

手把手教你开发路由器

视频讲解

系统源码

硬件开源

辅导答疑



物联网 (IoT) 网关开发实战

疯壳团队 陈万里 刘 燃 编著



西安电子科技大学出版社
<http://www.xduph.com>

手把手教你开发路由器

物联网(IoT)网关开发实战

疯壳团队 陈万里 刘 燃 编著

西安电子科技大学出版社

内 容 简 介

本书以 MT7688 无线路由器为例,按产品开发流程,详细讲解了 OpenWrt 系统开发知识,通过一套完整的物联网网关实现方案,逐步做出一台可量产的无线路由器产品。

本书共 3 章,主要内容包括:OpenWrt 开发前的准备、OpenWrt 开发基础、OpenWrt 开发实战。本书语言通俗易懂,即使从来没接触过 OpenWrt 和路由器开发的读者也能顺利上手。

本书对于想要从事物联网网关开发工作的在校学生、程序开发爱好者或转行从业者而言,是一本很好的入门教材;对于正在从事路由器开发的工程师,也有一定的参考和指导作用。

图书在版编目(CIP)数据

物联网(IoT)网关开发实战/疯壳团队,陈万里,刘燃编著. —西安:西安电子科技大学出版社,2019.4

ISBN 978-7-5606-5268-9

I. ① 物… II. ① 疯… ② 陈… ③ 刘… III. ① 互联网络—应用 ② 智能技术—应用 IV. ① TP393.4 ② TP18

中国版本图书馆 CIP 数据核字(2019)第 037533 号

策划编辑 高 樱

责任编辑 滕卫红 阎 彬

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfbx001@163.com

经 销 新华书店

印刷单位 咸阳华盛印务有限责任公司

版 次 2019 年 4 月第 1 版 2019 年 4 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 17

字 数 402 千字

印 数 1~3000 册

定 价 45.00 元

ISBN 978-7-5606-5268-9/TP

XDUP 5570001-1

*** 如有印装问题可调换 ***

前言

本书的硬件平台选定的是 MT7688,它是目前 IoT(物联网)开发的首选平台。MT7688 系统单芯片可应用于家庭自动化的桥接中心,它集成了 1T1R 802.11n WiFi radio、580 MHz MIPS[®] 24KEc[™] CPU、1-port fast Ethernet PHY、USB 2.0 host、PCIe、SD-XC、I2S/PCM,并支持多种低速输出、输入接口。MT7688 支持两种运作模式:IoT gateway 模式与 IoT device 模式。在 IoT gateway 模式下,MT7688 可通过 PCIe 接口连接至 802.11ac 芯片组,并作为双频 802.11ac 同步闸道。高速的 USB 2.0 接口可让 MT7688 连接至额外的 3G/LTE Modem 硬件,或连接到 H.264 ISP,用作无线 IP 相机。IoT gateway 模式也支持触摸板、Bluetooth Low Energy(BLE)、Zigbee/Z-Wave 和 Sub-1 GHz RF 等智能家居应用所需的硬件。

本书的软件开发平台是目前比较流行的 OpenWrt。OpenWrt 是嵌入式设备上运行的 Linux 系统。OpenWrt 的文件系统是写可的,开发者无需在每一次修改后重新编译整个系统,而且可以自由安装同一款平台编译的 *.ipk 软件,令它更像一个小型的 Linux 电脑系统,这样也加快了开发速度。

如果对 Linux 系统有一定的认识,并想学习或接触嵌入式 Linux 系统,那么 OpenWrt 是很适合的。OpenWrt 支持各种处理器架构,对 ARM、X86、PowerPC 或 MIPS 都有很好的支持作用。OpenWrt 拥有多达 3000 多种的软件包,如工具链(toolchain)、BootLoader(Uboot)、内核(Linux kernel)及根文件系统(rootfs)等。用户只需一个简单的 make 命令,即可方便快速地定制一个具有特定功能的嵌入式系统固件(Firmware)。

一般嵌入式 Linux 的开发,无论是 ARM、PowerPC 或 MIPS 的处理器,都必须经过以下 6 个步骤。

- (1) 创建 Linux 交叉编译工具链(toolchain)。
- (2) 移植 BootLoader(主要是 Uboot)。
- (3) 移植 Linux kernel。
- (4) 创建 rootfs(根文件系统)。
- (5) 编写设备驱动程序。
- (6) 编写应用软件。

OpenWrt 可以快速构建一个包括上述 6 个步骤的完整的 SDK 开发环境。随着 Linux 技术的成熟,大量不同的处理器内核和应用软件相继出现。熟悉这些嵌入式 Linux 的基本开发流程后,不应再局限于 MIPS 处理器和无线路由器的开发,可以尝试在其他处理器或者非无线路由器的系统移植嵌入式 Linux,定制适合自己的应用软件,并完成一个完整的嵌入式产品。

1. 本书的内容

本书的内容几乎涵盖了 shell、Makefile、裸机编程、Linux 驱动开发中的所有知识点,虽然有些知识点讲得并不是很深入,但作者会抛砖引玉,告诉读者如何获取相关资料。书

中的各章节内容都是根据实际项目开发步骤，按照从易到难的顺序编排的，建议读者按顺序学习。第1、2章是与 OpenWrt 平台相关的基础知识，读者首先需掌握开发环境的配置，然后掌握系统的编译方法。只有配置好开发环境，学会使用相应的指令编译代码并使之编译通过，才能进行后面章节的学习。在讲解完所有的知识点后，本书配套了一个商用的物联网网关平台，作为读者实战开发的调试设备，目的是以项目实战来提高读者的学习兴趣，让读者学会如何运用前面所学的知识点开发产品。

