



# 轻松玩转 Arduino编程



小左实验室 编著



机械工业出版社  
CHINA MACHINE PRESS





# 轻松玩转 Arduino 编程

小左实验室 编著

机械工业出版社

本书是由小左实验室所有成员共同编写的一本极具学习价值和参考价值的图书。本书主要以Arduino Mega2560为主线,展开对Arduino各方面知识系统全面的讲解,通过丰富详尽的例程讲授当前流行的Arduino知识,使初学者可以很快入手,并且创造出自己的Arduino项目。同时,本书采用手把手的教学方式,使读者学会如何使用各种电子元器件,以及学会如何实现使用Arduino编程对各电子元器件的控制和通信。

第1章详细地介绍了Arduino Mega2560的硬件资源及Arduino的编程环境;第2章讲述了Arduino语言的程序结构、基本语句、程序控制、操作符、变量、基本函数等;第3~6章通过实验的方法分别介绍了Arduino Mega2560的串口通信、I/O端口和时间函数的操作、模拟信号的采集并实现A-D转换及PWM的输出;第7章主要介绍了Arduino内部库,类的定义、结构、使用和特性;第8~11章主要通过实验的方法介绍了Arduino内部库和类的使用、I<sup>2</sup>C通信、SPI通信、中断、键盘显示和数据存储。

本书主要面向对象是Arduino的初学者,同时本书也介绍了一些有难度和实用性强的项目,对于相关技术人员也有很好的参考价值。

## 图书在版编目(CIP)数据

轻松玩转 Arduino 编程 / 小左实验室编著. — 北京:机械工业出版社, 2016.11

(创客+)

ISBN 978-7-111-55495-0

I. ①轻… II. ①小… 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(2016)第 279204 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑:江婧婧 责任编辑:江婧婧

责任校对:肖琳 封面设计:鞠杨

责任印制:李昂

北京中科印刷有限公司印刷

2017 年 3 月第 1 版第 1 次印刷

169mm×239mm·9.25 印张·194 千字

0 001—3000 册

标准书号:ISBN 978-7-111-55495-0

定价:55.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

服务咨询热线:010-88361066 机工官网:www.cmpbook.com

读者购书热线:010-68326294 机工官博:weibo.com/cmp1952

010-88379203 金书网:www.golden-book.com

封面防伪标均为盗版

教育服务网:www.cmpedu.com

# 前言

本书将针对自动化工程、电气工程程序设计零基础的学生或工程师，培养他们对单片机 Arduino (C++) 语言编程及其外围电路的设计能力，增强他们对单片机的亲近感和应用理解。通过本书的学习，使大家掌握单片机的基本原理，独立设计单片机外围电路，编写单片机的 Arduino 程序，掌握嵌入式系统的面向对象程序设计方法。

本书内容分为两部分，共 11 章，基础部分（第 1~6 章）：主要讲解 Arduino 开发板、Arduino 程序与 C++ 语言、串口通信、I/O 端口、A-D 采样、PWM 输出与模拟信号。进阶部分（第 7~11 章）主要讲解与外围电路的通信，及一些实用编程技术和应用，例如，类设计与库、I<sup>2</sup>C 通信、SPI 通信、中断、数据存储。

本书内容浅显易懂，实现方法多变，鼓励以自学为主，以动手实践为辅，内容学习之后安排电路搭建实验或程序设计实验作为练习。

## **本书适用人群与阅读目标：**

本书将针对电气工程、自动化工程、电子工程、通信工程以及非信息类工程有一定电路基础的学生和工程师，培养他们对单片机 Arduino (C++) 语言编程及其外围电路的设计能力，增强他们对单片机的亲近感和应用理解。

## **内容特色：**

通过本书的学习，使学生完全掌握单片机的基本原理和工程应用，能够独立设计单片机外围电路，编写单片机的 Arduino 程序，基本掌握嵌入式系统的面向对象程序设计方法。

