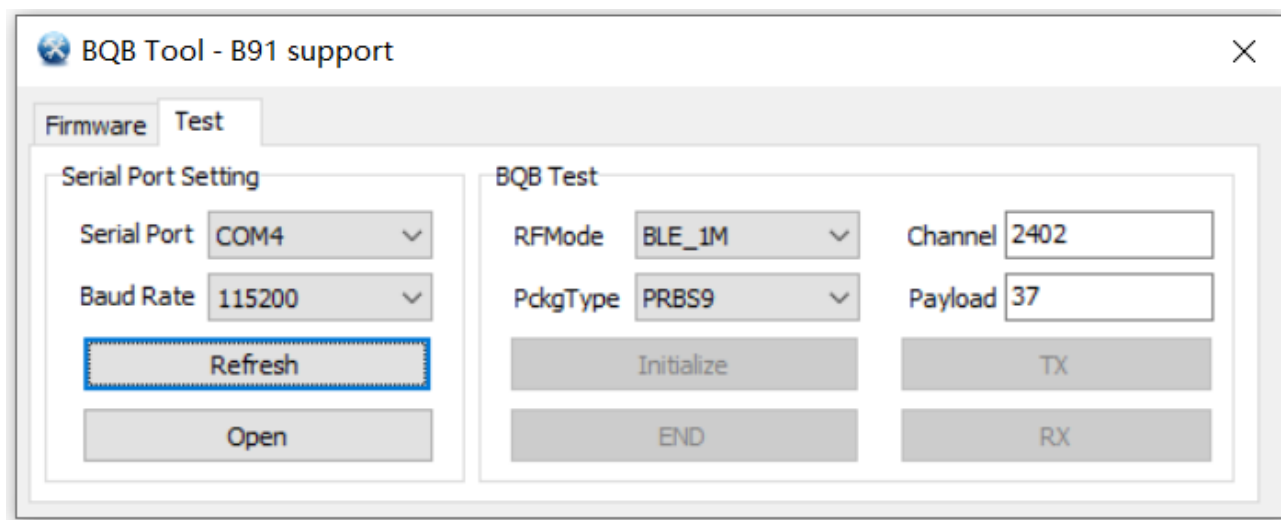


Figure 4.2: BQB 工具测试界面

4.1.2.1 芯片的选择

通过 BDT 主界面 MCU 列表来进行芯片型号的选择，选择好芯片之后可通过 BQB Tool 界面的标题栏来判断工具是否支持对应芯片。

4.1.2.2 固件的保存和下载

Step1 - 配置固件：固件下载界面“Configure”区域，Uart_TX(RX) 用来配置 UART 的 TX 和 RX 引脚，PA_TX(RX) 用来配置 PA 的 TX 和 RX 引脚，TX_Pow 用来配置发包的能量（default 为默认值，一般为 7dBm），CAP 用来配置内外部电容，Pow_mode 用来配置芯片内部的供电方式，Cal_Pos 用来配置校准数据的存储位置（根据实际情况选择，测试时可选择 SRAM）。

Step2 - 选择固件类型：固件下载界面“Operation”区域，通过单选框来选择类型，可供选择的类型包括 Flash 程序和 SRAM 程序两种。

Step3 - 保存/下载固件：固件下载界面“Operation”区域，通过点击“Save As...”按钮来保存固件，通过点击“Download & Run”按钮来下载固件。

4.1.2.3 收发包测试

Step1 - 连接串口：测试界面“Serial Port Setting”区域，“SerialPort”为可选择的串口（若列表为空，可通过“Refresh”按钮来更新），“BaudRate”为通信的波特率（默认选用 115200，与固件的串口通信速率保持一致），串口配置好之后点击“Open”按钮打开串口。

Step2 - 配置测试模式：测试界面“BQB Test”区域，RFMode 用来选择 RF 模式，Channel 用来配置测试频点，PckgType 用来选择包的数据类型，Payload 用来配置包的数据长度。

Step3 - 测试：测试界面“BQB Test”区域，“Initialize”按钮用来初始化 DUT 的状态，“TX”按钮用来控制 DUT 发包，“RX”按钮用来控制 DUT 收包，“END”按钮用来控制 DUT 停止收包或者发包（收包状态下点击 END 按钮可统计收包的个数）。

4.2 Jtag Tool

4.2.1 功能描述（Jtag Tool）

该工具的主要功能如下：

- 安装 Jtag 硬件相关的驱动
- 通过 Jtag 下载/擦除固件
- 通过 Jtag 读写参数
- 通过 Jtag 调试

4.2.2 界面组成及使用说明（Jtag Tool）

Jtag 工具的界面由两部分组成，分别是“Target”区域和“Debug”区域，具体如下图所示。其中“Target”区域负责设备的连接以及 Jtag 设备驱动的安装，“Debug”区域主要负责读、写、擦以及调试功能。

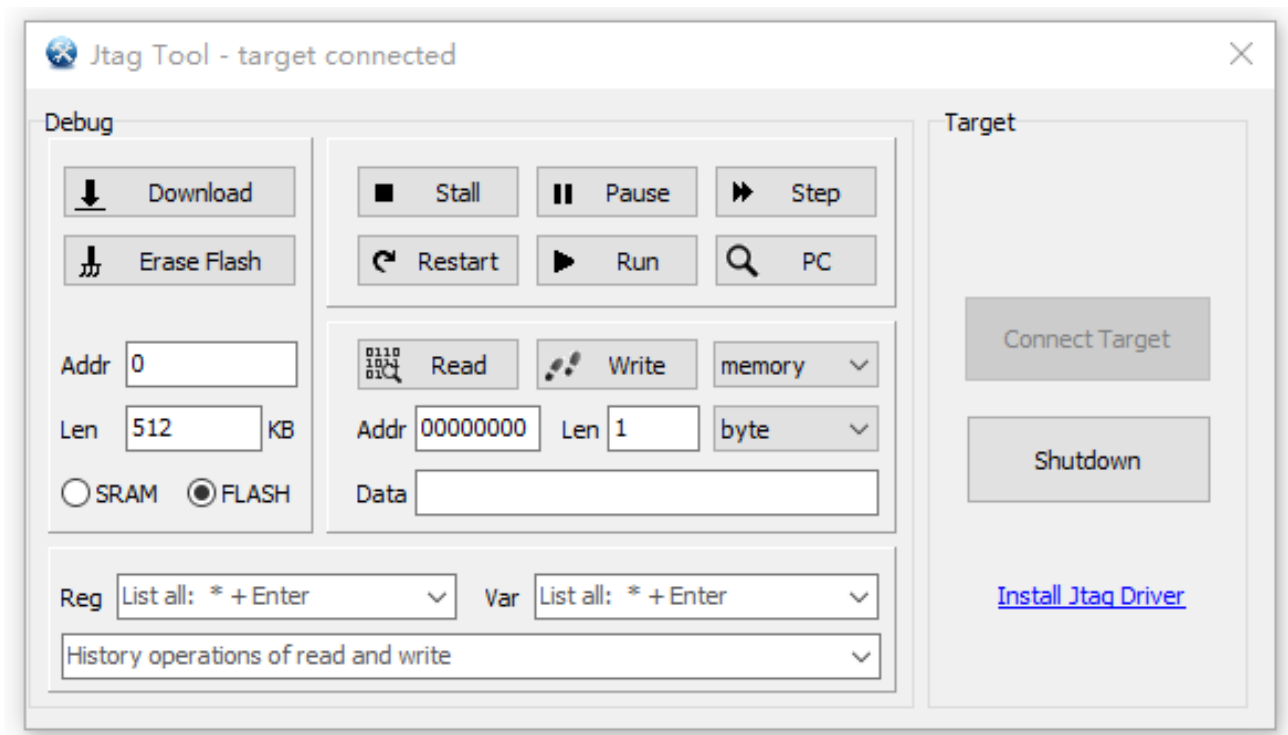


Figure 4.3: Jtag 工具界面

4.2.2.1 Jtag 设备的驱动安装

Step1 - 接入 Jtag 设备： Jtag 设备如下图所示，通过 USB 与 PC 连接。



Figure 4.4: Jtag 设备

Step2 - 检查设备驱动：打开设备管理器查看设备，若驱动已安装，则会被正常识别成“libusbK USB Devices”设备，如下图所示。

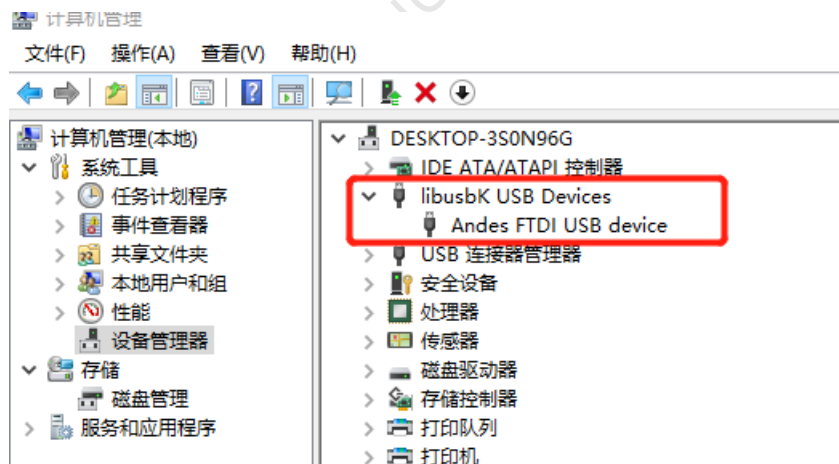


Figure 4.5: Jtag 设备驱动已安装

若驱动未正常安装，则会如下图所示。

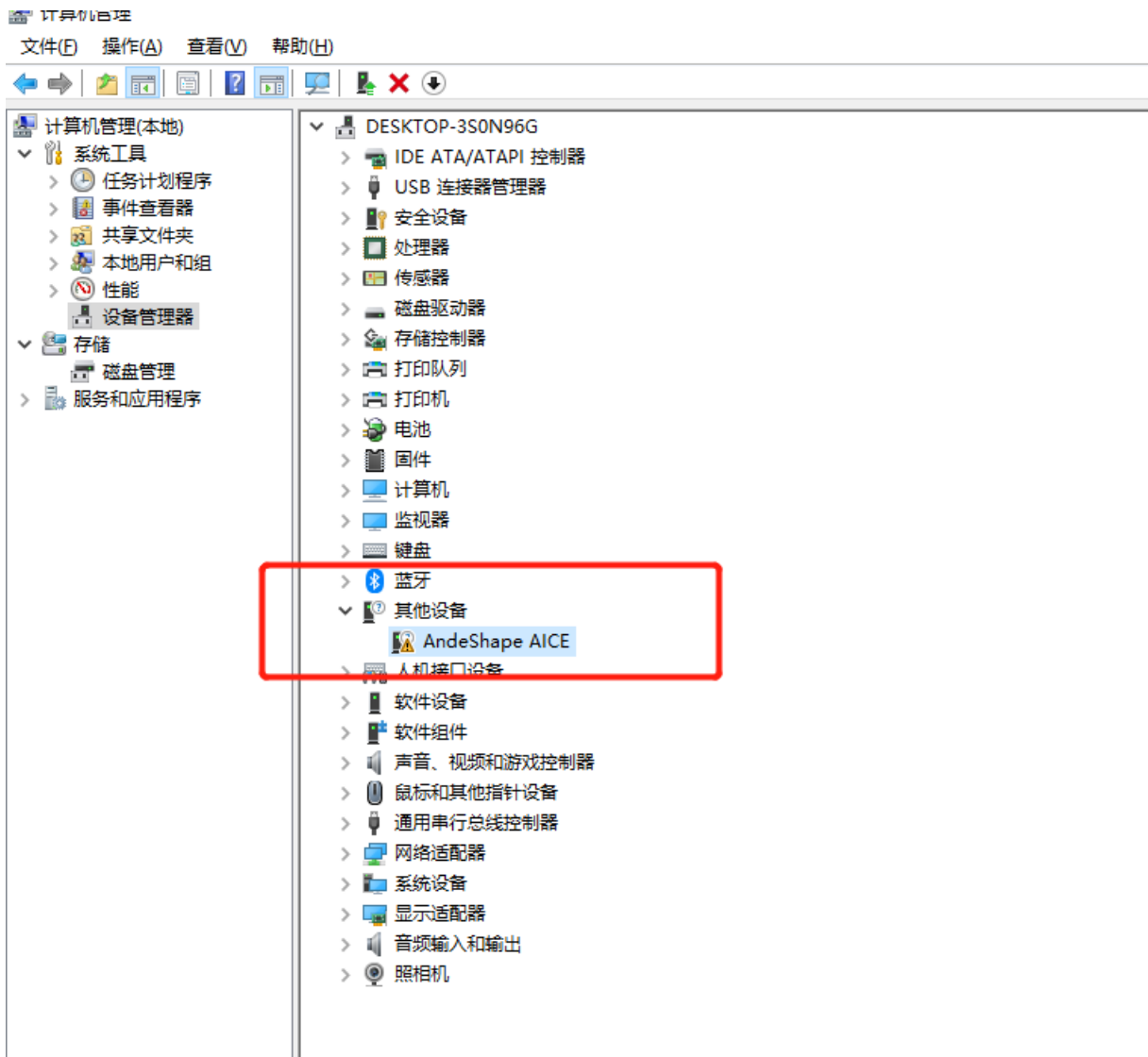


Figure 4.6: Jtag 设备驱动未安装

Step3 - 安装驱动： Jtag 工具界面“Target”区域，点击下方的“Install Jtag Driver”来安装驱动。

4.2.2.2 连接调试设备

Step1 - 接入调试设备： 将调试设备通过 Jtag 接口连接到 Jtag 设备上，正常连接状态下 Jtag 设备只有蓝灯常亮。若红灯同时处于常亮状态，则先检查 Jtag 的驱动是否安装。若驱动安装正常，接着查看 Jtag 盒子与测试板之间的连线是否正常。

Step2 - 连接目标设备： 调试设备成功接入后，点击 Jtag 工具界面“Target”区域的“Connect Target”按钮来连接调试设备。若 Jtag 工具界面的标题栏显示为“Jtag Tool - target connected”，则表示已经连接成功，此时“Debug”区域的控件变为可操作状态。若连接不成功，可通过插拔一下 USB 接口来解决。

4.2.2.3 下载功能

Step1 - 选择固件： BDT 工具主界面，通过点击“File->Open”来选择需要下载的固件。

Step2 - 设置起始地址： Jtag 工具主界面“Debug”区域，在“Addr”对应的输入框中输入下载的起始地址。

Step3 - 设置固件载入的对象： Jtag 工具主界面“Debug”区域，通过点击“FLASH”和“SRAM”两个单选框控件来选择。

Step4 - 执行下载： Jtag 工具主界面“Debug”区域，点击“Download”按钮来执行下载操作。

注意：

下载到 Flash 中的程序需通过点击“Restart”按钮来运行，而对于下载到 SRAM 中的程序则通过点击“Run”按钮来运行。

4.2.2.4 擦除功能

Step1 - 设置起始地址： Jtag 工具主界面“Debug”区域，在“Addr”对应的输入框中输入下载的起始地址。

Step2 - 设置擦除长度： Jtag 工具主界面“Debug”区域，在“Erase Flash”按钮下方“Len”对应的输入框中输入需要擦除的长度。

Step3 - 执行擦除： Jtag 工具主界面“Debug”区域，点击“Erase Flash”按钮来执行擦除操作。

4.2.2.5 读写功能

该工具支持 Memory、Flash 和 Analog 三种模式的读写，其中 Memory 方式可支持 byte、half-word (2 byte)、word (4 byte) 三种方式的读写，其他模式只支持按 byte 来读写。

Step1 - 设置读写模式： Jtag 工具主界面“Debug”区域，点击“Write”按钮后面的下拉框来选择模式。

Step2 - 设置读写宽度： Jtag 工具主界面“Debug”区域，点击读写模式选择框下方的下拉框来选择读写的宽度。

Step3 - 设置读写的地址、长度： Jtag 工具主界面“Debug”区域，在“Write”按钮下方“Addr”对应的输入框中输入地址，在“Len”对应的输入框中输入长度。如果是写数据，还需要在“Data”对应的输入框中输入需写入的数据。

Step4 - 执行读写操作： Jtag 工具主界面“Debug”区域，点击“Read”按钮来执行读操作，点击“Write”按钮来执行写操作。

4.2.2.6 调试功能

该工具支持的调试功能包括：读状态寄存器、读变量值以及常用的调试操作。

Step1 - 读状态寄存器： Jtag 工具主界面“Debug”区域，在“Reg”对应的输入框中输入状态寄存器的名称，接着按下回车键就可以查找对应的寄存器，接着通过点击下拉框来选择对应的寄存器就可以查看该寄存器的值。如果不知道寄存器的名称，也可通过输入“*”来列出所有可查看的寄存器。

Step2 - 读变量值： Jtag 工具主界面“Debug”区域，在“Var”对应的输入框中输入变量的名称，接着按下回车键就可以查找对应的变量，接着通过点击下拉框来选择对应的变量就可以查看该变量的值。如果不知道变量的名

称也可通过输入 “*” 来列出所有可查看的变量。**注意：**要实现该功能需要保证在 bin 文件目录下存在与之同名的 list 文件。

Step3 - 常用调试操作： Jtag 工具主界面“Debug”区域，点击“Stall”按钮来使设备停止，点击“Restart”按钮使设备从 flash 中重新加载固件运行，点击“Pause”按钮来使设备程序暂停，点击“Run”按钮使设备从程序暂停处继续运行，点击“Step”按钮来使设备程序单步运行，点击“PC”按钮来查看设备当前的 PC 指针。另外，如果在 bin 文件目录下存在与之同名的 list 文件，还可以查看对应的汇编代码，不过在使用该功能前还需要在“Var”对应的输入框中输入 “*” 并按下回车键。