2. 本书的特点

(1) 实用性强。以真实的商用产品方案 MT7688 为例，全面讲解了 Linux 驱动开发的流程和技能。虽然是以 MT7688 为例进行讲解，但是其中相应的知识可以衍生到任何使用 Linux 的设备中。

(2) 专业权威。本书的作者是物联网网关的一线开发者，拥有多年网关项目开发经验，负责了多款网关产品的开发及量产维护工作，书中很多内容是作者对真实项目的开发总结。

(3) 内容全面。本书基本涵盖了网关开发的所有知识点。

(4) 实验可靠。书中所有源码都经过真实环境验证，有极高的含金量。

(5) 售后答疑。所有读者都可在 https://www.fengke.club/GeekMart/su_fRTZ3qKY0.jsp 官网社区提问，作者会不定期答疑。

3. 本书的适用范围

(1) 想了解 IoT 网关设备的开发方法的开发者。

(2) 想从事 IoT 网关设备驱动研发工作的在校学生、程序开发爱好者或转行从业者。

(3) 已经入行或正在从事 IoT 网关设备驱动开发的工程师。

(4) 进行 IoT 网关设备驱动开发培训的机构和单位。

(5) 高校教师或学生。本书可用于高校实验课程教材。

本书由刘燃统稿，由陈万里、刘燃编写。在此要特别感谢深圳疯壳团队的各位小伙伴为本书的编写提供的可靠技术支持与精神鼓励。此外，还要感谢西安电子科技大学出版社给予了大力支持。

由于时间仓促，虽然本书的所有内容都经过作者认真校核，但难免会有一些纰漏，读者可通过社区论坛与作者互动。

本书的源码可到 https://www.fengke.club/GeekMart/su_fRTZ3qKY0.jsp 社区论坛免费下载。

作者

2019年1月

目 录

第 1 章 OpenWrt 开发前的准备	1
1.1 开发环境搭建	1
1.1.1 从零开始搭建 Ubuntu 开发环境	1
1.1.2 安装开发环境	7
1.1.3 安装编译工具和源代码	13
1.2 使用 git 管理源代码	19
1.2.1 用 git 命令初始化代码	20
1.2.2 用 gitk 工具来图形化管理(可以用 MobaXterm 启动)	20
第 2 章 OpenWrt 开发基础	24
2.1 shell 脚本编程基础	24
2.1.1 初识 OpenWrt shell	24
2.1.2 基本的 ash shell 命令	28
2.1.3 更多的 ash shell 命令	48
2.1.4 进一步理解 ash shell	62
2.1.5 理解 Linux 文件权限	73
2.1.6 使用 Linux 环境变量	81
2.1.7 使用编辑器 vi(vim)	89
2.1.8 脚本编程基础	94
2.2 Makefile 简介	119
2.2.1 GNU make 介绍	120
2.2.2 Makefile 基础	127
2.2.3 Makefile 的书写规则	134
2.2.4 Makefile 的书写命令	142
2.2.5 Makefile 的变量使用	145
2.2.6 make 命令的执行	159
2.2.7 Makefile 中的控制语句(条件判断)	167
2.2.8 Makefile 中的隐含规则	170

第 3 章 OpenWrt 开发实战	175
3.1 疯壳 Demo 开发板简单介绍	175
3.2 Uboot 简介——基于 MTK 官方提供的 Uboot	177
3.2.1 Uboot 配置过程	178
3.2.2 Uboot 编译	182
3.2.3 ROM 版本 Uboot——Uboot 烧写版本	183
3.2.4 RAM 版本 Uboot——Uboot 调试版本	185
3.2.5 Uboot 启动命令选项	189
3.2.6 增加缺省环境变量值	199
3.2.7 Uboot 的系统初始化	201
3.2.8 Uboot 第二阶段——启动内核过程	211
3.2.9 Uboot 基本命令	213
3.2.10 Uboot 常用命令详解	218
3.3 Kernel 简介——基于 MTK 官方提供的 Kernel	219
3.3.1 Kernel 配置过程	219
3.3.2 Kernel 编译——编译一个带有根文件系统的 image	226
3.3.3 移植一个新的内核模块到 SDK 中	227
3.3.4 系统启动时执行的命令——更新 rcS 文件	229
3.3.5 根文件系统 rootFs 中增加新的文件	229
3.3.6 裁剪 Image 的尺寸	230
3.3.7 Kernel 烧写和调试	233
3.4 应用程序库的编译(user Library)	238
3.4.1 50.Library 配置过程	238
3.4.2 Library 的移植	239
3.4.3 编译 user Library	241
3.5 应用程序简介	241
3.5.1 MTK 所有权的应用程序	241
3.5.2 桥的配置软件 bridge - utils	247
3.5.3 BusyBox 应用程序	248
3.5.4 ctorrent - dnh 3.2 应用程序	249
3.5.5 curl 应用程序	249

