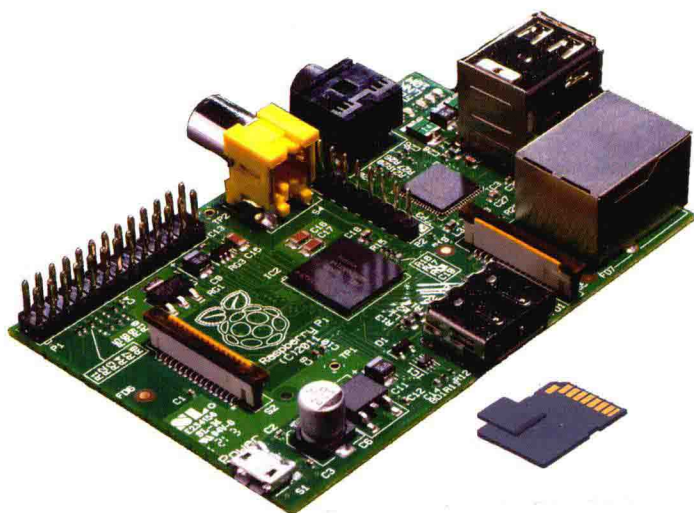


这不是一本介绍Python语言的书，也不是一本详细探索树莓派的书  
如果你买了树莓派但不知道如何开始，那么本书适合你  
如果你买了树莓派但不确定用它做什么，那么本书适合你



数字匠人

Apress®



Learn Raspberry Pi  
Programming

# Python 树莓派编程

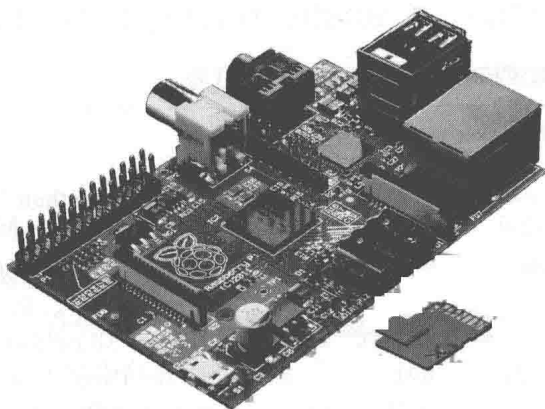
[美] 沃尔弗拉姆·多纳特 (Wolfram Donat) 著  
哈伊姆·克劳斯 (Chaim Krause) 审校  
韩德强 等译



机械工业出版社  
China Machine Press



数字匠人



Learn Raspberry  
Pi Pro

# Python

# 树莓派编程

[美] 沃尔弗拉姆·多纳特 (Wolfram Donat) 著  
哈伊姆·克劳斯 (Chaim Krause) 审校  
韩德强 王众 等译



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

Python 树莓派编程 / (美) 沃尔弗拉姆·多纳特 (Wolfram Donat) 著. 韩德强等译. —北京: 机械工业出版社, 2016.10

(数字匠人)

书名原文: Learn Raspberry Pi Programming with Python

ISBN 978-7-111-55030-3

I. P… II. ①沃… ②韩… III. 软件工具—程序设计 IV. TP311.561

中国版本图书馆 CIP 数据核字 (2016) 第 238493 号

本书版权登记号: 图字: 01-2014-7788

Wolfram Donat: Learn Raspberry Pi Programming with Python (ISBN: 978-1-4302-6424-8).

Original English language edition published by Apress L. P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA. Copyright © 2014 by Apress L. P. Simplified Chinese-language edition copyright © 2016 by China Machine Press. All rights reserved.

This edition is licensed for distribution and sale in the People's Republic of China only, excluding Hong Kong, Taiwan and Macao and may not be distributed and sold elsewhere.

本书原版由 Apress 出版社出版。

本书简体字中文版由 Apress 出版社授权机械工业出版社独家出版。未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

此版本仅限在中华人民共和国境内 (不包括香港、澳门特别行政区及台湾地区) 销售发行, 未经授权的本书出口将被视为违反版权法的行为。

## Python 树莓派编程

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 缪 杰

责任校对: 董纪丽

印 刷: 北京市荣盛彩色印刷有限公司

版 次: 2016 年 10 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 14.5

书 号: ISBN 978-7-111-55030-3

定 价: 49.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

## *The Translator's Words* 译者序

近年来，随着物联网、可穿戴设备、智能硬件、工业 4.0、互联网+等新技术、新概念的出现，涌现出了大量的创客及创客空间。在 2015 年两会上，李克强总理在政府工作报告中指出要把“大众创业、万众创新”打造成推动中国经济继续前行的“双引擎”之一。

