

## AT32 IAP using the USB HID

## 前言

对于大多数基于闪存的系统，一项重要要求是能够在最终产品中安装固件时进行更新。此功能称为应用程序内编程（IAP）。

本应用笔记的目的是提供在AT32微控制器上创建IAP by USB HID应用程序的一般准则。

支持型号列表：

支持型号	AT32F415xx
------	------------

## 目录

<b>1</b>	<b>IAP 在线升级原理概述 .....</b>	<b>5</b>
<b>2</b>	<b>AT32 USB HID IAP 快速使用方法 .....</b>	<b>8</b>
2.1	硬件资源 .....	8
2.2	软件资源 .....	8
2.3	IAP Programmer 使用 .....	8
2.4	进入 IAP Mode 方式 .....	9
<b>3</b>	<b>AT32 USB HID IAP 程序设置 .....</b>	<b>10</b>
3.1	地址分布 .....	10
3.2	IAP project 设置 .....	10
3.3	APP 设置 .....	10
3.4	USB HID IAP、APP 与上位机通信流程 .....	11
<b>4</b>	<b>USB HID IAP 烧写 SPI FLASH .....</b>	<b>14</b>
4.1	硬件资源 .....	14
4.2	软件资源 .....	14
4.3	使用 IAP Programmer 下载数据到 SPI FLASH .....	14
<b>5</b>	<b>版本历史 .....</b>	<b>15</b>

## 表目录

表 1. 地址分布 .....	10
表 2. 文档版本历史 .....	15

## 图目录

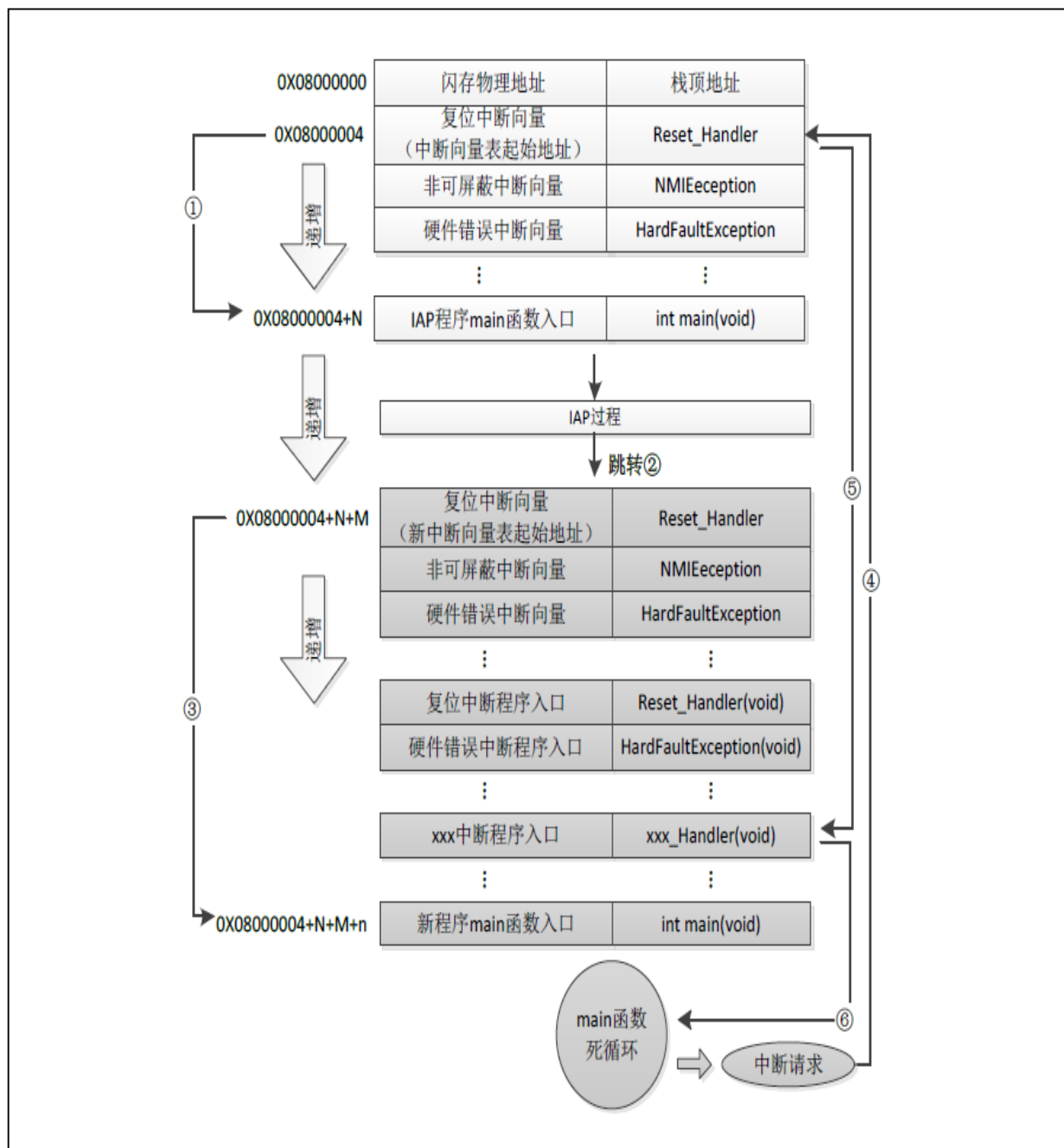
图 1. IAP 代码执行流程 .....	6
图 2. IAP demo 上位机 .....	9
图 3. IAP Project 中 Add1 在 Keil 设置 .....	10
图 4. IAP Project 中 Add2 在程序中设置 .....	10
图 5. APP Project 中 Add2 在 Keil 设置 .....	11
图 6. App 源程序设置 .....	11
图 7. 上位机通信流程 .....	12
图 8. IAP 端下位机通信流程 .....	13