4.3 TWS Tool

4.3.1 Download Tool

4.3.1.1 功能描述（Download Tool）

“Download Tool”主要用来烧写 bin 文件、参数信息等到 Flash 的指定地址当中。该工具支持一次烧录单个和多个对象，最多支持一次烧录 4 个 bin 文件、1 组 “BT Name”参数以及 1 组“BT MAC”参数。

4.3.1.2 界面介绍

“Download Tool”的界面如下图所示，主要包括“Setting”和“Operation”两个区域。其中“Setting”区域负责设置下载的信息，“Operation”部分负责执行下载操作。

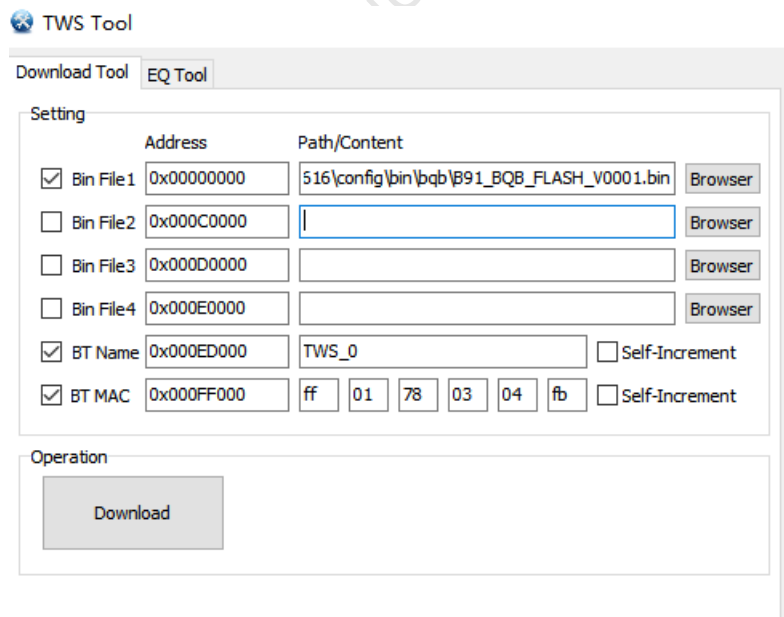


Figure 4.7: TWS_Tool/Download_Tool 界面

4.3.1.3 使用说明

Step1 - 设置烧录信息： Download 工具的主界面“Setting”区域，通过最左侧的单选框选择需要烧录的信息，在“Address”对应的输入框中输入各信息写入的 Flash 地址，在“Path/Content”对应的输入框中输入对应的信息。

对于固件下载，通过点击对应得“Browser”按钮来加载文件路径。而对于“BT Name/MAC”，则通过对应的输入框直接输入。“BT Name/MAC”都有自增的功能（每烧录完一组数据对应的值就会自增），通过右侧的单选框来选择。另外需要注意的是“BT Name”的最大长度不得超过 25 字符。

Step2 - 执行烧写操作： Download 工具的主界面“Operation”区域，通过点击“Download”按钮来执行下载操作。

4.3.2 EQ Tool

4.3.2.1 功能描述（EQ Tool）

该工具的主要功能如下：

- 生成 EQ 参数
- 进行在线 EQ 调参（需相关协议支持）

4.3.2.2 界面组成

EQ 工具的界面由 5 个区域构成，包括图表显示区域、“Configure”区域、“Parameter”区域、log 输出区域以及“Operation”区域，具体如下图所示。

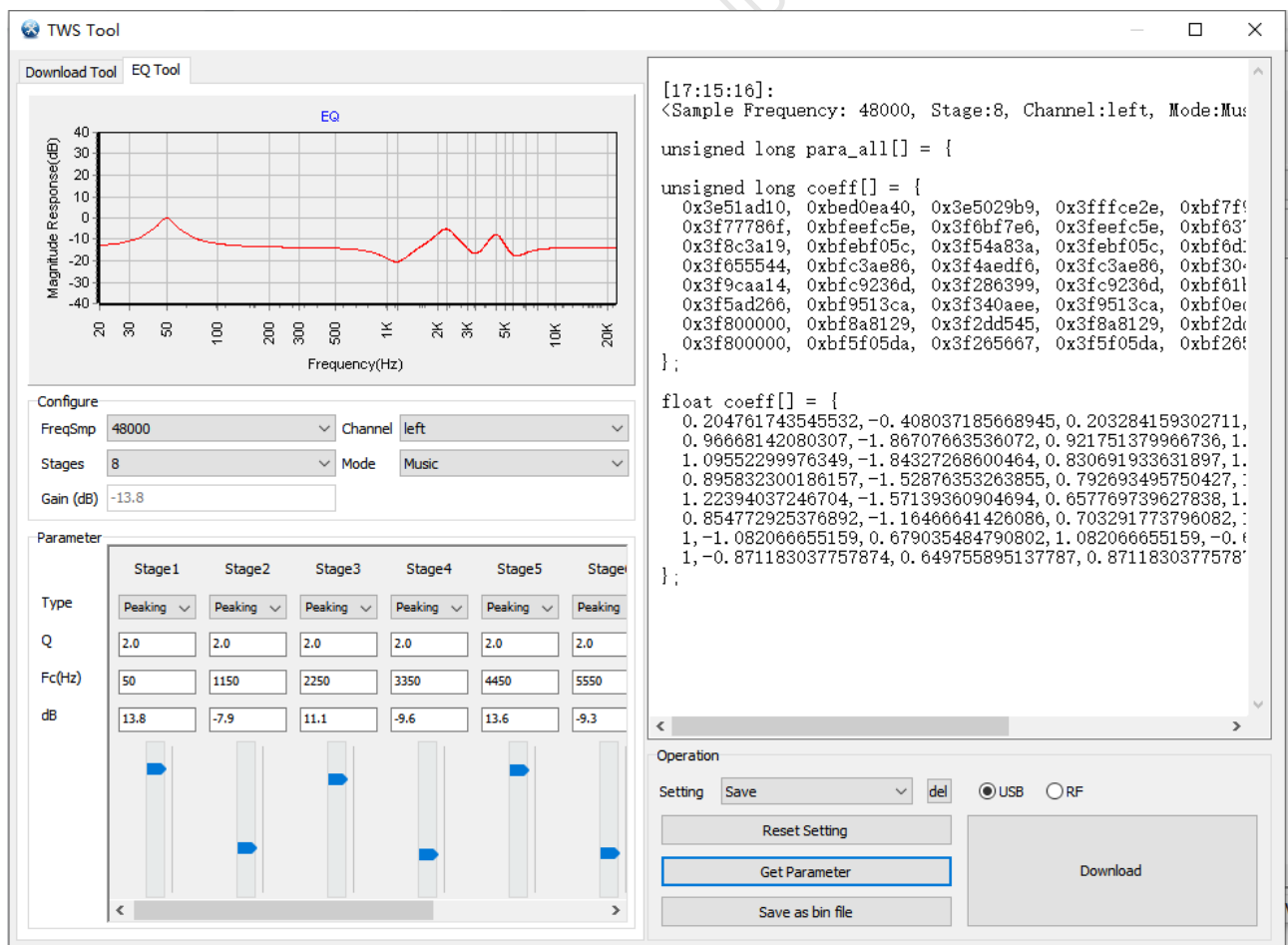


Figure 4.8: TWS_Tool/EQ_Tool 界面

4.3.2.3 使用说明

Step1 - EQ 参数调节：首先选择调参的模式，通过“configure”区域下的 Mode 进行选择，工具默认提供三种可选模式，分别是：“Music”模式（即 Line Out），“Speech MIC”模式（即 Line In），“Speech SPK”模式（即 Line In to Line Out）。接着选择声道，通过“configure”区域下的 Channel 进行选择。接着选择样本频率和 EQ 调参的级数，分别通过“configure”区域下的 FreqSmp 和 Stages 进行选择。最后配置各级参数，包括滤波器类型、Q 值、中心频率以及增益值，分别通过“parameter”区域下 Type、Q、Fc 以及 dB 来配置。

Step2 - 查看和保存 EQ 参数值：通过点击“Operation”区域中的“Get Parameter”按钮来获取参数值，通过点击“Save as bin file”按钮可以将生成的参数值保存成 bin 文件。

Step3 - 保存 EQ 配置：通过“Operation”区域中 Setting 对应的复选框可进行 EQ 配置的保存和还原。对于保存操作，首先复选框选择“Save”，接着在弹窗中输入配置名称，最后点击 OK。对于还原已保存的配置，通过复选框选择已保存的配置名称即可还原配置。

Step4 - EQ 在线调参：首先选择在线调参的方式，通过“Operation”区域中的单选框控件（“USB”和“RF”）来选择在线调参的方式。接着通过点击“Operation”区域中的“Download”按钮将 EQ 参数下载到设备中。

