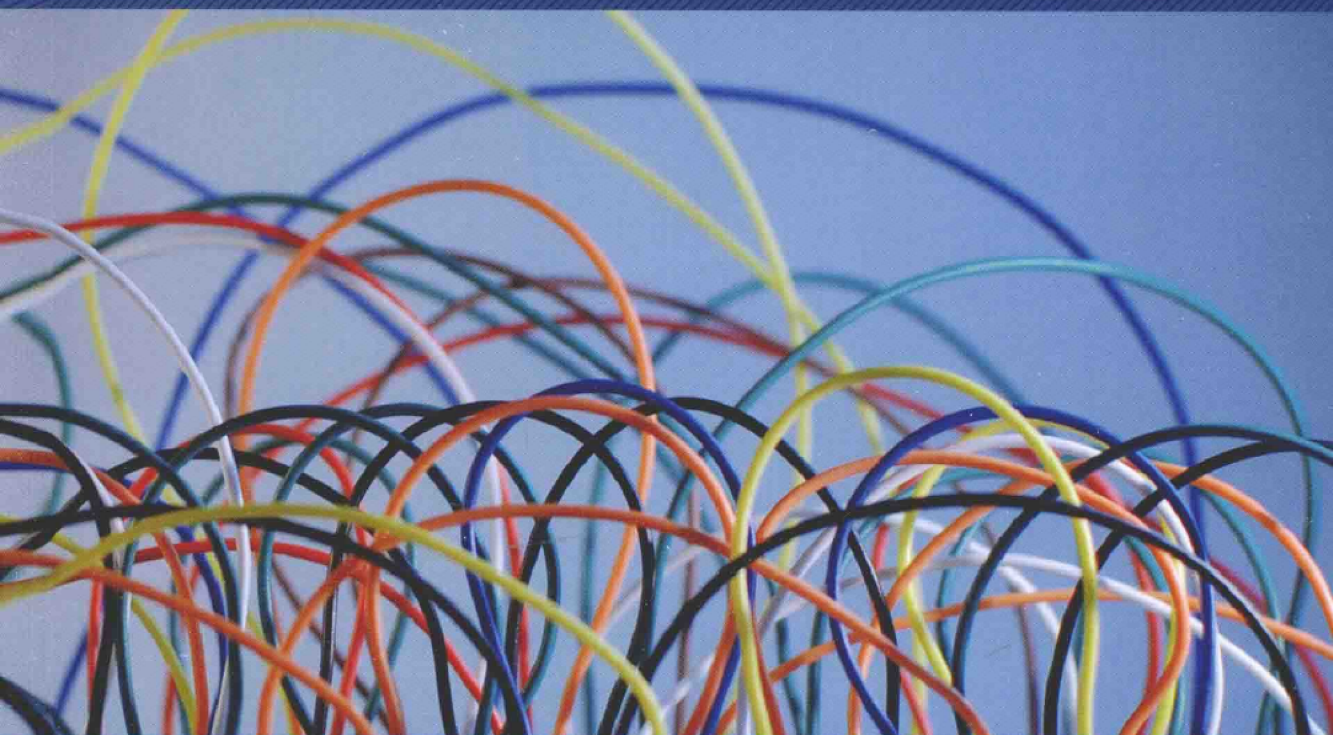




Arduino 程序设计基础

陈吕洲 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

Arduino 程序设计基础

陈吕洲 编著

北京航空航天大学出版社

内 容 简 介

Arduino 不仅仅是全球最流行的开源硬件,也是一个优秀的硬件开发平台,更是硬件开发的趋势。Arduino 简单的开发方式使得开发者更关注于创意与实现,更快地完成自己的项目开发,大大节约学习的成本,缩短开发的周期。越来越多的专业硬件开发者已经或开始使用 Arduino 来开发他们的项目和产品。

