

版本	V1.0
日期	202011

# APT32F102 系列 BOOTLOADER 使用指南



版权所有©深圳市爱普特微电子有限公司

本资料内容为深圳市爱普特微电子有限公司在现有数据资料基础上慎重且力求准确无误编制而成，本资料中所记载的实例以正确的使用方法和标准操作为前提，使用方在应用该等实例时请充分考虑外部诸条件，深圳市爱普特微电子有限公司不担保或确认该等实例在使用方的适用性、适当性或完整性，深圳市爱普特微电子有限公司亦不对使用方因使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。基于使本资料的内容更加完善等原因，公司保留未经预告的修改权。

Revision History			
版本	日期	描述	作者
V1.0	2020-11	为APT32F102系列新建使用说明	郑攀，魏柠柠

目录

1. 概述.....	1
2. 实现原理.....	1
3. 示例 WORKSPACE 说明.....	2
4. 更新应用代码.....	3
4.1 硬件连接 .....	4
4.2 工具和代码的适配 .....	4
4.3 更新.....	6
4.4 正常运行 .....	6

## 1. 概述

本文主要说明适用于 APT32F102x 系列的 **bootloader** 的实现原理和应用代码更新方法。

## 2. 实现原理

一个有**bootloader**的芯片内部代码存储空间如下：

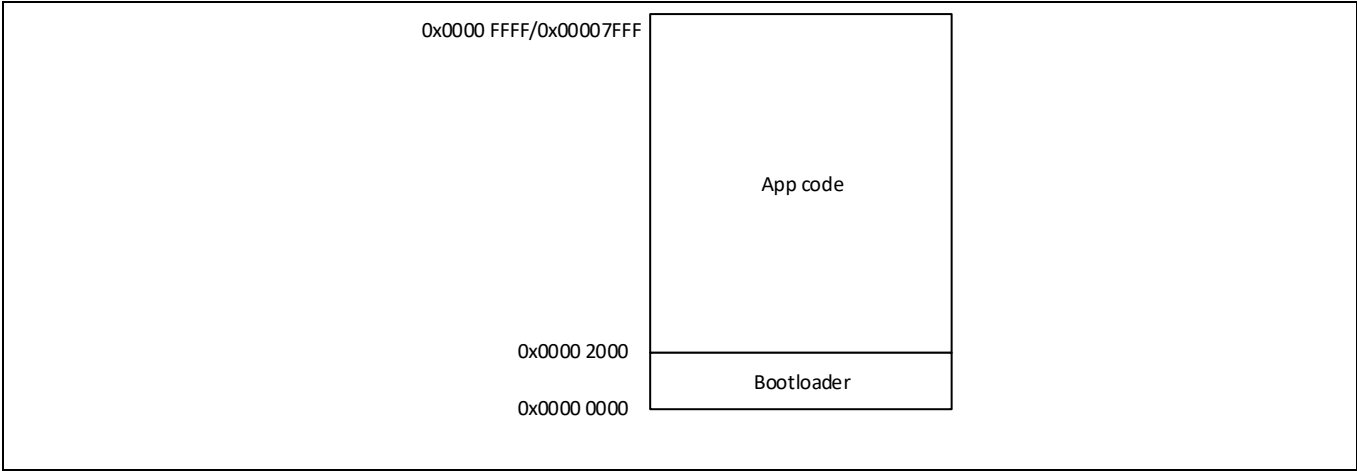


图1 芯片内部代码分布

有以下几点事实：

- A、汇编指令**jmp**，可跳转到**4GB**存储器空间的任意位置；
- B、CDK开发套件可通过修改链接文件**ckcpu.ld**，实现更改应用代码的起始地址；

