

前言

STMCube™ 计划源自意法半导体，旨在通过减少开发的工作量、时间与成本，使开发者受益。STM32Cube 涵盖 STM32 产品系列。

STM32Cube 1.x 版包括：

- 图形软件配置工具 STM32CubeMX，可通过图形化的向导生成初始化 C 代码。
- 针对每个系列提供综合的嵌入式软件平台，（比如用于 STM32F4 系列的 STM32CubeF4）
 - STM32 抽象层嵌入式软件 STM32Cube HAL，确保在 STM32 各个产品之间实现最大限度的可移植性
 - 一套一致的中间件，比如 RTOS、USB、STM Touch、FatFS 和图形
 - 所有嵌入式软件实用程序均配备一套完整的示例。

在应用中编程（IAP）是在 flash 中执行代码的同时，对同一 flash 存储器编程的方法。它提供了使用高速通信协议加载应用代码的可能性。

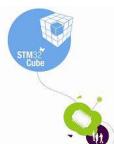
本用户手册的目标读者为在 STM32 微控制器上使用 STM32Cube 固件的开发者。它提供了如何使用以太网通信，实现在应用中编程（IAP）的全部说明。

在 LwIP TCP/IP 协议栈上提供了两种可行的解决方案：

- 使用 TFTP（简单文件传输协议）的 IAP
- 使用 HTTP（超文本传输协议）的 IAP

注：本文适用于具有以太网外设的所有 STM32 系列。然而，为简单起见，使用了 STM32F4xx 微控制器和 STM32CubeF4 作为参考平台。

相同的说明、文件名、屏幕截图也适用于提供以太网连接的其它系列，例如 STM32F107xx、STM32F2x7xx、STM32F7xx。若需了解更多在 STM32 系列上实现以太网 IAP 样例的信息，请参考相关 STM32Cube 固件包中提供的文档。



目录

1	IAP 概述	5
1.1	工作原理	5
1.2	使用 MCU 以太网接口实现 IAP	6
1.3	通过以太网实现 IAP	6
1.3.1	使用 TFTP 实现 IAP 的方法	6
1.3.2	使用 HTTP 实现 IAP 的方法	6
2	使用 TFTP 实现 IAP	7
2.1	TFTP 概述	7
2.2	对于 STM32F4xx, 使用 TFTP 实现 IAP	8
2.3	使用固件	9
3	使用 HTTP 实现 IAP	10
3.1	HTTP 文件上传概述	10
3.2	对于 STM32F4xx, 使用 HTTP 实现 IAP	10
3.3	使用固件	13
4	环境	14
4.1	应用设置	14
4.1.1	PHY 接口配置	14
4.1.2	MAC 和 IP 地址设置	14
4.2	评估板设置	14
4.3	固件文件结构	14
4.4	构建 IAP 映像	15
5	结论	16
附录 A	FAQ	17
A.1	怎样选择静态或动态 (DHCP) IP 地址分配	17
A.2	当以太网网线断开时, 应用如何处理	17
A.3	怎样将应用移植到不同的硬件	17
6	版本历史	18

表格索引

表 1.	TFTP 操作码	7
表 2.	文件结构	14
表 3.	版本历史	18
表 4.	中文版本历史	18

图片索引

图 1.	IAP 操作流程	5
图 2.	TFTP 包	7
图 3.	使用 TFTP 实现 IAP 的流程图	8
图 4.	TFTPD32 对话框	9
图 5.	文件上传 HTML 表单的浏览器视图	10
图 6.	登录网页	11
图 7.	文件上传完成	11
图 8.	使用 HTTP 实现 IAP 的流程图	12

