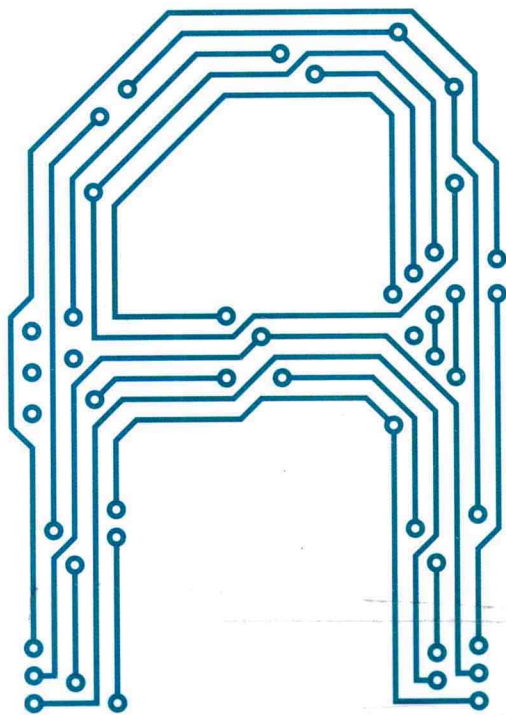


HZ BOOKS
华章科技

国内首本Arduino著作，全面而系统地讲解Arduino平台的功能特性和开发方法。
注重实战，包含大量实战案例，不懂硬件的工程师也能迅速搭建产品原型。
资深Arduino玩家、硬件工程师主笔，创客联盟与学术专家联袂推荐，权威性毋庸置疑。

单片机与嵌入式



AVR篇

Arduino 开发实战指南

程晨 著

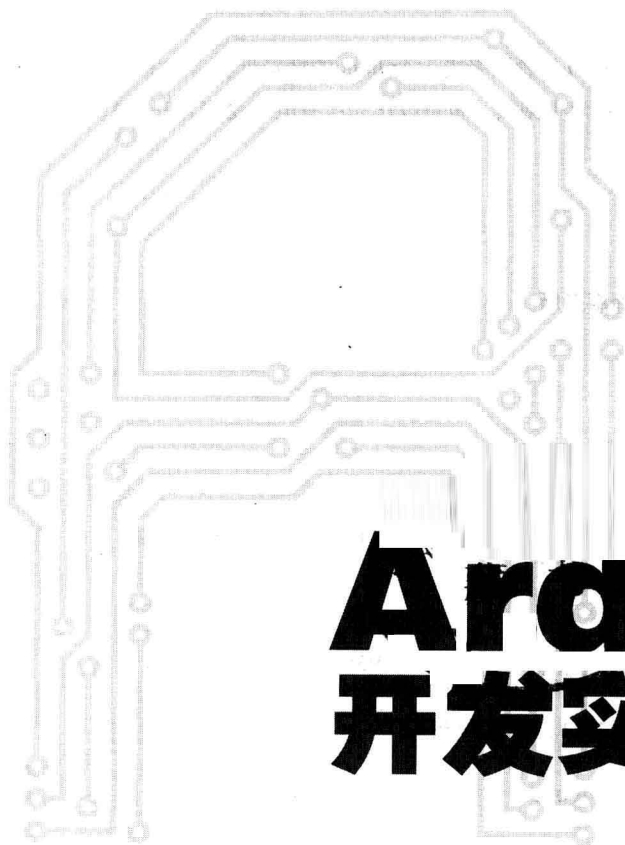


机械工业出版社
China Machine Press



单片机与嵌入式

程晨 著



AVR篇

Arduino 开发实战指南



机械工业出版社
China Machine Press

Arduino是一个注重实际动手操作的产品，所以本书以实际应用为纽带将各个章节联系起来。本书首先介绍Arduino的一些基础知识，接着针对具体应用介绍了一些扩展板以及Arduino扩展库，最后应用之前的内容完成了具有视频监控功能的履带车、遥控机械臂以及双足机器人的制作。

本书内容循序渐进，图文并茂，可以带领读者走入Arduino的精彩世界。本书适合电子专业、交互设计专业、新媒体技术专业学生阅读，也可以作为所有电子爱好者开展Arduino制作项目的参考手册。