树莓派自问世以来，就以出色的性价比受到众多计算机发烧友和创客的追捧。2016 年 2 月，推出了最新版的树莓派 3B 型，其搭载了 1.2GHz 的 64 位四核处理器（ARM Cortex-A53 1.2GHz 64-bit quad-core ARMv8 CPU）。在大大提高了处理能力的同时，还增加了 802.11 b/g/n 无线网卡和低功耗蓝牙 4.1 适配器等无线通信功能。另外，Linux 不再独享树莓派。2015 年，微软公司在推出 Windows 10 IoT Core 之初，树莓派作为 ARM 平台的代表，首先被支持。

Python 语言是一种面向对象的、解释型计算机程序设计语言。自问世以来，Python 以其简洁、易学、可扩展性以及丰富的库资源等优势，日益受到设计者的青睐。一些国际知名大学已经采用 Python 语言讲授程序设计课程，譬如卡耐基梅隆大学的编程基础、麻省理工学院的计算机科学及编程导论。另外，众多开源的科学计算软件包都提供了 Python 语言的调用接口，譬如著名的计算机视觉库 OpenCV、三维可视化库 VTK、医学图像处理库 ITK 等等。

本书的作者沃尔弗拉姆·多纳特（Wolfram Donat）是一名经验丰富的计算机工程师，他以工程的视角诠释了一个个工程项目的实现。本书有别于其他编程语言类图书，在介绍完基本的软硬件平台后，通过网络机器人、气象站、媒体服务器、家庭安防系统、猫玩具、无线电遥控飞机、气象气球、潜水艇等 8 个项目，介绍了网络通信协议、文件存储、传感器、执行部件等基础知识的编程方法。使读者真正理解了“通过做来学习”的先进理念。在突出实践能力培养的同时，又在每个项目中针对嵌入式系统的各个功能模块给出了相应的基础知识的介绍。另外，原书作者还非常重视工程习惯的培养，在实践过程中给出了一般的安全规则。以译者 20 多年嵌入式系统教学、工程经验来看，本书十分适合作为本科、高职高

专各专业的创新课程教材。

本书由北京工业大学计算机学院的部分教师翻译，其中简介和第1~4章由韩德强翻译，第5~7章由王众翻译，第7、8章由杨洪善翻译，第9章由吴有仙翻译，第10、11章由张丽艳翻译，第12~14章由王宗侠翻译，全书的审校由韩德强完成。

在本书的翻译过程中得到了机械工业出版社华章公司的张国强、缪杰两位编辑的大力支持，在此对二位表示由衷的感谢！

限于译者的水平，翻译中难免有错误或不妥之处，真诚希望各位读者批评指正。

韩德强

2016年8月于北京工业大学

在 2006 年，当 Eben Upton 和其他树莓派基金会的创办人看到大学计算机专业学生的编程状况时，他们感到无比沮丧。在美国，计算机专业的编程课程被缩减为“CS 101：如何使用 Word 程序”和“CS 203：优化你的 Facebook 主页”。他们意识到，不是所有人在上大学之前都学过编程。因此，他们酝酿了一个计划——打造一种小型廉价的计算机，使孩子们可以更加方便地学习编程，就如同昔日的 Amiga、Spectrum 和 Commodore 64。随后，他们便开发了一块以 ARM 处理器为核心的开发板，配有 512MB 内存并具有视频处理功能的 GPU，集成了如 USB 键盘、鼠标、HDMI 输出端口等接口。为了使其更易于编程，他们将 Python 设定为其主要的编程语言——一种强大且易学的脚本语言。这样，树莓派就诞生了。

多年以前，我在 Commodore VIC 20 上用 BASIC 语言编写了我的第一个程序。当时的计算机内存只有 5KB，比现如今很多的微处理器的计算能力都要弱，但我仍编写出了一个很有趣的游戏，并利用盒式磁带保存程序的进度。在之后的几年里，我先后使用过不同的计算平台，从 Windows 3.1 到 Macintosh OS 8，再到几个 Linux 系统。然而，树莓派的出现为陈旧的计算环境注入了一股新的气息，我为之激动不已。一方面是因为它具有小巧而便宜的特点，另一方面则是因为它易与外界进行交互——这对于那些感兴趣设计物质世界的人而言，是一种莫大的福音。所以当我听到它的发售消息之后，就同亿万的爱护者、黑客和工程师一样，立即对树莓派进行了预订并迫切期盼着它的到来。之后，我便开始用它构建一些东西并一发不可收拾了。

如果你买了树莓派但不知道如何开始，那这本书适合你。

如果你买了树莓派但不确定用它做什么，那这本书适合你。

如果你正犹豫是否要买树莓派，并思索着：“为什么我要买呢？我又做不出什么很酷的东西”，那毫无疑问，这本书更适合你。

这不是一本关于 Python 语言的教材，也不是一本详细探索树莓派的书籍。它以一种轻松的方式将你引向这台微型计算机。我希望你读完本书后，在进行创新的同时也能清楚地意识

到树莓派所能做的一切。

你可以按照本书的进度完成各种项目，也可以自由选择一些自己感兴趣的项目。在实践的过程中，我希望你能熟悉 Python 语言和树莓派（它们能让你走得更远），创建自己的项目，或许还能帮助其他同样对此感兴趣的人。总之，我希望你会喜欢本书及书中的项目。因为写作本书真的是一种非凡的体验！

好好享受这本书吧！

## 致谢

写一本书可能是一个人的工作，但绝对少不了别人的帮助。很多人都为这个项目做出了大量贡献，在此，我向他们致以最诚挚的谢意。感谢 Kevin 保证整件事情顺利进行，感谢 Anne Marie 理清我想说明的事情，感谢 Roger LeBlanc 使得这本书看起来像是我知道我在做什么。感谢作为技术指导的 Chaim。确保了我写下的东西和实际的执行效果一致。并且感谢 Michelle，最初能够注意到我。

最后，特别感谢 Becky 和 Reed。这是献给你们的。

# Contents 目 录

译者序  
前 言

<b>第 1 章 树莓派简介</b> ..... 1	
1.1 树莓派的历史..... 2	
1.2 探索树莓派..... 5	
1.2.1 SD 卡..... 5	
1.2.2 电源..... 6	
1.2.3 HDMI 接口..... 6	
1.2.4 以太网口和 USB 接口..... 6	
1.2.5 音频和 RCA 视频插孔..... 7	
1.2.6 GPIO 引脚..... 7	
1.2.7 片上系统..... 7	
1.2.8 树莓派同相似设备之间的对比..... 8	
1.3 树莓派的硬件需求..... 8	
1.3.1 通电..... 8	
1.3.2 添加显示器..... 9	
1.3.3 添加 USB 集线器..... 9	
1.3.4 使用 USB 无线网络适配器..... 10	
1.4 树莓派操作系统..... 11	
1.4.1 格式化 SD 卡..... 11	
1.4.2 使用 NOOBS..... 11	
1.5 连接外围设备..... 12	
1.6 配置树莓派..... 12	
1.7 关闭树莓派..... 14	
1.8 总结..... 14	

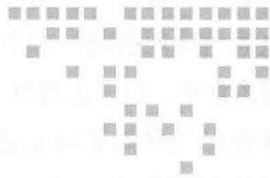
<b>第 2 章 轻松掌握 Linux</b> ..... 15	
2.1 开始使用树莓派的 Linux..... 16	
2.1.1 Linux 文件和文件系统..... 17	
2.1.2 root 用户与 sudo..... 18	
2.2 命令..... 18	
2.3 练习：在 Linux 文件系统中 进行导航..... 22	
2.4 Linux 中的 shell..... 23	
2.5 包管理器..... 24	
2.6 文本编辑器..... 25	
2.6.1 vim、emacs 与 nano..... 26	
2.6.2 Leafpad..... 29	
2.7 总结..... 30	
<b>第 3 章 Python 介绍</b> ..... 31	
3.1 脚本语言与程序设计语言..... 31	
3.2 Python 语言的理念..... 33	
3.3 Python 入门..... 34	
3.3.1 通过 IDLE 运行 Python..... 34	
3.3.2 通过终端运行 Python..... 37	
3.3.3 通过脚本运行 Python..... 37	
3.3.4 探究 Python 的数据类型..... 38	
3.4 利用 Python 进行编程..... 44	
3.4.1 if 测试..... 45	
3.4.2 循环..... 47	
3.4.3 函数..... 48	



3.4.4	对象和面向对象编程	49	5.2.1	网络通信协议	67
3.5	总结	50	5.2.2	网页格式	68
<b>第4章</b>	<b>电子知识大科普</b>	<b>51</b>	5.2.3	请求举例	68
4.1	基本电路常识	52	5.3	网络机器人的概念	69
4.2	开发所需要的工具	53	5.4	解析网页	70
4.2.1	螺丝刀	54	5.5	利用 Python 模块编码	71
4.2.2	钳子和剥线钳	54	5.5.1	使用 Mechanize 模块	71
4.2.3	钢丝钳	55	5.5.2	用 Beautiful Soup 解析	72
4.2.4	锉刀	55	5.5.3	利用 urllib 库下载	72
4.2.5	放大镜灯	55	5.6	决定下载的内容	73
4.2.6	热胶枪	56	5.6.1	选择起点	73
4.2.7	各类胶水	56	5.6.2	存储文件	74
4.2.8	万用表	56	5.7	编写 Python 网络机器人	74
4.2.9	电源	57	5.7.1	读取一个字符串并提取 所有链接	75
4.2.10	实验板	57	5.7.2	寻找并下载文件	75
4.2.11	插线板	58	5.7.3	测试网络机器人	76
4.2.12	电烙铁	59	5.7.4	创建目录并实例化一个列表	77
4.3	一般的安全规则	60	5.8	最终代码	79
4.3.1	认真对待温度	60	5.9	总结	80
4.3.2	认真对待锋利的物体	60	<b>第6章</b>	<b>气象站</b>	<b>81</b>
4.3.3	带安全镜	61	6.1	零件购物清单	82
4.3.4	准备好灭火器	61	6.2	使用 I2C 协议	83
4.3.5	在手边放置一个急救包	61	6.3	使用风速计	84
4.3.6	在通风的环境下工作	62	6.3.1	构造风速计	85
4.4	整理好你的工作环境	62	6.3.2	将风速计与树莓派连接	87
4.5	福利: 焊接技术	63	6.3.3	根据风速纠正的每秒转数	88
4.6	总结	64	6.4	连接数字指南针	90
<b>第5章</b>	<b>网络机器人</b>	<b>65</b>	6.5	连接温度 / 湿度传感器	92
5.1	机器人礼仪	66	6.6	连接气压计	94
5.2	网络的连接	67	6.7	连接所有部件	95

6.8	最终代码	97	8.5	最终代码	125
6.9	总结	98	8.6	总结	126
<b>第7章</b>	<b>媒体服务器</b>	99	<b>第9章</b>	<b>猫玩具</b>	127
7.1	零件购物清单	99	9.1	零件购物清单	127
7.2	使用 NTFS 格式的硬盘	100	9.2	玩具背后的设计理念	128
7.3	安装 Samba	102	9.3	创建和使用随机数	129
7.3.1	配置 Samba	103	9.4	使用 GPIO 库	130
7.3.2	Linux 权限设置	105	9.5	控制舵机	132
7.3.3	修复单引号错误	106	9.6	构建舵机结构	134
7.3.4	重新启动 Samba 服务	106	9.7	构建激光结构	135
7.4	与 Linux / OS X 连接	107	9.8	将激光连接到舵机上	137
7.5	Python 在哪	108	9.9	连接运动传感器	138
7.6	总结	109	9.10	连接所有的部件	140
<b>第8章</b>	<b>家庭安防系统</b>	110	9.11	最终代码	142
8.1	用于安防的狗	111	9.12	总结	143
8.2	用于安防的树莓派	111	<b>第10章</b>	<b>无线电遥控飞机</b>	144
8.3	使用传感器网络	112	10.1	零件购物清单	145
8.3.1	了解下拉电阻	112	10.2	将 GPS 接收器连接至树莓派	146
8.3.2	零件购物清单	113	10.3	设置日志文件	148
8.3.3	以无线方式连接你的 家庭网络	113	10.4	格式化 KML 文件	149
8.3.4	访问 GPIO 引脚	116	10.5	使用线程和对象	150
8.3.5	设置运动传感器	118	10.6	设置自启动	152
8.3.6	设置磁簧开关	119	10.7	连接所有部件	153
8.3.7	设置压力开关	121	10.8	最终代码	156
8.3.8	连接磁传感器	121	10.8.1	飞机飞行程序	157
8.3.9	设置树莓派的摄像机	122	10.8.2	KML 转换程序	157
8.3.10	利用树莓派发送短信	123	10.9	总结	158
8.3.11	实现回调	124	<b>第11章</b>	<b>气象气球</b>	159
8.4	连接所有的部件	124	11.1	零件购物清单	160

11.2	设置 GPS 接收器	160	12.8.3	电机的防水外壳	191
11.3	存储 GPS 数据	162	12.8.4	连接至 nunchuk	193
11.4	安装 PiFM	163	12.8.5	装配最终的产品	194
11.5	安装 festival	164	12.9	总结	197
11.6	安装 FFMPEG	164	<b>第 13 章 Gertboard 开发板</b>		198
11.7	准备树莓派	165	13.1	检查扩展板	199
11.8	使用线程和对象	166	13.1.1	GPIO 引脚	199
11.9	连接所有部件	167	13.1.2	Atmega 芯片	200
11.10	观察照片结果	169	13.1.3	AD、DA 转换器	201
11.11	最终代码	170	13.1.4	I/O 部分	201
11.12	总结	171	13.1.5	电机控制器	202
<b>第 12 章 潜水器</b>		172	13.1.6	集电极开路驱动器	203
12.1	零件购物清单	173	13.1.7	跳帽	204
12.2	访问树莓派的 GPIO 引脚	174	13.2	示例项目	205
12.3	安装树莓派摄像头模块	176	13.2.1	配置初步跳帽的设置	205
12.4	控制潜水器	177	13.2.2	点亮 LED 灯	206
12.4.1	连接 Wiichuck 适配器	178	13.2.3	用电机控制器进行实验	208
12.4.2	激活树莓派的 I2C	179	13.2.4	使用集电极开路驱动器	208
12.4.3	测试 nunchuk	180	13.2.5	使用数字 / 模拟转换器	209
12.4.4	从 nunchuk 读取数据	181	13.3	总结	210
12.4.5	通过 nunchuk 控制潜水器电机和摄像头	184	<b>第 14 章 树莓派和 Arduino</b>		211
12.5	远程启动程序	186	14.1	探索 Arduino	212
12.6	最终的控制程序	188	14.2	在树莓派中安装 Arduino IDE	213
12.7	最终代码	188	14.3	控制舵机	218
12.8	构造潜水器	190	14.4	Arduino 和 Gertboard	219
12.8.1	构建框架	191	14.5	总结	220
12.8.2	构建树莓派的外壳	191			



第 1 章

Chapter 1

## 树莓派简介

如果你现在有一台树莓派，你会用它来做什么呢？也许你已经熟悉了树莓派的基本结构，正在考虑可以拿它来做些什么；也许你之前使用过计算机但并不熟悉 Linux 或树莓派默认的 Raspbian 操作系统；也许你已经是一位 Linux 系统的爱好者，但却不知道如何用 Python 进行编程，而且正准备好好学一学；也许你根本就不了解计算机，仅仅停留在点击“开始”菜单，转发电子邮件，上网等操作，但是听说过“树莓派”这么个新奇的东西，想了解一下它到底是什么。

无论是何原因，欢迎你的到来！你即将加入我们的大家庭——这个家庭专业性并不是很强，因为你只需花费 35 美金及必要的邮费，无需其他，便可成为我们的一员。作为一名会员，你将会和所有想听你分享经验的人们讨论你对软件包管理器、ARM11 处理器，或是 .config 文件的一些见解，会了解关于驱动器和 API 的一些信息，会熟悉关于舵机、LED 灯、板载摄像头的内容，最重要的是，你可以与微型计算机进行通信，用任何一门编程语言（本书主要使用的是 Python 语言）进行编程，构建项目，在树莓派上实现这些项目，这样树莓派便可与真实世界进行交互，并做出一些非常酷的东西。

通过阅读本书，我将引导你进入树莓派的俱乐部。在这里，你之前的经验并不重要，因为我将一步步带着你对树莓派进行配置，这样你就可以轻松地使用它了。我会尽量将 Linux 的环境介绍得详细透彻，以便你可以理解屏幕背后所发生的一切。同时我会用比较长的篇幅来介绍 Python 语言，这是当前极客们热衷的脚本语言。谷歌使用 Python，NASA 使用 Python，Facebook 也使用 Python。Perl 的时代已经过去，让我们好好探究 Python 这种语言。我还会在书中介绍一些构建电子项目的基本常识——这是一些技术或编程书籍仅作参考

介绍，甚至完全忽略的部分。在构建好的项目同时，也还有一些安全因素需要考虑（例如，我曾经因为将电池短接而引起一次小型爆炸）。例如，你会学到如何焊出一个好的焊点，如何避免被刀片划伤食指，以及怎样区别 40Ω 和 40KΩ 的电阻。

当然，如果你已经熟悉了以上这些事情，可以跳过前面的介绍直接进入后面关于项目的部分。所有的项目都是用 Python 语言编写的，并且它们都可在一个周末（或一两个月，当然，这取决于你的积极性以及其他你需要做的事情）搭建完成。我会在每个项目之前为你准备一份购物清单，以及买到这些部件的地址，之后便直入主题。这些项目彼此之间并无依赖关系，在复杂程度上彼此也没有特定的顺序，这就意味着如果你想跳过家庭媒体服务器项目直接进入猫玩具项目也是没问题的。

那树莓派到底可以用来完成什么样的项目呢？你会惊讶地发现，树莓派小巧的尺寸并不能掩盖它强大的计算能力。它的应用范围很广，从网络服务器、车载电脑（carputer），到集群计算都可以看到它的身影。我希望当你读完本书之后，不仅可以发现一些新奇的想法，学到一些技术，最主要的是通过所学到的知识将你的想法变为现实。

无论你为何选择本书，你的主要目的就是快乐地学习并且学到一些东西！我会尽我所能去帮助你。

## 1.1 树莓派的历史

对外人而言，树莓派的确是很新颖的东西，在博客中有许多人也这样认为，还有一部分人甚至完全不知道树莓派是什么。大量树莓派的文章仍旧会以“树莓派是一个小型的、信用卡大小的计算机，爱好者们可以用其……”作为开始。但不同于树莓派，当人们谈及 Arduino 时，即便大多数人不知道 Arduino 是什么或是做什么用的，但至少都听说过。因为早在 2005 年，Arduino 就在全球范围内的爱好者们、极客们和喜爱 DIY 的人们中赢得了很好的口碑。

---

### Arduino

Arduino 是一个微控制器平台，它安装在可以轻松插入绝大多数计算机的芯板上。这使得用户可以通过类似 C 的编程语言对板上的 Atmega 芯片进行编程，以实现各种需求。这种程序称为 Sketch。一个典型的 Arduino Sketch 程序如下所示：

```
#include <Servo.h>
void setup()
{
    myservo.attach(9)
}
```

```
void loop()
{
  myservo.write(95);
  delay(100);
  myservo.write(150);
  delay(100);
}
```

以上代码的作用是控制一个和 Arduino 相连的舵机（一个可以通过软件精确控制转动角度的小型马达）持续进行前后转动，每次转动间隔 1 秒。

尽管其功能不像树莓派那样强大，但 Arduino 已经可以完成很多事情，这使得公众对一般的电子项目（尤其是微控制器）更加熟知。我们将在第 14 章对 Arduino 和树莓派如何相互完善进行更深入的介绍。

---

树莓派已经问世很多年了，早已不是什么新的词汇。它的创始人——Eben Upton、Rob Mullins、Jack Lang 和 Alan Mycroft——在 2006 年第一次提出了廉价 PC 机的想法。在英国剑桥大学的时候，他们就意识到：随着像 Commodore 64、Amiga 和 Spectrum 这种廉价的个人计算机逐步退出市场，取而代之的台式机和笔记本电脑的价格又十分高昂（若不是几千美元的话，也需要几百美元），年轻人无法随意地在这种家庭主要的电子产品上练习编程，而这一定会严重影响年轻人的编程能力。

与此同时，这些创始人意识到当前许多大学的计算机科学课程已经被缩减为教授“微软 Word 101”和“如何创建网页”。4 位创始人希望帮助新生提高编程知识，或许以后计算机科学和工程类的课程会变得更加有意义。

很明显，为实现以上目的，需要一台便宜的计算机。为此他们尝试过许多微控制器、各种芯片、面包板、PCB 板，等等，直到 2008 年，这个想法才得以实现。随着移动设备的爆炸式增长，芯片变得更小，更便宜，性能也更加强大。他们使用这些芯片设计了一台支持多媒体的设备，而不是仅支持命令行编程，做到这一点他们认为很重要（见图 1-1）。年轻人似乎更喜欢具备媒体功能的设备，因此也就更有可能用此设备去编程。

2008 年，4 位创始人，和 Pete Lomas 和 David Braben 一起，创立了树莓派基金会，3 年后，该基金会成为第一个大规模生产树莓派的生产线。