1 IAP 概述

1.1 工作原理

在应用中编程 (IAP) 是一种在现场通过 MCU 的通信接口 (例如 UART,USB,CAN 和以太网等) 进行固件升级的方式。

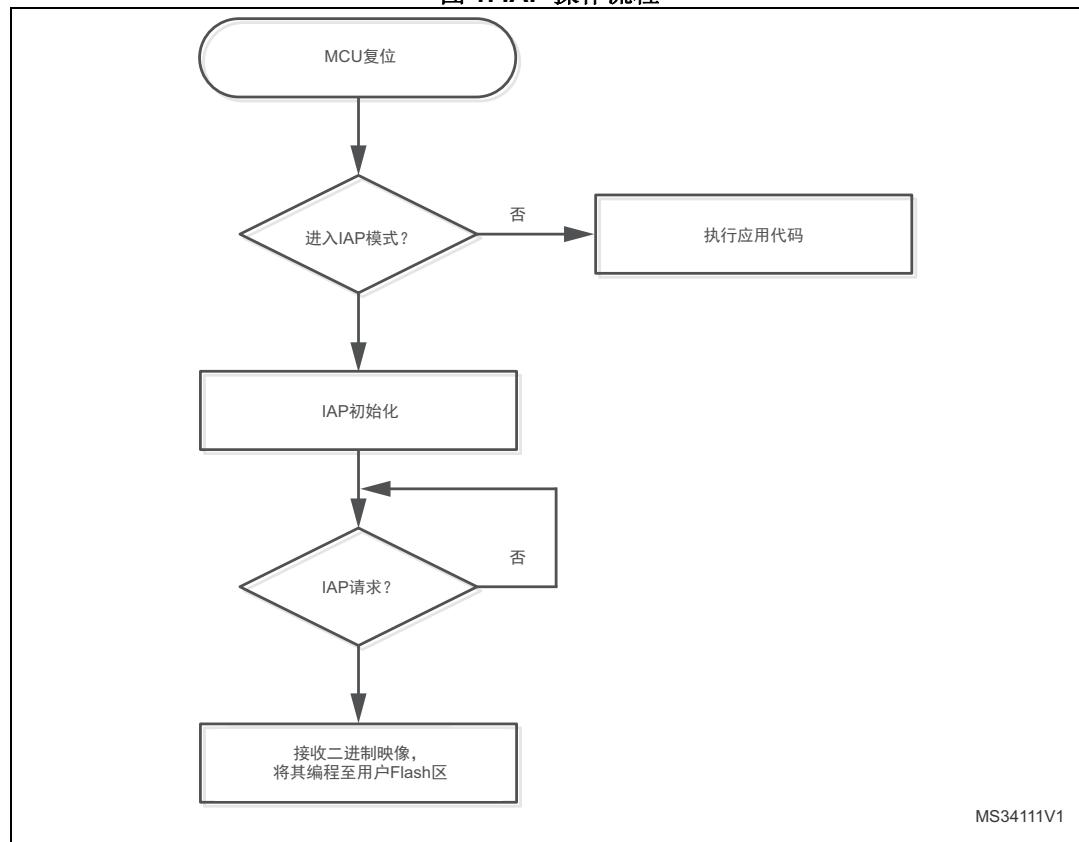
当您启动微控制器时, 您可选择以下两种模式之一:

- IAP 模式, 可执行 IAP 代码。
- 正常模式, 可执行应用代码。

IAP 代码和应用程序代码都在微控制器的嵌入式 Flash 中。通常 IAP 代码被存储在 MCU Flash 的前几页, 用户应用代码占据剩余的 Flash 区域。

[图 1](#) 显示了 IAP 操作流程:

图 1. IAP 操作流程



1.2 使用 MCU 以太网接口实现 IAP

当以太网可用时，它通常是在嵌入式应用中实现 IAP 功能的更好的接口。其优点是：

- 高速通信接口（10/100 Mbit/s）
- 通过网络（LAN 或者 WAN）进行远程编程
- TCP/IP 协议栈上的标准应用协议（例如 FTP、TFTP、HTTP）可用于实现 IAP

1.3 通过以太网实现 IAP

本用户手册说明了两种使用以太网通信外设在 STM32F4xx 上实现 IAP 的解决方案：

- 使用 TFTP（简单文件传输协议）的 IAP
- 使用 HTTP（超文本传输协议）的 IAP

两种解决方案都运行在 LwIP 协议栈上，它是 TCP/IP 协议套件的一种轻量级实现。

1.3.1 使用 TFTP 实现 IAP 的方法

使用 TFTP 实现 IAP 的方法广泛应用于需要固件升级能力的嵌入式应用（比如在嵌入式 Linux bootloader 中）。

TFTP 是工作在 UDP 传输层上的简单文件传输协议。它主要应用在 LAN 环境中。它基于客户端 / 服务器体系结构，其中客户端向文件服务器请求文件传输（读或写操作）。

在这个例子中，服务器只处理来自 PC TFTP 客户端的写请求，所以在 LwIP 栈上实现了一个简单的 TFTP 服务器。

1.3.2 使用 HTTP 实现 IAP 的方法

使用 HTTP 协议进行固件升级没有使用 TFTP 常见，但是当需要通过互联网进行远程编程时，它是一种有用的解决方案。在这种情况下，需要 TCP 传输协议来确保最优操作。

工作在 TCP 上的 HTTP，提供了一种用 HTML 表单从 web 客户端（Mozilla Firefox 或 Microsoft Internet Explorer）发送二进制文件的方法，称作 HTTP 文件上传（RFC 1867）。