封底无防伪标均为盗版

版权所有，侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目 (CIP) 数据

Arduino开发实战指南：AVR篇 / 程晨著. —北京：机械工业出版社，2012.2

ISBN 978-7-111-37005-5

I. A… II. 程… III. 单片微型计算机—指南 IV. TP368.1-62

中国版本图书馆CIP数据核字 (2011) 第281978号

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：秦 健

北京京师印务有限公司印刷

2012年3月第1版第1次印刷

186mm×240mm · 21印张

标准书号：ISBN 978-7-111-37005-5

定价：59.00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) 88378991；88361066

购书热线：(010) 68326294；88379649；68995259

投稿热线：(010) 88379604

读者信箱：hzjsj@hzbook.com

前 言

在2011年举行的Google I/O开发者大会上，Google发布了基于Arduino的Android Open Accessory标准和ADK工具，这使得大家对Arduino的前景十分看好。Phillip Torrone大胆地预测Google将用Android+Arduino的形式掀起自己的“Kinect模式”浪潮。目前，国内关注Arduino的人越来越多，但介绍Arduino的书籍却很少。笔者由于工作的关系，接触Arduino较早，所以希望通过自己的努力让更多的人了解Arduino，在近一年的时间里，通过不断学习、查阅Arduino相关知识，终于完成了书稿的撰写工作。但在书稿完成之后，心中却一直忐忑不安，Arduino是一个介于软件与硬件之间的产品，系统性不是很强，加上笔者水平有限，拙著中一定存在不少的缺点与漏洞，为此，笔者先为书中的不足之处致以真诚的歉意，同时诚挚地欢迎广大读者提出宝贵的意见并不吝赐教。

本书的内容及面向的读者

Arduino是一个注重实际动手操作的产品，所以本书以实际应用为纽带将各个章节联系起来。本书共9章，首先介绍Arduino的一些基础知识，接着针对具体应用介绍了一些扩展板以及Arduino扩展库，最后应用之前的内容完成了具有视频监控功能的履带车、遥控机械臂以及双足机器人的制作。

因为Arduino本身具有简单易用的特点，所以本书面向的读者是所有有兴趣使用Arduino进行项目开发的人。

当然，根据读者的情况不同，本书的阅读方式也不同。

如果读者是一个之前没有进行过单片机开发也没有进行过软件开发的人，现在想使用Arduino来实现自己的一些想法，那么首先要阅读本书的前两章，了解一些简单的编程思想以及程序结构，接下来阅读第3章的目录，了解Arduino都有什么基本函数，具体内容可以先不用看，当你之后使用这些函数遇到问题时再回过头来看一看相应的函数说明。然后将Arduino接到你的电脑上，翻开第4章，根据书中的内容，边学习边实践，4.5节可以跳不过

看。第5~7章介绍了Arduino周边的资源，以便帮助你尽快地实现想法，这3章的内容也可以采用跳跃式的阅读方式。第8、9章会告诉你前3章的内容是如何结合起来的，建议按照书中的内容至少动手完成一个项目的制作。

如果读者之前进行过AVR单片机的开发，想了解Arduino一些底层的知识，那么第2章的知识就可以跳过了，在简单地翻阅第3章的内容后，直接进入第4章，把Arduino连到电脑上实践一下，再回过头阅读第3章中关于Arduino的基本函数，结合自身已有的AVR单片机的知识了解Arduino底层的工作机制。需要说明的是，这里需要读者自己花一些精力，可能还需要学习一些C++方面的知识。第5章对Arduino硬件原理进行了详细介绍，若读者之前学习过，这一章可以选择性学习。第6章介绍的是Arduino的扩展库，如果读者也想开发一些Arduino扩展板，并以库的形式提供扩展板的软件资源，那么建议先学习最后一节，再从6.1节开始学习，深入地理解这些扩展库是如何与Arduino结合在一起的。至于剩下几章的内容，如果用开发单片机的思路来完成也是不难的，所以阅读的重点是看看如何用Arduino的思路进行项目的制作。

如果读者之前是做纯电脑软件开发工作的，即使用C++非常熟练，那么在阅读完第1章后，可以直接跳到第4章，感受一下Arduino给纯软件开发人员带来的那种完成硬件制作的感受，然后仔细阅读第5章，看看目前都有哪些扩展板可以为自己所用，控制电机、控制液晶之类的，硬件知识哪怕我们不用，也还是要了解一些的。接下来，对于第6章，可以仔细阅读一下与硬件关系不太大的扩展库以及如何创建自己的库，在今后底层硬件库不断丰富完善的情况下，开发一些注重应用、与底层关系不是太紧密的库时，这就是我们的用武之地。第7~9章的内容会告诉我们前面的知识是如何结合起来的——用纯软件思路，同样建议按照书中的内容至少动手完成一个项目的制作，做纯软件开发工作的人开发硬件也是很容易的。

致谢

首先要感谢本书的策划张国强先生，是他对Arduino的关注促成了本书的出版，同时在笔者撰写书稿时他也对本书提出了宝贵的写作建议，并对书稿进行了仔细审阅。

其次要感谢让我了解Arduino的庄明波先生，他不但在技术上给予了我很多的指导，同时也无私地提供了大量的Arduino扩展板的资料以及实物，供我在Arduino的程序调试中使用，同时与我共同探讨技术上遇到的问题。

