

CH563 网络 IAP 升级方案

介绍

在实际生产中，为了不断产品优化，厂家常常遇到设备的固件程序情况，如果对用户手中的大量设备进行现场 ISP 升级，则需要大量人力与时间。

在网络飞速发展的时代，将终端设备接入到网络中可以有效地解决上述问题，厂家只需要将最新的固件放到指定的服务器上，设备通过网络远程访问服务器，自动下载更新固件程序。方便快捷，稳定可靠，尤其是在大批量升级中，优点显著。

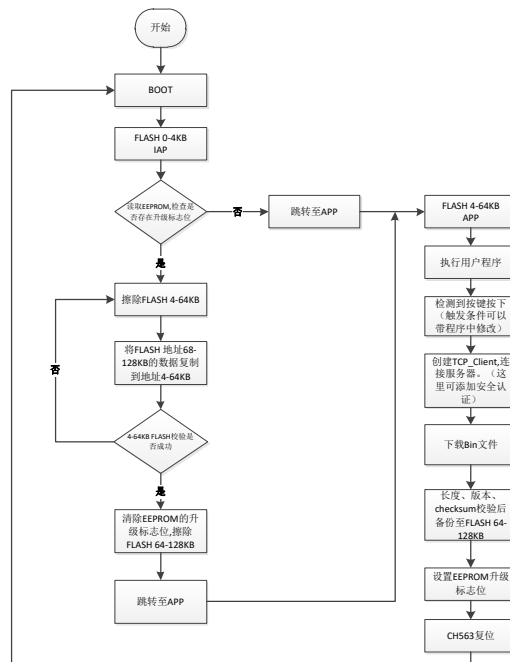
CH563 是一款带网络接口的单片机，可以广泛应用于各种嵌入式产品的开发，使用 CH563 可以很好的实现该方案：CH563 作为用户的终端设备，通过 TCP 协议访问厂家的服务器，服务器上保存有最新的固件，CH563 可以下载该固件，并进行升级。

方案简介

具体的操作流程可以看“CH563IAP 升级使用教程 V1.1”；

详细要点：

- CH563 FLASH 0-4KB 地址存放 IAP 代码，这部分固定存放，在升级过程中不会被擦除。
4-64KB 地址存放 APP 用户代码，IAP 代码最后会跳转到用户代码执行。64-128KB 地址存放待升级的代码。EEPROM 20-24KB 地址存放两个字节的 IAP 升级标志 0X57 0XAB。
- 升级的代码 BIN 文件由主机服务器通过 TCP 传给 CH563，经过 TransHexToBin 软件转换的 Bin 文件前面有固定 8 字节长度的信息：
0-1：固定 0XAA 0XBB ;2-3：两个字节的 bin 文件版本号，可以在软件中设置。
4-7：4 个字节的 BIN 文件长度（字节数），由软件算出。
8-11：checksum 校验和；12-15：保留



如何将两个代码分开放入 FLASH ,FLASH 划分操作:

1. CH563 有一个 SCF 文件, 这是一个 ROM 分配的文件。
2. CH563 的 IAP 程序因为是放在 0-4KB 的, 所以 SCF 不需要进行修改。用户代码因为是从 4KB (0x1000H) ,所以需要将 ROM_LOAD 和 FLASH_CODE 修改到 0x00001000 始。

```
; Define address for FlashROM/code_segment and SRAM/data_segment
; 64K-CODE-RAM + 64K-DATA-RAM

ROM_LOAD 0x00001000
{
    FLASH_CODE 0x00001000
    {
        STARTUP.O (RESET, +First)
        SYSFREQ.O
        * (+RO)
        L_".O (+RO, +Last)
    }
    SRAM_DATA 0x00010000 0xFC00
    {
        STARTUP.O (+RW, +ZI)
        * (+RW, +ZI)
    }
    SRAM_HEAP +0 UNINIT
    {
        STARTUP.O (HEAP)
    }
    SRAM_STACK 0x00020000 UNINIT
    {
        STARTUP.O (STACK)
    }
}
```

3. 因为 APP 用户程序被整体偏移了 4KB, 所以为了能够跳转到 IRQ_Handler 和 FIQ_Handler 里面, 跳转就得分两步。第一步, IAP 的启动文件里面先跳转到跳转 IRQ_Handler 这个命令的地址, 正常这个命令的地址 0x0018, 因为用户程序偏移了 4KB, 所以变为了 0x1018。同理, 跳转 FIQ_Handler 这个命令的地址变为了 0x101c。所以 IAP 程序的启动文件里面, 需要将这两个中断的跳转的命令地址偏移 0x1000。这样才能跳到正确的中断里面。

```
; B IRQ_Handler
B 0x1018
IF FIQ_USE_ASM = 0
IMPORT FIQ_Handler
; B FIQ_Handler
B 0x101c
ELSE
```

4. IAP 程序的最后会有一个 jumptoapp 函数, 里面执行的内容就是跳转到 0x1000 这个地址区运行。这样就完成了 IAP 程序跳转到 APP 用户程序。

```
asm void jumptoapp(void)
{
    B 0x1000
}
```