注意：

EQ 工具所支持的在线调参方式都需要有相关协议的支持，具体需咨询相关工作人员。

4.4 RISC-V Tool

支持芯片通过 USB 实现如下功能：

- 读写寄存器
- 读写擦 flash
- 上报调试信息
- VCD 功能

注意：

SDK 中必须有相应的交互协议，此工具才能使用。

4.4.1 读存储器

从特定存储空间（RAM/Analog/Flash）读取数据，操作步骤如下。

Step1 - 选择存储空间类型：在界面“Memory”区域点击下拉菜单选择类型。

Step2 - 设置访问区域：“Addr”输入框输入所访问空间起始地址，并在“Size”输入框输入所需访问地址的个数。

Step3 - 读取数据：将光标保持在“Addr”框中，通过按下键盘“Tab”按键，用户可以从指定的存储空间读取数据。

注意：

Size 输入框中的数据为十六进制。写入“10”代表读取 16 个字节数据。最多一次只能读取 256 个字节数据。

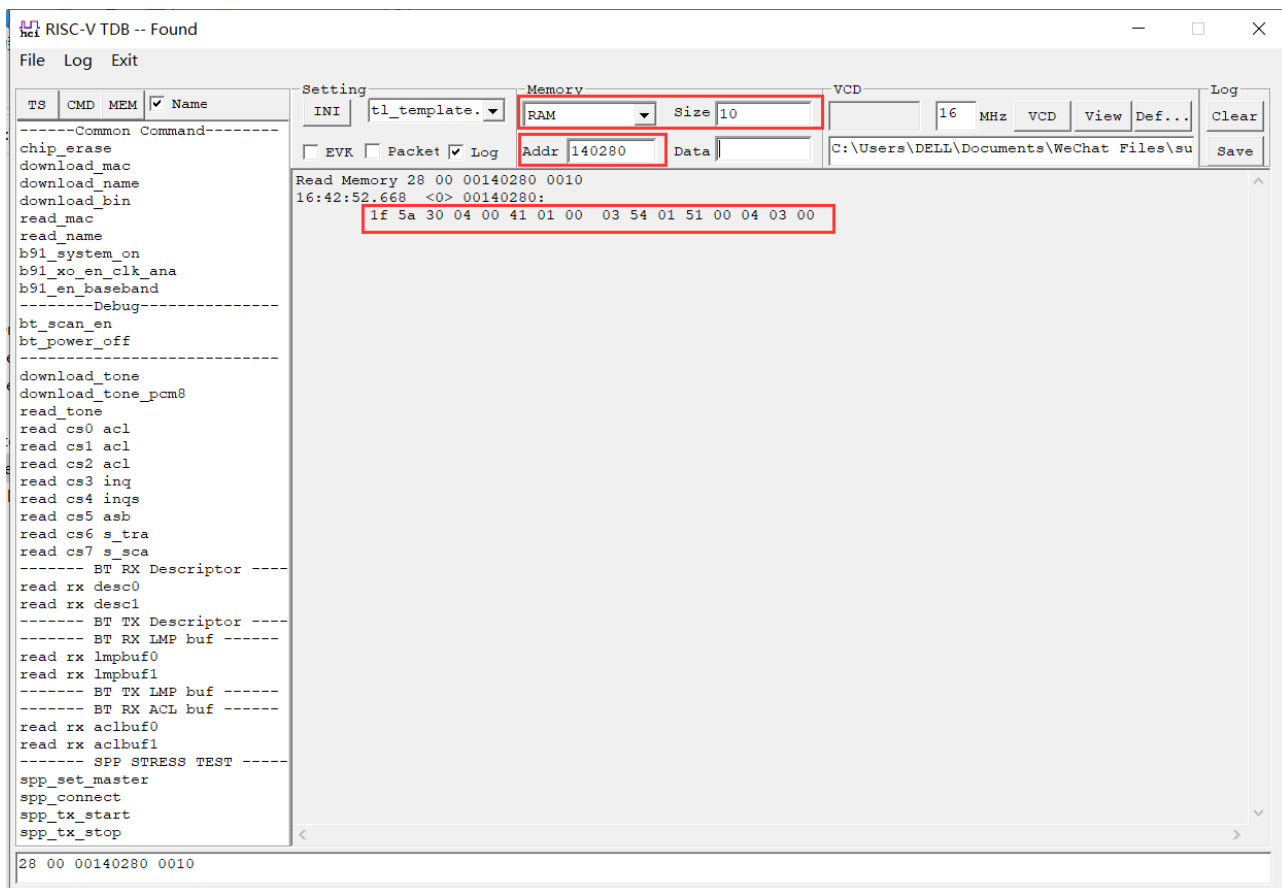


Figure 4.9: 读存储器

4.4.2 写存储器

在特定存储空间（RAM/Analog/Flash）写入数据，操作步骤如下。

Step1 - 选择存储空间类型：在界面“Memory”区域点击下拉菜单选择类型。

Step2 - 设置访问区域：“Addr”输入框输入所访问空间起始地址，并在“Size”输入框输入待写入数据的长度。

Step3 - 写入数据：Data 框输入待写入的目标数据，通过按下键盘“Enter”按键，用户可以将指定的数据写入到指定的存储空间。

注意：

写数据最多只能写入四个字节数据。

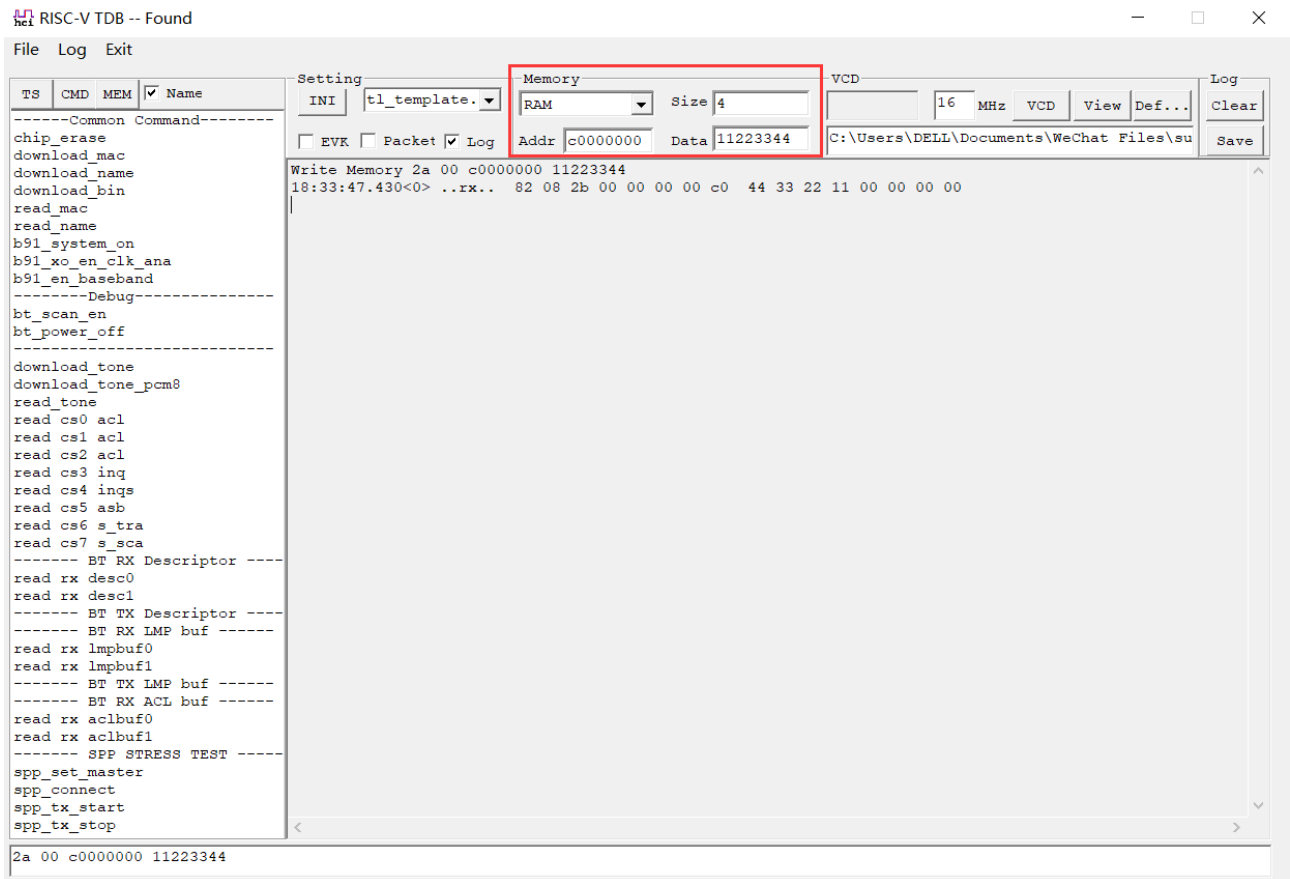


Figure 4.10: 写存储器

4.4.3 固件下载

将固件下载到 flash 中，操作步骤如下。

Step1 - 选择固件： 点击“File->Import bin”，选择所需下载的固件。

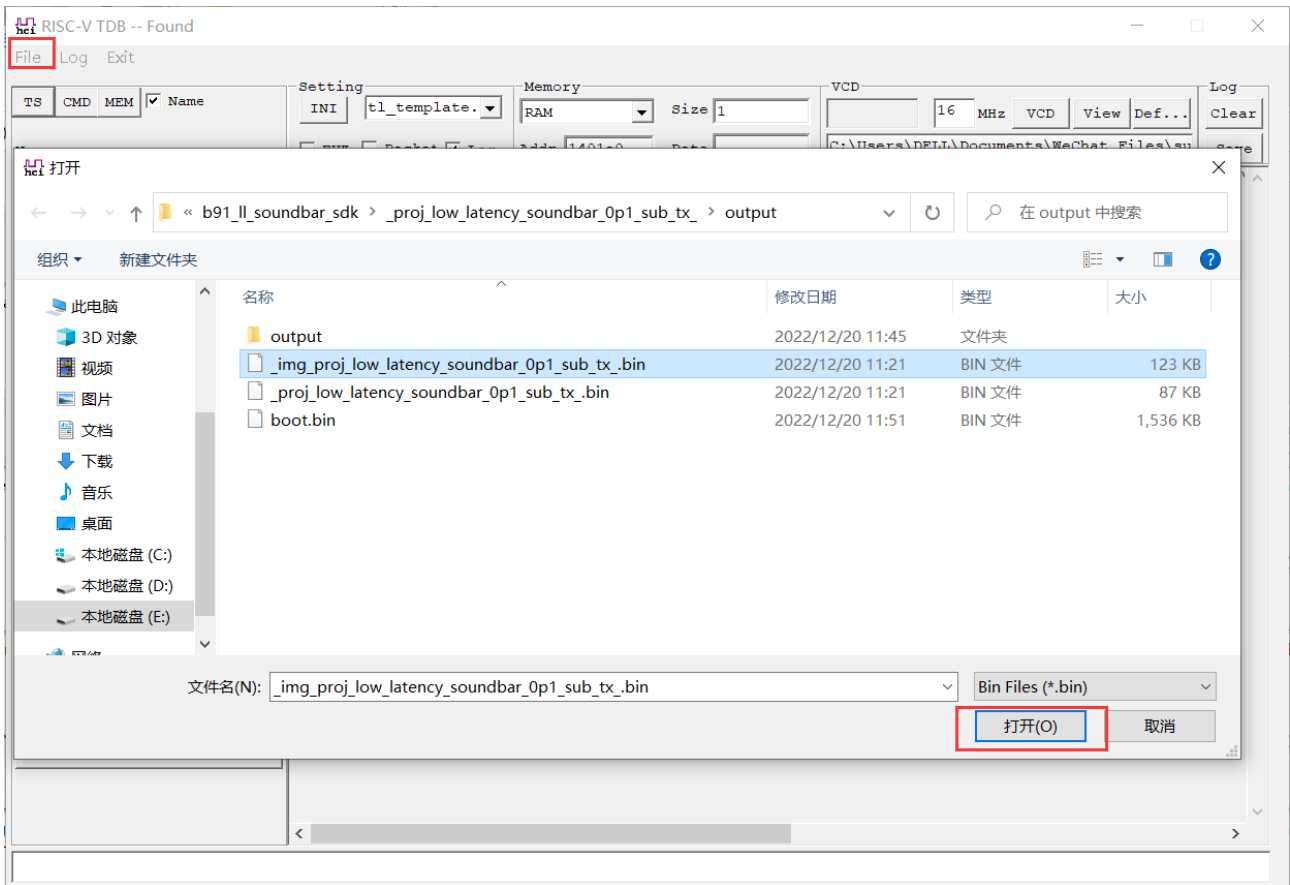


Figure 4.11: 选择固件

Step2 - 擦除 flash: 点击“TS”按钮，其次点击“Chip_Erase”按钮擦除 flash 所有数据。

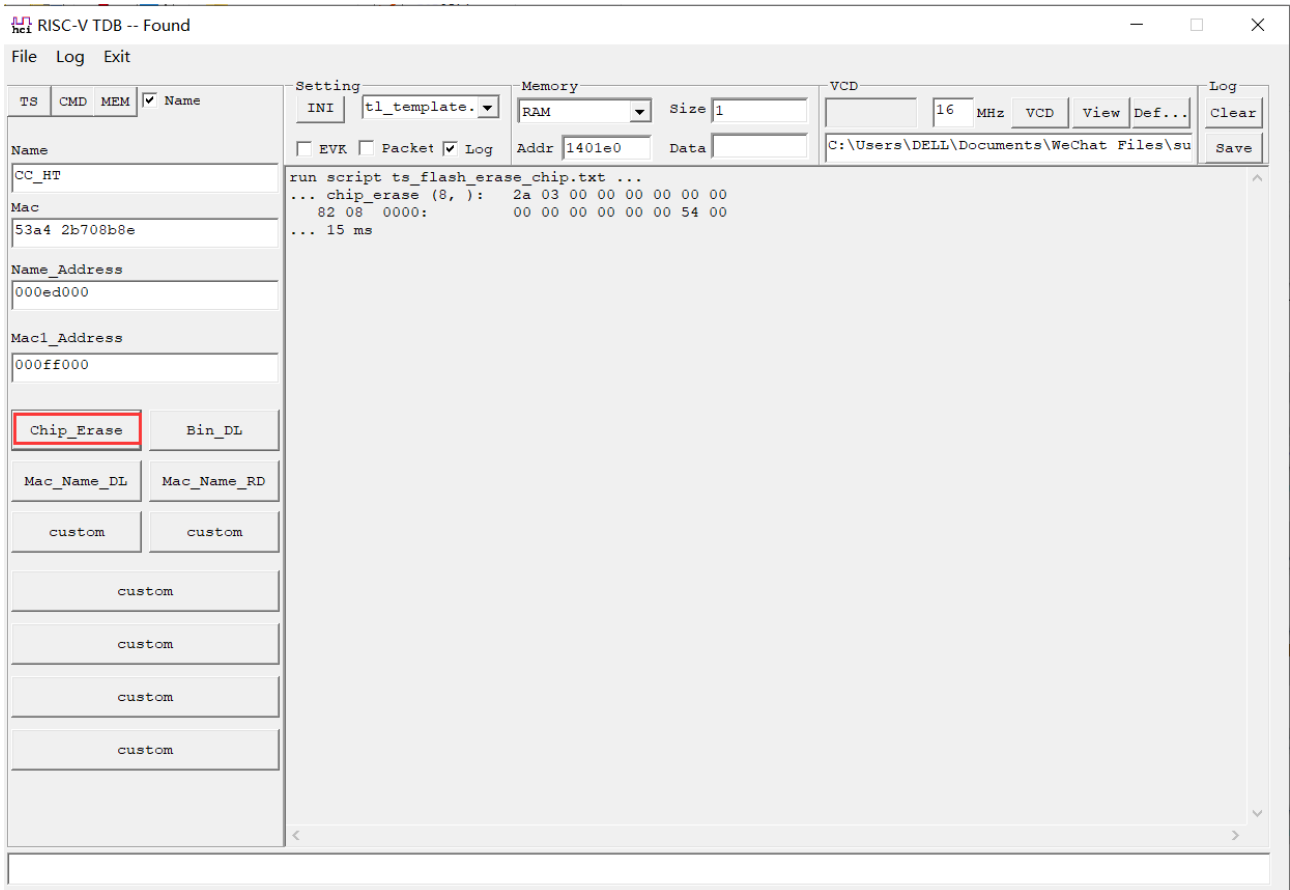


Figure 4.12: 擦除 flash

Step3 - 下载固件: 点击“Bin_DL”按钮下载固件。

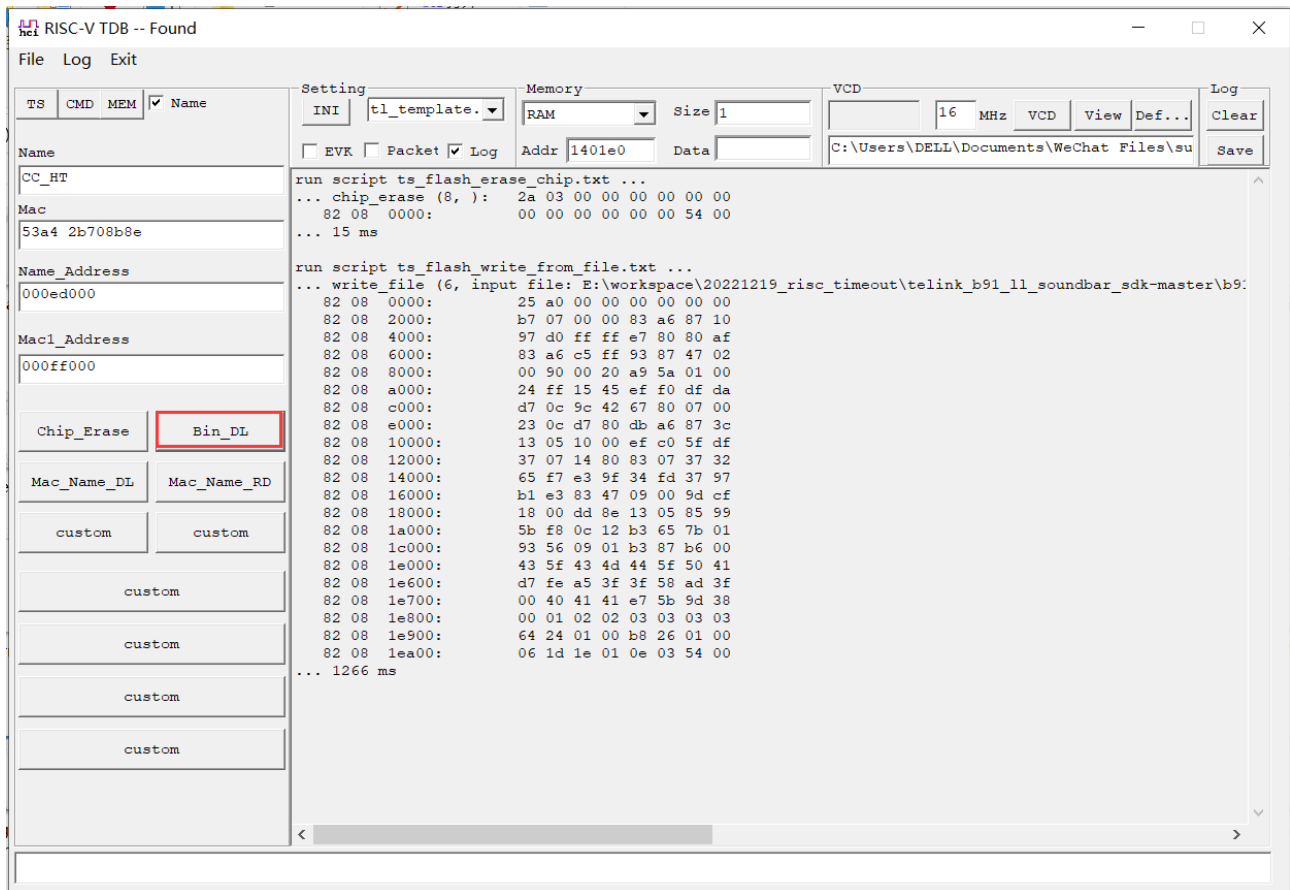


Figure 4.13: 下载固件

4.4.4 VCD 功能

VCD 功能操作步骤如下。

Step1 - 选择 header 文件： VCD 功能区点击“Def”按钮，选择对应的 header 文件。