最后要感谢现在正捧着这本书的您，感谢您肯花费时间和精力阅读本书，由于时间有限，书中难免存在疏漏与错误，诚恳地希望您批评指正，您的意见和建议将是我巨大的财富。希望在Arduino的领域结识更多的朋友。

目 录

前言

第一篇 基础篇

第1章 初识Arduino2

- 1.1 Arduino的历史2
- 1.2 Arduino的家族3
- 1.3 Arduino的资源6
- 1.4 Arduino的开发环境9
- 1.5 添加新硬件及设置开发环境9
- 1.6 Arduino开发环境的应用14

第2章 编写Arduino程序16

- 2.1 绘制流程图16
 - 2.1.1 流程图基本符号16
 - 2.1.2 流程图的三种基本结构17
- 2.2 C语言的标识符与关键字18
 - 2.2.1 标识符18
 - 2.2.2 关键字18
 - 2.2.3 运算符19
 - 2.2.4 分隔符21
 - 2.2.5 常量21
 - 2.2.6 注释符21
- 2.3 控制语句21
 - 2.3.1 if语句21
 - 2.3.2 switch语句22
 - 2.3.3 while语句23

- 2.3.4 do-while语句24
- 2.3.5 for语句25
- 2.3.6 break语句26
- 2.3.7 continue语句26
- 2.3.8 goto语句26
- 2.4 程序结构27

第3章 Arduino的基本函数29

- 3.1 数字I/O30
 - 3.1.1 pinMode(pin,mode)30
 - 3.1.2 digitalWrite(pin,value)31
 - 3.1.3 digitalRead(pin)32
- 3.2 模拟I/O33
 - 3.2.1 analogReference(type)33
 - 3.2.2 analogRead(pin)33
 - 3.2.3 analogWrite(pin, value)34
- 3.3 高级I/O37
 - 3.3.1 shiftOut(dataPin,clockPin, bitOrder,val)37
 - 3.3.2 pulseIn(pin,state,timeout)38
- 3.4 时间函数39
 - 3.4.1 millis()39
 - 3.4.2 delay(ms)40
 - 3.4.3 delayMicroseconds(us)40
- 3.5 数学库41
 - 3.5.1 min(x,y)41
 - 3.5.2 max(x,y)41

3.5.3	abs(x)	41	4.5.1	下载器AVRISP	63
3.5.4	constrain(amt,low,high)	41	4.5.2	AVR Studio	64
3.5.5	map(x,in_min,in_max,out_min, out_max)	41	4.5.3	烧写引导程序	65
3.5.6	三角函数	42	第二篇 模块篇		
3.6	随机数	42	第5章 Arduino基本扩展模块		
3.6.1	randomSeed(seed)	42	5.1	L293 Motor Shield	68
3.6.2	random(howsmall,howbig)	42	5.1.1	直流电机的工作原理	68
3.7	位操作	43	5.1.2	H桥驱动电路	70
3.8	中断函数	43	5.1.3	线性放大调速原理	71
3.8.1	interrupts()和noInterrupts()	43	5.1.4	PWM调速原理	72
3.8.2	attachInterrupt(interrupt, function,mode)	43	5.1.5	L293 Motor Shield的原理	72
3.9	串口通信	45	5.1.6	L293 Motor Shield的应用	74
3.10	SPI接口	48	5.1.7	程序设计	75
3.10.1	SPI接口概述	48	5.1.8	程序分析	76
3.10.2	SPI接口数据传输	48	5.1.9	程序的精练	77
3.10.3	SPI类及其成员函数	49	5.2	Input Shield	78
第4章 Arduino硬件平台			5.2.1	Input Shield原理图	79
4.1	Arduino的原理图	52	5.2.2	Input Shield的实例	79
4.2	串行通信口的使用	55	5.2.3	程序设计	80
4.2.1	实例功能	56	5.2.4	程序分析	81
4.2.2	硬件电路	56	5.2.5	使用摇杆控制直流电机转速	81
4.2.3	程序设计	56	5.3	LCD Keypad Shield	83
4.3	数字I/O口的使用	58	5.3.1	液晶显示原理	83
4.3.1	实例功能	59	5.3.2	标准1602液晶模块	83
4.3.2	硬件电路	59	5.3.3	1602液晶模块控制方式	84
4.3.3	程序设计	59	5.3.4	LCD Keypad Shield原理图	87
4.4	模拟I/O口的使用	61	5.3.5	LCD Keypad Shield应用实例	89
4.4.1	实例功能	61	5.3.6	程序设计	89
4.4.2	硬件电路	61	5.3.7	程序分析	92
4.4.3	程序设计	62	5.3.8	Arduino的液晶控制方式	93
4.5	烧写引导程序	62	5.3.9	“hello Arduino!”	94
			5.4	Ethernet Shield	97

