

五木 | 作品

嵌入式网络 那些事

STM32
物联实战

朱升林 欧阳骏 杨晶 著
粟思科 审

- ◎ 骨灰级一线嵌入式专家“五木”呕心力作
- ◎ 基于STM32实战环境，直观感受各层协议在嵌入式设备组网过程中的实际应用，学习TCP/IP不再是纸上谈兵
- ◎ 最实用的STM32学习资料，最新鲜的实战例程，轻松诙谐，简单直白
- ◎ 从协议栈原理及使用方法，到编程框架，讲解透彻
- ◎ 从性能与稳定性测试，到问题定位与调试，干货多多



中国水利水电出版社
www.waterpub.com.cn

嵌入式网络那些事—— STM32 物联实战

朱升林 欧阳骏 杨 晶 著

栗思科 审



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书面向网络 TCP/IP 协议初学者以及大量嵌入式网络开发人员,从当下流行的嵌入式网络协议栈 LwIP 的源代码入手,详细讲解 TCP/IP 协议的各层机制及其实现原理。同时,通过搭建简单的实验环境,本书还详细阐述 LwIP 在嵌入式设备组网过程中的移植过程、应用编程案例、编程框架、注意事项等。

本书共分为四大部分,第一部分讲解基于 STM32 平台下 LwIP 的移植过程,其中包括了 LwIP 代码组织结构与特点,以及 LwIP 裸机移植通用流程;第二部分,以 LwIP 源代码为基础,以网络数据包在协议栈内部的递交顺序做导向,结合 TCP/IP 协议基本知识,依次讲解 LwIP 内核各个模块的实现机制,包括数据包管理、网络接口管理、ARP、IP、ICMP、UDP、TCP 等;第三部分,讲解 LwIP 上层 API 实现机制,以及常见的网络应用程序结构以及编程方法,以及 LwIP 其他高级应用,如 IGMP、DNS、DHCP 等;第四部分,讲解内核调试、协议栈稳定性和性能、并发服务器设计等,以及 LwIP 在具体项目中的应用和实践过程,同时还包括了 LwIP 在实际应用中常见的问题及解决方法。

本书可作为 TCP/IP 学习者的快速入门及精通的指导书籍,还可作为广大嵌入式网络开发人员的参考手册。

本书附带源代码,读者可以从中国水利水电出版社网站和万水书苑上下载,网址为:
<http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。

图书在版编目 (C I P) 数据

嵌入式网络那些事. STM32物联实战 / 朱升林, 欧阳
骏, 杨晶著. — 北京: 中国水利水电出版社, 2015. 7
ISBN 978-7-5170-3312-7

I. ①嵌… II. ①朱… ②欧… ③杨… III. ①计算机
网络—通信协议 IV. ①TN915.04

中国版本图书馆CIP数据核字(2015)第140805号

策划编辑: 周春元 责任编辑: 张玉玲 加工编辑: 孙丹 封面设计: 李佳

书 名	嵌入式网络那些事——STM32 物联实战
作 者	朱升林 欧阳骏 杨晶 著 粟思科 审
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路1号D座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658 (发行部)、82562819 (万水) 北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市铭浩彩色印装有限公司
规 格	184mm×240mm 16开本 31印张 687千字
版 次	2015年7月第1版 2015年7月第1次印刷
印 数	0001—3000册
定 价	68.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前言

两三年前，当整个世界都在大力倡导物联网、宣扬物联网改变世界的时候，很多人还一直觉得它是很虚幻的东西。而如今，物联网技术却实实在在地出现在了我们的生活中。物联网涉及到一系列广泛的技术，并没有完全统一的标准，其本质是所有物体通过一定的方式连入互联网，能够共享自身的一些信息，这些信息或为用户使用，或成为用户远程控制设备的基础。物联网当前已经存在了多种纯熟的入网技术，比如 Wi-Fi、以太网、ZigBee、GPRS/3G/4G 等。不同的应用场合，根据产品成本和应用特性的考虑，设备可以选择不同的入网方法。

当前在物联网领域，既有像华为这样的世界级大公司，也有像 Sigfox 这样的新型创业型公司。物联网的一个重要方向是上十亿百亿的小物体的互联，这些小设备入网并不以大数据量交换为目的，它们只是简单地上报自身状态，或者使用传感器完成某些简单检测功能，比如**垃圾箱需要入网**，在适当的时候上报自身的储存状态；**售货机需要入网**，以上报售货状态，是否需要补货；**防盗门需要入网**，可随时上报异常开关状态；**珍惜动物需要入网**，方便对它们进行跟踪保护，以免遭到非法捕杀等。类似于这样的小型设备入网需求不胜枚举，它们并不需要太大的数据量通信，或许每小时几百字节的数据就足以满足应用的要求。在这种需求下，使用传统的入网方法无论从成本上还是从实现上都是难以接受的（**Wi-Fi 的有效范围太小，移动网络价格太贵，功耗太大**）。“物体的互联需要一张专门的无线互联网”，这是 Sigfox 公司提出的一个观点，它们正致力于为所有小物体入网构建一张专有的网络，这张网络可以和目前的 2G/3G/4G 网络相匹敌。Sigfox 的设计中有一种基于超窄带技术的物联网基站，它能向设备提供超远距离的无线互联（10~20 公里），以牺牲数据通信速率为前提，设备的无线功率损耗将降至最低，据 Sigfox 透露这样一款小型设备在一块普通电池的供应下能持续工作数年之久，而设备的每个模块成本可以降到几美元，每年的设备连接维护费也仅为 1 美元多一点。

