

1 简介

本应用笔记专为使用 STM32F407/STM32F417 微控制器的开发人员编写。它提供了如何使用 STM32F407/STM32F417 以太网通信接口实现在应用中编程 (IAP) 的解决方案。

有两种基于 LwIP TCP/IP 协议栈的解决方案：

- 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

目录

1	简介	1
2	IAP 概述	5
2.1	工作原理	5
2.2	使用 MCU 以太网接口实现 IAP	5
2.3	通过以太网在 STM32F407/STM32F417 上实现 IAP	6
2.3.1	使用 TFTP 实现 IAP 方法	6
2.3.2	使用 HTTP 实现 IAP 方法	6
3	使用 TFTP 实现 IAP	7
3.1	TFTP 概述	7
3.2	使用 TFTP 为 STM32F407/STM32F417 实现 IAP	8
3.3	使用软件	9
4	使用 HTTP 实现 IAP	10
4.1	HTTP 文件上传概述	10
4.2	使用 HTTP 在 STM32F407/STM32F417 上实现 IAP	12
4.3	使用软件	14
4.4	已知限制	14
4.4.1	二进制文件中添加的额外字节	14
5	环境	15
5.1	MAC 地址和 IP 地址设置	15
5.2	STM324xG-EVAL 电路板上的跳线设置	15
5.3	软件文件组成	16
5.4	代码长度测量	16
5.5	构建 IAP 映像	17
6	版本历史	18

表格索引

表 1.	TFTP 操作码	7
表 2.	跳线配置	15
表 3.	文件组成	16
表 4.	代码长度与配置选项	16
表 5.	文档版本历史	18

图片索引

图 1.	IAP 操作流程.....	5
图 2.	TFTP 数据包.....	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 表单的浏览器画面.....	10
图 6.	IE8 HTTP 头文件格式.....	11
图 7.	Mozilla Firefox HTTP 头文件格式.....	11
图 8.	登录 web 页面.....	12
图 9.	文件上传完成 web 页面.....	12
图 10.	使用 HTTP 实现 IAP 的流程图.....	13

2 IAP 概述

2.1 工作原理

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

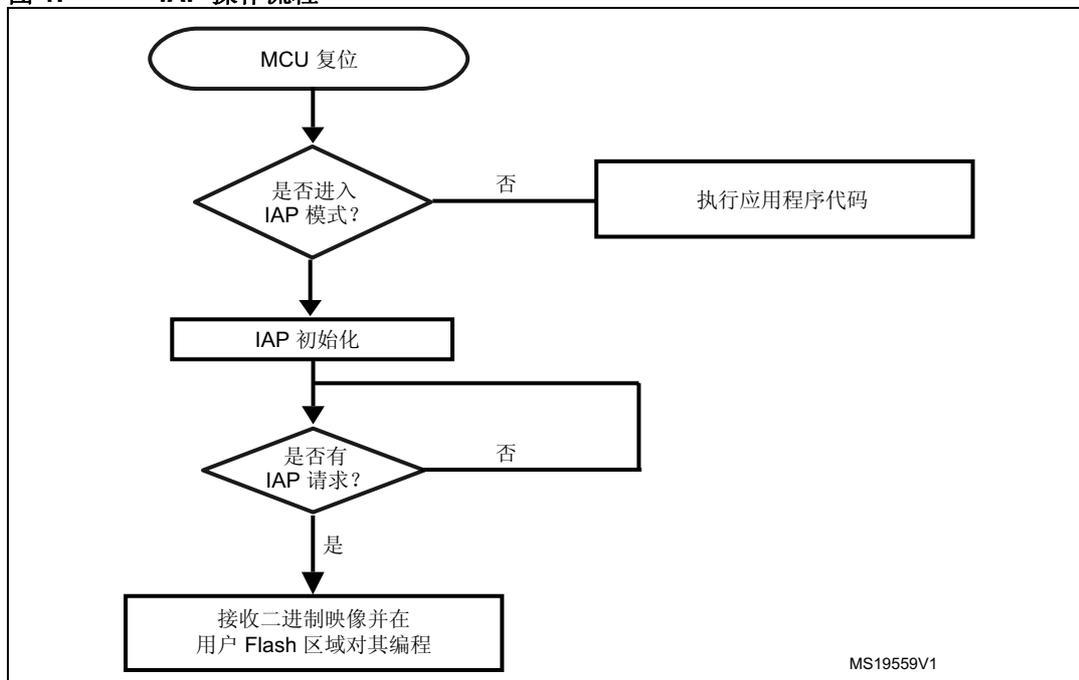
启动微控制器时，可以选择让其在以下任一模式运行：

- IAP 模式，用于执行 IAP 代码，
- 正常模式，用于执行应用程序代码。

无论是 IAP 代码还是应用程序代码都位于微控制器的内置 Flash 中。IAP 代码通常存储在 MCU Flash 的第一页，而用户应用程序代码则占据剩余的 Flash 区域。