- 5.4.1 Ethernet Shield原理图97
- 5.4.2 W5100芯片介绍97
- 5.4.3 W5100芯片的寄存器101
- 5.4.4 W5100芯片的使用105
- 5.4.5 Ethernet Shield应用实例 ...105
- 5.4.6 程序设计106
- 5.5 I/O扩展板109
 - 5.5.1 Xbee传感器扩展板V5109
 - 5.5.2 伺服电机控制110
 - 5.5.3 伺服电机应用实例111
 - 5.5.4 Interface shield114
 - 5.5.5 RGB LED Module114
 - 5.5.6 RGB LED Module应用实例...118
 - 5.5.7 程序的精练123
- 第6章 Arduino的扩展库126**
 - 6.1 Arduino扩展库介绍126
 - 6.1.1 Arduino扩展库的作用126
 - 6.1.2 Arduino扩展库的应用126
 - 6.2 对象和类130
 - 6.2.1 类的定义130
 - 6.2.2 对象的创建及成员函数的调用131
 - 6.2.3 对象的初始化和构造函数 ...132
 - 6.2.4 函数的重载133
 - 6.2.5 析构函数133
 - 6.3 LiquidCrystal库134
 - 6.3.1 构造函数136
 - 6.3.2 command()和write()139
 - 6.3.3 begin()140
 - 6.3.4 clear()142
 - 6.3.5 home()142
 - 6.3.6 setCursor()142
 - 6.3.7 noDisplay()和display() ...143
 - 6.3.8 cursor()和noCursor()143
 - 6.3.9 blink()和noBlink()143
 - 6.3.10 autoscroll()和noAutoscroll()144
 - 6.3.11 scrollDisplayLeft()和scrollDisplayRight()144
 - 6.3.12 print()145
 - 6.4 Ethernet库146
 - 6.4.1 EthernetClass类定义146
 - 6.4.2 Server类定义148
 - 6.4.3 Server类构造函数148
 - 6.4.4 Server类成员函数148
 - 6.4.5 Client类定义152
 - 6.4.6 Client类构造函数152
 - 6.4.7 Client类成员函数153
 - 6.5 SoftwareSerial库158
 - 6.5.1 构造函数159
 - 6.5.2 begin()160
 - 6.5.3 read()160
 - 6.5.4 print()和println()161
 - 6.5.5 使用限制164
 - 6.6 EEPROM库165
 - 6.6.1 read()165
 - 6.6.2 write()166
 - 6.7 Wire库166
 - 6.7.1 IIC总线概述166
 - 6.7.2 TwoWire类定义167
 - 6.7.3 begin()168
 - 6.7.4 requestFrom()168
 - 6.7.5 available()169
 - 6.7.6 receive()169
 - 6.7.7 beginTransmission()170
 - 6.7.8 endTransmission()170
 - 6.7.9 send()171
 - 6.7.10 onReceive()172
 - 6.7.11 onRequest()173

6.8 Servo库	174	6.11.7 write_bit()	201
6.8.1 构造函数	175	6.11.8 read_bit()	202
6.8.2 attach()	176	6.11.9 write()	202
6.8.3 write()	177	6.11.10 read()	203
6.8.4 writeMicroseconds()	177	6.11.11 select()	203
6.8.5 read()	178	6.11.12 skip()	204
6.8.6 readMicroseconds()	178	6.12 XBee库	204
6.8.7 attached()	178	6.12.1 XBee类定义	204
6.8.8 detach()	179	6.12.2 构造函数	205
6.9 Stepper库	179	6.12.3 begin()	206
6.9.1 步进电机概述	179	6.12.4 readPacket()	206
6.9.2 步进电机的基本参数	180	6.12.5 send()	209
6.9.3 步进电机的优缺点	181	6.13 创建自己的库	210
6.9.4 步进电机的工作原理	181	6.13.1 库的功能——Morse	210
6.9.5 步进电机的控制电路	183	6.13.2 MorseCode类的定义	213
6.9.6 Stepper类定义	185	6.13.3 MorseCode类的成员函数	214
6.9.7 构造函数	186	6.13.4 MorseCode库的使用	222
6.9.8 setSpeed()	188	6.13.5 关键字的定义	223
6.9.9 step()	188	第7章 无线模块的应用	224
6.10 TLC5940库	189	7.1 APC220	224
6.10.1 Tlc5940类的定义	190	7.1.1 APC220性能指标	224
6.10.2 init()	191	7.1.2 模块引脚定义	226
6.10.3 update()	192	7.1.3 模块的使用	226
6.10.4 set()	193	7.1.4 注意事项	227
6.10.5 get()	194	7.2 DFduino wireless	228
6.10.6 setAll()	194	7.2.1 DFduino wireless性能指标	228
6.10.7 clear()	195	7.2.2 模块引脚定义	229
6.11 OneWire库	195	7.2.3 模块的使用	229
6.11.1 单总线的结构	195	7.3 Bluetooth V3	231
6.11.2 单总线控制方式	195	7.3.1 Bluetooth V3性能指标	231
6.11.3 单总线信号形式	196	7.3.2 模块引脚定义	232
6.11.4 OneWire类	198	7.3.3 模块的使用	232
6.11.5 构造函数	200	7.4 XBee和XBee PRO	234
6.11.6 reset()	200		