本书涵盖 Arduino 基础知识及高级应用,中途穿插简单项目制作,用于巩固知识与扩展提高,同时提供常用的 API 参考,以便读者实践时查阅。

本书主要针对本科生及研究生阶段的 Arduino 教学实验进行编写,亦适用于相关开发人员及入门者学习。

图书在版编目(CIP)数据

Arduino 程序设计基础 / 陈吕洲编著. -- 北京:北京航空航天大学出版社, 2014.1
ISBN 978-7-5124-1339-9

I. ①A… II. ①陈… III. ①单片微型计算机—程序设计 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2013)第 303889 号

版权所有,侵权必究。

Arduino 程序设计基础

陈吕洲 编著

责任编辑 宋淑娟

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:18.75 字数:400 千字

2014 年 1 月第 1 版 2014 年 1 月第 1 次印刷 印数:3 000 册

ISBN 978-7-5124-1339-9 定价:39.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

前言

Arduino 不仅仅是全球最流行的开源硬件,也是一个优秀的硬件开发平台,更是硬件开发的趋势。Arduino 简单的开发方式使得开发者更关注于创意与实现,更快地完成自己的项目开发,大大节约学习的成本,缩短开发的周期。

因为 Arduino 的种种优势,使得越来越多的专业硬件开发者已经或开始使用 Arduino 来开发他们的项目和产品;越来越多的软件开发者使用 Arduino 进入硬件、物联网等开发领域;在大学里,自动化、软件专业,甚至艺术专业,也纷纷开设了 Arduino 相关课程。

笔者 2008 年开始接触 Arduino 时即被 Arduino 的简单易用所吸引;后来创建了 ArduinoCN 中文社区,致力于 Arduino 的教学与推广;目前从事硬件开发工作,在工作中经常使用 Arduino 进行开发,积累了一定的经验。

本书是笔者将过去撰写的 Arduino 相关教程与自身开发经验相结合整理而成,主要针对大学的 Arduino 教学实验进行编写,亦适用于相关开发人员及入门者学习。

本书体系结构清晰,内容丰富,涵盖 Arduino 基础知识及高级应用,中途穿插简单项目制作,用于巩固知识与扩展提高,同时提供常用的 API 参考,以便读者实践时查阅。

各章节所涉及内容如下:

第 1 章简单介绍 Arduino 的历史、软硬件及安装使用方法。

第 2 章讲解语言基础、基本输入/输出方法、串口通信和一些常用函数的使用。

第 3 章讲解 Arduino I/O 口的一些高级应用。

第 4 章详细讲解如何使用和编写 Arduino 类库。

第 5 章介绍串口、IIC、SPI 三种通信方式在 Arduino 上的使用方法。

第 6 章介绍如何使用 EEPROM 和 SD 卡记录和保存数据。

第 7 章讲解 Arduino 红外通信的使用方法。

第 8 章以 1602 LCD 和 12864 LCD 两种常见液晶显示器为例,介绍 Arduino 驱动液晶显示器的方法。

第 9 章针对特殊型号的 Arduino 控制器(Leonardo、Micro、Due 等)的 USB 功能进行讲解。

第 10 章讲解使用 Arduino 进行网络通信的方法。

前 言

附录介绍使用 Visual Studio 开发 Arduino 的方法及一些常见问题的处理方式，并附带一些控制器手册和相关配置表格，以便读者查阅。

掌握了前三章，便基本了解了 Arduino 的开发方法，从而可以完成很多小型项目的开发。第 4 章以后为进阶教程，讲解 Arduino 的一些常用类库的使用。如果用于大学教学，则按章节循序渐进即可，教师亦可以设计一些小型项目供学生实践练习；如果已有一定的软硬件开发基础，则可根据实际项目要求，直接阅读相应章节。

感谢以下网友对笔者写作本书提供的帮助：吴琼（大茶园丁）、杨奇伟（kiwi）、朱泽州（ZZZ）、沈金鑫（奔跑）、王翔（玳羽）、李海波（海神）、刘定杨（三水）。特别感谢潘拥军先生的悉心检查与耐心指导。