# 1 IAP 在线升级原理概述

IAP (In Application Programming) 即在应用编程，IAP是用户自己的程序在运行过程中对User Flash的部分区域进行烧写，目的是为了在产品发布后可以方便地通过预留的通信口对产品中的固件程序进行更新升级。通常实现IAP功能时，即用户程序运行中作自身的更新操作，需要在设计固件程序时编写两个项目代码，第一个项目程序不执行正常的功能操作，而只是通过某种通信方式(如USB、USART)接收程序或数据，执行对第二部分代码的更新；第二个项目代码才是真正的功能代码。这两部分项目代码都同时烧录在User Flash中，当芯片上电后，首先是第一个项目代码开始运行，它作如下操作：

1. 检查是否需要第二部分代码进行更新，
2. 如果不需要更新则转到 4
3. 执行更新操作
4. 跳转到第二部分代码执行

图 1. IAP 代码执行流程



在图上图所示流程中，AT32复位后，还是从0X08000004地址取出复位中断向量的地址，并跳转到复位中断服务程序，在运行完复位中断服务程序之后跳转到IAP的main函数，如图标号①所示。在执行完IAP以后（即将新的APP代码写入AT32的FLASH，灰底部分。新程序的复位中断向量起始地址为0X08000004+N+M），跳转至新写入程序的复位向量表，取出新程序的复位中断向量的地址，并跳转执行新程序的复位中断服务程序，随后跳转至新程序的main函数，如图标号②和③所示，同样main函数为一个死循环，并且注意到此时AT32的FLASH，在不同位置上，共有两个中断向量表。

在main函数执行过程中，如果CPU得到一个中断请求，PC指针仍强制跳转到地址0X08000004中断向量表处，而不是新程序的中断向量表，如图标号④所示；程序再根据我们设置的中断向量表偏移量，跳转到对应中断源新的中断服务程序中，如图标号⑤所示；在执行完中断服务程序后，程序返回main函数继续运行，如图标号⑥所示。

