

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应用程序的方法。

IAP_Programmer.exe上位机软件和嵌入式HID IAP示例的源代码位于BSP固件库的utilities文件夹内。

支持型号列表：

支持型号	AT32F4 系列
------	-----------

注：仅支持有USB外设的型号

目录

1	IAP 在线升级原理概述	5
2	AT32 USB HID IAP 快速使用方法	7
	2.1 硬件资源	7
	2.2 软件资源	7
	2.3 IAP Programmer 使用	7
3	AT32 USB HID IAP 程序设置	9
	3.1 地址分布	9
	3.2 bootloader project 设置	9
	3.3 app 设置	9
	3.4 IAP、APP 与上位机通信流程	10
4	USB HID IAP Protocol	13
	4.1 命令列表	13
	4.2 命令详解	13
5	版本历史	15

表目录

表 1. 地址分布	9
表 2. 文档版本历史	15

图目录

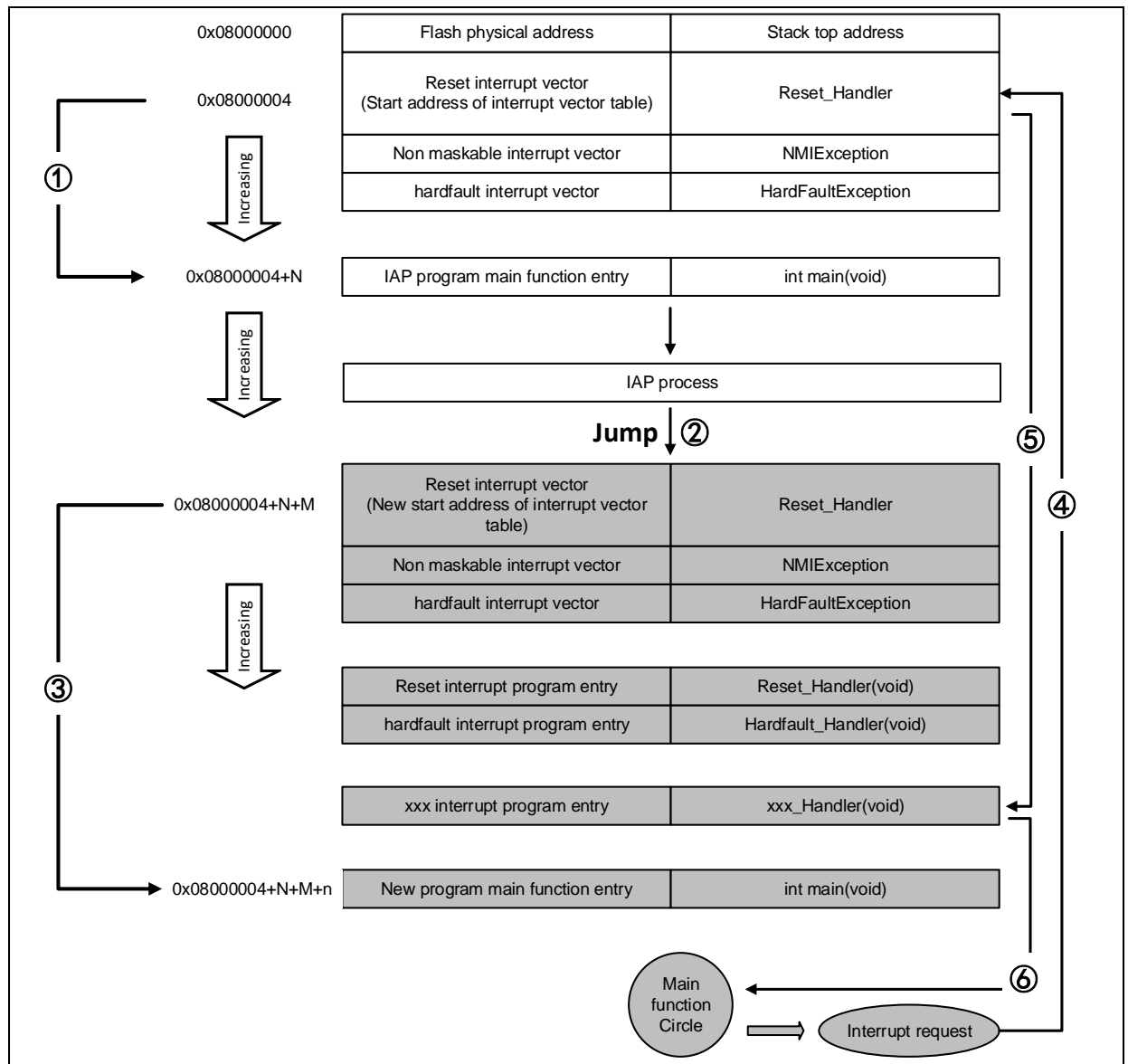
图 1. IAP 代码执行流程	5
图 2. IAP demo 上位机	8
图 3. bootloader project 中 add1 在 Keil 设置	9
图 4. bootloader project 中 add2 在程序中设置	9
图 5. app project 中 add2 在 Keil 设置	10
图 6. app 源程序设置	10
图 7. 上位机通信流程	11
图 8. IAP 端下位机通信流程	12