由于笔者水平有限，书中难免存在不足与错误之处，敬请读者批评指正。可以通过 ArduinoCN 中文社区(<http://www.arduino.cn>)参与本书相关内容的讨论，亦可直接与我联系，我的新浪微博是：<http://weibo.com/coloz>。

陈吕洲
2013 年 8 月

目 录

第 1 章 初识 Arduino	1
1.1 什么是 Arduino	1
1.2 Arduino 的由来	4
1.3 为什么使用 Arduino 作为开发平台	5
1.4 Arduino 硬件——选择一款适合自己的 Arduino 控制器	6
1.4.1 认识不同型号的 Arduino 控制器	6
1.4.2 众多的 Arduino 外围模块	10
1.4.3 从 Arduino UNO 开始	11
1.5 Arduino 软件	13
1.5.1 下载配置 Arduino 开发环境	13
1.5.2 认识 Arduino IDE	14
1.5.3 安装 Arduino 驱动程序	16
1.6 Blink——Arduino 的 Hello World!	19
第 2 章 基础篇	24
2.1 Arduino 语言及程序结构	24
2.1.1 Arduino 语言	24
2.1.2 Arduino 程序结构	25
2.2 C/C++ 语言基础	25
2.2.1 数据类型	26
2.2.2 运算符	27
2.2.3 表达式	28
2.2.4 数 组	28
2.2.5 字符串	29
2.2.6 注 释	29

目 录

2.2.7	用流程图表示程序	30
2.2.8	顺序结构	30
2.2.9	选择结构	30
2.2.10	循环结构	33
2.3	电子元件和 Arduino 扩展模块	36
2.4	传感器扩展板的使用	38
2.5	Arduino I/O 的简单控制	39
2.5.1	数字 I/O 的使用	39
2.5.2	项目:人体感应灯(继电器模块与数字传感器)	50
2.5.3	模拟 I/O 的使用	53
2.5.4	项目:电子温度计	62
2.5.5	数字传感器与模拟传感器的使用	64
2.6	与计算机交流——串口的使用	66
2.6.1	串口输出	66
2.6.2	串口输入	68
2.6.3	实验:串口控制开关灯	70
2.7	时间控制函数	71
2.7.1	运行时间函数	71
2.7.2	延时函数	72
第 3 章	I/O 口高级应用	73
3.1	调声函数	73
3.2	项目:简易电子琴	77
3.3	脉冲宽度测量函数及超声波测距	79
3.3.1	脉冲宽度测量函数	79
3.3.2	超声波测距	79
3.4	设置 ADC 参考电压	83
3.5	外部中断	83
3.5.1	外部中断的使用	84
3.5.2	实验:外部中断触发蜂鸣器报警	86
第 4 章	使用和编写类库	88
4.1	编写并使用函数,提高程序的可读性	88
4.2	使用 Arduino 类库	90
4.3	编写 Arduino 类库	94
4.3.1	编写头文件	94

4.3.2	预处理命令	95
4.3.3	宏定义	96
4.3.4	文件包含	96
4.3.5	条件编译	96
4.3.6	版本兼容	97
4.3.7	编写.cpp文件	97
4.3.8	关键字高亮显示	98
4.3.9	建立示例程序	99
4.4	类库优化与发布	100
第5章	通信篇	101
5.1	硬件串口通信——HardwareSerial类库的使用	101
5.1.1	HardwareSerial类库成员函数	103
5.1.2	print()和write()输出方式的差异	108
5.1.3	read()和peek()输入方式的差异	109
5.1.4	串口读取字符串	111
5.1.5	串口事件	112
5.1.6	实验:串口控制RGB LED调光	114
5.1.7	更好的串口监视器——串口调试助手	117
5.2	软件模拟串口通信——SoftwareSerial类库的使用	117
5.2.1	SoftwareSerial类库的局限性	118
5.2.2	SoftwareSerial类库成员函数	118
5.2.3	建立一个软串口通信	119
5.2.4	实验:Arduino间的串口通信	119
5.2.5	同时使用多个软串口	123
5.3	IIC总线的使用——Wire类库的使用	124
5.3.1	IIC主机、从机与引脚	124
5.3.2	Wire类库成员函数	125
5.3.3	IIC连接方法	127
5.3.4	主机写数据,从机接收数据	128
5.3.5	从机发送数据,主机读取数据	131
5.4	SPI总线的使用——SPI类库的使用	133
5.4.1	SPI引脚	134
5.4.2	SPI总线上的从设备选择	135
5.4.3	SPI类库成员函数	135
5.4.4	SPI总线上的数据发送与接收	137