本文档的下一部分给出了关于这两种 IAP 方法实现的详细信息并解释了如何使用这个软件。

2 使用 TFTP 实现 IAP

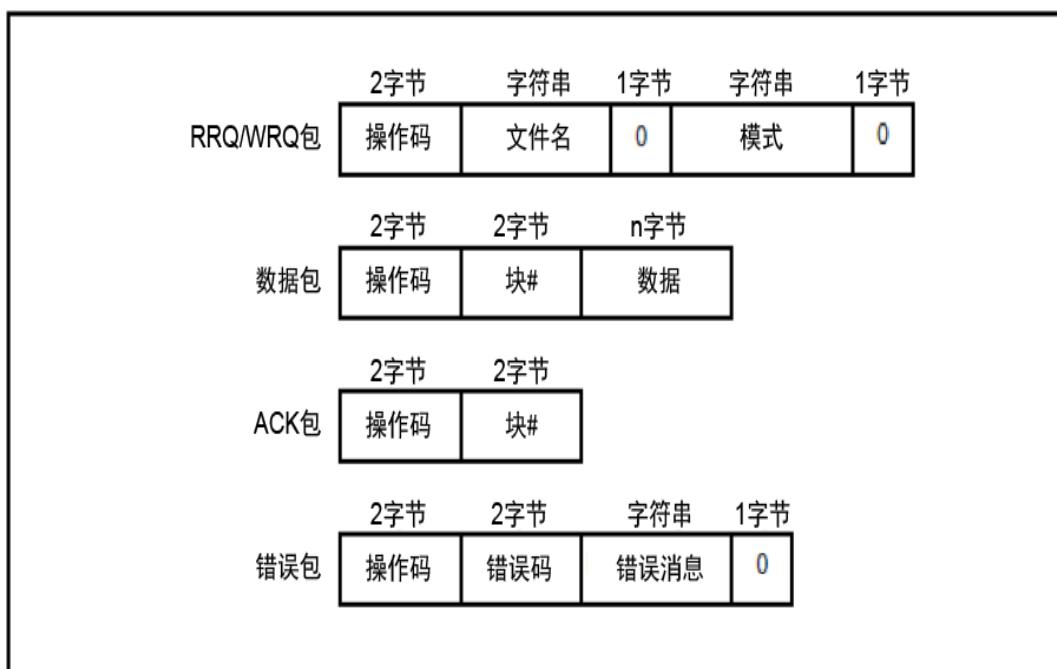
2.1 TFTP 概述

TFTP 是工作在 UDP 传输层上的简单文件传输协议。文件传输是从 TFTP 客户端发起的，TFTP 客户端向 TFTP 服务器发送读或写请求。当服务器响应该请求后，文件数据传输开始。数据按固定大小的数据块发送（如 512 字节的数据块）。