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

文档中是用 AT-START-AT32F403A 实验板的硬件条件为例，IAP demo 源代码还包括 AT32 其他型号，用户只需编译对应型号工程烧录于 AT-START 实验板运行即可。

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

2.2 软件资源

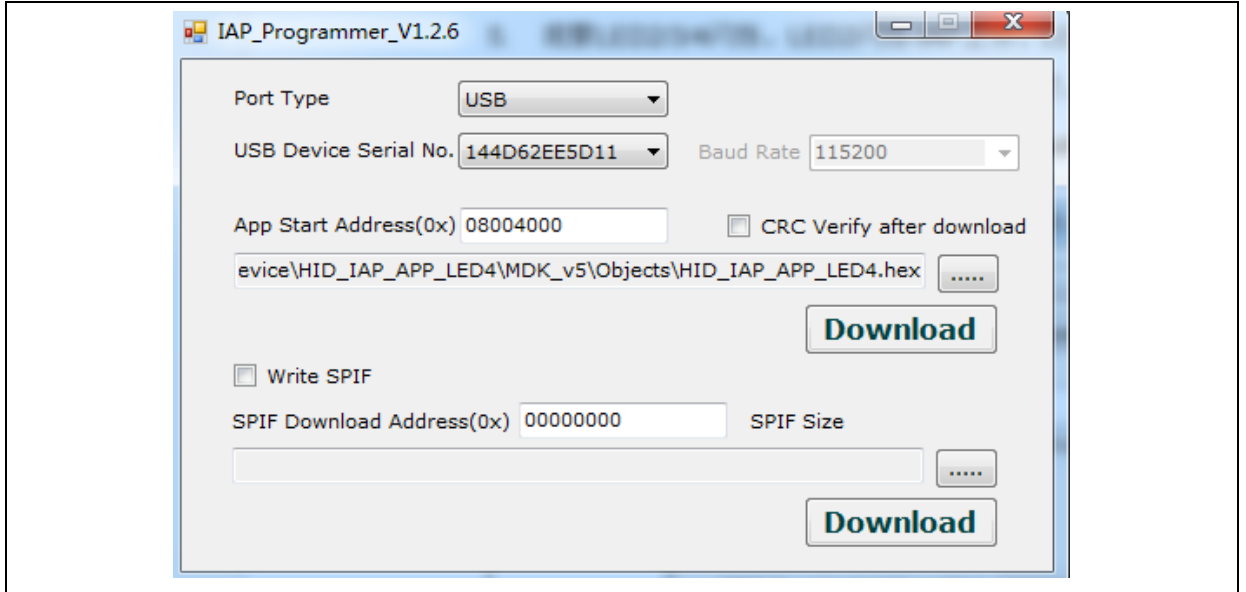
1. APP_Release
 - IAP_Programmer.exe PC机tool
2. source_code
 - bootloader, IAP源程序, 运行LED2闪烁
 - app_led3_toggle, app1 LED3闪烁源程序
 - app_led4_toggle, app2 LED4闪烁源程序

注：源码位于AT32F403A_407_Firmware_Library_V2.x.x\utilities\at32f403a_407_usb_iap_demo\source_code, 示例工程基于keil v5和IAR8.2建立, 若用户需要在其他编译环境上使用, 请参考AT32F403A_407_Firmware_Library_V2.x.x\project\at_start_f403a\templates中各种编译环境 (例如IAR6/7/8,keil 4/5,eclipse_gcc) 进行对应修改即可。

2.3 IAP Programmer 使用

1. 打开bootloader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-bootloader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