结合以上两点，合理组织好启动代码，就可以实现**bootloader**的功能。芯片启动流程如下：

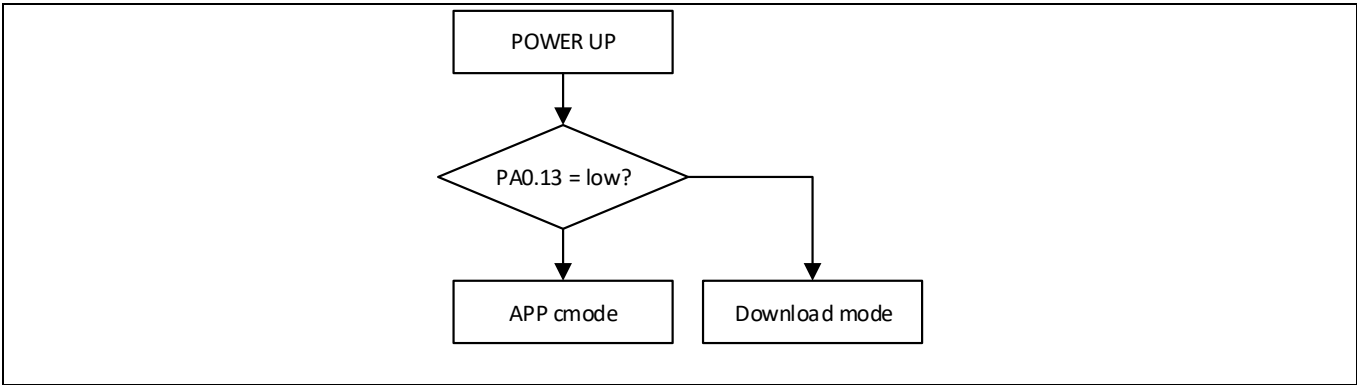


图2 bootloader基本流程

### 3. 示例workspace说明

示例workspace下面有两个工程，bootloader即芯片上电时开始运行的代码，app是用户代码。

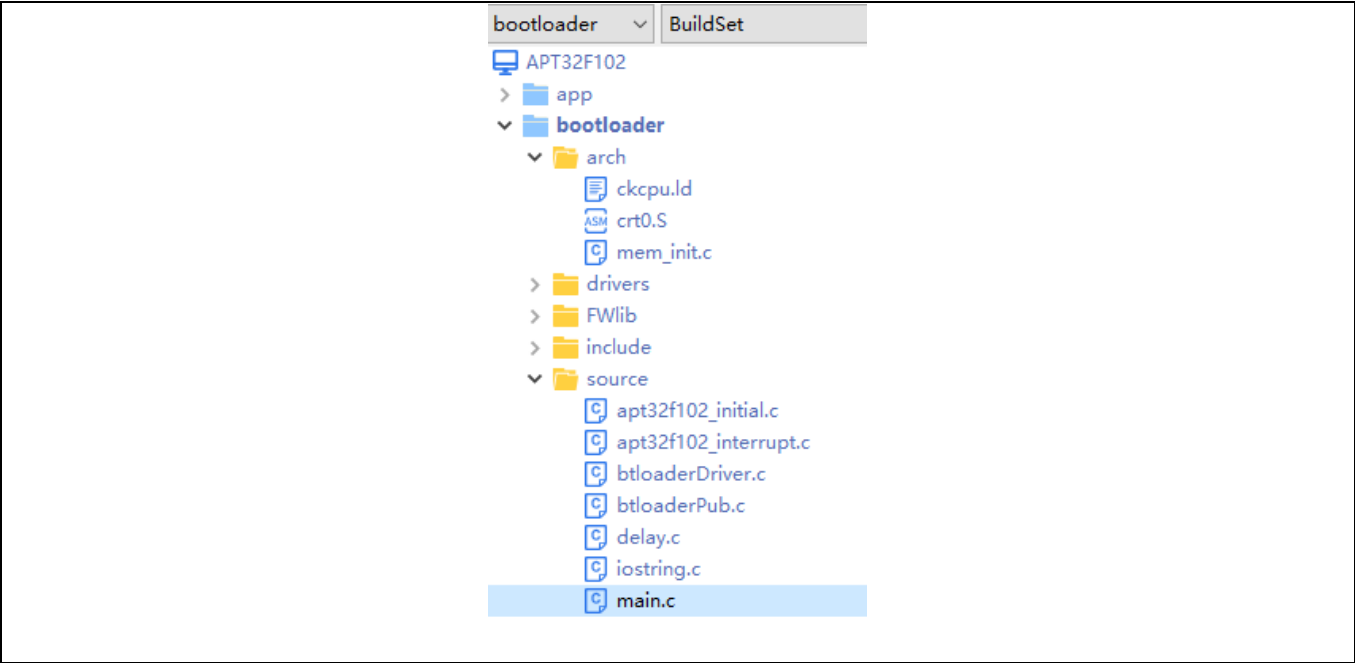


图3 示例workspace结构

- 示例bootloader的大小控制在0x2000以内，所以app工程的link文件（ckcpu.ld）中，ROM起始地址可以设在0x00002000。同时bootloader.h中 APP\_START\_ADDR = 0x2000。注意两边的一致性。

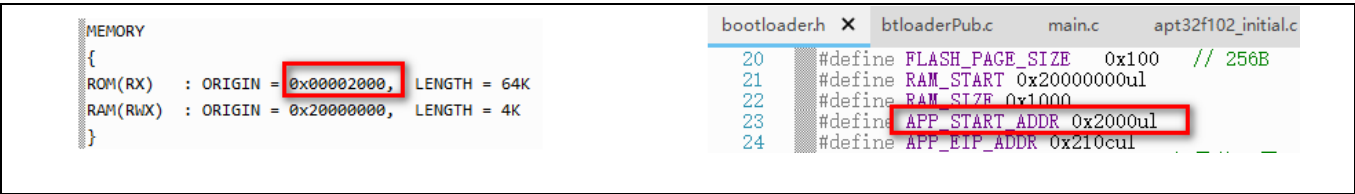


图4 app工程中link文件修改内容和bootloader工程中app起始地址的匹配

- 根据app的hex文件，修改bootloader.h中的跳转地址（APP\_EIP\_ADDR）。如下图所示：

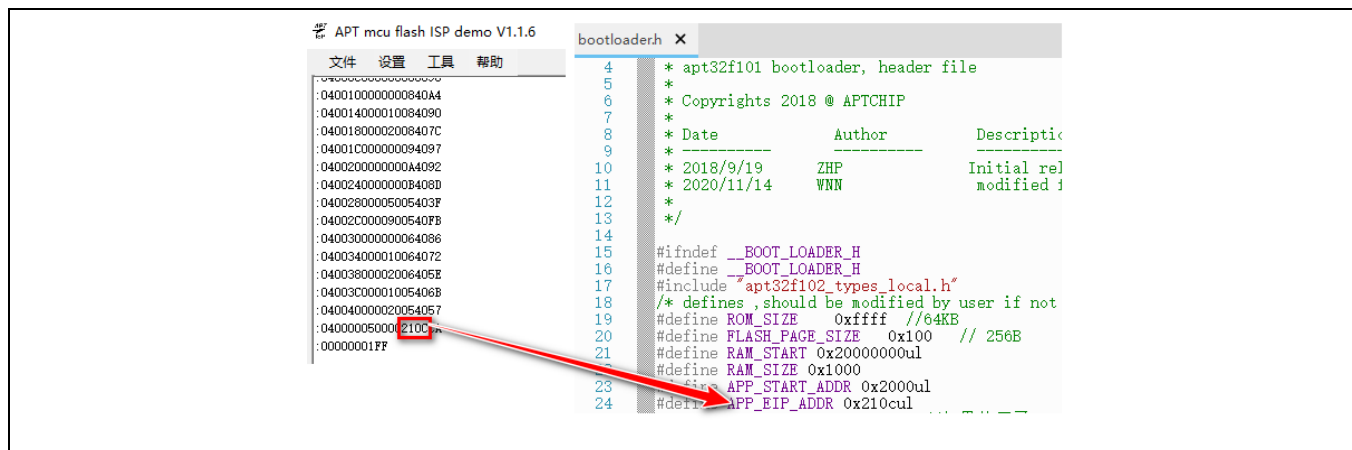


图5 跳转地址

- 根据实际电路可能要调整UART接收口。示例bootloader中使用了PA0.13（RX和状态判断脚）和PB0.0（TX）作为和上位机通信的端口。如需调整，注意bootloader.h中如下代码：

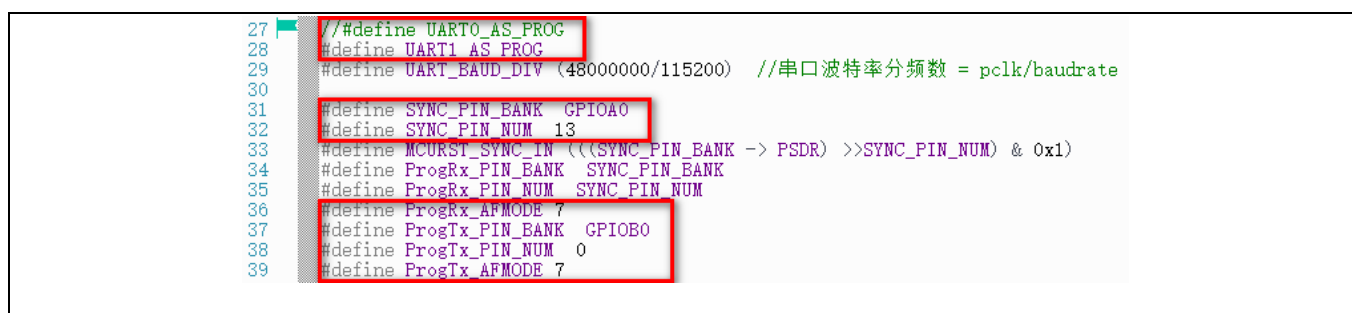


图6 硬件连接不同时需要调整的内容

注：这里SYNC和RX使用了同一个管脚。

## 4. 更新应用代码

使用ISPdemoV1\_1\_6.exe完成应用代码的更新步骤如下：

4.1 硬件连接

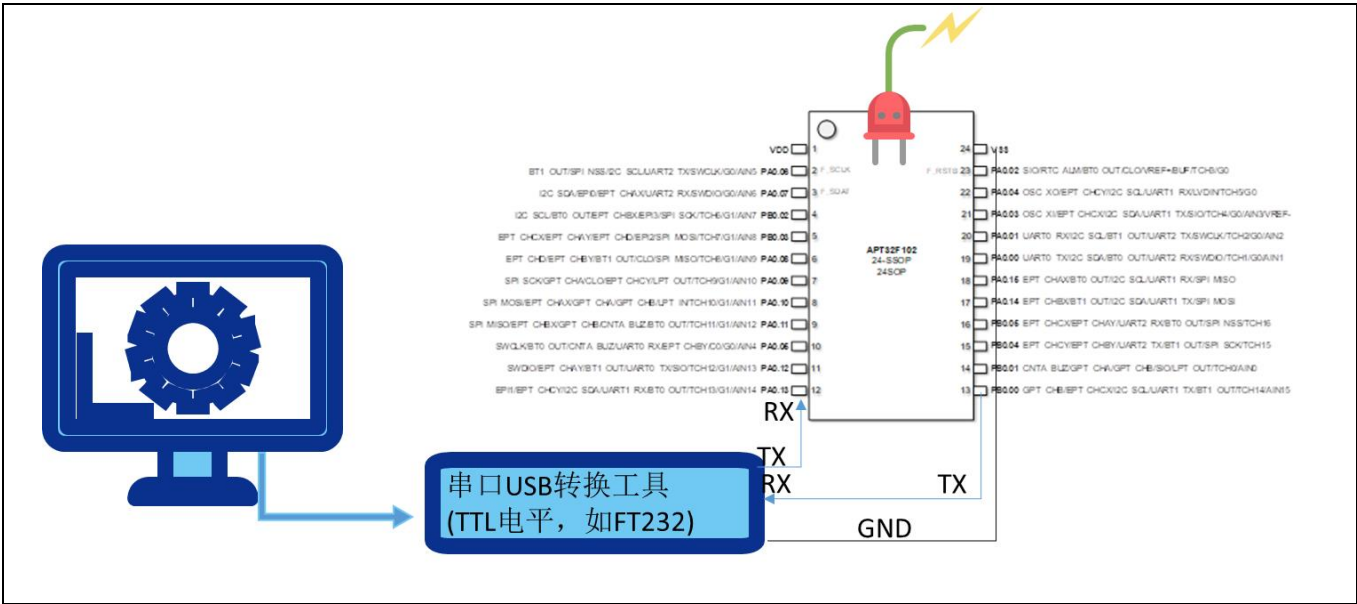


图7 硬件连接示意图

4.2 工具和代码的适配

1) 串口设置。请将串口设置成下图所示（端口要根据实际连接确定）。

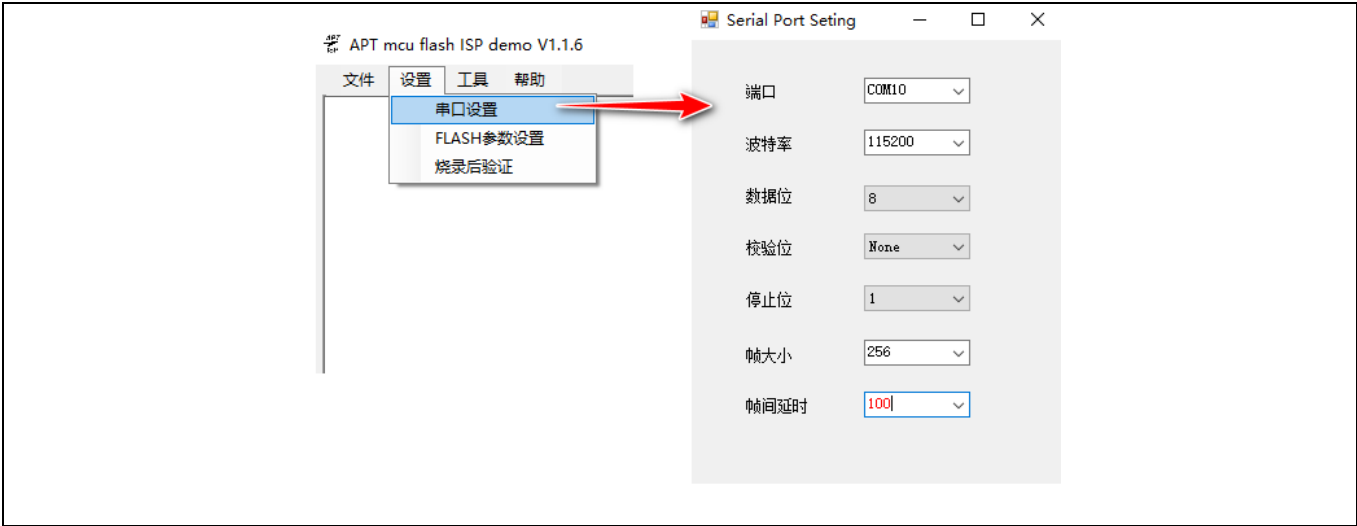


图8 串口设置

2) FLASH参数设置。APT32F102系列的flash擦除后为全零。

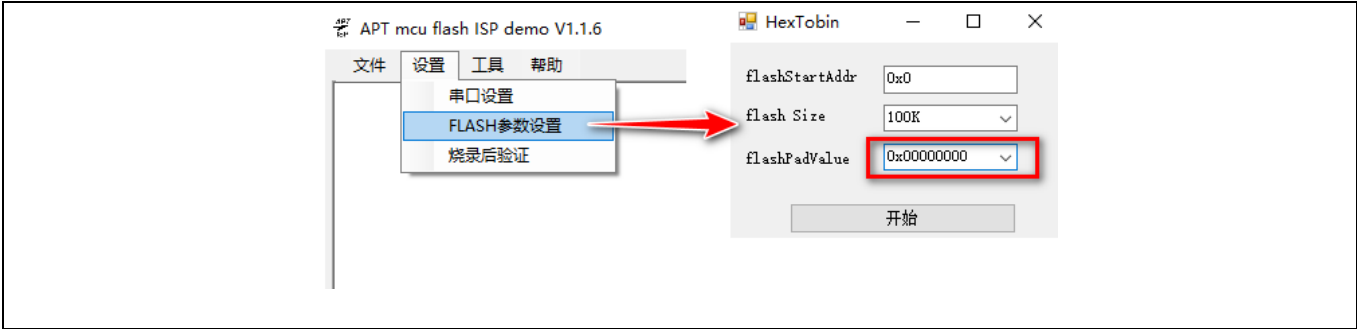


图9 FLASH参数设置

- 3) 烧录后验证。如果不勾选，工具端不再检查bootloader回传的校验码，但bootloader端仍然会做校验。如果校验不通过直接擦除。



图 10 工具端烧录验证选项

- 4) 装载app ihex文件。注意检查代码起始地址。