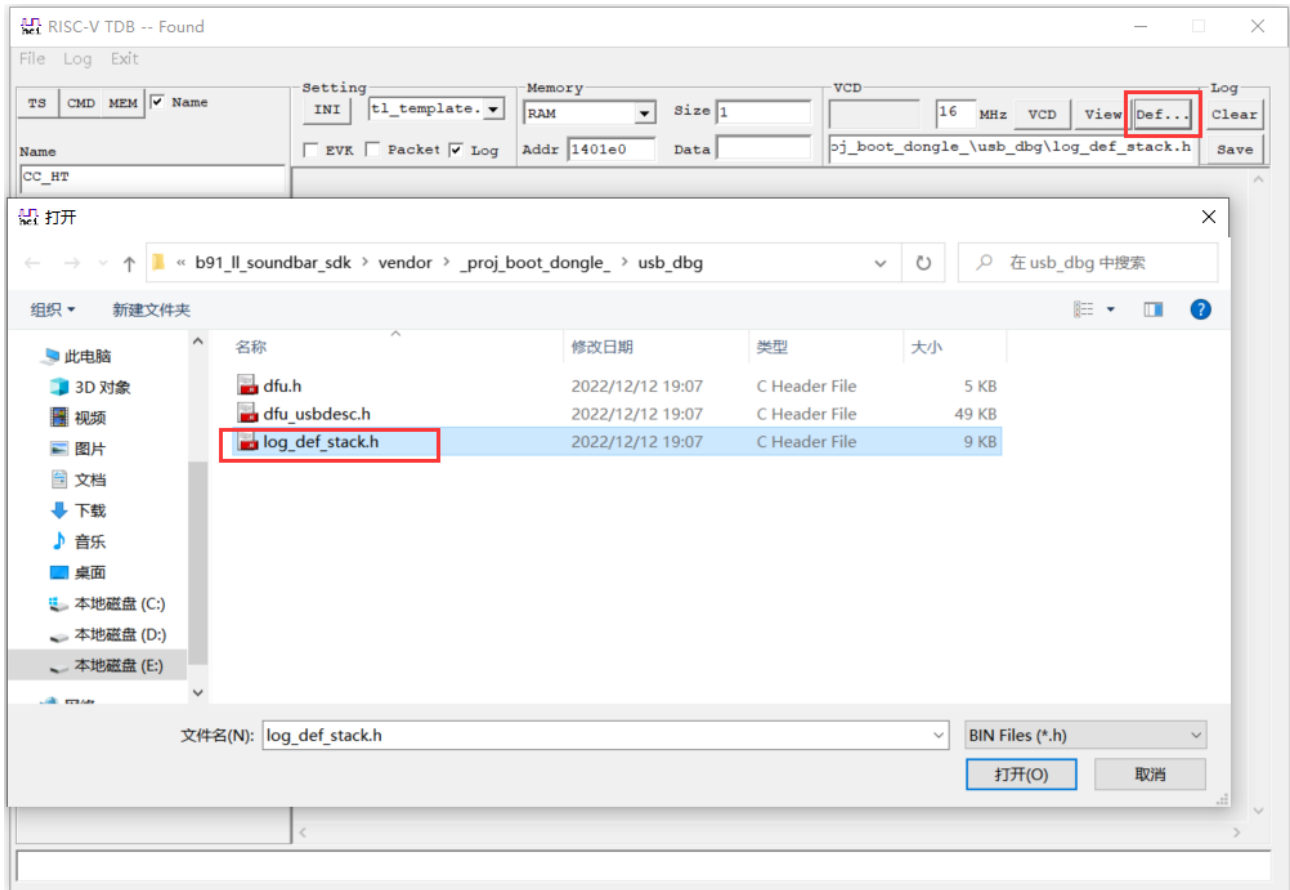


Figure 4.14: 选择 header 文件

Step2 - 启动 VCD 功能： 点击“VCD”按钮启动。

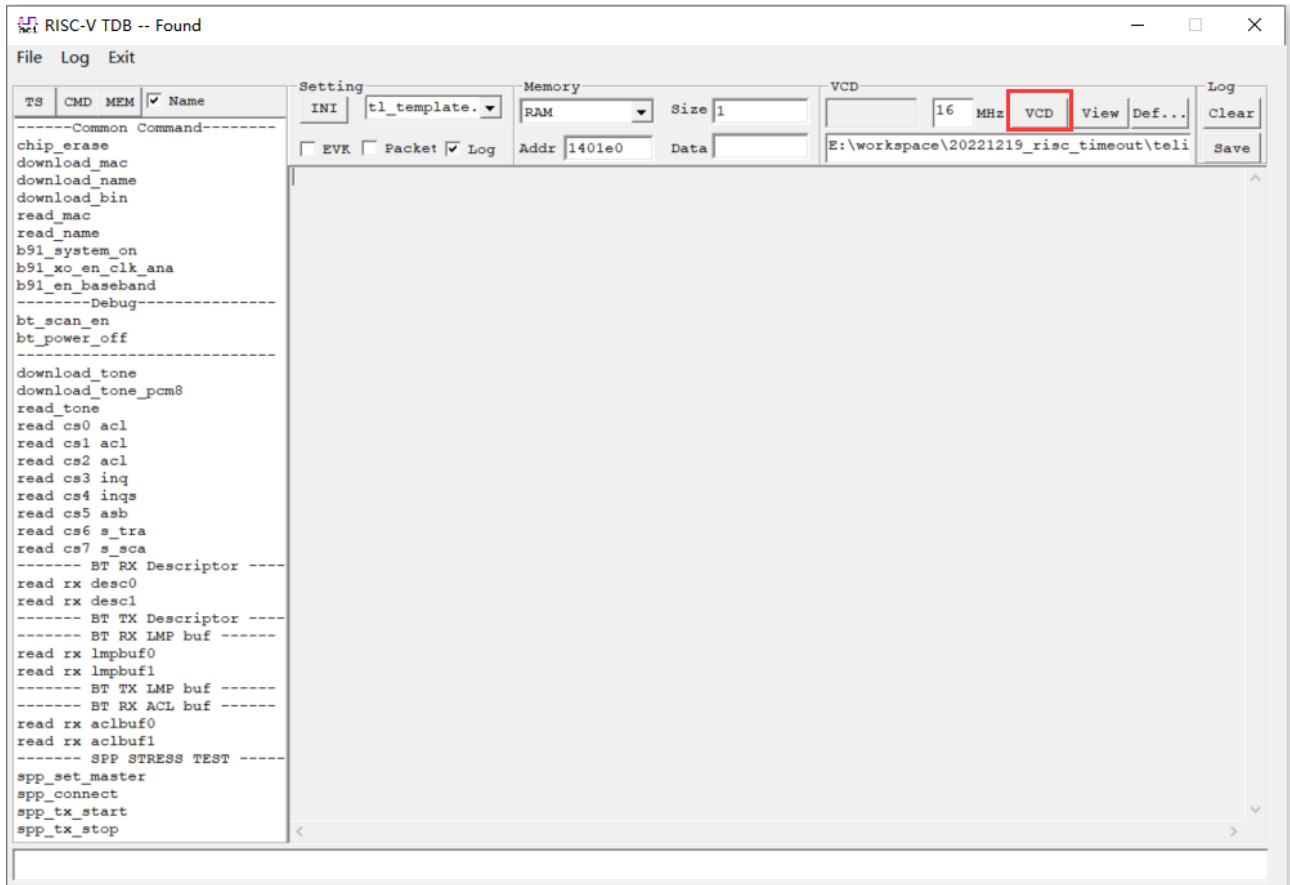


Figure 4.15: 启动 VCD 功能

Step3 - 查看相关任务运行状态： 点击“View”按钮，查看任务运行状态。

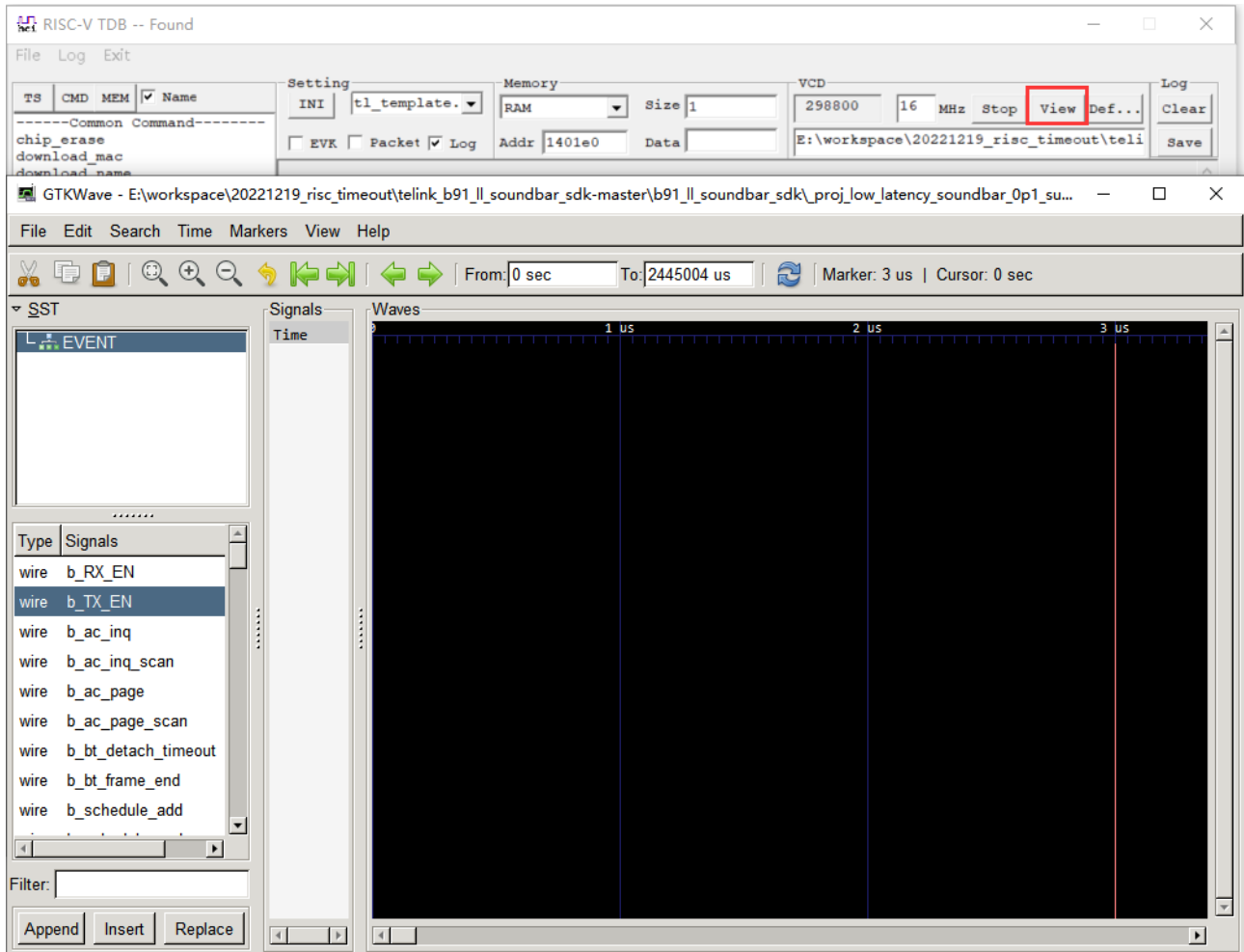


Figure 4.16: 查看状态

注意：

header 文件的定义和使用请咨询相关 SDK 负责人员。

4.4.5 命令行说明

为方便调试，可输入对应命令执行相应操作。

4.4.5.1 读命令

命令格式: [28] [Type] [Address] [Size]

Type: 访问空间类型, “00”代表访问数字寄存器; “01”代表访问模拟寄存器; “02”代表访问 flash。

Address: 访问空间的地址, 占四个字节。

Size: 访问地址的个数, 占 4 个字节。

示例: 读取数字寄存器 0x140280 地址一个字节数据。

命令：28 00 00140280 000001

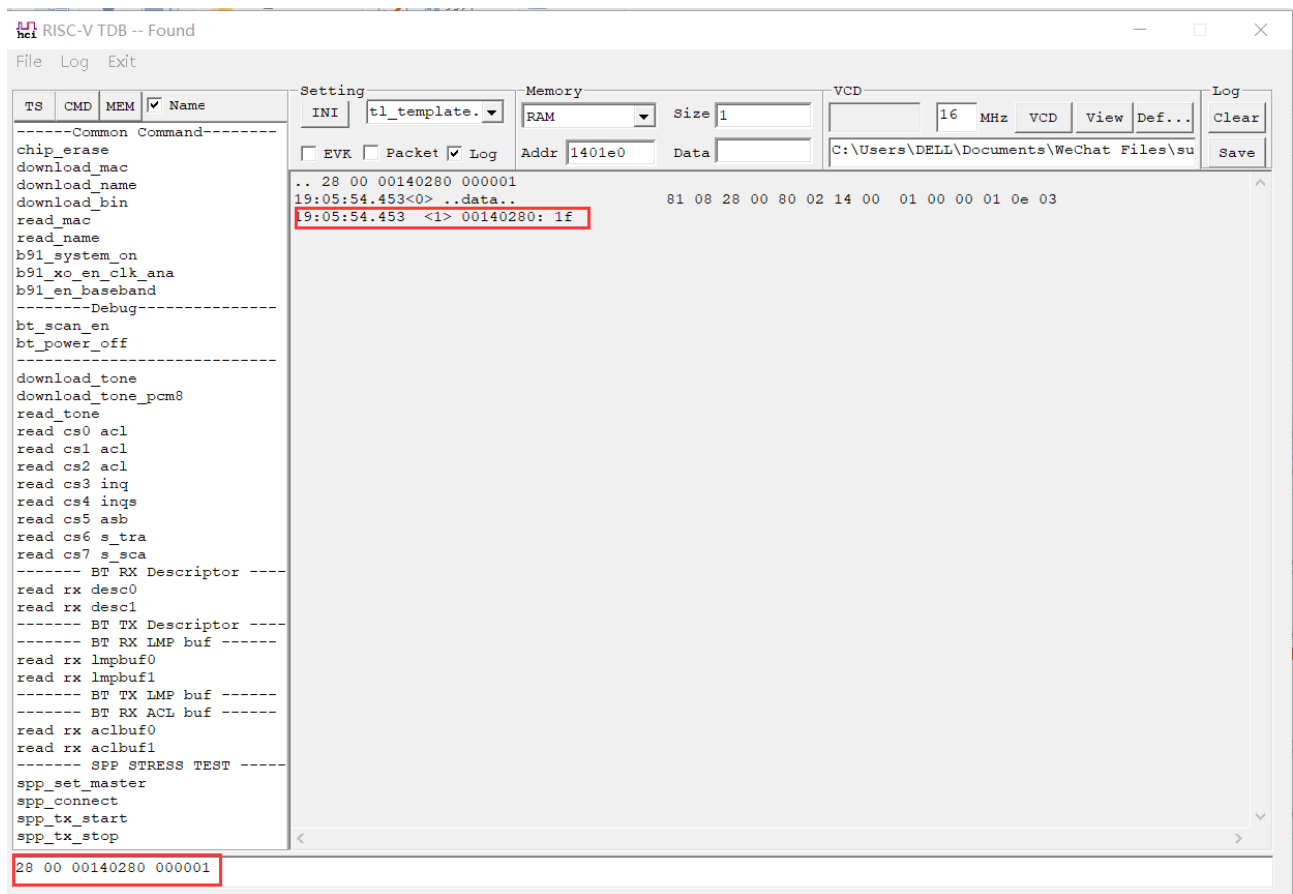


Figure 4.17: 读命令

4.4.5.2 写命令

命令格式 [2a] [Type] [Address] [Data1] [Data2].....[DataN]

Type: 访问空间类型, “00”代表访问数字寄存器; “01”代表访问模拟寄存器; “02”代表访问 flash。

Address: 访问空间的地址, 占四个字节。

DataN: 待写入的数据。

示例: flash 0x1000 地址写入数据“0x1122334455”。

命令: 2a 02 00001000 55 44 33 22 11

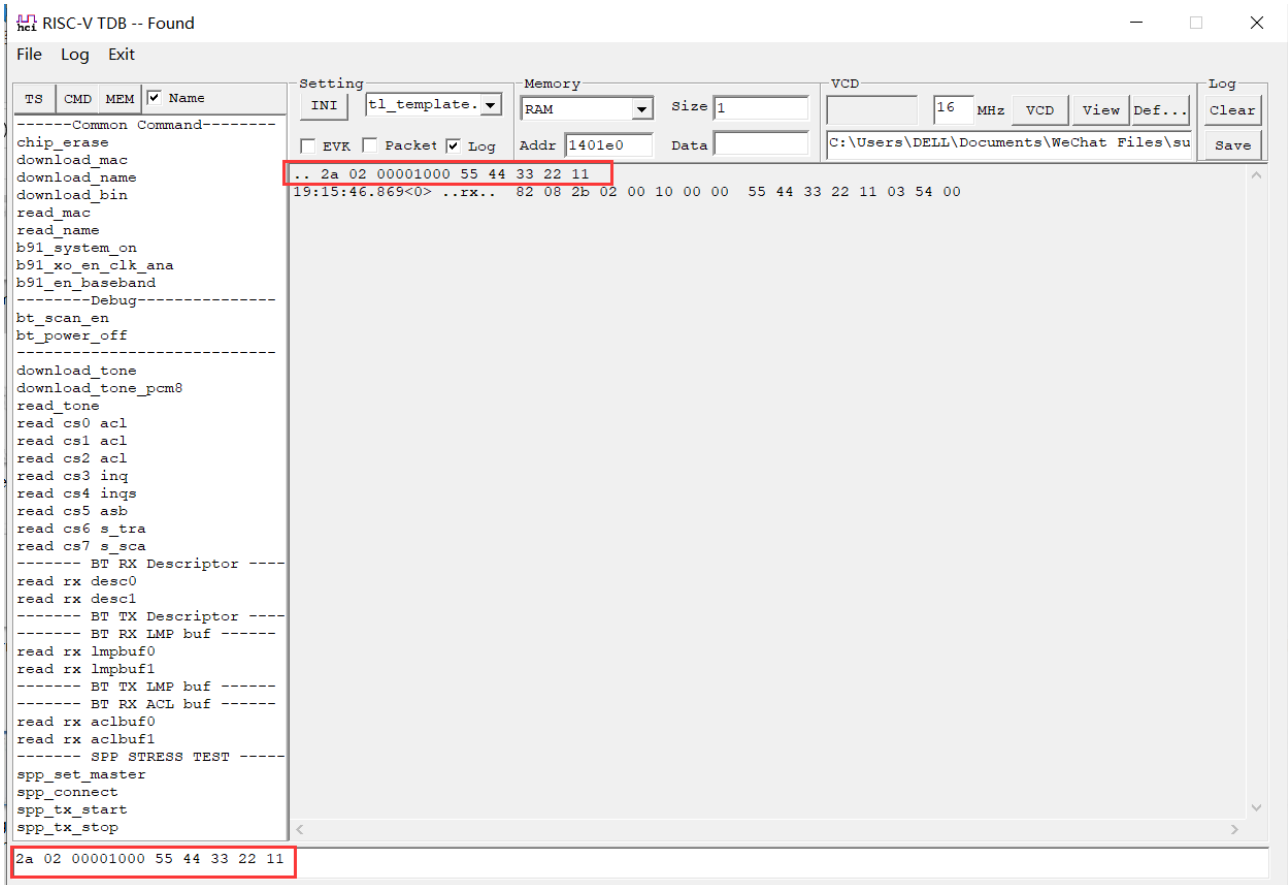


Figure 4.18: 写命令

4.4.5.3 擦 flash 命令

命令格式 [2a] [03] [Address] [Size]

Address: 访问空间的地址, 占四个字节。

Size: 擦除 flash 的空间大小, 占四个字节。

示例: flash 0x1000 地址擦除 4k 数据。

命令: 2a 03 00001000 00001000

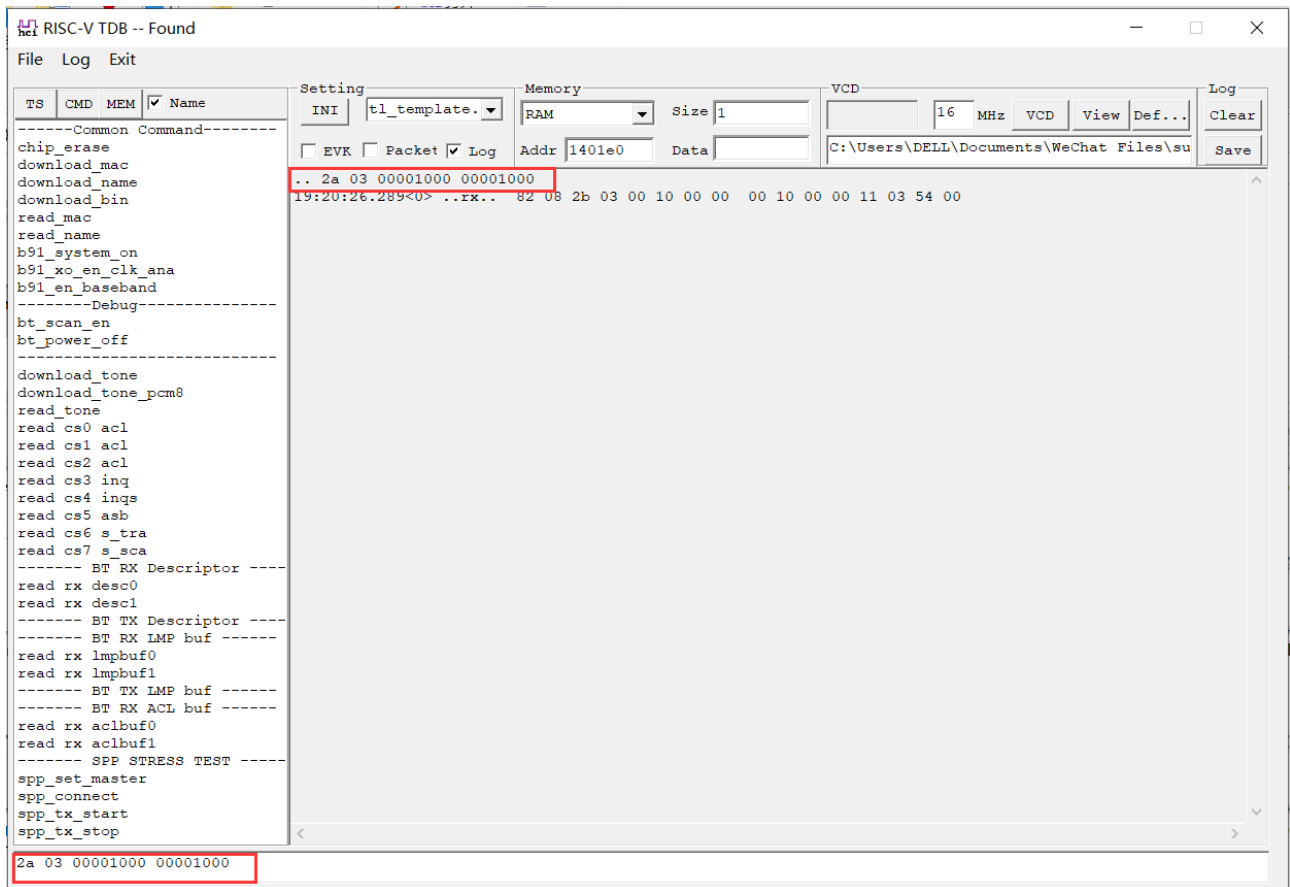


Figure 4.19: 擦 flash 命令

示例：擦整片 flash。

命令：2a 03 00000000 00000000

4.4.5.4 保存所读数据

命令格式 [r] [String] [28] [Type] [Address] [Size] [BinPath]

