

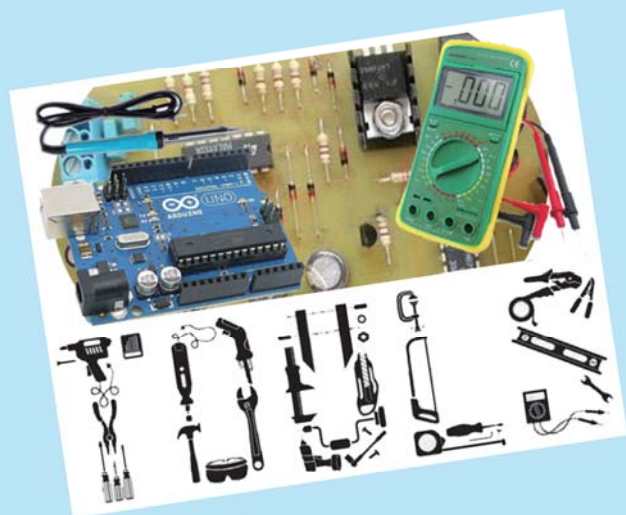
第一章

开源硬件设计



随着创客教育的普及和开展，开源硬件现已成为创客最常用的工具之一。开源硬件不仅被设计师、工程师等专业人员用于科技创新和原型开发，同时也被艺术家、大中小学生等业余爱好者所喜爱。他们利用开源硬件，自主设计、制作，将各种奇思妙想转化为实物作品或产品。其中，Arduino 就是开源硬件的典型代表，它是一个连接虚拟与现实的工具和平台，利用这个平台可以编写计算机程序，并控制现实世界中的各种电子设备。

本章将通过“感受 Arduino 的魅力——电子琴”项目，进行自主、协作、探究学习，让同学们认识开源硬件及 Arduino 平台。通过 LED、感光灯、报警器、触摸传感器等实验，感受 Arduino 的魅力，学会初步使用 Arduino 平台，并能进行简单的实验项目开发，从而将知识建构、技能培养与思维发展融入运用数字化工具解决问题的过程中，促进创新素养达成，完成项目学习目标。



- ◎ 认识开源硬件及 Arduino
- ◎ 初试身手——LED
- ◎ 感光灯
- ◎ 蜂鸣器——报警器
- ◎ 触摸电子琴



项目范例：感受 Arduino 的魅力——电子琴

● 情境

随着全球创客运动的兴起，国内“大众创业，万众创新”热潮风起云涌，各类创客活动也逐步推广并进入中小学校。开源硬件是创客运动的一个缩影，其中的 Arduino 更是众多创客的首选平台之一。基于 Arduino 的开发平台，即使是零基础的非电子专业人员，也可以快速地开发出一些好玩、实用的互动性小作品，解决生活中的一些实际问题，给工作、学习、生活增添更有趣的创意。那么，什么是开源硬件和 Arduino 呢？它们具体能做出什么样的作品呢？

● 主题

感受 Arduino 的魅力——电子琴。

● 规划

根据项目范例的主题，在小组中组织讨论，利用思维导图工具，制订项目学习规划，如图 1-1 所示。

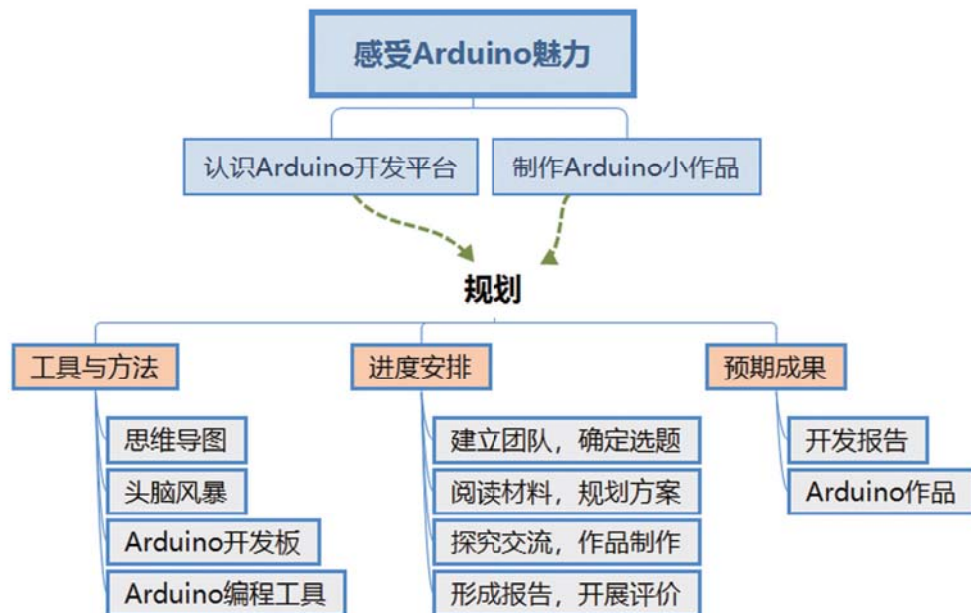


图 1-1 “感受 Arduino 的魅力”项目学习规划

● 探究

根据项目学习规划的安排，通过调查和案例分析、文献阅读或网上搜索资料，开展“感受 Arduino 的魅力——电子琴”项目学习探究活动，如表 1-1 所示。

表 1-1 “感受 Arduino 的魅力——电子琴”项目学习探究活动

探究活动	学习内容	知识技能
认识开源硬件及 Arduino	开源硬件特征	理解开源硬件的概念；认识开源硬件的特征；了解开源硬件的发展历程；了解 Arduino 软硬件平台
	开源硬件发展历程	
	Arduino 硬件、软件开发平台使用	
初试身手——LED	LED 灯的使用	体验利用 Arduino 进行简单实验
感光灯	感光灯的制作	Arduino 数字信号的输入和输出
蜂鸣器——报警器	蜂鸣器的使用	声音传感器的使用
触摸电子琴	Arduino 电子琴作品制作	触摸传感器、蜂鸣器等综合使用及项目制作

● 实施

实施项目学习探究活动，认识开源硬件特征和发展，了解 Arduino 平台特点，并能利用开源硬件特别是 Arduino 制作简单、有趣的作品或解决实际问题。

● 成果

在小组开展项目范例学习的过程中，利用思维导图梳理小组成员在“头脑风暴”等活动中的观点和意见，记录探究过程和结果，运用“写得”等数字化学习工具综合加工和表达，形成项目范例可视化学习成果，如“感受 Arduino 的魅力——电子琴”可视化报告，运用开源硬件创作简易作品，并通过各种途径和平台进行分享发布。

● 评价

根据本书附录 2 的“项目活动评价表”，对项目范例的学习过程和学习成果在小组和班级上进行交流，开展项目学习活动评价。

● 项目选题

请同学们以 3～6 人组成一个小组，选择下面一个参考主题，或者自拟一个感兴趣的主题，开展项目学习，创作简易小作品。

- (1) 体验 Arduino 的其他型号开发板与软件。
- (2) 体验掌控板或 micro:bit 开发平台。
- (3) 体验树莓派或其他开源硬件开发平台。

● 项目规划

各小组根据项目选题，参照项目范例的样式，利用思维导图工具，制定相应的项目方案。



● 方案交流

各小组将完成的方案在全班进行展示交流，师生共同探讨，完善相应的项目方案。

第一节 认识开源硬件及 Arduino

什么是开源硬件？Arduino 是什么？它们之间有什么关系？Arduino 有什么作用？

一、开源硬件

开源硬件（Open Source Hardware）是开源软件定义的延伸，是指源代码开放的硬件（设备、机器或者其他实物），任何人都可以拿来使用，重新制造、重新设计以及重新销售。开源硬件的源代码，指的是开源硬件的设计，包括项目组成、概述、原始设计文件、辅助设计文件、物料清单、软件、照片和解释说明等。核心体现在电路原理图、物料清单、PCB（电路板）图等。

硬件与软件的不同之处是，硬件资源始终致力于创造实物产品，如 3D 打印、无人驾驶飞机等技术，都是利用开源硬件的很好的例子。目前，比较流行的开源硬件有 Arduino、掌控板、Raspberry Pi（树莓派）、micro:bit 等。其中，Arduino 是开源硬件界第一个获得大规模成功的项目，由它衍生了大量的产品。

得益于创客运动，开源精神和开源文化从电子爱好者、开源硬件社区开始，逐步走向大众，使普通人也能通过各种开源分享，将创意、创新转化为现实，甚至改变世界。

● 探究活动

【拓展】

开源与开源软件

开源，即开放源代码（Open Source），1998 年 OSI（开放代码促进会）成立时，“开源”一词正式被使用。

软件源代码（source）是可供人阅读的计算机指令文本，是软件产品最直接、最完整的技术描述。商业软件公司通常采取申请版权或者商业秘密的方式对软件源代码实施严格保护，以防止其技术思想被窃取，从而保证软件的销售利润。

开源软件，即开放源代码软件，是指源代码完全公开，自由分发，任何人都可以免费获取。开源软件由无报酬的志愿者通过网络合作的方式完成。开源软件独特的生产和发布方式，取得了令人瞩目的成功。今天，开源软件在全球已被广泛使用于各行各业，改变着世界软件业的发展轨迹。例如，在开源软件方面，有 Linux、Apache、MySQL 等；在互联网方面，有 PHP、Python、Firefox 等，在移动设备操作系统方面，有 Android（安卓）等。

【体验】

选择一个开源硬件或软件，下载或配置其使用环境，了解并体验其使用方法，感受开源文化。

二、Arduino 平台

Arduino 是开源硬件的典型代表之一，是一种便捷灵活、编程简易、能快速解决实际问题的开源电子原型平台，既适用于工程师、艺术家、设计师，也适用于一切对开发互动装置或互动式开发环境感兴趣的爱好者。

Arduino 包含两大部分：硬件部分是可以用来做电路连接的 Arduino 电路板；软件部分是 Arduino IDE，两者都是开放的。使用者既可以获得 Arduino 开发板的电路图，也可以获得 Arduino IDE 的源代码。除了购买 Arduino 电路板外，不需要支付额外的费用。Arduino 使用一种简单的专用编程语言，使用者不必掌握汇编语言和 C 语言等复杂技术就可以进行开发。

1. Arduino 硬件

由于 Arduino 属于开源平台，自然人人都可以制作并生产类似的控制板，因此在市面上有各种各样的 Arduino 兼容控制器，如图 1-2 所示。由于各厂家生产的主板，其核心都是一致的，因此，在学习本书的过程中，任意一款兼容 Arduino 的控制板均可正常使用，不必纠结于具体是哪个厂家生产的。但是值得注意的是，如果没有官方授权是不能使用 Arduino 商标的，否则就是侵权行为。

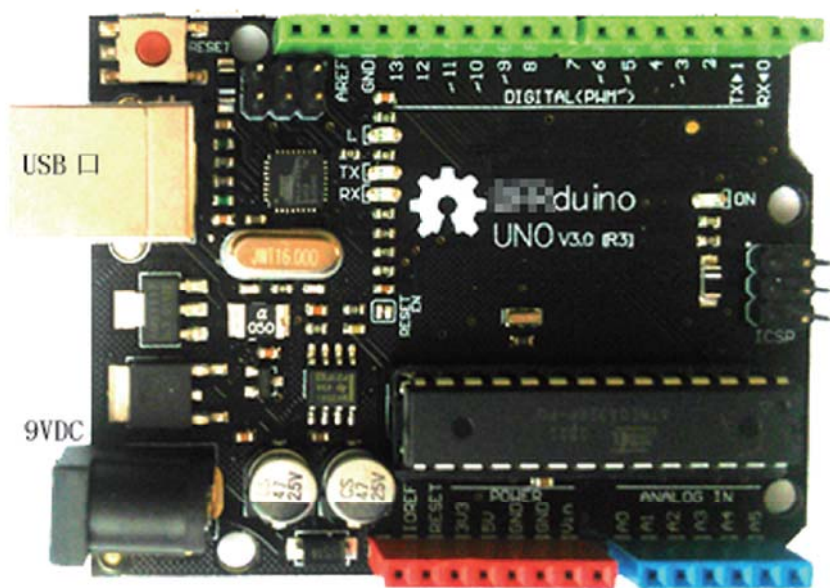


图 1-2 国内某厂家生产的 Arduino 板实物图

【拓展】

上网查找 Arduino 的相关资料，通过各种渠道了解 Arduino 的购买方式，并注意进行甄别和比较。

Arduino 板是一个简单易用的微型控制器，可以连接各种输入设备，一般通过传感器传入信息，也可以驱动各类输出设备，如自平衡车、LCD 显示器、电子玩具、机器人等。Arduino 板最适合进行科学实验，快速实现一些简单的创意，能有效地缩短电子产品的开发周期。

尽管 Arduino 板的改良产品五花八门，但其基本结构都一样，其平面示意图如图 1-3 所示，具体的针脚含义如表 1-2 所示。

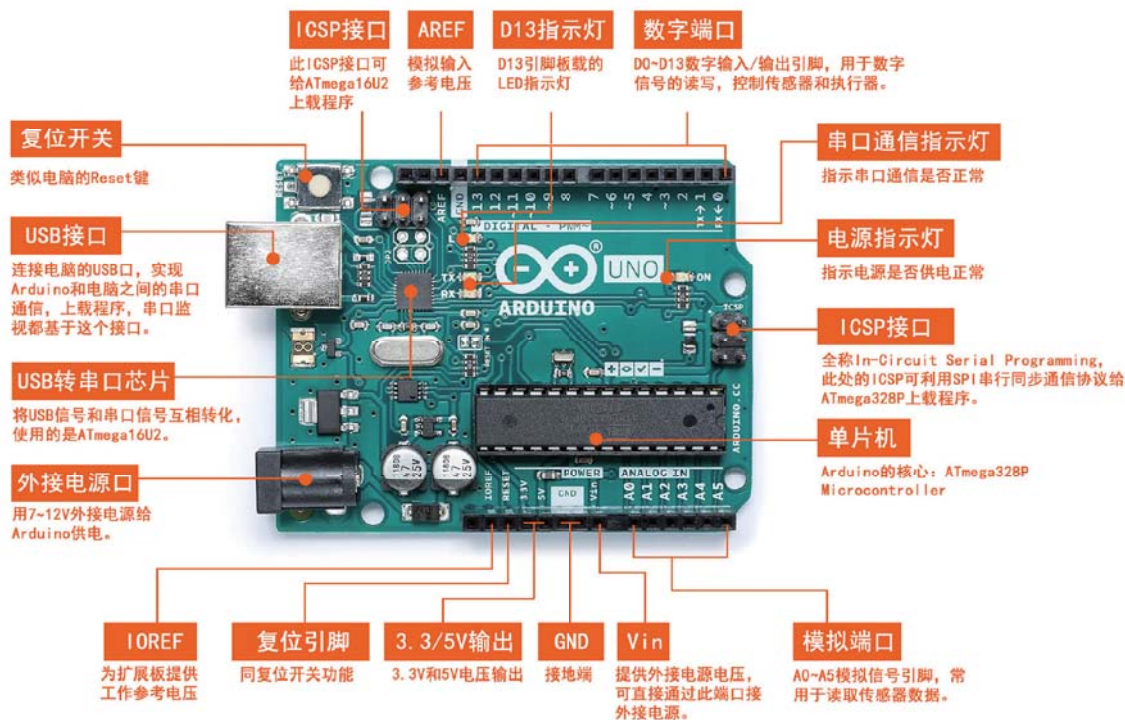


图 1-3 Arduino 板平面图

表 1-2 Arduino 板各引脚说明

上排引脚	下排引脚
RESET: 复位开关，点击可重启主板	IOREF: 输入输出参考电压，预留管脚，扩展用
AREF: 基准电压输入	RESET: 复位引脚，功能与复位开关相同
GND: 接地或地线	3.3V: 3.3V 电压输出
D0 ~ D13: Digital 数字输入 / 输出引脚	5V: 5V 电压输出
TX: Transmit 传送，串行输入 / 输出	VIN: 外接输入电源电压
RX: Receive 接收，串行输入 / 输出	A0 ~ A5: 模拟输入 [0,1023]，模拟输出 [0,255]

2. Arduino 软件

Arduino 板本质上是一台单片机，其核心就是 AVR 单片机（挪威 Atmel 公司 1997 年推出），要建立其输入、输出之间的联系，就需要一个编写其输入 / 输出关联程序的软件平台。Arduino IDE（Integrated Development Environment，集成开发环境）就是一个基于开放源代码的软硬件平台，是专门面向 Arduino 的开发工具，具有使用类似 Java、C 语言的 Processing/Wiring 开发环境。

1) Arduino IDE 的安装和使用

要使用 Arduino IDE，有两种方式：在线使用或下载软件安装使用。

在线方式可直接通过访问 Arduino 的官网 arduino.cc 进入，点击 ONLINE 链接，无须安装即

可使用，但是会受制于网络等因素的干扰，不太稳定。

一般学习过程中通过下载并安装的方式进行（本书后续内容均采用此方式）。安装 Arduino IDE 软件，可以到其官网下载最新版的 Arduino IDE（注意选择与电脑操作系统匹配的版本，如 Arduino For Windows），安装比较简单，只要按步操作即可。安装时可选择“简体中文”，安装完成后，桌面上会出现一个 Arduino IDE 快捷图标，如图 1-4 所示。



图 1-4 Arduino IDE 快捷图标

2) Arduino 的设置

首先，将 Arduino 板接上电脑，然后依次选择“计算机→管理→设备管理器→端口”，检查 Arduino 板所在的端口，如 COM4 等，并记下来。

然后，运行 Arduino IDE，在“工具”菜单的“板：”子菜单中勾选 Arduino Uno，在“端口：”子菜单中勾选 Arduino 板所在的端口（如 COM4），如图 1-5 所示。

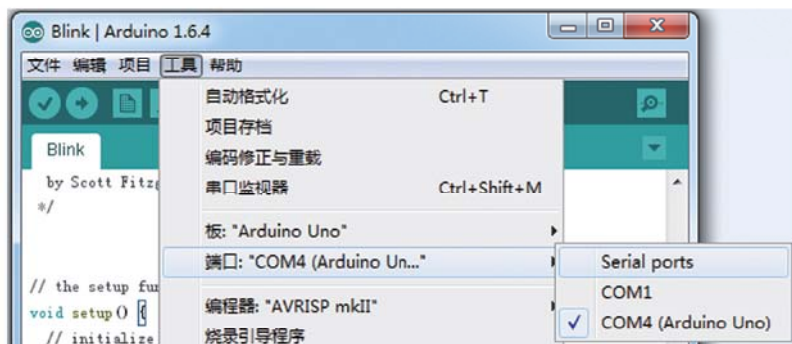


图 1-5 Arduino 的设置

简而言之，设置 Arduino 就是要将所使用的 Arduino 板与 Arduino IDE 对应上。

3) Arduino 程序

Arduino IDE 包括代码编辑器、编译器、调试器等。用 Arduino IDE 可以编写有关 Arduino 板的控制程序，并上传（或烧写）到 Arduino 板脱机执行。

Arduino IDE 程序一般包括设置 setup() 和循环 loop() 两个部分，基本格式参见下面的 LED 闪灯程序（其中的“//”为注释语句标识符）。

（1）LED 与 Arduino 板的连接，如图 1-6 所示。

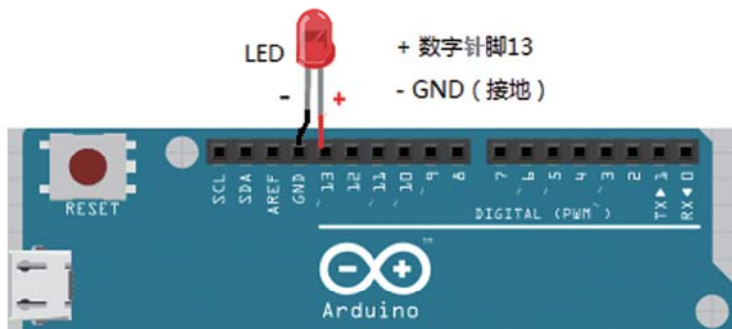


图 1-6 LED 与 Arduino 板连接



(2) LED 闪灯程序:

```
void setup() { // 只运行一次
    pinMode(13, OUTPUT); // 设置数字引脚 13 为输出
}
void loop() { // 循环运行
    digitalWrite(13, HIGH); // 点亮 LED, HIGH 表示高电平
    delay(500); // 等待 500ms
    digitalWrite(13, LOW); // 熄灭 LED, 将 13 引脚设为低电平
    delay(500); // 等待 500ms
}
```

(3) 验证与上传: 点击“验证”按钮 (见图 1-7), 如未发生代码错误, 就可点击“上传”按钮 (见图 1-8), 将程序烧写到 Arduino 板上脱机执行。



图 1-7 “验证”按钮



图 1-8 “上传”按钮

【交流】

通过对开源软件的使用和体验, 结合实际, 讨论其与闭源软件之间的区别和联系。

3. 其他 Arduino 编程工具

用 Arduino IDE 编写程序, 需要具有一定的程序设计语言基础, 如 C 语言等。对于不太熟悉程序设计语言的人来说, 用代码编程既麻烦也费时间, 有没有简便的编程工具呢? 这里, 我们推荐两个图形化编程工具 ArduBlock 和 Mixly。本书后续部分程序将以 ArduBlock 为例进行介绍。

1) ArduBlock

ArduBlock 是一款针对 Arduino 开发的图形化编程软件, 是 Arduino 官方编程环境的第三方软件, 依附于 Arduino IDE, 用它编程就像搭积木一样, 可视化和交互性程度高, 编程门槛低, 适合于初学者。

(1) ArduBlock 的下载安装。

ArduBlock 的下载安装很简单, 只需要将其安装包文件放置到 Arduino 的工具文件夹中即可。具体操作如下。

先从 Arduino 的“文件”→“首选项”里找到项目文件夹的位置, 如“……\Arduino”, 并依次在其中建立分级子文件夹“\tools\ArduBlockTool\tool”。

再将下载的 ArduBlock 压缩文件包直接保存到上述文件夹“……\Arduino\tools\ArduBlockTool\tool”中即可。

(2) ArduBlock 的初步使用。

运行 Arduino IDE, 会发现“工具”菜单中增加了 ArduBlock, 这就是 ArduBlock 软件, 点击它就可进入 ArduBlock 的主界面, 如图 1-9 所示。



图 1-9 ArduBlock 主界面

(3) ArduBlock 模块简介。

ArduBlock 主界面分三部分,上边是工具区,包括“保存”“打开”“下载到 Arduino”三个按钮,可实现编程文件的存取,将图块程序转换为代码程序(方便初学者对比学习 Arduino IDE 代码编程);右边是编程区,采取拖曳、移动的方式像搭积木一样对各模块进行组合、嵌合;左边是各类模块区。

① “控制”——do 循环(主程序 do),当循环(当条件满足 执行……),指定次数循环(重复……),判断结构(如果条件满足 执行……否则执行……;如果条件满足 执行……)。

② “引脚”——数字引脚,模拟引脚,设定数字引脚值,设定模拟引脚值,舵机(引脚,角度),超声波引脚(trigger 发射,echo 回波),以及发声函数(tone 引脚 频率 毫秒, noTone 引脚)。

③ “数字/常数”——数字、字符串的常量、变量,逻辑变量等。

④ “常用计算”——逻辑运算、关系运算、数学运算等。

⑤ “实用命令”——延时(毫秒,微秒),随机数(1024 以内),上电运行时间,串口输出,数字映射, I²C 读、写等。

(4) ArduBlock 程序实例。

以前面的 LED 闪灯为例,只要从“控制”类模块中拖曳出“主程序 do”,从“引脚”类模块中拖曳出“设定数字引脚值”,从“实用命令”类模块中拖曳出“微秒延迟”,并按图 1-9 右边编程区所示程序修改、嵌合,将其“下载到 Arduino”,再进行“验证”“上传”,就可以使 Arduino 板上的 LED 闪烁起来。

2) Mixly (米思齐)

Mixly 是一款免费开源的图形化编程软件,支持包括 Arduino、micro:bit、ESP32 等开发板。Mixly 软件具有丰富的传感器模块和第三方库,同时支持用户的定制化开发,方便用户编写各种从简单到复杂的应用程序。Mixly 可将编写完成的程序直接上载到 Arduino 开发板。Mixly 的界面如图 1-10 所示。

Mixly 界面主要包括模块区、编程区、代码区和编译区四部分。Mixly 的部分模块功能如表 1-3 所示。

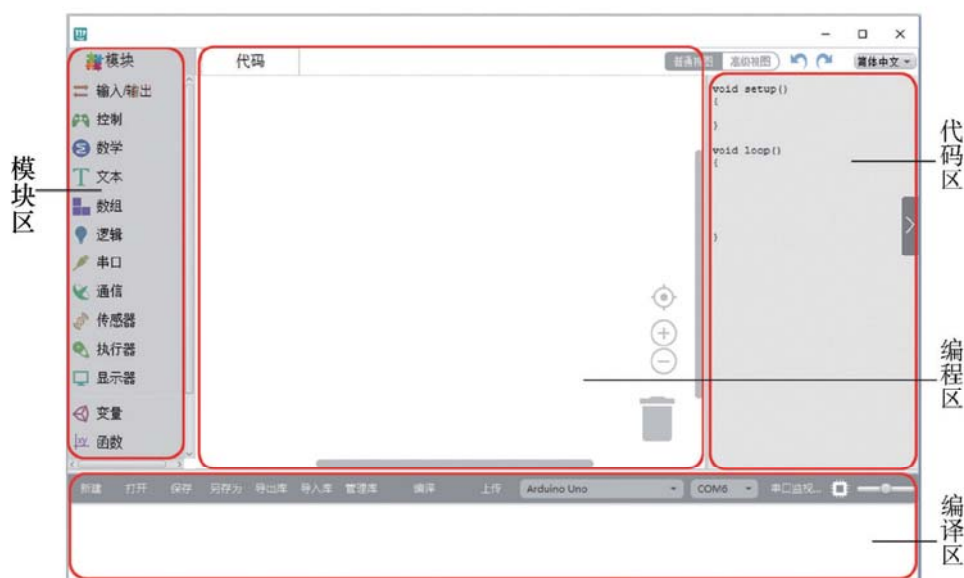


图 1-10 Mixly 的界面

表 1-3 Mixly 的部分模块及其功能

模块类名称	功 能
输入 / 输出	用于编写输入、输出指令的模块，包括数字引脚的输入 / 输出、模拟引脚的输入 / 输出、中断控制、脉冲长度设置等
控制	用于控制程序结构的模块，包括初始化、延迟、条件执行、循环执行、系统运行时间、定时器等
数学	用于数学运算的模块，包括数字映射、数字约束、数字运算、取整、三角函数等
文本	用于处理文本字符的模块，包括字符连接、字符截取、字符比较、字符与数字转换等
数组	用于编写一维数组的模块，包括数组初始化、数组创建、数组项赋值和取值等
逻辑	用于逻辑运算的模块，包括比较、判断、逻辑值等
串口	用于串行通信的模块，包括串口读 / 写、串口中断等

以前面的 LED 闪灯为例，可从“输入 / 输出”模块区中拖曳出“数字输出”模块，并设置其管脚和值，从“控制”模块区中拖曳出“延时”模块，并输入时间，如图 1-11 所示，然后将其“上传”，就可以使 Arduino 板上的 LED 闪烁起来。



图 1-11 Mixly 闪灯实验代码

【实践】

安装并使用上述 Arduino 编程软件，感受不同软件的编程风格，并选择其中一个进行编程实验。

● 项目实施

各小组根据项目选题及拟定的项目方案，结合本节所学知识，了解开源硬件及 Arduino 平台，熟悉各针脚的位置，选择一个编程工具，配置开发平台，进行 Arduino 小实验。

第二节 初试身手——LED

Arduino 是一种开展创客活动的辅助工具，它能帮助我们实现对真实问题的快捷解决。从本节开始，我们将学习用 ArduBlock 和 Arduino 进行各种科学实验。

做一个在电脑屏幕上显示的虚拟红绿灯动画，方法多种多样，也很容易实现。然而，若想做一现实中的红绿灯或流水灯，似乎就不那么容易，一是要找到合适的元器件，二是要考虑怎么让红绿灯能够自动交替闪亮……

现在，有了 Arduino，要做一个现实的红绿灯，就变得简单了。

一、制作演示用的红绿灯

作为演示用的简单红绿灯，可直接将红色 LED 和绿色 LED 接在 Arduino 板上临时进行，如图 1-12 所示。

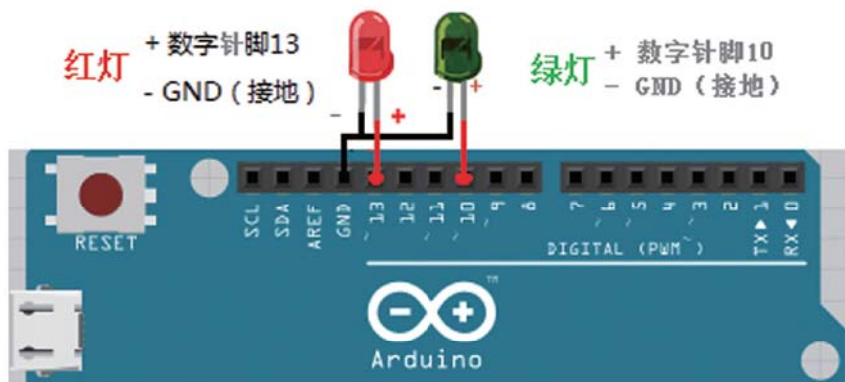


图 1-12 简单红绿灯接线图

具体的程序实现也比较简单，只要将数字引脚 13 设置为高电平，5s（间隔时间可自定）后再设置为低电平，然后将数字引脚 10 设置为高电平，5s 后再设置为低电平，如此循环，如图 1-13 所示。

注意：图 1-13 中的“主程序 do”是一种常用的循环结构，对应于 Arduino IDE 中的“void loop() {}”，只要是针对 Arduino 的程序，就必定会用到这种结构。



图 1-13 简单红绿灯程序

● 探究活动

【观察】

1. 将程序下载到 Arduino，看看红绿灯是怎样闪烁的？修改延迟时间会发生什么现象？
2. 观察马路上实际红绿灯的运行过程，总结并提取其运行规律。

二、制作实际红绿灯

制作实际的红绿灯，需要考虑实际元器件的参数，还要加一黄色闪灯。

1. 实际红绿灯电路图

根据中学物理知识，红绿灯所需元器件包括 LED 灯、电阻 R 、电源 E 、控制开关、导线等，其构成电路如图 1-14 所示。由图可知，只要交替不停地按压、释放红、黄、绿三路控制开关，就可实现红绿灯功能。

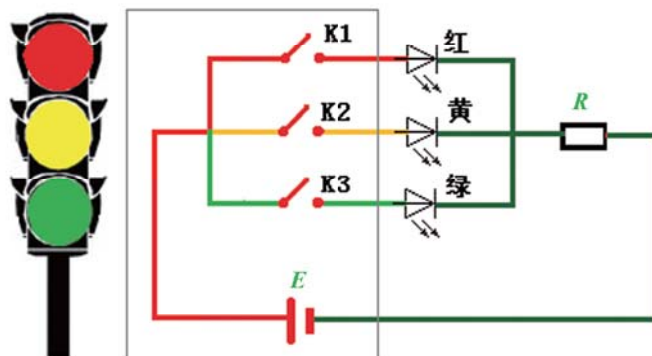


图 1-14 红绿灯所需元器件及连接电路图

2. 红绿灯闪烁控制

手动操作实现红绿灯闪烁是不符合实际需求的。这里，我们让 Arduino 来充当闪烁控制及电源的角色：先让 Arduino 给红色 LED 支路提供电压（高电平 HIGH），等待 5s 后断电（低电平 LOW）；接着让黄色 LED 支路闪烁 4 次（每次给黄色 LED 高电平 0.5s，低电平 0.5s）；然后给绿色 LED 支路提供高电平，等待 5s 后断电；再接着让黄色 LED 闪烁 4 次，如此循环。

3. 实施过程

（1）利用扩展板连线：在 Arduino 扩展板（就是在保留 Arduino 板基本功能针脚的基础上另外增加一块面包板，扩展引脚的数量以方便使用）上，按图 1-15 所示电路原理连接好红、黄、绿 LED（注意正负极）和电阻（这里采用 220Ω 的色环电阻），从 D10（数字针脚 10）、D11、D12 各引出导线分别连接红、黄、绿 LED 的 + 极，如图 1-15 所示。

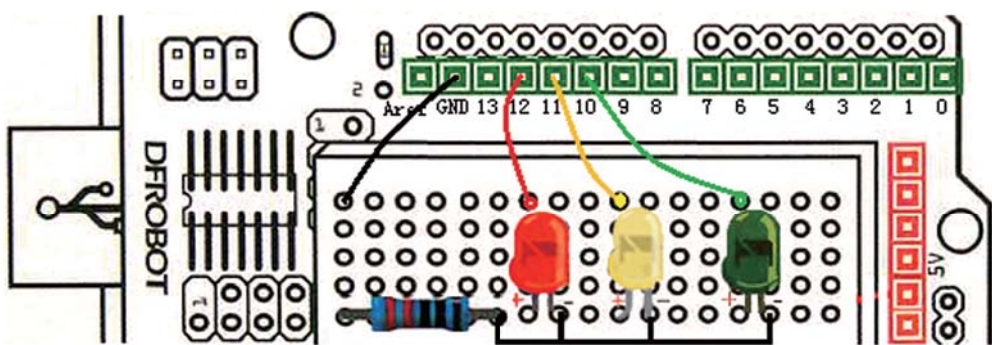


图 1-15 红绿灯的连线

（2）连接 Arduino：用 USB 线将 Arduino 板与电脑连接，注意检查 Arduino 板与电脑连接的端口号。

（3）编写 ArduBlock 程序：利用 ArduBlock 的相关模块，编写实用的红绿灯闪烁程序，如图 1-16 所示。



图 1-16 红绿灯闪烁程序



注意：如果仅仅是实现红绿灯闪烁的程序，则只要将图 1-16 中的代码进行克隆并修改，即将“红灯亮 5s 熄灭，黄灯闪 4s”克隆后改为对应的绿灯针脚，就可达到目的。但我们可考虑在此基础上进行一个改进，在程序中引入一个变量 zj，并且通过判断和给变量赋值使其在 10 和 12 这两个数字间跳转（即在红灯 10 针脚和绿灯 12 针脚间跳转），使得程序代码不是简单的重复，且可以进一步缩短代码长度，同时也引入一些程序设计知识，供阅读思考。改进后的程序如图 1-17 所示。



图 1-17 改进后的红绿灯程序

【思考】

如果需要把 LED 接到其他针脚，硬件和软件应该如何修改？

【拓展】

一、LED

LED 是一种二极管，也叫发光二极管，加电后能发出不同颜色和亮度的光线，包括光谱中的紫外线和红外线。

仔细观察 LED 会发现，LED 的针脚长度不同，长针脚为 +，短针脚为 -，电源给 LED 加电时若正负接反，则灯不会亮。

单个小功率 LED，颜色不同则其要求的电压也不同。红 / 黄：一般为 1.8 ~ 2.1V，白 / 绿 / 蓝：一般为 3.0 ~ 3.6V。

二、面包板

面包板是一种可重复使用的非焊接元件，常用于制作电子线路原型或者线路设计。面包板表面是打孔的塑料，底部（下表面）有金属条与所打的孔铆接，中间设有夹片，可以实现插上

即可导通（无须焊接）的作用，如图 1-18 所示。

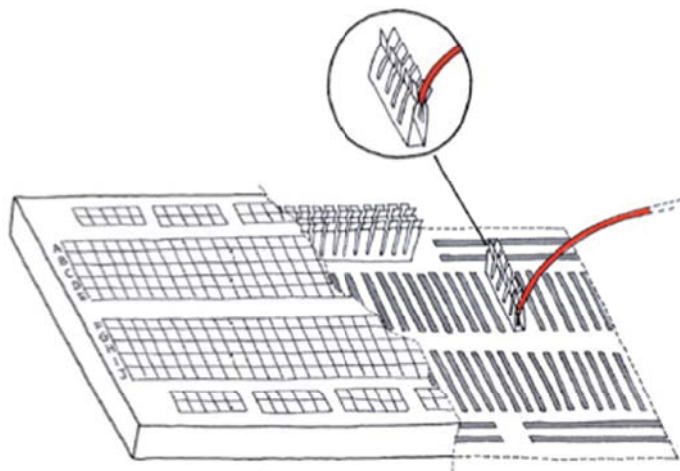


图 1-18 面包板示意图

面包板内部有横向和纵向两个方向之分，其中上半部分的孔为横向连通，中间部分的孔为纵向连通，同时还需要特别注意各个块之间的连通情况，如图 1-19 所示。

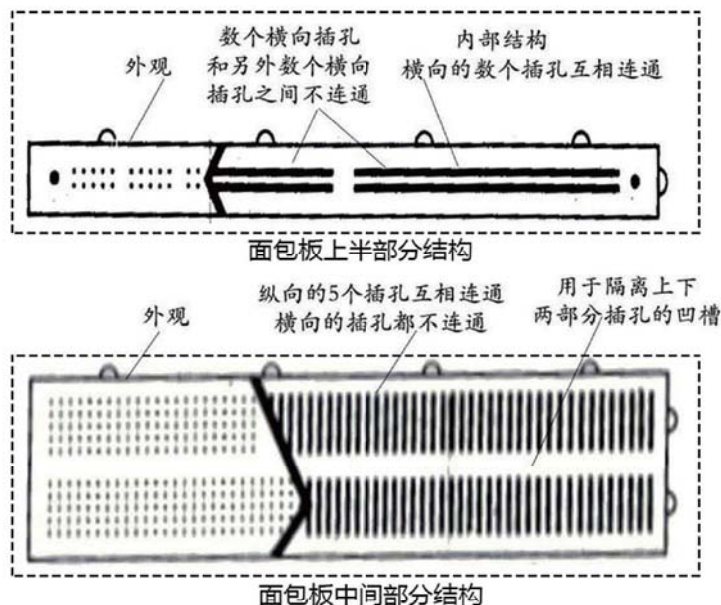


图 1-19 面包板内部结构图

三、色环电阻

色环电阻是应用于各种电子设备最多的电阻类型，使用中，应正确读出其阻值。表 1-4 为色环电阻中各色环的含义，其中“有效数字”一栏对应于三环电阻的第 1 环（四环电阻的第 1、2 环，五环电阻的第 1、2、3 环），“数量级”一栏对应于三、四、五环电阻的倒数第 2 环，“允许偏差”一栏对应于三、四、五环电阻的倒数第 1 环，只有“温度关系”一栏是专指六色环电阻第 6 环所表达的电阻意义。据此，可以读出一个色环电阻的阻值和允许偏差等信息。

表 1-4 色环电阻各环含义对照表

含义	银	金	黑	棕	红	橙	黄	绿	蓝	紫	灰	白	无
有效数字	—	—	0	1	2	3	4	5	6	7	8	9	—
数量级	10^{-2}	10^{-1}	10^0	10^1	10^2	10^3	10^4	10^5	10^6	10^7	10^8	10^9	—
允许偏差 (%)	± 10	± 5	—	± 1	± 2	—	—	± 0.5	± 0.25	± 0.1	± 0.05	—	± 20
温度关系 / ($\times 10^3 / ^\circ\text{C}$)	—	—	—	100	50	15	25	—	10	5	—	1	—

例 1 有一色环电阻（共 4 环），如图 1-20 所示。第 1 环为棕色，表示数字 1；第 2 环为黑色，表示数字 0；第 3 环为红色，表示数量级 100；第 4 环为银色，表示允许偏差 ± 10 。综合起来，电阻值为 10×100 ，即 1k，允许偏差 $\pm 10\%$ 。



图 1-20 色环电阻阻值的读取

例 2 有一五环电阻，难以确定哪个是第 1 环，哪个是第 5 环。若顺读为棕、黑、黑、黄、棕，则其值为 $100 \times 10^4 = 10^6 \Omega$ ，误差为 1%；若逆读为棕、黄、黑、黑、棕，则其值为 $140 \times 1 = 140 \Omega$ ，误差为 1%。显然，按照逆读得出的电阻值和误差值，在实际生产中是不存在的。因此，判读色环电阻也要从实际出发。

在实践中，若发现色环电阻难以准确辨认，则可以用伏安法测量（测出通过电阻的电压和电流，根据欧姆定律，电阻 = 电压 / 电流），或者直接用图 1-21 所示万用表进行测量。



图 1-21 数字式、指针式万用表

● 项目实施

各小组根据项目选题及拟定的项目方案，结合本节所学知识，熟悉 LED 的使用方法，并将其应用到自选项目中，增加项目的灯光效果，提高可观赏性。

第三节 感 光 灯

每到傍晚，路灯就自动点亮了，不仅让城市的夜晚充满了光明，而且更加节能。那么，路灯是怎样自动点亮的？

下面来了解光敏二极管，并学会制作感光灯。

一、光敏二极管

光敏二极管又叫光电二极管（photodiode），是一种能够将光信号转换成电流或者电压信号的光探测器。

光敏二极管的管芯是一个具有光敏特征的 PN 结，对光的变化非常敏感，具有单向导电性，而且光强度不同的时候会改变电学特性，因此，可以利用光照强弱来改变电路中的电流。

由于光敏二极管的单向导电性，因而连接到电路中时应注意其正负极，出厂时设定长管脚为正极，短管脚为负极。实物及元件符号如图 1-22 所示。



图 1-22 光敏二极管及其符号

光敏二极管能将光能转换成电能，在电路中产生光电流。对于光敏二极管所在的电路，有光照射，电路中的电流就增大，这就等效于其所在电路中的电阻减少，因此，光敏二极管也可叫光敏电阻，其特性是光照强度越大，其电阻越小，在电路中分担的电压就越小。

二、简易感光灯

尽管在光敏二极管中产生的光电流只是微电流，其两端的电压变化也很微小，但只要能够

捕捉到这种变化并加以利用，就可以对与光照相关的设备进行自动控制。简易感光灯就是利用这一原理实现光照控制的。

1. 实验原理

设计感光灯，需要两个电路，一个是灯电路（可以根据实际情况加一电阻分压），另一个是感光电路，如图 1-23 所示。

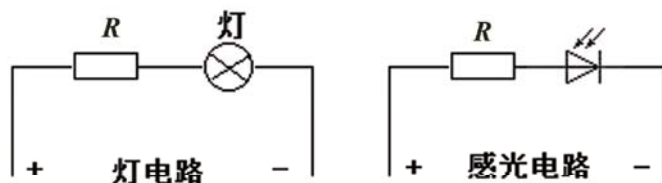


图 1-23 感光灯的两个电路

感光电路由电阻和光敏二极管构成（电阻分压，以保护光敏二极管），相当于一个光控开关，以控制灯电路的开与关。当有光照时，让灯的电路处于常关状态；当无光照或光照强度减弱到一定程度时，即光敏二极管两端的电压升高到某一临界值，便可以让灯电路自动接通。

2. 电路连接

作为实验，两个电路的电压我们都从 Arduino 板获取，如图 1-24 所示。

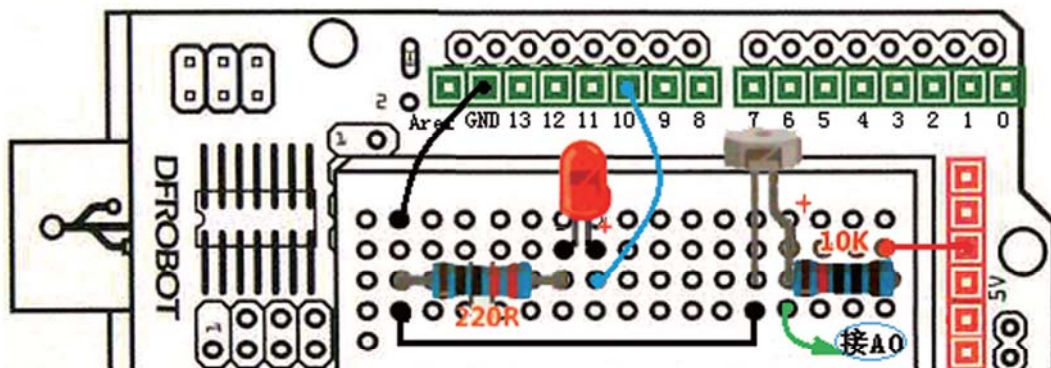


图 1-24 感光灯电路连接

LED 灯与电阻串联接在数字引脚 10 和 GND（接地）；光敏二极管与 $10k\Omega$ 电阻串联，电阻接 5V，光敏二极管负极接地，再从光敏二极管与电阻连接点引出连接线到模拟引脚 A0 作为模拟输入。

电路连接实验需要的元器件有：Arduino 板，光敏二极管 1 只， $10k\Omega$ 电阻 1 只， 220Ω 电阻 1 只，LED 灯 1 只。

3. 编写程序

灯电路的电压从引脚 10 获取，而引脚 10 是否为高电平则由感光电路中模拟引脚 A0 的值决定。若引脚 A0 的电压模拟值大于 1017，则设定数字引脚 10 为高电平；否则设定数字引脚 10 为低电

平。参考程序如图 1-25 所示。

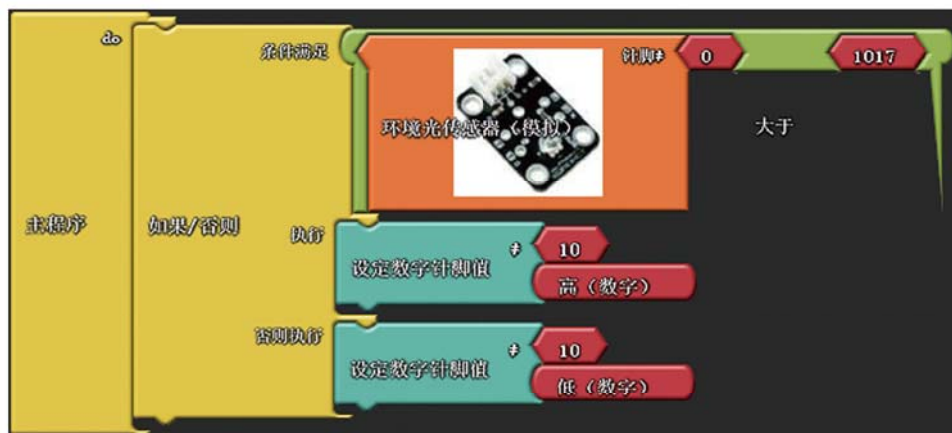


图 1-25 感光灯 ArduBlock 程序

注意：程序中的“针脚 0”是指模拟针脚 A0；当光电二极管被完全挡光时，从 A0 针脚获得的电压模拟值为 1023 (5V)，而有光照射时约为 1015 或更小，故 1017 是设定的一个临界值（这个数值可通过测试调整，模拟值与电压的映射关系为 $1023 \rightarrow 5V$, $0 \rightarrow 0V$ ）。

● 探究活动

【实验】

将程序下载到 Arduino，用手遮挡光敏二极管周围的光线，看看 LED 灯是怎样亮灭的？若将 1017 改为 1012 或 1022 会发生什么现象？

三、传感器的制作

有关电子类的产品常常是通过获取输入信号来控制输出，而输入信号则需要通过各种传感器来获取。下面介绍环境光传感器和温度传感器的制作。

1. 环境光传感器

环境光传感器，实际上就是上例中感光电路的变形。只要将整个感光电路包装成一体，并引出三个针脚，就是一个环境光传感器，如图 1-26 所示。

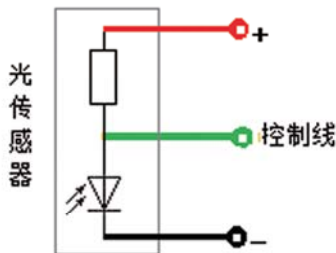


图 1-26 环境光传感器的电路结构



2. 温度传感器

温度传感器也可以根据类似的方法进行制作。

1) 温敏电阻

温敏电阻也叫热敏电阻，如图 1-27 所示，其由半导体材料烧结而成，是一种对温度敏感的元件，在不同的温度下会表现出不同的电阻值。温敏电阻分为两种，一种是正温敏电阻（PTC），电阻值随温度的升高而增大；另一种是负温敏电阻（NTC），在温度越高时电阻值越低。

2) 感温电路

感温电路由一个温敏电阻和一个普通电阻构成，若从感温电路引出三个针脚，就可以成为一个温度传感器，如图 1-28 所示。



图 1-27 热敏电阻实物图

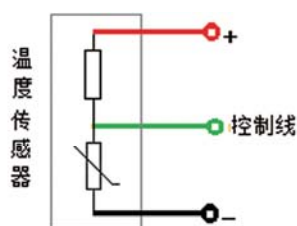


图 1-28 温度传感器的电路结构

3) 几种常见的温度传感器

温度传感器按用途和其探头的不同样式，可分为探针式、螺栓式、密封罐式几种类型，实验用的温度传感器常常用 LM35，如图 1-29 所示。



图 1-29 温度传感器的类型

除了环境光传感器、温度传感器外，还有各种用来获取输入信号的传感器，如压力传感器、pH 传感器、氧气传感器等。各种传感器的设计制作都是根据其感知元件的基本功能展开的，这些感知元件包括热敏元件、光敏元件、气敏元件、力敏元件、磁敏元件、湿敏元件、声敏元件、放射线敏感元件、色敏元件和味敏元件十大类。传感器的存在和发展，让物理世界中的物体有了触觉、味觉和嗅觉等感官。

【调查】

利用网络搜索常用的传感器，查看其参数和使用方法，判断并选择其是否符合自选项目中