必须在每个传输的数据块被接收端响应后，才能发送下一个数据块。这种回应机制通过随每个数据块同时发送的数据块编号来实现。数据块小于固定块大小表示文件传输结束。

[图 2](#) 说明了各类 TFTP 包的格式：

图 2. TFTP 包



[表 1](#) 列出了 TFTP 操作码。

表 1. TFTP 操作码

操作码	操作
0x1	读请求 (RRQ)
0x2	写请求 (WRQ)
0x3	数据
0x4	响应 (ACK)
0x5	错误

2.2 对于 STM32F4xx, 使用 TFTP 实现 IAP

该 IAP 实现包括一个 LwIP TCP/IP 栈上的 TFTP 服务器。

该服务器响应从远程 TFTP 客户端（PC）接收的 WRITE 请求。

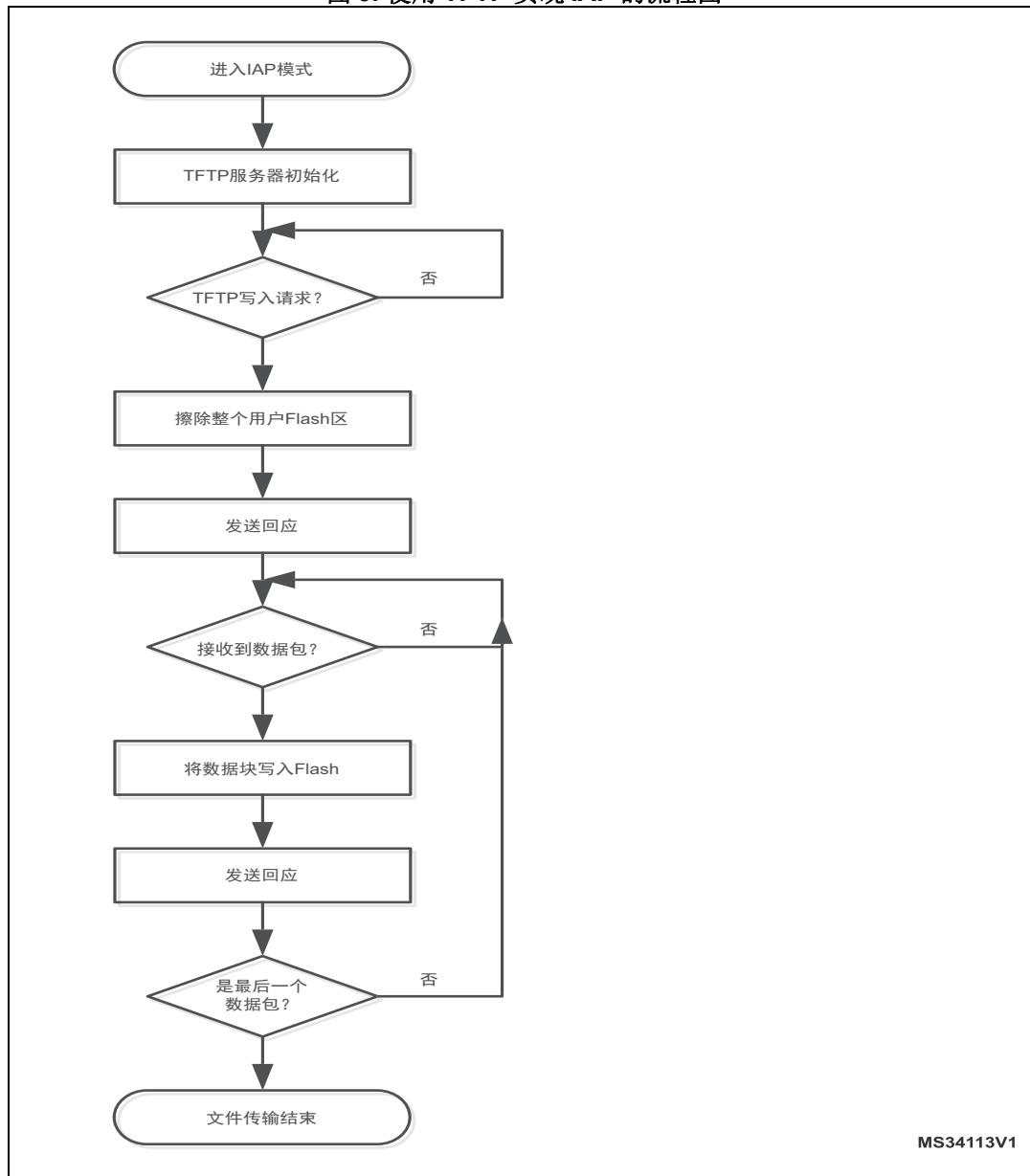
TFTP 读请求被忽略。

服务器将接收到的数据块写入 MCU Flash（在用户 Flash 区域），而不是将接收的文件写入文件系统。

注意：在本实现中，数据块大小固定为 512 字节。

图 3 给出了使用 TFTP 实现 IAP 操作的过程。

图 3. 使用 TFTP 实现 IAP 的流程图

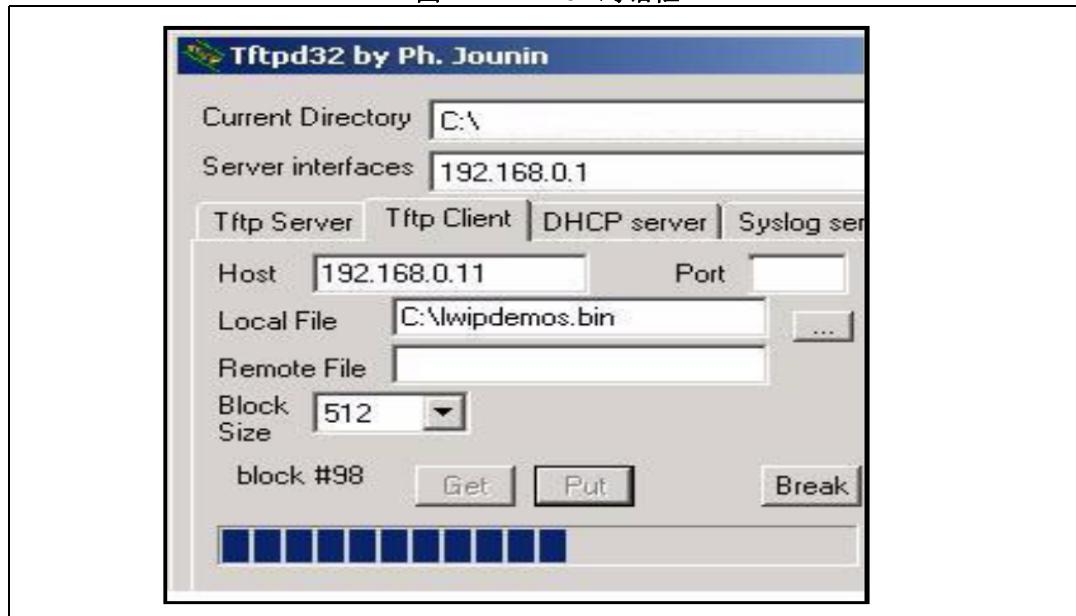


2.3 使用固件

为了通过 TFTP 对 IAP 进行测试，遵循如下步骤：

1. 在评估板上确保正确的跳线设置（参见 [第 4.2 节](#)）
2. 在 main.h 文件中，取消选项 `#define USE_IAP_TFTP` 的注释，并且根据需求您还可以取消注释 / 注释其它选项，比如 `#define USE_DHCP` 或 `#define USE_LCD`。
3. 重新编译固件。使用生成的 map 文件，确保 IAP 代码区域（从地址 0x0 开始）和开始于 `USER_FLASH_FIRST_PAGE_ADDRESS`（在 main.h 中定义）的用户 Flash 区域之间没有重叠。
4. 将固件下载到 STM32F4xx Flash 中并运行它。
5. 按住 **Key** 按钮的同时，按下并释放 **Reset** 键，进入 IAP 模式。
6. 若在 main.h 文件中定义了 `USE_LCD`，LCD 屏幕上会显示消息，显示已经进入 IAP 模式。如果采用了 DHCP（main.h 文件中定义了 `USE_DHCP`），LCD 屏幕上也会显示一条消息，显示 DHCP IP 地址分配的成功或失败。