r: 读命令。

String: 字符串，可自定义。

Type: 访问空间类型，“00”代表访问数字寄存器；“01”代表访问模拟寄存器；“02”代表访问 flash。

Address: 访问空间的地址，占四个字节。

Size: 所需要读取数据大小，占四个字节。

BinPath: 所读数据待保存的路径。

示例：从 flash 0 地址读取 4k 数据，并保存到 D 盘。

命令：r read 0228 00000000 00001000 -d:\read.bin

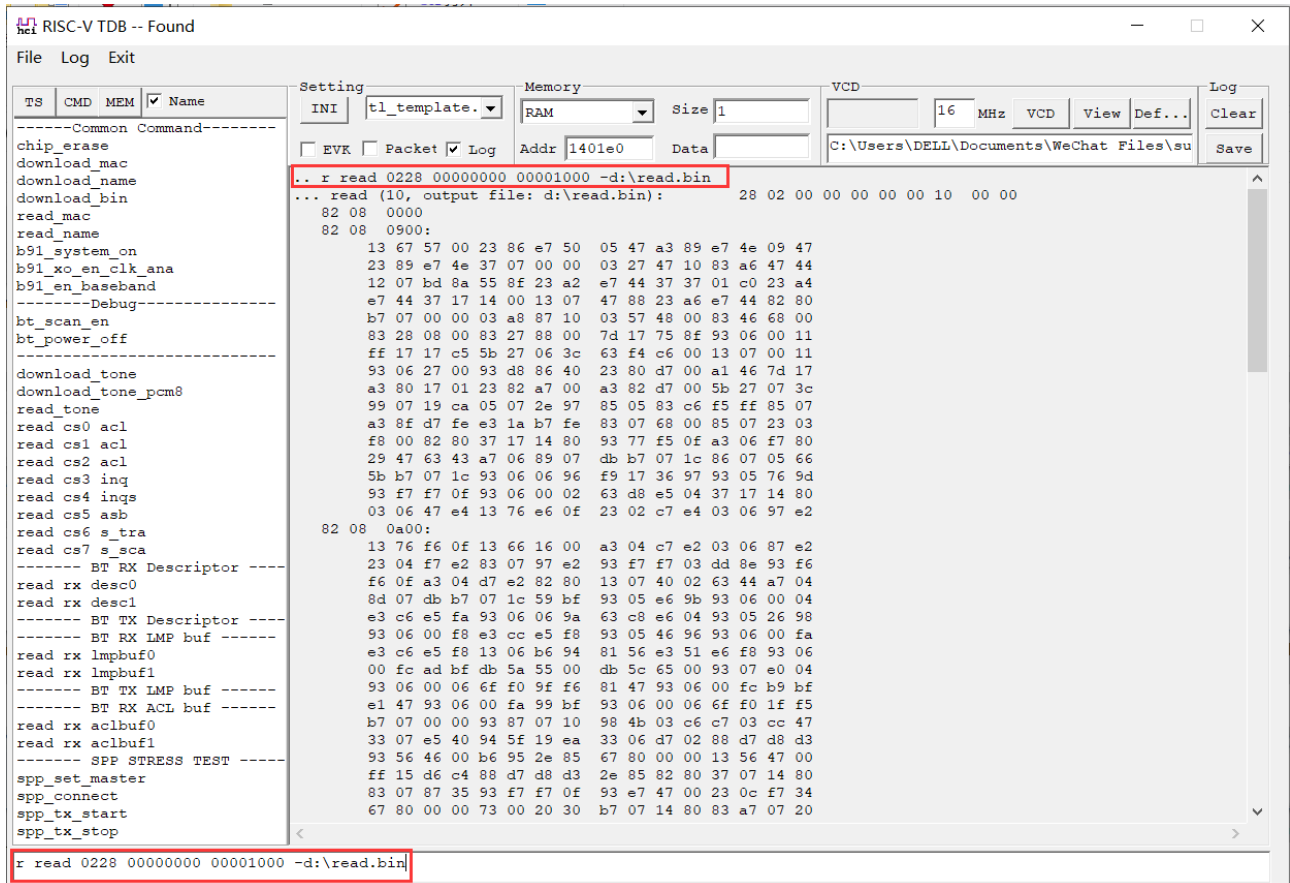


Figure 4.20: 保存所读数据

4.4.6 日志清除及保存

日志清除：点击界面 Clear 按钮即可清除日志。

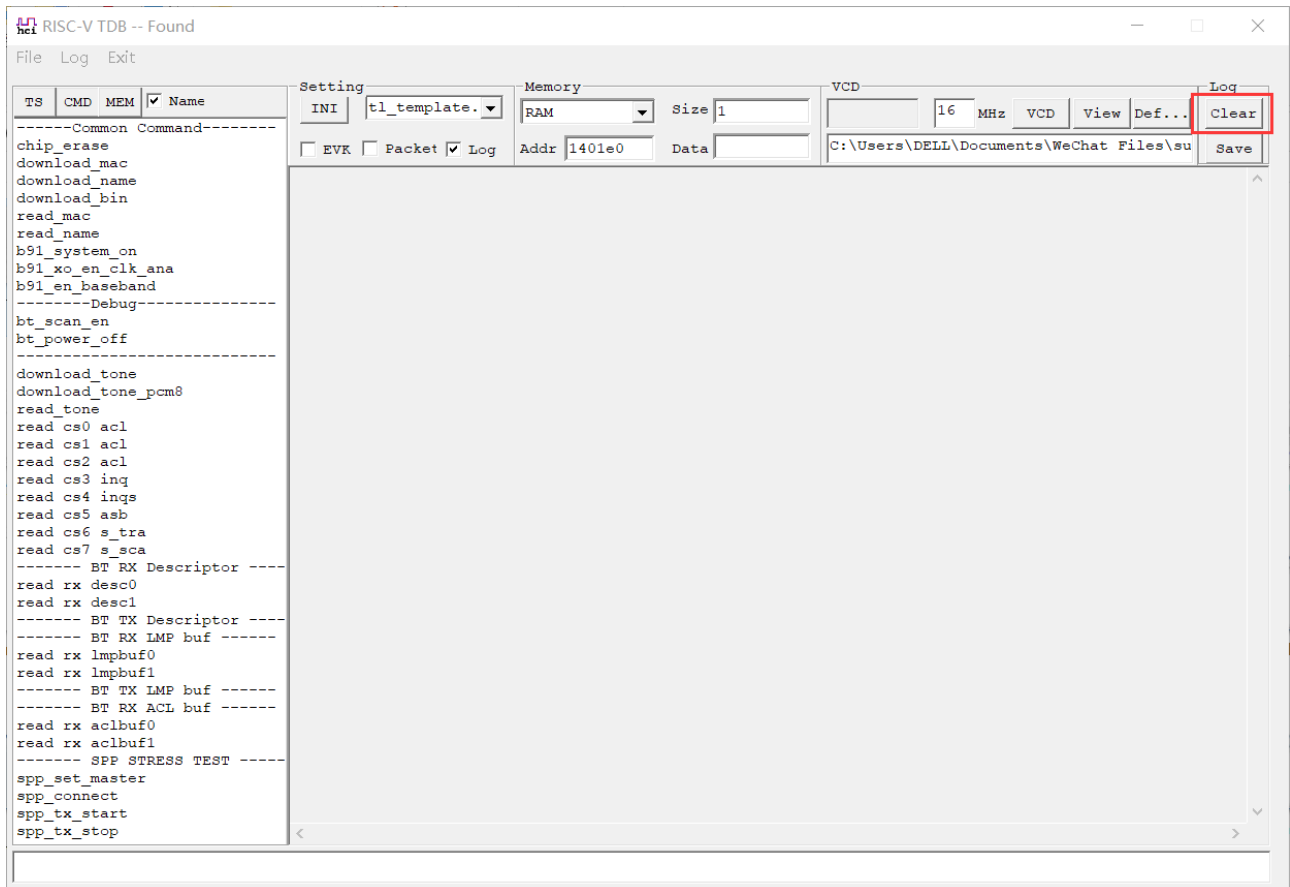


Figure 4.21: 清除日志

日志保存：点击界面 Save 按钮，选择文件保存路径并输入文件名即可。

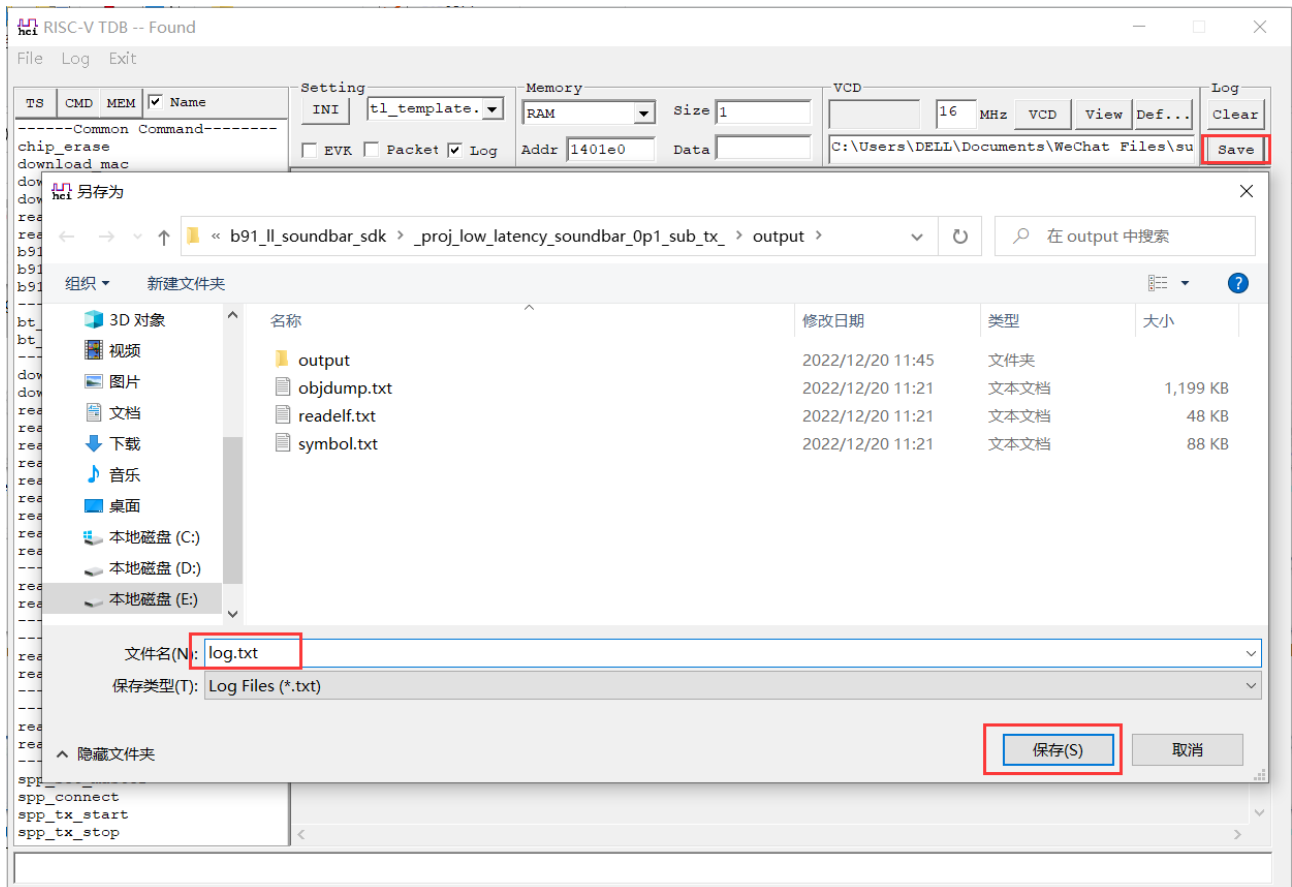


Figure 4.22: 保存日志

4.5 EMI Tool

4.5.1 功能描述 (EMI Tool)

该工具有两个主要的功能：

- 配置和下载测试固件。
- 进行 EMI 测试。
- 进行非信令测试。
- 进行 RF 电流测试。

4.5.2 界面组成及使用说明 (BQB Tool)

该工具主要由固件配置下载界面、EMI 测试界面、非信令测试界面以及 RF 电流测试界面四个部分组成，具体如下图所示。

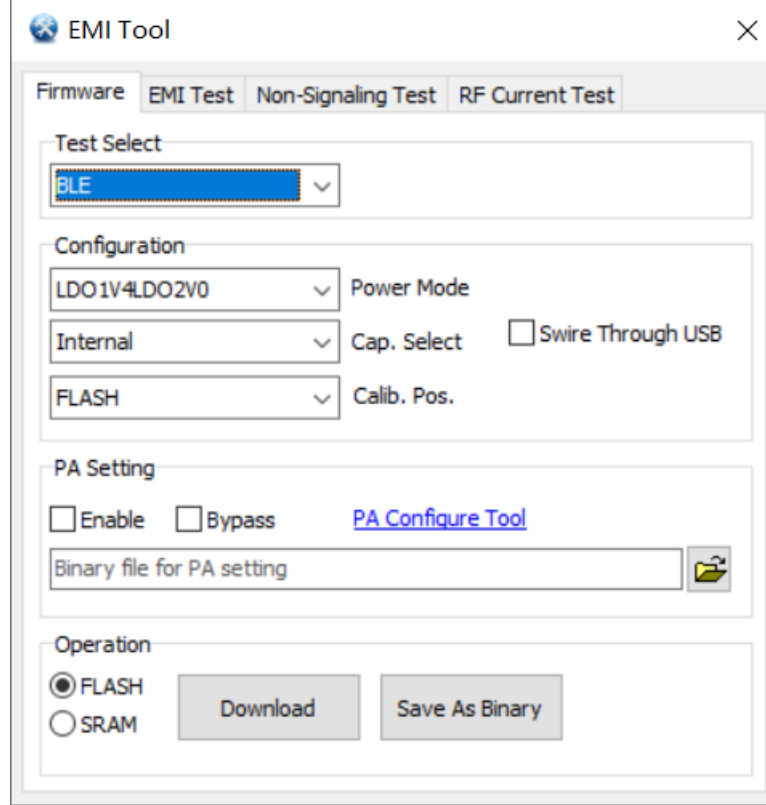


Figure 4.23: EMI 工具固件配置下载界面

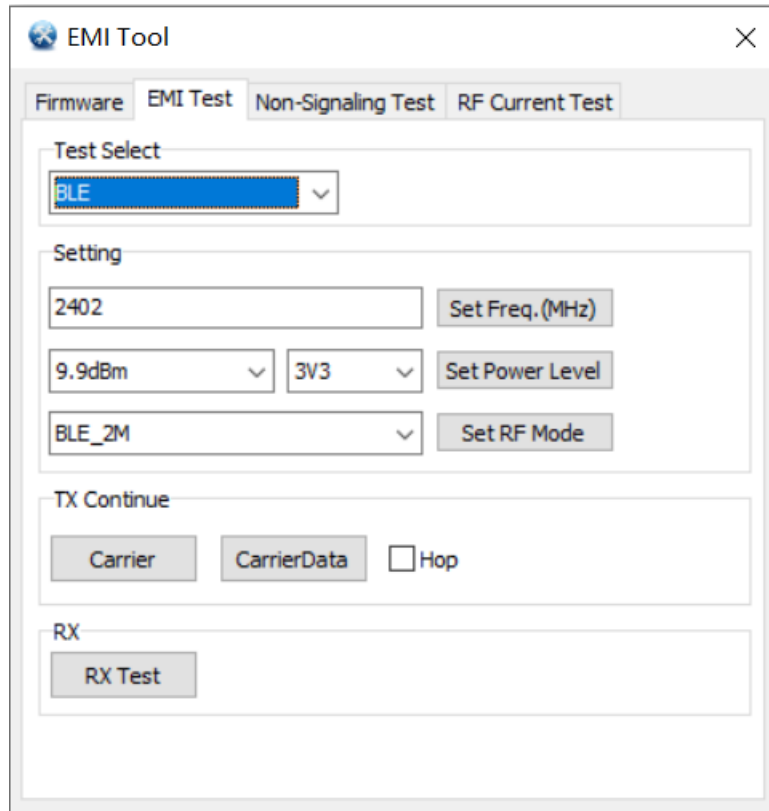


Figure 4.24: EMI 工具 EMI 测试界面

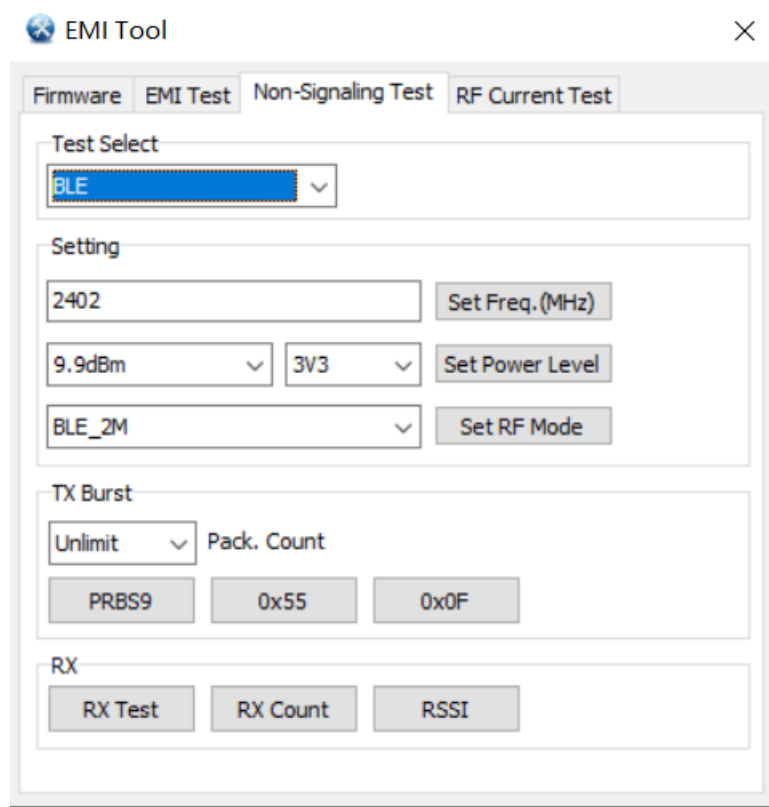


Figure 4.25: EMI 工具非信令测试界面

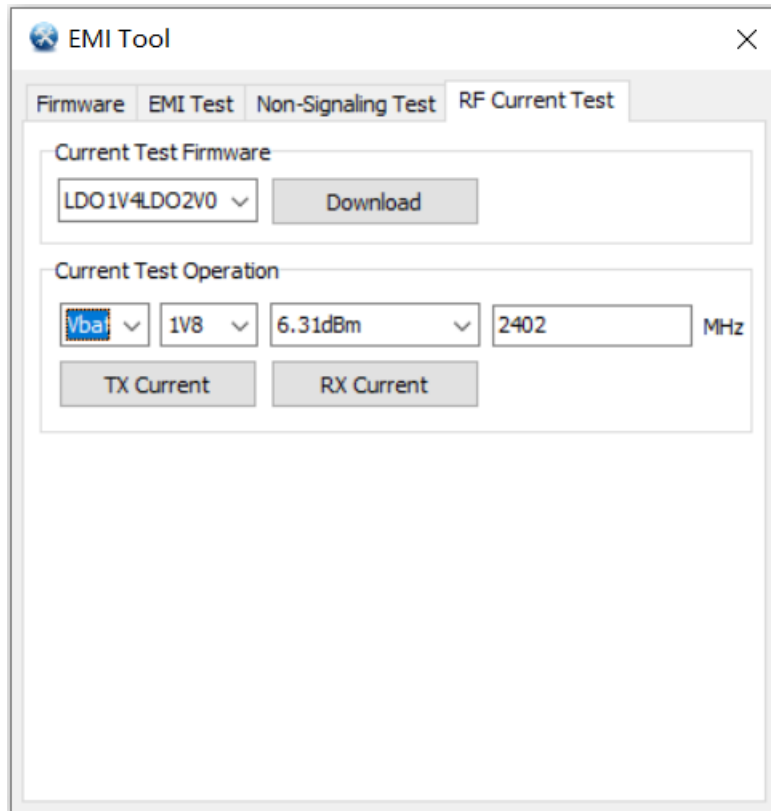


Figure 4.26: EMI 工具 RF 电流测试界面

4.5.2.1 芯片的选择

通过 BDT 主界面 MCU 列表来进行芯片型号的选择。

4.5.2.2 固件的保存和下载

Step1 - 配置固件：固件下载界面“Configure”区域，“Cap. Select”用来配置内外部电容，“Power Mode”用来配置芯片内部的供电方式，“Calib. Pos.”用来配置校准数据的存储位置（根据实际情况选择）。如果使用“Swire Through USB”功能，则需勾选对应的选项。