图 1 介绍了 IAP 操作流程：

图 1. IAP 操作流程



2.2 使用 MCU 以太网接口实现 IAP

如果有以太网可用，则它通常是在嵌入式系统中实现 IAP 功能的首选接口。其优势包括：

- 高速通信接口 (10/100 Mb/s)
- 通过网络 (LAN 或 WAN) 进行远程编程
- 可以使用 FTP、TFTP、HTTP 等基于 TCP/IP 栈的标准应用协议实现 IAP

2.3 通过以太网在 STM32F407/STM32F417 上实现 IAP

本应用笔记将介绍两种使用以太网通信外在 STM32F407/STM32F417 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

这两种解决方案均基于 LwIP 栈 (v1.3.2)，它是轻量级的 TCP/IP 协议栈。

2.3.1 使用 TFTP 实现 IAP 方法

使用 TFTP 实现 IAP 的方法广泛应用于需要具有固件升级功能的嵌入式应用中（例如，嵌入式 Linux bootloader 中）。

TFTP 是一种在 UDP 传输层上执行的简单文件传输协议。此协议非常适合在局域网环境中使用。它基于客户端/服务器架构，在这种架构中，客户端会向服务器发出文件传输请求（读取或写入操作）。

为实现 IAP，需要在 LwIP 协议栈上实现一个简单的 TFTP 服务器，服务器只需处理来自 PC 的 TFTP 客户端的写入请求即可。

2.3.2 使用 HTTP 实现 IAP 方法

使用 HTTP 协议进行固件升级没有使用 TFTP 常见，但是在需要通过 Internet 进行远程编程时，这种解决方案就显得极为有用。这时，需要使用 TCP 传输协议来实现 http 服务。

HTTP 基于 TCP 协议运行，它提供了一种以 HTML 表单形式从 Web 客户端（Mozilla Firefox 或 Microsoft Internet Explorer）发送二进制文件的方式。这称为 HTTP 文件上传 (RFC 1867)。

本文档中的后续章节将详细介绍这两种 IAP 方法的实现，并会对如何使用软件进行说明。

3 使用 TFTP 实现 IAP

3.1 TFTP 概述

TFTP 是一种基于 UDP 的简单文件传输协议。文件传输由 TFTP 客户端发起，会向 TFTP 服务器发送读取或写入请求。服务器确认请求后，即开始进行文件数据传输。数据将以固定大小的块进行发送（例如每块含 512 个字节）。