Sigfox 成功地解决了小型设备入网的功耗和成本问题，为世界上数以亿计的设备连入互联网提供了完美的解决方案。“成为物体而非人的运营商”是 Sigfox 目前的发展方向，目前该公司已将技术提交给了欧洲电信标准协会，希望将其专有技术变成标准。值得指出的是，华为也正致力于远距离低功耗无线通信技术的研发，这点同 Sigfox 的技术不谋而合，这也许是未来

小物体互联发展的热点。或许在不久的将来，我们就会见证另一个中国移动、中国电信的诞生。

以此为契机，越来越多的小设备会连入互联网，尽管在物理层上各种入网方案存在差别，但在这之上，嵌入式设备入网需要使用标准的 TCP/IP 协议，目前业界可以使用的 TCP/IP 协议栈较少，嵌入式设备入网比较流行的解决方案主要是使用硬件 TCP/IP 芯片或者收费 TCP/IP 协议栈软件，这两种方案一定程度上增加了设备的成本，使用灵活性也受到一定限制，进而阻碍了嵌入式网络产品的大力发展。

LwIP 是 TCP/IP 协议的一种开源实现，具有丰富完善的功能，最新版本还支持 DHCP、DNS 等服务，适合于在 32 位的嵌入式处理器上使用。LwIP 作为一款开源软件，可免费使用，能作为设备入网的一种比较完美的解决方案，由于其代码量小，特别适合于小型嵌入式领域使用。目前 LwIP 在国内已经有了一定的市场占有率，但是相关的资料仍然缺乏，缺少专业的书籍对协议栈原理、使用方法、编程框架、性能与稳定性测试、问题定位与调试等方面做详细说明，这影响了 LwIP 在物联网领域的广泛使用以及大量开发者进入嵌入式网络领域的脚步。

笔者的《嵌入式网络那些事：LwIP 协议深度剖析和实战演练》一书在一定程度上满足了开发者对 LwIP 相关知识的需求，在市场上也有很好的销量，得到了读者的好评。但是这本书还是存在一定的不足：一是书的开发平台目前已经停产，由于不是目前最流行的嵌入式处理器，因此该书的实验移植过程对很多读者来说理解有困难；第二，该书更侧重于 LwIP 理论的讲解，以及对 TCP/IP 协议原理深入的学习，书中的实验相对较简单，这并不能满足广大开发者以及学校实验环境的需求。因此，笔者一直筹划推出本书作为第二版，主要在以下几方面做了改进：首先会在目前流行的 STM32 处理器上完成所有实验与测试，同时在应用上加入更多丰富实用的实战例程，并且对于很多编程方法抽象出程序框架，开发者只要在这个框架基础上做简单的改动，便可以完成自己的应用程序；其次对于有一定经验的 LwIP 开发者，可能更加关注于内核调试、协议栈稳定性和性能相关内容，本书也从这两方面着手，提出了一定的实验和方法，满足广大开发者的需求；再次，为了方便初学者入门，本书丰富了 socket 标准接口编程方面的原理和实践内容，同时增加了并发服务器设计的相关知识，可以更好地满足如今在市场上越来越复杂的联网需求；最后，笔者从亲自参与并设计的一款物联网网关入手，从原理设计、需求分析、程序架构设计、程序调试等方面详细介绍 LwIP 在现实项目中的实战经验，希望可以给广大物联网开发工程师带来一定的参考意义。

写本书的目的可以归纳为三点：第一，给需要使用 LwIP 的人一个借鉴和解决问题的方法，这也是当初《嵌入式网络那些事：LwIP 协议深度剖析和实战演练》一书的写作初衷；第二，也是最基础的，写给那些需要学习 TCP/IP 的同学，TCP/IP 的重要性就不在这里多说了，遥想笔者当年毕业找工作之季，辗转于各大公司的招聘面试场所，TCP/IP 真的是面试官最青睐的问题，笔者也恰好因为在这方面有着独到的理解，因此顺利拿下了国内外众多知名企业的 offer，所以本书也想告诉那些正在努力的同学，学习 TCP/IP 是绝对值得的，这是一条康庄大道；第三，也是最重要的，嵌入式网络离不开实践，本书以目前流行的 STM32 处理器作为实践平台，读者能在实验平台上实际运行、调试网络应用程序，更可以修改协议栈内核，修改的

每一行代码都能在实验板上看出实际效果。

网络编程大神们可以绕道了，这里没有 IPsec，没有 SSL，没有并发服务器，没有每秒上亿次的服务请求，我们玩的是低端玩意；神秘黑客们也可以绕道了，这里没有系统漏洞，没有文件系统，没有账号，玩不转木马，您都不屑于和我们较真。本书的使用者定位在嵌入式初学者、网络初学者以及对 LwIP 应用有兴趣或疑问的同学，你们一定可以在本书中找到想要的东西。

本书背景

TCP/IP 技术是目前最流行的 IT 技术之一，也是当代互联网及相关技术应用、发展的基石。近年来，TCP/IP 技术在各大高校俨然已成为学习和研究的热点；而另一方面，TCP/IP 知识也成了各 IT 企业招聘时关注的重点问题。对于广大 TCP/IP 初学者来说，要真正掌握并精通 TCP/IP 协议其实是有难度的：首先，TCP/IP 是个非常庞大的协议簇，其源代码一般只存在于各个大型操作系统内核中，协议的整体结构对于使用者来说不可见，学习者只能使用系统内部提供的 API 完成网络编程功能，但对协议内部的实现细节并不明了，这样 TCP/IP 的学习者只能停留在上层应用，如 Socket 编程的层次，而事实上，了解协议内部运作细节是编写高效可靠应用程序的基础；其次，目前市场上针对 TCP/IP 的书籍较少，而与 TCP/IP 协议内部机制相关的实验书籍就更加缺乏了，读者只能从书本上学到一些关于 TCP/IP 的基础理论知识，但很难将这些理论用于实际编程实践中，因为系统内核对于读者来说并不可见。