通过以上两个过程的分析，我们知道IAP程序必须满足两个要求：

1. 新程序必须在IAP程序之后的某个偏移量为x的地址开始
2. 必须将新程序的中断向量表相应的移动，移动的偏移量为x

## 2 AT32 USB HID IAP 快速使用方法

### 2.1 硬件资源

- 1) 指示灯LED2/LED3/LED4
- 2) USB(PA11/PA12)
- 3) AT-START-F415 V1.0实验板

### 2.2 软件资源

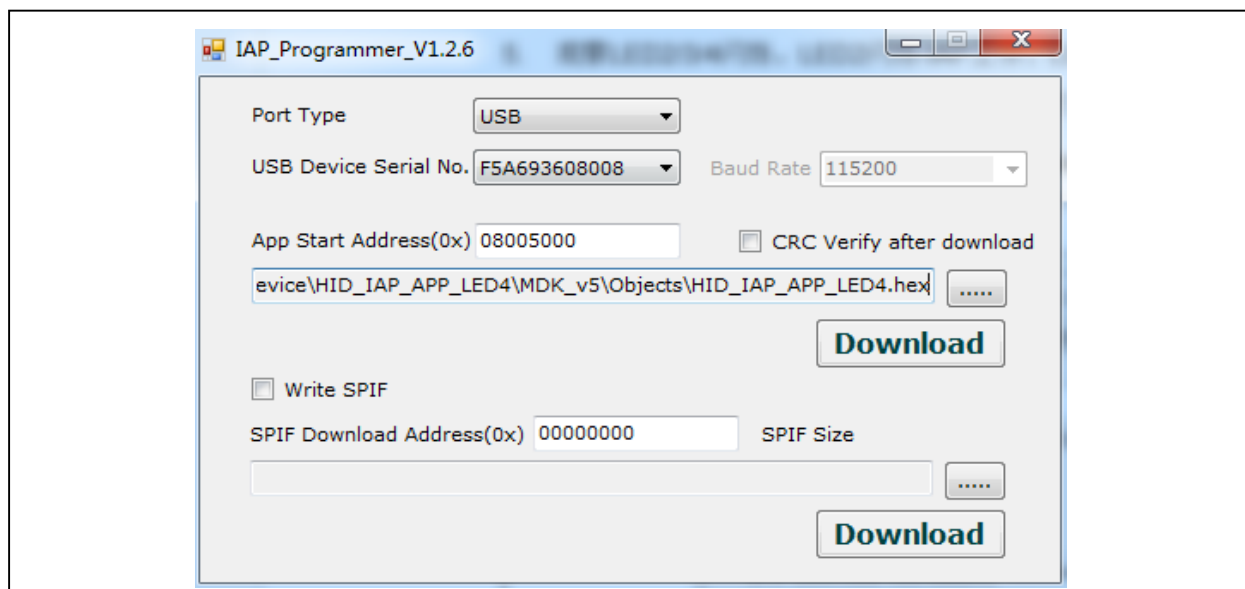
1. APP\_Release
  - IAP\_Programmer.exe PC机tool, 下载时 LED2闪烁
2. SourceCode
  - HID\_IAP\_Demo, IAP源程序
  - HID\_IAP\_APP\_LED3, APP1 LED3闪烁源程序
  - HID\_IAP\_APP\_LED4, APP2 LED4闪烁源程序
3. Doc
  - 1) AN0007\_AT32\_IAP\_using\_the\_USB\_HID
  - 2) AT32 USB HID IAP Protocol V1.0.0 AT32 使用IAP 升级时使用的协议

### 2.3 IAP Programmer 使用

1. 打开HID\_IAP\_Demo源程序, 编译后下载到实验板
2. 打开IAP Programmer.exe
3. 如图, 选择USB设备,使用的是HID设备, 因此不需要驱动
4. 选择APP下载地址(下载地址需要与IAP设置的下载地址相同)和bin文档, 选择是否需要CRC校验, 点击Download下载
5. 观察LED2/3/4闪烁, LED2闪烁-IAP工作, LED3闪烁-APP1工作, LED4闪烁-APP2工作
6. 支持断电保护, 当程序没有download成功时, 下次启动还在IAP模式



图 2. IAP demo 上位机



## 2.4 进入 IAP Mode 方式

1. 一直按住User按键，再按Reset键，进入IAP模式(LED2闪烁)
2. 在User APP 中收到特定指令(参考HID\_IAP\_APP\_LED3/ HID\_IAP\_APP\_LED4)，擦除flag，然后Reset，进入IAP模式（LED2闪烁）

