

第 3 章 Arduino IDE 的安装与使用

本章主要介绍作为 Arduino 使用必不可少的工具之一——Arduino IDE 的安装和使用。为了避免用户在开发环境的安装过程中出现的种种问题，下面将详细地介绍在常用的 Windows 系统下安装 Arduino IDE 以及 Arduino 驱动，之后还对 Arduino 示例程序进行运行和分析。在本章的最后还推荐了一个基于 Arduino 的高级 IDE 以供读者使用。

3.1 Arduino IDE 的安装

Arduino IDE 是 Arduino 官方提供的集成开发环境，可以很方便地为 Arduino 开发板编写程序。本节就详细地介绍了这款 IDE 的下载及安装方式。由于 Arduino 在 Windows 操作系统下的安装过程都类似，因此这里只介绍在 Windows 7 下的安装过程。

3.1.1 Arduino IDE 的安装包下载

Arduino IDE 的安装包可以从 Arduino 官网下载，官网网址为：<http://arduino.cc/en/Main/Software>。在编写本书时，Arduino IDE 的最新稳定版本是 1.0.5，读者可以下载最新版本来使用。官网分别提供了 Windows、Mac OS X、Linux 三大系统的安装包，如图 3.1 所示。

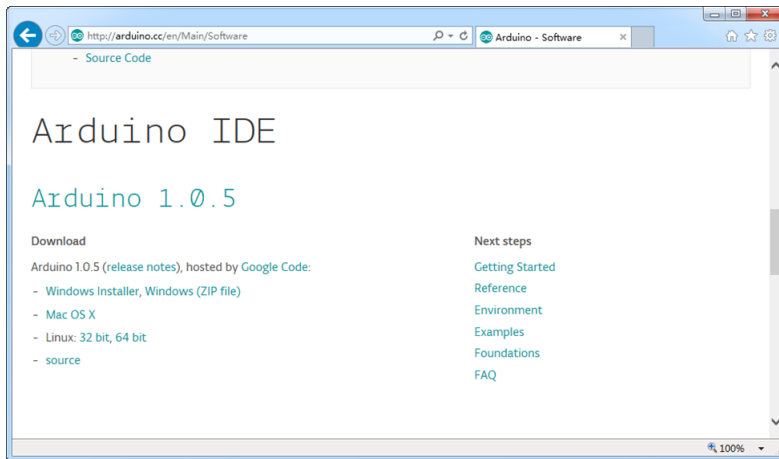


图 3.1 Arduino IDE 下载页面

这里我们将介绍 Windows 操作系统下的安装，所以读者可以下载对应于 Windows 操作系统的版本。下载的文件名为 `arduino-1.0.5-windows.zip` 或者 `arduino-1.0.5-windows.exe`。

3.1.2 使用二进制安装包安装 Arduino IDE

Arduino IDE 的二进制安装包应该是以 .exe 为后缀的文件。这里以在 3.1.1 节中下载的名为 arduino-1.0.5-windows.exe 的二进制文件进行演示安装。首先，双击该文件后会出现如图 3.2 所示的界面。

该界面要求读者阅读相关的许可文件，如果用户同意相关的许可，需要单击右下角的 I Agree 按钮来进行下一步的安装。出现的界面如图 3.3 所示。

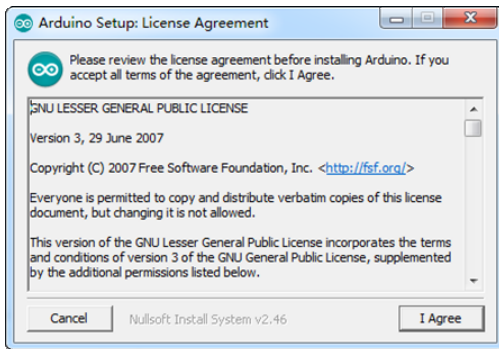


图 3.2 Arduino IDE 安装第一步

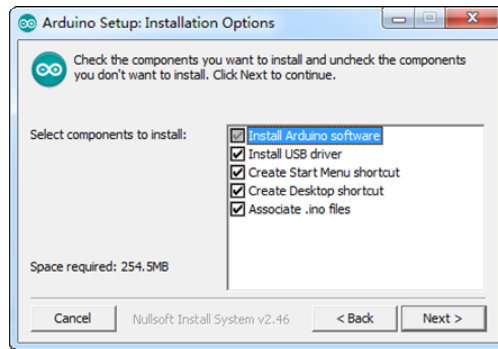


图 3.3 Arduino IDE 安装第二步

该界面要求用户选择要安装的组件。其选项含义如下：

- 安装 Arduino 软件（该项为必选项）；
- 安装 USB 驱动（该项可选）；
- 在开始菜单中创建快捷方式（该项可选）；
- 在桌面创建快捷方式（该项可选）；
- 关联.ino 文件（该项可选）。

用户可以根据自己的需要进行选择，然后单击右下角的 Next >按钮进行下一步的安装。出现的界面如图 3.4 所示。

该界面要求用户选择 Arduino IDE 安装的路径，用户可以根据自己的需求更改。在所有配置确认无误后就可以单击右下角的 Install 按钮开始安装。在安装接近尾声的时候会出现驱动安装提示界面，如图 3.5 所示。

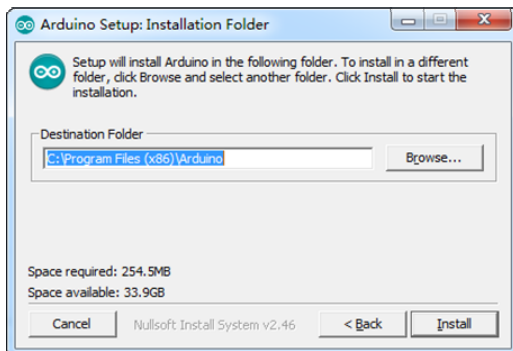


图 3.4 Arduino IDE 安装第三步

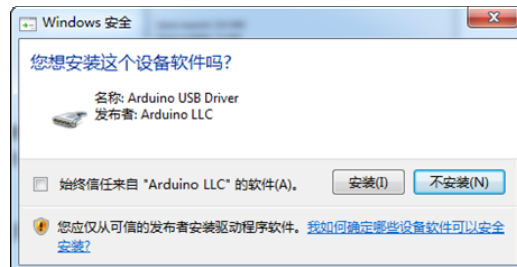


图 3.5 驱动安装提示

读者可以选择一个选项后继续安装。安装完成后会出现如图 3.6 所示的界面。

至此，Arduino IDE 已经被安装到了我们的计算机中，用户可以通过桌面或者“开始”菜单中的快捷方式打开 Arduino IDE。Arduino IDE 的主界面如图 3.7 所示。