# 目 录

## 前言

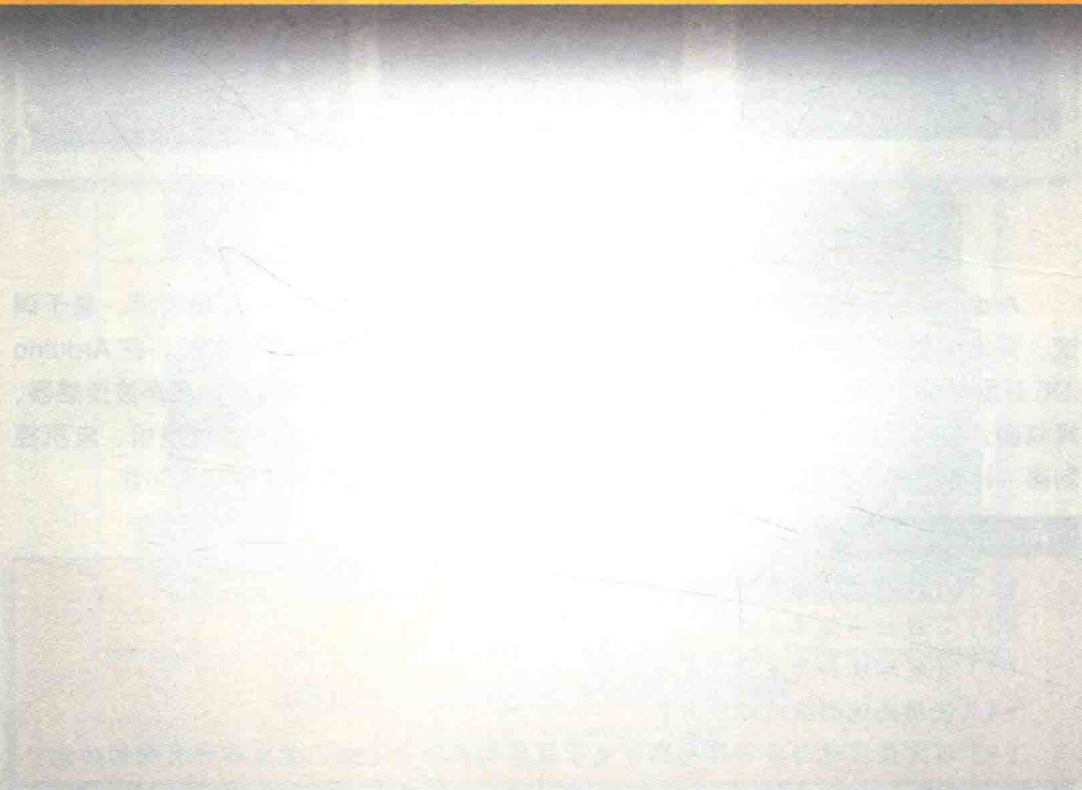
|                                       |           |
|---------------------------------------|-----------|
| <b>第 1 章 Arduino 初解与 Mega2560 实验板</b> | <b>1</b>  |
| 1.1 Arduino 的前世今生                     | 2         |
| 1.2 Mega2560 R3 开发板                   | 3         |
| 1.3 Arduino 软件安装                      | 6         |
| 1.4 Arduino 编程环境                      | 10        |
| 1.5 第一个程序运行                           | 13        |
| 1.6 硬件与软件结合的产物                        | 14        |
| <br>                                  |           |
| <b>第 2 章 Arduino 程序与 C++ 语言</b>       | <b>18</b> |
| 2.1 程序结构                              | 19        |
| 2.2 基本编写规则                            | 19        |
| 2.3 常量                                | 20        |
| 2.4 变量                                | 21        |
| 2.5 类型转换函数                            | 22        |
| 2.6 数学表达式                             | 22        |
| 2.7 程序控制                              | 23        |
| 2.8 函数                                | 25        |
| 2.9 Arduino 基本函数资源                    | 27        |
| <br>                                  |           |
| <b>第 3 章 串口通信</b>                     | <b>35</b> |
| 3.1 Arduino Mega2560 与 Arduino IDE 连接 | 36        |
| 3.2 TTL 电平与 RS-232 串口通信               | 36        |
| 3.3 串口通信协议与函数                         | 36        |
| 3.4 字符串通信                             | 38        |
| 3.5 Arduino 程序运行监控                    | 39        |
| 3.6 GPS 模块串口通信                        | 41        |

|                                          |           |
|------------------------------------------|-----------|
| <b>第 4 章 I/O 端口与时间计数 .....</b>           | <b>44</b> |
| 4.1 指定端口输出与读取 .....                      | 46        |
| 4.2 时间函数 .....                           | 47        |
| 4.3 8 位数据直接读写 .....                      | 49        |
| 4.4 七段 LED 数码管 .....                     | 50        |
| <br><b>第 5 章 A-D 采样与模拟信号采集 .....</b>     | <b>55</b> |
| 5.1 模拟量读取 .....                          | 56        |
| 5.2 电位计读取 .....                          | 57        |
| 5.3 酒精浓度检测 .....                         | 58        |
| 5.4 湿度报警 .....                           | 59        |
| 5.5 空速管差压传感器 .....                       | 60        |
| <br><b>第 6 章 PWM 波与模拟信号输出 .....</b>      | <b>63</b> |
| 6.1 PWM 与 LED 灯亮度控制 .....                | 64        |
| 6.2 舵机 ESC 信号 .....                      | 67        |
| 6.3 无刷电机 ESC 信号 .....                    | 70        |
| <br><b>第 7 章 类设计、对象与库 .....</b>          | <b>73</b> |
| 7.1 Arduino 库 .....                      | 74        |
| 7.2 类的特点 .....                           | 74        |
| 7.3 类的构成 .....                           | 79        |
| 7.4 继承 .....                             | 80        |
| 7.5 类文件与库 .....                          | 83        |
| <br><b>第 8 章 I<sup>2</sup>C 通信 .....</b> | <b>85</b> |
| 8.1 I <sup>2</sup> C 通信电路与库函数 .....      | 86        |
| 8.2 磁阻计数据读取 .....                        | 88        |
| 8.3 三轴加速度和角加速度读取 .....                   | 94        |

|                                                             |            |
|-------------------------------------------------------------|------------|
| <b>第 9 章 SPI 通信</b> .....                                   | <b>104</b> |
| 9.1 SPI 通信与电路 .....                                         | 105        |
| 9.2 SPI 通信函数 .....                                          | 106        |
| 9.3 气压高度计数据读取 .....                                         | 107        |
| 9.4 三轴加速度和三轴角速度传感器 SPI 通信读取 .....                           | 113        |
| <hr/>                                                       |            |
| <b>第 10 章 中断、键盘和显示</b> .....                                | <b>120</b> |
| 10.1 Arduino Mega2560 开发板的中断资源 .....                        | 121        |
| 10.2 键盘输入 .....                                             | 122        |
| 10.3 字符型液晶显示模块 .....                                        | 125        |
| <hr/>                                                       |            |
| <b>第 11 章 数据存储</b> .....                                    | <b>131</b> |
| 11.1 E <sup>2</sup> PROM 数据存储与读取 .....                      | 132        |
| 11.2 I <sup>2</sup> C 总线的 E <sup>2</sup> PROM 数据存储与读取 ..... | 134        |
| 11.3 SPI 总线的 SD 存储卡数据读写 .....                               | 138        |