树莓派（Raspberry Pi）这个名字同早期微机根据水果命名的规则一样，如早期的苹果（Apple）和橘子（Tangerine），而派（Pi）则来源于 Python 这个脚本语言。

---

一年内，基金会卖出了 100 多万台设备。基金会成员多次表示他们对这种爆炸式的热情感到十分震惊。最初的目的仅仅是推出一种廉价的、可编程的设备，使教育工作者和学生获

益，显然现在这个目的已经达成了。不仅如此，现在的成果比最初设想的要大得多。很明显，他们并不是唯一希望拥有廉价可编程设备的一类人，世界各地的爱好者，连同 element14、Premier Farnell 和 RS Electronics 都迫切需要订单，以至于那些预订了树莓派的人不得不等待半年的时间，随后产量才满足需求量。许多消费者现在或之前都是程序员，现如今都渴望一览这台新的体积小性能强的计算机。（我第一次学习编程是在 Commodore VIC-20 上，这台设备仅有 20KB 的内存，当时用的还是 BASIC 语言……好吧！这都是很久以前的事了。）

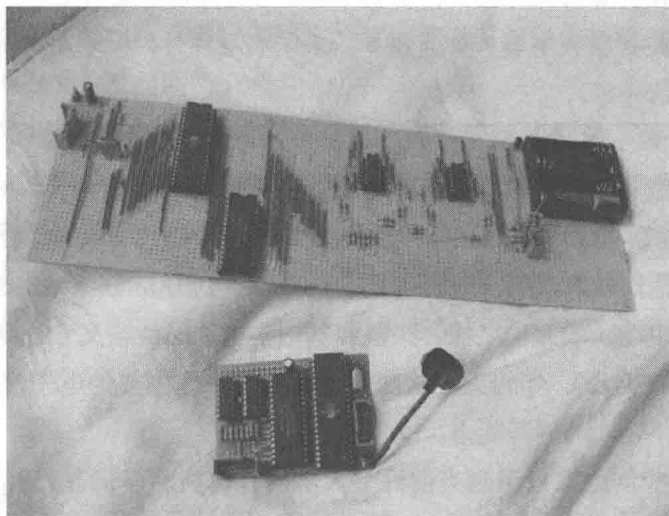


图 1-1 Eben Upton 在 2006 年设计的树莓派原型 (image©Raspberry Pi Foundation)

但无论是之前还是现在，人们在各领域内都使用树莓派。就像美国树莓派基金会网页上写的那样：

我们已经收到来自教育机构数不尽的热情、支持和帮助。在看到来自机构大量的咨询时，我们十分激动，而当人们对设备的使用目的与我们的初衷相差甚远时，我们又感到有些羞愧。在发展中国家，由于部分地区的电力无法为传统的桌面 PC 设备提供所需要的功率和硬件设备，因此他们对树莓派很感兴趣。医院和博物馆已经联系我们，希望树莓派能够驱动显示设备。一些重度残疾孩子的父母也跟我们联系，谈论关于监控和可沟通应用的事情；与此同时，似乎还有数以百万的人正拿着电烙铁准备制作机器人呢。

幸运的是，供应及时满足了需求。现在买树莓派便无需等待了，而且每位用户也不再限购了。树莓派基金会正策划一个“买一赠一”的活动，即每卖出一台树莓派便向教育机构捐赠一台树莓派。最近发行的树莓派板载摄像头是一台小型芯片集成摄像头，可直接插入树莓派，用户可以用它拍摄静态照片，也可拍摄视频。基金会保证将会为这台小型计算机开发更多的外部设备。如今创始人正在积极说服其他公司复制他们的模式，接下来会发生的事

情，让我们拭目以待吧。

## 1.2 探索树莓派

那树莓派上究竟有什么呢？有什么能适合这个如此之小的设备呢？

目前为止，一共有两款树莓派：A 版和 B 版（B 版详情见图 1-2）。两个版本之间的差距非常小，B 版仅比 A 版多了一点功能，当然价格也要稍微贵一些。A 版内存为 256MB，而 B 版内存为 512MB；A 版有一个 USB 接口，而 B 版有两个。A 版不具备以太网口，而 B 版有一个。你可以根据自己的需求订购任何一款，A 版为 25 美元，而 B 版售价 35 美元。由于二者在价格上差距不是很大，因此我建议购买 B 版。因为有一个或两个 USB 端口之间的差别是非常大的，而且以太网线的接入会使得诸如更新设备或者连接至小型 ad-hoc 网络的操作变得十分简单。