目前，嵌入式产品连入互联网已成了越来越多企业关注的热点，这使得在嵌入式产品上移植 TCP/IP 协议成为一种必然的趋势。LwIP 是应用于嵌入式领域的一款开源 TCP/IP 协议，它的实现同 BSD 的实现有很大的相似性，可以作为 TCP/IP 协议的典型代表，其功能完备，除了实现 TCP/IP 的基本通信功能外，其新版本还支持 DNS、SNMP、DHCP、IGMP 等高级应用功能。当前网络上关于 LwIP 的讨论很多，但市场上却鲜见相关书籍。

本书特点

理论与实践结合是本书编写的一个重要出发点，不但让读者掌握 TCP/IP 理论知识，而且让读者通过实验深入地理解并运用这些知识，达到学以致用目的。首先，整个协议栈对读者来说全部可见，这样读者能对 TCP/IP 协议有宏观的把握，摒弃掉 TCP/IP 以往神秘的面纱；本书以网络数据包在协议栈内部的递交顺序为编写导向，思路明显，条理清楚，语言精简幽默，更利于读者对 TCP/IP 协议基本知识的理解；最后，本书内容大部分建立在读者开发经验之上，其中包括了很多 LwIP 在实际应用中遇到的问题及其解决方法，这对大部分刚接触 LwIP 的嵌入式开发人员来说，有很大的指导意义。

概括来讲，本书具有如下特点：

轻松诙谐的语言、简单直白的讲解能让读者摆脱阅读时的枯燥感觉，技术学习也可以很轻松。从 LwIP 协议栈移植到网络应用程序设计，内容由浅入深，知识点覆盖全面，特别适合于嵌入式初学人员。

协议与实现的完美结合。着眼于 TCP/IP 基本理论,以 LwIP 源代码为基础,全面讲解 TCP/IP 协议的原理与具体实现,全部源代码对读者可见, TCP/IP 不再神秘,讲解通俗,步骤详细。每个案例的开发步骤都是以通俗易懂的语言阐述,并穿插图片和表格。

理论与实践并重。每章均配有实战例程,读者能直观感受到各层协议在嵌入式设备组网过程中的实际应用,学习 TCP/IP 不再是纸上谈兵。基于目前流行的 STM32 实验平台,可以帮助开发者快速上手嵌入式网络编程,本书的许多实验都可以抽象成一个基本的编程框架,学习者在学习的同时可以对框架进行填充裁剪等,基于框架快速完成应用程序开发。

目的明确,实用性强。本书可以作为嵌入式网络开发人员的指导手册,提供的许多应用程序可以直接用在开发项目中,缩短开发周期,节省开发成本;本书还会介绍并发服务器设计、协议栈性能测试、协议栈调试手段等知识,可以帮助开发者处理项目调试遇到的各种问题。

LwIP 协议栈全部用 C 语言写成,所有代码对用户可见,用户可以宏观地把握整个协议的代码组织结构。在本书中,通过在实验板上移植、使用协议栈,读者可以清楚地知道 TCP/IP 协议内部各个机制的实现方法以及不同参数设置对协议栈内核性能的影响,比如:如何自行配置 TCP 发送超时间隔与重发次数以满足特定场合的需求,又如用户可否根据实际使用情况更改 TCP 状态机以提高内核效率等。通过各个实验,读者可以清楚地看到整个协议的运行过程,加深对协议的理解。

组织结构

本书以开源 TCP/IP 协议栈——LwIP 的源代码为基础,详细为读者讲解 TCP/IP 协议的内部各层结构与实现细节,既包括 ARP、ICMP、IGMP、IP、UDP、TCP 等基本协议知识,也包括 DHCP、DNS 等高级应用知识,以 LwIP 为起点学习 TCP/IP,将为广大读者各方面的学习创造捷径。同时,LwIP 代码组织和实验平台搭建简单,读者可以方便地进行协议栈内部的相关实验。

如何简单快速地在当前流行的 STM32 处理器上完成协议栈移植,如何快速上手协议栈应用编程,如何移植现有的通用网络应用程序是许多读者关心的重点。LwIP 如何移植,有没有一套普遍适用的流程,各种编程接口的调用规则与实际使用例子,都是本书中将涉及到的内容。本书的许多应用程序代码可以为开发者提供一个普遍使用的程序框架,开发者只需要在这个基础上做简单改动就可以编写出适合自己的网络应用程序。对于有一定经验的 LwIP 开发者来说,可能更加关注于内核调试、协议栈稳定性和性能、并发服务器设计等方面的内容,也更期望看到 LwIP 在具体项目中的应用和实践过程,这些也都是本书将涉及的内容。

TCP/IP 技术是互联网技术发展的基础,在各软硬件 IT 企业招聘面试中,TCP/IP 相关知识成为面试官青睐的问题,对于广大学子来说,精通 TCP/IP 相关知识无疑将为自己打开一道成功择业的大门。但目前很多人想学习 TCP 却无处下手,这也成为许多初学者面临的问题。通过本书,从基于 STM32 的网络实验入手,深入贯通 TCP/IP 理论及其应用,通过学习,读者可以对协议栈、TCP/IP 细节有很好的把握;另一方面,嵌入式系统的组网技术是目前流行的