# 第1章

## Arduino 初解与 Mega2560 实验板



## 1.1 Arduino 的前世今生

Arduino 是古代意大利北部的一个国王的名字，他的名字用在现代社会，却代表了一种编写单片机程序的开发语言和编程环境，这得感谢意大利 Ivrea 一家科技设计学校的老师 Massimo Banzi 和当时正在这所学校做访问学者的 David Cuartielles 教授，是他们为非专业类学生学习现代嵌入式计算机技术而研制了这种工具，并在其所在的大学讲解相关课程。虽然如此，但由于其简单易懂，语言开发简单明了，也受到诸多电子类专业的学生、老师和专业开发工程师的青睐，尤其适合开放源代码的项目。

Arduino 开发环境界面是基于开放源代码的原则，可以免费下载，让你很快上手，开发出令人惊奇的电子产品，其网址为 <https://www.arduino.cc>。目前适合 Arduino 的开发芯片，主要包括 Atmel 公司的 Mega 系列 8 位单片机，如 Mega16, Mega328, Mega2560 等。目前主要的学习开发板参见图 1-1，包括 Arduino UNO、Arduino Leonardo、Arduino Mega2560。其中，Arduino Mega2560 是一款资源相对丰富，功能最为典型的开发板，因此本书将以 Mega2560 开发板为主要学习目标，带领读者学习 Arduino 嵌入式软件编程和硬件设计知识。

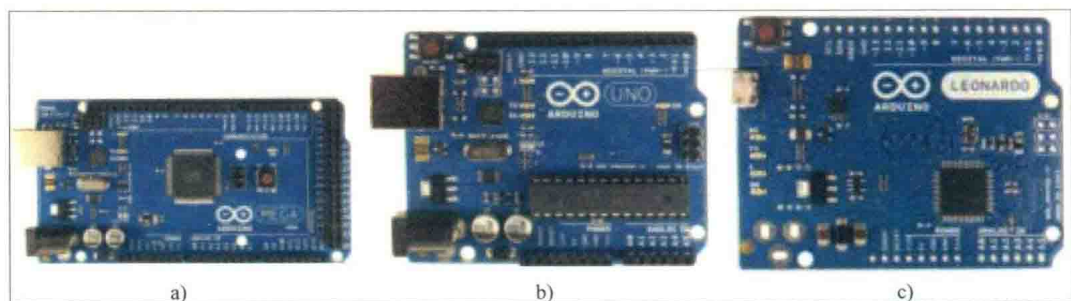


图1-1 Arduino官网开发板图例

a) Arduino Mega2560 b) Arduino UNO c) Arduino Leonardo

Arduino 是一款开源的单片机开发平台，具有编程语言和开发环境简单、易于调试、开发周期短、资源丰富等特点，能够让你快速做出很多有趣的东西。在 Arduino IDE 开发环境下，配合 Mega2560 单片机与其他开发套件，如 LED 灯、超声波传感器、蜂鸣器、按键、GPS、陀螺仪、光敏电阻、通信模块、直流电机、步进电机、电机控制器、计算机等，搭建集成硬件系统，经过简单的编程就可以实现期望的功能。

### Arduino 的特点：

- (1) 开放源代码的电路设计。
- (2) 下载程序简单方便。
- (3) 开发环境和开发语言简单易懂。
- (4) 使用高速的微处理单片机。
- (5) 可方便实现与各种传感器等电子设备构成硬件系统以完成各种有趣的功能。

听听看过 Arduino 相关书籍的学生的评价，会增加你的学习兴趣和自信心：

※ 我彻底爱上了 Arduino，它很自由，很交互，很易用，深深地吸引了我。

※ 没有复杂的单片机底层代码，没有难懂的汇编，只有简单的函数，它必将引领一个新的时代。

※ 它有丰富的接口，简便的编程环境，极大的自由度，可扩展性能非常高，标准化的库模式为它的发展奠定了坚实的基础。

听到这些激动人心的评价，感觉怎么样？下面我们就要开始 Arduino 的探索之旅啦！

## 1.2 Mega2560 R3 开发板

Arduino Mega2560 R3 是一款基于 ATmega2560 单片机的学习开发板，它有 54 路数字输入、输出端口，有 15 个端口可以产生 PWM 输出，16 路模拟信号输入，4 个 UARTS ( Hardware Serial Ports ) 串口，16MHz 晶振，USB 接口，电源端口，ICSP 编程口和一个复位按钮。它包含了支持微处理器的众多能力，一般情况下可简单的通过 USB 线缆与计算机通信并供电，如果外围电路功耗较大，可直接通过开关电源供电。