- 3.5.6 dnsmasq - 2.40 应用程序 249
 - 3.5.7 dropbear - 0.52 应用程序 250
 - 3.5.8 ebttables 应用程序 250
 - 3.5.9 iptables 应用程序 250
 - 3.5.10 ntfs - 3g 应用程序 250
 - 3.5.11 samba 应用程序 250
 - 3.5.12 strace 应用程序 251
 - 3.5.13 tcpdump 应用程序 251
 - 3.5.14 wireless_tools 应用程序 251
 - 3.5.15 wpa_supplicant - 0.5.7 应用程序 251
- 3.6 移植新的应用程序 252
- 3.7 OpenWrt 移植、编译、使用 254
 - 3.7.1 OpenWrt 介绍 254
 - 3.7.2 关于 SDK 254
 - 3.7.3 编译 SDK 255
 - 3.7.4 Web 访问 258
- 参考文献..... 264

第1章 OpenWrt 开发前的准备

1.1 开发环境搭建

本章主要以 Ubuntu 32 bits 操作系统为基础,讲解如何在操作系统上搭建开发环境、安装常用的工具软件,并逐步帮助用户完成源代码的管理和编译工作。主要内容包括如何搭建 Ubuntu 开发环境。

1.1.1 从零开始搭建 Ubuntu 开发环境

疯壳团队为大家准备了一个安装好的不带有任何编译工具包,并基于免费的 VirtualBox (VirtualBox - 5.1.18 - 114002 - Win. exe)的 Ubuntu 12.04 原始 32 位虚拟机 ubuntu - 12.04.5 - i386 - raw. vdi,初学者可以在此基础上按照本书的指导一步一步完成编译环境的搭建工作。如果希望尝试安装 Ubuntu 操作系统,疯壳团队准备了可从 Ubuntu 官网下载的 ubuntu - 12.04.5 - desktop - i386. iso 文件,用户可以用此文件镜像安装一个全新的操作系统。

需要注意的是:本书所有的内容都基于 32 位 Ubuntu 12.04 版本演示,希望读者用此版本完成本书学习。如果想尝试在其他 Ubuntu 版本上完成实验,则遇到的问题可能与本书描述的不一样,同时也得不到相应的解决方案,这样可能会降低学习的积极性。

下面讲解如何设置虚拟机。复制 ubuntu - 12.04.5 - i386 - raw. vdi 到当前目录并改名为 ubuntu - fengke - mt7688. vdi,如图 1-1 所示。复制一个新的 vdi 文件的目的是:在接下来对虚拟机的操作中如果出现了错误可以推翻重来。