3 AT32 USB HID IAP 程序设置

3.1 地址分布

表 1. 地址分布

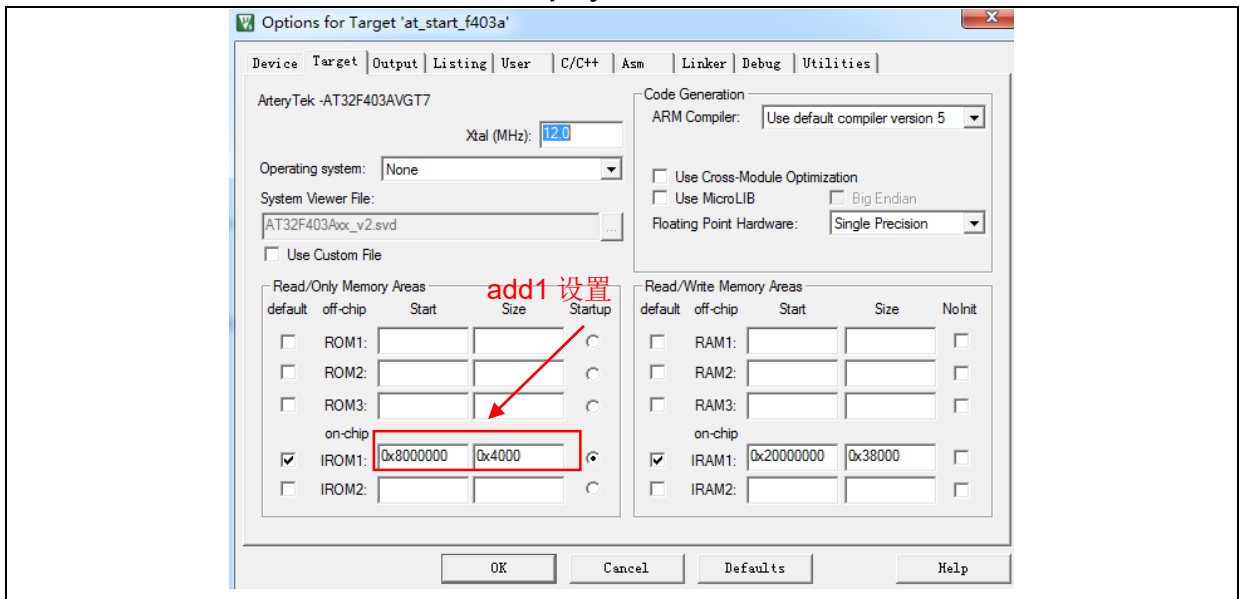
ITEM	Address and Size
app code	Add2:0x800 4000 ~
bootloader code	Add1:0x800 0000 size:0x4000(16K Byte)

Note :bootloader 区域最后一个扇区，用于放置防掉电丢失的 flag，用户修改 bootloader 时请勿操作该段地址。

3.2 bootloader project 设置

1) Keil 设置

图 3. bootloader project 中 add1 在 Keil 设置



2) bootloader源程序修改

hid_iap_user.h文件中

图 4. bootloader project 中 add2 在程序中设置

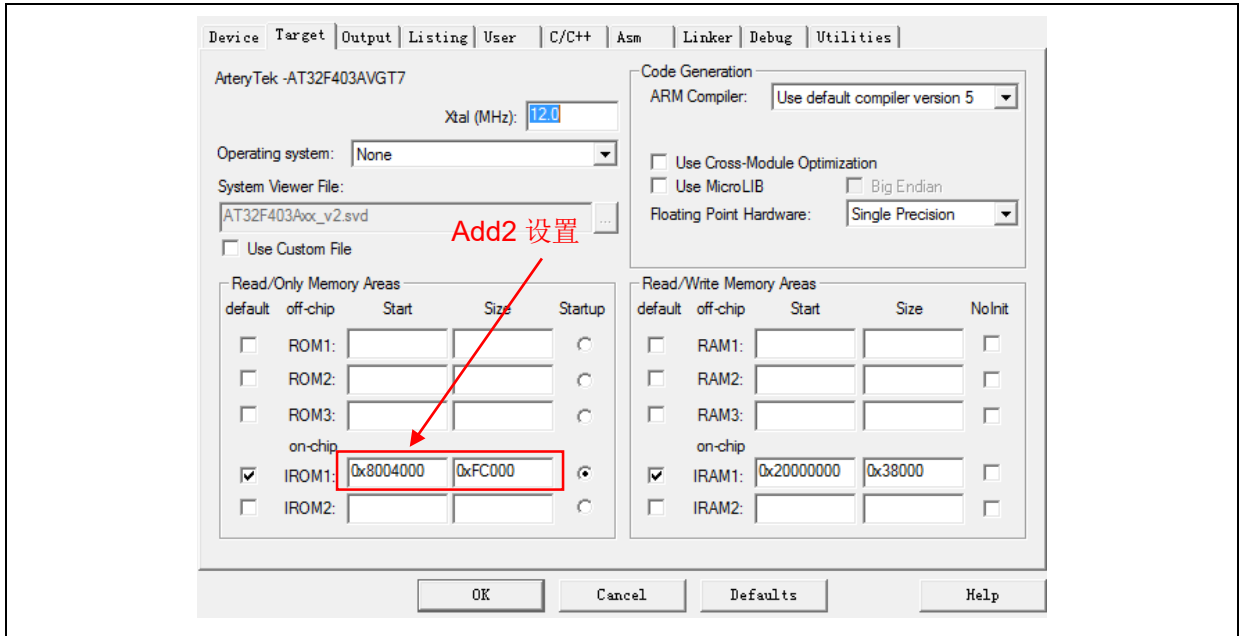
```
#define FLASH_APP_ADDRESS          0x08004000
```

3.3 app 设置

该demo提供了2个app程序供测试用，皆以add2（0x800 4000）为起始地址。app1 LED3闪烁，app2 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

```
/* config vector table offset */
nvic_vector_table_set(NVIC_VECTTAB_FLASH, 0x4000);
```

中断向量表的偏移量设置

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.bin的APP程序，通过bootloader程序即可实现更新。

4. 开启debug app code功能

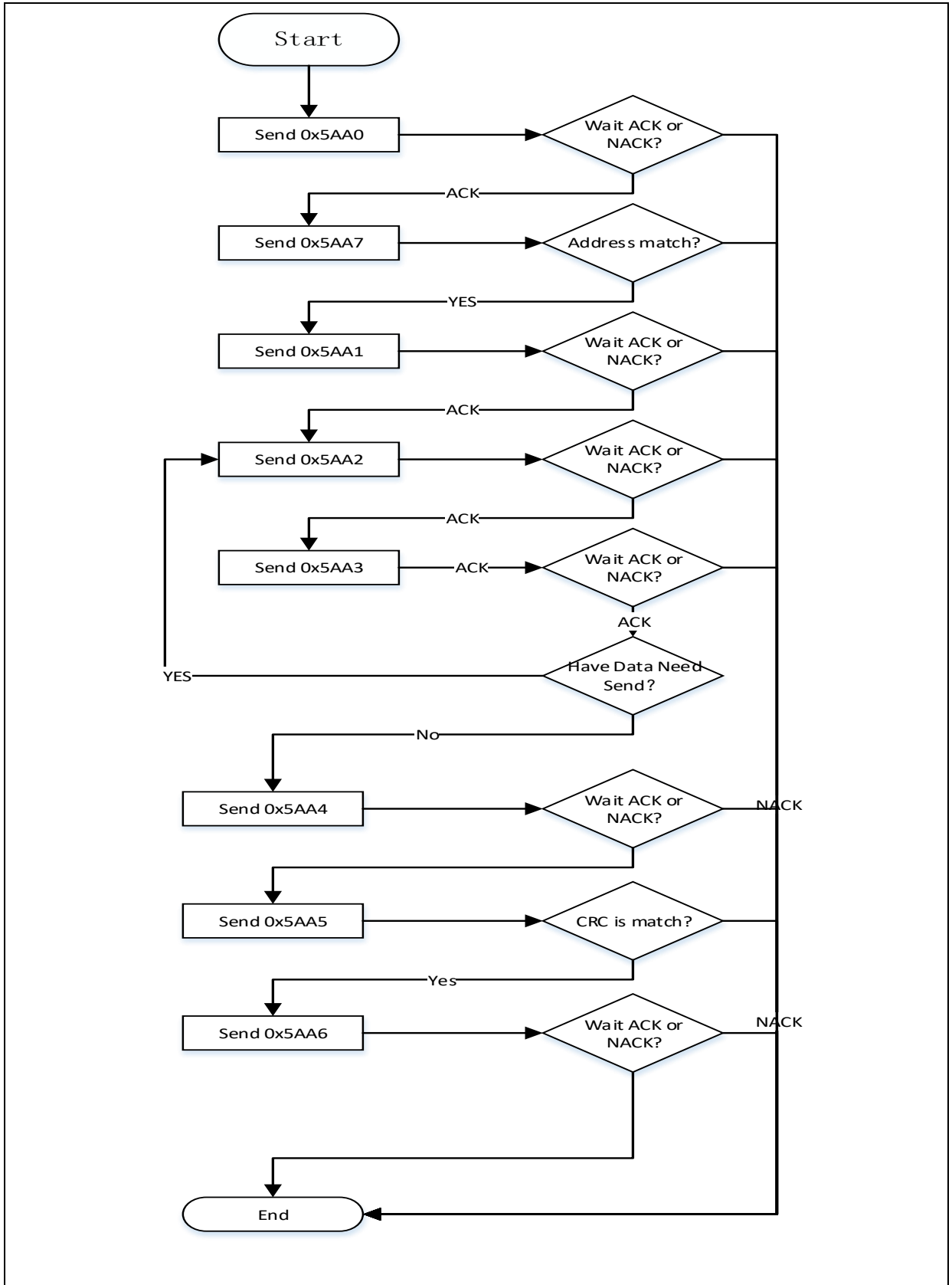
如果在设计app code过程中需要对app project进行单独调试，请按照以下操作。