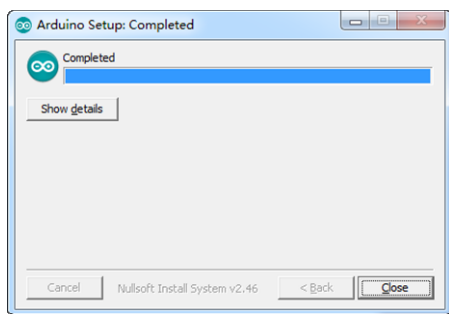


图 3.6 Arduino IDE 安装完成

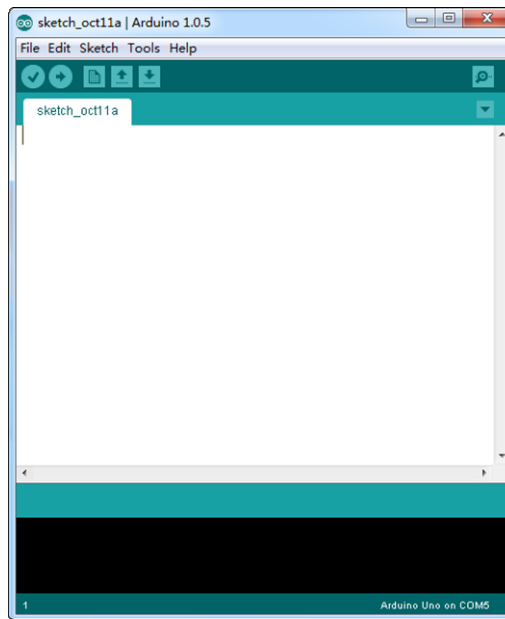


图 3.7 Arduino IDE 主界面

3.1.3 使用压缩包形式安装 Arduino IDE

Arduino IDE 的安装的压缩包应该是以 .zip 为后缀的文件。这里以 3.1.1 节中下载的名为 arduino-1.0.5-windows.zip 的压缩文件进行演示安装。压缩包形式安装的步骤非常简单，只需要使用解压缩工具将压缩包中的内容解压到我们指定的位置即可。在 Arduino IDE 1.0.5 版本中的文件如图 3.8 所示。

名称	修改日期	类型	大小
drivers	2013/5/17 星期...	文件夹	
examples	2013/5/17 星期...	文件夹	
hardware	2013/5/17 星期...	文件夹	
java	2013/5/17 星期...	文件夹	
lib	2013/5/17 星期...	文件夹	
libraries	2013/5/17 星期...	文件夹	
reference	2013/5/17 星期...	文件夹	
tools	2013/5/17 星期...	文件夹	
arduino.exe	2013/5/17 星期...	应用程序	840 KB
cygiconv-2.dll	2013/5/17 星期...	应用程序扩展	947 KB
cygwin1.dll	2013/5/17 星期...	应用程序扩展	1,829 KB
libusb0.dll	2013/5/17 星期...	应用程序扩展	43 KB
revisions.txt	2013/5/17 星期...	文本文档	38 KB
rxtxSerial.dll	2013/5/17 星期...	应用程序扩展	76 KB

图 3.8 以压缩包形式安装后的文件夹内容

提示：用户可以在桌面为 arduino.exe 创建一个快捷方式以方便使用。

通过执行其中名为 arduino.exe 的文件可以打开 Arduino IDE。其主界面如图 3.9 所示。

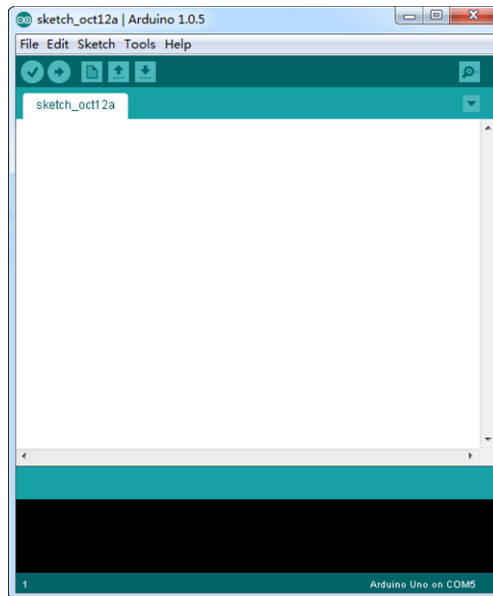


图 3.9 Arduino IDE 主界面

从图 3.9 中可以得知以二进制包和压缩包形式安装的 Arduino IDE 在运行时是没有任何区别的。以压缩包形式安装的优点在于，不会在桌面或者开始菜单中添加新的项目，而且也不会修改注册表项目。因此更加适合喜欢简洁安装的用户。

3.1.4 Arduino IDE 中文化

Arduino IDE 是一款开源的软件，因此对于国际化语言的支持是比较好的。默认的 Arduino IDE 语言是英语。如果读者想要使用简体中文或者其他语言，可以在 Arduino IDE 的 File 菜单中的 Preferences 选项中修改，如图 3.10 所示。

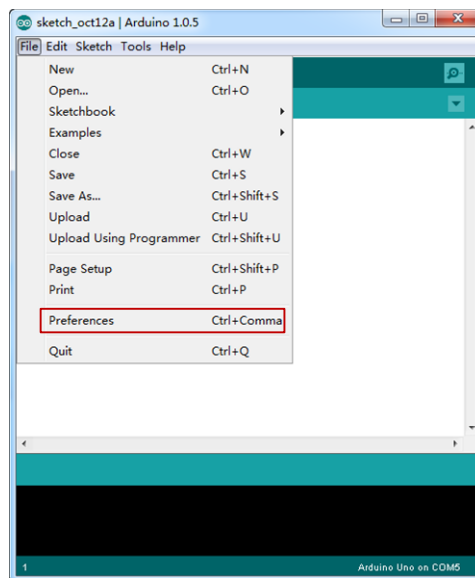


图 3.10 File 菜单项

选择 Preferences 选项后会出现如图 3.11 所示的窗口。

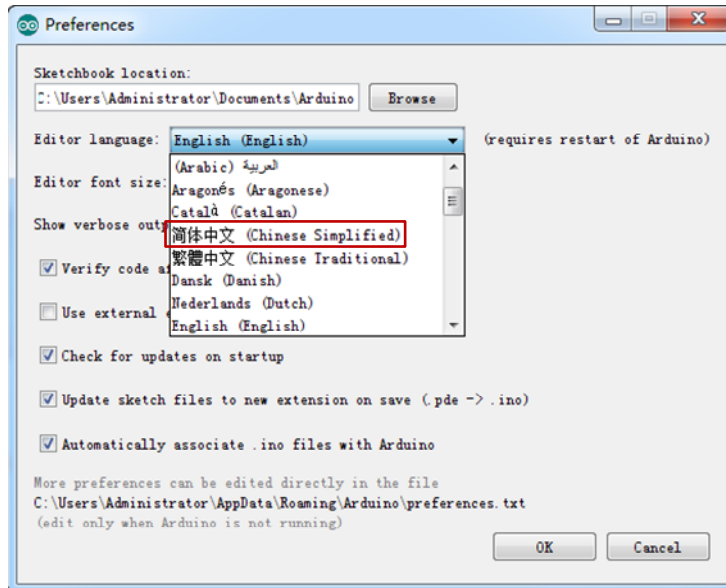


图 3.11 Preferences 窗口

用户在 Editor language 项的下拉框中可以选择需要的语言。这里以选择“简体中文”为例，重新启动 Arduino IDE 后的主界面及其菜单项如图 3.12 所示。

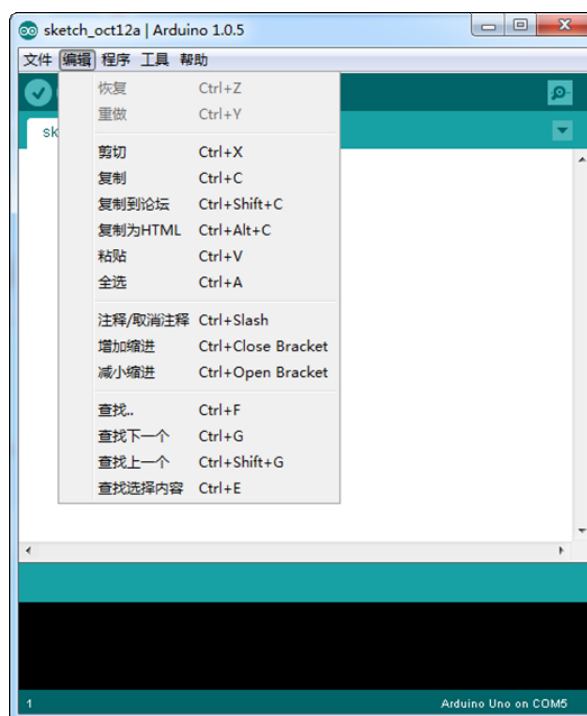


图 3.12 设置语言后的主界面及菜单

可见 Arduino IDE 的主界面及菜单项都做了中文化处理。

3.2 Windows 7 下的 Arduino 驱动安装

在以二进制包形式安装 Arduino IDE 的情况下, Arduino 的驱动通常会在安装时被正确地安装。但是以压缩包的形式进行安装时系统并不会提示用户安装相关驱动, 这就需要用户自己来安装 Arduino 的驱动。如果驱动没有正确地被安装, 那么就会导致 Arduino IDE 和开发板无法连接。因此, 下面将详细介绍 Windows 各系统版本下驱动的安装方法。

3.2.1 自动安装 Arduino 驱动

在安装驱动之前首先需要确认相关驱动是否正确地被安装了。在“设备管理器”中可以查看插入电脑中的设备的驱动情况, 如图 3.13 所示为 Arduino 驱动安装正确的情况(只有 Arduino 开发板连接主机后才显示)。

从图 3.13 中可以看到该设备对应的名称及端口号。驱动没有被正确安装的情况如图 3.14 所示。

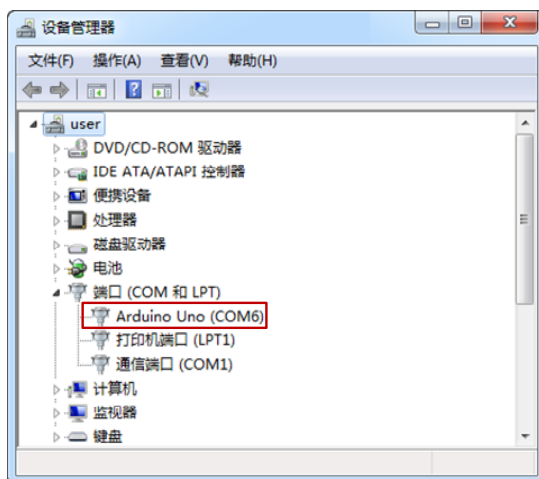


图 3.13 Arduino 驱动安装正确



图 3.14 Arduino 驱动未被正确安装

如果读者在将 Arduino 开发板连接到电脑后在“设备管理器”中看到了图 3.14 中所示的情形, 那么就需要为该设备安装驱动。

在 Windows 7 操作系统下可以在“设备管理器”中右键单击对应的设备来为其安装驱动。如图 3.15 所示。

在选择“更新驱动程序软件”命令后会出现如图 3.16 所示的界面。

在界面中提供了两种方式来搜索驱动程序。选择“自动搜索更新的驱动程序软件”后会出现如图 3.17 所示的界面, 其中显示了当前的搜索和安装进度。

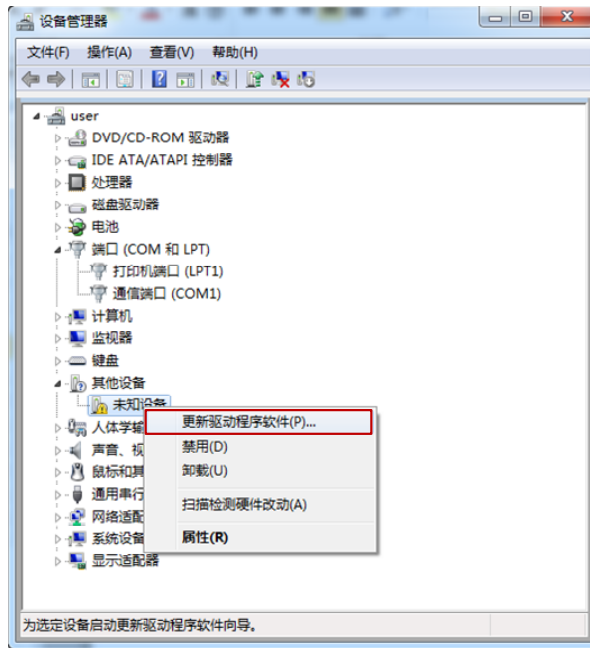


图 3.15 为 Arduino 安装驱动程序

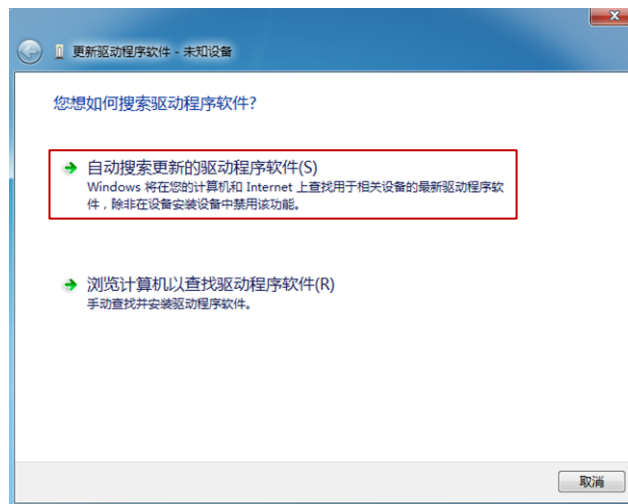


图 3.16 搜索驱动程序方式

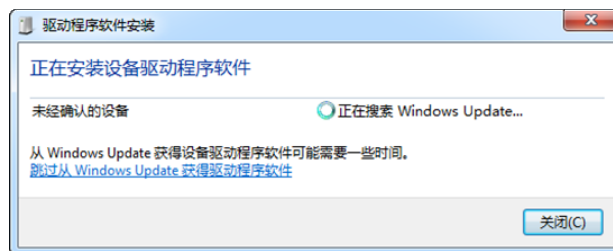


图 3.17 在线安装程序过程

如果出现如图 3.18 所示的界面表示驱动安装成功。

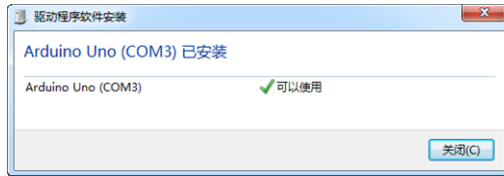


图 3.18 驱动安装成功

在驱动安装成功后就可以正常地使用 Arduino 开发板了。

3.2.2 手动安装 Arduino 驱动

如果 Arduino 开发板的驱动没有正确安装，并且在线安装失败或者没有网络连接的时候，就需要手动为 Arduino 安装驱动。手动安装需要准备 Arduino 的驱动。在 Arduino IDE 的安装包中提供了驱动文件，相关文件在 drivers 目录下，如图 3.19 所示。

名称	修改日期	类型	大小
drivers	2013/5/17 星期...	文件夹	
examples	2013/5/17 星期...	文件夹	
hardware	2013/5/17 星期...	文件夹	
java	2013/5/17 星期...	文件夹	
lib	2013/5/17 星期...	文件夹	
libraries	2013/5/17 星期...	文件夹	
reference	2013/5/17 星期...	文件夹	
tools	2013/5/17 星期...	文件夹	
arduino.exe	2013/5/17 星期...	应用程序	840 KB
cygiconv-2.dll	2013/5/17 星期...	应用程序扩展	947 KB
cygwin1.dll	2013/5/17 星期...	应用程序扩展	1,829 KB
libusb0.dll	2013/5/17 星期...	应用程序扩展	43 KB
revisions.txt	2013/5/17 星期...	文本文档	38 KB
rxSerial.dll	2013/5/17 星期...	应用程序扩展	76 KB

图 3.19 Arduino 驱动所在目录

提示： 下载二进制形式安装包的用户也可以将安装文件解压缩后找到对应的文件夹。

接下来读者可以参考 3.2.1 节的内容打开搜索驱动方式页面。与自动安装驱动的不同之处在于，这里需要选择“浏览计算机以查找驱动程序软件”选项，如图 3.20 所示。

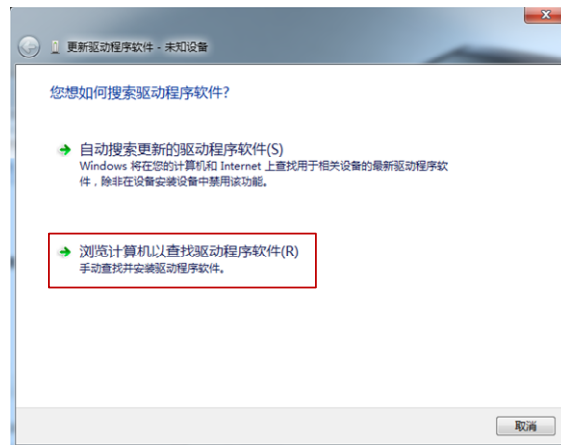


图 3.20 选择驱动安装方式

在选择该方式后会弹出要求用户指定驱动所在目录的窗口，如图 3.21 所示。



图 3.21 浏览驱动程序文件

在选择了正确的驱动文件路径后就可以单击右下角的“下一步”按钮开始安装驱动。在安装的过程中会提示用户是否安装这个设备软件，如图 3.22 所示。



图 3.22 安全提示

在这里单击“安装”按钮后驱动开始自动安装。安装完成后会出现如图 3.23 所示的提示界面。

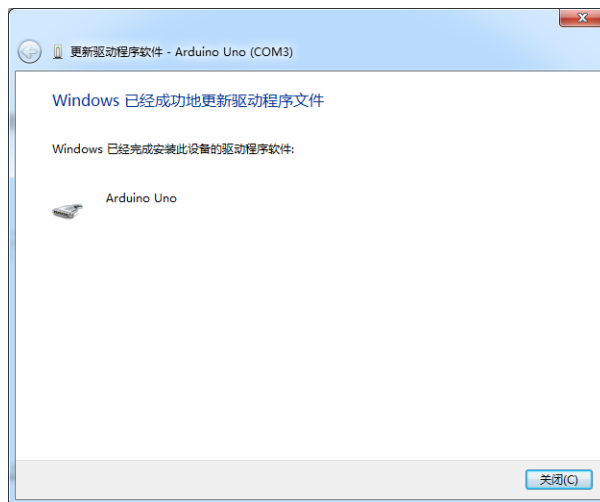


图 3.23 驱动安装完成

至此，Arduino 的驱动就被正确地安装了。其他 Windows 版本为 Arduino 安装驱动的过程与 Windows 7 系统类似，读者可以参考本节以及 3.2.1 节的讲解进行安装。

3.3 在 Arduino 上运行程序

在正确安装了 Arduino IDE 和 Arduino 驱动后，就可以使用 Arduino IDE 为 Arduino 开发板开发程序了。当然本节的主要内容是为读者演示 Arduino IDE 的基本使用方法，以及不同类型的两种 Arduino 程序的示例。

3.3.1 Arduino IDE 主界面简介

Arduino IDE 的主界面是非常简洁明了的。如图 3.24 所示为对 Arduino IDE 主界面各部分的介绍。

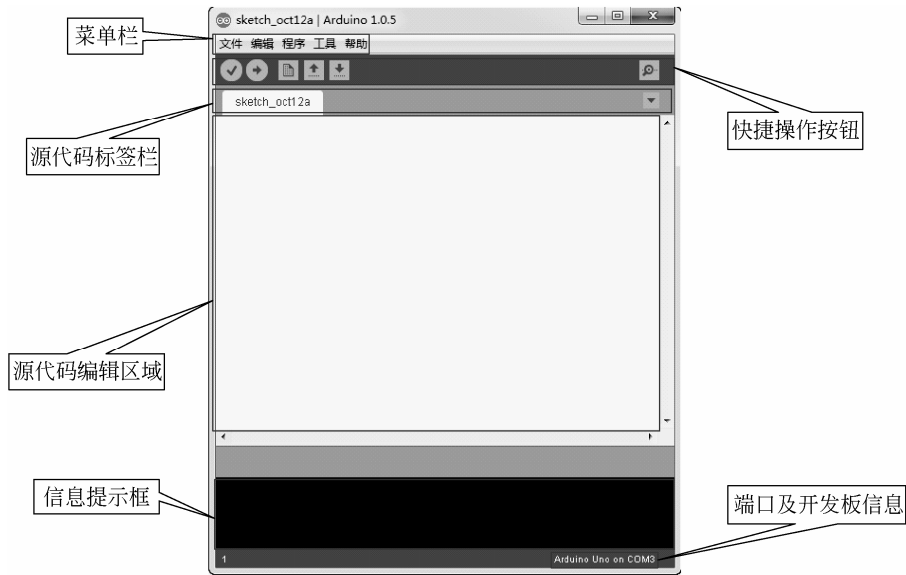


图 3.24 Arduino IDE 主界面

下面简单介绍 Arduino IDE 主界面中各部分的作用。在了解了所有这些界面部分以后，读者就可以很轻松地使用 Arduino IDE 了。

1. 菜单栏

同大部分应用程序类似，Arduino IDE 的菜单栏将各类不同操作聚合到对应的菜单项中，用户可以根据自己想要操作的类别来选择不同的菜单项。

2. 快捷操作按钮

快捷按钮是将用户最常用的操作以按钮的形式呈现出来。快捷操作按钮所实现的功能通过菜单栏同样可以实现。从左到右快捷按钮的名称及功能如下所述。

- ❑ 校验：检查程序中是否存在问题；
- ❑ 下载：将编译后的文件下载到 Arduino 开发板中；
- ❑ 新建：新建一个源文件；
- ❑ 打开：打开一个 Arduino 程序；
- ❑ 保存：将编写好的程序源代码保存；
- ❑ 串口监视器：从指定串口接收数据或者向指定串口发送数据。

3. 代码编辑区域

大部分程序源文件的编写过程都在这里进行。

4. 信息提示框

提示一些操作的过程信息，例如编译过程和下载程序过程的操作信息。

5. 端口及开发板信息

用来提示开发板的类型，以及位于主机的哪个端口。

3.3.2 运行一个闪烁 LED 示例程序

Arduino IDE 提供了大量的示例来供用户学习，本节就来演示一个不需要其他扩展就可以运行的 Arduino 程序。这个示例程序可以通过选择“文件”|“示例”|01.Basics|Blink 命令打开。该源文件会在一个新窗口中打开，如图 3.25 所示。

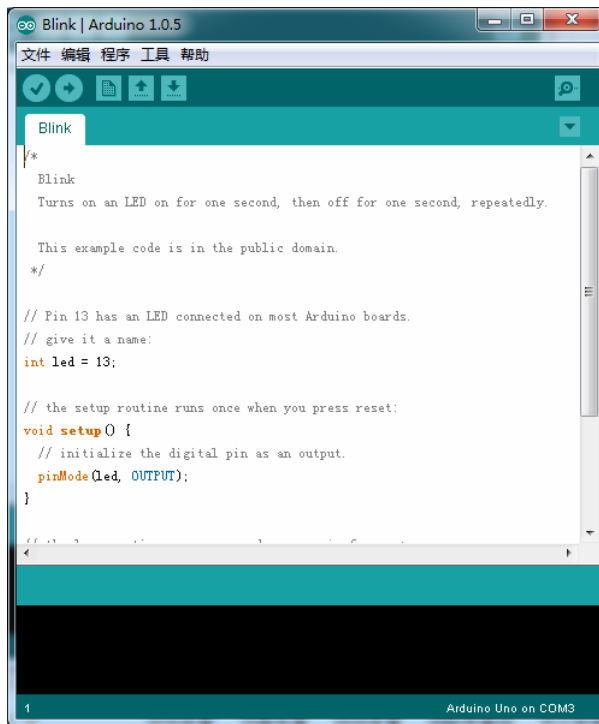


图 3.25 Blink 程序源文件

打开 Blink 示例源文件后读者就可以单击校验快捷键进行校验。程序在没有错误的情况下，信息提示窗口显示编译完成后的二进制程序的大小，如图 3.26 所示。



图 3.26 校验源代码

在出现上述提示后，用户就可以将该程序下载到 Arduino 开发板中了。这段程序的作用是，使开发板上的名为 L 的 LED 灯以一秒的间隔时间不间断地闪烁。

注意：在将程序下载到开发板之前需要正确地 Arduino IDE 选择一个开发板型号。用户可以通过选择“工具”|“板卡”命令来选择正确的开发板。Arduino 开发板需要正确连接到电脑，并且相关驱动运行正常。

在将二进制程序下载到开发板后程序便开始执行了，并且 Arduino 开发板上的 LED 开始闪烁。本节的重点在于演示 Arduino IDE 的操作，因此读者在这里无需理解示例程序的含义。相关的内容将在后续章节中逐步介绍。

3.3.3 运行一个控制台输出示例程序

在有些情况下 Arduino 板子要与主机做一些交互。例如，输出调试信息等。这些信息可以通过串口传输到主机，在主机中通过串口软件就可以读取到 Arduino 开发板发出的信息。Arduino IDE 提供了一个简易的串口监视器，如图 3.27 所示。

提示：该工具可以通过快捷按钮或选择“工具”|“串口监视器”命令打开。

现在，通过 Arduino IDE 将示例中 Communication 分类中的 ASCII Table 程序下载到

Arduino 开发板中，该程序会从串口输出一张 ASCII 码表。在用户使用串口监视器连接到 Arduino 开发板的时候，该程序就会开始发送这些表的内容，如图 3.28 所示为串口监视器接收到内容并将它们显示出来。

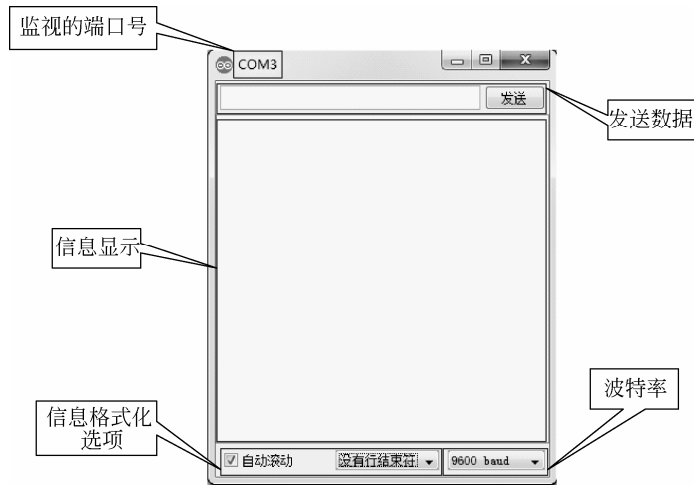


图 3.27 串口监视器事项

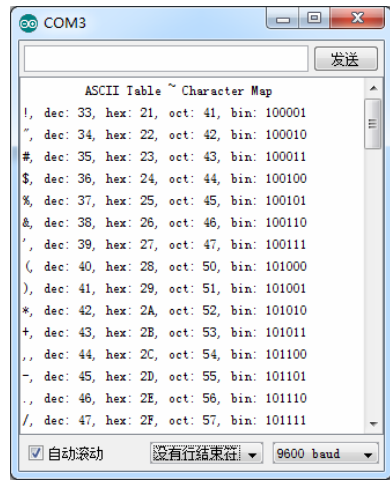



图 3.28 串口监视器显示信息


串口监视器中输出的信息为对应符号的十进制、十六进制、八进制和二进制的表示形式。

3.4 Arduino IDE 编码流程

Arduino IDE 只是一个更加符合 Arduino 特性的集成开发环境。因此，有过使用 IDE 经验的读者可以很容易上手。如果读者完全没有相关的 IDE 操作经验，那么就可以通过本节的讲解来快速上手。通常，保存代码的文件称为源文件，Arduino 官方将保存 Arduino 程序的文件称为 sketche，在本书的编写中将不做明显的区分。

3.4.1 创建、保存和打开源文件

新建源文件可以通过菜单栏中的“文件”|“新建”命令、快捷按钮或快捷键 Ctrl+N 完成。如图 3.29 所示为新建一个名为 sketch_oct14a 的源文件。

在创建了源文件后就可以在这个文件的基础上编写代码了。在编写代码的过程中，所编辑的内容都是保存在缓存中的，随着 Arduino IDE 的关闭，这些修改将会被丢弃。如果想要将这些编辑的内容长期存储，就需要将这个文件存储。存储文件可以通过菜单栏中的“文件”|“保存”命令、快捷按钮或快捷键 Ctrl+S 保存，Arduino IDE 程序会将文件默认保存为以 .ino 为后缀的文件。

提示：一个新创建的文件在第一次保存的时候会要求用户为该文件指定保存位置，在之后执行保存操作的时候只会将对文件的修改写入对应的文件。

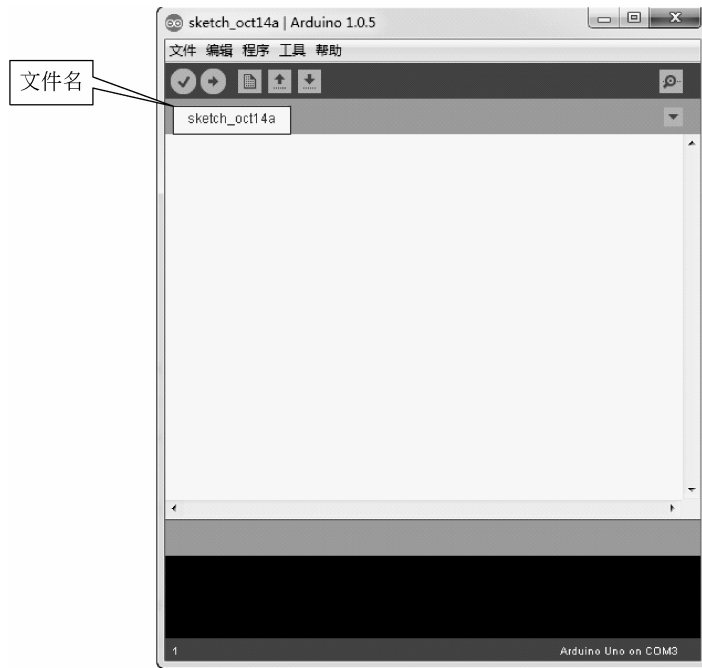



图 3.29 创建源文件

在 Arduino IDE 中可以将一个程序写在多个文件中，这些文件可以通过单击标签栏中的  按钮、在弹出的菜单中选择“新建标签”命令或者使用快捷键 Ctrl+Shift+N 创建。如图 3.30 所示为创建一个名为 first 的文件。

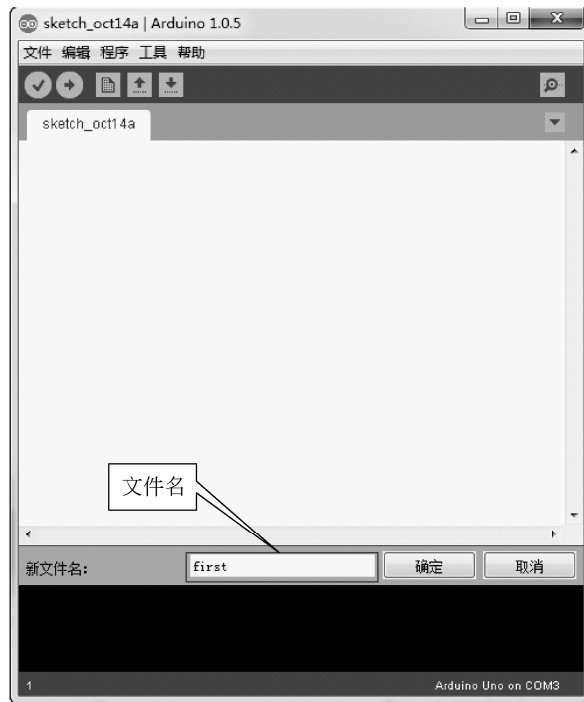


图 3.30 创建多个文件

在创建完成后，该文件会并列地出现在标签栏中，如图 3.31 所示。

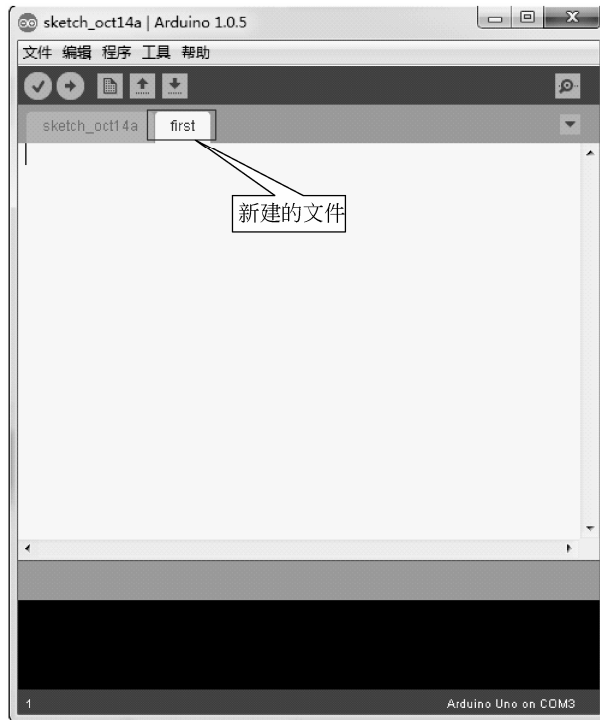



图 3.31 新建的文件

这些在一个标签栏中的文件在校验或者编译的过程中会合成为一个大的源程序。将一个大的程序分布在多个文件中可以使程序更加容易组织和修改。

如果需要修改以前编写的源代码，可以通过菜单栏中的“文件”|“打开...”命令、快捷按钮  或快捷键 Ctrl+O 完成。执行打开文件操作后 IDE 会要求用户指定期望打开的文件名称，如图 3.32 所示为打开名为 myobj.ino 的文件。

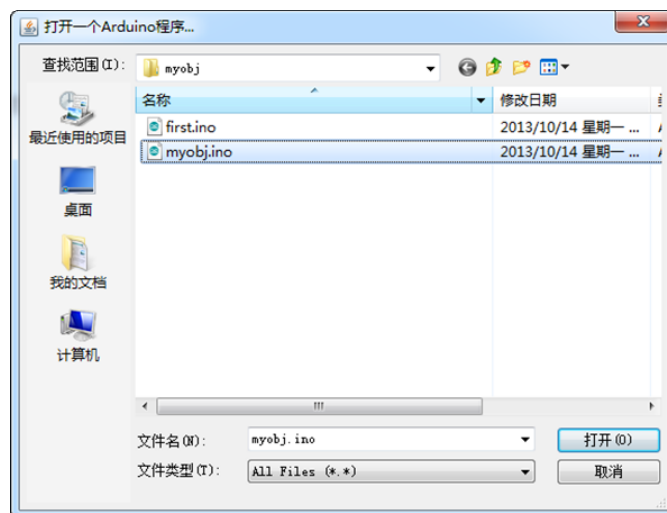



图 3.32 打开文件

 **注意：**在打开一个多文件程序的时候需要打开与程序所在文件夹同名的.ino 文件。否则 Arduino IDE 会提示将该文件保存在另一个文件夹中。

3.4.2 编辑源文件

在 Arduino IDE 中做的大部分工作就是编辑源文件。编辑源文件时只需要将期望的代码写入编辑区域即可，如图 3.33 所示。

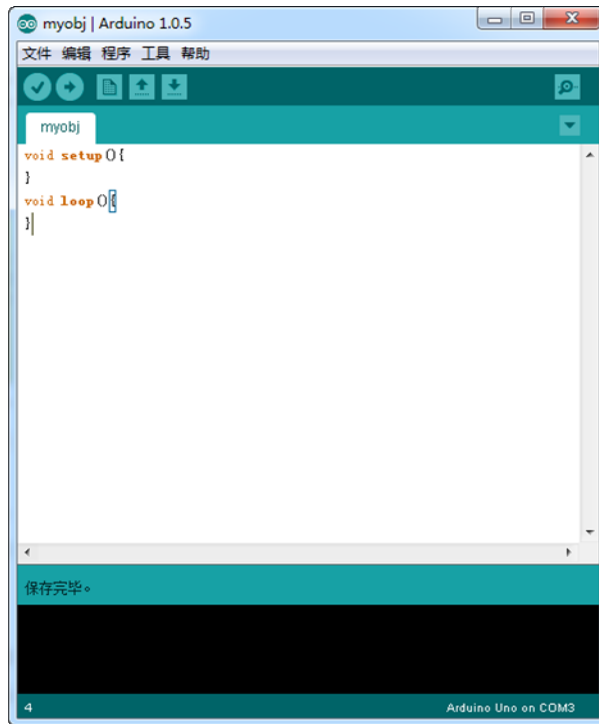




图 3.33 编辑源文件

前面提到过可以将一个程序分别写入到多个文件中。因此，将以上源文件内容写入到多个文件与写入到一个文件中是等价的。

 **提示：**在编码过程中可以随时通过 Ctrl+S 快捷键保存编辑内容以防止数据意外丢失。

3.4.3 校验源文件

校验源文件就是通过编译源代码来检查代码中是否有错误。校验可以通过菜单栏中的“程序”|“校验/编译”命令、快捷按钮  或快捷键 Ctrl+R 完成。如果源代码中没有错误，则 Arduino IDE 会将编译成的二进制程序的大小输出，如图 3.34 所示。

如果程序中有错误，那么校验结果信息中会提示相应的错误位置及原因，如图 3.35 所示。



图 3.34 源代码正确的校验结果

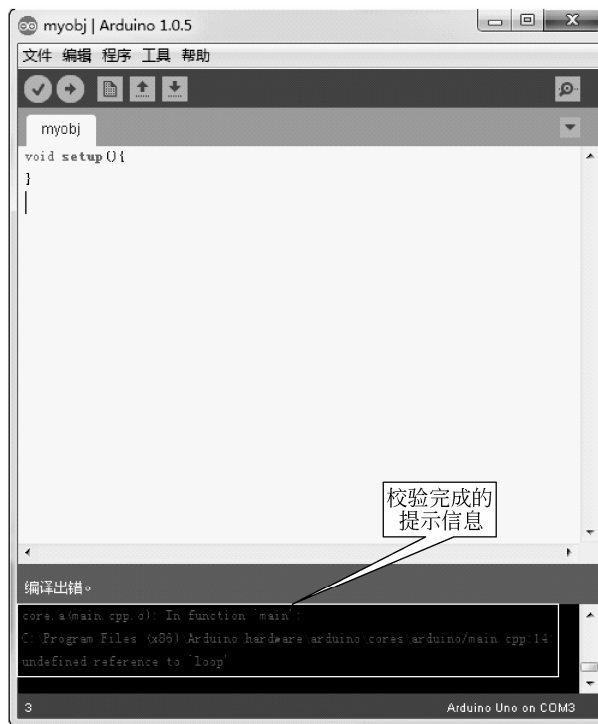



图 3.35 提示错误信息

用户可以根据提示信息修正代码后重新执行校验直到成功编译。

3.4.4 下载程序到开发板

下载的作用就是将编译好的二进制程序文件传输到 Arduino 开发板中。可以通过菜单栏中的“文件”|“下载”命令、快捷按钮或快捷键 Ctrl+U 完成。

说明：在下载之前 Arduino IDE 会首先校验源文件，在生成正确的二进制程序文件后才会启动下载。如果程序出错，则不会将其下载到 Arduino 开发板中。

3.5 高级的 Arduino IDE——MariaMole

Arduino 官方提供的 Arduino IDE 已经可以满足很多的使用情景了，这里再推荐一款基于 Arduino IDE 的高级 IDE——MariaMole。

MariaMole 可以从其官网免费下载，其网址为：<http://dalpik.com>。当前的版本号为 0.5.5，目前只有 Windows 操作系统的版本，其他系统的版本正在开发中。MariaMole 的安装过程非常简单，这里不做详细的介绍。MariaMole 的主界面如图 3.36 所示。

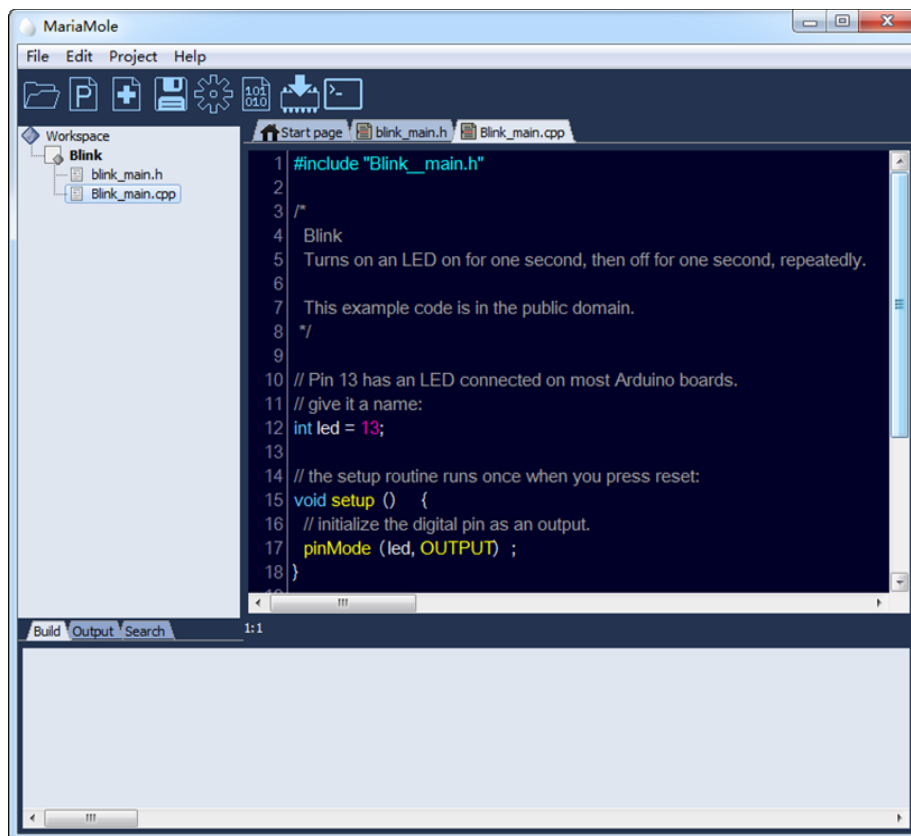


图 3.36 MariaMole 主界面

由于 MariaMole 并不是一个独立的 IDE，它需要依赖 Arduino IDE 中的一些工具。因

此，在 MariaMole 安装完成后需要通过 MariaMole 菜单中的 Edit | Preferences 命令来指定 Arduino IDE（参照 3.1 节进行安装）的安装位置，如图 3.37 所示。

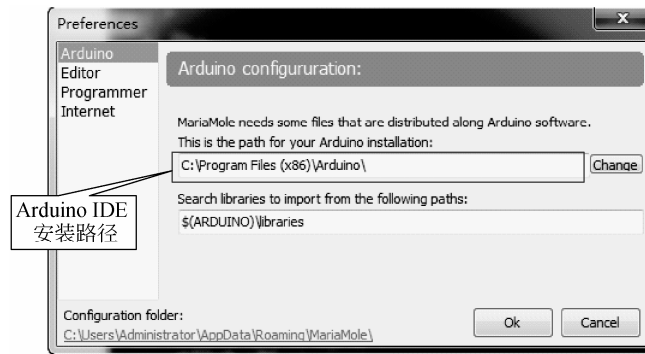


图 3.37 配置 MariaMole

相对于 Arduino IDE，MariaMole 有如下的特性：

- 基于 Arduino IDE；
- 工作区支持多项目；
- 同时支持多个串口监视器；
- 编译流程可配置；
- 可以很容易导入 Arduino 示例和代码；
- 可以很容易导入 Arduino 库；
- 可以自定义 IDE 主题。

由于我们在之后的学习过程中主要使用的是 Arduino IDE，因此，关于 MariaMole 的使用方式这里不做详细介绍。