Mega2560 与以往 Leonardo、UNO 等学习开发板与计算机的连接方式不同，Leonardo、UNO 等学习开发板是通过 FTDI USB-Serial 驱动芯片与计算机串口相连，而 Mega2560 是通过 ATmega16U2 编程的方式虚拟一个 USB-Serial 转换器，简单实现程序下载和串口通信功能。

Mega2560 学习开发板的实物图参见图 1-2，相关电路图和 PCB 资料参见网址：<http://arduino.cc/en/Main/ArduinoBoardMega2560>。

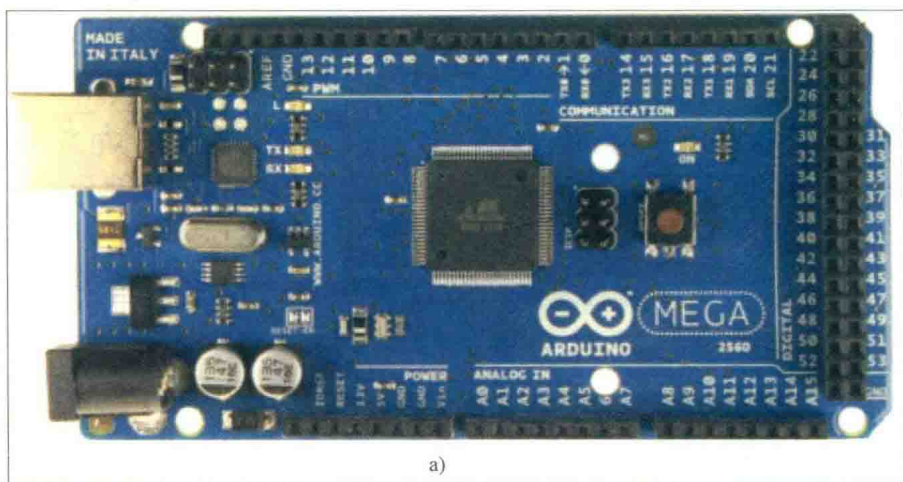


图1-2 Arduino Mega2560 R3学习开发板实物图

a) Arduino Mega2560 开发板正面照

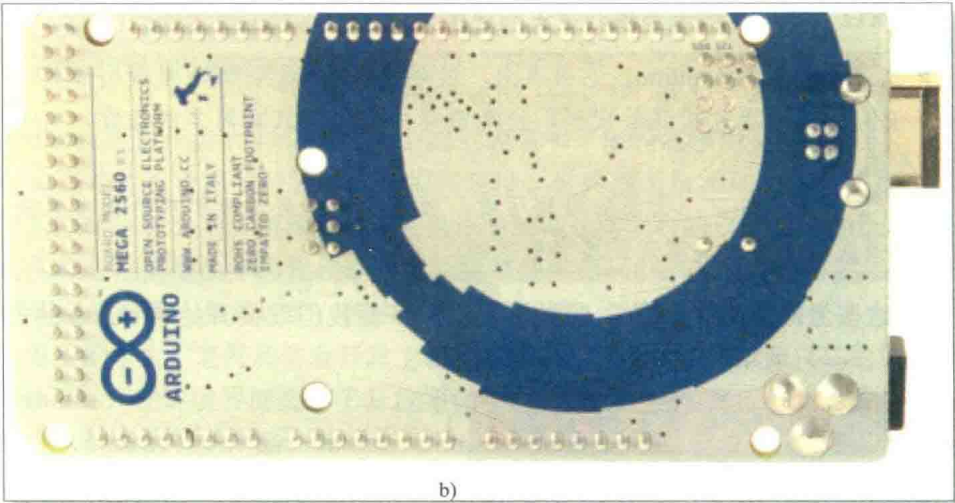


图1-2 Arduino Mega2560 R3学习开发板实物图（续）

b) Arduino Mega2560 开发板背面照

Mega2560 开发板基本参数见表 1-1：

表 1-1 Mega2560 开发板基本参数

| 名 称        | 参 数        | 名 称                     | 参 数                          |
|------------|------------|-------------------------|------------------------------|
| 处理器        | ATmega2560 | 3.3V 端口输出电流             | 50mA                         |
| 工作电压       | 5V         | Flash 存储器               | 256KB                        |
| 供电电压       | 6~20V      | SRAM 内存                 | 8KB                          |
| 数字输入 / 输出  | 54         | E <sup>2</sup> PROM 存储器 | 4KB                          |
| 模拟信号输入     | 16         | Clock 时钟                | 16MHz                        |
| I/O 端口输出电流 | 40mA       | 外部接口                    | 4 UARTs、I <sup>2</sup> C、SPI |

(1) 电源

Arduino Mega2560 R3 可通过 USB 连接供电，或者外部供电，电源自动选择。外部电源来自 AC-DC 适配器，与开发板上的电源插座相连。其外部电源供电电压范围为 6~20V，如果低于 7V 供电，开发板上 5V 输出电源可能低于 5V 或者不稳定，如果高于 12V 供电，电压调节器可能过热并损坏开发板，因此建议的供电电压范围在 7~12V。