## 3 AT32 USB HID IAP 程序设置

### 3.1 地址分布

表 1. 地址分布

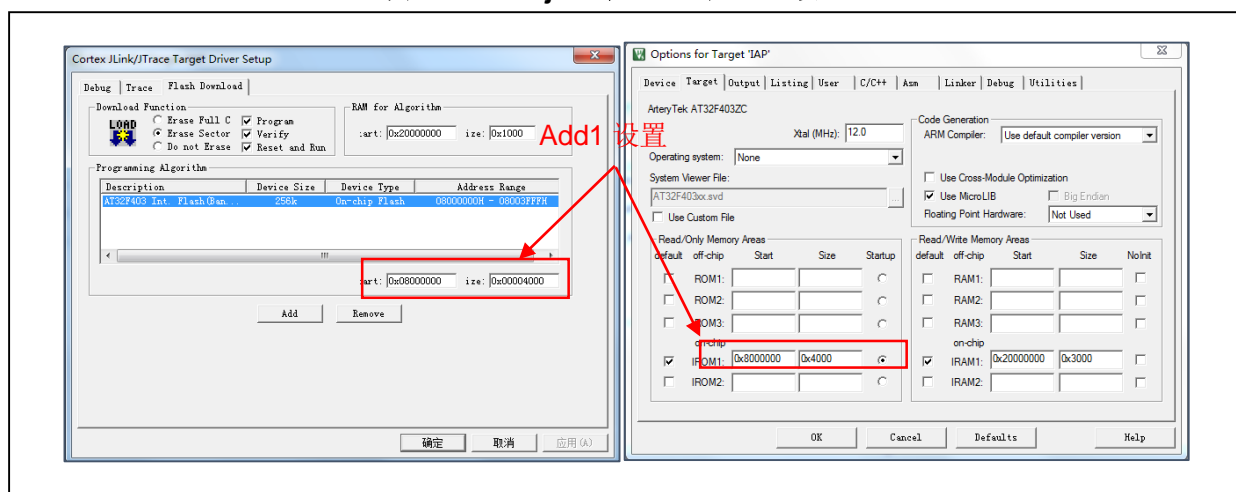
ITEM	Address and Size
APP code	Add2:0x800 5000 ~
IAP code	Add1:0x800 0000 size:0x5000(20K Byte)

Note :IAP 区域最后一个 page(2KByte), 用于放置防掉电丢失的 flag, 用户修改 IAP 时请勿操作该段地址。

### 3.2 IAP project 设置

#### 1) Keil 设置

图 3. IAP Project 中 Add1 在 Keil 设置



#### 2) IAP源程序修改

lap\_user.h文件中

图 4. IAP Project 中 Add2 在程序中设置

```
#define FLASH_APP1_ADDR 0x08005000 /*User App Start address*/

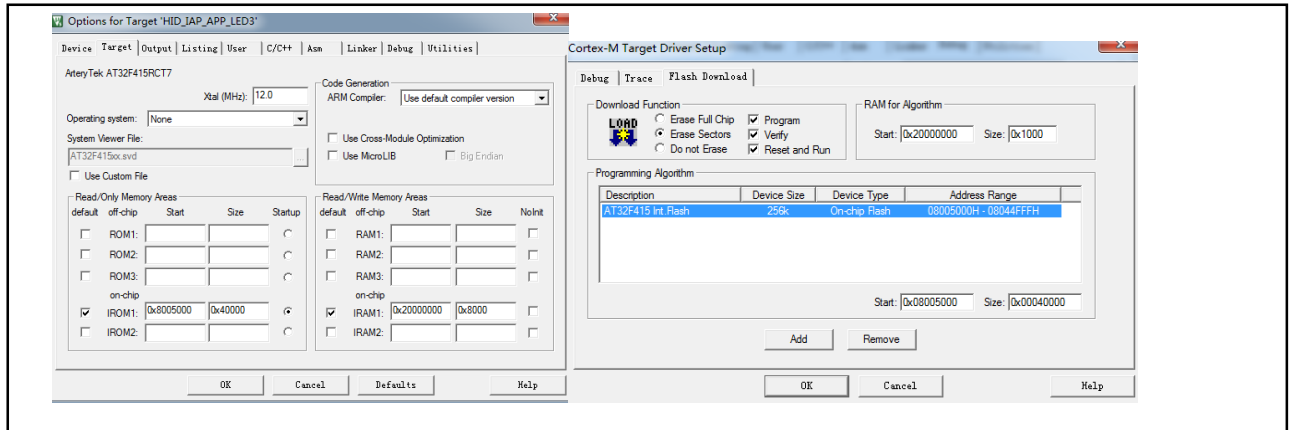
#define IAP_CMD_FINISH_FLAG_ADDR (FLASH_APP1_ADDR - 0x800) /*用于断电保护*/
#define IAP_DOWNLOAD_FLAG 0x41544B38
```