必须在每个发出的数据块都得到接收方确认后，才可以发送下一个数据块。这种确认机制通过随各个数据块一同发送的块编号来实现。数据块小于固定块大小表示文件传输的结束。

图 2 描述了各种 TFTP 数据包的格式：

图 2. TFTP 数据包

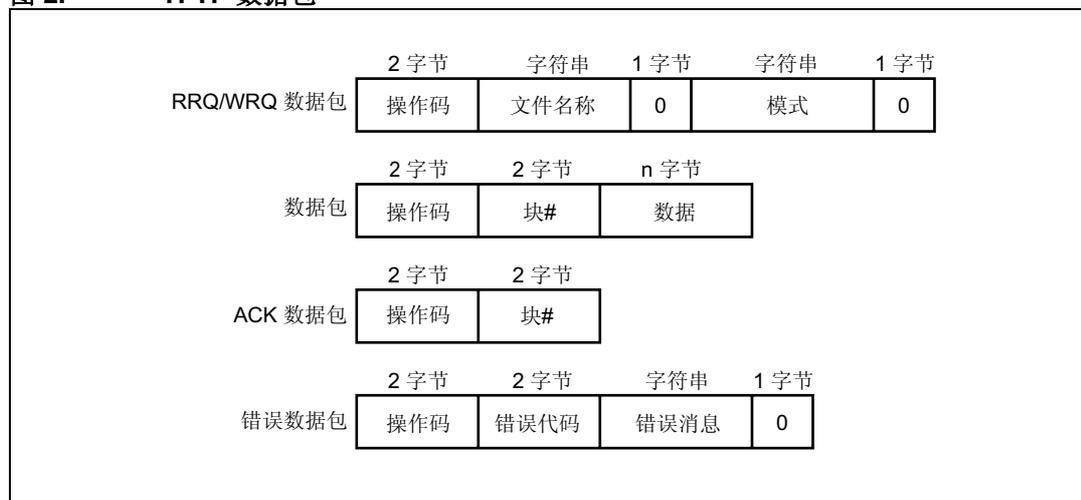


表 1 列出了 TFTP 操作码。

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	确认 (ACK)
0x5	错误

3.2 使用 TFTP 为 STM32F407/STM32F417 实现 IAP

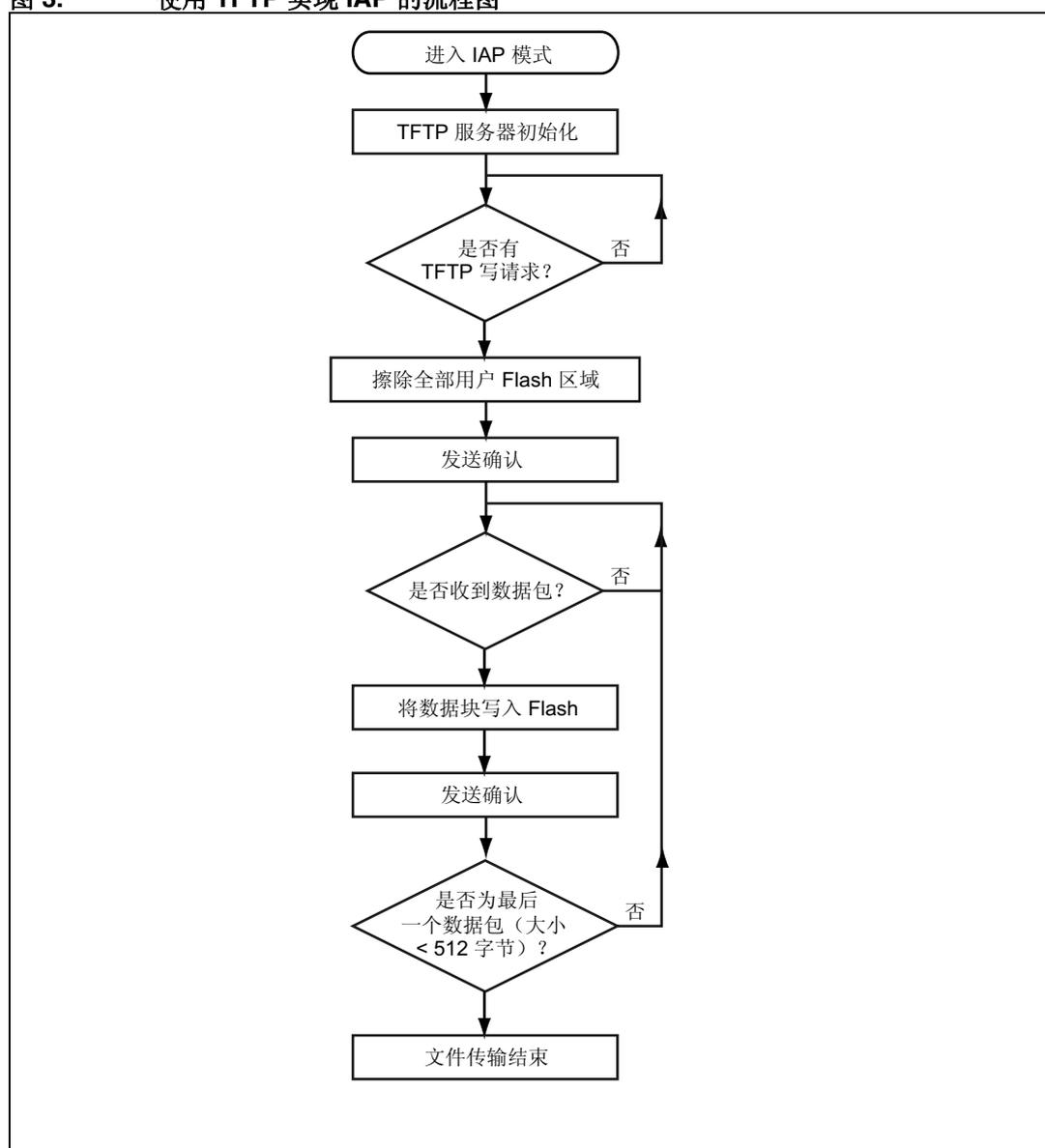
此 IAP 实现由基于 LwIP TCP/IP 栈的 TFTP 服务器组成。此服务器会对远程 TFTP 客户端 (PC) 发来的写请求做出响应。TFTP 读请求会被忽略。

TFTP 通常会将接收到的文件写入到文件系统，但是该服务器却并非如此，它会将接收到的数据块写入到 MCU Flash（用户 Flash 区域）。

注：在这个实现过程中，数据块大小固定为 512 个字节。

图 3 概述了使用 TFTP 实现 IAP 操作的过程。

图 3. 使用 TFTP 实现 IAP 的流程图



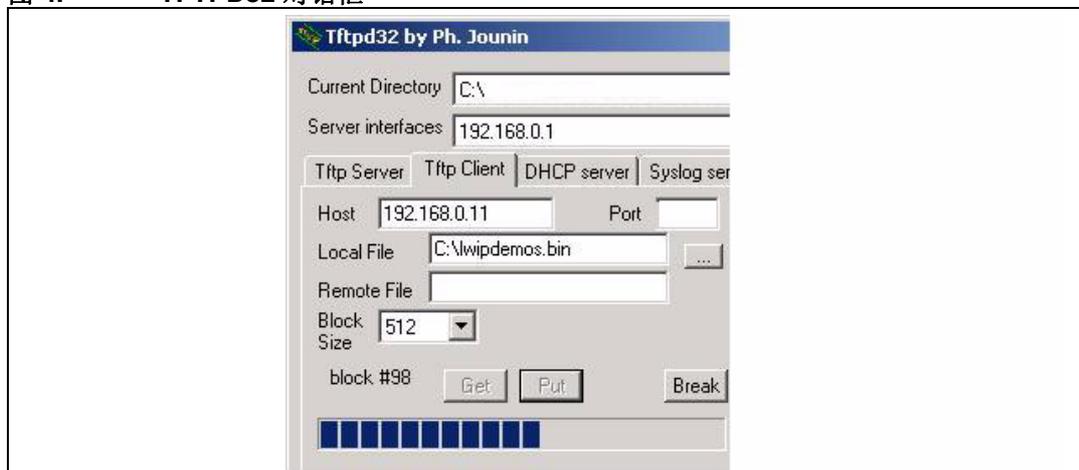
3.3 使用软件

要通过 TFTP 对 IAP 进行测试，需执行以下步骤：

1. 确保 STM324xG-EVAL 电路板上的跳线设置正确（参见表 2）。
2. 在 `main.h` 文件中，取消 `USE_IAP_TFTP` 选项的注释。另外，还可以根据需要取消注释/注释其它选项，例如 `USE_DHCP` 或 `USE_LCD`。
3. 重新编译软件。使用生成的映射文件，确保 IAP 代码区域之间没有重叠（从地址 `0x0` 开始），而且用户 Flash 区域从以下地址开始：`USER_FLASH_FIRST_PAGE_ADDRESS`（在 `main.h` 中定义）。
4. 在 STM32 Flash 中编写并运行软件程序。
5. 要进入 IAP 模式，需要在按住按键按钮的同时按下再释放复位按钮。
6. 如果在 `main.h` 文件中定义了 `USE_LCD`，则 LCD 屏幕会显示一条消息，指出已进入 IAP 模式。同样，如果使用的是 DHCP（在 `main.h` 中定义的 `USE_DHCP`），LCD 屏幕也会显示一条消息，指出 DHCP IP 地址分配是否成功。
7. 分配完 IP 地址后（可以是静态或动态地址），用户即可启动 IAP 流程。
8. 在 PC 侧，打开 TFTP 客户端（例如 TFTP32），然后配置 TFTP 服务器地址（TFTP32 中的主机地址）。
9. 浏览到要在 STM32 Flash 中加载的二进制映像（在 `/project/binary` 文件夹中提供了两个二进制映像作为示例）。
10. 单击 TFTP32 实用程序中的 **Put（写入）** 按钮，启动文件写请求。
11. 如果使能 `USE_LCD`，IAP 操作的进程会显示在 LCD 上。
12. 在 IAP 操作结束时，可以复位评估电路板并在 STM32 Flash 中运行刚刚编写的应用程序。

注：如果在使能 `USE_LCD` 时出现连接问题，LCD 屏幕上会显示一条错误消息，指示连接故障。

图 4. TFTP32 对话框



4 使用 HTTP 实现 IAP

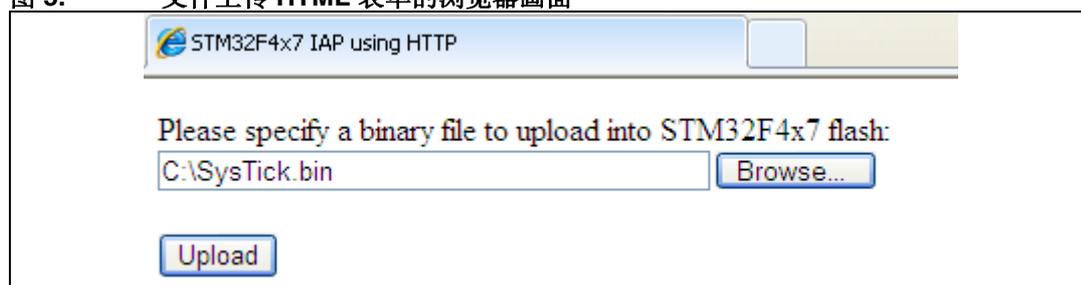
4.1 HTTP 文件上传概述

RFC1867 中定义了使用 HTTP 进行文件上传。此文件上传方法是基于 HTML 表单。发送原始二进制数据时，要使用 HTML POST 方法而不是 GET 方法。

以下是一个 HTML 代码示例，用于实现基于表单的文件上传：

```
<form action ="/upload.cgi" enctype="multipart/form-data" method="post">  
  <p>指定要上传到 STM32F4x7 Flash 的二进制文件：  
    <br>  
    <input type="file" name="datafile" size="40">  
  </p>  
  <div>  
    <input type="submit" value="Upload">  
  </div></form>
```

图 5. 文件上传 HTML 表单的浏览器画面



单击 **Browse (浏览)**，选择要上传的二进制文件，然后按下 **Upload (上传)** 按钮进行发送。根据文件大小，数据会以连续 TCP 分段的形式发送到 web 服务器。

*注：*在发送文件数据前，web 客户端会首先发送 HTTP 头文件数据，其中包含诸如文件名称和内容长度等信息，web 服务器必须对其中的一些信息进行解析。

Web 客户端使用的 HTTP 头文件格式并不总是相同。图 6 显示的是 Internet Explorer 在 POST 请求中的 HTTP 头文件格式。图 7 显示的是 Mozilla Firefox 的 HTTP 头文件格式。

http web 服务器必须能够处理这些不同的格式。

图 6. IE8 HTTP 头文件格式

```

Hypertext Transfer Protocol
POST /upload.cgi HTTP/1.1\r\n
Accept: image/gif, image/jpeg, image/pjpeg, application/x-shockwave-flash, applic.
Referer: http://192.168.0.10/checklogin.cgi\r\n
Accept-Language: en-us\r\n
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET CLR 1.1.4322;
Content-Type: multipart/form-data; boundary=-----7db1672b29603a8\r\n
Accept-Encoding: gzip, deflate\r\n
Host: 192.168.0.10\r\n
Content-Length: 1969\r\n
[Content length: 1969]
Connection: keep-alive\r\n
Cache-Control: no-cache\r\n
\r\n
MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----
[Type: multipart/form-data]
First boundary: -----7db1672b29603a8\r\n
Encapsulated multipart part: (application/octet-stream)
Content-Disposition: form-data; name="datafile"; filename="STM324xG_EVAL_systick.bin"\r\n
Content-Type: application/octet-stream\r\n\r\n
Media Type
Last boundary: \r\n-----7db1672b29603a8--\r\n

```

图 7. Mozilla Firefox HTTP 头文件格式

```

Hypertext Transfer Protocol
POST /upload.cgi HTTP/1.1\r\n
Host: 192.168.0.3\r\n
User-Agent: Mozilla/5.0 (windows NT 5.1; rv:7.0.1) Gecko/20100101 Firefox/7.0.1\r\n
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
Accept-Language: en-us,en;q=0.5\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
Connection: keep-alive\r\n
Referer: http://192.168.0.3/checklogin.cgi\r\n
Content-Type: multipart/form-data; boundary=-----41184676334\r\n
Content-Length: 1961\r\n
[Content length: 1961]
\r\n
MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----
[Type: multipart/form-data]
First boundary: -----41184676334\r\n
Encapsulated multipart part: (application/octet-stream)
Content-Disposition: form-data; name="datafile"; filename="STM324xG_EVAL_systick.bin"\r\n
Content-Type: application/octet-stream\r\n\r\n
Media Type
Last boundary: \r\n-----41184676334--\r\n

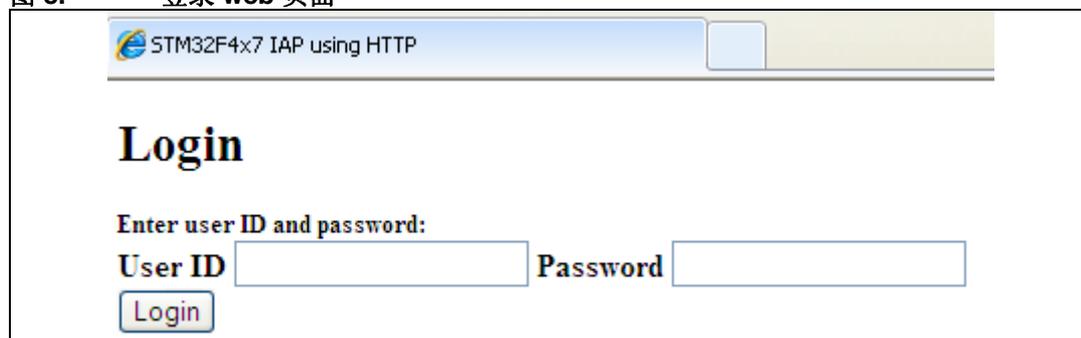
```

4.2 使用 HTTP 在 STM32F407/STM32F417 上实现 IAP

此 IAP 实现由基于 LwIP 栈的 HTTP Web 服务器组成。

在浏览器中输入 STM32 的 IP 地址后，将显示登录 web 页面（图 8）。此登录 web 页面只有已获授权的用户才能使用 IAP 文件上传功能。

图 8. 登录 web 页面



输入正确的 **User ID**（用户 ID）和 **Password**（密码）（*main.h* 文件中已预先定义），然后单击 **Login**（登录）按钮。这将加载文件上传 web 页面（参见图 5）。

- 注：
- 1 默认的 **User ID**（用户 ID）为：*use*，**Password**（密码）为 *stm32*。
 - 2 如果 **User ID**（用户 ID）或 **Password**（密码）不正确，登录 web 页面会重新加载。

登录成功后，浏览并选择要上传到 STM32 Flash 的二进制文件。

注： 确保二进制文件大小不超过 STM32 用户 Flash 区域的总容量。

单击 **Upload**（上传）按钮后（参见图 5），将向服务器发出 POST 请求。这时，服务器开始擦除用户 Flash 区域的全部内容，等待接收二进制文件原始数据。然后将接收到的数据写入用户 Flash 区域。

注意，要接收的数据的总长度信息将从传输开始时发出的 HTTP 头文件数据中提取。

在 IAP 操作结束后，web 页面将指示 IAP 操作成功，同时显示一个可用于复位 MCU 的按钮。

图 9. 文件上传完成 web 页面

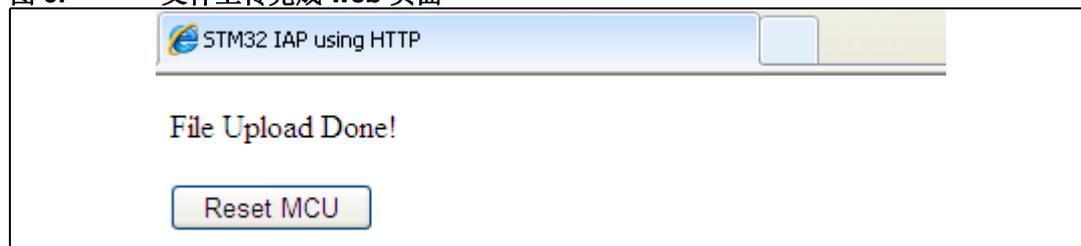
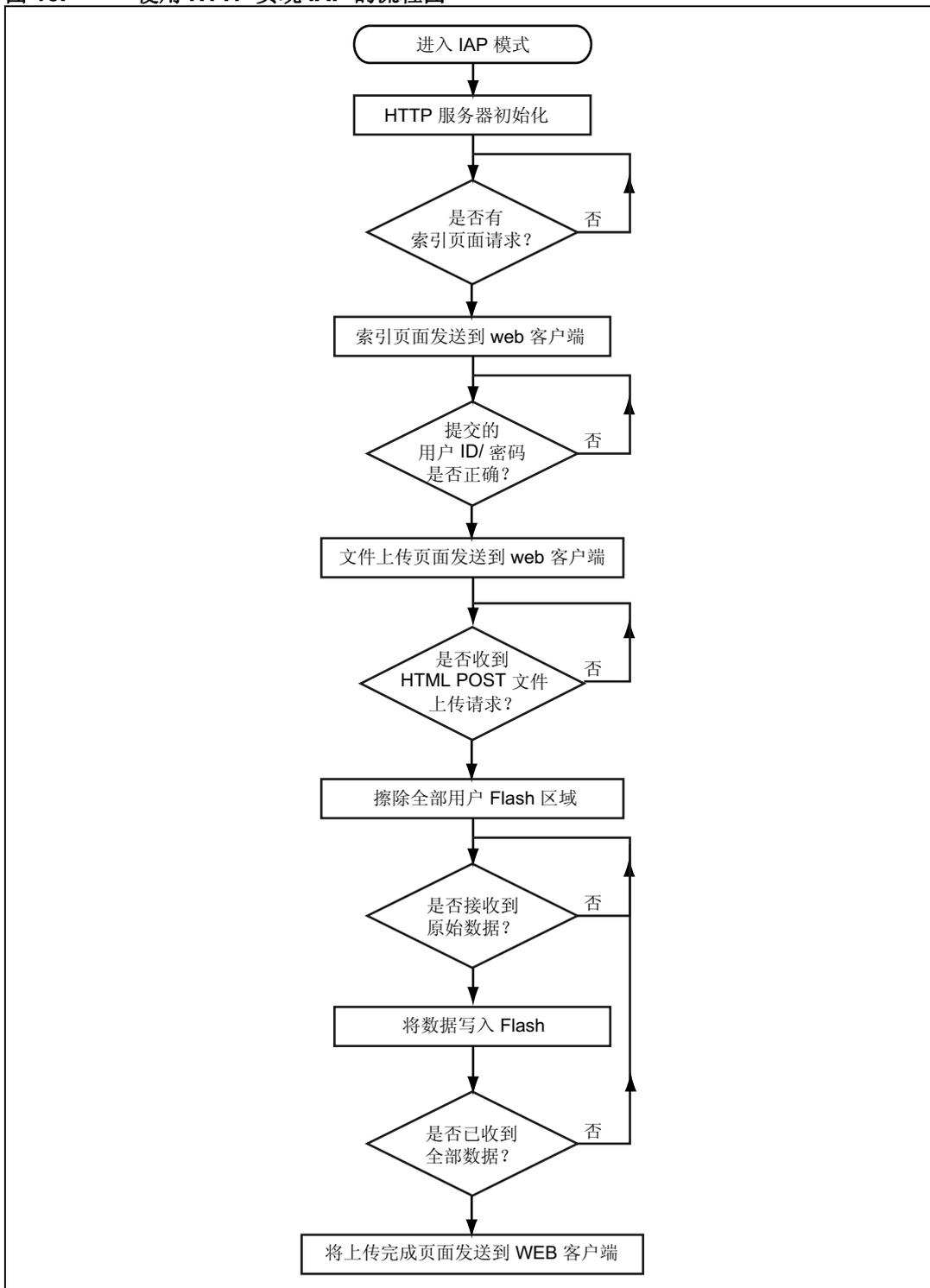


图 10 对使用 HTTP 实现 IAP 方法进行了总结。

图 10. 使用 HTTP 实现 IAP 的流程图



4.3 使用软件

要使用 HTTP 对 IAP 进行测试，需执行以下步骤：

1. 确保 STM324xG-EVAL 电路板上的跳线设置正确（参见表 2）。
2. 在 *main.h* 文件中，取消选项 USE_IAP_HTTP 的注释，另外还可以根据需要取消注释/注释其它选项，例如 USE_DHCP 或 USE_LCD。
3. 重新编译软件。使用生成的映射文件，确保 IAP 代码区域之间没有重叠（从地址 0x0 开始），而且用户 Flash 区域从以下地址开始：USER_FLASH_FIRST_PAGE_ADDRESS（在 *main.h* 中定义）。
4. 在 STM32 Flash 中编写并运行软件程序。
5. 要进入 IAP 模式，需要在按住 **按键** 按钮的同时按下再释放 **复位** 按钮。
6. 如果在 *main.h* 文件中定义了 USE_LCD，则 LCD 屏幕会显示一条消息，指出已进入 IAP 模式。同样，如果使用的是 DHCP（在 *main.h* 中定义的 USE_DHCP），LCD 屏幕也会显示一条消息，指出 DHCP IP 地址分配是否成功。
7. 分配完 IP 地址后（可以是静态或动态地址），用户即可启动 IAP 流程。
8. 打开 web 客户端（Mozilla Firefox 或 Microsoft Internet Explorer），输入 STM32 IP 地址。
9. 会显示登录 web 页面。在 **User ID（用户 ID）** 字段中输入“user”，在 **Password（密码）** 字段中输入“stm32”，然后按下 **Login（登录）** 按钮。
10. 这将加载 *fileupload.html* web 页面。浏览到要在 STM32 Flash 中加载的二进制映像，然后按下 **Upload（上传）** 按钮，启动 IAP 进程。
11. 如果使能 USE_LCD，IAP 操作的进程会显示在 LCD 上。
12. IAP 操作结束后，将加载新的 web 页面，指示文件上传操作已成功完成。
13. 可以按下 **RESET MCU（复位 MCU）** 按钮复位 MCU，然后在 STM32 Flash 中运行刚刚编写的应用程序。

- 注：
- 1 如果在使能 USE_LCD 时出现连接问题，LCD 屏幕上会显示一条错误消息，指示连接故障。
 - 2 使用以下 Web 客户端对软件进行测试：Microsoft Internet Explorer 8 和 Mozilla Firefox 7.0.1。

4.4 已知限制

4.4.1 二进制文件中添加的额外字节

Internet 浏览器（Microsoft Internet Explorer 或 Mozilla Firefox）会在上传的二进制文件的末尾添加一个随机边界标记（根据 RFC 1521 的规定，此标记不得超过 72 个字节）。在最新的 IAP 软件版本中，并没有删除此边界标记，而是在空间足够的情况下将其存储在 Flash 中。如果没有足够空间，则不会在 Flash 中写入额外字节，也不会返回错误。

5 环境

5.1 MAC 地址和 IP 地址设置

在 *main.h* 文件中对 MAC 地址和 IP 地址进行了定义。

默认的 MAC 地址固定为：00:00:00:00:00:02

IP 地址可以设置为静态地址，也可以设置为由 DHCP 服务器分配的动态地址。默认的静态 IP 地址为：192.168.0.10

可以通过在 *main.h* 文件中使能 USE_DHCP 来选择 DHCP 模式。

注意，如果选择由 DHCP 来配置 IP 地址，但应用程序没有找到它之前在网络中所连接的 DHCP 服务器，则 IP 地址将自动设置为静态地址 (192.168.0.10)。

5.2 STM324xG-EVAL 电路板上的跳线设置

要运行软件，请按照表 2 中所示来配置 STM324xG-EVAL 电路板。

在 project\inc 文件夹下的 *main.h* 文件中选择 MII 和 RMII 配置。

例如，要选择 RMII 模式：

```
//#define MII_MODE
#define RMII_MODE
```

如果在 *main.h* 文件中对 MII_MODE 和 PHY_CLOCK_MCO 都进行了定义，则在 MII 模式下，PHY 时钟可以来自外部晶振或者由 STM32 通过 MCO 提供。

- 注：
- 1 对于 RMII 模式，必须通过在位于 CN3 下方的 U3 封装中焊接一个 50 MHz 的振荡器（参见 SM7745HEV-50.0M 或相应手册）并断开 JP5 上的跳线来提供 50 MHz 的时钟。这个振荡器并未随电路板一同提供。有关详细信息，请参见 STM3240G-EVAL 评估电路板用户手册 UM1461。
 - 2 在本文档中出现的所有 STM324xG-EVAL 电路板均指 STM3240G-EVAL 和 STM3241G-EVAL 电路板。