- 下载bootloader 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bootloader启动)
- debug app project

3.4 IAP、APP 与上位机通信流程

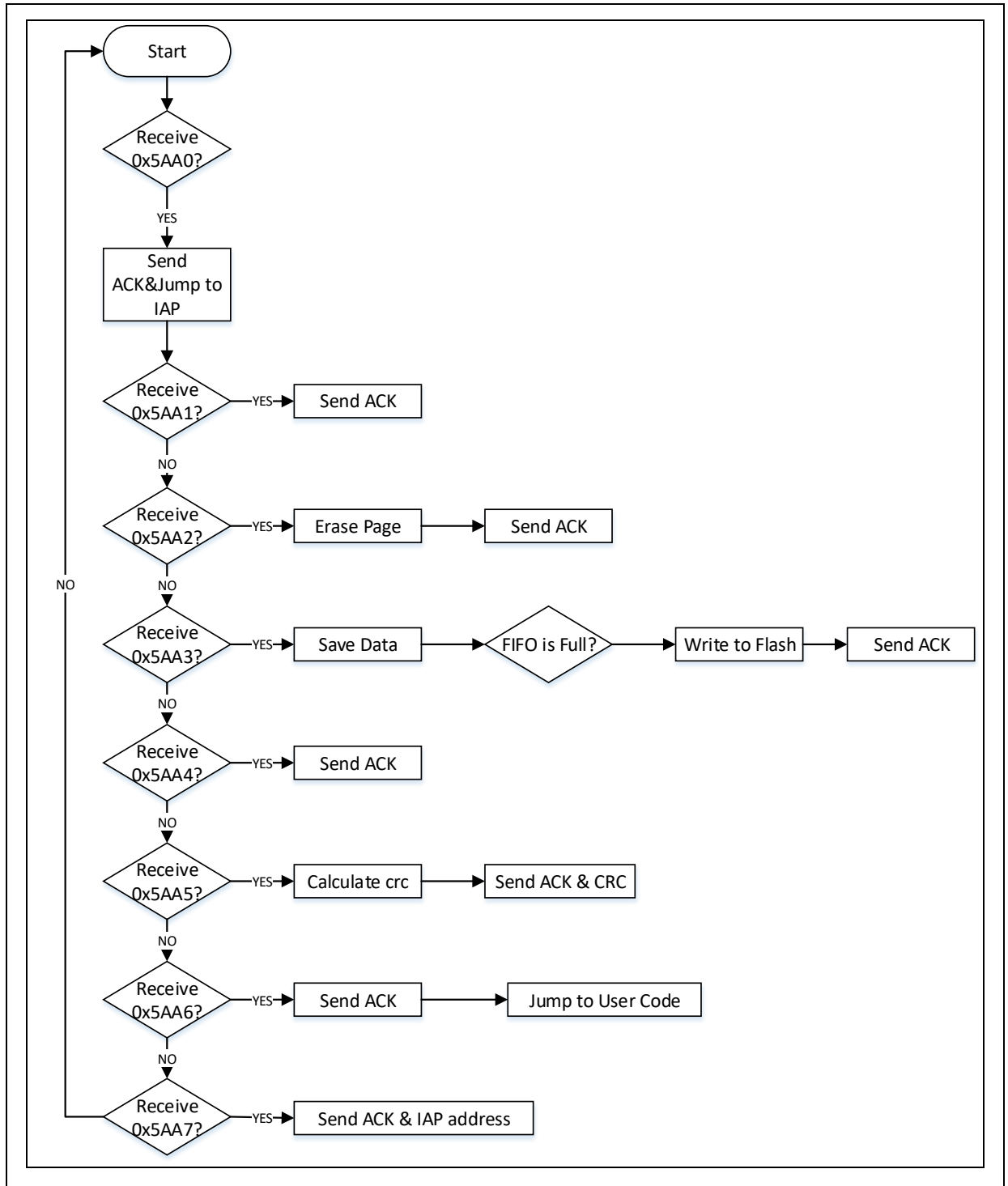
1. 上位机通信流程

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



注意：具体协议请参考AT32_HID_IAP_Protocol.pdf

4 USB HID IAP Protocol

本节描述 USB HID 升级协议，使用此协议与上位机进行通信，达到升级固件的目的。

4.1 命令列表

命令值	描述
0x5AA0	进入 IAP 模式
0x5AA1	开始下载
0x5AA2	设置地址(按 1K 对齐)
0x5AA3	下载数据命令
0x5AA4	下载结束
0x5AA5	CRC 校验
0x5AA6	跳转命令(跳转到用户 code)
0x5AA7	获取 IAP 设置的 User Code 地址

注意 1: HID MaxPacket = 64Byte

注意 2: 每一个包的前面两个 Byte 固定为命令

注意 3: 命令按照 MSB,LSB 的顺序传输

ACK: 0xFF00, NACK: 0x00FF

4.2 命令详解

1. 0x5AA0进入IAP模式

作为一个特定的命令，当用户APP收到这个命令之后将进入IAP模式。实现方式为收到这个命令之后擦除flag然后reset

上位机: [0x5A, 0xA0]

IAP设备响应: [0x5A, 0xA0, ACK/NACK]

2. 0x5AA1 开始下载

上位机: [0x5A,0xA1]

IAP设备响应: [0x5A,0xA1,ACK/NACK]

3. 0x5AA2设置下载地址

设置下载地址需按照1KB对齐，每下载1Kbyte数据之后，都需要重新设置下载地址。

上位机(命令+地址): [0x5A, 0xA2, 0x08, 0x00, 0x40, 0x00]

IAP设备响应: [0x5A,0xA2, ACK/NACK]