### 3.3 APP 设置

该demo提供了2个APP程序供测试用, 皆以Add2 (0x800 5000) 为起始地址。App LED3闪烁, App LED4闪烁。以App LED3为例, 设计步骤如下:

#### 1. Keil工程设置

图 5. APP Project 中 Add2 在 Keil 设置



## 2. App 源程序设置

修改main.c 中的中断向量偏移

图 6. App 源程序设置

```
SCB->VTOR = FLASH_BASE | 0x5000;
```

## 3. 编译生成bin文件

通过在User选项卡，设置编译后调用fromelf.exe，根据.axf文件生成.bin文件，用于IAP更新。

以上3个步骤，我们就可以得到一个.hex的APP程序，通过IAP程序即可实现更新。

## 4. 开启Debug app code功能

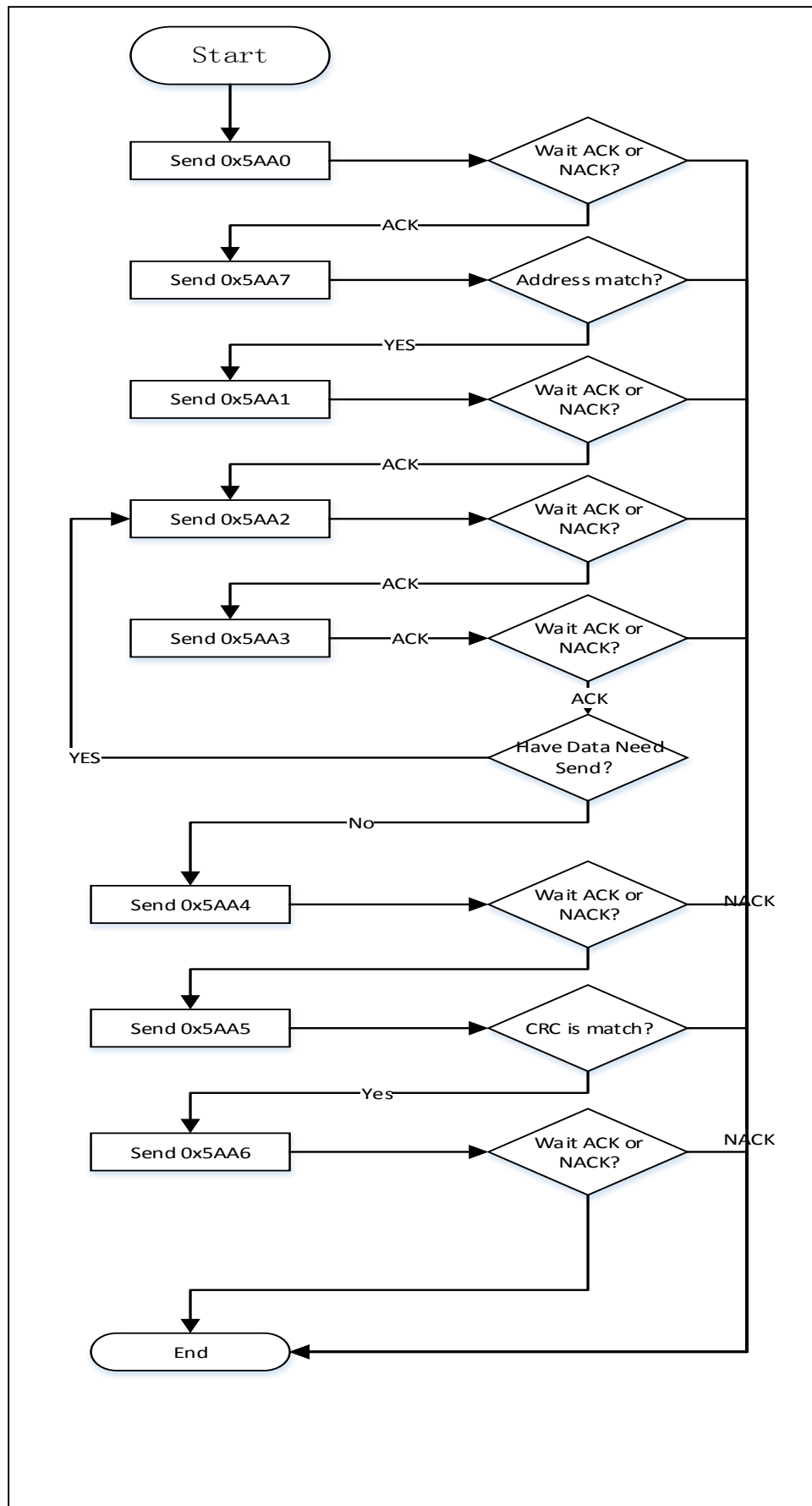
如果在设计app code过程中需要对app project进行单独调试，请按照以下操作。