开发板上的电源插针描述如下：

- +VIN：当用外部电源时，作为开发板电源输入端。
- +5V：通过开发板调节输出 5V 电压，来自于 DC 电源、USB 或者 VIN 供电端口。
- +3.3V：3.3V 供电端口，来自于电源调节器，最大电流 50mA。
- +GND：接地端。
- +IOREF：开发板提供参考电压的端口，可以通过配置使其工作在 5V 或者 3.3V。

## (2) 存储器

ATmega2560 有 256KB 的 Flash 存储器, 用于存储程序代码, 其中有大概 8KB 用于存储 bootloader; 有 8KB 的 SRAM 随机存储器或称为内存; 4KB 的 E<sup>2</sup>PROM, 在掉电情况下可保存程序重要数据。

## (3) 输入 / 输出

ATmega2560 的 54 个数字端口可设置为输出或者输入, 通过函数 `pinMode()` 设置端口的输入 / 输出的状态, 通过函数 `digitalWrite()` 和 `digitalRead()` 对端口进行读写。端口工作电压为 5V。每个端口可以提供或接收 40mA 的 DC 电流, 端口有内部的上拉电阻, 阻值为 20~50k $\Omega$ 。

+UARTS: Serial 0: 0 (RX), 1 (TX); Serial 1: 19 (RX), 18 (TX) Serial 2: 17 (RX), 16 (TX) Serial 3: 15 (RX), 14(TX)。其中, 0、1、14、15、16、17、18、19 为芯片引脚。

Eternal Interrupts: 2 (Interrupt 0), 3 (Interrupt 1), 18 (Interrupt 5), 19 (Interrupt 4), 20 (Interrupt 3), 21 (Interrupt 2)。

+PWM: 端口 2~13 和 44~46 输出脉宽调制波, 通过 `analogRead()` 函数提供 8-bit PWM 输出。

+SPI: 4 线 SPI 接口, 50(MISO), 51 (MOSI), 52 (SCK), 53 (SS)。这些端口通过 SPI 函数库支持 SPI 通信。

LED 13: 在开发板上有 LED 灯与数字端口 13 相连, 当 13 端口为高电压时, 灯亮, 为低电压时, 灯灭。

I<sup>2</sup>C: 20(SDA), 21(SCL)。通过 `Wire` 函数库支持 I<sup>2</sup>C 双线通信模式。

模拟电压输入: 有 16 个端口可以接收模拟电压输入, 每个端口提供 10bits 的分辨率, 即范围在 0~1023。最大模拟输入电压为 5V, 但也可以通过 `analogReference()` 函数设置 3.3V 参考电压输入模拟信号。

+AREF: 模拟信号的参考电压, 使用 `analogReference()` 函数可对其进行设置。

+RESET: 信号为低时重启微处理器。

## (4) 通信

Arduino Mega2560 开发板可以通过多种方式与计算机、其他 Arduino 开发板或其他微处理器通信。它提供 4 个 TTL 电平的 UARTs 串口通信。ATmega16U2 提供 USB-serial 功能与计算机通信。Arduino 软件包括串口通信, 允许简单的文本数据通过 `serial0` 传递到计算机上, 这时在开发板上的 RX 和 TX LEDs 将闪烁, 数据将经过 ATmega16U2 芯片通过 USB-serial 传输到计算机上。

`SoftwareSerialLibrary` (软串口库函数), 允许通过任意的 Mega2560 串口进行通信。同时, `WireLibrary` 库函数还支持 Mega2560 TWI 总线通信, `SPILibrary` 库函数支持 SPI 通信。

## (5) 编程

Arduino Mega2560 开发板通过 Arduino IDE 软件进行编程，编程细节有单独的章节描述。开发板芯片上有预先烧录的 **bootloader**，允许开发者上载新代码而无需外部硬件编程器。它采用原始的 **STK500** 协议进行通信。当然，也可以通过 **ICSP (In-Circuit Serial Programming)** 上载 **bootloader** 固件。

开发板上 **ATmega16U2** 的固件资源代码参见 **Arduino** 资源，它加载的是 **DFU bootloader**。在 **Mega2560 R3** 版本的开发板上，由一个电阻将 **16U2 HWB** 线连接到 **GND**，使 **ATmega16U2** 进入到 **DFU** 模式。你可以用 **Atmel** 公司的 **FLIP** 软件加载新固件，或者使用外部编程器通过 **ISP** 接口烧录。

### (6) 自动重启

在加载程序之前，无需手动按压 **Reset** 按钮，**Arduino Mega2560** 允许在连接计算机时通过软件运行 **Reset**。**ATmega16U2** 的硬件流程控制线 (**DTR**) 与 **ATmega2560** 的 **Reset** 线通过 **100nF** 的电容相连。当这条线端口拉低电位，同时，**Reset** 线拉低时间足够长时，重启芯片。**Arduino** 软件正是利用这一特点，允许程序员上载代码仅仅通过简单的单击 **Arduino IDE** 开发环境中的 **Upload (上传)** 按钮。此操作意味着有一定的时间延迟，拉低 **DTR** 线能够很好的协调上载程序的启动。

### (7) USB 过电流保护

**Arduino Mega2560** 开发板上有 **500mA** 自恢复熔丝，当 **USB** 端口短路或者过电流时来保护计算机。尽管计算机提供自己的内部保护，熔丝也同时提供一层外部保护层，如果有 **500mA** 电流通过 **USB** 端口，那么自恢复熔丝将自动断开连接，直到短路或负载恢复。