表 2. 跳线配置

跳线	MI I 模式配置	RMII 模式配置
JP5	1-2: 通过外部晶振提供 25 MHz 时钟 2-3: 通过 PA8 上的 MCO 提供 25 MHz 时钟	不适用
JP6	2-3: 使能 MII 接口模式	1-2: 使能 RMII 接口模式
JP8	断开: 选择 MII 接口模式	闭合: 选择 RMII 接口模式

5.3 软件文件组成

表 3 介绍了项目源文件：

表 3. 文件组成

文件名称	说明
main.c	主应用程序文件
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	高级别 Flash 访问函数
netconf.c/.h	高级别以太网接口函数
stm32f4x7_eth_bsp.c/.h	STM32F4x7 以太网硬件配置
stm32f4x7_it.c/.h	中断处理程序
fsdata.c	作为 ROM 文件系统的 HTML 文件
lwipopts.h	LwIP 配置选项

注：表格中没有列出标准固件库和 LwIP 栈中所用的文件。

5.4 代码长度测量

表 4 中提供了使用 main.h 文件中不同配置选项所得到的代码长度测量结果。

表 4. 代码长度与配置选项

代码长度（字节）	USE_IAP_TFTP	USE_IAP_HTTP	USE_LCD	USE_DHCP
19796	X			
22956	X		X	
28500	X		X	X
31484		X		
33884		X	X	
39820		X	X	X
41196	X	X	X	X

注：软件使用 IAR EWARM v6.21.3 进行编译，并对代码长度进行了高度优化。

5.5 构建 IAP 映像

为了构建 IAP 映像（将会使用 IAP 软件加载），应确保以下几点：

1. 编译/链接的软件必须从用户 Flash 区域的起始地址开始运行（此地址应与 *main.h* 的 `USER_FLASH_FIRST_PAGE_ADDRESS` 中所定义地址相同）。
2. 将向量表的起始地址配置为用户 Flash 区域的起始地址。

可以通过以下两种方法配置向量表的基本偏移量：

- a) 在应用程序代码中，使用 *misc.h/c* 驱动程序的 `NVIC_SetVectorTable` 函数来重新定位应用程序加载地址的向量表。

例如，将向量表基本位置设置为 `0x08010000`：

```
NVIC_SetVectorTable(NVIC_VectTab_FLASH, 0x10000);
```

- b) 通过修改 *system_stm32f4x7.c* 文件中定义的 `VECT_TAB_OFFSET` 常量的值。

例如，将向量表基本位置设置为 `0x08010000`：

```
#define VECT_TAB_OFFSET 0x10000
```

3. 编译后的软件大小不超过用户 Flash 区域的总容量。

在“使用 USART 实现 STM32F40x/STM32F41x 应用编程 (AN3965)”中包括一个软件示例，介绍如何使用预配置项目来构建要通过 IAP 加载的应用程序。

6 版本历史

表 5. 文档版本历史

日期	版本	变更
2011 年 10 月 21 日	1	初始版本。

请仔细阅读：

中文翻译仅为方便阅读之目的。该翻译也许不是对本文档最新版本的翻译，如有任何不同，以最新版本的英文原版文档为准。

本档中信息的提供仅与ST产品有关。意法半导体公司及其子公司（“ST”）保留随时对本档及本文所述产品与服务进行变更、更正、修改或改进的权利，恕不另行通知。

所有ST产品均根据ST的销售条款出售。

买方自行负责对本文所述ST产品和服务的选择和使用，ST概不承担与选择或使用本文所述ST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。如果本档任何部分涉及任何第三方产品或服务，不应被视为ST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ST的销售条款中另有说明，否则，ST对ST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。

意法半导体的产品不得应用于武器。此外，意法半导体产品也不是为下列用途而设计并不得应用于下列用途：（A）对安全性有特别要求的应用，例如，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，且/或（D）航天应用或航天环境。如果意法半导体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，即使采购商向意法半导体发出了书面通知，采购商仍将独自承担因此而导致的任何风险，意法半导体的产品设计规格明确指定的汽车、汽车安全或医疗工业领域专用产品除外。根据相关政府主管部门的规定，ESCC、QML或JAN正式认证产品适用于航天应用。

经销的ST产品如有不同于本档中提出的声明和/或技术特点的规定，将立即导致ST针对本文所述ST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ST的任何责任。

ST和ST徽标是ST在各个国家或地区的商标或注册商标。

本档中的信息取代之前提供的所有信息。

ST徽标是意法半导体公司的注册商标。其他所有名称是其各自所有者的财产。

© 2013 STMicroelectronics 保留所有权利

意法半导体集团公司

澳大利亚 - 比利时 - 巴西 - 加拿大 - 中国 - 捷克共和国 - 芬兰 - 法国 - 德国 - 中国香港 - 印度 - 以色列 - 意大利 - 日本 - 马来西亚 - 马耳他 - 摩洛哥 - 菲律宾 - 新加坡 - 西班牙 - 瑞典 - 瑞士 - 英国 - 美国

www.st.com