- 下载IAP project
- 第一次使用Debug功能需要IAP Programmer .exe 成功下载一次APP.(成功下载之后会写FLAG，表示下次从APP 启动。默认FLAG 会从IAP启动)
- debug app project

## 3.4 USB HID IAP、APP 与上位机通信流程

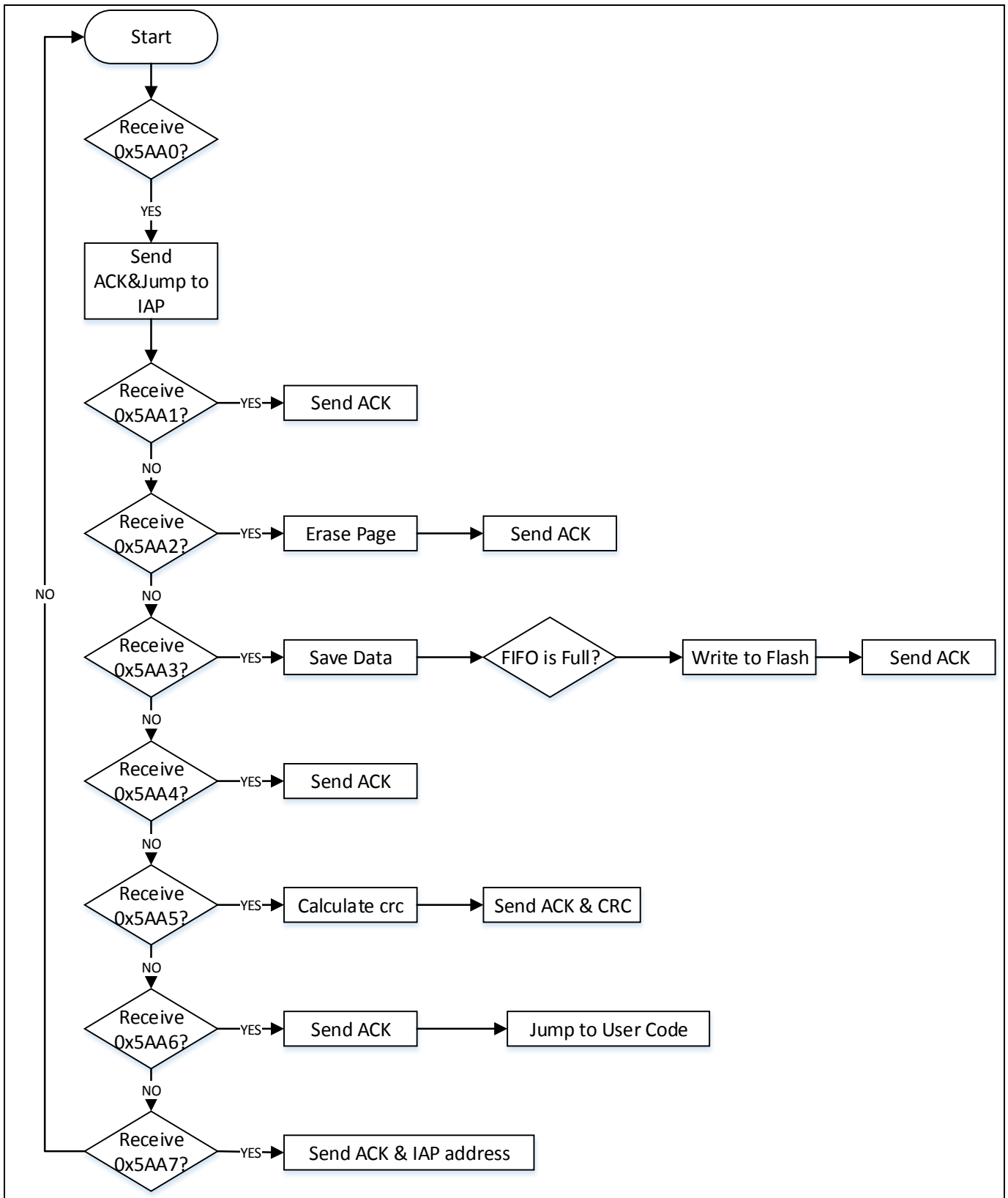
### 1. 上位机通信流程

图 7. 上位机通信流程



## 2. IAP 端下位机通信流程

图 8. IAP 端下位机通信流程



NOTE: 具体协议请参考 AT32\_HID\_IAP\_Protocol\_V1.0.0.pdf

## 4 USB HID IAP 烧写 SPI FLASH

### 4.1 硬件资源

1. USB (PA11&PA12)
2. SPI2 (PB12&PB13&PB14&PB15) (SPI 可根据需要配置@at\_spi\_flash.h)
3. 目前支持Winbond 25Q64FV / Winbond 25Q128FV

### 4.2 软件资源

#### 1. APP Release

- IAP\_Programmer.exe PC机tool

#### 2. Source Code

- HID\_IAP\_Demo, IAP源程序
- iap\_user.c: 与上位机进行通信命令的实现（可增加其它命令以实现新的功能）
- at\_spi\_flash.c: SPI FLASH 的初始化及操作接口（用户可根据不同的FLASH进行配置）
- IAP\_Programmer: 上位机源程序