### (8) 开发板外观

**Arduino Mega2560** 开发板长宽外形  $4\text{in}^{\ominus} \times 2.1\text{in}$ ，带 **USB** 连接和电源插头，有 3 个螺钉孔用于固定，数字针 7 与数字针 8 的距离是  $160\text{mil}^{\ominus}$ ，其他针间距为  $100\text{mil}$ 。**Mega2560** 设计的很巧妙，与其他开发板具有一定的兼容性。

## 1.3 Arduino 软件安装

(1) 在淘宝上或者联系本书作者，购买 **Arduino Mega2560** 开发板。

(2) 下载 **Arduino** 软件开发环境。

**Arduino** 是开源软件，最新版本是 **ver1.6.3**，有运行在不同操作系统上的软件，下载时可以参考开发者的操作系统，软件下载地址 <http://arduino.cc/en/Main/Software>。解压缩下载软件，生成软件目录参见图 1-3。目录中文件 **drivers** 为 **USB** 转串口驱动目录，文件 **Arduino** 是软件运行文件。

$\ominus$   $1\text{in}=0.0254\text{m}$ 。

$\ominus$   $1\text{mil}=25.4 \times 10^{-6}\text{m}$ 。

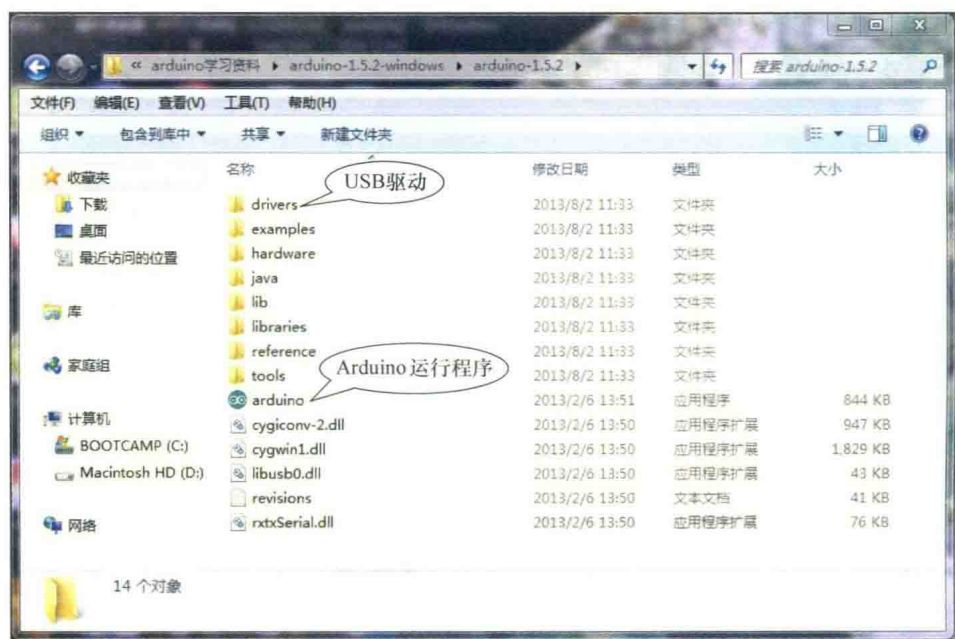


图1-3 Arduino软件目录

### (3) USB 驱动安装

使用 USB 线缆将开发板与计算机相连，系统自动探测到有新硬件。在桌面上，选择“我的电脑”，单击鼠标右键出现弹出菜单，选择“管理”，打开计算机设备管理器，如图 1-4 所示，鼠标右键单击“未知设备”，出现弹出菜单，选择“更新驱动程序软件”。