目 录

5.4.5	实验:使用数字电位器 AD5206	137
5.4.6	软件模拟 SPI 通信	141
5.4.7	实验:使用 74HC595 扩展 I/O 口	142
第 6 章 存储篇		146
6.1	断电也能保存数据——EEPROM 类库的使用	146
6.1.1	EEPROM 类库成员函数	146
6.1.2	写入操作	147
6.1.3	读取操作	148
6.1.4	清除操作	149
6.1.5	使用共用体存储各类型数据到 EEPROM	150
6.2	保存大量数据——SD 卡类库的使用	152
6.2.1	格式化 SD 卡	152
6.2.2	SD 卡类库成员函数	152
6.2.3	使用 SD 卡读/写模块	157
6.2.4	创建文件	158
6.2.5	删除文件	160
6.2.6	写文件	162
6.2.7	读文件	164
6.3	项目:SD 卡环境数据记录器	165
6.3.1	DHT11 温湿度检测模块的使用	166
6.3.2	硬件连接方法	166
6.3.3	温湿度记录器	169
6.3.4	在 Excel 中分析数据	171
第 7 章 无线通信篇——红外遥控		175
7.1	IRremote 类库成员函数	176
7.2	红外接收	178
7.3	红外发射	181
7.4	实验:遥控家电设备	182
第 8 章 LCD 显示篇		187
8.1	1602 LCD 的使用——LiquidCrystal 类库的使用	187
8.1.1	将 1602 LCD 与 Arduino 连接	187
8.1.2	4 位数据线接法	188
8.1.3	1602 LCD 相关参数	189

目 录

8.1.4	LiquidCrystal 类库成员函数	189
8.1.5	实验:“hello,world!”	194
8.1.6	实验:将串口输入数据显示到 1602 LCD 上	195
8.1.7	实验:显示滚动效果	197
8.1.8	实验:显示自定义字符	198
8.2	项目:制作电子时钟	201
8.2.1	DS1307 时钟模块的使用	202
8.2.2	电子时钟	207
8.3	图形显示器的使用——u8glib 类库的使用	209
8.3.1	选择 LCD 并连接到 Arduino	209
8.3.2	连接到 Arduino 并建立 u8g 对象	210
8.3.3	u8glib 程序结构	210
8.3.4	纯文本显示	211
8.3.5	数据显示	212
8.3.6	实验:绘制图形	214
8.3.7	实验:显示图片——位图取模	215
第 9 章	USB 类库的使用	218
9.1	USB 设备模拟相关函数	218
9.1.1	USB 鼠标类的成员函数	218
9.1.2	USB 键盘类的成员函数	220
9.2	模拟键盘输入信息	222
9.2.1	Arduino Leonardo 在模拟 USB 设备后,无法正常下载程序	224
9.2.2	模拟键盘组合按键	224
9.3	实验:使用摇杆模块控制计算机鼠标	227
9.4	项目:PPT 播放遥控器	230
第 10 章	Ethernet 类库的使用	233
10.1	Ethernet 相关硬件介绍	233
10.2	Ethernet 类库	237
10.2.1	Ethernet 类	237
10.2.2	IPAddress 类	238
10.2.3	EthernetServer 类	238
10.2.4	EthernetClient 类	240
10.3	Ethernet 的初始化	243
10.3.1	自定义 IP 地址	243

目 录

10.3.2 DHCP 获取 IP 地址	243
10.4 通过 Telnet 建立简易聊天室	244
10.4.1 建立 Arduino Telnet 聊天服务器	245
10.4.2 Windows Telnet 客户端的使用	246
10.5 Ethernet 与 Web 应用	248
10.5.1 HTTP 协议简介	248
10.5.2 HTTP 响应	250
10.5.3 网页客户端	251
10.5.4 网页服务器	253
10.6 UDP 通信	256
10.6.1 EthernetUDP 类	256
10.6.2 使用 UDP 收/发数据	258
10.7 项目:网页控制 Arduino	260
附 录	266
A.1 使用专业 IDE 编写 Arduino 项目	266
A.2 常见问题及解决方法	268
A.3 Arduino Leonardo 手册	272
A.4 Arduino Due 手册	276
A.5 ASCII 码对照表	281
A.6 串口通信可用的 config 配置	284
A.7 USB 键盘库支持的键盘功能按键列表	285
参考文献	287

第 1 章

初识 Arduino

Arduino 自 2005 年推出以来便广受好评,如今已成为最热门的开源硬件之一。对于没有接触过 Arduino 的朋友来说,可能对其还有很多疑问,本章就将为大家一一解答。

1.1 什么是 Arduino

在回答什么是 Arduino 之前,先来看看几个基于 Arduino 开发的项目。

1. ArduPilot

ArduPilot (图 1-1) 是基于 Arduino 开发的无人机控制系统,是目前最强大的基于惯性导航的开源飞行控制器之一,集成了陀螺仪、加速度传感器、电子罗盘传感器、大气压传感器和 GPS 等部件。图 1-2 为使用 ArduPilot 制作的四轴飞行器。

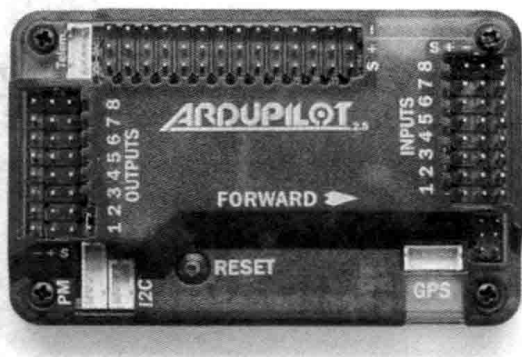


图 1-1 ArduPilot 控制器

2. MakerBot

MakerBot(图 1-3)是一款使用 Arduino Mega 作为主控制器的 3D 打印机。Arduino 负责解读 G 代码,并驱动步进电机和打印喷头等部件打印出 3D 物体。

3. ArduSat

ArduSat 是美国加州的 NanoSatisfi 团队在众筹网站 Kickstarter 上推出的采用 Arduino 制作的微型人造卫星项目(图 1-4),目的是让任何人都可以用更低的成本从事有关宇宙的研究。

该卫星只有 10 立方厘米大小,能以 18 倍声速的速度围绕地球飞行,并配备照相机和多达 25 种的感应器,搭载的感应器包括电磁波测定装置、分光器、振动传感器、光传感器、GPS、盖革计数器、陀螺仪、磁场传感器、二氧化碳检测传感器,等等。

4. 智能灯控系统

图 1-5 为通过 ZigBee、GPRS、Arduino 等多种技术集合制作的无线灯控系统。

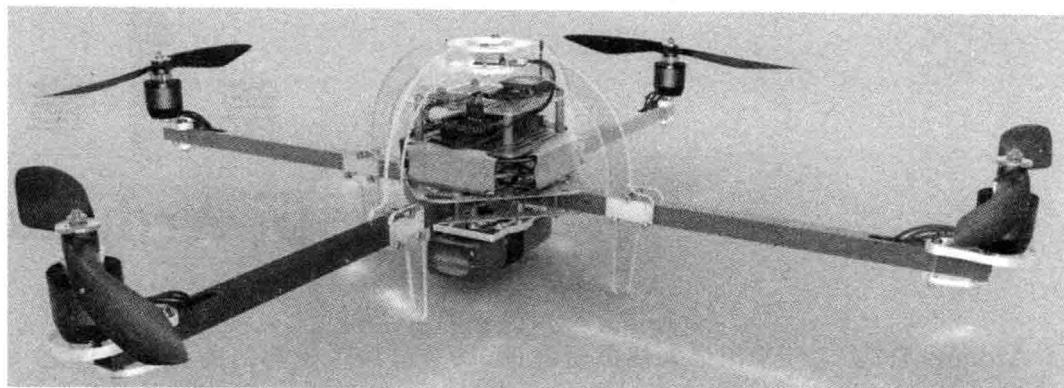


图 1-2 使用 ArduPilot 制作的四轴飞行器

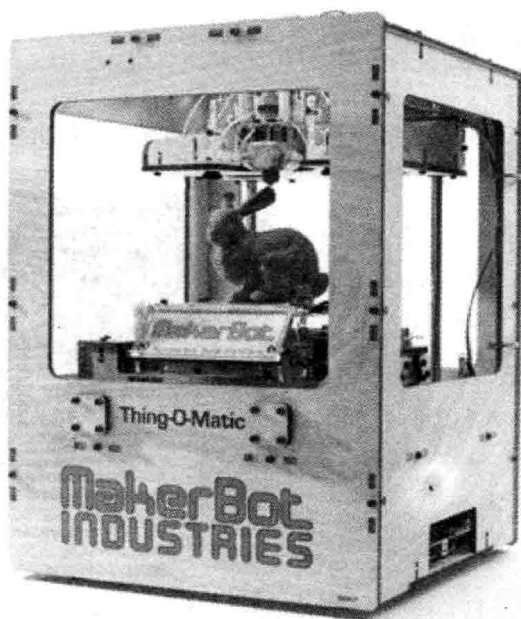


图 1-3 MakerBot 3D 打印机

如图 1-5 所示为一个路灯控制节点,它由电源模块、互感器、继电器、电能计量芯片、ZigBee 无线模块和 AVR 单片机组成。其中 AVR 单片机中的程序便是使用 Arduino 库写成的。

Arduino 并不仅仅是一块小小的电路板,还是一个开放的电子开发平台。它既包含了硬件——电路板,也包含了软件——开发环境和许许多多开发者、使用者创造的代码、程序。

Arduino 抛开了传统硬件开发的复杂操作,不需要了解硬件的内部结构和寄存器设置,也不需要过多的电子知识和编程知识,而只需通过简单的学习,了解各个引脚和函数的作用,便可利用它开发出各种出色的项目。

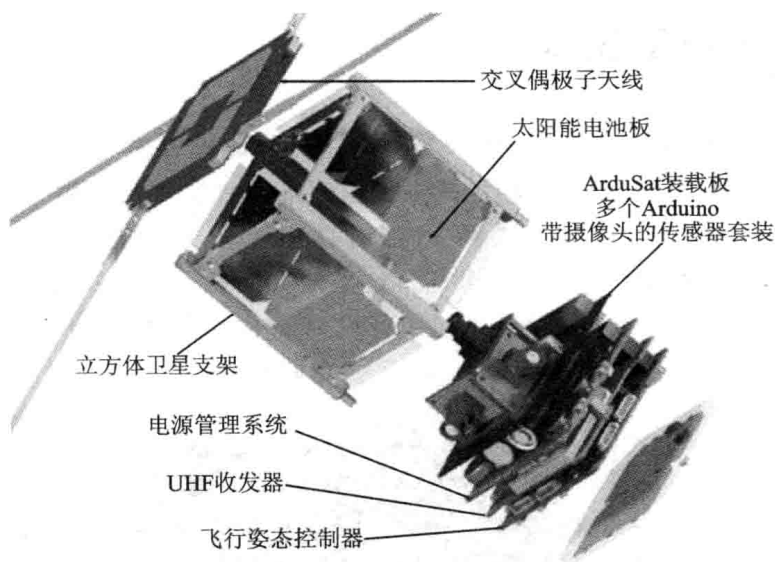


图 1-4 ArduSat 卫星结构示意图

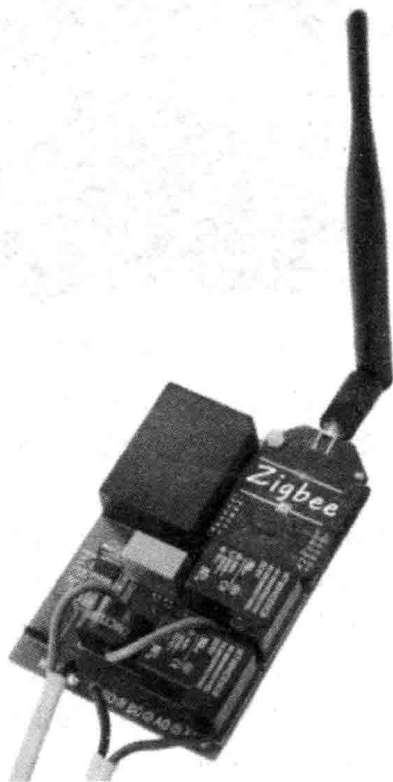


图 1-5 无线路灯控制终端

通过将 Arduino 与多种软件结合(如 Flash、Max/Msp、Vvvv、Processing),还可以制作出有趣的互动作品。

第1章 初识 Arduino

1.2 Arduino 的由来

Arduino 创始团队(图 1-6)中的 Massimo Banzi 之前是意大利 Ivrea 一家高科技设计学校的老师。他的学生们经常抱怨找不到既便宜又好用的微控制器。David Cuartielles 是一个西班牙籍芯片工程师,当时在这所学校做访问学者。2005 年冬天,Massimo Banzi 跟 David Cuartielles 讨论了这个问题。两人决定设计自己的电路板,并吸收了 Banzi 的学生 David Mellis 为电路板设计编程语言。两天以后,David Mellis 就写出了程序代码。又过了三天,电路板就完工了。



图 1-6 Arduino 创始团队

据说 Massimo Banzi 喜欢去一家名叫 di Re Arduino 的酒吧,该酒吧是以 1 000 年前意大利国王 Arduin 的名字命名的。为了纪念这个地方,他将这块电路板命名为 Arduino。

几乎任何人,即使不懂电脑编程也能用 Arduino 做出很酷的东西,比如对传感器作出回应,闪烁灯光,还能控制马达。随后 Banzi、Cuartielles 和 Mellis 把设计图放到了网上。版权法可以监管开源软件,却很难用在硬件上,为了保持设计的开放源码理念,他们决定采用 Creative Commons(CC)的授权方式公开硬件设计图。在这样的授权下,任何人都可以生产电路板的复制品,甚至还能重新设计和销售原设计的复制品。人们不需要支付任何费用,甚至不用取得 Arduino 团队的许可。然而,如果重新发布了引用设计,就必须声明原始 Arduino 团队的贡献。如果修改了电路板,则最新设计必须使用相同或类似的 Creative Commons(CC)的授权方式,以保证新版本的 Arduino 电路板也会一样是自由和开放的。唯一被保留的只有 Arduino 这个名字,它被注册成了商标,在没有官方授权的情况下不能使用它。短短几年时间,Arduino

在全球积累了大量用户,推动了开源硬件、创客运动,甚至是硬件创业领域的发展。越来越多的芯片厂商和开发公司宣布自己的硬件支持 Arduino。

在 Arduino 的推动下诞生了许多优秀的开源硬件项目,有趣的是,Arduino 本身也是多个开源项目融合的成果。图 1-7 为 Arduino 使用的开源项目。

Arduino 编译器使用的是 GCC,这是 GNU 开源计划的核心,是使用最为广泛的编译器之一。Arduino 语言衍生自 Wiring 语言,是一个开源的单片机编程架构,同时 Arduino 语言又是基于 AVR-Libc 这个 AVR 单片机扩展库编写的,AVR-Libc 也是一个优秀的开源项目。Arduino 集成开发环境是基于 Processing 的,Processing 是一个为设计师设计的新型语言,当然这这也是一个开源项目。Processing 开发环境是用 JAVA 编写的,