7. 分配完 IP 地址后（动态或静态地址），用户可启动 IAP 进程。
8. 在 PC 上，打开 TFTP 客户端（例如，TFTPD32）并配置 TFTP 服务器地址（TFTPD32 中的主机地址）。
9. 选择一个二进制映像下载到 STM32F4xx Flash 中（/project/binary 文件夹中提供了一个二进制映像作为例子）。
10. 在 TFTPD32 应用程序上点击 **Put** 按钮，启动一个文件写请求。
11. 如果定义了 LCD，IAP 操作的进度会显示在 LCD 上。
12. 在 IAP 操作结束时，可以复位评估板来运行您刚才在 STM32F4xx Flash 中编写的程序。

图 4. TFTPD32 对话框



3 使用 HTTP 实现 IAP

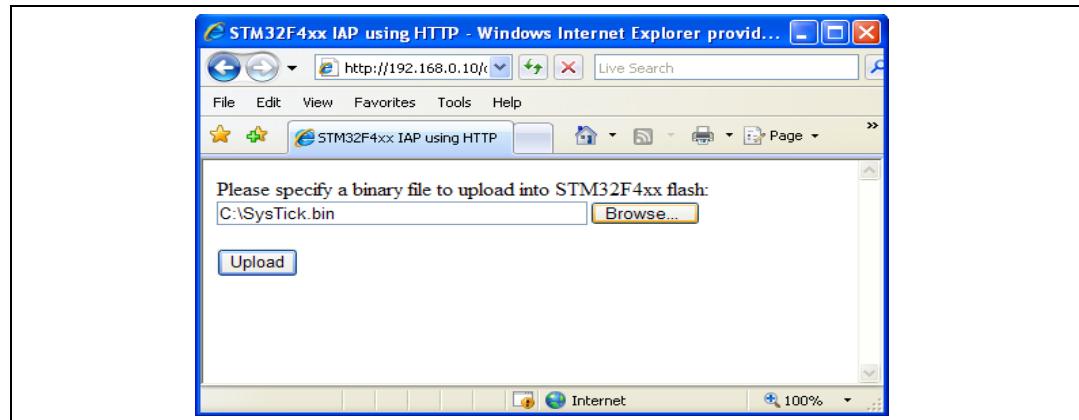
3.1 HTTP 文件上传概述

RFC1867 中定义了使用 HTTP 进行文件上传。该上传文件方法基于 HTML 表单。通常用 HTML POST 方法代替 GET 来发送原始二进制数据。

下面是一个实现基于表单的文件上传的 HTML 代码示例：

```
<form action ="/upload.cgi" enctype="multipart/form-data" method="post">
<p>Specify a binary file to upload into STM32F4xx Flash:
<br>
<input type="file" name="datafile" size="40">
</p>
<div>
<input type="submit" value="Upload">
</div></form>
```

图 5. 文件上传 HTML 表单的浏览器视图



按下**浏览**按钮选择一个二进制文件上传，然后按下**上传**按钮发送它。

根据文件大小，数据以连续的 TCP 段形式发送给 web 服务器。

注：发送文件数据前，Web 客户端发送包括文件名、内容长度等信息的 HTTP 头数据，其中一些信息必须由 Web 服务器进行解析。

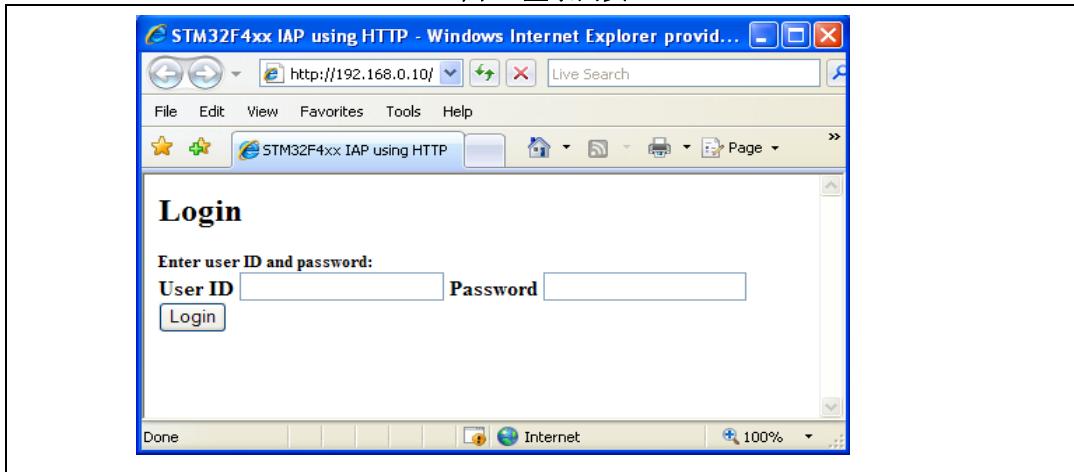
Web 客户端并不总是拥有相同的 HTTP 头格式。http Web 服务器必须能够处理这些差异。