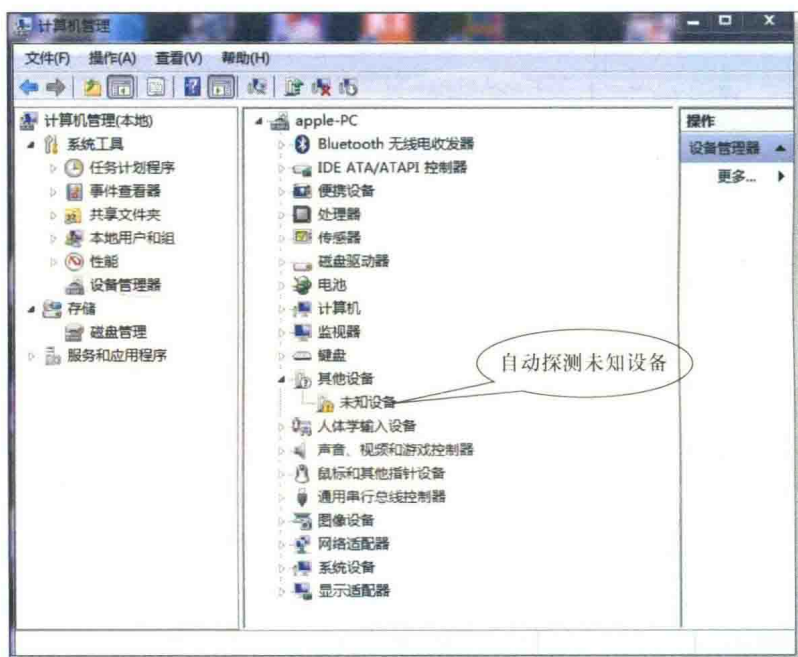


图1-4 设备管理器自动出现新硬件信息

出现驱动程序选择画面后，选择浏览计算机上驱动程序文件，选择 Arduino 软件目录下的 drivers 目录，参见图 1-5。

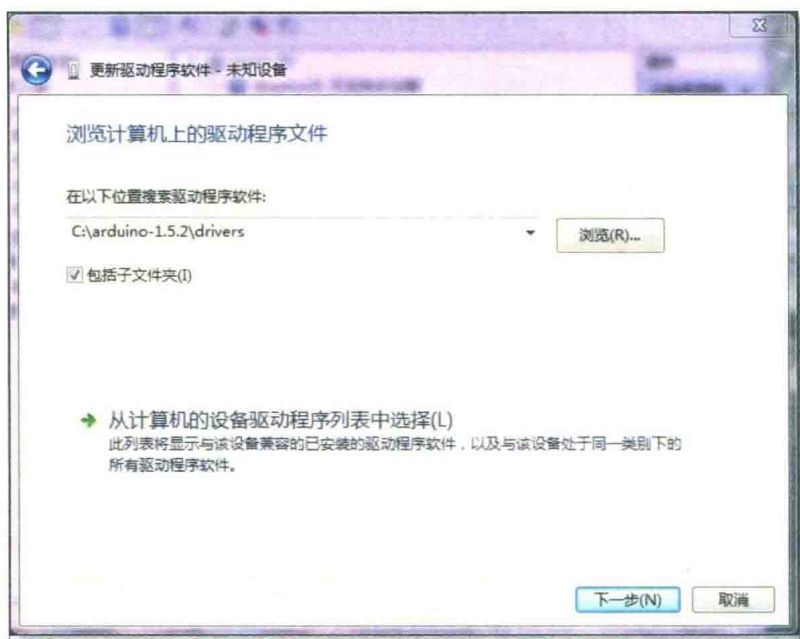


图1-5 选择驱动程序目录

单击“下一步”，按照提示安装驱动，正确安装后设备管理器的画面参见图 1-6。

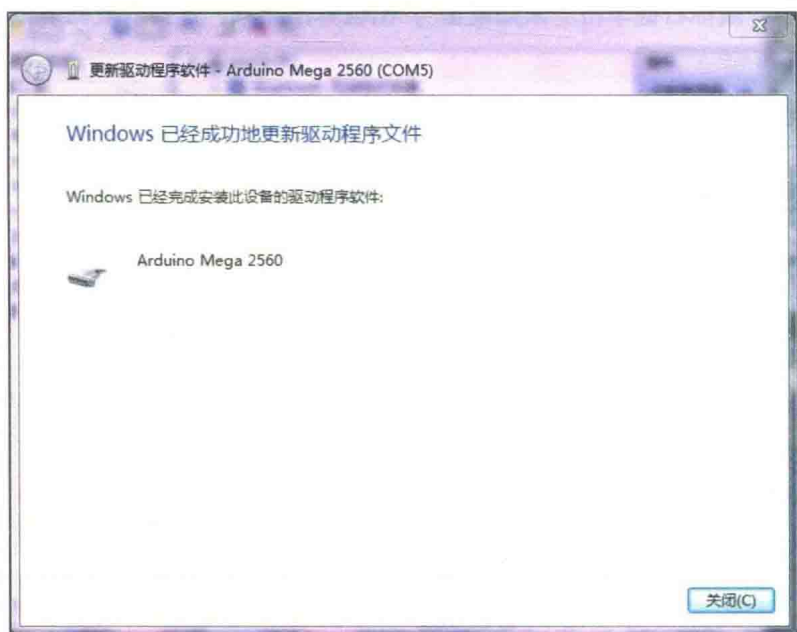


图1-6 USB-serial驱动安装完成

打开设备管理器，未知设备已经安装完成，参见图 1-7。

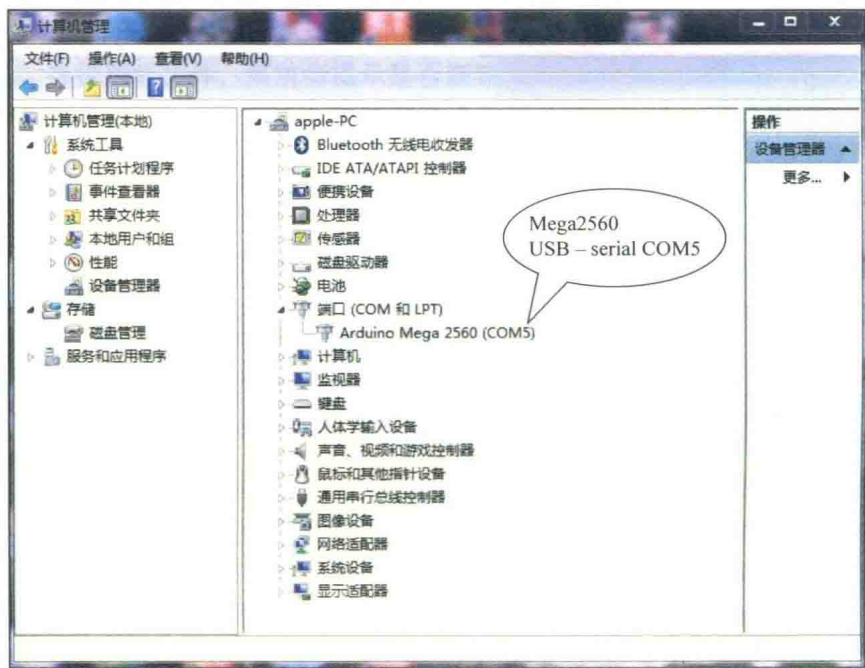


图1-7 正确安装Arduino Mega2560驱动程序的设备管理器

#### (4) 运行 Arduino 软件

正确安装 USB-serial 驱动程序后, 运行软件下载目录下的 Arduino 软件。由于 Arduino 软件是独立运行软件, 无需安装, 因此直接双击 Arduino 运行程序, 运行成功后, 画面如图 1-8 所示。

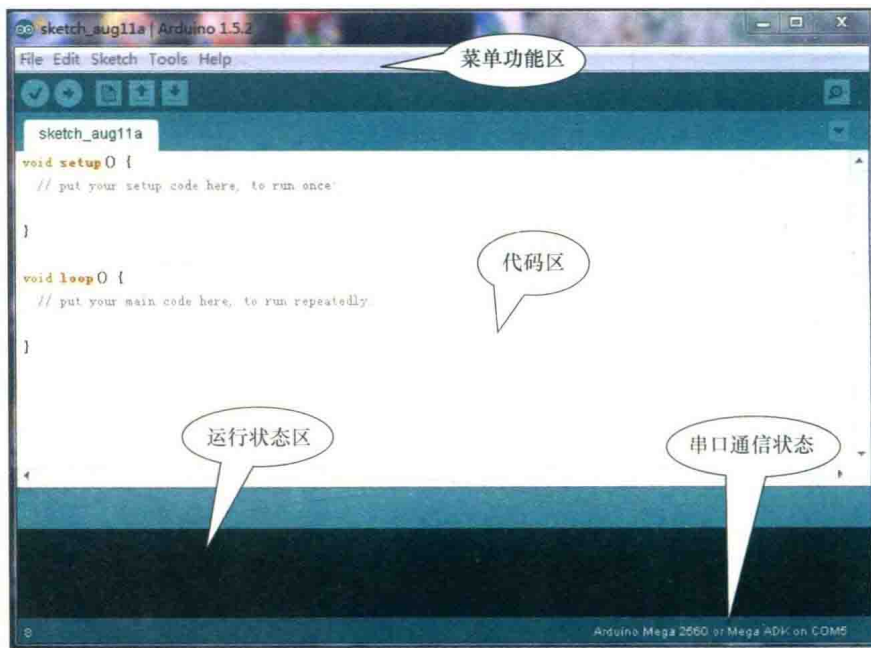


图1-8 Arduino软件开发环境

## 1.4 Arduino 编程环境

### 1. Arduino 编程环境

Arduino 编程环境运行后画面参见图 1-8，画面分为 4 个区：菜单功能区、代码区、运行状态区和串口通信状态区。

菜单功能区：实现软件的主要功能，包括上传程序、程序编辑、设置开发板、串口通信监控、设置烧写器、编译程序、添加文件、引入库函数文件、烧录 bootloader 等。

代码区：程序代码编写区，项目工程可以有多个文件。

运行状态区：程序编译、链接、上传等状态显示，错误信息提示等。

串口通信状态区：显示当前通信的串口状态。

Arduino 软件菜单的主要功能说明如下：

“File”→“New”：新建 Arduino 文件，Arduino 新文件扩展名为 ino。

“File”→“Upload”：通过 USB-serial 方式上传编译好的程序。

“File”→“Upload using programmer”：通过外部编程器的 ICSP 方式上传编译好的程序。

“Edit”→“Comment/Uncomment”：通过 // 方式注释程序或者反注释。

“Sketch”→“Verify/Compile”：验证编译程序。

“Sketch”→“Add File”：添加项目文件。

“Sketch”→“Import Library”：引入特定库文件，用于实现特定功能。

“Tools”→“Auto Format”：对程序代码的自动格式。

“Tools”→“Serial Monitor”：串口通信监视，程序可以通过 serial 0 输出数据。

“Tools”→“Board”：设置开发板型号，如 Arduino Mega2560 或 Mega ADK。

“Tools”→“Serial Port”：设置编程的串口，如 COM5。

“Tools”→“Programmer”：设置外部程序烧录器。

“Tools”→“Burn Boot loader”：烧录引导程序 bootloader。

### 2. Arduino 工程

Arduino 的程序文件以 ino 为扩展名，一个 Arduino 工程可以由多个 ino 文件组成，Arduino 环境自动将工程的相关文件加载。

例如：

(1) 运行 Arduino 软件，系统默认给出一个文件 sketch\_sep23a.ino，注意该文件没有保存到磁盘中。

(2) “File”→“save” 保存文件到一新建目录，取名 file1.ino，该目录将用于本例新建的工程。

(3) “File”→“new” 再次在工程中建立一个新文件，将该文件保存到单独目录，取名 file2.ino。

(4) “Sketch”→“Add file...” 在新建的例程中添加文件，选择文件 file2.ino。当