4. 0x5AA3 下载数据命令(1KB 对齐多个包发送)

下载数据命令采用 命令+长度+数据的格式进行发送，每包最大数据量为60Byte（64 - 命令 - 长度），当发送数据达到1KB时，上位机需要等待设备的ACK响应。此时设备需将1KB的数据写到FLASH。

上位机（命令(2Byte)+长度(2 Byte)+数据(n byte)）:[0x5A,0xA3,LEN1, LEN0,DATA0...DATAn]

收完1KB数据后IAP设备响应: [0x5A, 0xA3, ACK/NACK]

5. 0x5AA4 下载结束

上位机: [0x5A, 0xA4]

IAP设备响应: [0x5A, 0xA4, ACK/NACK]

6. 0x5AA5 固件CRC校验

上位机传输固件起始地址和固件大小/1KB（固件大小按1KB对齐，不足补0xFF），由IAP计算CRC之后返回给上位机。

上位机: [0x5A,0xA5, 0x08, 0x00, 0x40, 0x00, LEN1, LEN0]

IAP设备响应: [0x5A, 0xA5, ACK/NACK, CRC3, CRC2, CRC1, CRC0]

7. 0x5AA6 跳转命令

跳转命令将跳转到用户代码进行运行

上位机: [0x5A,0xA6, 0x08, 0x00, 0x40, 0x00]

IAP设备响应: [0x5A,0xA6,ACK/NACK]

8. 0x5AA7 获取IAP设置的app地址

返回IAP设置的app地址

上位机: [0x5A, 0xA7]

IAP 设备响应: [0x5A, 0xA7, ACK/NACK, 0x08, 0x00, 0x40, 0x00]

5 版本历史

表 2. 文档版本历史

日期	版本	变更
2021.09.02	2.0.0	初始版本
2022.01.17	2.0.1	修改截图中芯片型号
2023.02.08	2.0.2	增加 sourcecode 在 BSP 中的路径描述

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。

© 2023 雅特力科技 (重庆) 有限公司 保留所有权利