3.2 对于 STM32F4xx，使用 HTTP 实现 IAP

该 IAP 实现包括一个基于 LwIP 栈的 HTTP Web 服务器。

当在浏览器上输入 STM32 IP 地址时，将显示登录网页如所示。（图 6）。此登录网页限制只有授权用户可以进行 IAP 文件上传。

图 6. 登录网页



输入正确的用户 ID 和密码（在 main.h 文件中预定义的）并点击登录按钮。然后载入一个文件上传页面（参见图 5）。

注：

1. 默认用户 ID 为：user，密码为 stm32。
2. 如果用户 ID 或密码不正确，登录网页将会重新加载。成功登录后，浏览选择要加载进 STM32 Flash 的二进制文件。
3. 确保该二进制文件的大小不超出 STM32 用户 Flash 区域的总大小。
4. 点击上传按钮（参见图 5）时，就向服务器发送了一个 POST 请求。此时服务器开始擦除所有的用户 Flash 区域并等待二进制文件的原始数据。然后接收到的数据将会被写入用户 Flash 区域。
5. 请注意，要接收的数据总长度是从传输开始时发送的 HTTP 头数据中提取出来的。
6. 在 IAP 操作结束时，一个网页指示 IAP 操作成功，网页上显示了一个可以复位 MCU 的按钮。

图 7. 文件上传完成

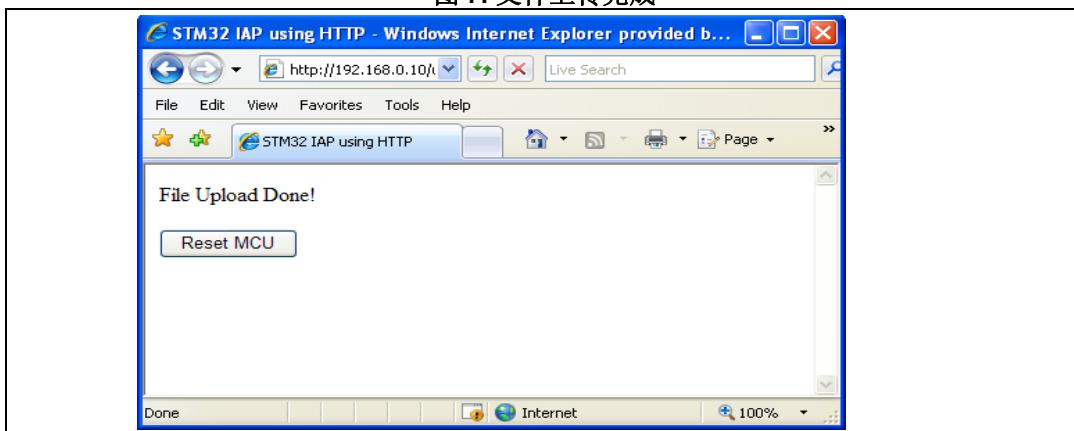
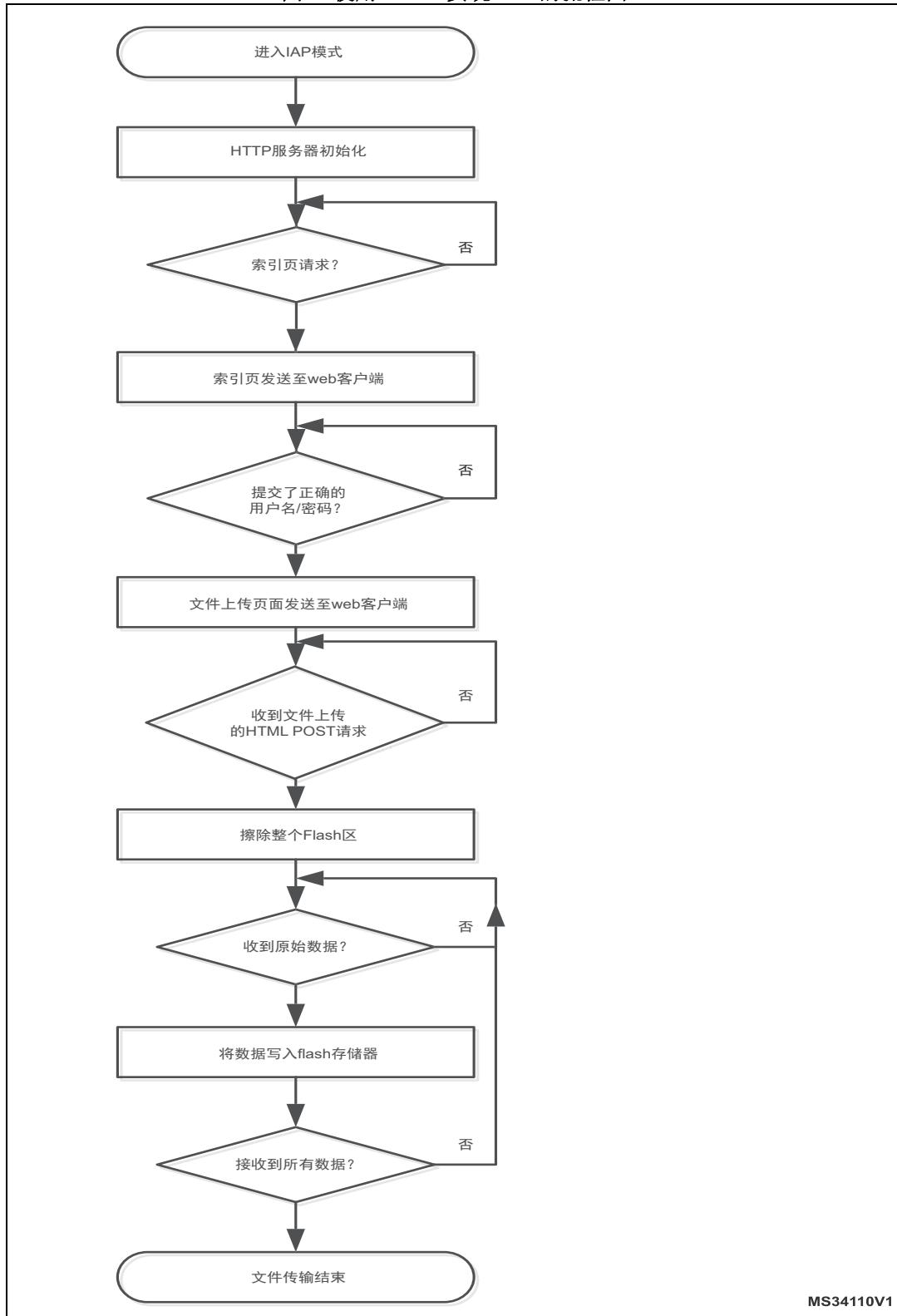


图 8 总结了使用 HTTP 实现 IAP 的方法。

图 8. 使用 HTTP 实现 IAP 的流程图



3.3 使用固件

为了通过 TFTP 对 IAP 进行测试, 请遵循如下步骤:

1. 在评估板上确保正确的跳线设置 (参见 [第 4.2 节](#))。
2. 在 `main.h` 文件中, 取消选项 `USE_IAP_HTTP` 的注释, 并且根据需求您还可以取消注释 / 注释其它选项, 比如 `USE_DHCP` 或 `USE_LCD`。
3. 重新编译固件。使用生成的 `map` 文件, 确保 IAP 代码区域 (从地址 `0x0` 开始) 和开始于 `USER_FLASH_FIRST_PAGE_ADDRESS` (在 `main.h` 中定义) 的用户 Flash 区域之间没有重叠。
4. 将程序下载到 `STM32F4xx Flash` 中并运行它。
5. 按住 **Key** 按钮的同时, 按下并释放 **Reset** 键, 进入 IAP 模式。
6. 若在 `main.h` 文件中定义了 `USE_LCD`, LCD 屏幕上会显示消息, 指示已经进入 IAP 模式。同样在使用 `DHCP` (`main.h` 文件中定义了 `USE_DHCP`) 的情况下, LCD 屏幕上也会显示一条消息, 指示 `DHCP IP` 地址分配的成功或失败。
7. 分配完 IP 地址后 (动态或静态地址), 用户可启动 IAP 进程。
8. 打开一个 web 客户端 (Mozilla Firefox 或 Microsoft Internet Explorer) 并输入 STM32 IP 地址。
9. 将会显示一个登录网页。在 **User ID** 字段中输入 “`user`” 并在 **Password** 字段中输入 “`stm32`”, 然后按 **Login** 按钮。
10. 然后会加载 `fileupload.html` 网页。选择一个要加载进 STM32 Flash 的二进制映像, 然后按 **Upload** 按钮开始 IAP 进程。
11. 如果定义了 LCD, IAP 操作的进度会显示在 LCD 上。
12. 在 IAP 操作结束时, 会加载一个新的网页, 指示文件上传操作成功。
13. 按 **RESET** MCU 按钮复位 MCU 来运行您刚才在 `STM32F4xx Flash` 中编写的程序。

注:

1. 在 LCD 使能的情况下, 如果出现连接问题, 则 LCD 屏幕上会显示错误消息, 指示连接失败。
2. 该软件已用下列 Web 客户端测试: Microsoft Internet Explorer 8 和 Mozilla Firefox 24。

4 环境

4.1 应用设置

4.1.1 PHY 接口配置

以太网外设与外部 PHY 接口，提供物理层通信。PHY 寄存器定义和定义语句位于 HAL 配置文件 “stm32f4xx_hal_conf.h” 中。

PHY 工作于 MII 和 RMII 两种模式；要选择所需的模式，用户必须在初始化以太网外设时，填写 “Init” 结构体的 “MedialInterface” 参数。

注：请参考您设备以太网 IAP 样例内提供的 *readme* 文件，以了解在支持的板子上可用的 PHY 接口模式。

4.1.2 MAC 和 IP 地址设置

默认 MAC 地址设为 00:00:00:00:00:02。若要更改此地址，请修改 *stm32f4xx_hal_conf.h* 文件中定义的六个字节。

默认 IP 地址设为：192.168.0.10。若要修改此地址，请修改 *main.h* 文件中定义的这六个字节。

4.2 评估板设置

在运行以太网 IAP 样例之前，请阅读相应的 *readme* 文件，以了解如何配置板子跳线，确保正常工作。

4.3 固件文件结构

以太网 IAP 样例源代码在 *Projects\STM324xx_EVAL\Applications\LwIP\LwIP_IAP* 下，其中 *STM324xx_EVAL* 指的是 STM32F4xx 评估板（例如 STM32F407/417 产品线为 STM324xG-EVAL）。[表 2](#) 介绍了示例源文件：

表 2. 文件结构

文件名	说明
main.c	主应用文件
main.h	主配置文件
httpserver.c/.h	HTTP 服务器实现
tftpserver.c/.h	TFTP 服务器实现
flash_if.c/.h	高级闪存访问函数
stm32f4xx_it.c/.h	中断处理程序
fsdata.c	HTML 文件作为 ROM 文件系统
lwipopts.h	LwIP 配置选项

表 2. 文件结构

文件名	说明
ethernetif.c.h	LwIP 和以太网驱动之间的接口
stm32f4xx_hal_conf	HAL 配置文件

注:

表中未列出所用的 *STM32Cube HAL*、*BSP* 库和 *LwIP* 栈的文件。

4.4 构建 IAP 映像

为了构建 IAP 映像（将用 IAP 软件进行加载），请确保：

1. 被编译和链接的固件必须从用户 Flash 区域起始地址开始运行（该地址应与 main.h 文件中的 `USER_FLASH_FIRST_PAGE_ADDRESS` 定义的地址相同）。
2. 向量表起始地址配置为用户 Flash 区域的起始地址。通过修改 `system_stm32f4xx.c` 文件中定义的 `VECT_TAB_OFFSET` 常量值，可配置向量表起始地址偏移。例如，将向量表起始地址位置设置为 `0x08020000`: `#define VECT_TAB_OFFSET 0x20000`
3. 编译后的软件大小不超出总的用户 Flash 区域。

5 结论

本用户手册说明了对于 STM32F4xx 微控制器，使用 STM32Cube HAL 驱动进行以太网在应用中编程（IAP）。

为支持 HTTP 和 TFTP 协议，提供了两种解决方案；它们两者都基于将 LwIP 栈作为 TCP/IP 通信的中间件组件。

附录 A FAQ

A.1 怎样选择静态或动态（DHCP）IP 地址分配

当“main.h”中的宏定义 `#define USE_DHCP` 被注释时，会为 STM32 微控制器分配静态 IP 地址（默认为 192.168.0.10，可在“main.h”文件中修改此值）。

若去掉宏 `#define USE_DHCP` 的注释，则启用 DHCP 协议，STM32 将作为 DHCP 客户端

A.2 当以太网网线断开时，应用如何处理

当网线断开时，以太网外设停止收发业务，网络接口关闭。如果使用了 LCD 控制器，则会显示一条消息，告知用户网线未连接，否则评估板上的红色 LED 灯会亮起。

当用户重新链接网线时，以太网外设再次工作，网络接口建立。如果使用了 LCD 控制器，则会显示一条消息，告知用户静态或动态分配的新 IP 地址，否则评估板上的黄色 LED 灯会亮起。

A.3 怎样将应用移植到不同的硬件

当使用另一个硬件平台时，必须检查 `HAL_ETH_MspInit()` 函数中以太网外设的 GPIO 配置，若应用需要更多 PPP 外设，则还需检查 `HAL_PPP_MspInit()` 或 `HAL_MspInit()`。

6 版本历史

表 3. 版本历史

日期	版本	变更
2014 年 3 月 28 日	1	初始版本
2015 年 2 月 5 日	2	更新了 前言 和 第 1 节: IAP 概述
2015 年 5 月 27 日	3	前言 更新并与 STM32Cube 概述 章节合并。 删除了有关 RMII 模式的注释，并对于支持的板子在 第 4.1.1 节: PHY 接口配置 中增加了注释。 删除了 第 4.2 节: 评估板设置 中专用评估板设置章节。 更新了 第 4.3 节: 固件文件结构 简介。

表 4. 中文版本历史

日期	版本	变更
2017 年 6 月 30 日	1	中文初始版本

重要通知 - 请仔细阅读

意法半导体公司及其子公司（“ST”）保留随时对 ST 产品和 / 或本文档进行变更、更正、增强、修改和改进的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2017 STMicroelectronics - 保留所有权利 2017