7.4.1 XBee及XBee PRO性能指标	235
7.4.2 模块引脚定义	235
7.4.3 模块的使用	236
7.4.4 程序设计	236

第三篇 应用篇

第8章 打造自己的遥控履带车

8.1 履带车的驱动	242
8.1.1 实现功能	242
8.1.2 所需器材	242
8.1.3 硬件连接	243
8.1.4 程序设计	246
8.1.5 MotorCar类	251
8.1.6 类的应用	255
8.2 添加感知器件	257
8.2.1 实现功能	257
8.2.2 所需器材	257
8.2.3 器材介绍	257
8.2.4 硬件连接	258
8.2.5 程序设计	258
8.3 添加无线模块	261
8.3.1 实现功能	261
8.3.2 所需器材	261
8.3.3 硬件连接	261
8.3.4 程序设计	262
8.4 制作遥控器	264
8.4.1 实现功能	264
8.4.2 所需器材	265
8.4.3 硬件连接	265
8.4.4 程序设计	265
8.5 履带车遥控调速	267
8.5.1 实现功能	267
8.5.2 程序设计	267

8.6 添加无线摄像头	272
8.6.1 实现功能	272
8.6.2 所需器材	272
8.6.3 器材介绍	272
8.6.4 硬件连接	273
8.6.5 程序设计	277
8.7 环境信息获取器件	283
8.7.1 实现功能	283
8.7.2 所需器材	284
8.7.3 器材介绍	284
8.7.4 硬件连接	285
8.7.5 程序设计	286

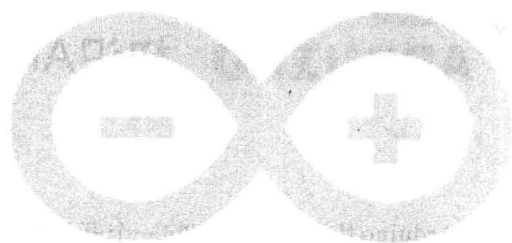
第9章 仿生机器人

9.1 遥控机械臂	295
9.1.1 实例功能	295
9.1.2 器材列表	295
9.1.3 搭建硬件环境	296
9.1.4 安装控制部分	298
9.1.5 Wii游戏手柄	298
9.1.6 机械臂程序设计	300
9.2 双足机器人	304
9.2.1 实例功能	304
9.2.2 器材列表	304
9.2.3 搭建硬件环境	305
9.2.4 双足机器人程序设计	307
9.2.5 PC调试软件编写	310
9.2.6 双足机器人的调试	317

附录A Arduino引脚与AVR单片机管脚对应关系

附录B Arduino扩展板

附录C 其他可扩展模块



ARDUINO 第一篇

基础篇

- 第1章 初识Arduino
- 第2章 编写Arduino程序
- 第3章 Arduino的基本函数
- 第4章 Arduino硬件平台



第1章 初识Arduino

Arduino是源自意大利的一个开放源代码的硬件项目平台，该平台包括一块具备简单I/O功能的电路板以及一套程序开发环境软件。Arduino可以用来开发交互产品，比如它可以读取大量的开关和传感器信号，并且可以控制电灯、电机和其他各式各样的物理设备；Arduino也可以开发出与PC相连的周边装置，能在运行时与PC上的软件进行通信。Arduino的硬件电路板可以自行焊接组装，也可以购买已经组装好的模块，而程序开发环境的软件则可以从网上免费下载与使用。

1.1 Arduino的历史

说到Arduino的起源似乎有点令人感觉无心插柳柳成荫。Massimo Banzi是意大利米兰互动设计学院的教师，他的学生常常抱怨不能找到一块价格便宜且功能强大的控制主板来设计他们的机器人。2005年的冬天，Banzi和David Cuartielles讨论到这个问题，David Cuartielles是西班牙的微处理器设计工程师，当时在这所学校做访问研究。他们决定自己设计一块控制主板。他们找来了Banzi的学生David Mellis，让他来编写代码程序。David Mellis只花了两天时间就完成了代码的编写，然后又过了3天，板子就设计出来了，取名为Arduino。很快，这块板子受到了广大学生的欢迎。这些学生当中那些甚至完全不懂计算机编程的人，都用Arduino做出了“很炫”的东西：有人用它控制和处理传感器，有人用它控制灯闪烁，有人用它制作机器人……之后Banzi、Cuartielles和Mellis将设计图上传到网上，然后花了3000欧元加工出第一批板子。