### 4.3 使用 IAP Programmer 下载数据到 SPI FLASH

**注意:** 在目前Demo中, 设置SPI FLASH的起始地址必须按照Sector对齐, 目前Sector为4K, 具体需求用户可进行调整。

1. 打开HID\_IAP\_Demo源程序, 编译后下载到实验板
2. 打开IAP Programmer.exe
3. 如图, 选择USB设备,使用的是HID设备, 因此不需要驱动
4. 选择FLASH下载地址（下载地址需要按照Sector对齐, 用于判断是否要进行擦除sector）和bin文档, 选择是否需要CRC校验, 点击Download下载

## 5 版本历史

表 2. 文档版本历史

日期	版本	变更
2018.07.09	1.0.0	最初版本
2018.10.15	1.0.1	增加烧写 SPI FLASH
2020.07.17	1.0.2	修改部分描述

**重要通知 - 请仔细阅读**

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途(及其依据任何司法管辖区的法律的对应情况)，或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：(A) 对安全性有特别要求的应用，如：生命支持、主动植入设备或对产品功能安全有要求的系统；(B) 航空应用；(C) 汽车应用或汽车环境；(D) 航天应用或航天环境，且/或(E) 武器。因雅特力产品不是为前述应用设计的，而采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险由购买者单独承担，并且独力负责在此类相关使用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2020 雅特力科技 (重庆) 有限公司 保留所有权利