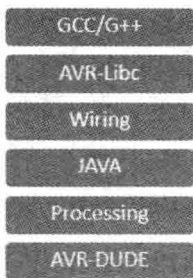


图 1-7 组成 Arduino 的开源项目

的,JAVA 是众所周知的开源项目。要想将编译好的 Arduino 程序下载到 Arduino 控制器中,还需要用到 AVR-DUDE,这也是一个开源项目的成果。

可以说,没有这些开源项目,就没有今天的 Arduino。

1.3 为什么使用 Arduino 作为开发平台

用 Arduino 创作或者进行产品开发的优势是很明显的。

1. 跨平台

Arduino IDE 可以在 Windows、Mac OS X 和 Linux 三大主流操作系统上运行,而其他的大多数控制器只能在 Windows 上开发。

2. 简单清晰的开发

Arduino IDE 基于 Processing IDE 开发,这对于初学者来说极易掌握,同时又有着足够的灵活性。Arduino 语言是基于 Wiring 语言开发的,是对 AVR-GCC 库的二次封装,并不需要太多的单片机基础和编程基础,只要简单地学习后就可以快速地进行开发。

3. 开放性

Arduino 的硬件原理图、电路图、IDE 软件及核心库文件都是开源的,在开源协议范围内可以任意修改原始设计及相应代码。

4. 社区与第三方支持

Arduino 有着众多的开发者和用户,因此可以找到他们提供的众多开源的示例代码和硬件设计。例如,可以在 Github.com、Arduino.cc、Openjumper.com 等网站上找到 Arduino 的第三方硬件、外设和类库等支持,以便更快、更简单地扩展自己的

第1章 初识 Arduino

Arduino 项目。

5. 硬件开发趋势

Arduino 不仅仅是全球最流行的开源硬件,也是一个优秀的硬件开发平台,更是硬件开发的趋势。Arduino 简单的开发方式使得开发者更关注于创意与实现,可以更快地完成自己的项目开发,大大节约学习的成本,缩短开发的周期。

鉴于 Arduino 的种种优势,越来越多的专业硬件开发者已经或开始使用 Arduino 来开发项目和产品;越来越多的软件开发人员使用 Arduino 进入硬件、物联网等开发领域;在大学里,自动化、软件专业,甚至艺术专业,也纷纷开设了 Arduino 相关课程。

1.4 Arduino 硬件——选择一款适合自己的 Arduino 控制器

Arduino 发展至今,已经有了多种型号及众多衍生控制器推出。在此列举出一些使用广泛且最有特点的 Arduino 控制器,为大家做一简单介绍。在使用 Arduino 控制器完成项目制作之前,应该对各个型号有一定的了解,以便选择适合自己项目的控制器。

1.4.1 认识不同型号的 Arduino 控制器

1. Arduino UNO

Arduino UNO(图 1-8)是目前使用最广泛的 Arduino 控制器,具有 Arduino 的所有功能,是初学者的最佳选择。本书大部分章节将用 Arduino UNO 进行教学演示。在掌握了 UNO 的开发技巧以后,就可以将自己的代码轻松地移植到其他型号的控制器上。

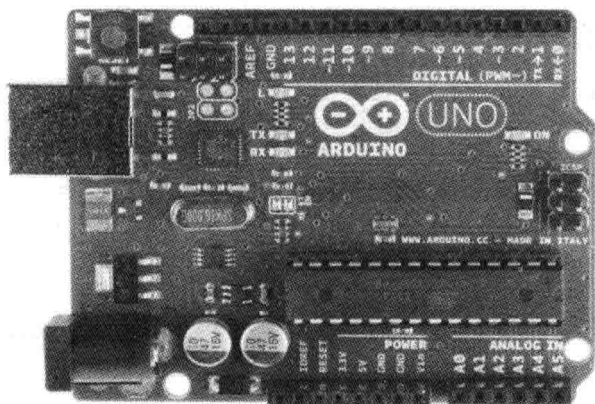


图 1-8 Arduino UNO