Step2 - PA 配置：固件下载界面“PA Setting”区域，如果需要使用 PA 功能，则勾选上“Enable”，如果使用 PA 功能但工作在 Bypass 模式下，则勾选“Bypass”。除了上述操作，用户还需要将 PA 操作的 binary 文件路径输入到对应的文本框中。PA 操作相关的 binary 文件可通过“PA Configure Tool”来生成，可点击“PA Configure Tool”来打开该工具，工具的使用可参考对应的章节。

Step3 - 选择固件类型：固件下载界面“Operation”区域，通过单选框来选择类型，可供选择的类型包括 Flash 程序和 SRAM 程序两种。

Step4 - 保存/下载固件：固件下载界面“Operation”区域，通过点击“Save As Binary”按钮来保存固件，通过点击“Download”按钮来下载固件。

4.5.2.3 EMI 测试

Step1 - 配置频点、功率和 RF 模式： EMI 测试界面“Setting”区域，输入测试频点、选择 TX 功率（TX 测试需要）、选择 RF 模式。

Step2 - TX 测试： TX 测试包括两种 RF 类型的信号，分别是单载波信号和连续的调制信号，可通过“TX Continue”区域的“Carrier”和“CarrierData”按钮来分别设置。另外，连续的调制信号是支持跳频模式的，可通过勾选“Hop”来设置。

Step3 - RX 测试： 通过“RX”区域的“RX Test”按钮来控制 DUT 进入 RX 测试模式。

4.5.2.4 非信令测试

Step1 - 配置频点、功率和 RF 模式： 非信令测试界面“Setting”区域，输入测试频点、选择 TX 功率（TX 测试需要）、选择 RF 模式。

Step2 - TX 测试： 非信令测试界面“TX Burst”区域，通过“Pack. Count”来选择连续的发包个数（仅支持单次发送 1000 个包数据和不限个数两种模式），通过分别点击“PRBS9”、“0x55”以及“0x0F”来选择发送不同类型的数据包。

Step3 - RX 测试： 通过“RX”区域的“RX Test”按钮来控制 DUT 进入 RX 测试模式，通过点击“RX Count”按钮来查看当前的收包数，通过点击“RSSI”按钮来查看 RF 信号源的功率。

4.5.2.5 RF 电流测试

Step1 - 下载 RF 电流测试所需的固件： RF 电流测试界面“Current Test Firmware”区域，配置 MCU 内部供电方式，点击“Download”按钮来下载电流测试的固件（每次开始时需要对 DUT 进行下电上电操作）。

Step2 - 进行 TX 电流测试： RF 电流测试界面“Current Test Operation”区域，配置供电方式和功率以及频点，通过点击“TX Current”按钮来进入 TX 电流测试。

Step3 - 进行 RX 电流测试： RF 电流测试界面“Current Test Operation”区域，配置频点，通过点击“RX Current”按钮来进入 RX 电流测试。

4.6 PA Configure Tool

4.6.1 功能描述（PA Configure Tool）

该工具主要用来配置生成 PA 工作所需的 Binary 文件。

4.6.2 界面组成及使用说明（PA Configure Tool）

该工具的界面如下图所示。

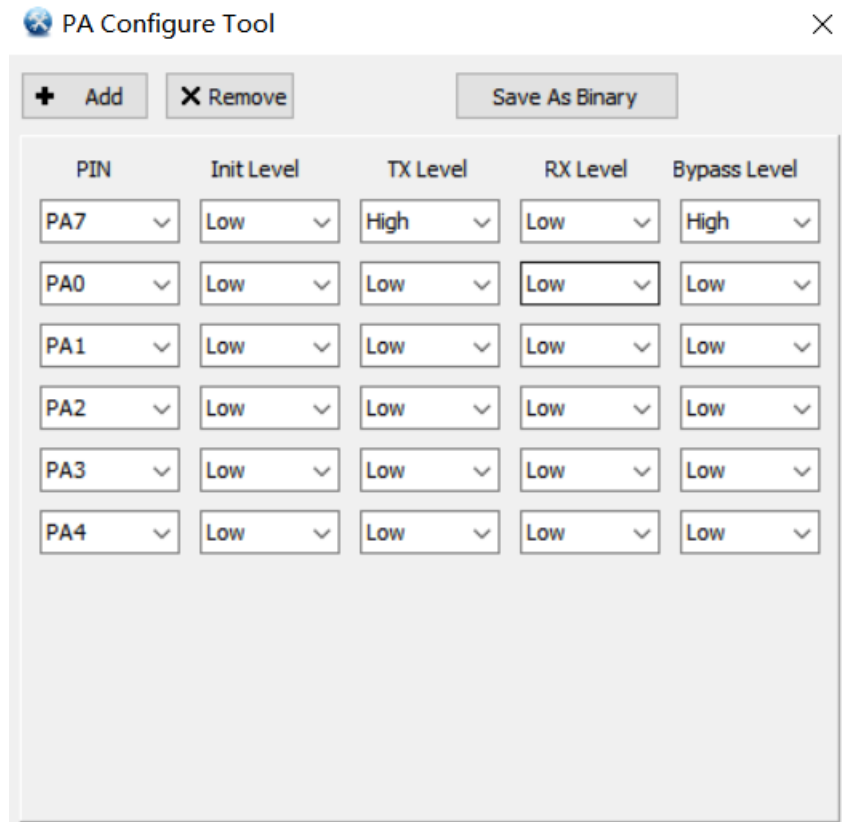


Figure 4.27: PA Configure Tool 界面

4.6.2.1 配置生成 PA 工作文件

Step1 - 添加 PA 控制的 pin 脚： 通过点击“Add”按钮来添加 PA 工作所需的 pin 脚。

Step2 - 配置各 pin 脚在不同工作模式下的电平： 分别配置“Init”、“TX”、“RX”以及“Bypass”模式下各 pin 脚的电平。

Step3 - 生成 Binary 文件： 通过点击“Save As Binary”按钮来将配置保存成 binary 文件。

4.7 EVK Offline Tool

4.7.1 功能描述 (EVK Offline Tool)

该工具用来配合带有离线下载功能的 BurningEVK 使用，主要功能包括：

- 更新 BurningEVK 中工具固件。
- 下载用户程序到 BurningEVK 中。

4.7.2 界面组成及使用说明 (EVK Offline Tool)

该工具的界面包括工具固件界面和用户工具界面，具体如下图所示。

EVK Offline Tool

Tool Bin User Bin

Q List all Update

NO.	File Name	Chip Type	Bin Type	ID	Bin Size	CRC32(Hex)	Download Address(Hex)
1	dut_8366_flash_v0001.bin	01	1	1	3660	2e6657bc	00008000
2	dut_8368_flash_v0246.bin	01	1	2	3588	aedd3c59	00008000
3	dut_8231_flash_j_v0002.bin	11	1	3	3884	fc2a03dd	00008000
4	dut_8231_flash_e_v0002.bin	11	1	4	3884	e11342a3	00008000
5	dut_8231_flash_j_v0002.bin	11	1	5	3884	fc2a03dd	00008000
6	dut_8231_flash_e_v0002.bin	11	1	6	3884	e11342a3	00008000
7	dut_8232_flash_v0001.bin	11	1	7	8584	ea334f61	00008000
8	dut_8266_flash_v0245.bin	12	1	8	5740	ce666a64	00008000
9	dut_8267_flash_v0001.bin	12	1	9	10112	2e14ef8c	00008000
10	dut_8267_flash_v0001.bin	12	1	10	10112	2e14ef8c	00008000
11	dut_b80_flash_v0002.bin	21	1	11	6964	6aa1561c	00040000
12	dut_b85_flash_v0001.bin	22	1	12	11944	7c5ccd74	00040000
13	dut_b87_flash_v0002.bin	23	1	13	12120	b377fda4	00040000

Figure 4.28: EVK Offline Tool 工具固件界面

EVK Offline Tool

User Bin

Add Remove Download

	Chip Name	Destination	Addr(Hex)	Bin Name	File Path	File Size	CRC(Hex)
☒	B91	Flash	000000	91BQB	C:\Users\Admin\Desktop\BDT\config\bin\bqb\B91_BQB	22476	cd2f8a68
☒	B80	Flash	000000	80BQB	C:\Users\Admin\Desktop\BDT\config\bin\bqb\B80_BQB	10340	18a2176c
☒	B85	Flash	000000	85BQB	C:\Users\Admin\Desktop\BDT\config\bin\bqb\B85_BQB	14184	cca40cf0
☒	B87	Flash	000000	87BQB	C:\Users\Admin\Desktop\BDT\config\bin\bqb\B87_BQB	15328	f7dfad59

Figure 4.29: EVK Offline Tool 用户固件界面

4.7.2.1 更新工具固件

Step1 - 查看 BDT 中的工具固件：通过点击“List all”按钮来查看现有的工具固件。

Step2 - 更新固件：通过点击“Update”按钮来更新工具固件到 BurningEVK 中。

注意：

工具固件界面默认是隐藏的，需要通过修改配置文件来显示。

4.7.2.2 下载用户固件

Step1 - 添加用户固件：通过点击“Add”按钮来添加用户固件，并配置好“Chip Name”、“Bin Name”、“Destination”、“Addr”以及“File Path”等信息。

Step2 - 下载用户固件：先将需要下载到 BurningEVK 中的固件勾选上，接着点击“Download”按钮来完成下载。

4.8 Secure Boot Tool

该工具主要功能如下：

- 支持芯片配置信息以及固件下载。
- 支持重新使能 Debug 接口功能。
- 支持读取芯片配置信息。

4.8.1 界面组成及使用说明

Secure Boot 工具界面如下图所示。

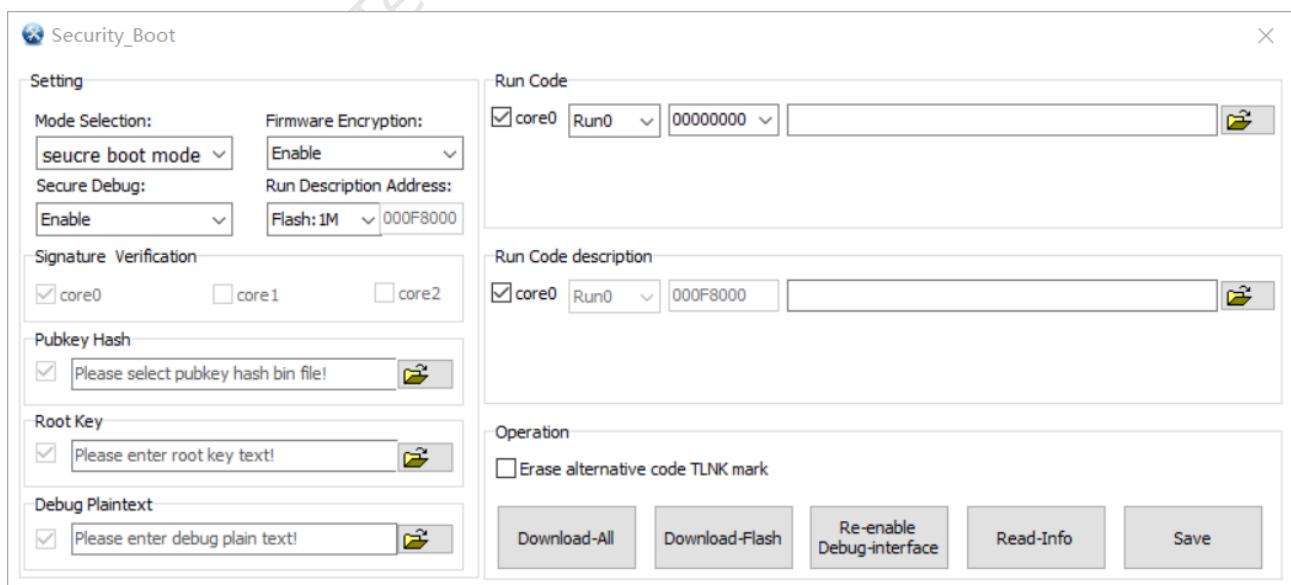


Figure 4.30: Secure Boot 工具界面

4.8.1.1 Mode Selection

选择“secure boot mode”或“normal mode”任意一种模式。

4.8.1.2 Firmware Encryption

选择启用或禁用“Firmware Encryption”功能。

4.8.1.3 Secure Debug

选择启用或禁用“Secure Debug”功能。

4.8.1.4 Run Description Address

选择 Flash 容量，相应的代码描述地址将自动确认。

注意：

此功能仅在“secure boot mode”模式下使用。

4.8.1.5 Pubkey Hash

选择待烧录的“public_key_hash.bin”文件。单击“Download-All”按钮后，程序会自动检查文件长度是否满足条件。若不满足条件，会进行报错提醒。