Banzi等人当时加工了200块板子，卖给学校50块，起初还担心剩下的150块怎么卖出去，但是几个月后，他们的设计作品在网上得到了快速传播，接着他们收到了几个上百块板子的订单。这时他们明白Arduino是很有市场价值的，所以，他们决定开始Arduino的事业，但是有个原则——开源。他们规定任何人都可以复制、重设计甚至出售Arduino板子。人们不用花钱购买版权，连申请许可权都不用。但是，如果你加工出售Arduino原板，版权还是归Arduino团队所有。如果你是在基于Arduino的设计上修改，你的设计必须也和Arduino一样开源。

Arduino设计者们唯一所有的就是“Arduino”这个商标。如果你的设计也想用Arduino命

名,那么你就得支付费用。这样做是为了保护“Arduino”这个商标不被低劣的作品损坏。

对于最初决定硬件开源,几位设计者也有不同的动机。Cuartielles认为自己是个“左倾学术主义者”,不喜欢因为赚钱而限制大家的创造力,从而导致自己的作品得不到广泛使用。“如果有人要复制它,没问题。复制只会让它更出名。”Cuartielles在某次演讲中甚至说:“请你们复制它吧!”Banzi则恰恰相反,他更像一个精明的商人。他现在已经退休了,不再教书,开了一家科技设计公司。他猜想,如果Arduino开源,相比那些不开源的作品,会激发更多人的兴趣,从而得到更广泛的使用。还有一点就是,一些电子疯狂爱好者会去寻找Arduino的设计缺陷,然后要求Arduino团队做出改进。利用这种免费的劳动力,他们可以开发出更好的新产品。

实际情况也正如他所料,在接下来的几个月内,很多人提出重新布线、改进编程语言等建议。后来曾有销售商要求代理Arduino产品。2006年,Arduino方案获得了Prix Art Electronica电子通信类方面的荣誉奖。那一年,他们销售了5000块板子。第二年,他们销售了30 000块。Arduino被电子疯狂爱好者用来设计机器人、调试汽车引擎、制作无人飞机模型等。

1.2 Arduino的家族

Arduino设计之初的目的是希望让设计师和艺术家们能够很快地通过它学习电子和传感器的基础知识,并应用到他们的设计当中。设计中所要表现的想法和创意才是最主要的,至于单片机如何工作,硬件的电路是如何构成的,设计师和艺术家们并不需要考虑。

Arduino的出现,大大降低了互动设计的门槛,没有学过电子知识的人也能够使用它制作出各种充满创意的作品。越来越多的艺术家、设计师开始使用Arduino制作交互艺术品。为了针对不同的应用领域,目前Arduino已设计出很多不同的型号以满足不同使用者的需要,在这里简单介绍一下几类主要产品,详细信息可登录Arduino的主页<http://www.arduino.cc>查阅。

1. Arduino Duemilanove

这是一款基本的Arduino产品,控制器采用ATmega168或ATmega328,支持直流电源供电和USB端口供电,如图1.1所示。后续的很多产品都是在这款产品的基础上发展起来的。

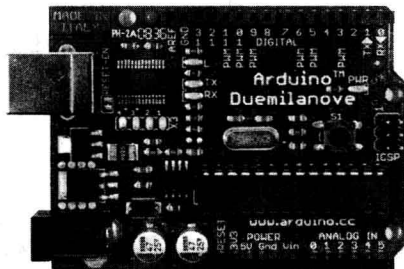


图1.1 Arduino Duemilanove

2. Arduino Nano

Arduino Nano在设计中去掉了直流电源接口，采用了Mini-B标准的USB接口来连接电脑，除了外观变了，其他接口及功能保持不变，控制器同样采用ATmega168或ATmega328，是一款缩小版的Arduino Duemilanove，如图1.2所示。

3. Arduino mini

考虑到存在一些对空间要求十分严格的使用者，Arduino mini（见图1.3）在设计时甚至去掉了USB接口和复位开关，这样能减小Arduino的尺寸。唯一的问题是连接电脑或烧写程序时需要一个USB或RS232转换成TTL的适配座，Arduino官方也有相应的适配座——Mini USB Adapter (<http://www.arduino.cc/en/Main/MiniUSB>上有相关的资料)。