技术，嵌入式网络方向也是未来的热点发展方向，本书将从嵌入式裸机出发，向读者讲解网络协议栈移植过程以及应用编程细节，同时也包括了一些开发过程中常见的各种问题及解决方法，因此，对 LwIP 应用开发者来说，本书将是一本重要的指导手册。

编者与致谢

本书由朱升林、欧阳骏、杨晶著，粟思科审。最后，笔者想感谢一年多来一直陪伴和帮助我的人，没有你们的帮助和付出，这本书不会这么顺利地完成，因为你们的存在，使得这段时光更加珍贵。

感谢 LwIP 技术交流群（群号：193936206、224362301）中的上千位网友，是你们每天孜孜不倦的学习和讨论，让 LwIP 这款开源软件这几年在国内有了快速的成长和发展，谢谢你们的支持和肯定，大家的相互探讨也让我学到了很多東西。

感谢我的家人，是你们的宽容和细心帮我克服了生活中的许多困难和难题，你们的默默支持和付出是我继续完成此书的最大动力。

同时参与本书编写工作的还有：王治国、钟晓林、王娟、胡静、杨龙、张成林、方明、王波、陈小军、雷晓、李军华、陈晓云、方鹏、龙帆、刘亚航、凌云鹏、陈龙、曹淑明、徐伟、杨阳、张宇、刘挺、单琳、吴川、李鹏、李岩、朱榕、陈思涛和孙浩，在此一并表示感谢。

配套服务

感谢成都智造者科技有限公司（www.smartmaker.cn）为本书提供的 STM32F103VET6-EV 开发平台，该平台定位为物联网领域的首款综合性实验平台，本书基于该平台的以太网接口实现了嵌入式产品的以太网入网实战。此外，该平台预留的接口可方便地扩展该公司的其他物联网模块，如 ZigBee 模块、蓝牙模块、RFID 模块等，能快速地实现物联网领域多种实践。

本书中的所有实验代码读者可以通过中国水利水电出版社或万水书苑网站下载，网址为 <http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com> 以下方式获得。由于作者水平所限，网络技术日新月异，本书难免存在错误和不妥之处，恳请广大读者批评指正。我们为读者和用户尽心服务，围绕协议、产品和项目市场，探讨应用与发展，发掘热点与重点；开展相关教学工作。TCP/IP 协议俱乐部 QQ: 183090495，电子邮件 hwhpc@163.com，欢迎 TCP/IP 协议爱好者和用户联系我们。

共同探讨一直是学习的一种高效方式，值得指出的是，本书最后一节的 LwIP 常见问题与解决方法就得益于众多网友的反馈，这里，再次对他们表示感谢。

II

目 录

前言	
第 1 章 实验平台背后的故事	1
1.1 嵌入式网络技术一统江湖	1
1.1.1 嵌入式网络的发展现状	1
1.1.2 嵌入式 LwIP 协议栈	4
1.2 基于 STM32 的实验平台浅析	6
1.2.1 STM32 处理器的兴起	6
1.2.2 实验平台简介	9
1.2.3 网卡连接情况	11
1.3 本书主要内容	12
第 2 章 LwIP 协议栈初体验	15
2.1 庐山真面目之 LwIP 代码结构	15
2.1.1 LwIP 简介	15
2.1.2 源代码结构	17
2.2 怎样用 SI 阅读源代码	22
2.3 千里之行从 LwIP 框架做起	27
2.3.1 协议栈分层思想	27
2.3.2 进程模型	30
2.3.3 协议栈编程接口	31
第 3 章 LwIP 移植初探——无操作系统模拟层	35
3.1 移植说明	35
3.2 移植头文件	37
3.3 网卡驱动编写	40
3.4 安装系统时钟	47
3.5 协议栈初始化	47
3.6 查询接收数据包	49
3.7 数据包中断接收	51
3.8 如何定位 Ping 问题	53
第 4 章 内核动力之源——内存管理	55
4.1 内存管理背后的故事	55
4.1.1 内存管理概述	55
4.1.2 常见内存分配策略	57
4.2 以不变应万变——动态内存池	59
4.2.1 数据结构描述	59
4.2.2 函数实现	65
4.3 见招拆招——动态内存堆	67
4.3.1 数据结构描述	67
4.3.2 函数实现	69
4.4 其他内存分配策略	74
第 5 章 网络数据包——内核血液	78
5.1 数据包管理的必要性	78
5.1.1 LwIP 的分层特点	78
5.1.2 协议进程模型	80
5.2 数据包管理原理大揭秘	80
5.2.1 数据包结构 pbuf	81
5.2.2 pbuf 的类型	82
5.2.3 数据包申请函数	85

5.2.4	数据包释放函数	88	8.1.5	网络地址转换 (NAT)	150
5.2.5	其他数据包操作函数	90	8.1.6	单播、多播与广播	151
第 6 章	网络抽水机——网络接口管理	92	8.2	什么是 IP 数据报	153
6.1	引言	92	8.2.1	数据报组成结构	153
6.2	接口管理的始与末	93	8.2.2	数据结构	158
6.2.1	数据结构	93	8.3	IP 数据报之走出去	159
6.2.2	函数实现	96	8.3.1	发送数据报	159
6.3	特殊的网络接口——环回	99	8.3.2	数据报分片	162
6.3.1	实现原理	99	8.4	IP 数据报之引进来	166
6.3.2	应用程序环回通信实战	102	8.4.1	数据报接收	166
6.4	如何动态更改 IP 地址	109	8.4.2	分片重装数据结构	171
6.4.1	重要函数	109	8.4.3	分片重装函数	174
6.4.2	实战程序	110	8.4.4	分片插入与检查	178
第 7 章	漫漫鹊桥路——ARP 协议	113	第 9 章	互联网医生——ICMP 协议	182
7.1	ARP 协议之扫盲篇	113	9.1	ICMP 扫盲班	182
7.1.1	物理地址与网络地址	113	9.1.1	ICMP 协议	182
7.1.2	ARP 协议的本质	115	9.1.2	报文交付	183
7.2	关键数据结构	117	9.1.3	报文类型	184
7.2.1	ARP 表	117	9.1.4	报文格式	185
7.2.2	ARP 报文	121	9.1.5	差错报文	185
7.3	ARP 之数据包引进来	126	9.1.6	查询报文	188
7.3.1	以太网数据包递交	126	9.2	ICMP 实现大起底	189
7.3.2	ARP 数据包处理	129	9.2.1	数据结构	189
7.3.3	ARP 攻击	131	9.2.2	发送差错报文	191
7.3.4	ARP 缓存表的更新	132	9.2.3	ICMP 报文的处理	192
7.4	ARP 之数据包走出去	137	9.2.4	ping 命令	196
7.4.1	ARP 层数据处理总流程	138	9.2.5	ICMP 洪水	197
7.4.2	广播包与多播包的发送	138	9.3	如何发送 Ping 请求	198
7.4.3	单播包的发送	140	9.3.1	实现原理——raw_pcb	199
第 8 章	互联网身份证——网际协议 (IP)	144	9.3.2	发送 ping 请求	200
8.1	IP 协议之扫盲班	144	9.3.3	测试结果	203
8.1.1	概述	144	第 10 章	Raw API 无招胜有招之 UDP 篇	205
8.1.2	IP 地址	145	10.1	UDP 协议扫盲班	205
8.1.3	特殊 IP 地址	147	10.1.1	传输层协议	205
8.1.4	子网划分与子网掩码	148	10.1.2	UDP 协议	206

10.1.3	端口号	207	11.2.3	紧急数据	252
10.1.4	UDP 报文的交付	208	11.2.4	强迫数据交互	253
10.1.5	UDP 报文格式	209	11.2.5	报文首部数据结构	253
10.1.6	UDP 伪首部与校验和	210	11.3	TCP 基础——连接	254
10.2	UDP 数据结构详解	212	11.3.1	建立连接	254
10.2.1	报文首部结构	212	11.3.2	断开连接	255
10.2.2	控制块	212	11.3.3	复位连接	256
10.3	UDP RAW 编程秘籍	216	11.3.4	TCP 状态转换图	257
10.3.1	使用 UDP 编程	216	11.3.5	特殊的状态转换	259
10.3.2	新建控制块	216	11.4	TCP 核心——控制块	260
10.3.3	绑定控制块	217	11.4.1	控制块数据结构	260
10.3.4	连接控制块	218	11.4.2	控制块链表	263
10.3.5	其他控制块操作函数	219	11.4.3	接收窗口	264
10.4	UDP RAW 编程初体验	220	11.4.4	发送窗口	264
10.4.1	UDP 编程秘诀	220	11.5	TCP RAW 编程秘诀	265
10.4.2	回送程序	221	11.5.1	控制块新建	265
10.4.3	测试结果	222	11.5.2	控制块绑定	267
10.4.4	回调编程的缺陷	224	11.5.3	控制块侦听	267
10.5	内核报文处理细节	224	11.5.4	控制块连接	268
10.5.1	报文的发送	224	11.5.5	发送数据	270
10.5.2	报文接收与递交	228	11.5.6	关闭连接	270
10.6	挑战更复杂的 UDP 应用	231	11.5.7	其他函数	272
10.6.1	使用 TFTP 传送文件	231	11.6	TCP RAW 编程初体验	272
10.6.2	应用程序实现	234	11.6.1	TCP 编程本质	272
10.6.3	TFTP 测试结果	242	11.6.2	HTTP 服务器	278
第 11 章	Raw API 无招胜有招之 TCP 篇	244	11.6.3	打不死的客户端	280
11.1	TCP 协议大揭秘	244	11.7	TCP 输出处理细节放送	283
11.1.1	TCP 的必要性	244	11.7.1	报文段缓冲	283
11.1.2	TCP 的特性	245	11.7.2	报文段构建	285
11.1.3	连接的定义	247	11.7.3	报文段发送	287
11.1.4	数据流编号	248	11.8	TCP 输入处理细节放送	289
11.1.5	滑动窗口	248	11.8.1	TCP 层函数调用流程	289
11.2	TCP 入门——报文	249	11.8.2	报文输入	291
11.2.1	报文格式	249	11.8.3	TIME_WAIT 状态的报文处理	294
11.2.2	TCP 选项	251	11.8.4	LISTEN 状态的报文处理	294

11.8.5	SYN 攻击	296	13.1.3	内核进程	359
11.8.6	TCP 状态机函数	297	13.1.4	处理定时事件	361
11.8.7	TCP 数据输入	301	13.2	论消息机制的重要性	363
11.9	可靠的传输服务	305	13.2.1	消息结构	363
11.9.1	超时重传与 RTT 估计	306	13.2.2	数据包消息	364
11.9.2	慢启动与拥塞避免	311	13.2.3	协议栈 API 实现	365
11.9.3	快速重传与快速恢复	314	13.2.4	API 消息	367
11.9.4	糊涂窗口与避免	316	13.3	协议栈接口全搜罗	369
11.9.5	零窗口探查	319	13.3.1	用户数据缓冲 netbuf	369
11.9.6	保活机制	321	13.3.2	如何操作数据缓冲	371
11.9.7	TCP 定时器	322	13.3.3	连接结构 netconn	373
11.10	挑战更复杂的 TCP 应用	327	13.3.4	内核回调接口	375
11.10.1	Telnet 服务器	327	13.3.5	协议栈 API 秘籍	377
11.10.2	测试结果	333	13.4	Sequential API 大会战	383
第 12 章	LwIP 乾坤大挪移——操作系统		13.4.1	网页控制设备	383
	模拟层	335	13.4.2	UDP 性能测试	386
12.1	模拟层是怎样炼成的	336	13.4.3	TCP 性能测试	388
12.1.1	移植综述	336	13.4.4	并发服务器	390
12.1.2	添加文件	338	第 14 章	万剑归宗之 Socket 编程	395
12.1.3	修改头文件 lwipopts.h	339	14.1	套接字接口来龙去脉	395
12.1.4	头文件 sys_arch.h	339	14.1.1	如何实现 Socket	395
12.1.5	全局变量与初始化	340	14.1.2	套接字函数秘籍	397
12.1.6	信号量函数	341	14.2	Socket 实战大串烧	401
12.1.7	邮箱函数	343	14.2.1	智能的 TCP 客户端	402
12.1.8	创建新进程	348	14.2.2	UDP 时间服务器	404
12.1.9	修改内核初始化函数	349	14.2.3	Winsock 客户端设计	406
12.1.10	移植结果测试	350	14.3	基于 Select 的并发服务器	408
12.2	Sequential API 牛刀小试	350	14.3.1	神奇的 Select 函数	409
12.2.1	回显服务器	351	14.3.2	Select 原理全解析	410
12.2.2	Web 服务器	352	14.3.3	Select 实战——并发服务器	419
12.2.3	操作系统下的 raw API	355	第 15 章	LwIP 工程实战——物联网关	424
第 13 章	Sequential API 大显身手	356	15.1	项目背景	424
13.1	探秘内核定时事件	356	15.1.1	中央空调系统与温控器	424
13.1.1	定时结构	356	15.1.2	新需求的出现	427
13.1.2	定时链表	357	15.1.3	要做什么与如何去做	430

15.2 智能网络温控器.....	431	16.2.3 是骡子是马——遛遛 DHCP.....	460
15.2.1 温控器实现方案.....	431	16.3 如何实现组播——Internet 组管理.....	464
15.2.2 Modbus TCP 服务器.....	432	16.3.1 IGMP 协议大讲堂.....	464
15.3 物联网关设计.....	440	16.3.2 IGMP 组播实践.....	465
15.3.1 物联网关实现方案.....	440	第 17 章 最后的战役	468
15.3.2 网关服务器.....	441	17.1 如何配置协议栈.....	468
15.4 现场调试与测试.....	446	17.1.1 内核配置文件 opt.h.....	468
15.4.1 温控器测试.....	446	17.1.2 如何提高 TCP 性能.....	471
15.4.2 物联网关测试.....	447	17.2 内核调试常用方法.....	472
第 16 章 天下之术皆为我用——LwIP 高级篇	449	17.2.1 内核日志输出.....	472
16.1 外网访问之道——域名解析.....	449	17.2.2 网络调试助手.....	474
16.1.1 域名系统 (DNS).....	449	17.2.3 Wireshark 抓包.....	475
16.1.2 DNS 牛刀小试.....	450	17.3 为了更好的明天.....	476
16.1.3 下载外网数据.....	453	17.3.1 常见移植问题.....	476
16.2 如何自动分配 IP 地址——DHCP.....	456	17.3.2 常见应用问题.....	478
16.2.1 DHCP 大讲堂.....	456	17.4 故事结局.....	483
16.2.2 DHCP 之实现详解.....	458	参考文献	484

1

实验平台背后的故事

本书的特点定位于从实践动手，从实践中学习基于 STM32 的嵌入式网络以及 LwIP 的基本知识，因此网络相关的实验将在本书中占据相当大的篇幅。作为开篇，有必要将本书的写作背景以及读者最关心的操作平台环境交代清楚，为后续协议移植及实战打好基础。由于本书实验是基于 STM32 处理器开发平台的，它将伴随读者走过整本书的学习过程，因此本章也将会对 STM32 处理器以及实验平台做简要的说明。本章主要包含以下内容：

- 嵌入式网络介绍，说明其发展现状以及未来，为本书奠定理论基础；
- 目前在小型嵌入式设备领域很流行的 LwIP 协议介绍、研究现状及其发展；
- 实验平台 STM32 处理器简要概述；
- 基于 STM32F103 的实验平台介绍。

平台介绍是本章的重点，希望读者通过本章的学习能对 STM32 处理器以及本书实验平台有个感性的认识。

1.1 嵌入式网络技术一统江湖

1.1.1 嵌入式网络的发展现状

说到嵌入式网络这个词，相信读者并不陌生，在物联网大肆发展的今天，网络俨然已经成为一个炙手可热的名词。3G、4G LTE、WIFI、ZigBee、蓝牙、以太网等各种网络技术正将越来越多的服务器、手机终端、用户 PC、工业设备等连入因特网，给人们构建了一个丰富多彩的网络新世界，大大改变了人们生产、消费、学习和娱乐的方式。本书讨论的技术范畴既包括了以太网这种流行的组网技术，也包括了所有因特网技术的基础——TCP/IP 技术，它们都

是发展最早，更是目前最流行的技术。

嵌入式产品连入因特网，当下正在各个应用领域如火如荼地展开。在设备上实现网络功能，用户可以足不出户地对产品进行配置、控制设备的行为、获取自己关心的设备数据信息，可以大大提高工作效率。目前，嵌入式网络产品的经典应用已经不胜枚举了，特别是在智能家居、智能门禁、智能电网这些国家大力倡导的领域，近年来已不断涌现出各种基于嵌入式网络的创新应用。笔者在《嵌入式网络那些事：LwIP 协议深度剖析与实战演练》一书中曾以网络打印机为例，阐释了网络给办公场所打印设备带来的方便。这里，笔者迫不及待地要介绍另一个嵌入式网络设备的应用场景，相信它的发展将会为所有读者未来生活增添不少奇妙的体验，通过这个例子，笔者更期望它能激发读者对网络学习的兴趣，为网络的发展贡献一份力量。

云计算如今应该是个耳熟能详的词语了，但是云计算具体是什么样子？它能为我们未来的生活提供什么样的服务和改变呢？如图 1-1 所示，它显示了未来云计算将给个人社区带来的改变。未来云主机供应商将会在社区提供社区云服务，社区用户再也不用花费大量的时间和钞票去电脑城购买 PC 了，而是可以直接到云主机供应商那里去租一台虚拟主机。供应商会给你提供一个叫做瘦客户机的盒子，可别小看这个盒子，麻雀虽小，五脏俱全，它基本集成了普通 PC 主机上的所有接口，可以插显示器、键盘、鼠标、网线和其他设备（比如 USB 设备等）。当用户把鼠标、键盘、显示器、网线插到这个盒子上后，便得到了一台可以高效运行的完整主机。

这个主机是如何工作的呢？首先用户去云服务器提供商那里租赁，会得到一个唯一的账号和密码，通过这个账号和密码，用户可以连接到社区云中属于自己的虚拟主机上，虚拟主机在用户租赁时是可以定制的，虚拟机的处理器性能、内存大小、硬盘大小等可以各不相同，随用户的喜好与需求定制，当然不同配置的虚拟主机的租赁价格肯定也是有区别的，配置越好租金越贵。在社区云中运行的是数十台甚至上百台大型服务器，它们之间通过存储虚拟化、计算虚拟化、网络虚拟化等技术组织成为一个大的服务器集群，为社区提供桌面云服务。所有虚拟主机在集群中独立运行，互不影响，却可以共用集群的计算、存储等资源，使得整套集群系统的性能和功能达到最佳，提高资源的优化与利用率。

除了上述的各种虚拟化技术外，社区云中的瘦客户机也起着非常关键的作用，虽然大多数情况下它只发挥了一个简单的数据传递的所用。如图 1-1 所示，瘦客户机把终端用户的操作事件（比如键盘按键、鼠标移动、鼠标点击等信息）通过算法加密、数据压缩等处理后传入到社区云中的用户虚拟主机上，虚拟主机完成对事件的处理并将显示内容返回，瘦客户机最终将数据递交给显示器显示。

这样的云虚拟主机与传统的 PC 机相比有什么优势？从下面几个方面来看：

首先，价格上以租赁的方式进行，将以较低的门槛为用户提供服务，租房子总比买房子容易吧；用户可以通过不同的需求定制不同的虚拟主机配置，租赁价格随着配置的变化而变化。

其次，由于这种虚拟主机可以定制，当然也可以对用户注销的主机进行回收，同样，所有虚拟机使用不可能同时在线，所以正在运行的虚拟机可以使用其他离线虚拟机的计算和网络资源，这样云服务提供商可以达到资源整合与最优化配置的目的。

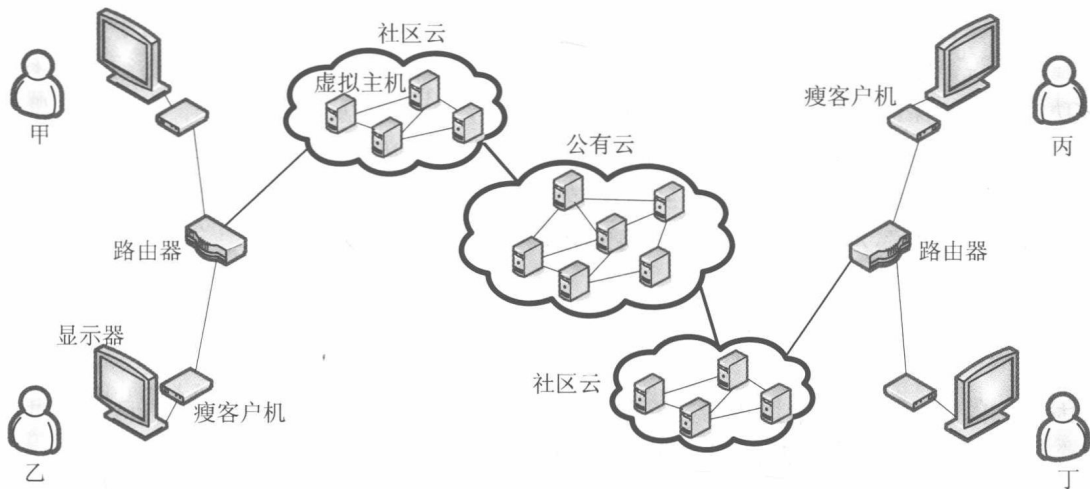


图 1-1 未来社区桌面云服务

再次，这样的一个虚拟主机不会中病毒，不会受黑客攻击，用户不必担心有一天操作系统挂了怎么办，不会出现蓝屏这种 PC 特有的让用户手足无措的情况。所有这一切恐怖的事情都已由供应商在身后帮你搞定了。