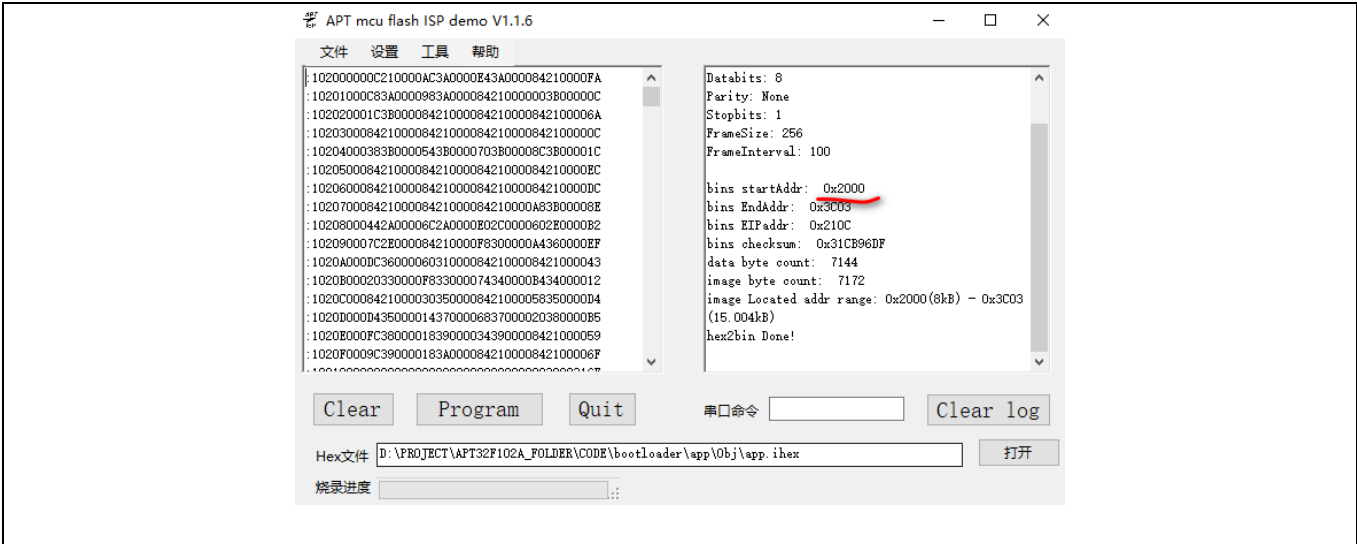


图 11 装载 app hex 文件

### 4.3 更新

点Program（拉低串口电平，避免串口给芯片供电）→芯片断电→芯片上电→点Continue

烧录成功后会显示：

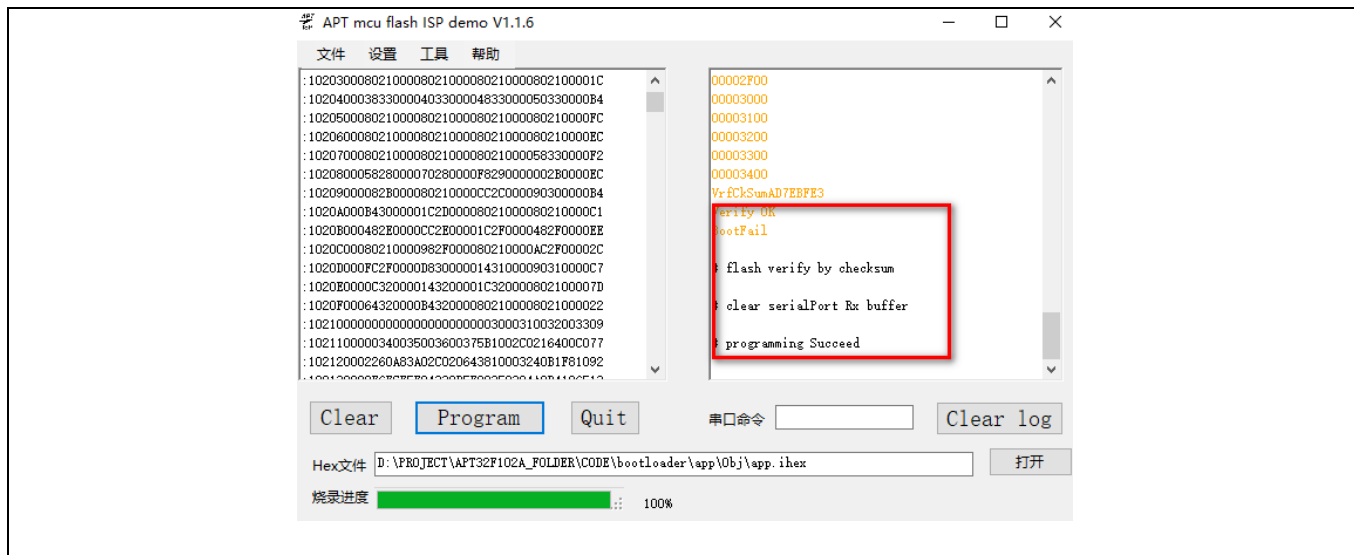


图 12 app 代码更新成功

### 4.4 正常运行

因为程序在上电时会判断 PA0.13 的电平，如果低电平会进入接收代码的流程，只有保证该引脚在上电初期是高电平，才能正常跳转到 app 用户代码区。