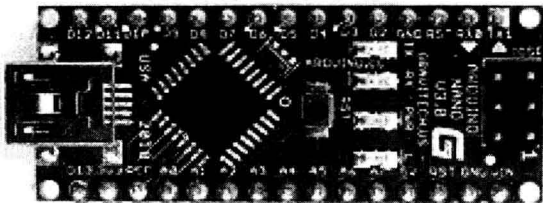


图1.2 Arduino Nano

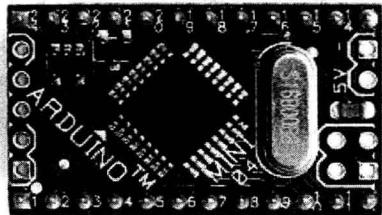


图1.3 Arduino mini

4. Arduino BT

Arduino BT（见图1.4）本身包含了一个Bluegiga WT11蓝牙模块，支持蓝牙无线串行通信，但不支持蓝牙音频设备。若没有USB接口，连接电脑或烧写程序可以通过蓝牙适配器与Arduino BT连接实现无线程序下载与控制。

5. LilyPad Arduino

这是一款真正有艺术气质的产品，面向的主要使用者是从事服装设计之类工作的设计师，它可以使用导电线或普通线缝在衣服或布料上，LilyPad Arduino每个引脚上的小洞大到足够缝纫针轻松穿过，如图1.5所示。如果用导电线缝纫的话，既可以起到固定的作用，又可以起到传导的作用。比起普通的Arduino板，LilyPad Arduino相对比较脆弱，比较容易损坏，但它的功能基本都保留了下来，除了一点，即它没有USB接口，所以LilyPad Arduino连接电脑或烧写程序时同Arduino mini一样需要一个USB或RS232转换成TTL的适配座。

6. Arduino Pro和Arduino Pro Mini

设计Arduino Pro的目的是为了那些需要便利性和低成本的高级用户。为了降低成本，它省去了USB接口、直流电源接口和引脚排针，连接电脑或烧写程序时需要一个USB或RS232转换成TTL的适配座。Arduino Pro更像是一个大号的Arduino mini，如图1.6所示。需要注意的是，Arduino Pro有3.3V/8MHz和5V/16MHz两个版本，使用的时候要留心点。另外Arduino Pro同样有一个Arduino Pro Mini的版本，如图1.7所示。

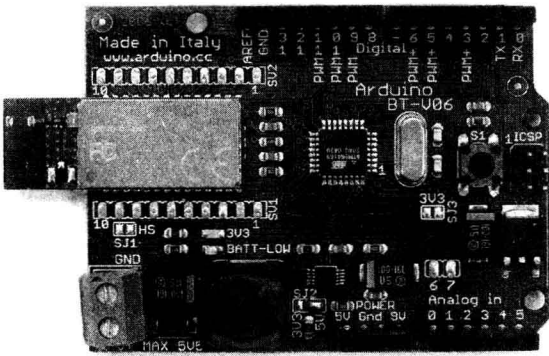


图1.4 Arduino BT

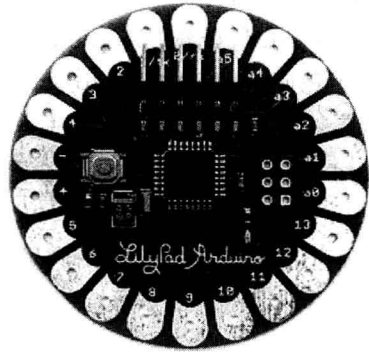


图1.5 LilyPad Arduino

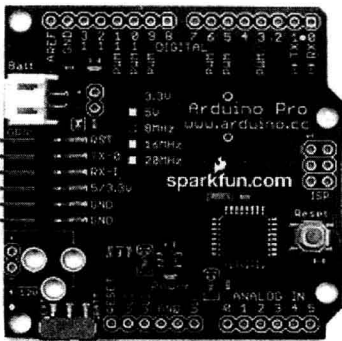


图1.6 Arduino Pro

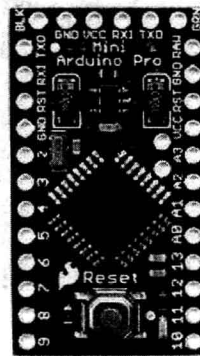


图1.7 Arduino Pro Mini

7. Arduino Fio

Arduino Fio (见图1.8)的工作电压是3.3V,控制器的工作频率是8MHz,采用了Mini-B标准的USB接口,提供一个锂聚合物电池接口,底部预留了一个XBee模块插座(美国DIGI的zigbee模块,本书的第7章有XBee模块的相关介绍,也可登录<http://www.digi.com.cn>了解XBee模块的更多信息),XBee模块可使Arduino方便地应用于无线网络。