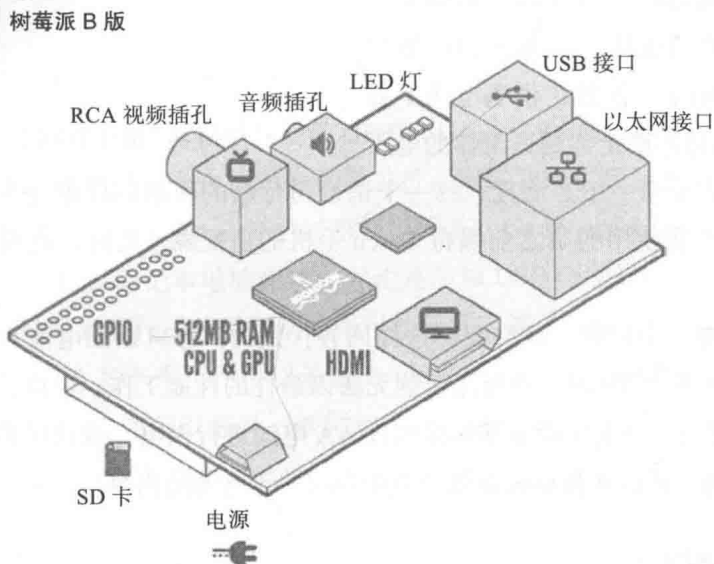


图 1-2 树莓派 B 版的结构 (image©Raspberry Pi Foundation)

树莓派（自此开始，我们假设你买的是树莓派 B 版）的尺寸为 85.6mm × 56mm × 21mm（很明显，这不是在美国制造的），其中包含着一些大小不同的端口和 SD 卡接口。参考图 1-2，并将其按照逆时针顺序旋转，在接下来的章节中，我会对一些部件进行更详细的介绍。

### 1.2.1 SD 卡

从图 1-2 中可以看到，在树莓派这个小板子上搭载了的东西。树莓派中最节省空间的



特性就是它没有像台式机中硬盘一样的硬件设备，取而代之的是 SD 卡，它的作用类似于固态硬盘。你可以根据不同场合更改 SD 卡的大小来改变存储容量。至少需要 2GB 的存储空间才能正常启动系统，如果你想安装其他软件，则至少需要 4GB 的存储空间。经测试，树莓派最高可支持 32GB 的存储卡，所以一旦你需要十分大的存储容量时，请注意及时备份。

## 1.2.2 电源

树莓派的电源采用 5V micro-USB 输入，同一般手机或平板的电源一致。实际上，一般手机的充电器均可作为树莓派供电（见图 1-3）。

提醒一句，树莓派没有板载的电源稳压器！如果你习惯了使用 Arduino，你就会知道即便电源输出 9V 电压，也是安全的，因为 Arduino 有板载的电源稳压器。但如果你给树莓派提供 9V 的电压，树莓派很可能自此以后就无法使用了。因此，供电电压不要超过 5V——如果你不确定你的适配器的电压

输出值，用万用表测量一下。而且，每一个销售树莓派的网站同样也会有合适的适配器供你选择。至于我？我使用的是之前黑莓 Torch 手机的适配器（是的，我用的黑莓手机，别嘲笑我）。

如果你正考虑一个问题，或许我会这样回答：是的，你可以用电池为树莓派供电，但当电池的输出电压低于 5V 时，树莓派可能无法以最佳的性能工作。也许最简单的方法是用一枚 9V 电池，或用一个电压调节器连接四枚 AA 电池进行供电，或使用类似于遥控车中的电池组也是可以的。我将在移动树莓派章节中介绍关于电源的内容。

## 1.2.3 HDMI 接口

树莓派配备了 HDMI（High Definition Multimedia Interface，高清多媒体接口）输出接口，很多人评价说这是树莓派最有特色的地方，因为这使得树莓派拥有处理 10 亿像素/秒的能力，能够输出 1080P 高清图像。通过对芯片内 OpenGL 和 OpenVG 库的调用，树莓派的 GPU 可以处理蓝光级别的视频信息。

## 1.2.4 以太网口和 USB 接口

以太网和 USB 接口的功能（B 版树莓派）是通过板子上的 LAN9512 芯片实现的。根据



图 1-3 常见的 USB 充电适配器