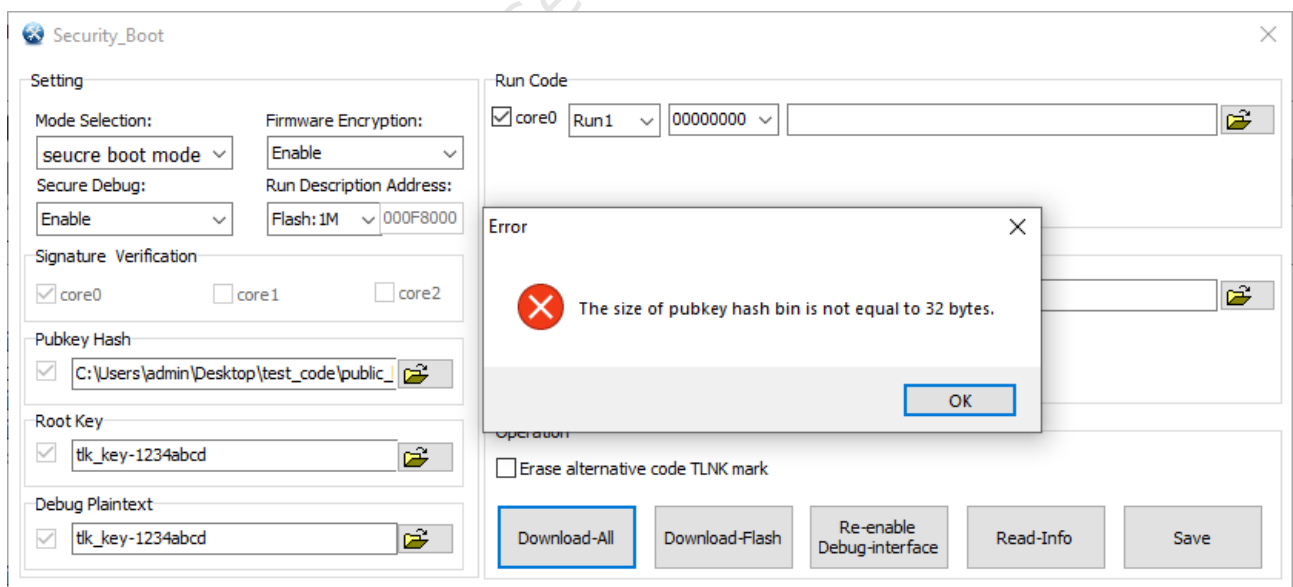


Figure 4.31: 公钥哈希文件错误

注意：

此功能仅在“secure boot mode”模式下使用。

4.8.1.6 Root Key

输入框输入 16 个字符的字符串或 32 个十六进制字符的字符串作为根密钥，同时也支持选择文本文件导入密钥。

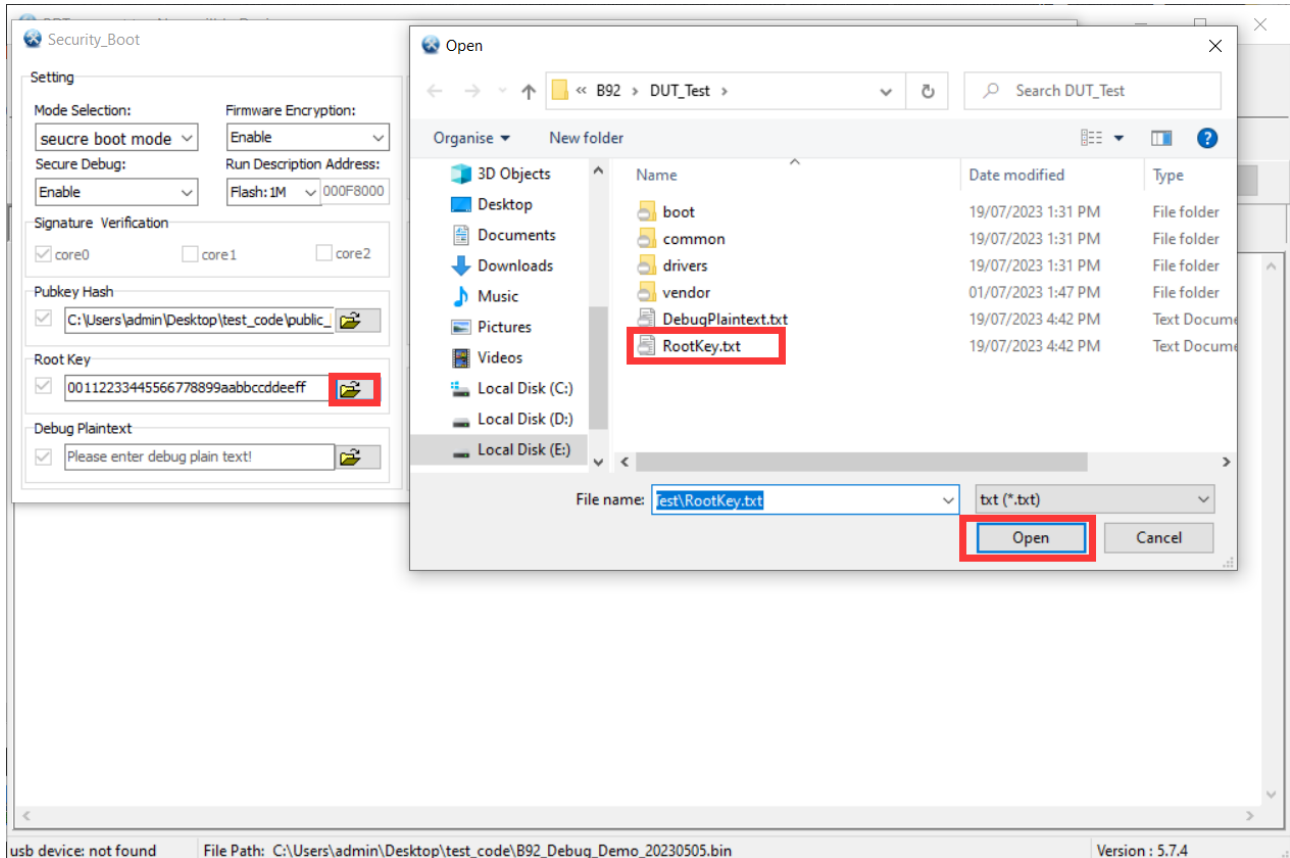


Figure 4.32: 导入 Root Key 文本

输入 32 个十六进制字符的字符串示例:00112233445566778899aabbccddeeff 或 00112233445566778899AAB-BCCDDEEFF。

输入 16 个字符的字符串示例: tlk_key-1234abcd。

单击“Download-All”按钮后，程序会自动检查输入的字符串是否满足条件。若不满足条件，会进行报错提醒。

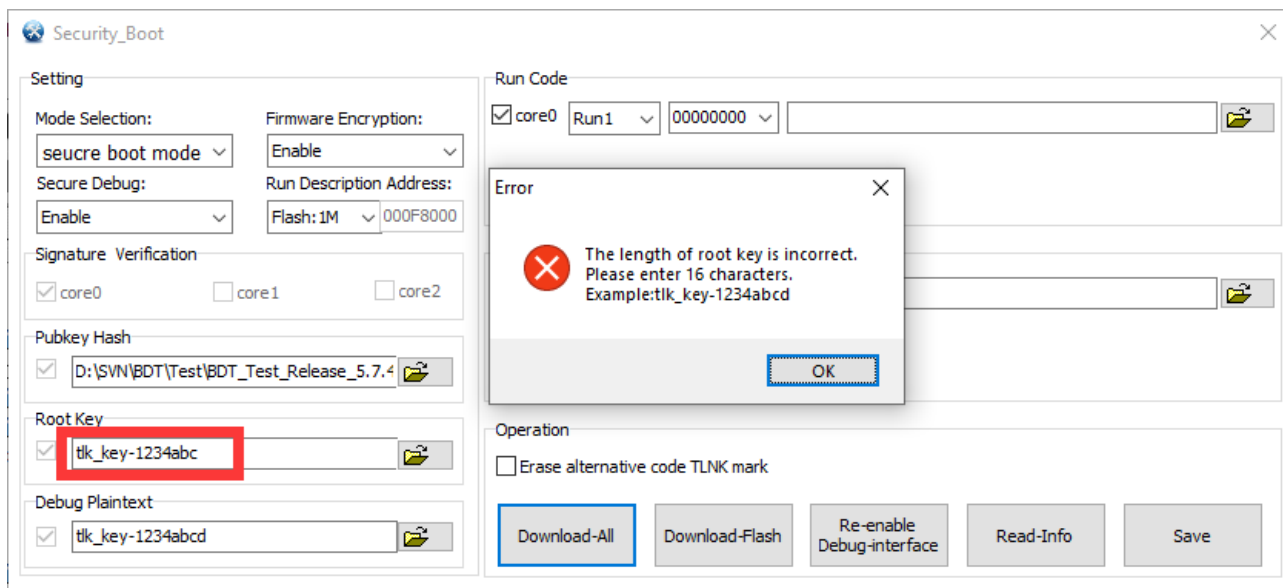


Figure 4.33: Root Key 格式错误 1

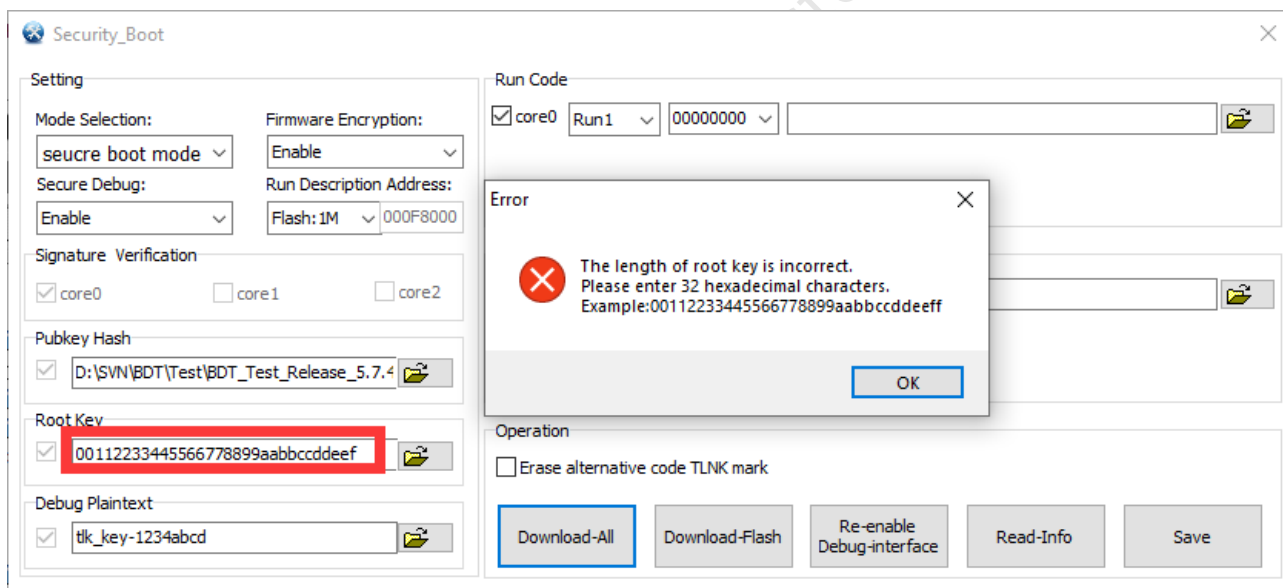


Figure 4.34: Root Key 格式错误 2

注意：

Firmware Encryption 或 Secure Debug 功能启用时才需设置 Root Key。

4.8.1.7 Debug Plaintext

输入 16 个字符的字符串或 32 个十六进制字符的字符串作为根密钥，同时也支持选择文本文件导入密钥。

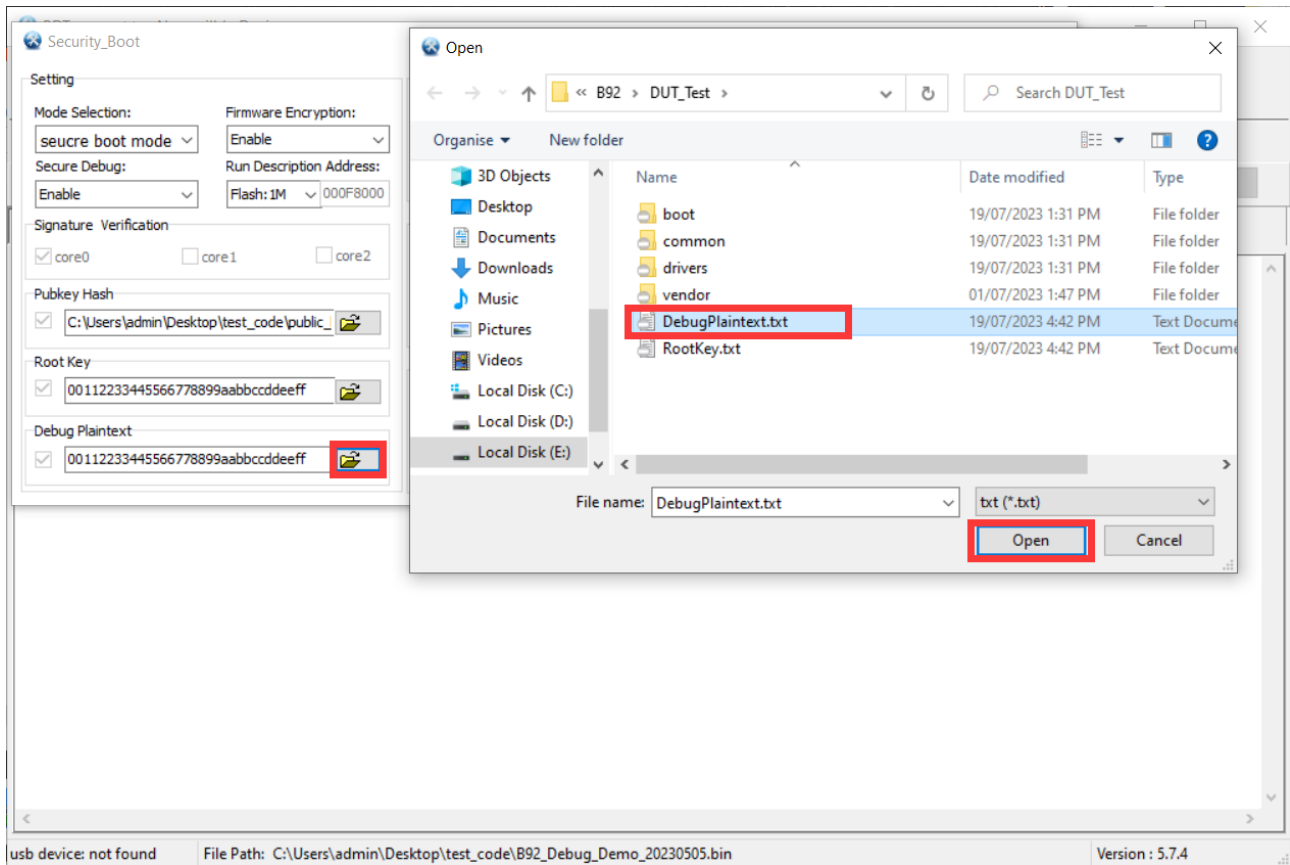


Figure 4.35: 导入 Debug Plaintext 文本

输入 32 个十六进制字符的字符串示例: 00112233445566778899aabbccddeeff 或 00112233445566778899AAB-BCCDDEEFF。

输入 16 个字符的字符串示例: tlk_key-1234abcd。

单击“Download-All”按钮后, 程序会自动检查输入的字符串是否满足条件。若不满足条件, 会进行报错提醒。

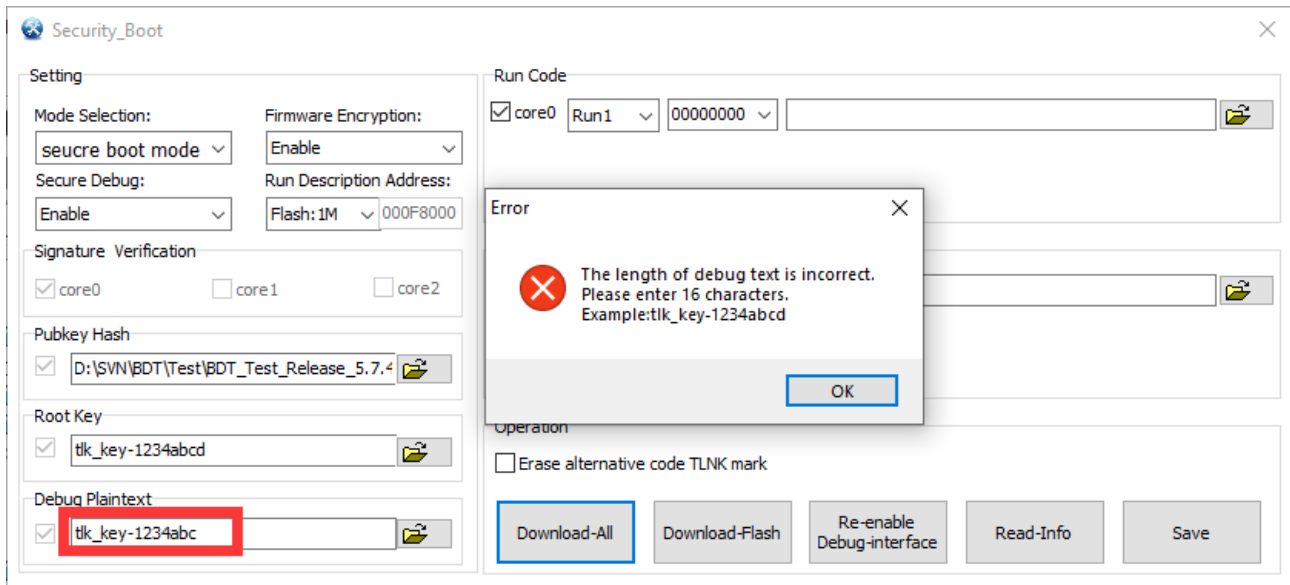


Figure 4.36: Debug Plaintext 格式错误 1

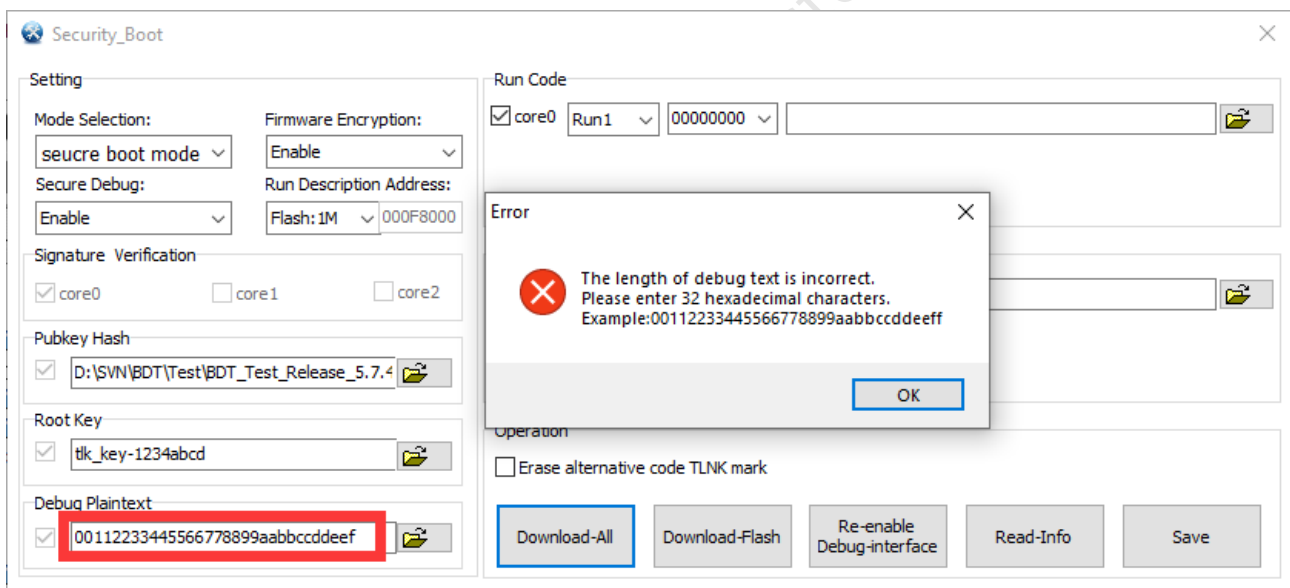


Figure 4.37: Debug Plaintext 格式错误 2

注意：

Secure Debug 功能启用时才需设置 Debug Plaintext。

4.8.1.8 Run Code

勾选复选框，选择运行代码 0 或代码 1 以及运行代码地址，并导入固件路径。

4.8.1.9 Run Code description

勾选复选框，选择运行代码描述文件路径并导入。

注意：

此功能仅在“secure boot mode”模式下使用。

4.8.1.10 Download-All

“Setting”区域芯片配置信息下载到 eFuse 中。“Run Code”以及“Run Code description”区域选择的文件下载到 Flash 中。

