

1. 概述

基站侧启动包括两部分：UC8088启动与UC8288启动，这两个CPU在上电时同时启动，通过SPI Master进行通信，UC8088作为主控设备，UC8288作为从设备，而UC8288是AP的基带部分，以下需要描述UC8288都将用基带进行描述，基站侧如图1所示：

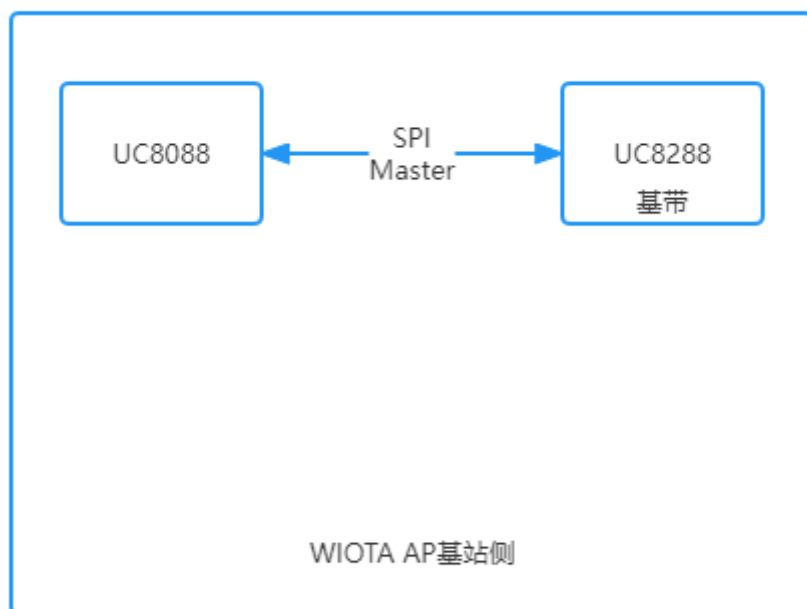


图1 WIOTA AP基站侧

2. 8088集成基带镜像

在编译[基站侧代码](#)时，会有个前置操作，读取bin目录下ap8288.bin文件生成头文件，存储到代码根目录下，文件名为flat_ap8288.h，文件内容如图2所示

```
1  #ifndef __AP8288_BIN_H__
2  #define __AP8288_BIN_H__ ①
3
4  const unsigned int ap8288_bin_timestamp = 1648869903;
5  const unsigned int ap8288_bin_len = 154605;
6  const unsigned char ap8288_bin[] = {
9671
9672  ① #endif // end of __AP8288_BIN_H__
9673
```

① 头文件卫士

镜像生成时间戳

镜像文件大小

图2 flat_ap8288.h文件

- ap8288_bin_timestamp: 基带镜像生成时间戳, 精确到秒
- ap8288_bin_len: 镜像文件大小, 单位为字节
- ap8288_bin: 镜像二进制内容

注: Linux下需要安装Python3.5以上版本, 并把python3加入到环境变量\$PATH中, 用以执行该前置操作的脚本。

3. 基带版本校验过程

WIOTA AP基站侧启动, 需要校验基带版本的时间戳信息, 校验过程如图3所示:

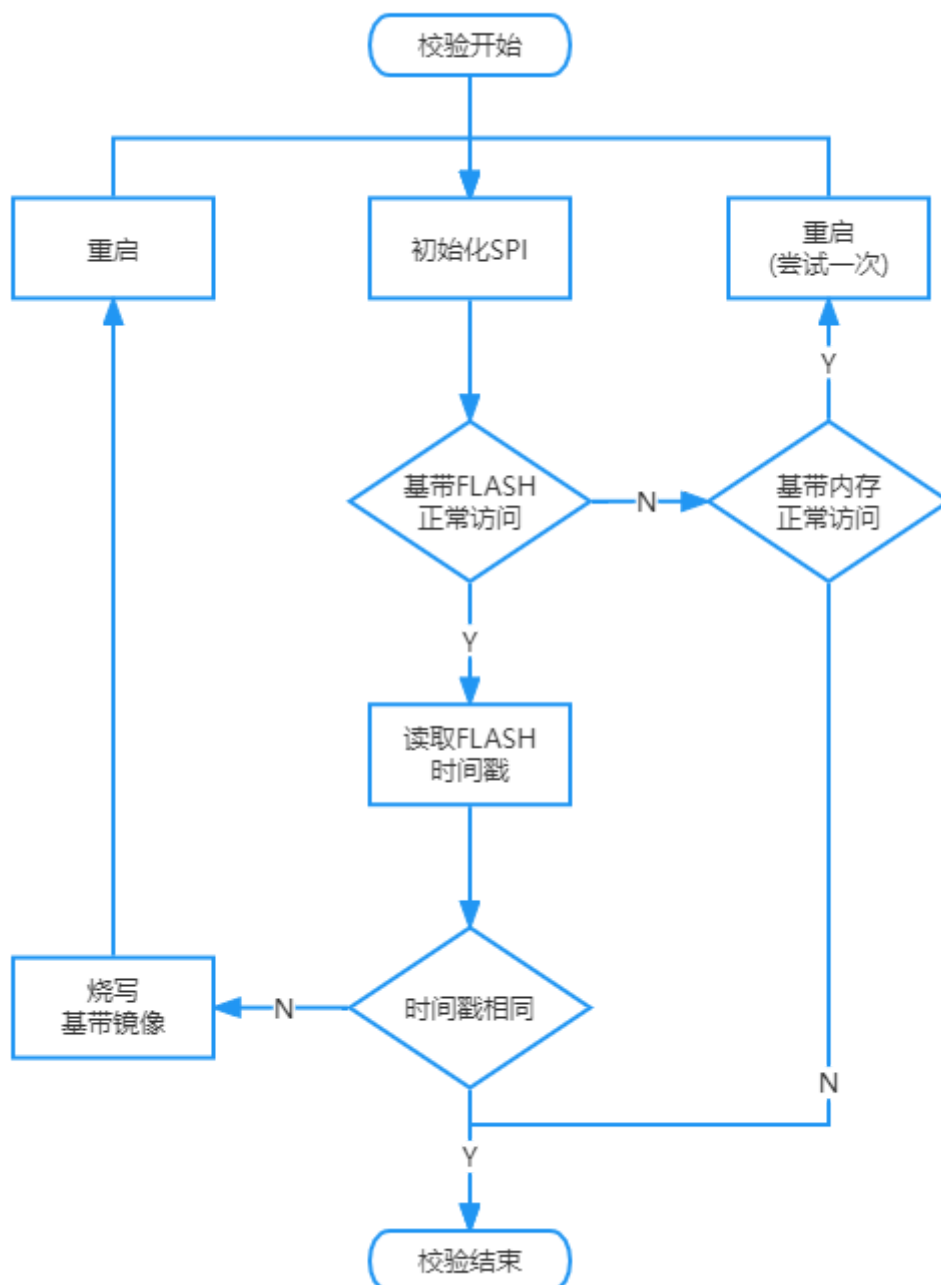


图3 基带校验过程

- 正常校验: 先初始化SPI相关寄存器配置, 尝试对基带的FLASH进行访问, 如果能访问, 并获取其时间戳, 与8088本地的时间戳进行校验, 相同则进入YMODEM选择或系统, 完成校验。
- 异常情况:

- **时间戳不同**，当发现本地时间戳（8088）与基带FLASH的时间戳不同时，会重新烧写基带的镜像，使本地保存镜像与基带的FLASH保持一致，烧写镜像后，基站侧系统会重启一次。
重新烧写时，会提示“reflash 8288”，烧写完成后，会提示“ready reboot”重启8088及基带。
- **基带的FLASH不能正常访问**，若基带的CPU能正常访问，系统会尝试一次对基带的重启，若重启后提示“Plase reboot 8288 manually !”，表示仍然无法正常访问基带的FLASH，需要手动对基带进行一次复位。

注：由于基带的时间戳是存储在FLASH的最后四个字节，若通过下载器单独烧写基带的FLASH时，不能整片擦除，也不能下载大于508KB的镜像，否则WIOTA AP基站侧启动时因为读不到正确的时间戳，导致基带镜像被重写。

4. AT指令

如果通过YMODEM单独下载基带镜像时发现镜像存在bug，则可通过“**AT+REFLASH**”指令，实现对基带的FLASH的镜像恢复。[详见](#)