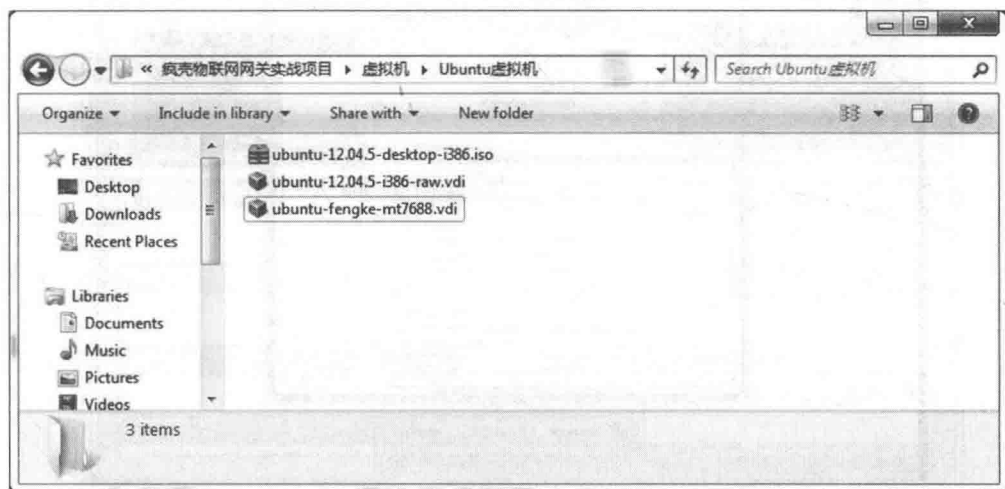


图 1-1 虚拟机文件

1. 创建虚拟机

(1) 点击 New 新建一个虚拟机，填入名字 ubuntu - fengke - mt7688，然后点击 Next，如图 1-2 所示。

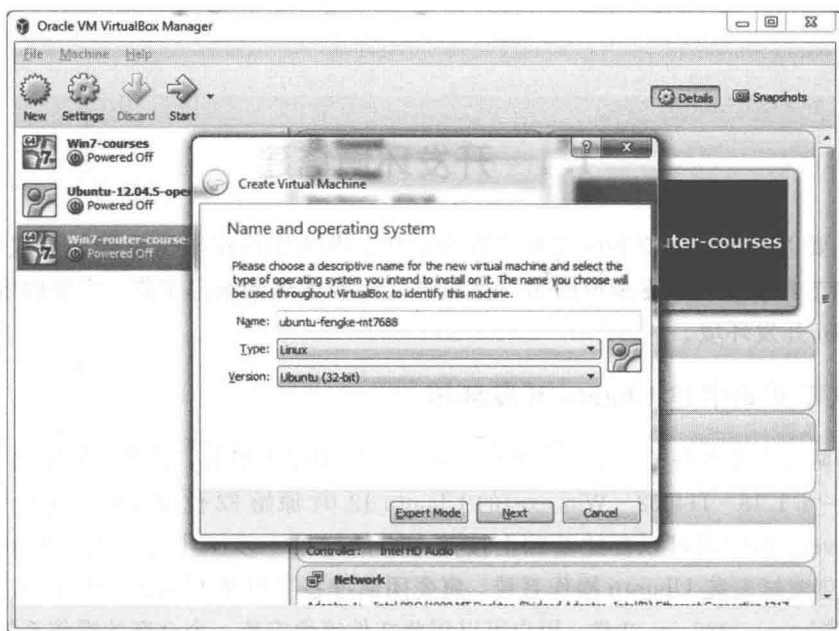


图 1-2 点击 New 新建一个虚拟机

(2) 设置内存大小为 1024 MB，点击 Next，如图 1-3 所示。

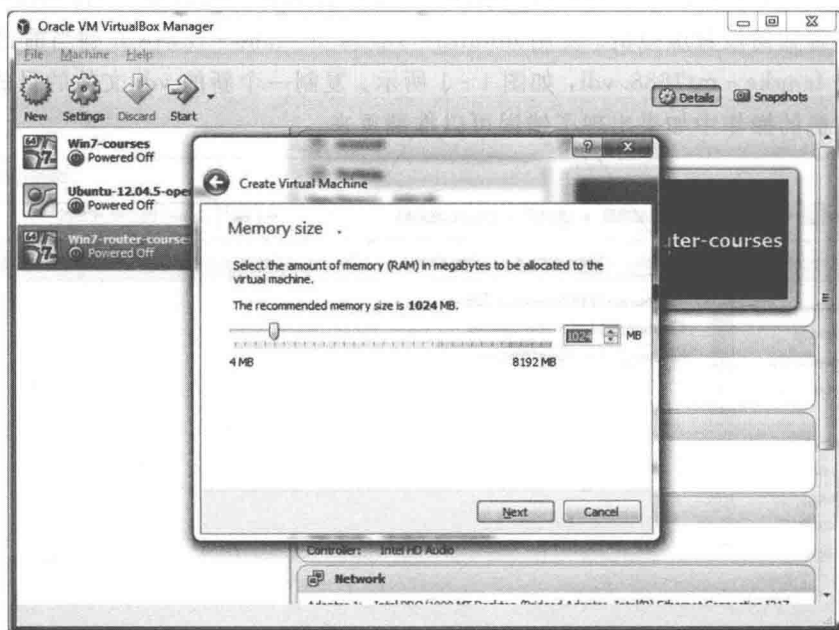


图 1-3 设置内存大小为 1024 MB

(3) 选择已经存在的虚拟硬盘 ubuntu - fengke - mt7688.vdi, 并点击 Create, 如图 1 - 4 所示。

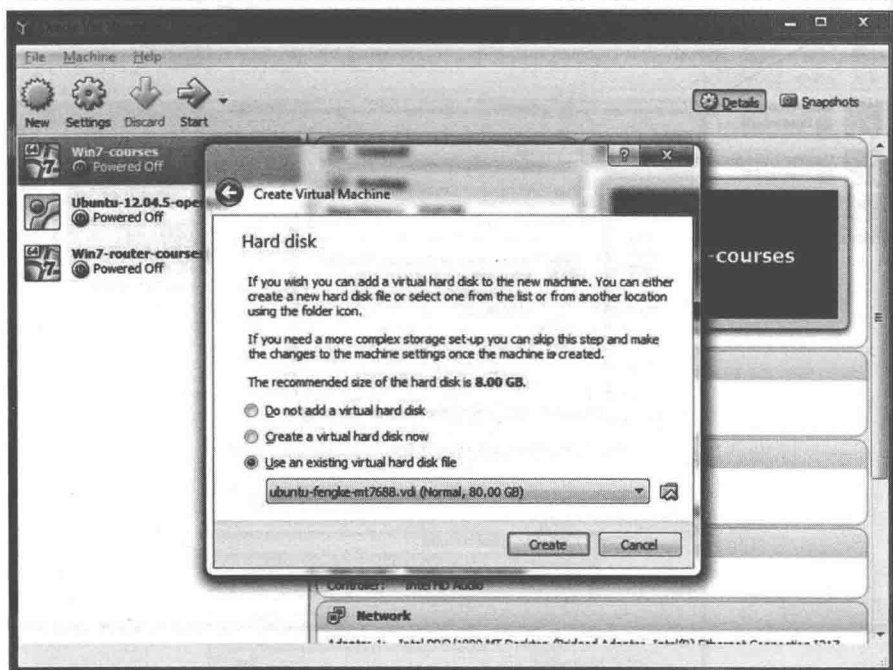
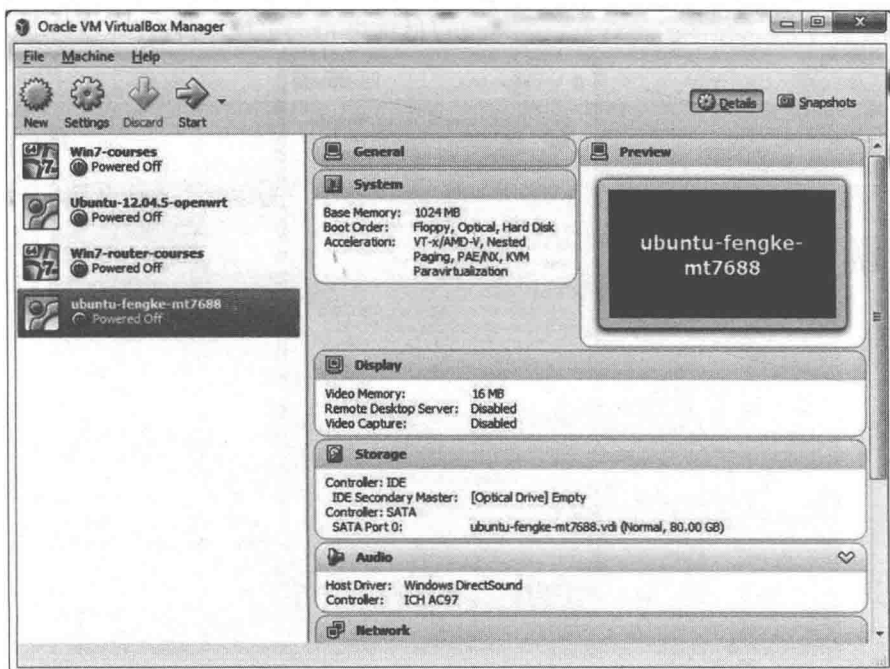


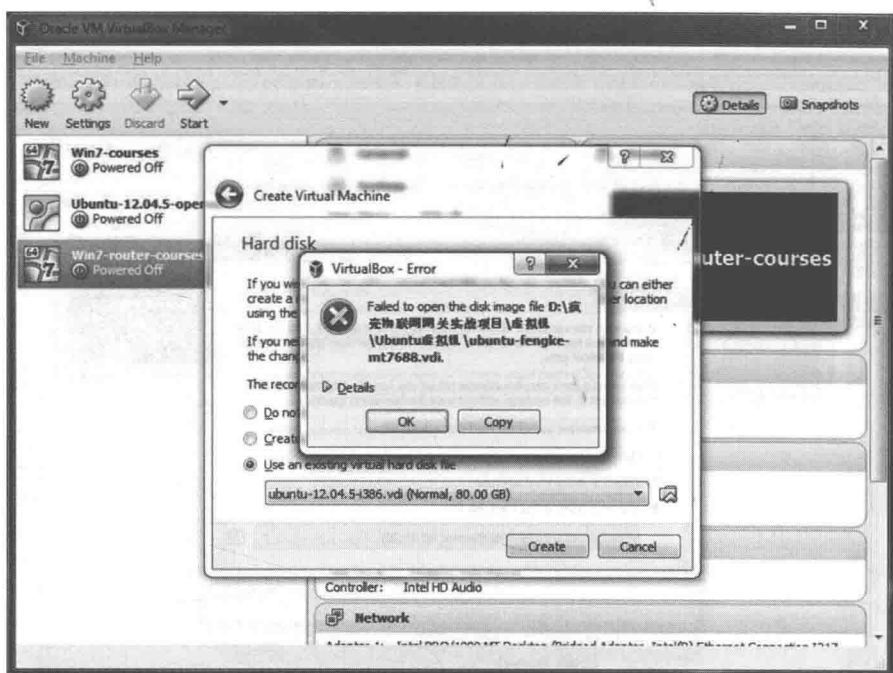
图 1 - 4 选择已经存在的虚拟硬盘

(4) 生成一个新的虚拟机 ubuntu - fengke - mt7688, 如图 1 - 5(a) 所示。



(a) 生成一个新的虚拟机

点击 Create, 有时会提示出错, 如图 1-5(b) 所示。



(b) 点击 Create, 有时会提示出错

图 1-5 生成新的虚拟机及可能出现的出错信息

(5) 点击 Details, 查看错误的详细信息, 如图 1-6 所示。

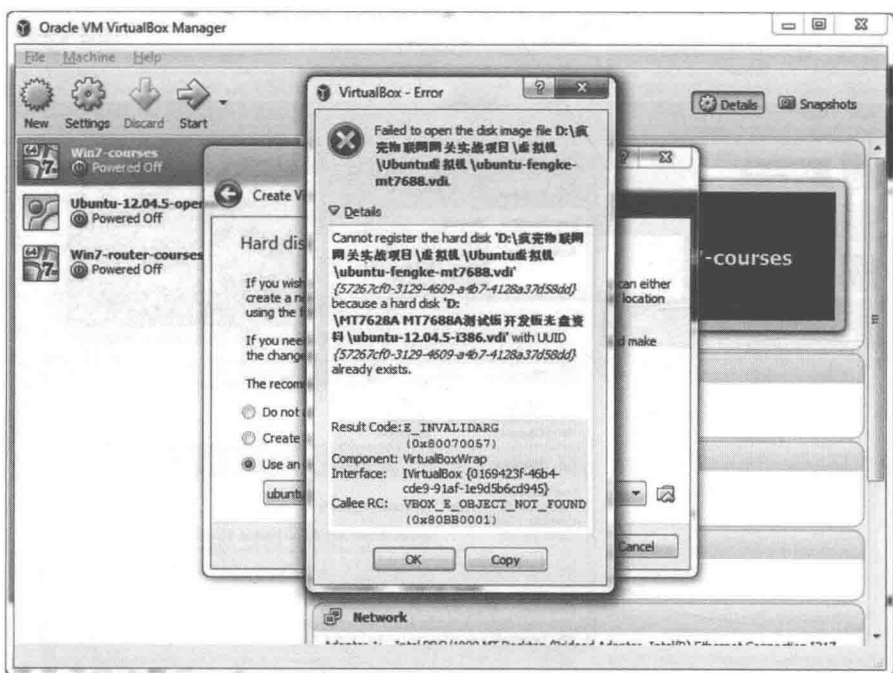


图 1-6 查看错误的详细信息——UUID already exists

上述错误出现的原因是 VirtualBox 系统通过复制镜像的方式安装了多个内容一样的 Ubuntu12.04 虚拟机(第一次安装时不会出现以上错误),这样系统就会提示 UUID already exists。遇到此错误情况时,解决方法如下:

① 打开命令行 cmd,点击左下角 Windows 图标后输入 cmd,如图 1-7 所示。



图 1-7 cmd 的打开方式

② 在命令行中输入命令: `VBoxManage.exe internalcommands sethduid D:\疯壳物联网网关实战项目\虚拟机\Ubuntu 虚拟机\ubuntu-fengke-mt7688.vdi`,设置一个新的 UUID,如图 1-8 所示。

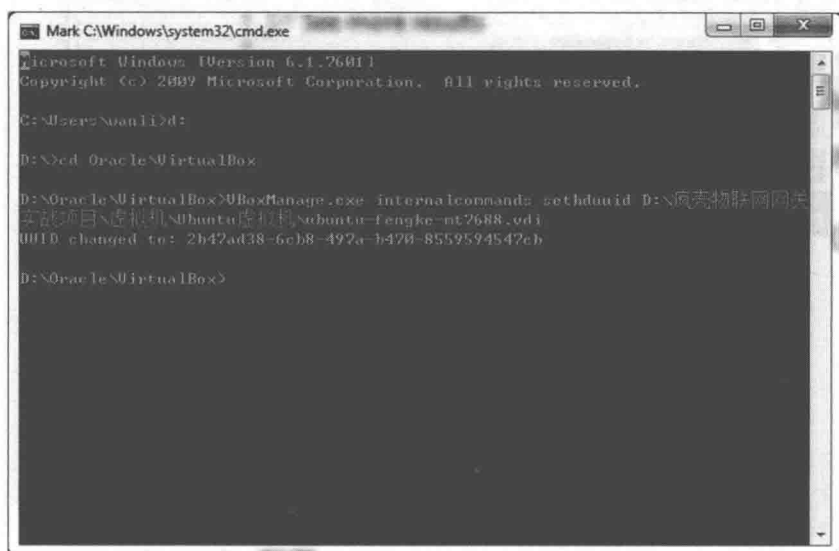


图 1-8 设置一个新的 UUID

③ 用设置了新 UUID 的 ubuntu-fengke-mt7688.vdi 重新创建虚拟机。

2. 设置网络

为了能够和主机共享网络,这里选择设置成桥(bridge)模式。点击网络(Network),如图 1-9 所示,在弹出的对话框中选择桥接模式(Bridged Adapter),如图 1-10 所示。

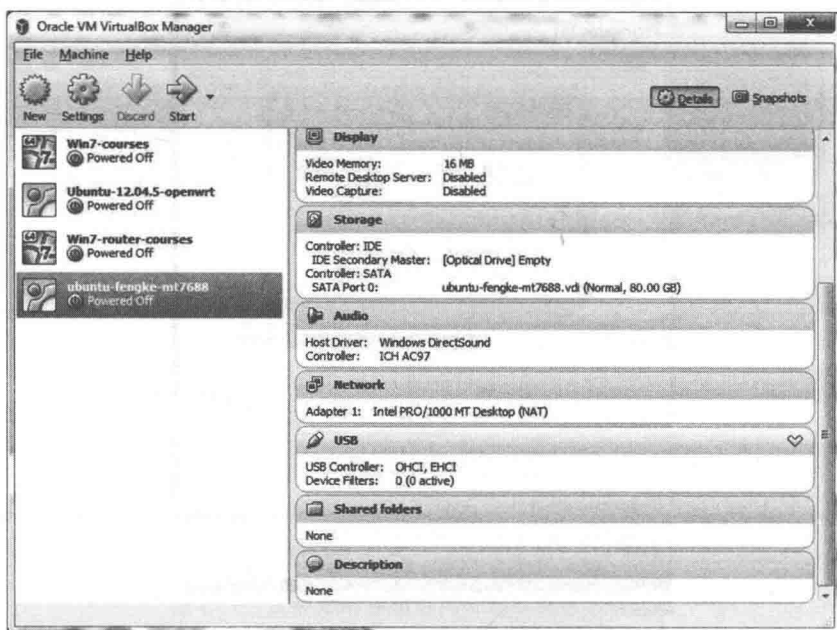


图 1-9 点击网络(Network)

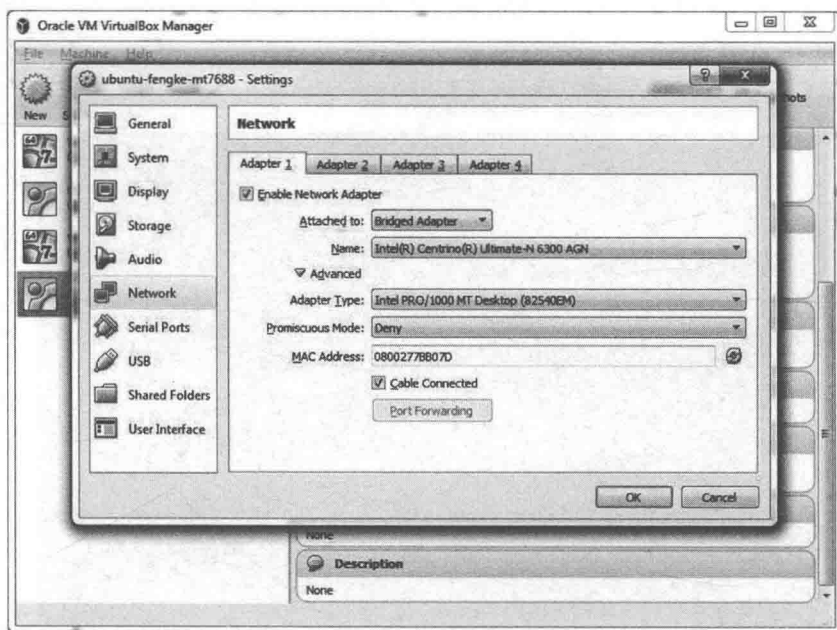


图 1-10 选择桥接模式(Bridged Adapter)

3. 启动虚拟机

启动虚拟机，先输入密码，如图 1-11 所示。

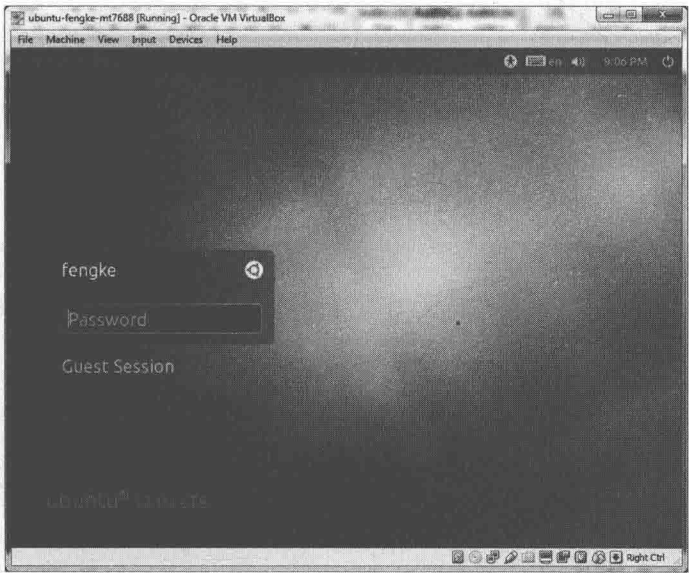


图 1-11 输入密码

注意：此虚拟机已经安装了客机增强功能(Guest Additions)。

1.1.2 安装开发环境

一开始必须在虚拟终端(Terminal)环境下安装常用工具软件。进入虚拟终端环境，如图 1-12 所示。

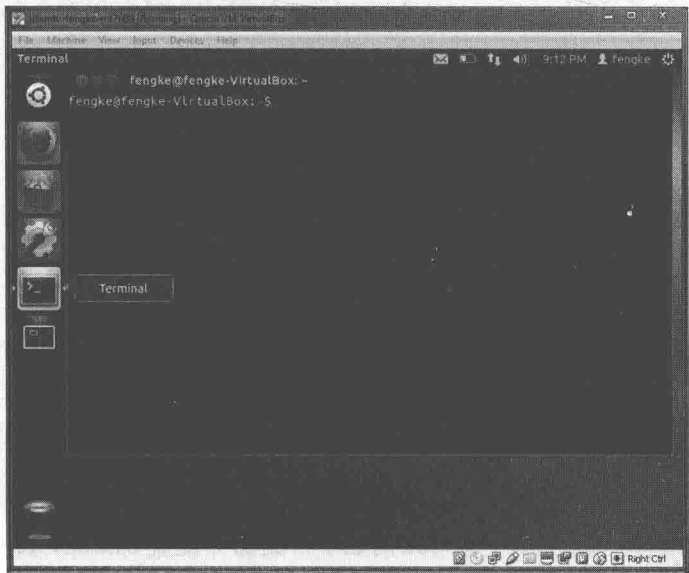


图 1-12 虚拟终端环境

1. 工具安装准备

依次输入如下命令(所有的命令都是在虚拟机里的虚拟终端输入):

```
sudo apt-get install vim
sudo apt-get install samba
sudo apt-get install openssh-server
```

如果第一次输入 `sudo` 命令,则会提示输入超级用户密码(密码是 `fengke`),如图 1-13 所示。如果希望一次性安装所有工具,则可以用如下命令(命令间用“;”分隔):

```
sudo apt-get install vim;sudo apt-get install samba;sudo apt-get install openssh-server
```

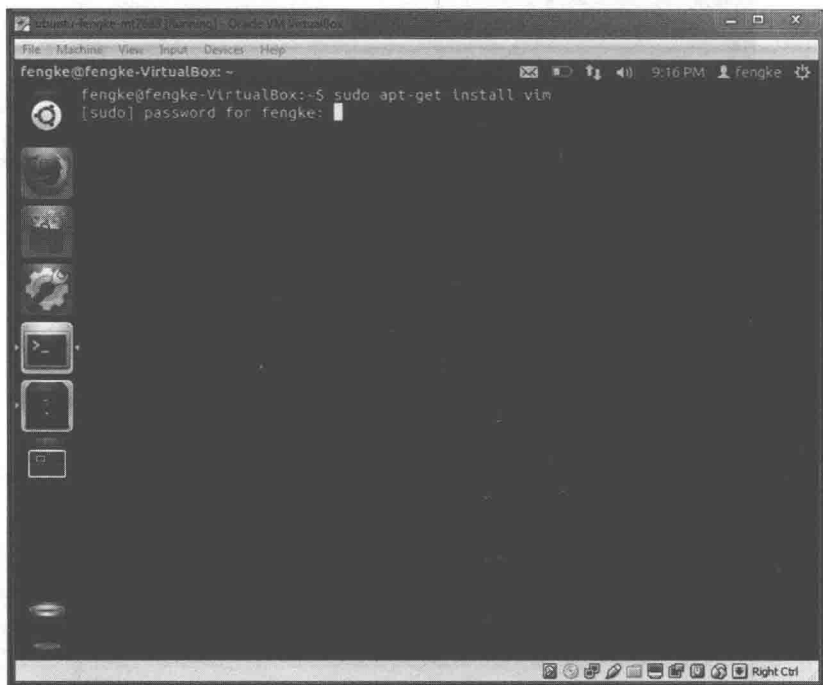


图 1-13 输入超级用户密码

2. samba 配置

(1) 用 `vim` 工具编辑 `samba` 配置文件 `/etc/samba/smb.conf`,如图 1-14 所示。

在文件 `smb.conf` 的末尾增加如下语句:

```
[fengke]
commet = share directory
path = /home/fengke
writeable = yes
```

创建 `samba` 用户并重启 `smaba` 服务:

```
sudo smbpasswd -a fengke --此命令要求输入密码(这里输入的密码是 fengke)
sudo service smbd restart --重启 samba 服务命令
```

(2) 如果不习惯用 `vim` 工具,也可以用如下一组命令配置 `samba`,如图 1-15 所示。相应的命令就不一一列举,用户可以分析一下各个命令的作用。如果有不了解的地方,可以参看后面的 `shell` 脚本讲解。

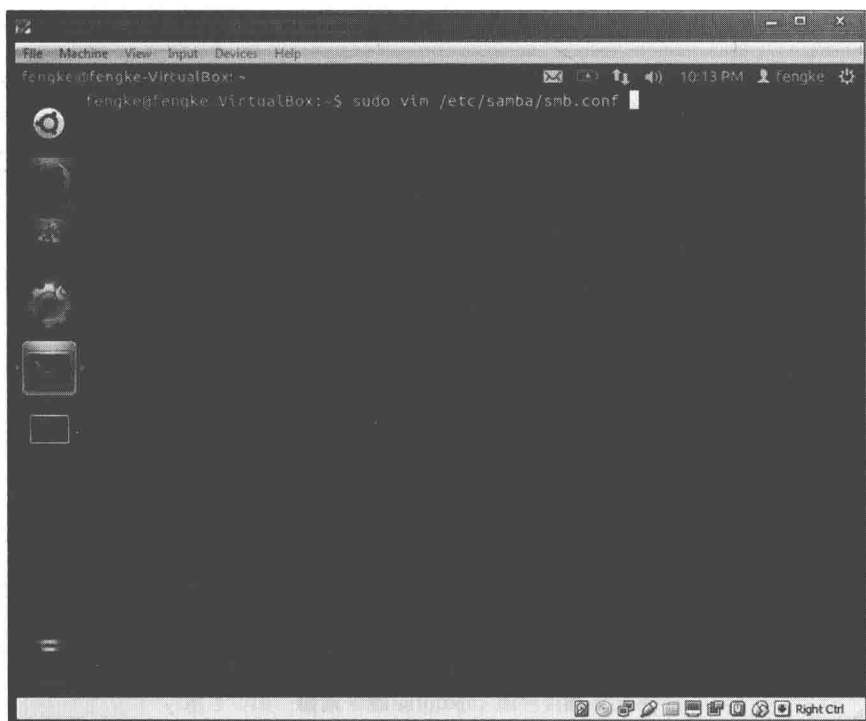


图 1-14 编辑 samba 配置文件

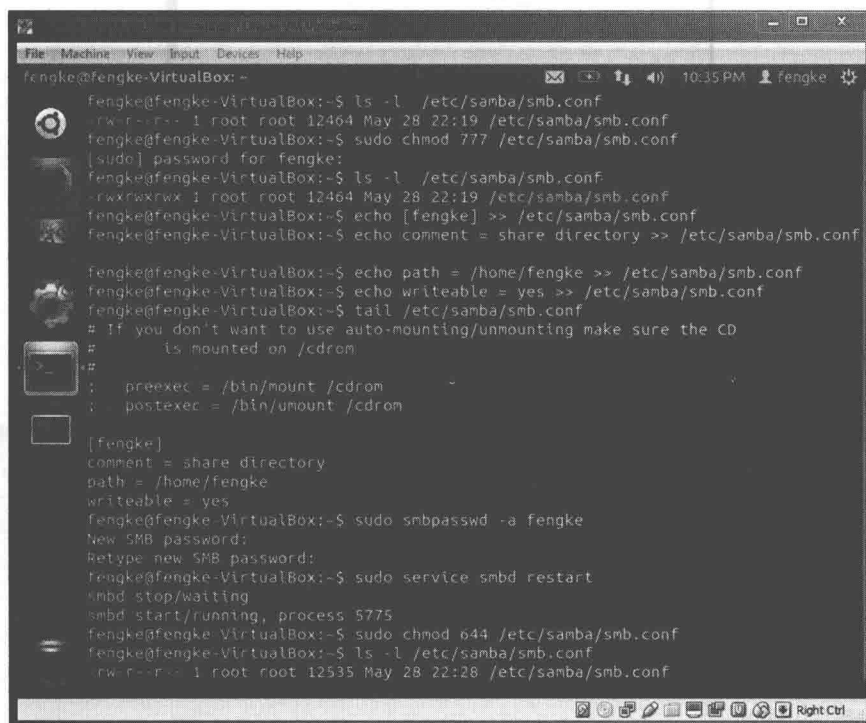


图 1-15 命令行编辑 samba 配置文件