注意：

Secure Debug 功能启用，点击“Download-All”按钮，无论下载是否成功都会在工具 config\sboot\key 路径下产生 Debug Key 文件。

4.8.1.11 Download-Flash

“Run Code”以及“Run Code description”区域选择的文件下载到 Flash 中。

注意：

已烧录过 eFuse，只需要更新固件程序，点击 Download-Flash 按钮即可。

4.8.1.12 Re-enable Debug-interface

“Secure Debug”功能启用后，芯片 Debug 功能会被禁用。点击 “Re-enable Debug-interface” 按钮会弹出对话框。选择 Yes，使用默认文件夹中的 DebugKey.txt 文件作为 Debug Key 去恢复 Debug 功能。选择 No，需要手动导入文件作为 Debug Key 去恢复 Debug 功能。

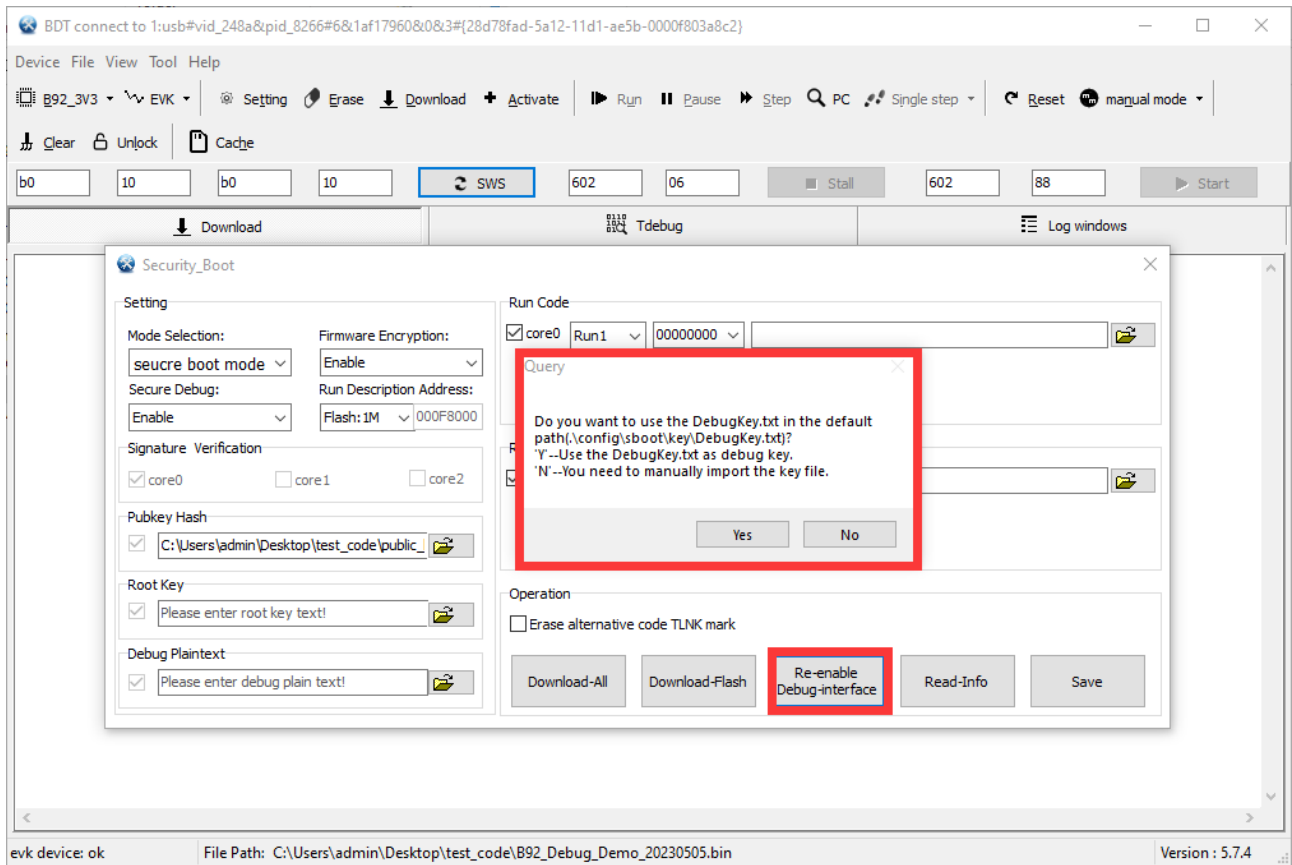


Figure 4.38: 重新使能调试接口

4.8.1.13 Read-Info

读取芯片 eFuse 中配置信息，不包括 Root Key 以及 Debug Plaintext 中的信息。

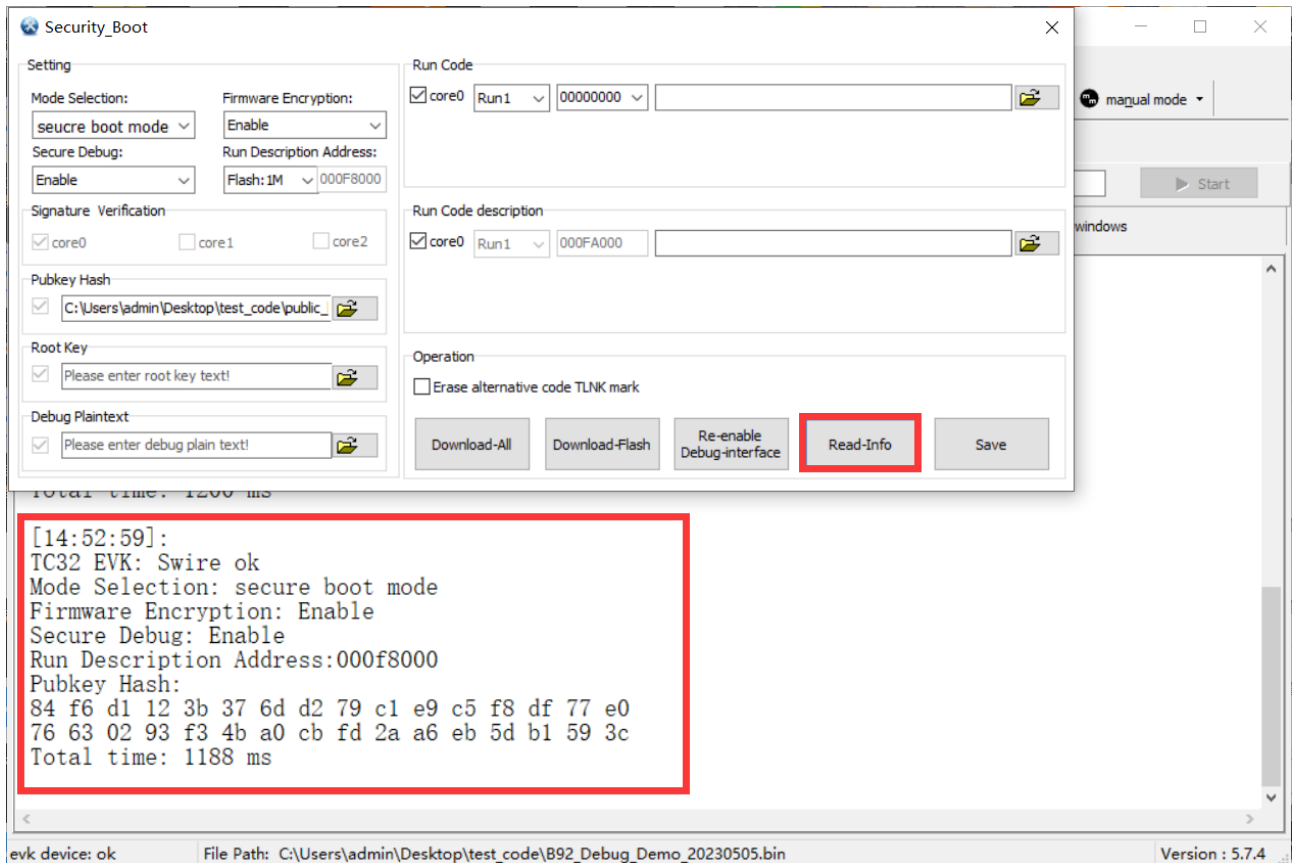


Figure 4.39: 读取信息

4.8.1.14 Save

保存界面配置信息，将 Root Key 以及 Debug Plaintext 中的信息保存到 config\sboot\key 路径。



Figure 4.40: key 文件夹路径

4.8.2 常用安全机制组合和工具的使用步骤说明

Secure Boot、Firmware Encryption 以及 Secure Debug 安全机制可单独使用并且可以自由组合。然而，在实践中建议使用以下组合：

- Option 0: Secure Boot + Secure Debug
- Option 1: Firmware Encryption + Secure Debug
- Option 2: Secure Boot + Firmware Encryption + Secure Debug

4.8.2.1 Secure Boot + Secure Debug

Step1: “Mode Selection” 下拉框中选择“secure boot mode”。

Step2: “Firmware Encryption” 下拉框中选择“Disable”。

Step3: “Secure Debug” 下拉框中选择“Enable”。

Step4: 根据芯片封装选择对应 Flash 容量。此处默认选择“Flash: 1M”。

Step5: “Run Code”区域选择运行代码 0 或代码 1 以及运行代码地址，并导入固件路径。

Step6: 上一步骤完成后，“public_key_hash.bin”文件路径会自动导入到“Pubkey Hash”输入框中，若未成功，则手动导入。

Step7: “Run Code description”区域导入运行代码描述文件路径。

Step8: 在“Root Key”以及“Debug Plaintext”输入框中输入密钥。

Step9: 点击“Download-ALL”按钮完成下载。

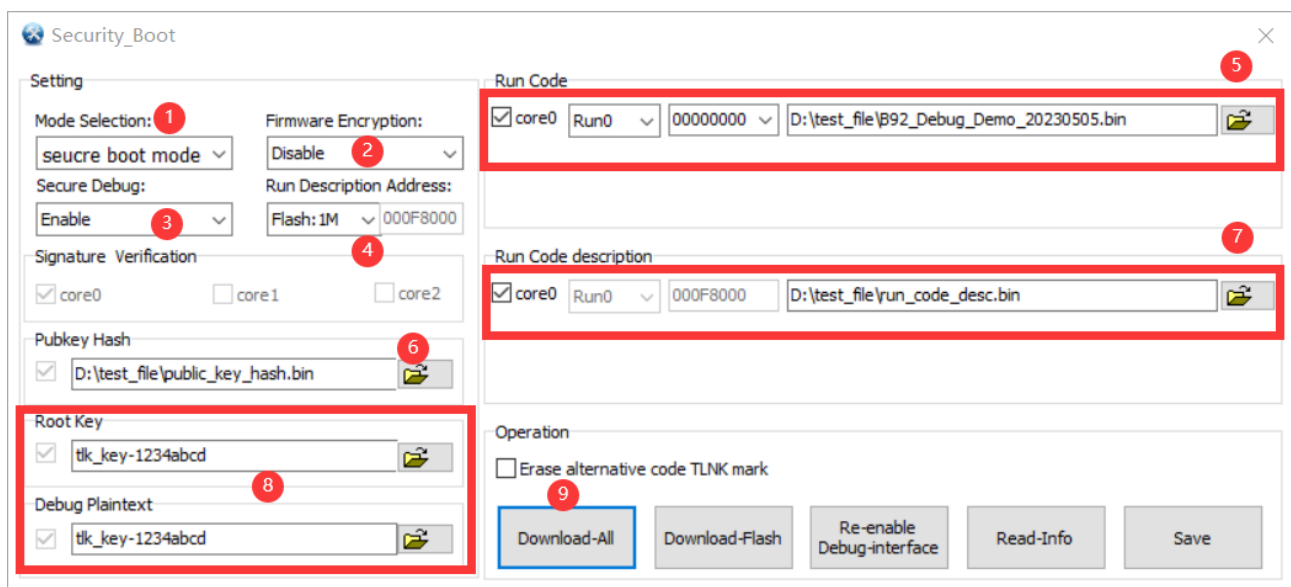


Figure 4.41: Secure Boot + Secure Debug 烧录步骤

4.8.2.2 Firmware Encryption + Secure Debug

Step1: “Mode Selection” 下拉框中选择“normal mode”。

Step2: “Firmware Encryption” 下拉框中选择“Enable”。

Step3: “Secure Debug” 下拉框中选择“Enable”。

Step4: 选择固件下载地址并导入固件路径。

Step5: 在“Root Key”以及“Debug Plaintext”输入框中输入密钥。

Step6: 点击“Download-ALL”按钮完成下载。

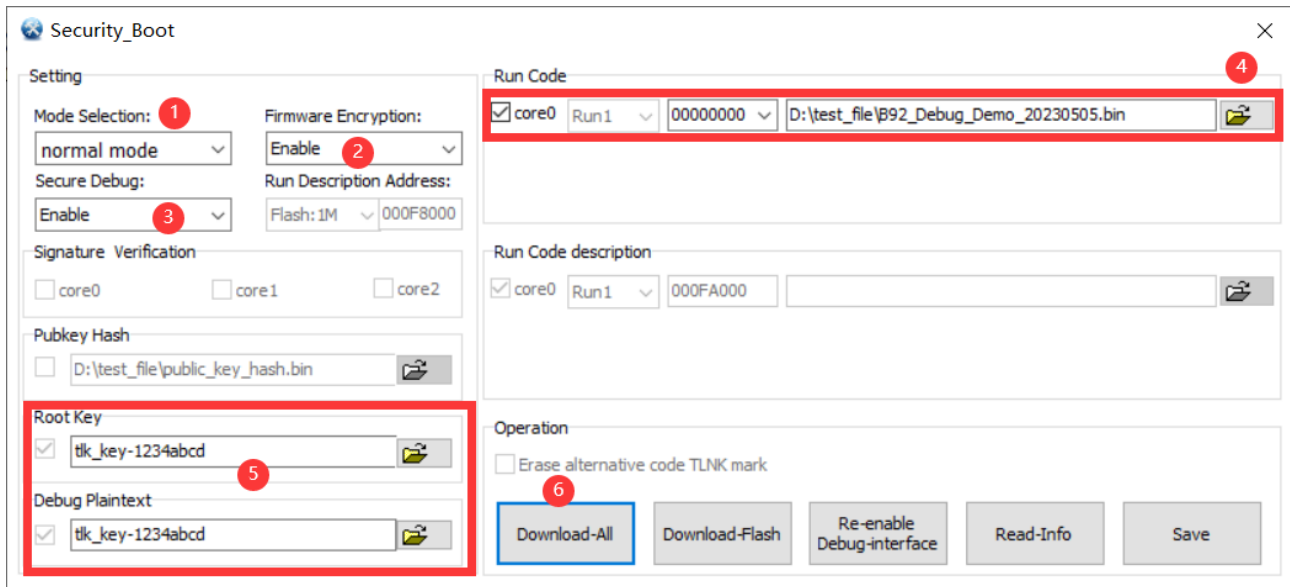


Figure 4.42: Firmware Encryption + Secure Debug 烧录步骤

4.8.3 Secure Boot + Firmware Encryption + Secure Debug

- Step1:** “Mode Selection” 下拉框中选择“secure boot mode”。
- Step2:** “Firmware Encryption” 下拉框中选择“Enable”。
- Step3:** “Secure Debug” 下拉框中选择“Enable”。
- Step4:** 根据芯片封装选择对应 Flash 容量。此处默认选择“Flash: 1M”。
- Step5:** “Run Code”区域选择运行代码 0 或代码 1 以及运行代码地址，并导入固件路径。
- Step6:** 上一步骤完成后，“public_key_hash.bin”文件路径会自动导入到“Pubkey Hash”输入框中，若未成功，则手动导入。
- Step7:** “Run Code description”区域导入运行代码描述文件路径。
- Step8:** 在“Root Key”以及“Debug Plaintext”输入框中输入密钥。
- Step9:** 点击“Download-ALL”按钮完成下载。

Security_Boot

Setting

Mode Selection: 1
seucre boot mode

Secure Debug: 3
Enable

Signature Verification 4
☒ core0 ☐ core1 ☐ core2

Pubkey Hash 6
☒ D:\test_file\public_key_hash.bin

Root Key 8
☒ tlk_key-1234abcd

Debug Plaintext
☒ sun_key-1234abcd

Firmware Encryption: 2
Enable

Run Description Address:
Flash: 1M 4 000F8000

Run Code 5
☒ core0 Run0 00000000 D:\test_file\B92_Debug_Demo_20230505.bin

Run Code description 7
☒ core0 Run0 000F8000 D:\test_file\run_code_desc.bin

Operation 9
☐ Erase alternative code TLNK mark

Download-All Download-Flash Re-enable Debug-interface Read-Info Save

Figure 4.43: Secure Boot + Firmware Encryption + Secure Debug 烧录步骤

5 功能支持列表

并非所有 MCU 类型都支持此工具的所有功能，请参考下表。

MCU Type ITEMS	8266			8267/8269			8366/8368			8232/8233	8258		
	USB	SWIRE	SWIRE2USB	USB	SWIRE	SWIRE2USB	USB	SWIRE	SWIRE2USB	SWIRE	USB	SWIRE	SWIRE2USB
Download(Flash/SRAM)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Download(OTP)	×	×	×	×	×	×	✓	✓	✓	×	×	×	×
Erase FLASH	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Activate	×	✓	×	×	✓	×	×	✓	×	✓	×	✓	×
PC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Run/pause/step	×	×	×	✓	✓	✓	×	×	×	✓	✓	✓	✓
Memory access (OTP)	×	×	×	×	×	×	✓	✓	✓	×	×	×	×
Memory access (FLASH/CORE/ANALOG)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 5.1: 功能支持列表