然后，云中的数据永久保存，用户再也不必担心自己 PC 的硬盘坏了怎么办，通常云中的数据都有多个备份，有些安全级别高云服务还可能布置跨地域的数据备份，可以避免地震、火灾等大型灾难带来的数据毁坏。

还有，这个盒子功耗低、辐射小、噪声小，即节能又环保，确实是居家生活、出门远行之必需。

另外，文件共享的速度非常快，所有数据的传输是在云端之间的各个虚拟主机之间直接进行，它们之间可能是使用光纤等更快的通道进行数据的交互。

最后，这种虚拟主机使用很方便，用户体验极佳，用两个假设来看看。假设 1，图 1-1 中的甲家某天电源跳闸了，他又急需使用电脑，怎么办？很简单，他只要到邻居乙家，通过乙家的瘦客户机，登录自己的账号和密码，立刻就可以连接到自己的虚拟主机上。假设 2，丙正在编写一份重要的文档，突然家中跳闸了怎么办？不用担心，他的虚拟主机还稳稳地运行于云端，不会因为瘦客户机的断电而停止，只要重新通过瘦客户机登录自己的虚拟主机，所有东西还像断电前那样正常运行着。这种用户体验很适合于企业、学校、医院、政府等大型机构，特别是在会议室的时候，你再也不用 U 盘拷贝，也不用 MSTSC 到你的 PC 机上去找你需要的会议资料了，直接登录自己的云账号就能解决所有问题。

事实上值得指出的是，上面的描述并不是一种虚构场景，目前很多企业、学校和医疗机构都引进了这种云服务，即所谓的“桌面云服务”，这种服务在方便用户的同时更降低了企业的运营硬件开销，相信未来这种技术进入社区为个人提供服务并不是很难的事情。

通过上面的啰嗦，想必读者也和笔者一样，对未来的社区云充满了期待，但是这里笔者

想从美好的憧憬里面先跳出来，看看所有问题的关键以及我们正面临的问题。对于前面所述的瘦客户机来说，最重要的是什么？网络！要实现这样的预想，我们的瘦客户机必须具备强大的网络功能，如何保证网络的高效性、可靠性、稳定性，实现数据的有效传输，是目前云供应商们正在面临和亟待解决的问题。如何在低可靠性的网络环境下提供可靠无误的社区云传输服务，还有如何在低带宽的网络环境下实现高效的数据传输，以保证大型游戏、高清视频等用户应用的需求？这也都为现在正在学习网络的同学提出了机遇与挑战。

1.1.2 嵌入式 LwIP 协议栈

因特网的迅猛发展，给世界带来了翻天覆地的变化，使人类真正进入了信息产业时代。时至今日，嵌入式领域也不断涌现出连入因特网的巨大需求，特别是物联网产业、移动终端等发展方向。嵌入式设备要连入因特网，就必须遵循网络通信协议，即 TCP/IP 协议。目前，嵌入式产品主要通过两种方式来实现 TCP/IP 功能，一是使用专门的硬件网络 TCP/IP 协议栈芯片，如 WIZnet 的 W5200 等，主控制器只需要通过一定的接口去对这些网络芯片编程，便可以得到芯片中的 TCP/IP 数据，这些芯片的价格一般在几十元或者上百元不等，支持的 TCP、UDP 连接数也有限制，不适合广泛地应用于嵌入式领域；另一种使用更为广泛的方式是在主控制器上移植嵌入式网络 TCP/IP 协议栈，使用免费的协议栈可以大大降低设备开发成本。下面例举了几个在嵌入式领域经常使用的 TCP/IP 协议栈，同时对各个协议栈特点及使用情况做简要的介绍。

1. uIP

uIP 是由瑞典计算机科学学院 (Swedish Institute of Computer Science) 的 Adam Dunkels 等人专门为 8 位和 16 位控制器设计的一个非常小的 TCP/IP 协议栈。它去掉了完整 TCP/IP 协议中不常用的功能，简化了通信流程，其代码完全用 C 编写，可移植到各种不同的结构和操作系统上，一个编译过的协议栈可以在几 kB ROM 和几百字节 RAM 中运行。uIP 代码容量小巧，实现功能精简，已经在嵌入式领域得到了广泛的应用，且有很多基于 uIP 的产品出现。但另一方面，uIP 不完备的 TCP/IP 实现限制了其在一些较高要求场合下的应用，如对可靠性要求高或大数据量传输的场合。

2. uC/IP

uC/IP 是由 Guy Lancaster 编写的一套基于 uC/OS 操作系统的开放源码的 TCP/IP 协议栈，它是一套完全免费的、可供研究的 TCP/IP 协议栈。uC/IP 大部分源码是从公开源码的 BSD-Lite 移植过来的，最初被设计在串行链路上使用以实现 PPP 协议的支持，作者在将其成功地应用在了某个嵌入式产品上后，便将其相应代码共享出来，希望有利于其他嵌入式产品的开发。后来该协议栈不断完善，并实现了对以太网的支持，协议栈需要的代码容量空间在 30~60kB 之间。

uC/IP 还有许多不足的地方，首先其对网络应用的支持不足，不能像 LwIP 那样提供多种