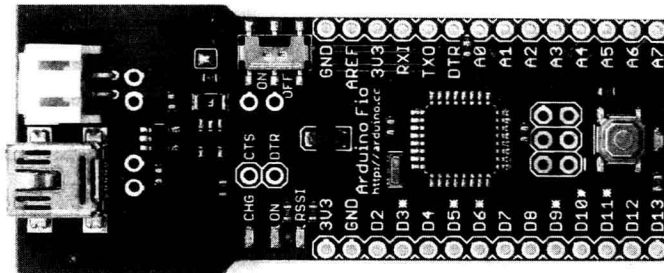


图1.8 Arduino Fio

8. Arduino Uno

Arduino Uno是最新的Arduino产品系列，如图1.9所示，它与之前的Arduino板最大的不同在于它不是使用FTDI USB-to-serial串行驱动器芯片，而是采用Atmega8U2芯片进行USB到串行数据的转换。目前Arduino Uno已成为Arduino主推的产品。

9. Arduino Mega2560

Arduino Mega2560（见图1.10）的控制器采用的是ATMega2560，它的资源要比之前的Arduino产品丰富很多，用于满足需使用较多资源进行产品设计与开发的用户需求，具体资源会在下一节描述。同时，Arduino Mega2560也兼容之前基于Arduino Duemilanove的设计。



图1.9 Arduino Uno

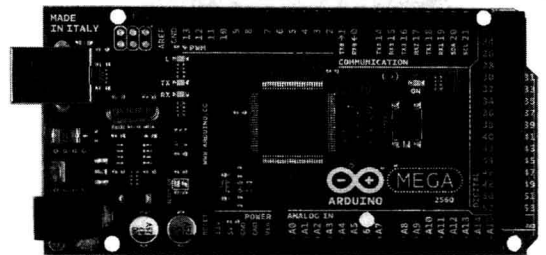


图1.10 Arduino Mega2560

1.3 Arduino的资源

Arduino的硬件电路设计以创作公用约定（creative commons）的形式提供授权。相应的原理图和电路图都可以从Arduino网站上免费获得。Arduino Duemilanove具有14个数字I/O口（其中6个可提供PWM输出），6个模拟I/O口，一个复位开关，一个ICSP下载口，支持USB接口，可通过USB接口供电，也可以使用单独的7~12V电源供电。Arduino的资源在板子上已经明确标注，使用者可以很方便地了解具体的资源分配，DIGITAL一边有14个数字I/O口0~13，ANALOG IN一边有6个模拟I/O口0~5，其他还有POWER、TX、RX、PWM等标识，如图1.11所示。

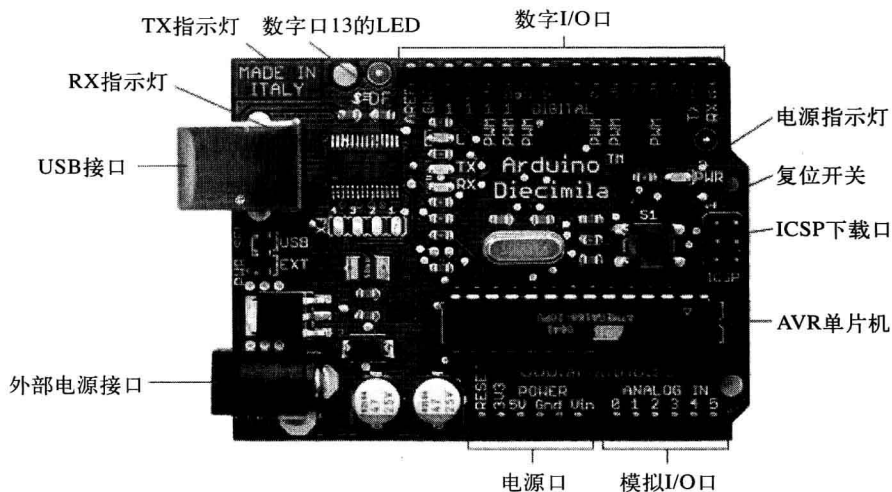


图1.11 Arduino的资源

Arduino Duemilanove总体参数如表1.1所示。

表1.1 Arduino总体参数

名 称	参 数
微控制器	ATmega168/ATmega328
操作电压	5V
推荐输入电压	7~12V
极限输入电压	6~20V
数字I/O脚数	14, 其中6个提供PWM输出
模拟输入脚数	6
I/O脚直流电流	40 mA
3.3伏脚的电流	50 mA
闪存	16 KB (ATmega168)或32 KB (ATmega328), 其中2KB用于引导程序
SRAM	1 KB (ATmega168)或2 KB (ATmega328)
EEPROM	512 byte (ATmega168)或1 KB (ATmega328)
时钟频率	16 MHz
尺寸	6.0cm × 5.33cm

各引脚定义如下：

- 数字引脚：0~13
- 模拟引脚：A0~A5（为区分数字引脚，在引脚号前加A）
- 串行通信：0, 1（0作为RX，接收数据；1作为TX，发送数据）
- 外部中断：2, 3
- PWM输出：3, 5, 6, 9, 10, 11
- SPI通信：10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK)