



普通高中教科书

信息技术

选择性必修6

开源硬件项目设计

普通高中教科书

信息技术

选择性必修6

开源硬件项目设计

闫寒冰 主编

主 编：闫寒冰

副 主 编：赵 健 魏雄鹰

本册主编：谢作如

编写人员（按姓氏笔画排列）：

林仁蛟 章璋明 谢作如

信息技术作为当今先进生产力的代表，已经成为我国经济发展的重要支柱和建设网络强国的战略支撑。在这样的大背景下，教育部全面修订并颁布了《普通高中信息技术课程标准（2017年版）》，为这门课程设定了与新时代相符的育人目标：帮助学生掌握信息技术基础知识与技能、增强信息意识、发展计算思维、提高数字化学习与创新能力、树立正确的信息社会价值观。

本套教材依据《普通高中信息技术课程标准（2017年版）》编写，包括两本必修教材《数据与计算》《信息系统与社会》，六本选择性必修教材《数据与数据结构》《网络基础》《数据管理与分析》《人工智能初步》《三维设计与创意》《开源硬件项目设计》，两本选修教材《算法初步》《移动应用设计》。

本套教材的编写组汇集了来自信息技术、课程与教学、教育技术等领域的高校学者与教学一线专家。编者们通力合作，从课程内容、教材体例、技术选择、教学方法、学习方法等方面精心打磨，期待以最专业的样态帮助学生达到课程预期的育人目标。

具体而言，本套教材体现了如下特点：

1. 体例上——为核心素养的培养创造空间和条件：将核心学习内容与支持学习的方法有机融合在一起，支持学生在自主、合作、探究的学习情境下发展核心素养。

2. 内容上——体现概念、内容与方法的精准与专业：在增强教材可读性的同时，精炼提升综合素养所必需的核心内容，强调所有概念、内容与方法的精准与专业。

3. 活动上——着力提升学生的高级思维能力：精心设计与布局教材中的练习、思考、讨论、实践与项目学习，追求对高级思维能力的培养。

4. 案例上——体现信息科技的多层需求与多维格局：把案例的呈现作为开阔视野的重要手段，帮助学生理解信息技术对于社会发展所具有的价值与意义。

5. 技术上——引领学生拓宽视野与发展思维：将每种具体应用软件都作为解决某些问题的一条路径来看待，期待学生通过具体的技术操作体验，理解其背后的原理与格局、特点与局限，拓宽视野、发展思维。



本册教材为选择性必修《开源硬件项目设计》，针对学生个性化发展的需要，按照开源硬件项目设计流程而设置。基于开源硬件的项目设计与开发是典型的STEAM(科学、技术、工程、人文艺术与数学)教育，它有益于激发学生创新的兴趣、培养学生动手实践的能力。通过本教材的学习，希望同学们能搜索并利用开源硬件及相关资料，体验作品的创意、设计、制作、测试、运行的完整过程，初步形成以信息技术学科方法观察事物和问题求解的能力，提升计算思维与创新能力。

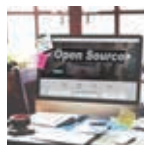
就教材本身所讲述的知识内容而言，我们相信，如果同学们潜心自学就可以基本掌握。但“知识内容”只是发展信息技术核心素养的基础部分，所以，我们希望同学们不要仅满足于对具体知识与具体技术的掌握，还要重视教材中的各类学习活动，与老师和学友一起，更多地去创造、研究、解决问题、制作、交流、合作和评价，唯有如此，同学们才能藉由这门课程的学习全面地提升信息素养，增强在信息社会的适应力与创造力，为实现中华民族伟大复兴的宏伟目标做出更大贡献！

本册教材在编写过程中得到了各方面的大力支持。北京大学计算机系李晓明教授、浙江大学计算机学院卜佳俊教授和翁恺教授、北京航空航天大学欧阳元新副教授在百忙之中对书稿内容进行了审阅。深圳开放创新实验室的李大维、上海蘑菇云的叶琛为书中有关开源运动发展史方面的内容提供了图文资料。李琦、李曙强、陈基伟、夏正仁等多位高中教师为本书的撰写提供了建议，并给出了宝贵的修改意见。

由于水平有限，本书可能还存在不足之处。希望大家在教材使用过程中，能够及时将意见和建议反馈给我们，对此，我们深表谢意。

第一章 开源思想与开源硬件

- 1.1 开源思想与开源运动 4
- 1.2 开源硬件的发展及特征 8
- 1.3 开源硬件的经典案例 13



第二章 开源硬件项目的开发流程

- 2.1 开源硬件项目的案例剖析 22
- 2.2 项目的开发过程与方法 27



第三章 开源硬件项目的开发基础

- 3.1 常见的开源硬件 42
- 3.2 开源硬件编程基础 45
- 3.3 传感器的接入 62
- 3.4 控制信号的输出 69
- 3.5 无线通信的实现 85



第四章 开源硬件项目的开发实践

- 4.1 人机互动的项目开发 98
- 4.2 多机通信的项目开发 106
- 4.3 基于物联网的项目开发 114



第五章 开源硬件项目的发布与维护

5.1 开源硬件项目的发布	128
5.2 开源硬件项目的维护	136



开源思想与开源硬件



创新是一个民族进步的灵魂，是一个国家兴旺发达的不竭源泉。在第三次浪潮——信息化浪潮到来之际，创新日益凸显其重要性，创新驱动发展成为国家面向未来的一项重大战略。“苟日新，日日新，又日新”，创新不是靠一时的顿悟，而是建立在继承基础上的飞跃，正如人类的文明一样，是一代一代人的智慧结晶。在信息化时代，信息共享变得越来越方便，人们参与创新的机会越来越多，创新能力也不断提高。开源运动的兴起与开源硬件的流行，正是知识共享与创新需求推动下的一种发展趋势和方向。



问题与挑战

- 很多发达国家都建立了完善的知识产权保护制度，如英国早在1709年便制定了世界第一部著作权法——《安娜女王法令》，1883年出台的保护工业产权的《巴黎公约》，被视为英国专利法和商标法的基础；德国出台了《商标法》《著作权法》《专利法》等，成立德国联邦专利法院，对世界影响深远。这些针对专利和著作权的法律法令的制定是为了保护发明者或著作者的利益，那么开源软件和开源硬件的设计者为什么愿意放弃这种保护，而免费开放自己的作品？

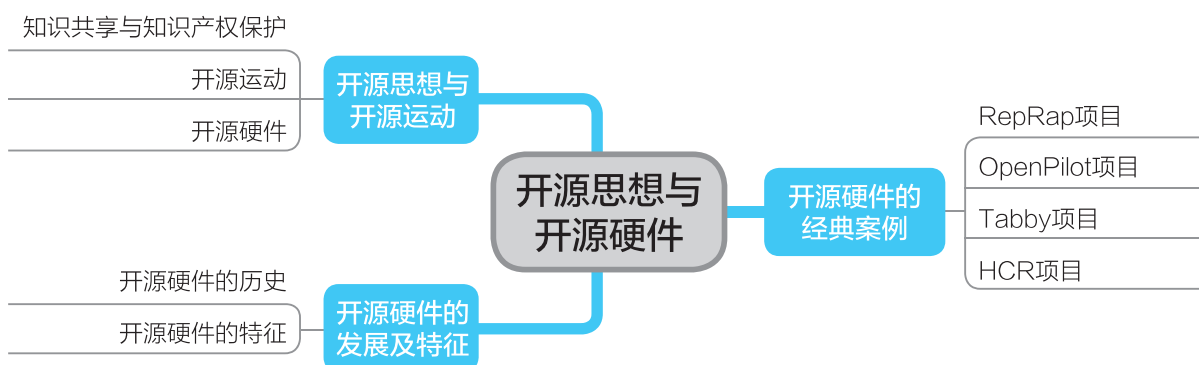
- **MakerBot**是一个著名的开源3D打印机项目。有人利用这一开源项目的资源，仿造了一款名叫**TangiBot**的3D打印机，进行低价出售，来抢占**MakerBot**的市场。这种“暴力克隆”的行为引发了3D打印社区的广泛争论，争论涉及道德、法律、开源许可协议等开源硬件的各个层面。开源硬件有没有办法防止大规模的设计剽窃和产品盗版行为？“暴力克隆”行为是否会阻碍开源运动的发展？

- 如果你是一名创客，利用其他的开源软件和硬件资源设计了一个新产品，获得了很高的评价，你会不会为自己的产品申请专利，以此来保护自己的利益？

学习目标

1. 了解开源思想、开源运动与开源硬件的发展历程。
2. 基于实例分析，认识开源硬件的特征。
3. 理解开源硬件对信息技术创新的意义和价值，理解保护知识产权的意义。

内容总览



1.1

开源思想与开源运动

开源（Open Source）即开放源代码，指软件发布的一种方式。事实上，开源不仅仅代表软件源代码的开放，更意味着自由、共享和充分利用资源。开源的思想最初起源于人们对知识合理共享的需要。从对传统的著作权归属问题的思考、争议和处理，逐渐发展到计算机领域的开源运动和开源文化，开源已经成为当今信息技术发展的重要特征，体现了互联网时代中知识的共享，是技术进步、经济变革和文化发展的基础力量。

1.1.1 知识共享与知识产权保护

2017年10月，著名物理学家霍金为了响应母校剑桥大学论文电子化的号召，向社会大众公开了自己的博士论文。同时，剑桥大学也要求今后所有的剑桥毕业生必须提供其论文的电子版并向公众公开。这一事件再次引发了公众对知识共享与知识产权保护话题的热议。

人类社会发展历程中的每一次重大理论突破和技术变革，都不是一个人或一代人的努力结果，而是一代又一代人积累的知识沉淀。每一个后续的创新和发明都建立在初始成果或前代成果的基础上，是一个迭代创新的过程。为了鼓励更多的人参与创造发明并愿意公开科研成果，人们设计了专利制度。专利制度强调“技术公开、利益独享”，一定程度上激励了技术创新。但利益垄断的形式同时也影响了迭代创新的速度和质量，越来越多的人开始反思专利制度是否需要更新。霍金免费公开博士论文，就是希望对知识产权的保护不应该形成知识壁垒。

然而，在知识产权的保护和无偿公开之间，始终存在着两难的选择。传统的知识产权，包括著作权和发明专利，其所属通常为两种极端，一端是“保留所有权利”，另一端则是“不保留任何权利”。一般而言，前者有利于维护知识产权，保障作者和发明者的权益，但同时也可能妨碍成果在更大范围的共享和共享基础上的连续迭代与创新；后者即著作权和发明专利的无偿开放，则有利于大众在作者和发明者的成果基础上，无成本或低成本地实施共享或进一步改良和创新，促进行业的快速发展，但是没有著作权和专利收益显然可能抑制作者和发明者的积极性，还可能使其在再创作过程中经费短缺从而影响一个行业的长远发展。

这样的矛盾在计算机领域的发展中同样存在。20世纪70至80年代计算机行业发展的早期，程序员和开发者为了改进软件设计开始共享软件代码，相互学习已经显露端倪，但是当时非开放的商业化软件仍然占据主流地位。1983年起，受黑客（Hacker，这里指追求完美、酷爱技术的编程高手和电子爱好者）文化和学术潮流的影响，自由软件（Free Software）运动蓬勃兴起。这场运动使用户获得自由学习、运行和修改软件并对软件进行再发布的权利。开源运动正是在此基础上发展而来。

拓展链接

自由软件

自由软件的兴起缘自Unix操作系统的闭源。Unix由美国电话电报公司(AT&T)旗下的贝尔实验室和多所大学联合开发,研究人员彼此之间可以自由分享Unix代码。1982年,AT&T被分解后,Unix的代码开始闭源,但更多的程序员希望能继续自由地分享软件代码。1983年,来自麻省理工学院人工智能实验室的研究员理查德·斯托曼(Richard Stallman)发起GNU(Gnu's Not Unix)项目,并在《Dr. Dobbs's Journal of Software Tools》杂志上发表了GNU宣言,阐述了自由软件的要旨,提出要“重现当年软件界合作互助的团结精神”。几个月后,他创建了自由软件基金会。

问题与讨论

蒸汽机的发明,是人类第一次工业革命的重要标志。詹姆斯·瓦特为此申请了长达30年的专利保护。在瓦特专利保护期内,蒸汽机的燃料效率提升不大,产量也不高。专利失效后,蒸汽机在生产 and 效率方面都有了爆炸性发展,涌现了一批改良后的机器。那么,专利保护是否阻碍了蒸汽机的进一步创新?请查阅资料,围绕专利和创新,谈谈各自的观点,并给出相应的理由。

1.1.2 开源运动

1998年1月,网景公司(Netscape)发布了一则声明,称其将网络浏览器Navigator的源代码完全免费开放,“开源”一词开始正式出现。随后,在美国加利福尼亚州帕罗奥图市召开的自由软件高峰会议上,一批很有影响力的程序员参与了会议。他们通过投票,确定了用“开源”的概念来取代自由软件。同年11月,“开放源代码促进会”(Open Source Initiative,简称OSI)正式成立,并注册了如图1.1.1所示的徽标。这个组织以倡导和推广开源理念为己任,开启了一个开源运动的时代。



图1.1.1 OSI的徽标

1. 开源许可协议

在此之前,虽然自由软件的理念得到广泛拥护,但是英文“自由”(free)同时兼有“免费”之意,大大制约了企业在软件源代码共享和合作领域的积极性。为了保障拥有软件开发能力的企业或者个人,在分享的同时也能合理获取收益,“开源”的定义中除了



“开放”以外，还包含了一个被称之为“开源许可协议”（License，也称为许可证）的规定。这是开源软件和自由软件的最大区别。

开源许可协议，指的是针对计算机软件和其他产品的源代码、设计图等在一一定的约束条款和条件下授权使用、更新和共享的许可。该许可协议允许终端用户和商业企业针对自己的定制需求、兴趣或者要解决的问题查看和修改源代码、设计图。同时，开源许可协议保护软件原创者的专利，如要求在代码中保留作者名字和著作权声明，或者要求再发布时必须遵循同一个许可协议。

换言之，开源许可协议既对使用源代码的后继者有限制，也对发布源代码的开发者有保障。其中最基本的限制就是规定使用人和修改人必须承认发起人的著作权和所有参与人的贡献，此外，任何人都拥有自由复制、修改、使用这些源代码的权利，开源软件不得设置针对任何人或团体领域的限制，不得限制开源软件的商业使用等。许可协议有不同的类型，对用户的权限规定也不尽相同。如 GPL（GNU General Public License，GNU 通用公共许可协议），允许用户免费使用、引用、修改源代码，开源衍生代码，但不允许修改后或衍生的代码作为闭源的商业软件进行发布和销售。

开源许可协议对于开源运动的发展具有重要意义。通过一种协议约定的方式，将知识共享中所涉及的权利与义务、开放与约束等道德和伦理问题，纳入对开发者和使用者具有双重权利保护作用的法制范畴，为源代码的共享提供了一个可持续发展的机制，这样，拥有源代码的大公司和开发者参与共享的积极性大大提高。

拓展链接

开源软件 GPL 许可证的“反垄断”诉讼

2006年3月20日，美国联邦法官驳回了一起针对开源软件 GPL 许可证的“反垄断”诉讼。法官裁定：按照 GPL 许可证，某些软件可以复制、修改、重新发行而不违反软件的著作权保护，GPL 允许的这种行为，是鼓励而不是阻碍自由竞争和计算机操作系统的发行，它使消费者直接受惠，包括得到更低的价格，能更好地掌握软件或做更多的创新。开源软件的法律地位在司法实践中得以确立。

2. 开源社区

开源运动的另一个重要标志是开源社区的形成。开源的技术，吸引了许多志趣相同的人士通过特定的社交平台分享思路、讨论问题、发布成果等。如拥有众多开源软件的 SourceForge 社区、GitHub 和开源中国社区等，都是其中的优秀代表。这些社区通过论坛、博客跟帖、照片与视频分享、新闻推送等社交工具，将个人发明和个人制造转变成共同创作行为，从而形成了以“自由、分享和开放”为根本特征的开源文化。

近几年，各种开源社区的价值也随着新技术的普及而不断凸显出来。以开源社区中最为热门的 OpenStack（一个开源的云计算管理平台项目）为例，它几乎得到所有主流软硬

件厂商的支持，2017年私有云市场的占有率达到了60%。微软、英特尔、IBM等企业都纷纷发布开源软件，建立相关的开源社区。

开源运动和开源文化有力地促进了全球经济的创新和发展。很多公司和个人得以在共享的源代码基础上进行开发和更新，这不仅推动了整个IT产业从封闭走向开放，还大大降低了社会创新的门槛和成本，为大众创业和万众创新提供了平等的机会。

问题与讨论

在开源运动中，关于著作权保护的边界在哪里，一直争议不断。各国法律对著作权保护的边界认定，将直接影响开源理念的发展。美国有几个著名的司法案例，其判决结果成为开源运动史上的里程碑。

(1) 原告罗伯特·雅各布森 (Robert Jacobsen) 团队开发了名为“DecpderPro”的模型火车控制软件，在“Artistic许可证”下放到一个著名开源软件网站供公众分享。被告卡兹尔及卡曼公司 (Katzner/Kamind) 程序员利用该软件的部分源代码，开发了一种类似该软件的产品并予以销售。2008年，原告起诉被告的侵权案件被美国联邦上诉法院判决原告胜诉。

(2) 2009年甲骨文公司收购了拥有开源软件Java API (应用程序编程接口) 的升阳公司后，于2010年起诉谷歌公司在其开源软件安卓系统中，复制了JAVA的37个API，并以侵犯著作权诉求高达97亿美元的赔偿。2016年，谷歌在经历了多次审理后，最终被判决其行为构成合理使用而胜诉，原告甲骨文败诉。

同样起诉开源软件侵权，两个原告一个胜诉、一个败诉。你认为这两个司法案例的审理结果是否合理？请查阅更多资料，谈谈你的观点。

1.1.3 开源硬件

“开源”尽管发起于软件的源代码开放，然而随着开源运动的发展，硬件“开源”也逐步成为开源文化的重要组成部分。《创客：新工业革命》的作者克里斯·安德森 (Chris Anderson) 指出：“开源意味着开放一切，电子元件、软件、实体设计、文档，甚至是标识。”开源硬件 (Open Source Hardware) 是指在遵循开源许可协议的前提下，将计算机和电子硬件，包括电路原理图、材料清单、设计图等，完全以开源的方式授权用户自由地使用、分享和再创作。开源硬件的驱动程序也通常选择开源软件形式进行发布。

开源硬件的上述定义是由开源运动的发起者之一、著名的开源项目Busybox的创始人布鲁斯·佩伦斯 (Bruce Perens) 创建的。开源硬件的本质就是共享一个硬件的设计文件以方便他人进行修改或据其制作硬件，用户可以在已有开源硬件项目的基础上进行迭代创意，创造出新的开源硬件。基于开源硬件开发的产品正日益进入寻常百姓的生活，例如广泛应用于各领域的无人机，就得益于开源硬件项目OpenPilot的发展；在创客空间里随处可见的3D打印机，也是得益于RepRap项目的开源。

1.2

开源硬件的发展及特征

作为开源文化的一部分，开源硬件受开源软件的启发而确立，并扩展了开源的概念，但其历史却比开源软件还早，可追溯到集成电路发展初期。

1.2.1 开源硬件的历史

早在20世纪60至70年代，尽管还没有形成“开源”的概念，但硬件却都是开源的模式，如打印机、计算机的整个设计原理图都是公开的。后来出于贸易壁垒、技术壁垒、著作权等问题的考虑，很多公司如三星、苹果等，开始选择了闭源。

1. 开源硬件的起源

开源硬件的雏形出现于1997年，发起者之一布鲁斯·佩伦斯是Linux系统的程序员，也是无线电爱好者。他发布了一个开放硬件认证程序，将“开放硬件”（Open Hardware）注册为徽标，如图1.2.1所示，并注册了openhardware.org的域名专用于该认证项目。



图1.2.1 开源硬件的徽标

“开放硬件认证计划”是布鲁斯·佩伦斯将开源文化从软件向硬件延伸的最早尝试。硬件开发者通过开放硬件认证程序可以将自己的产品自行认证为开源硬件。取得认证后，开发者可以把开源硬件的徽标印在产品的包装上，并在广告里提及产品已取得开源认证。这个“开放硬件认证计划”的目的是为了让硬件制造商能够自行认证他们的开放硬件产品，允许用户能为设备更换操作系统，同时确保即使硬件制造商倒闭，仍有人能为设备编写新的软件。

1998年，荷兰代尔夫特理工大学的雷纳德·朗伯茨（Reinoud Lamberts）在互联网上建立了第一个开源硬件协作项目组Open Design Circuits，致力于协作设计开放、低成本的电路。朗伯茨之后创立了opencores.org开源社区和一个致力于低成本和开放设计电路的协作网站Open Design Circuits。

然而由于半导体产业的特殊性，即使开源硬件在设计上已经没有障碍，但由于流片（像流水线一样通过一系列工艺步骤制造芯片，即试生产）和生产成本过高，又无法通过规模生产来降低成本，因而开源硬件的发展步履维艰，大多数项目尝试运营一两年后就逐渐销声匿迹了。

2. 开源硬件的兴起

开源硬件兴起于21世纪的前十年中期，得益于互联网的普及，硬件设计可以得到更快更广的分享。而开源软件的流行以及开源企业取得的巨大商业成功，再一次刺激了开源硬件的发展。另外，嵌入式系统（Embedded System）市场的快速扩张，使得制造工具的成本逐渐降低，流片和生产成本过高的问题得以解决，开源硬件迎来高速发展期。特别是开源硬件Arduino的出现，使得电子爱好者又拥有了可以快速实现创意的简单且便宜的工具。

随后，越来越多的公司和项目开始加入开源硬件的队伍中，这一时期影响力较大的有OpenCores、RepRap，Facebook设立的开源计算机项目（Open Computer Project），Arduino、BeagleBoard等基于FPGA的开源硬件项目，MakerBot、Egg-Bot等开源桌面3D打印机项目等。

开源硬件一直借鉴和使用开源软件的许可协议。2007年，一家由业余无线电爱好者建立的非营利机构创建了第一个针对硬件的开源许可协议——塔克森业余封包电台租住开源硬件许可证（Tucson Amateur Packet Radio System Open Hardware License，简称TAPR）。之后，CERN（European Organization for Nuclear Research，欧洲核子研究组织）和Solderpad（Solderpad Open Hardware License）协议逐步出现。但相对来说，开源硬件许可证的普及度不高。

2009年7月，在美国班夫中心（Banff Center）举行的开源硬件研讨会上，成立了开源硬件及设计联盟，提出了起源于开源软件运动又适用于硬件的四种自由：出于任何目的使用设备的自由，学习该设备的工作原理并改编的自由，对设备的重新分发及设计（再生产）的自由，改进设备及设计、向公众发布改进（或综合意义上的改变）方案的自由。

2010年3月，一个非营利组织——知识共享组织（Creative Commons）诞生了，徽标如图1.2.2所示。知识共享组织的诞生是为了解决现代知识产权以及著作权法在信息共享方面的问题。在该组织的推动下，2010年9月在纽约举办了第一届“开源硬件峰会”（Open Hardware Summit，简称OHS），通过峰会及随后几个月时间的充分研讨，开源硬件1.0版本（OSHW定义）于2011年2月10日获得大多数参与者的支持和认可，并最终颁布。此后，开源硬件峰会每年都会举办一届。随着开源硬件实践和社群的持续增长，开源硬件峰会也渐成规模。

一个教育型非营利组织——开源硬件协会在第一届开源硬件峰会后成立，该协会致力服务开源硬件运动，开展开源硬件的公众普及教育，帮助建立国际分支机构，搜集和发表关于开源硬件运动的正面信息，以及鼓励各种项目向教育和经济发展等领域开放。



图1.2.2 知识共享组织的徽标

3. 开源硬件的现状

开源硬件社区的不断涌现、开源硬件的日趋成熟、开源硬件许可协议的日益规范等，

推动着开源硬件向前发展。

开源社区 opencores.org 成立以来，一直关注数字模块中 IP 核部分，是世界上较大的开源硬件 IP 核社区，总共汇集了将近 900 个项目，其中最著名的项目是基于 ASIC 设计的 OpenRisc1000 开源处理器。随着开源硬件的兴起，全球各个大大小小的开源社区不断涌现。国外人气旺盛的开源社区有 GitHub、Arduino.cc 等，国内也涌现了众多开源硬件技术社区，如开源中国社区、开源高校推进联盟、DF 创客社区和 Arduino 中文社区等。那些极具创意的开源硬件项目在社区中得到关注、认可和迭代，同时不断得到相应产业的响应，社区的活跃促进着开源硬件的蓬勃发展。如图 1.2.3 就是 Arduino 中文社区的徽标。



图1.2.3 Arduino中文社区徽标

现今，多种开源硬件日趋成熟，主流的开源硬件逐渐建立起了自己的用户群。如 Arduino，包含硬件（各种型号的 Arduino 板）和软件（Arduino IDE），吸引了一大批艺术家、设计师和对“互动”有兴趣的爱好者。众多科技巨头也在布局开源硬件，不断有新的开源硬件成为后起之秀。2014 年，英特尔公司携带 Edison（英特尔公司推出的开源硬件）进入可穿戴设备领域，国外技术爱好者结合 Edison 制造出了可穿戴四旋翼无人机 Nixie 产品。2016 年，英国广播电视公司（BBC）推出了专为青少年编程教育设计的微型计算机开发板 micro:bit，其在智能穿戴项目上的良好应用吸引了众多追随者。2018 年 7 月，国内一些创客名师联合国内高校、创客空间和创客企业，先后发布了名为“虚谷号”和“掌控板”的两款开源硬件，得到了很多创客企业和一线教师的关注。

开源硬件许可协议的日益规范，较好地确保了企业在软件源代码共享和合作领域的积极性。无论是传统科技巨擘像微软、苹果公司，还是新兴的互联网公司如脸书（Facebook）都陆续将部分产品开源。这些行为在不同程度上助推了开源硬件的发展。

整个开源硬件的发展简史如图 1.2.4 所示。



图1.2.4 开源硬件发展史

4. 开源硬件与创客运动

随着技术开发的门槛不断下降，以开源硬件作为开发平台来实现各种创意的人群也不断壮大。人们将这些不以营利为目标，努力把各种创意转变为现实的人称为“创客”（Maker）；而在全世界广大范围内鼓励人们利用身边的各种材料及计算机相关设备、程序及其他技术性资源（如开源软件），通过自己动手或与他人合作创造出独创性产品的行动，

则被称为“创客运动”(Maker Movement)。

创客运动关注青少年的科技创新和动手实践,对教育产生了较大影响,于是创客教育应运而生,成为综合学习、跨学科学习的典型代表。开源硬件有着便利的设计、丰富的元件、开源的文化三个特点,在创客教育中独具优势。目前,许多开源硬件已经进入学校课堂,成为创客教育的必备工具,如Arduino、micro:bit等。

创客运动的发展,也在不断地推动硬件技术的更新和发展。传统的硬件开发是一个“市场调研—做实验—开发—推广”自上而下的过程。而创客秉承了分享与开放的理念,通过开源的方式开放自己的成果,去除了创新个体之间的隔阂,相互之间进行资源配置,让创新整体效率变得更高,也更有利于技术的发展与进步。因此,开源硬件成就创客运动的同时,创客运动也推动着开源硬件的更大发展。

1.2.2 开源硬件的特征

开源硬件的出现和兴起使其越来越被大众所接受,也越来越展示出开源硬件共享、开放、迭代与集智的创新特质。

1. 开放与共享性

开源硬件开放的信息,包括产品的功能目标、实现原理、设计图纸、硬件参数、技术细节、软件代码等。开源硬件作品的拥有者尽可能提供详尽的说明文档和友好的接口,与所有人共享这些信息,从而帮助关注作品的人们理解和使用这些信息和产品,鼓励后继者在其他更多的开源硬件项目中运用这些信息与产品。

事实上,这样的开放与共享正是开源精神的实践与贯彻,开源精神的宗旨就是让所有人和组织都可以得到和使用已经发布的产品,让所有人(包括不懂硬件和软件的人)都有权利进入这个领域并成长创新。

2. 迭代与集智性

开源硬件最大的一个优势是快速的迭代创新,用户可以对已有的产品进行修改和创造,进而得到一个衍生产品。在开源许可证的许可下,允许用户在与原创产品同样的许可条款下进行发布,这个特性大大缩短了开源硬件的迭代周期。对于开发者来说,从一个基础产品或者相似产品进行迭代远比从零开始要容易得多,既可以避免重复劳动,又可以促进同样产品的精益求精。

其次,在开源社区里,有着相同兴趣方向的开发者常常汇聚在某一个或某一类开源硬件项目中。针对特定问题的解决方案、解决思路、解决策略或技术路线等,他们不断地进行交流与探讨、实践与尝试。众人拾柴火焰高,经过这样的分化与集成、竞争与协作的社区创新机制下的思想交流,人们的智慧得到了充分的汇聚与发挥,同时让所有参与其中的开发者也获得了更多的知识与技能。



再者，开源硬件的广大用户同时也是开源硬件发展的强大推动力。对于一个产品来说，足够多的测试者和合作开发者往往可以发现足够多的问题，并且能涌现出足够多的问题解决方案。开源硬件的这种设计模式把用户当成开发合作者，为快速提升产品质量提供了快捷、有效的途径。

? 思考与练习

当今社会智能产品层出不穷，一次次刷新人们的认知。“深度思维”公司将其开发的“阿尔法围棋”的发展分为四个阶段：第一个版本在2015年战胜欧洲围棋冠军樊麾，标志着人工智能首次战胜人类职业棋手；第二个版本在2016年战胜曾多次夺得世界冠军的韩国棋手李世石，标志着人工智能战胜人类顶级棋手；第三个版本在2017年4月战胜当时世界排名第一的柯洁，并在与多位有世界冠军头衔的人类棋手“群战”中完胜；同年10月发布的第四个版本（“阿尔法零”）完全是“从零开始”，通过自我对弈强化学习，取得飞速进步。这些产品都是在初始发明的基础上，经过一次次迭代不断完成更新和提高。

请关注身边感兴趣的一些产品，选择其中的一个，追根溯源，探究它的迭代进程，并制作一份思维导图向同学们介绍。

1.3 开源硬件的经典案例

随着开源硬件的普及，越来越多的人参与开发各种开源硬件项目，涌现出的很多优秀的案例不仅为大众实现自己的梦想提供了可能的途径，还助推了信息技术的进步，影响了社会的方方面面。

1.3.1 RepRap项目

早在20世纪80年代，美国就出现了3D打印机的雏形，但是当时3D打印机的价格极其昂贵且打印质量很差。2004年，英国巴斯大学的艾德里安·鲍耶（Adrian Bowyer）博士萌发了开发自我复制3D打印机的想法。2007年，第一台RepRap机器“达尔文”（Darwin）研发成功，如图1.3.1所示。随后，鲍耶博士把“达尔文”的安装过程和相关文件及程序源码，通过社区平台进行开源。

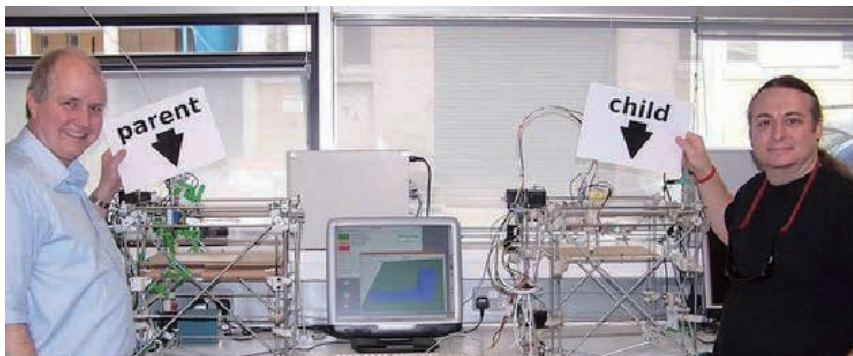


图1.3.1 RepRap “达尔文”，右边的打印机由左边的打印机复制而来

RepRap项目先后发布了4个版本的3D打印机：2007年3月发布了“达尔文”，2009年10月发布了“孟德尔”（Mendel），2010年发布了“Prusa Mendel”和“赫胥黎”（Huxley）。研发者采用生物学家的名字来命名，寓意“复制和进化”。得益于RepRap项目的开源，很多人进行了3D打印机的调试和迭代实践。3D打印机的价格也越低，稳定性越来越好，精度越来越高。全球最大的桌面级3D打印机MakerBot就是基于RepRap项目快速发展起来的。

1.3.2 OpenPilot项目

1935年，“蜂后”式无人机的问世开启了无人机时代。由于无人机的研发需要非常专业的知识，且无人机往往只用于军事用途，在后来很长的一段时间里，研发进展缓慢。近几年无人机之所以能快速普及，很大程度上得益于开源飞控（Open Source Auto Pilot，开

源自动飞行控制器)的发展。OpenPilot是一个著名的开源飞控项目,图1.3.2所示的是一款OpenPilot竞技无人机。

OpenPilot的最大特点是硬件架构简单,且支持多种硬件。利用OpenPilot可以快速制作多旋翼飞行器、直升机、固定翼飞机。现在,从很多方面的应用中都能看到OpenPilot的影子,如供人道主义搜索和搜救使用的空中平台,供摄影师使用的航空摄影平台,用于科学探究活动的科研平台,以及航模爱好者的DIY(Do It Yourself,自己动手制作)平台。



图1.3.2 基于OpenPilot开发的竞技无人机

拓展链接

各种各样的无人机

无人机是无人驾驶飞机(Unmanned Aerial Vehicle)的简称,是利用无线电遥控设备和自备的程序控制装置的不载人飞机,包括无人直升机、固定翼机无人机、多旋翼无人机、无人飞艇、无人伞翼机。广义地看,它还包括临近空间(20~100千米空域)飞行器,如平流层飞艇、高空气球、太阳能无人机等。从某种角度来看,无人机可以在无人驾驶的条件下完成复杂空中飞行任务和各种负载任务,可以看作是“空中机器人”。

按照不同平台构型来分类,无人机主要有固定翼无人机、无人直升机和多旋翼无人机三大平台,其他小种类无人机平台还包括伞翼无人机、扑翼无人机和无人飞船等。

固定翼无人机是军用和多数民用无人机的主流平台,是指机翼外端后掠角可随速度自动或手动调整的机翼固定的一类无人机。这类无人机的最大特点是飞行速度较快。

无人直升机是灵活性最强的无人机平台,是指由无线电地面遥控飞行或自主控制飞行的可垂直起降不载人飞行器,在构造形式上属于旋翼飞行器,在功能上属于垂直起降飞行器。与固定翼无人机相比,无人直升机可垂直起降、空中悬停、朝任意方向飞行,其起飞着陆场地小。

多旋翼(多轴)无人机是消费级和部分民用用途的首选平台,是一种具有三个及以上旋翼轴的特殊无人驾驶直升机。它通过每个轴上的电动机转动,带动旋翼,从而产生升推力。其灵活性介于固定翼和直升机中间(起降需要推力),但操纵简单、成本较低。

1.3.3 Tabby项目

Tabby是意大利的汽车制造公司Open Source Vehicle(OSV)设计开发的全球首款开源DIY汽车,如图1.3.3所示。该车的组装无须特殊工具,无论是2轮或3轮、2座或4座,高尔夫车还是敞篷车,用户都可以按照自己的意愿组装或改装,最高时速可达85千米。



图1.3.3 基于Tabby制造的汽车

依据 Tabby 项目，用户只需购买一些基础零件，根据组装指南便可以在一小时内自行组装出一辆汽车。

1.3.4 HCR 项目

智能移动机器人是集环境感知、动态决策与规划、行为控制与执行等多种功能于一体的综合系统。它集中了传感器技术、信息处理、电子工程、计算机工程、自动化控制工程以及人工智能等多学科的研究成果。起初，商用移动机器人平台由于价格高昂无法让更多的爱好者进入这个领域。2008年，中国的几位创客发布了HCR（Home Care Robot，家庭移动机器人）开源项目，如图1.3.4所示。HCR项目通过社区的方式召集开发人员开发完成。

HCR最大的特点是采用 Arduino 开源硬件作为主控器，硬件层面全部开源，包括机械结构设计。HCR支持高达30多种不同的传感器，具有非常好的延展性。2014年，HCR被麻省理工学院和斯坦福大学的计算机课程选为标准的开发平台。2016年，HCR被英特尔选用，成为 Intel RealSense 的第一台展示样机。HCR还广泛应用于移动机器人的学术研究和商用开发中。



图1.3.4 HCR项目作品

思考与练习

1. 查阅资料，列举开源硬件项目在智能家居中的应用，并选择一个典型案例向同学们介绍。

2. 了解了这么多的开源硬件项目，你是否有创造的冲动？认真思考后说说你想做一个怎样的开源硬件项目，并且说明该项目应用的具体场景。



巩固与提高

1. “如果你有一个苹果，我有一个苹果，彼此交换，我们每个人仍只有一个苹果；如果你有一种思想，我有一种思想，彼此交换，我们每个人都有了两种思想。”萧伯纳的这句名言启示我们要学会分享。在开源世界里，大力鼓励这样集智式的交流和分享。但在人类文明发展的进程中，关于如何推进技术的创新有着各种不同的观点，请你选择其中的一些观点，探究其所支撑的理由和论据，发表自己的见解。

2. 列举两个在同一领域的来自“开源”项目的产品和选择“闭源”的产品，说说它们各自的功能特色，比较它们的优势和不足，完成下表。

项目	开源	闭源
产品		
功能特色		
优势		
不足		

项目挑战

近三年最具影响力的中国开源硬件事件的评述

随着开源硬件的兴起，开源硬件已经是公众视野中一个无法忽视的重要元素。学习本章内容后，你对于开源硬件的历史和特点有了一定的了解，那么你了解开源硬件在中国的发展情况吗？

▶ 项目任务

查找资料，了解开源硬件在中国的发展历程，尤其是近三年内中国开源硬件领域的重大事件、热点话题和关键人物。根据获取的资料，思考开源硬件对信息技术、创新产业以及未来生活的影响。

▶ 过程与建议

本次项目任务主要分为两个部分。第一部分是了解开源硬件的发展情况，尤其是近三年内的重大事件。在这一部分任务的完成过程中，主要是资料的收集和整理。搜集资料时，除了使用常用的搜索引擎以外，你还可以访问开源社区、论坛、名家网站等获取更为专业和直接的资料。第二部分是基于事实的思考和展望，这个过程要体现的是同学们自己的思考和感受，切不可抄袭或沿用他人的内容。

1. 资料收集

查找和收集中国开源硬件发展的资料。

2. 资料整理

将收集到的资料按照时间进行整理，明确每一时间段中所发生的代表性事件、热点问题以及关键人物。

3. 思想碰撞

以小组为单位根据已有资料，围绕“中国开源硬件的发展对于信息技术、创新产业以及未来生活的影响”展开讨论，及时记录和整理讨论内容，形成一个小组报告。

4. 展示交流

选择合适的展示形式（PPT、网页、视频、文字稿等），通过班级或校园网站、网络论

坛等多种平台进行展示。

▶ 评价标准

请根据项目实施的过程、效果以及成果展示交流的结果，对自己完成项目的情况进行客观的评价，并思考后续完善的方向。把评价结果和完善方案填写在下面的表格中。

评价条目	说明	评分（1~10分）	评分主要依据阐述	后续完善方向
资料收集和整理	能通过不同方式查找所需资料，并按要求对信息进行整理			
思考和体会	能从多角度分析中国开源硬件发展对中国社会发展的影响			
报告	报告完整，表述准确，逻辑清晰			
展示	方式新颖，内容详实，效果良好			

▶ 拓展项目

以本章所学内容和项目挑战内容为依据，围绕开源的价值组织一场班级辩论赛。辩论主题有如下选择：

1. 正方：我看好开源运动的发展前景。反方：我不看好开源运动的发展前景。
2. 正方：我接受开源理念。反方：我不接受开源理念。
3. 正方：开源硬件有助于中国创新产业发展。反方：开源硬件不利于中国创新产业发展。

在辩论赛后，根据辩论双方的论据撰写一篇议论文，对开源硬件的未来发展做出更为全面的思考。

开源硬件项目的开发流程



在数字加工工具和开源硬件的支持下，“创意当实现”的门槛不断下降，越来越多的非专业人士甚至中小学生都开始涉足硬件项目领域。他们针对自己在生活中遇到的实际问题提出了很多解决方案并着手实施，成为一个个开源硬件项目的开发者。

一个优秀的开源硬件项目从创意诞生到结果呈现并不容易，中间要经历很多曲折。了解一些开源硬件项目的开发过程，掌握创意生成和需求分析的科学方法，有助于开源硬件项目的顺利开发。

问题与挑战

● DIY就是自己动手制作，从自己动手进行住房整修、庭园维护，到计算机组装等，逐渐演绎成为一种流行的生活方式。DIY倡导通过自己动手，充分发挥个性和创意，创作一份“表达自我”的作品。因此，创客运动也被认为是DIY在数字时代的延伸。随着互联网的发展，专业领域之间的界线越来越模糊，项目的开发者往往也是使用者，这种变化给项目的开发及后续的使用带来了哪些优势？

● 当前已经进入了一个创意驱动的时代。有些人苦于找不到创意，有些人怀揣着一大堆天马行空的想法却难以实现。如何从生活中寻找创意的灵感？又如何有效地区分哪些创意可以实现，哪些创意是不切实际的空想？

● 从想法到设计、从设计到蓝图、从蓝图到成品，一个好的设想需要转化为现实才能体现其真正的价值。开源硬件的出现降低了普通人实现创意的门槛，但开发一个优秀的开源硬件项目仍需要对每个环节进行精心设计、反复调试、迭代，这个过程中存在着诸多可能导致整个项目失败的客观因素。如何科学规划整个项目的开发过程，避免一些潜在的风险？

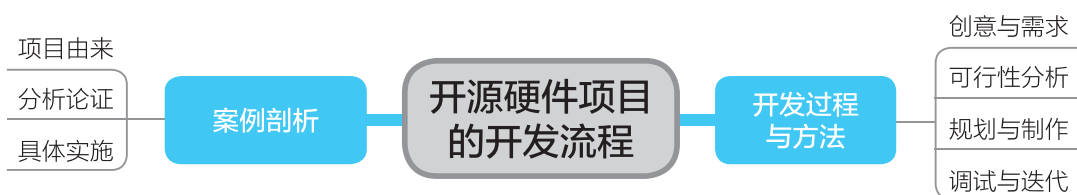


学习目标

1. 了解一个开源硬件项目从创意诞生到分析论证，再通过具体实施形成作品实物的过程。
2. 通过剖析开源硬件项目的实例，体验项目开发的一般过程，理解在开发过程中用到的常见方法。
3. 基于事物特征的分析，设计基于开源硬件的作品开发方案，描述作品各组成部分及其功能作用，明确各组成部分之间的调用关系。



内容总览



2.1

开源硬件项目的案例剖析

以前开发一个硬件项目需要多方面的专业知识，而开源硬件的出现让技术鸿沟得以跨越，这让非专业领域的人根据自身的需求开发开源硬件项目成为可能。“智能鸟蛋”项目就是一个非常典型的例子。

2.1.1 项目由来

因为人为的狩猎和生态的恶化，亚洲秃鹫在最近二十年灭绝了90%，成为地球上最濒危的鸟类之一。国际猛禽中心因此启动了一个保护秃鹫的公益项目，目的是想通过研究秃鹫蛋的孵化环境，尝试以人工孵化的方式拯救这些猛禽。但是秃鹫不同于养殖场的鸡，无法进行驯化饲养。有人曾尝试将检测环境参数的仪器放到秃鹫窝里，但这些设备很快就被秃鹫发现并破坏，因此很难获取秃鹫蛋的孵化环境参数。

要人工孵化就得先了解秃鹫蛋的孵化环境，有人提出做一个伪装成鸟蛋的智能检测仪器放到秃鹫窝里，让秃鹫把它当作自己的蛋来孵化，从而观察秃鹫到底是怎么孵蛋的。

2.1.2 分析论证

正确的温度与湿度是鸟蛋能成功孵化的重要因素，“智能鸟蛋”的核心功能是获取温度和湿度。但经过观察，秃鹫在孵蛋过程中会不断调整蛋的位置，因此获取假鸟蛋表面的温度分布和孵化过程中的姿态变化，有助于深入了解秃鹫蛋在孵化过程中的一些细节。

母秃鹫如果在孵化过程中发现鸟蛋是假的就会破坏它，因此需要把假鸟蛋设计得更加逼真。外观的逼真，就意味着“塞到”鸟蛋内的硬件体积要足够小。此外，秃鹫蛋的孵化一般需要两个多月时间，智能鸟蛋因此至少需要能连续工作70天左右，这就需要设计一种低功耗、长续航的解决方案。

经过分析归纳，“智能鸟蛋”项目的主要需求可以概括为：

- ①获取鸟蛋表面温度分布。
- ②获取环境湿度。
- ③获取鸟蛋的姿态。
- ④逼真的外观。
- ⑤小体积的硬件电路。
- ⑥低功耗，长续航。

拓展链接

一些体积小巧的 Arduino 开发板

Arduino 被誉为开源硬件中的“神器”。对于电子爱好者们来说，拥有这样一种适合大部分场合下使用的核心板，就可以大大地简化设计流程。但 Arduino 的外形尺寸相对固定，不太适合那些空间需求较高的项目。因此，一些企业和创客设计了体积非常小巧兼容 Arduino 的开发板。

1. Microduino

Microduino 是美科科技开发的兼容 Arduino 的开源硬件平台，大小和一枚一元硬币差不多，适用于对体积有要求的项目。开发者可以根据实际情况配合其他兼容模块，对项目进行功能扩展。目前 Microduino 的核心板分别为 Microduino-Core、Microduino-Core+ 和 Microduino-FT232R。

2. RFDuino

RFDuino 是由 Open Source RF 开发的一款仅有指尖大小的开发板，配备 Nordic 32 bit ARM Cortex-M0 处理器，并支持 Bluetooth 4.0。

3. BLEduino

BLEduino 是一个来自波多黎各的硬件众筹项目，这是一款支持 Bluetooth 4.0 的 Arduino，体积很小。

4. Digispark

Digispark 是一款基于 ATtiny85 的开发板，虽然功能不及 Arduino 强大，但是代码和 IDE 都与 Arduino 保持着一致。它的特别之处是可以直接使用 USB 连接计算机，不需要转换器。

为了让研究人员可以远程了解“智能鸟蛋”的情况，甚至让在千里之外的研究者也可以参与这个项目，就有必要让假鸟蛋把采集到的数据实时传回。因此该项目的次要需求有：

- ① 远程无线数据传输功能。
- ② 直观地显示采集到的数据。

只有主要需求都被满足时，“智能鸟蛋”项目才有意义。如果次要需求也可以被满足，那么该项目的实用价值会大大提高。因此在项目真正着手实施前，有必要对上述需求逐一进行论证。

项目开发者优先论证了主要需求的可行性，如表 2.1.1 所示。

表 2.1.1 项目主要需求的可行性分析

序号	主要需求	大致的解决方案
1	获取鸟蛋表面温度分布	可以利用分布在鸟蛋不同位置的多个温度传感器进行数据采集来实现
2	获取环境湿度	通过湿度传感器实现
3	获取鸟蛋的姿态	利用加速度传感器和陀螺仪传感器实现
4	逼真的外观	3D 打印技术及激光切割技术应当可以做到其外观与真鸟蛋足够接近

续表

序号	主要需求	大致的解决方案
5	小体积的硬件电路	各种传感器模块本身集成度都较高，若采用同样体积小巧的控制板，应当能满足该需求，而市场上就有很多体积小巧的开源硬件
6	低功耗，长续航	一般设计方法很难实现这么长的续航时间，但是参考智能水表等的设计方案，若将假鸟蛋中硬件的功能尽可能精简，只保留必要的数据采集和发送部分，且除数据采集和发送以外，其他时间让硬件进入低功耗休眠，则可以解决续航问题

为了提高该项目的实用价值，次要需求的可行性也需要论证，如表2.1.2所示。

表2.1.2 项目次要需求的可行性分析

序号	次要需求	大致的解决方案
1	远程无线数据传输功能	数据传输可以使用各种成熟的无线方式进行，但是考虑到硬件体积和功耗以及远程传输的需求，采用低功耗蓝牙加数据中继方式会较好
2	直观显示采集到的数据	设计一个网页，将接收到的数据以图表形式呈现出来

以上对每个需求均给出了大致的解决方案，而且都有较高的实现可能性。

2.1.3 具体实施

在“智能鸟蛋”项目中，因为“蛋”内待机要长达70天，必须考虑项目的低功耗设计，大部分的数据处理、计算都让“蛋”外的中继来完成，包括数据的储存和通过网络的数据交互。当然，为了接入互联网，还需要设计一个云端平台。

整个项目可拆分为蛋内采集端、数据中继端和服务器云端三个部分。通过蛋内采集端获取温度、湿度及姿态数据，利用低功耗蓝牙发送给附近的数据中继端，再由数据中继端把数据通过互联网发送给服务器云端。再把以上数据进行储存，并由一个网页展示这些数据，系统运行流程如图2.1.1所示。

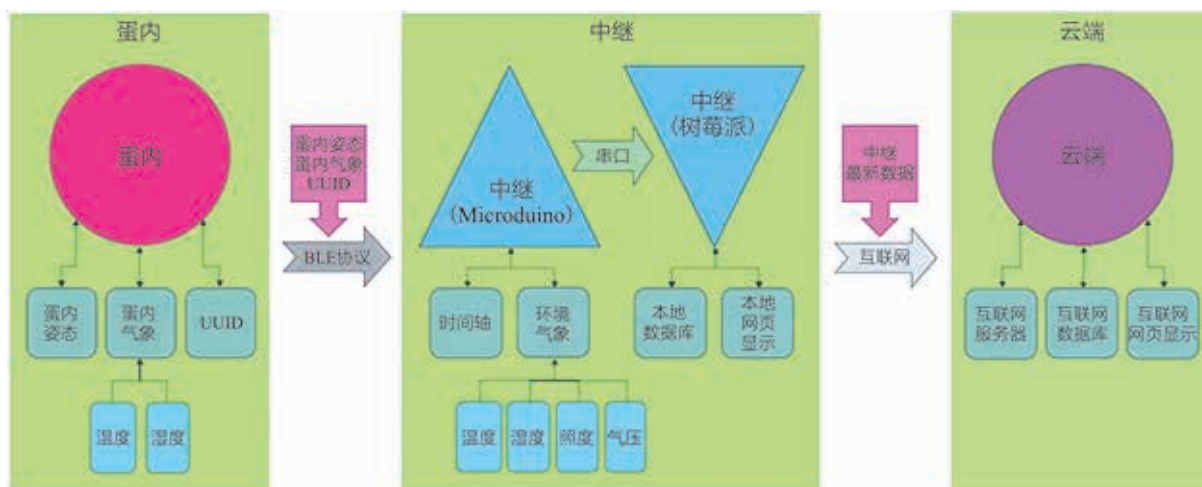


图2.1.1 “智能鸟蛋”项目的系统框架图

该项目中蛋内采集端、数据中继端都涉及硬件，其中电路设计部分做如下考虑：

因为要求蛋内采集端的体积非常小，所以蛋内采用体积小巧的Microduino主控制器。而数据中继端需要处理较为复杂的数据，对处理器的运算性能要求较高，但对于体积没有特殊限制，因此采用树莓派作为数据中继端的主控制器。

同时采集多个点的温度需要连接多个温度传感器，因此选择DS18B20温度传感器。这种传感器通过单总线通信，一个I/O端口即可连接十几个传感器。湿度传感器选择SHT21，该传感器是通过I2C总线与主控制器通信，功耗低，性价比高，尺寸小。

考虑到在获取假鸟蛋姿态的同时又要做到省电，所以获取姿态的传感器采用MPU6050。该传感器不仅可以使I2C总线与主控制器通信而节省数据端口，还可以利用传感器内的中断，通过中断唤醒的方法，避免控制器长时间刷新数据，从而实现省电。

有了数据中继端，蛋内采集端只需使用低功耗蓝牙，定时将数据发送给数据中继端即可，选择了常用来做智能手环的BLE模块CC2541。

而在数据中继端，还需要一个准确的时间使得接收到的数据与时间匹配，因此增加了RTC时钟模块，该模块使用的是PCF8563芯片，也是一款I2C总线接口的功耗极低的芯片。

蛋内采集端和数据中继端还涉及结构设计部分。一个外观逼真的假鸟蛋，用3D打印技术来实现。通过研究真实的秃鹫蛋样本，利用三维建模软件对假鸟蛋的外形进行建模并打印。

考虑到在孵化过程中，假鸟蛋会被母秃鹫经常性地翻动，因此需要设计一个稳固的内部支架固定电路部分。“智能鸟蛋”项目组使用激光切割技术，利用木板和激光切割机做了一个稳定的鸟蛋支架，如图2.1.2所示。

而项目中的蛋内采集端、数据中继端以及服务器云端都涉及软件设计。因为软件开发平台与硬件相关性较高，所以首先要为项目选择合适的软件开发平台：

①蛋内采集端采用了Microduino主控制器，因此选择了为该控制器深度定制的Arduino IDE进行软件开发，利用该开发平台可以便捷地实现传感器数据采集与BLE通信。

②数据中继端使用的是树莓派，因此选择Python作为编程工具，开发速度快。

③服务器云端为了方便地记录和显示数据，因此采用了Microduino的mCotton物联网平台。

④数据传输是整个项目中各部分的联系纽带，合理的数据传输格式不仅可以使项目各部分更容易开发，还可以为后期调试提供便利。

“智能鸟蛋”项目的蛋内采集端将在采集各类型数据后对数据进行一定的整理，按照一定的数据格式传输给数据中继端。该数据格式被定义为：

0xAA 0xBB [type] [data] 0x0D 0x0A

其中，0xAA 0xBB是固定的数据头；[type]是类型代码，分别用不同字节表示姿态、温度和湿度；[data]是数据，按数据类型的不同，数据长度也做了定义；0x0D 0x0A是固定



图2.1.2 蛋内采集端的电子电路和结构支架

的数据尾。数据在传输过程中可能会造成丢失，而数据头和数据尾用来验证接收数据的有效性。

通过连续几个月的不懈努力，“智能鸟蛋”项目组最终做出的假鸟蛋能够以假乱真，放到秃鹫窝里，母秃鹫对它呵护有加，如图2.1.3所示。

项目进展很顺利，假鸟蛋获取的数据经过数据中继端再上传到云端，最终以直观的形式展现在了研究人员面前，如图2.1.4所示。



图2.1.3 以假乱真的智能鸟蛋

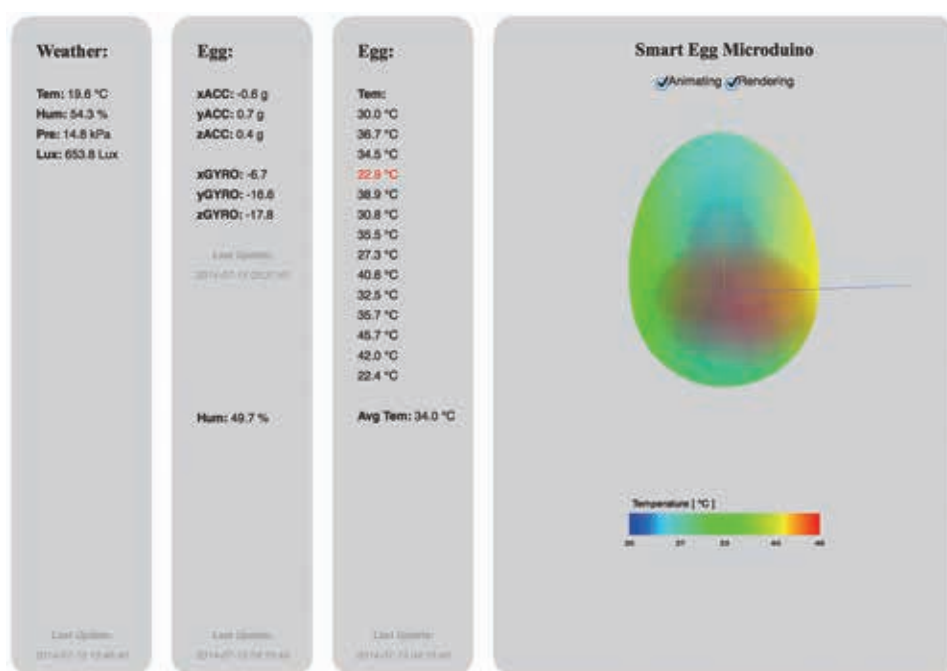


图2.1.4 “智能鸟蛋”项目数据的直观呈现

有了数据支撑的人工孵化技术，使孵化的秃鹫存活率大大提高。“智能鸟蛋”项目不仅可以帮助人类拯救秃鹫这一濒危物种，还可以将这一项目的技术方案移植到拯救其他生物的项目上，意义非凡。

思考与练习

1. 通过查阅资料，深入了解某个你感兴趣的开源硬件项目，再制作一个简单的演示文稿，向同学介绍这一项目从创意的诞生到实物呈现的过程。
2. 写一个你想做的开源硬件项目，并指出该项目应用的具体场景。

2.2 项目的开发过程与方法

铁制的菜刀用完后如果不保持干燥，就容易生锈并滋生细菌，于是有人开发了一个简单实用的“自动杀菌并吹干的菜刀架”；在脑电波传感技术普及后，有人通过脑电波传感器获取脑部活动信息，并结合3D打印技术制造出了一个“脑电波机械手”。“不满是向上的车轮”，就发明创造而言，在生活中细心观察发现问题并动手动脑，才会有源源不断的创新灵感。同样，只有掌握更多的知识和技术，才能设计出更多更精彩的项目。

不同的开源硬件项目的内容可以千差万别，但是开发过程却有一定的规范可循。掌握开源硬件项目开发的过程与方法，不仅有助于保证项目的质量，还可以在很大程度上规避开发风险。

2.2.1 创意与需求

一个开源硬件项目的产生往往源于某些创意，而创意的来源又有多种途径。有些创意诞生于考虑解决某些实际问题的过程中，比如前文提到的“自动杀菌并吹干的菜刀架”的创意就属于此类；也有些创意是根据现有的技术通过跨应用领域迁移而来，比如为残疾人士带来福音的“脑电波传感器”就是跨应用领域而产生的创意。

支持创意生成的方法有很多，比如观察记录法、头脑风暴法、思维树法等。

①观察记录法。随身携带笔记本，养成在生活中发现问题就及时记录的习惯，在翻阅记录并考虑如何解决这些问题的过程中，往往会萌生很多创意。观察记录法的核心是“问题源于生活”。

②头脑风暴法。这是指一群人围绕着一个特定的兴趣或领域进行改善或创新，产生新点子、提出新办法的过程。在头脑风暴过程中，要鼓励每个人针对某个具体问题及其解决办法畅所欲言，各抒己见，在集体讨论中提及创意实施的可行性。

③思维树法。从一个实际场景开始，通过思维导图建立一棵思维树的方式，呈现出问题的解决方案。图2.2.1就是以家庭卫生间为场景建立的思维树。

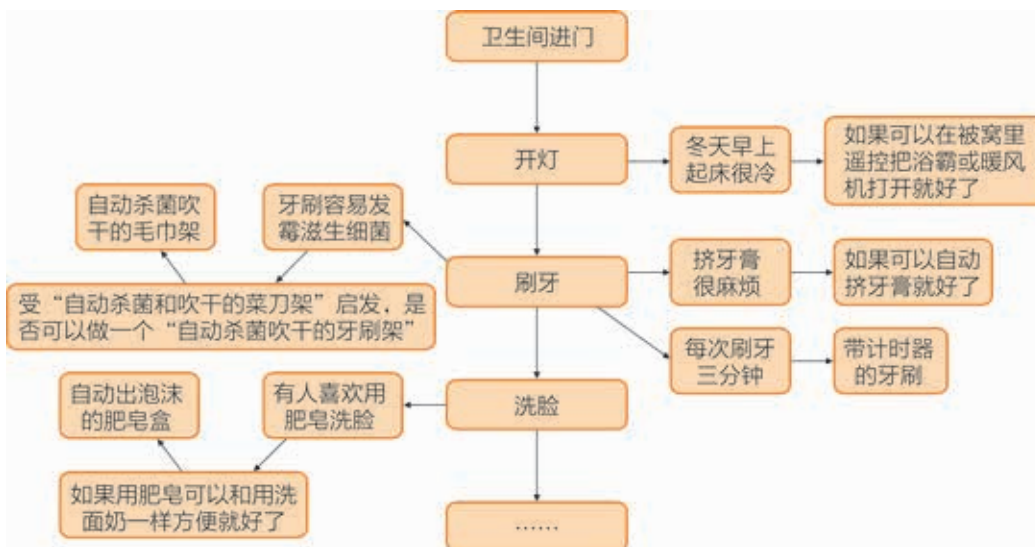


图2.2.1 家庭卫生间的思维树（部分）

有了好创意，不急着马上实施。首先需要根据目标来细分具体的需求。值得注意的是，这里的需求不是指“想要做怎么样一个项目”，而是“这个项目需要实现怎样的功能”，是对整体功能的细分。如在上一节的“智能鸟蛋”项目中，其整体功能是“获取鸟蛋孵化的环境数据”，根据整体功能结合实际情况，开发小组列出了6条主要需求和2条次要需求。

对于一个项目的需求获取，比较有效的方法有“小组讨论法”“同理心地图法”等。小组讨论法类似头脑风暴法，这种方法比较适合项目开发者就是项目使用者的情况，常用于对需求较为明确的项目中。同理心地图法通过对被调查人员所处的环境、行为、关心事项及内心的深入了解，为确定用户的具体需求提供充分的观点，更适合使用对象为另一类人群的项目。为了让项目的需求尽可能完整，先应该尽可能多地列出项目需求，再把这些需求进行归类整理，把相似的需求合并，删除明显不合理的需求，最后进行主要需求和次要需求的划分。

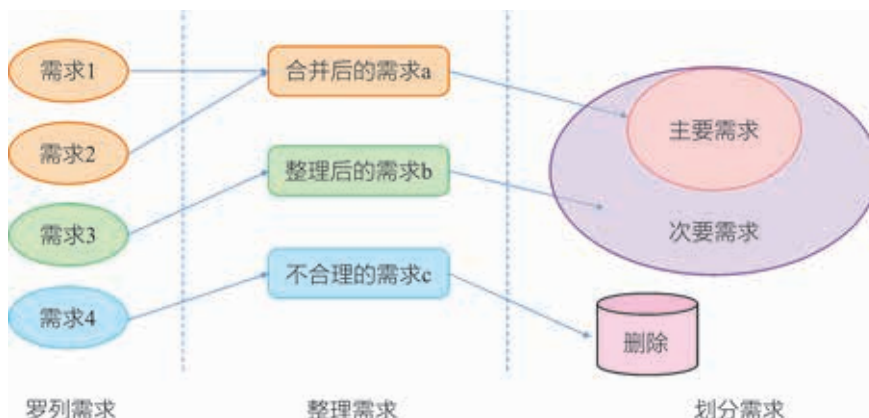


图2.2.2 项目需求的获取过程

拓展链接

同理心地图

同理心地图（The Empathy Map）是由XPLANE公司开发的一款工具，旨在观察、调查研究对象所处的环境、行为和需求等，从而为改进产品质量、研究开发新产品、提升用户满意度等提供充分的观点。其完整的应用步骤主要包含以下几个环节：

1. 确定研究对象。
2. 画出LOOK、LISTEN、THINK、TALK、PAIN和GAIN六个区域模块，观察、询问研究对象并填充到各个模块。
3. 整理、归类和挖掘。整理分析被研究者的话语、情境、动作及感受，将意思相同的条目归为一类。
4. 连接。分析不同类别的条目，重点关注不同类别条目之间的相关性及矛盾冲突。
5. 检查、归纳与提炼。邀请其他人检查自己的同理心地图并提出改进建议，归纳描述其中存在的问题。
6. 生成需求表述。

主要需求和次要需求的划分依据为：该需求是否为项目所必需。若少了该需求项目即失去意义，则为主要需求；反之，则为次要需求。当然，次要需求对项目来说并非可有可无，次要需求的实现可以在很大程度上增加项目的实用性。

需求分析的结果是进一步明确项目的目标，得到该项目较为完整的需求清单。需求清单同样可以分为主要需求清单和次要需求清单两部分，每部分都可以参考如表2.2.1所示的格式。

表 2.2.1 需求清单

编号	主要 / 次要需求	原因说明
【例】	逼真的鸟蛋外观	秃鹫非常聪明，只有足够逼真的鸟蛋，才能骗过母秃鹫。
1		
2		

2.2.2 可行性分析

可行性分析是项目实施前必不可少的关键环节。所谓可行性分析，是运用多种科学手段对项目进行论证的过程。可行性分析可以大大降低项目的开发风险。开发一个开源硬件项目不同于科幻想象，是要通过各种技术手段最终实现出来的。在列出项目的需求清单后，需要逐一考虑清单中需求的可行性，即能否通过现有的技术手段进行实现。针

对某一条需求进行可行性分析，可以参考图 2.2.3 的流程。

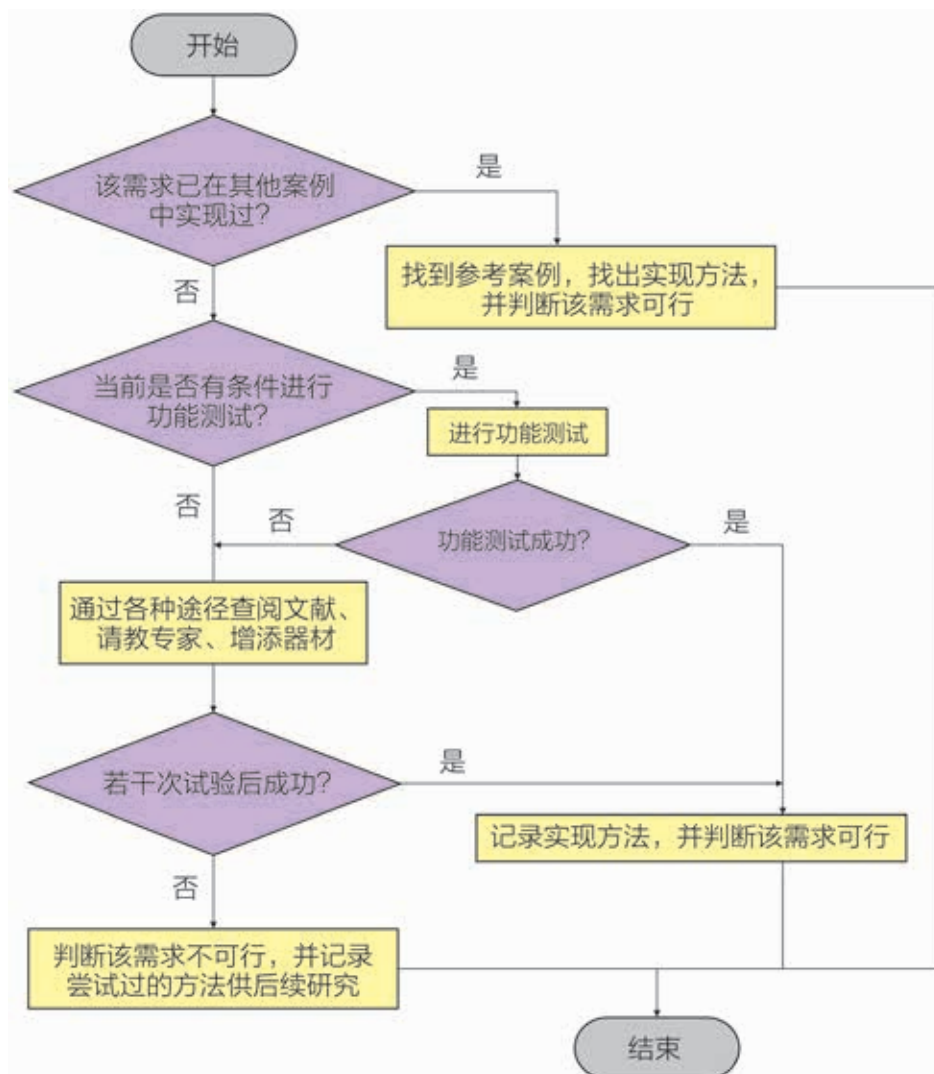


图2.2.3 对某一条需求进行可行性分析的流程

在进行可行性分析时，要时刻留意该需求的实现方法是否会对其他需求产生影响。如果产生影响，次要需求要服从主要需求，所以可行性分析的次序一般是先分析主要需求再分析次要需求。若主要需求无法满足，则意味着项目的既定目标无法实现，需要重设项目目标。若次要需求无法满足，则需要评估项目的实用性会有多大影响，考虑是否还有其他替代解决方案。值得注意的是，一旦需求经过更改，就得再次进行可行性分析。

2.2.3 规划与制作

对项目的需求逐一进行可行性分析后，才能确定该项目是否可以继续进行。但可行性分析只讨论了该项目的需求能否满足，却没有讨论各需求之间的联系，也没有确定具体解决方案。在真正动手制作某个项目之前，还需要对项目进行整体规划。

针对开源硬件项目的整体规划，TRIZ理论中提出了两个原则：

1. 分割原则。一个较为复杂的项目应该尽可能被分割为若干个相对独立的小项目或者小的组成部分，这些小项目或者小的组成部分要尽可能容易实现。

2. 局部性原则。这是指从整体中分割出来的局部应当具有较完整的内容，各个局部之间尽可能通过简单的方式连接，这样做不仅易于对项目进行分工，使得负责各部分制作的人只需专注于自己那部分即可，还可以使项目后期的调试变得简单。

如在“智能鸟蛋”项目中，整个项目按照“分割原则”和“局部性原则”被拆分为蛋内采集端、数据中继端、服务器云端这三个各自具有明确功能的部分，而且各部分之间通过简单易实现的数据通信方式连接成整体。

问题与讨论

在普通计算机中，CPU、内存条、硬盘、主板等部件相互独立，各部件之间分工明确、协同工作，还可以方便地替换各种器件，而苹果公司的MacBook产品把计算机CPU、内存，甚至硬盘都集成在一块主板上。

这种高度集成的方式是否违背了“分割原则”和“局部性原则”？该如何正确看待这种情况？

拓展链接

TRIZ原则

TRIZ的全称为“发明家式的解决任务理论”，是由苏联发明家阿利赫舒列尔（G. S. Altshuller）在1946年创立的。

阿利赫舒列尔发现任何领域的产品改进以及技术的变革、创新和生物系统一样，都存在产生、生长、成熟、衰老、灭亡的过程，是有规律可循的。人们如果掌握了这些规律，就能更好地进行产品设计并能预测产品的未来趋势。在之后数十年中，阿利赫舒列尔穷其毕生的精力致力于TRIZ理论的研究和完善。在他的领导下，苏联的研究机构、大学、企业组成TRIZ的研究团体，分析了全世界近250万份高水平的发明专利，总结出各种技术发展进化遵循的规律模式，以及解决各种技术矛盾和物理矛盾的创新原理和法则，建立一个由解决技术和实现创新开发的各种方法、算法组成的综合理论体系，并综合多学科领域的原理和法则，建立起TRIZ理论体系。

开源硬件项目往往包含硬件部分和软件部分。在具体项目的开发过程中，硬件部分与软件部分既高度协调又分工明确。因此在规划了整体架构后，还需要对项目功能的实现进行软硬件规划，软硬件规划包括硬件规划与软件规划。

硬件规划包括电路规划和结构规划。电路规划需要确定电路中各个模块的具体型号及相互之间的电气连接方式，结构规划需要考虑具体的制造技术、制作材料等。

软件规划需要结合不同的开源硬件考虑选择不同的软件开发平台、设计各部分之间的数据传输格式与人机交互方式等。

项目的具体制作也不是一蹴而就的，对较为复杂的开源硬件项目的开发，一般可以遵循如图2.2.4所示的顺序。

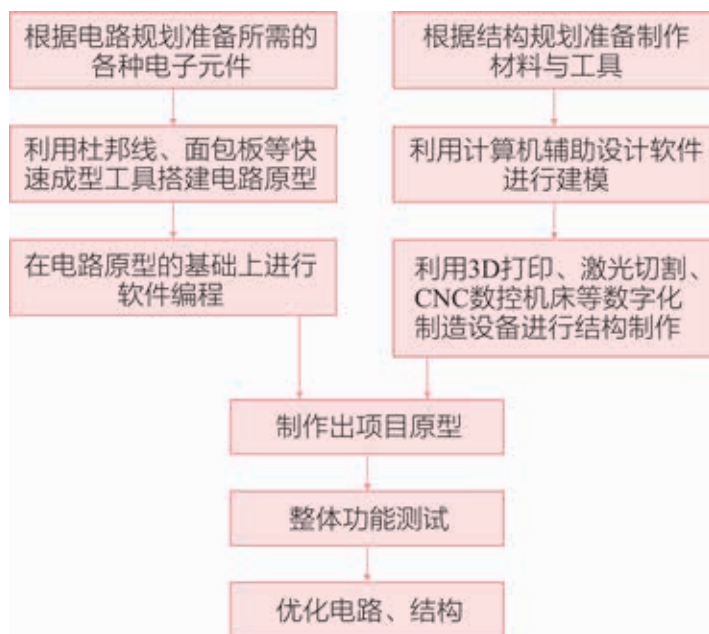


图2.2.4 典型的开源硬件项目制作顺序

值得注意的是，在项目制作过程中需要及时整理和归档相关文件，如程序的源代码文件、设计图纸等，以便随时进行回溯。另外，随着计算机辅助设计（CAD）与数字化制造技术（CAM）的发展，硬件项目的可复制性大大增加了。为了以后方便对项目进行迭代更新，应尽可能利用3D打印、激光切割、CNC数控机床等数字化制造的方式来加工制作项目的硬件结构部分。

拓展链接

常用的数字化制造工具

在开源硬件项目的硬件部分开发过程中一般先使用 AutoCAD、SolidWorks、3DOne 等软件进行数字化建模，再利用 3D 打印机、激光切割机、CNC 数控机床等数字化工具进行加工。

3D 打印是一种增材制造技术，不同的 3D 打印机支持不同的打印材料。比如最常见的熔融层积技术（FDM）打印机，支持聚乳酸（PLA）、工程塑料（ABS）等材料；光固化成型技术（SLA 或 DLP）打印机，主要以光敏树脂为打印耗材；选择性激光烧结（SLS）打印机，以金属粉末为材料进行打印，可以直接制造金属结构。

激光切割是在数字化控制下利用激光对材料进行切割的技术。常见的激光切割机有二氧化碳激光切割机、金属激光切割机等。二氧化碳激光切割机适合加工亚克力、木板等非金属材料，价格相对便宜；金属激光切割机则可以加工金属板材，价格比较昂贵。

CNC 数控机床主要用于加工金属件，它通过多个步进电机控制主轴精确运动，并通过主轴带动雕刻刀高速转动来雕刻金属。为了防止金属在高温下软化，在加工过程中一般需要用水冷却雕刻刀。

为了使用安全，在使用数字化加工工具前，应该仔细阅读操作手册，并在使用过程中严格遵守操作规范。例如，激光切割机在使用过程中需要佩戴防护眼镜以避免激光对人眼的伤害；CNC数控机床在加工过程中，操作者要远离主轴，防止飞出的碎屑造成人身伤害。

2.2.4 调试与迭代

一个开源硬件项目问世前，需要进行反复调试。调试的目的是测试项目的功能与稳定性，从中发现新的问题，并分析解决这些问题，从而提高整个项目的质量。常见的调试过程有完整性测试、鲁棒性测试等。

①完整性测试。对照项目的需求清单，逐条测试需求的实现程度，检查项目在功能上是否有遗漏。

②鲁棒性测试。鲁棒是Robust的音译，也就是健壮和强壮的意思。鲁棒性测试根据对项目性能的不同定义，又分性能鲁棒性测试和稳定鲁棒性测试。性能鲁棒性测试主要检测系统在输入错误、运行过载甚至被部分破坏的情况下是否还可以继续正常工作，比如项目中不同的部分在进行数据通信时产生了一个错误格式的数据包，这时就要求系统不会因为这个错误的数据包而产生异常，在具体实施时，可以使用黑盒测试法进行测试。稳定鲁棒性测试主要用于检验某些系统长时间在恶劣的环境中是否可以稳定、持续工作，具体检验项目包括电路抗老化程度、抗环境干扰能力以及结构的抗疲劳能力等。

一个项目从诞生问世到后期日趋成熟往往经历了多次迭代，开源硬件项目也是如此，每次迭代都可以修正原有项目的不足、增加新的功能，从而不断地优化项目。前面介绍的“智能鸟蛋”项目也经历过多次迭代，如前期版本的数据中继部分使用的是树莓派，而在后来项目的迭代过程中被改成了智能手机，因为智能手机本身集成了蓝牙模块和网络模块，可以直接从蓝牙接收蛋内的数据并通过4G网络传输给服务器云端；而在蛋内采集端的温度传感器，也改成了可以用I2C总线进行通信的LM75传感器。

III 实践与体验 III

以“智慧文具”为主题展开讨论

文具是学习生活中必不可少的一类用品，随着时代的发展，一部分传统文具逐步被充满科技感的智能文具所替代。为使大家在讨论时能对“智慧文具”各抒己见，可以采用“头脑风暴”这一活动形式。

实践内容：

1. 围绕“智慧文具”的主题，以小组为单位展开头脑风暴活动。

2. 对头脑风暴活动的结果进行归类、整理以及总结。

实践步骤：

1. 以4—5人为一组进行分组，每个人都准备一张“头脑风暴记录表”，并在记录表上写上自己的姓名。

2. 用4分钟时间在记录表上通过文字、草图描述自己的创意思法以及简单的实现过程。

3. 将小组内的记录表进行轮换，用2分钟时间在他人的记录表上记录自己对其创意的意见或者自己受到启发后的一些新想法。重复此过程，直到轮换的记录表回到自己手上。

姓名：	
我的创意描述：	
第一轮：	
第二轮：	

4. 结合记录表中他人的意见完善自己的创意。

思考讨论：

1. 在小组中发表自己的观点，评选出组内最具有价值的创意。
2. 向全班同学展示组内最具有价值的创意，并说明具体的需求。

? 思考与练习

1. 绘制一个流程图表示开源硬件项目的开发过程，并用流程图向同学们介绍各过程的注意事项。

2. Makey Makey是麻省理工学院的杰伊·西弗（Jay Silver）和埃里克·罗森巴姆（Eric Rosenbaum）开发的一个开源硬件项目，该项目可以将身边常见物品变成计算机控制器。请通过查阅资料说明该项目经历了哪些迭代过程，每次迭代后都改进了哪些不足。

巩固与提高

1. 通过查阅资料，了解更多开源硬件项目的开发流程。
2. 在开源硬件项目开发过程中，需求分析是否做一次即可？为什么？
3. 教室墙上的挂钟和全球卫星定位系统都需要有计时功能，但两者对计时的精度要求却截然不同，因此实现计时功能的方法完全不一样。试列举几个类似的产品或者项目，分析是否存在需求类似但因精度要求不同而实现方法不同的情况。

项目挑战

创客训练营——项目创意与构思

随着技术的不断发展，设计与开发创新创意项目的技术门槛越来越低，更多的人可以成为研发队伍中的一员，通过创造个性化的产品，满足人们日趋丰富多样的需求。对于这些努力把各种创意转变为现实的创客而言，作品是他们的标签。而要研发出有价值的作品，首先要有好的创意与构思。

▶ 项目任务

在生活中发现并确定有现实需求且可以用开源硬件来实现的项目，集思广益，进行可行性分析，完成项目的构思与策划。

▶ 过程与建议

1. 组建团队

团队是为实现某一个共同目标而相互协作的个体所组成的共同体。每个成员因各自优势与特长不同，需要在不同的位置上各尽所能。越是复杂的项目，越需要有序的组织、合理的分工、正确的策略与严格的执行。现在，是你和学友组建创客团队的时候了。

团队至少要有三个人。为了使团队优势互补，组建时可以考虑团队中包括以下长项的成员：

- (1) 善于综合分析与创意的人。
- (2) 具有技术优势的人。
- (3) 善于组织管理并有序推进项目的人。

2. 确定开源硬件项目的主题

一个好的开源硬件项目，一定是能够解决某种实际需求并有所创新的。为了确保这一项目的创新性与可行性，建议从“现实问题”和“开源硬件功能”两个方面去挖掘可能的项目主题。

- (1) 从现实生活中发现问题，挖掘主题。

很多创新发明都是在发现现有事物问题的基础上创造出来的，为了发现新的创新点，“吹毛求疵”并非坏事。

要点：选择某个物体或某项工作流程等，不断质疑；然后更换物体和流程，再次开展质疑。

质疑问题	你的发现	可能的改进方向
它是完美的吗？它有哪些问题？		
每个人在使用它的时候是否都感觉良好？什么人会不愿意使用它？为什么？		
它在使用时一直是顺畅的吗？在哪种情况下，它会有较大的问题？		

(2) 从已有的开源硬件功能中发现可能性，挖掘主题。

有很多创新项目是被新的技术功能启发出来的。开源硬件的发展，使一些原来很复杂、很困难的工作变得容易起来，也在一定程度上引领了人们的需求。

要点：从开源硬件的功能出发，思考这样的功能应用到哪些事物中，将会发生怎样的变化。

开源硬件的功能	可应用的情境	将会发生怎样的变化
例：能读取智能卡信息	门禁系统	出门无须携带一大串钥匙，只需要带一张智能卡就可以了

综合前面两个方向的主题挖掘，需要兼具创新性和可行性两个方面来形成开源硬件项目的主题方向。

- 选定的项目主题是：_____。
- 想要解决的问题是：_____。
- 该主题的大致构思是：_____。

3. 对项目的需求进行可行性分析

创客团队集思广益，利用下表对项目的需求进行可行性分析，并确定它们是主要需求还是次要需求。

需求	大致的解决方案	主要需求/次要需求

4. 绘制草图，展示交流

当一个小组（群体）针对某个作品有很多创意时，这些离散的想法可能是一个个抽象的概念或者口头的描述，大家不一定有直观的认识。要想理解彼此的想法，尽快达成共识，就需要利用视觉艺术将想法直观地“描绘”出它的实体、情景、流程等，这就是“快速原型”的理念。

在这一步里，请将你们的项目构思用草图呈现，以能够说明基本意向和概念为佳，不要求十分精细和详实。然后利用草图向教师、学友或家长介绍你们的项目，获取反馈意见。

5. 调整构思

在获取了他人的反馈后，你们的项目构思有哪些调整？思考这些问题并建立文档，将所有的过程性信息进行记录（包括最初的构思、有价值的反馈意见、调整后构思等）。

▶ 评价标准

请根据项目实施的过程、效果以及成果展示交流的结果，对自己完成项目的情况进行客观的评价，并思考后续完善的方向。把评价结果和完善方案填写在下面的表格中。

评价条目	说明	评分（1~10分）	评分主要依据阐述	后续完善方向
小组合作	所有的团队成员都积极参与，对自己团队的作品充满热情			
主题挖掘	主题鲜明、贴近生活，着眼于解决实际问题			
项目分析	项目需求把握准确，划分清晰，给出的解决方案可行性高			
草图绘制	绘制的草图可以清晰地表达所描绘的实体、情景及流程，组内认可度高			
展示及反馈调整	展示效果好，展示过程通俗易懂，可以根据他人反馈及时调整项目，记录文档完整			

▶ 拓展项目

办公室一族由于长时间坐在计算机前，加上坐姿不正确等原因，容易造成肩膀、腰椎等各种不适，也容易导致大脑供血不足，出现头痛、头晕、耳鸣、听力下降或记忆力减退等症状。现在，某科技公司想请你开发一个项目，如果使用者坐姿不正确，这款产品就会通过震动的方式进行提醒。

以学习小组为单位，通过组内讨论，写出该项目的需求清单与可行性说明。

开源硬件项目的开发基础



工欲善其事，必先利其器。从“智能鸟蛋”项目分析可以知道，要对项目需求做出正确的可行性分析，然后一一实现，需要很强的技术支持。如果不了解当前有哪些开源硬件，以及它们具有哪些特点，支持哪些周边拓展模块，我们几乎不可能做出准确的项目规划。正如实现算法至少要掌握一门编程语言一样，开发硬件项目也需要先熟悉一款开源硬件，并且基于这一开源硬件，结合传感器、执行器和通信等周边硬件设备实现一些简单的应用，为开发项目打好基础。

问题与挑战

● 夜晚彩灯点亮的那一刻，绚丽夺目的灯光映入眼帘。行走在道路上，两边是被灯光包围的树，让人不禁开始想象，穿越这里，道路的另一端会是怎样的梦幻世界？还有公园里那些随着音乐律动、随着人流改变的互动彩灯，总让人流连忘返。如果要设计一款个性化的彩灯，需要掌握怎样的硬件技术？

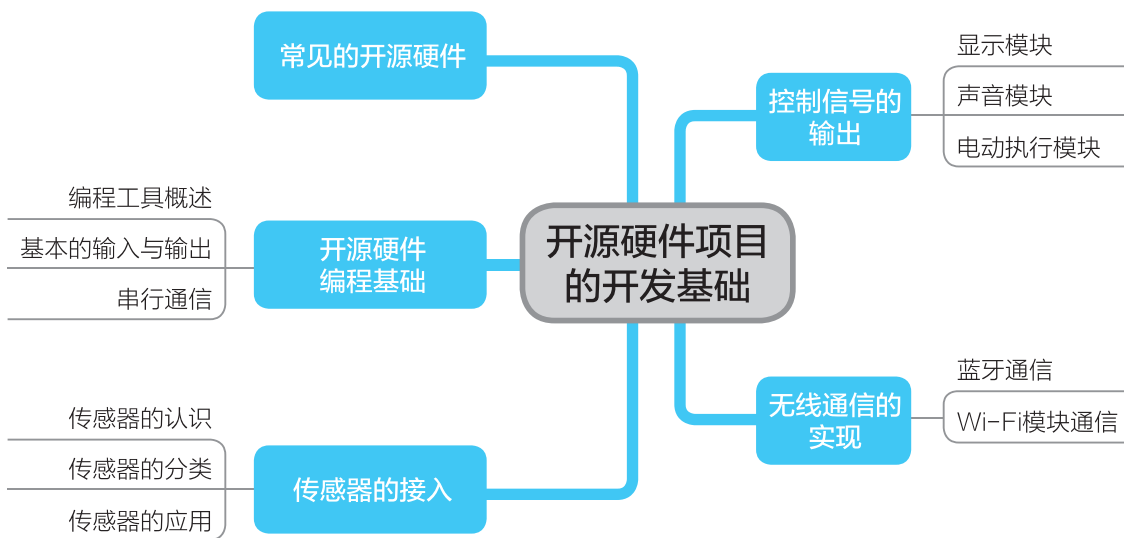
● 现代农业生产离不开环境控制，农业大棚控制系统是实现温室生产管理自动化、科学化的基本保证。控制系统结合作物生长规律，控制环境条件，使作物在不适宜生长的反季节中，获得比室外生长更优的环境条件，从而使作物达到优质、高产、高效的栽培目的。这样一款农业大棚控制系统，需要使用哪些硬件设备来监测环境中的各种参数？

● 随着通信网络的快速发展，物联网从早期的两个或多个设备之间的有线近距离数据传输，发展到各种无线传输方式，再到各类设备直接纳入互联网的“互联网+”时代。这些通信技术分别适用于哪些应用场景？设计一款与物联网相关的信息系统，需要使用哪些通信模块？

学习目标

1. 通过对常用开源硬件的分析，了解它们各自的特征，能够根据项目需求选择合适的开源硬件。
2. 了解各种开源硬件的设计工具及编程语言，并通过具体实例，掌握 micro:bit 的基本输入和输出的方法以及常见函数。
3. 了解各种常用的传感器、输出设备及通信设备，并能够利用这些硬件设备实现简单的项目。

★ 内容总览



3.1 常见的开源硬件

为了方便开发，人们把微控制单元（Microcontroller Unit，简称MCU）和必要的外围电路放在一块电路板上，使用业内流行的接口来连接各种传感器、输出、通信等硬件设备，这样的电路板称为开发板。开发板及配套的编程工具等资源组成了开源硬件的开发平台。借助开源硬件，各种传感器、输出、通信等硬件设备的连接变得越来越简单，开发者可以方便地实现各种创意，如交互艺术、智能玩具、家庭自动化等。

1. Arduino

Arduino 于 2005 年开发，起初是为了方便设计学院的大学生设计互动作品，后来发展为全球知名的开源硬件。

Arduino 包含硬件（各种型号的 Arduino 主板）和软件（Arduino IDE）。Arduino 有很多个硬件版本，分别适用于不同的开发需求，其中 UNO 是最常用的一种，如图 3.1.1 所示。通过扩展板，Arduino 可以便捷地连接传感器等各种各样的电子元件。



图3.1.1 Arduino UNO开发板

拓展链接

Arduino 的由来

Massimo Banzi 是意大利一家高科技设计学校的教师，他的学生经常抱怨找不到便宜好用的开发板。2005 年冬天，Massimo Banzi 认识了一名到学校做访问学者的芯片工程师 David Cuartielles。为了解决缺乏开发板的问题，两人决定自己设计，并邀请 Banzi 的学生 David Mellis 为开发板设计编程语言。结果没过几天，他们就设计出了开发板原型和编程工具，Banzi 并将这块开发板命名为 Arduino。

随后，Banzi、Cuartielles 和 Mellis 把设计图放到了网上，让更多的人使用。为了推广“开源”理念，他们决定采用 Creative Commons (CC) 的授权方式公开硬件设计图。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。

2. 树莓派

树莓派（Raspberry Pi，简称Rpi或者RasPi）是一款由英国的树莓派基金会（The Raspberry Pi Foundation）开发的基于Linux系统的迷你计算机。如图3.1.2所示，树莓派的尺寸只有一张信用卡大小。树莓派的开发目的是以低价的硬件和开源软件来推进中小学的计算机科学（Computer Science，简称CS）教育。它的运行速度比台式机稍慢，但已具备了计算机的基本功能，只需连接电视机（显示器）和键盘鼠标，就可以进行编辑文字、处理电子表格、编程等基本操作，还可以玩游戏、播放高清视频等。

树莓派支持Python、Java、C/C++、Perl等主流编程语言，可以用来开发各种如儿童识字机、数控机床的电机控制器、路由器、分布式计算机集群等项目。



图3.1.2 树莓派

3. micro:bit

micro:bit是一款由英国广播电视公司为青少年编程教育设计的入门级开发板（如图3.1.3），并由微软、三星、ARM、英国兰卡斯特大学等合作伙伴共同开发完成。它的尺寸（4cm×5cm）很小，上面嵌有5×5的红色LED阵列，还整合了2个可编程按钮、加速度计、电子罗盘、温度计等模块。具体硬件架构图如图3.1.4所示。

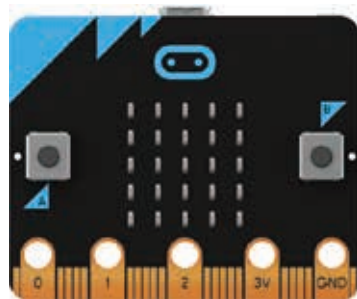


图3.1.3 micro:bit开发板

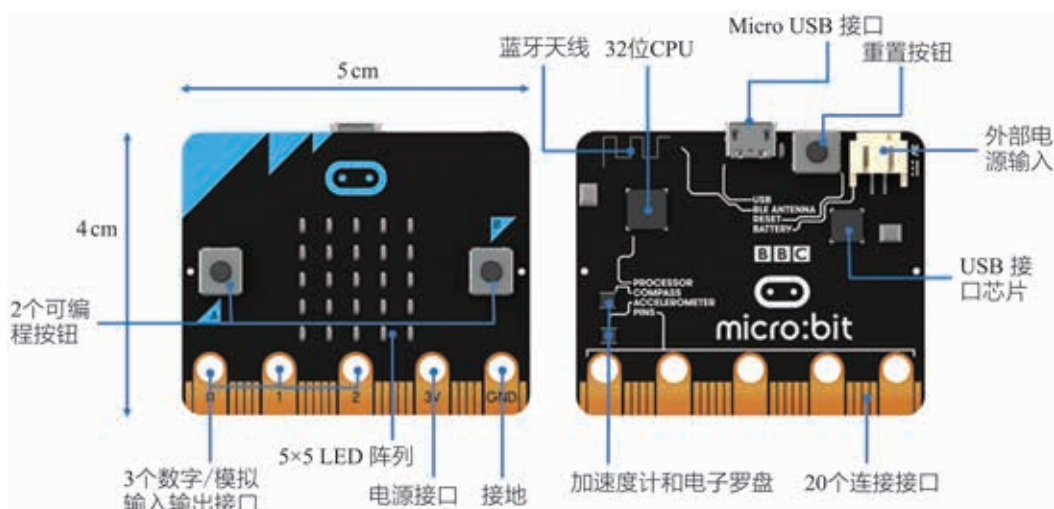


图3.1.4 micro:bit硬件架构图

micro:bit可以通过鳄鱼夹与各种电子元件互动，使用专用的扩展板后能连接多达20个外部模块，支持读取传感器数据、控制舵机与RGB灯带等，能够轻松胜任各种编程相关的教学与开发场景。micro:bit支持MicroPython、Java等编程语言，可以用于声光互动、机

器人控制、科学实验、可穿戴设备等开发，具体实例如图3.1.5所示。



图3.1.5 micro:bit开发应用

micro:bit具有电子模块集成度高、支持时下最热门的多种编程语言等多项优势，在节省教学成本的同时，又大大增加可玩性和拓展空间，适用于开源硬件的入门学习。

拓展链接

发光二极管

发光二极管（Light Emitting Diode，简称LED）是一种能够将电能转化为可见光的固态半导体器件，具有体积小、功耗低、亮度高、发热低、使用寿命长等特点。LED只能往一个方向导通（通电），一般用长脚表示正极，短脚表示负极。LED的光线波长、颜色跟所采用的半导体材料种类及掺入的元素杂质有关。

LED的光线颜色有红、黄、蓝、绿、白等，一般只能显示一种颜色，而全彩（RGB）LED灯实际上内置了红色（R）、绿色（G）、蓝色（B）三种颜色的灯珠，通过控制不同颜色灯珠的亮度，根据三原色原理混合调出多种颜色，如图3.1.6所示。



图3.1.6 单色LED和全彩LED

开源硬件除了上述以外，还有BeagleBone-Black、96board、虚谷号和掌控板等。根据项目开发的需求，一般可从尺寸、运算速度、板载等方面考虑，选择合适的开源硬件，以便更快地完成项目的研发。

思考与练习

1. 通过网络搜索，列举采用Arduino制作的项目实例。
2. 根据本节对micro:bit的介绍，思考micro:bit最适合的应用范围。

3.2 开源硬件编程基础

有了开源硬件，只需要将外接设备如LED、按钮等与开发板的引脚相连，再使用适当的编程语言，对外接设备进行读取和控制，就可以开始开源硬件项目开发之旅。

3.2.1 编程工具概述

随着开源硬件的发展，与开源硬件配套的编程工具不断涌现。以Arduino为例，除了官方提供的Arduino IDE以外，还有ArduBlock、Mixly（米思齐）和Mind+等。支持micro:bit的编程工具也有很多，如MakeCode、Mu Editor、BXY Python Editor、mPython、Mixly和Mind+等。

1. Arduino IDE

Arduino IDE是Arduino的官方编程工具，改写自开源软件Processing IDE（一款以图形表达、互动设计为特色的编程工具）。Arduino使用C/C++语言编写程序，把常见的微控制单元底层操作命令封装为各种函数进行调用，提供了大量的范例，对于初学者来说极易掌握。软件支持第三方库的导入，功能强大，使用灵活。Arduino IDE有多个版本，可以在Windows、Mac OS X、Linux三大主流操作系统上运行，采用文本方式进行编程。具体界面如图3.2.1所示。

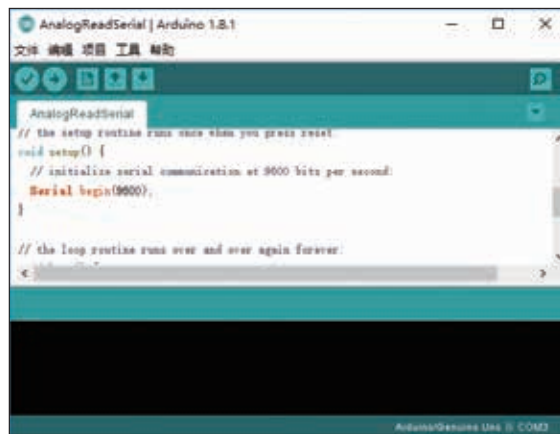


图3.2.1 Arduino IDE编程工具

拓展链接

集成开发环境

集成开发环境（Integrated Development Environment，简称IDE）是用于提供程序开发环境的应用程序，一般包括代码编辑器、调试器和图形用户界面工具。它是一种成熟的图形界面，有着各式各样的菜单和窗口，类似Web浏览、文字处理软件和绘图软件。

目前，用于编写Python程序的IDE较多，如Python自带的IDLE、Wing、PyCharm、Eric、Eclipse、Ulipd等。

2. Mixly

Mixly是一款开源的图形化开源硬件编程软件，同时支持Arduino和micro:bit，由北京师范大学创客教育实验室基于Blockly开发。Mixly使用图形化模块代替复杂的文本操作，在使用时直接拖动模块就可以编程，不同功能的模块以不同的颜色表示，如图3.2.2所示。这种方式加强了编程的可视化和交互性，降低了编程的门槛，即使没有任何编程经验的人也可以编写出简单的程序。

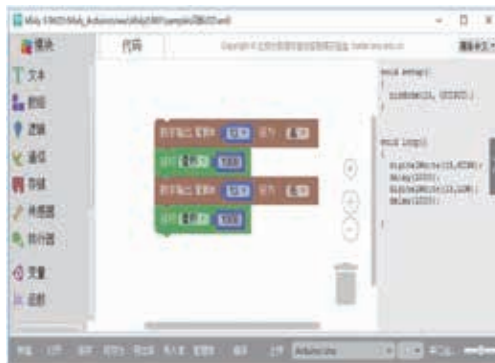


图3.2.2 Mixly编程工具

3. BXY Python Editor

BXY Python Editor（简称BXY）是由国内某创客团队自行研发的一款专门为MicroPython设计的编程工具。BXY可以运行于Windows及Mac OS X操作系统，采用文本方式进行编程，内置了各种基础操作库，界面简洁，操作便利，具体编辑界面如图3.2.3所示。

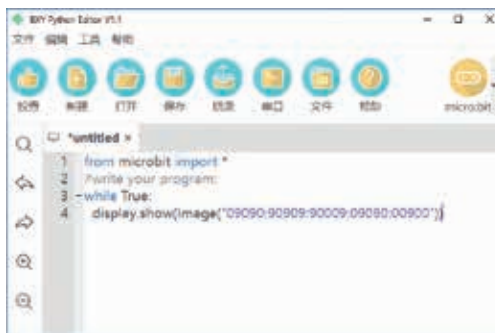


图3.2.3 BXY Python Editor编程工具

在开发开源硬件项目时，要根据开发者各自的特长及项目的具体功能选择合适的编程工具。

问题与讨论

通过对文本方式的开发环境和图形化开发环境的实践，你认为文本方式的开发环境是否会被图形化开发环境所取代？谈谈你对这两类开发环境各自优缺点的看法。

实践与体验

加载第一个程序

通过加载第一个程序，熟悉BXY开发环境的使用。

实践内容：

通过micro:bit内置的LED阵列，滚动显示“Hello!”。

实践步骤:

1. 通过USB线将计算机和micro:bit连接，如图3.2.4所示。

2. 打开BXY开发环境，输入如下代码，或直接打开micro:bit示例Basics中的 Hello 程序。

```
from microbit import * #引用micro:bit库
display.scroll("Hello!") #LED阵列滚动显示Hello!
```



图3.2.4 计算机和micro:bit连接

3. 调试并烧录程序。

在BXY开发环境中（如图3.2.5）；可以单击工具栏中的“检查”按钮，检测编辑框中的程序是否有语法错误，随后弹出窗口：“语法检测完毕！”若有语法错误，则用粉色表明错误行。

（注：语法检查只检测语法，不会检测逻辑等错误。）

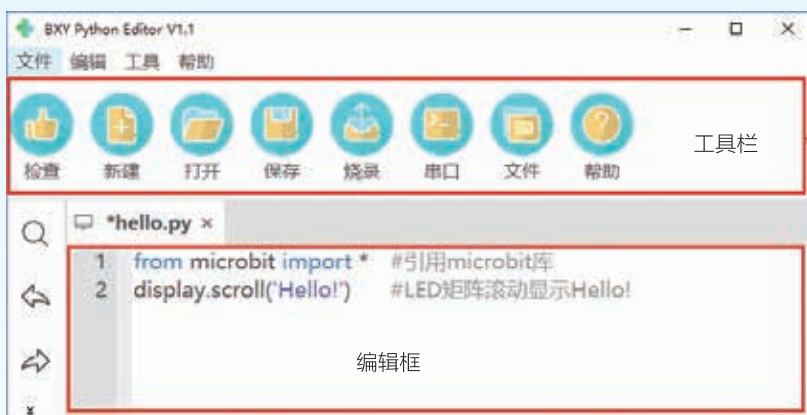


图3.2.5 BXY Python Editor开发环境

若micro:bit未与计算机连接，点击工具栏中的“烧录”按钮，可以将程序以HEX文件格式保存到计算机磁盘中。使用USB线将micro:bit与计算机连接，在计算机上出现如图3.2.6所示的MICROBIT磁盘。

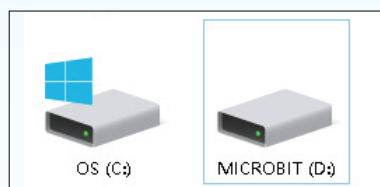


图3.2.6 名为MICROBIT的磁盘

将保存的HEX格式的文件复制到MICROBIT磁盘，micro:bit板上的LED会不断闪烁，说明正在将程序烧录至micro:bit。烧录成功后，在micro:bit的LED阵列上滚动显示“Hello!”。

（注：若已安装micro:bit驱动并将micro:bit与计算机连接，点击“烧录”按钮可直接将程序烧录至micro:bit。）

思考讨论:

若要在micro:bit中循环显示“Hello!”，该如何修改程序？

3.2.2 基本的输入与输出

基于开源硬件来开发项目，核心工作是对开发板的引脚（其实就是MCU的引脚）进行编程，连接各种外接设备（如LED灯、温度传感器等）。这些引脚是开发板与外接设备沟通的桥梁，开发板通过这些引脚可以驱动外接设备或读取外接设备的数据，实现对数据信号的输入与输出（Input/Output，即I/O）。

如果开发板的某个引脚连接了传感器，那么要设置这个引脚为“输入引脚”，通过读取引脚电信号来获取传感器的数据。如果引脚连接的是LED灯或电机模块之类的执行设备，那么它就要作为“输出引脚”，通过对该引脚输出特定的数据信号来控制执行器执行某些行为。不同的引脚，具备的功能也不同，具体需要参照开发板的引脚说明。

1. 数字输出

当开发板控制执行器时，通常以高电平和低电平的形式输出控制信号，这类信号叫作数字信号，用于表示不连续的信号。在开发板中，采用二进制形式表示这类信号，数字“1”表示高电平，数字“0”表示低电平，如图3.2.7所示。开发板通过相应的引脚，实现数字信号输出的行为称为数字输出。通过数字输出，可以控制指示灯的亮与灭、继电器的吸合与释放、电机的启动与关闭以及脉冲信号的计数与定时等。

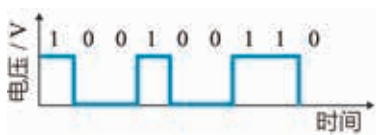


图3.2.7 数字信号

●●● 例1 “闪烁的小星星”

实现效果：利用开发板实现LED发光二极管循环闪烁。

器材清单：①micro:bit；②LED发光二极管；③USB线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，再将LED发光二极管的长脚与micro:bit的2号引脚连接，短脚与micro:bit的GND引脚连接。具体连接如图3.2.8所示。

(2) 程序编写并烧录

在BXY开发环境中，输入如图3.2.9所示的代码，点击“烧录”按钮，micro:bit板上的LED将开始闪烁。等到停止闪烁，表示程序烧录成功。

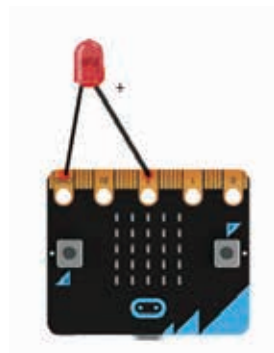


图3.2.8 “闪烁的小星星”硬件连接



图3.2.9 “闪烁的小星星”离线版IDE编程

在本实例中，通过“pin2.write_digital(1)”语句来设置2号引脚值为1（即高电平），将LED点亮。通过“pin2.write_digital(0)”语句来设置2号引脚值为0（即低电平），将LED熄灭。

2. 数字输入

在开源硬件项目设计时，不但能够通过二进制的逻辑“1”和“0”的数字量来控制外围设备，也能够通过接收外围设备的高、低逻辑电平信号，实现数字信号的输入。如开关的接通（数字1）与断开（数字0）状态，就是一种基础的数字输入形式。

●●● 例2 “触动的星星”

实现效果：利用 micro:bit 的 0 号和 1 号引脚内置的触摸开关，实现 LED 发光二极管的点亮和熄灭。

器材清单：① micro:bit；② LED 发光二极管；③ USB 线；④ 鳄鱼夹连接线。

操作步骤：

（1）硬件连接

通过 USB 线将 micro:bit 连接到计算机，再将 LED 发光二极管的长脚与 micro:bit 的 2 号引脚连接，短脚与 micro:bit 的 GND 引脚连接。

（2）程序编写

在编程工具中输入如下代码：

```

from microbit import * #引用micro:bit库
while True: #循环
    if pin0.read_digital()==1:
        #引脚被触摸时，读取的数字信号值为“1”，否则读取的数字信号值为“0”
        pin2.write_digital(1) #设置2号引脚值为1，LED点亮
    if pin1.read_digital()==1:
        pin2.write_digital(0) #设置2号引脚值为0，LED熄灭

```

在本实例中，通过“pin0.read_digital()”语句读取0号引脚的数字信号，通过“pin1.

read_digital()”语句读取1号引脚的数字信号。当0号引脚读取的逻辑值为“1”时，设置2号引脚值输出高电平（数字1），LED点亮。当1号引脚读取的逻辑值为“1”时，设置2号引脚值输出低电平（数字0），LED熄灭。

（3）调试并烧录

将程序烧录至micro:bit，烧录完成后，将鳄鱼夹连接线一头的鳄鱼夹与micro:bit的GND引脚连接。当另一头的鳄鱼夹与0号引脚相连时，0号引脚被触摸，LED点亮；与1号引脚相连时，1号引脚被触摸，LED熄灭，如图3.2.10所示。

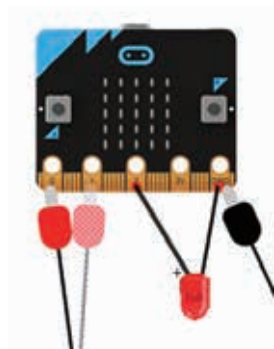


图3.2.10 “触动的星星”实例调试

3. 模拟输入

模拟信号（Analog Signal）就是指用连续变化的物理量表示的信息，如声音的高低、温度的变化等，其信号的幅度、频率、相位随时间连续变化。数字信号与模拟信号的波形比较，如图3.2.11所示。

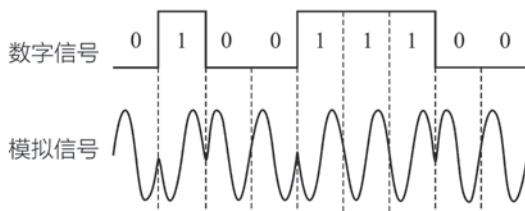


图3.2.11 数字信号和模拟信号比较

数字信号与模拟信号最直观的区别，就是数字信号仅有两种可能，即0和1，而模拟信号可以在一定的范围内变化。开发板从指定的模拟引脚读取模拟量电压值，并将其进行量化。micro:bit默认采用10位量化精度（即范围是0~1023），参考电压为3.3V，如读取的值为511，则其表示1.65V。

●●● 例3 亮度的大小

实现效果：利用光敏传感器模块，采集环境光的数据信号，并通过串口输出。

器材清单：①micro:bit；②micro:bit扩展板；③模拟光敏传感器模块；④USB线。

操作步骤：

（1）硬件连接

通过USB线将micro:bit连接到计算机，再将光敏传感器模块通过扩展板与micro:bit的VCC、GND和2号引脚连接，具体连接如图3.2.12所示。

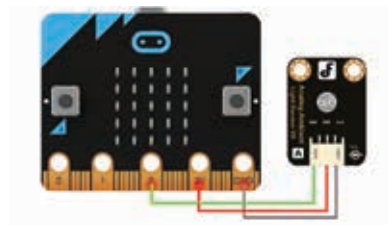


图3.2.12 “亮度的大小”实例硬件连接

（2）程序编写

在编程工具中输入如下代码：

```
from microbit import *
while True:
    intensity = pin2.read_analog()
```

```
print(intensity)
sleep(1000)
```

在本实例中，通过“pin2.read_analog()”语句来读取2号引脚的模拟信号值，并通过print(intensity)在串口输出采样的数据。

(3) 调试并烧录

将程序烧录至micro:bit，点击BXY工具栏中的“串口”按钮（如图3.2.13），打开串口监视器，输出如图3.2.14所示的信息。从图中可以看到，每隔一秒显示micro:bit从光敏传感器读取的亮度值。



图3.2.13 BXY工具栏

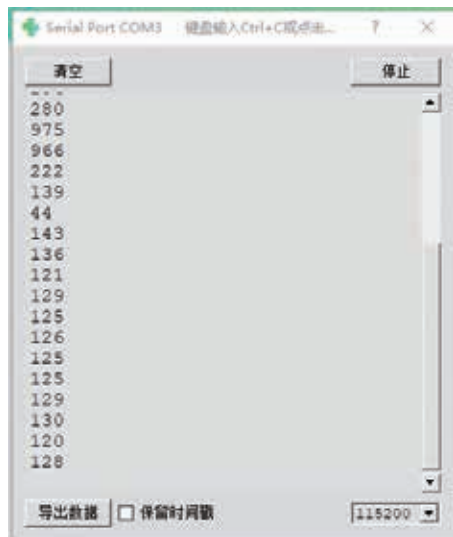


图3.2.14 串口监视器输出的亮度值

拓展链接

扩展板的使用

在设计开源硬件项目时，常常需要借用面包板来连接其他设备，这就需要先掌握一定的电子知识，以便搭建复杂的电路。而使用扩展板则只需要通过连接线，就可以方便地将各种模块接插到扩展板上。如图3.2.15所示的是一款micro:bit扩展板。

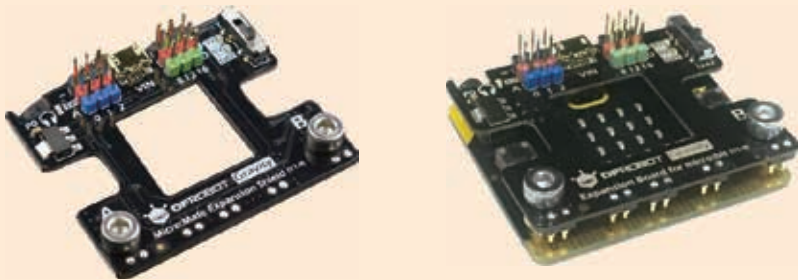


图3.2.15 micro:bit扩展板

通过扩展板转换，可以把开发板的引脚转化为更方便插接的“标准”接口，并且每一个接口中已经包含电源（VCC）和接地（GND）。通常情况下，习惯用红色来代表电源（VCC）、黑色代表接地（GND）、其他颜色代表信号（signal）。与扩展板连接的数据线也通常遵守这样的颜色习惯。常用的传感器模块、执行器模块等组件往往提供了类似的接口，以便与扩展板进行连接。如图3.2.16所示的是一款模拟光敏传感器模块和专用的连接数据线。



图3.2.16 光敏传感器组件及专用数据线

4. PWM（模拟）输出

与模拟输入功能对应的是模拟输出功能，但开发板并不能输出真正意义上的模拟信号，而是通过一种特殊的方式“模拟”出来，这种方式叫作脉冲宽度调制（Pulse Width Modulation，简称PWM），被广泛应用在测量、通信、功率控制与变换等领域。

PWM脉冲宽度调制信号有三个基本参数：脉冲宽度变化幅度（最小值/最大值）、脉冲周期（1秒内脉冲频率个数的倒数）、电压高度（如0~3.3V），如图3.2.17所示。



图3.2.17 PWM脉冲宽度调制示意图

PWM接口输出的电压值是通过输出的高电平和周期时间来进行计算的。

$$\text{输出电压值} = \text{高电平时间} / \text{周期时间} \times \text{最大电压值}$$

其中，“高电平时间/周期时间”就是PWM信号的占空比。通过图3.2.18所示的三个方波可以更形象地了解PWM脉冲宽度调制。

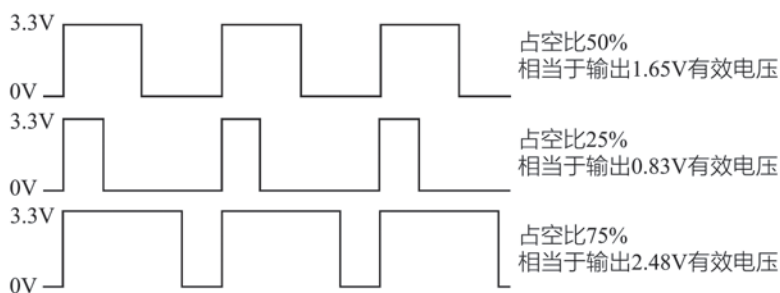


图3.2.18 三个PWM脉冲宽度调制输出示例

这些信号都具有相同的周期，但具有不同的占空比。在MicroPython编程工具中，可以使用write_analog()函数产生不同的PWM脉冲宽度调制信号。例如，图3.2.18中的三个不同的PWM脉冲宽度调制信号的产生方法如下：

- 第一个由 write_analog(511)生成，它的占空比为50%，高低电平各占一半，其结果相当于输出1.65V。
- 第二个由 write_analog(255)生成，占空比为25%。它的效果如同引脚上输出0.83V。
- 第三个由 write_analog(767)生成，占空比为75%。它的输出电压是第二个信号的三倍，相当于在引脚上输出2.48V。

●●● 例4 “呼吸灯”

实现效果：利用PWM脉冲宽度调制信号，实现LED灯像呼吸一样有节奏地由暗到亮的变化。

器材清单：①micro:bit；②micro:bit扩展板（可选）；③LED发光模块（或LED发光二极管）；④USB线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，采用LED发光二极管的连接方式与“闪烁的小星星”实例类似，选择LED发光模块则通过扩展板与micro:bit的2号引脚连接。

(2) 程序编写

在编程工具中输入如下代码：

```
from microbit import *           #引用micro:bit库
while True:                       #循环
    for n in range(0,1023,10):     #n的值不断增大
        pin2.write_analog(n)
        sleep(50)
    for n in range(1023,0,-10):   #n的值不断减小
        pin2.write_analog(n)
        sleep(50)
```

在本实例中，通过“for n in range(0,1023,10)”和“for n in range(1023,0,-10)”语句来产生不断变化的n值。通过“pin2.write_analog(n)”语句来设置2号引脚输出为PWM脉冲宽度调制信号，从而实现LED不断由灭到亮、由亮到灭的效果，产生“呼吸”一样变化的节奏。

拓展链接

micro:bit 引脚和 Arduino UNO 引脚

micro:bit的每个引脚是一个对象，并有编号。因此，如果要调用N引脚，可以用PinN表示。micro:bit和Arduino UNO的具体引脚如图3.2.19所示，其中micro:bit总共有19个引脚，编号为0—16和19—20，引脚17和18未引出。

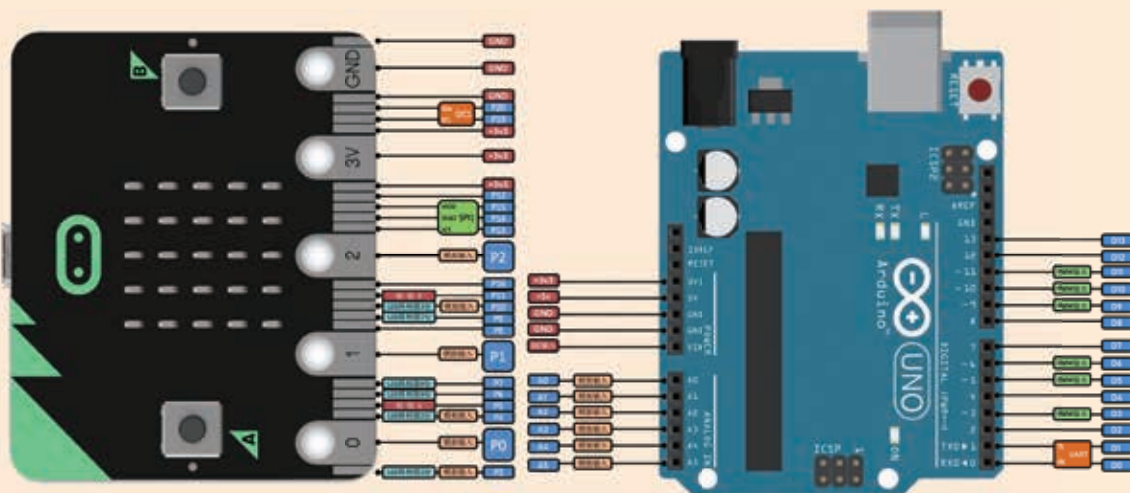


图3.2.19 micro:bit和Arduino UNO引脚说明

micro:bit 引脚和 Arduino UNO 引脚的分类如表 3.2.1 所示。

表 3.2.1 micro:bit 引脚和 Arduino UNO 引脚比较

引脚分类	micro:bit	Arduino UNO
数字输入	P0—P16, P19—P20	D0—D13
数字输出	P0—P16, P19—P20	D0—D13, A0—A5
模拟输入	P0—P4, P10	A0—A5
模拟输出	P0—P16, P19—P20	D3, D5, D6, D9, D10, D11
I2C 总线	SDA=P20, SCL=P19	SDA=A4, SCL=A5
SPI 总线	MOSI=P15, MISO=P14, SCK=P13	MOSI=D11, MISO=D12, SCK=D13
UART	所有端口支持软件 UART	RX=D0, TX=D1

注: micro:bit 在引脚 0, 1 和 2 上安装了外部弱上拉 (阻值为 10M 欧姆), 能实现触摸感应工作。

3.2.3 串行通信

在开源硬件项目的开发和应用中, 经常需要对开发板进行调试或者双向通信。常用的方式是利用数据线通过串行通信技术进行通信, 即用一条数据线, 将数据一位一位地依次传输, 每一位数据占据一个固定的时间长度, 如图 3.2.20 所示。



图 3.2.20 串行通信

大部分开发板都集成了 UART、I2C 和 SPI 三种常见的串行通信方式。通过这些通信接口, 开发板既可以和计算机进行通信, 又可以和基于这些通信方式的传感器或模块进行通信。

1. UART

通用异步收发传输器 (Universal Asynchronous Receiver Transmitter, 简称 UART) 使用 RX 和 TX 两条线实现数据的通信。为了确保通信可靠, 需要在通信两边接共地, 因此, 较为可靠的 UART 通信一般需要 3 条线。如图 3.2.21 所示, 其中 TX 是发送端, RX 是接收端, 通信双方使用交叉互联, RX 接对方 TX, TX 接对方 RX, 并共用电源地 (GND)。

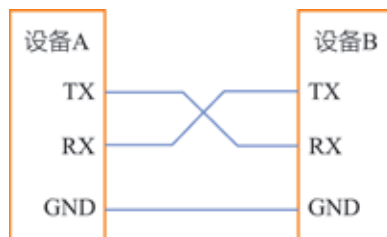


图 3.2.21 UART 串行通信示意图

在开发板与其他设备通信的过程中, 数据以电平的高低形式 (即数字信号) 传输, UART 通信也是如此, 当输出数据时, 发送端会输出一连串的数字信号, 这些数字信号

称为数据帧。例如，通过平台向其他设备发送数据“A”时，实际发送的数据格式如图3.2.22所示。

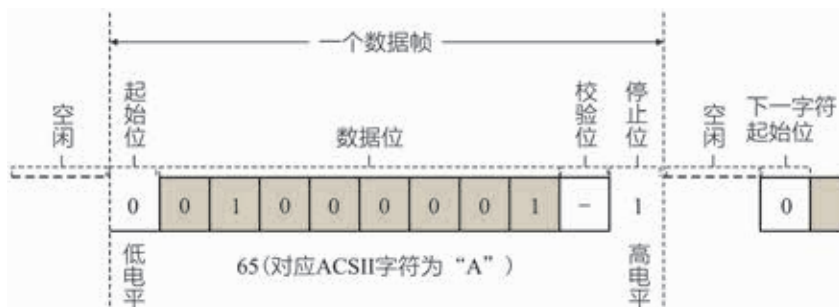


图3.2.22 串口数据帧格式

- 起始位：先发出一个逻辑“0”的信号，表示传输字符的开始。
- 数据位：紧跟起始位之后，承载了实际发送的数据的数据段。通常采用ASCII码。从最低位开始传送，靠时钟来定位。

- 校验位：校验位是串口通信中一种简单的检错方式。可以没有校验位。
- 停止位：它是一个字符数据的结束标志；可以是1位、2位的高电平（逻辑“1”）。
- 空闲位：处于逻辑“1”状态，表示当前线路上没有数据的传送。

在串行通信时，还需要设定串口通信的速率，即波特率。通信双方需要约定使用一致的波特率才能正常通信。

拓展链接

波特率

在串行传输中，携带数据信息的信号单元叫作码元。每秒钟通过串口传输的码元个数称为码元传输速率，简称波特率，其单位用波特（Bd）表示。例如，每秒钟传送240个码元，而每个码元格式包含10位（1个起始位，1个停止位，8个数据位），这时的波特率为240Bd。

串行传输中常用的波特率为1200、2400、4800、9600、19200、38400、115200Bd等。如micro:bit与计算机进行串行通信，为了直观地看到调试结果，需要使用micro:bit虚拟串口功能，其默认波特率为115200Bd。波特率越大，代表传输信号频率越高。相同的电缆下，信号频率越高时，信号在传输线上的损耗也就越大。因此，波特率越高，传输的距离也越短。

开发板和计算机进行UART串行通信时，一般通过一个转换芯片与计算机的USB连接。该转换芯片再通过USB接口在计算机上虚拟出一个用于与开发板通信的串口。因此，当使用USB线将开发板与计算机连接时，两者之间便建立了UART串口连接。通过此连接，开发板就可以与计算机相互传送数据了。

在例3“亮度的大小”实例中，开发板通过光敏传感器采集环境亮度，再通过串口向计算机输出采集到的数据。当然，计算机也可以通过串口向开发板发送指令。

●●● 例5 键盘的控制

实现效果：利用键盘中的T键和F键来控制LED灯的亮和灭。

器材清单：①micro:bit；②micro:bit扩展板（可选）；③LED发光模块（或LED发光二极管）；④USB线。

操作步骤：

（1）硬件连接

通过USB线将micro:bit连接到计算机，采用LED发光二极管的连接方式与“闪烁的小星星”实例类似，选择LED发光模块则可以通过扩展板与micro:bit的2号引脚连接。

（2）程序编写

在编程工具中输入如下代码：

```

from microbit import *           #引用microbit库
while True:                       #循环
    if uart.any():                #判断是否有数据
        incoming = str(uart.readall(), "UTF-8") #读入串行的全部数据
        incoming=incoming.strip('\n') #移除字符串头尾的换行符
        if incoming=='T':        #如果传输的数据为“T”
            pin2.write_digital(1) #点亮LED
            print("Turn on the light") #向串口发送“Turn on the light”
        elif incoming=='F':      #如果传输的数据为“F”
            pin2.write_digital(0) #熄灭LED
            print("Turn off the light") #向串口发送“Turn off the light”
        else:                     #向串口发送传输内容有误
            print("err")
    
```

（3）调试并烧录

将程序烧录至micro:bit。查看“设备管理器”的“端口（COM和LPT）”项，找到名称类似为“mbed Serial Port (COM3)”的设备。如图3.2.23所示，判定当前COM端口号为COM3。



图3.2.23 串口COM端口号

- 使用串口调试助手调试

打开串口调试助手，当输入“T”或“F”时，micro:bit上的LED灯点亮或熄灭，并且串口监视器还能接收到“Turn on the light”或“Turn off the light”的提示。

- 利用Python编程调试

打开Python IDE，依次输入如下代码：

```
import serial                #导入serial模块
ser = serial.Serial()
ser.baudrate = 115200       #设置波特率
ser.port = 'COM3'           #设置串口号
ser.open()                  #开启串口
while True:                 #循环
    f=input("输入T开灯，输入F关灯") #等待输入
    if f=='T':               #判断输入字符
        ser.write('T'.encode()) #按指定的编码格式编码字符串
    elif f=='F':
        ser.write('F'.encode())
    elif f=="O":
        break                #退出循环
ser.close()                 #关闭串口
```

2. I2C

I2C（Inter-Integrated Circuit），也可以简称为IIC。与UART一对一的通信方式不同，I2C是同步半双工通信协议，使用时只需要一条数据线SDA、一条时钟线SCL，即可实现一个主机（Master）和最多128个I2C从机（Slave）设备进行通信。



图3.2.24 I2C总线示意图

I2C上的所有通信都由主机发出，主机负责启动和终止数据传送，同时还要输出时钟信号；总线上的从机设备间彼此独立，有各自的地址，主机可以通过这些地址向总线上的任一从机设备发出连接，从机响应通信请求并建立连接后，便可进行数据传输。

I2C是一种半双工通信方式，总线上的设备通过SDA引脚传输通信数据，数据的发送和接收由主机控制。在通信过程中，通信的速率也由主机控制完成，主机通过SCL引脚输出时钟信号，供总线上的所有从机使用。

如图3.2.25所示都是采用I2C总线通信的外围设备，这种通信方式具有接口线少、控制方式简单、器件封装形式小、通信速率较高等优点。

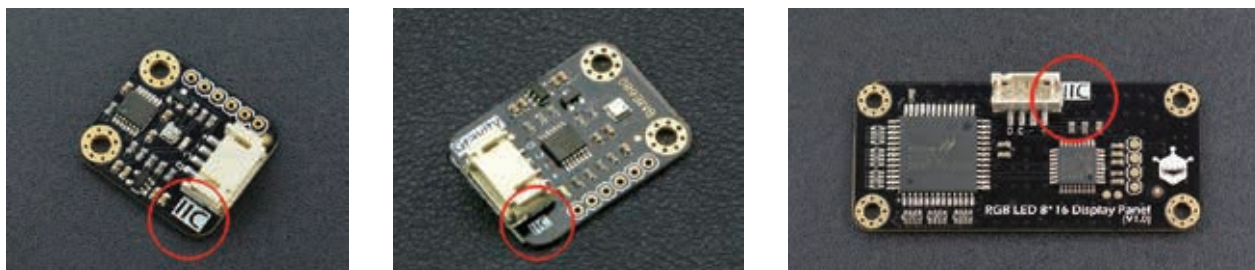


图3.2.25 采用I2C总线通信的外围设备

拓展链接

同步通信、异步通信和全双工、半双工

根据同步方式的不同，串行通信可分为同步通信和异步通信。

异步通信中，收发双方有各自的时钟。接收方不知道数据何时到达，发送方发送的时间间隔也不一致，通过数据起始位和停止位实现信息同步。这种通信通常以一个字节为一组，在每一个字节开始和结束的地方加上标识，即加上开始位和停止位。因为每一字节都需要添加辅助位，所以异步通信的效率较低。

同步通信中，双方使用频率一致的时钟，将多个字节的数据合并为一组，通过特定字符作为开始和结束的标识。发送方要以固定的频率发送数据，而接收方要时刻做好接收数据的准备。因为分组较大，需要添加的辅助位较少，因此效率较高，适合用于对速度要求高的传输，但这种方式对时序的要求较高。

根据信息的传送方向，串行通信还可以分为单工、半双工和全双工三种。

在通信过程的任意时刻，信息只能由一方A传到另一方B，则称为单工；信息既可由A传到B，又能由B传到A，但只能存在一个方向上的传输，则称为半双工传输；线路上存在A到B和B到A的双向信号传输，则称为全双工。

3. SPI

串行外围设备接口（Serial Peripheral Interface，简称SPI）是一种全双工同步串行通信接口，该接口一般使用4条线：串行时钟线SCK、主机输入/从机输出数据线MISO、主机输出/从机输入数据线MOSI和低电平有效的从机选通线SS或CS。SPI设备中通常有如表3.2.2所示的几个引脚。

表3.2.2 SPI通信引脚

引脚名称	说明
MISO (MasterInSlaveOut)	主机数据输入，从机数据输出
MOSI (MasterOutSlaveIn)	主机数据输出，从机数据输入

续表

引脚名称	说明
SCK (SerialClock)	通信同步的串行时钟信号，由主机产生
SS (SlaveSelect) 或 CS (ChipSelect)	从机使能 (Enable) 信号，由主机控制

在SPI总线中也有主、从机之分，主机负责输出时钟信号及选择通信的从设备。时钟信号会通过主机的SCK引脚输出，提供给通信从机使用。而对于通信从机的选择，由从机上的CS引脚决定，当CS引脚为低电平时，该从机被选中；当CS引脚为高电平时，该从机被断开，数据的收、发通过MISO和MOSI进行。多个SPI设备的连接方法如图3.2.26所示。

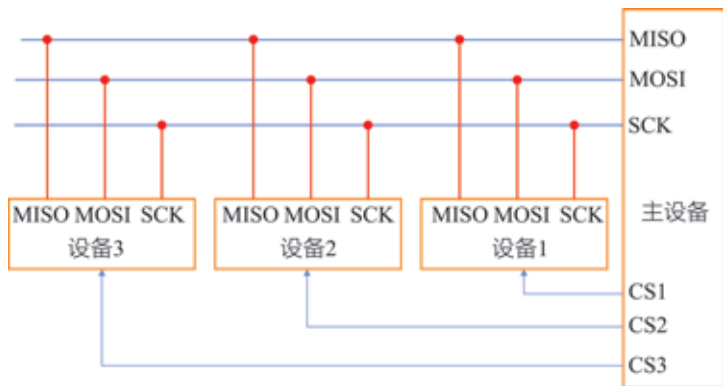


图3.2.26 SPI总线示意图

拓展链接

并行通信

并行通信即一组数据的各数据位在多条线上同时被传输。

如图3.2.27所示，在并行通信中，一个字节（8位）数据在8条通信线上同时传输。例如，传送一个字节，串行通信至少需要8个时间周期，而并行通信只需要1个时间周期。



图3.2.27 并行与串行通信示意图

因此，并行通信传输速度快、效率高，多用在实时、快速的场合，但并行通信时要求的通信线多、成本高，只适合于短距离、高速率的数据传送，通常要求传输距离小于30米，不宜进行远距离通信。

拓展链接

micro:bit 常用输入/输出函数

在 MicroPython 的 microbit 库中，针对引脚的常用输入/输出（I/O）函数如表 3.2.3 所示。

表 3.2.3 micro:bit 库中常用输入/输出（I/O）函数

函数	描述
write_digital(value)	设置引脚高、低电平输出。若 value 为 1，将引脚设置为高电平；若为 0，就将其设置为低电平
read_digital()	读取引脚状态为高、低电平。若引脚为高电平，则返回 1；若引脚为低电平，则返回 0
write_analog(value)	设置引脚 PWM 脉冲宽度调制输出。value 作为 PWM 输出的值。value 可以是 0（0% 占空比）与 1023（100% 占空比）之间的数
read_analog()	读取引脚的电压，并将其作为 0（0V）和 1023（3.3V）之间的整数返回
is_touched()	检测引脚是否被触摸（只适用 0，1 和 2 引脚）。若引脚被触摸，则返回 True；否则，返回 False

前面实例中已经涉及了引脚的数字输入与输出，即 read_digital() 和 write_digital() 函数，也涉及了引脚的模拟输入与输出，即 read_analog() 和 write_analog() 函数。is_touched() 函数是针对 micro:bit 的 0，1 和 2 引脚，若这三个引脚被触摸则返回 True，否则返回 False。micro:bit 还有内置按钮的相关函数，如表 3.2.4 所示。

表 3.2.4 micro:bit 内置按钮相应函数

函数	描述
is_pressed()	若内置 button_a 或 button_b 被按下，则返回 True；否则，返回 False
was_pressed()	若指定的按钮在开机或复位之后被按下，则返回 True；否则，返回 False
get_presses()	返回开机或复位后 button 被按下的次数，并且调用此方法后，会将计数置为 0

III 实践与体验 III

小夜灯

制作一个小夜灯：当夜晚关掉房间的大灯时，小夜灯能自动开启，从暗到明地逐渐点亮；当环境光线渐渐变亮时，它能从明到暗地自动调节灯光亮度，直到关闭。

实践内容：

利用光敏传感器、LED 模块、micro:bit 以及扩展板，制作能根据环境光线自

动控制的小夜灯。

实践步骤：

1. 连接光敏传感器模块、LED发光模块和micro:bit平台。
2. 绘制程序对应的流程图。
3. 利用MicroPython开发环境，编写程序。
4. 调试并烧录程序。

思考讨论：

分组讨论，对程序进行怎样的迭代，能设计出更时尚的小夜灯。参考资料，利用micro:bit内置的传感器，实现更多功能。

思考与练习

1. 在实现“闪烁的小星星”的例子中，要实现：2号引脚值为0时，LED点亮；2号引脚值为1时，LED熄灭。LED发光二极管和micro:bit的连接要做怎样的改动？

2. 查阅资料，统计常见的开发板的数字口的个数或对应的编号。

3. 例4的效果还未达到理想的状态，因为灯光强弱的过渡不够柔和。为了实现更真实的“呼吸灯”的效果，可以采用正弦函数进行采样，得到一个范围在 $[0, 1]$ ，按正弦函数规律变化的数组。具体步骤如下：

- ①使用for循环，定义采样间隔变量 x （如 $0^\circ \sim 179^\circ$ ）。
- ②将角度间隔（ 1° ）转换为弧度间隔，并对正弦函数进行采样。
- ③对采样值进行适当转换，缩放到应用需要的亮度取值范围之内。
- ④使用上述转换后的值驱动外设。
- ⑤适当延时以保证正弦采样正常进行。

采用正弦函数，改写“呼吸灯”实例，烧录至开发板，对程序进行调试，观察两种方式实现的效果差异。

3.3 传感器的接入

设计一个智能化的开源硬件项目，首先要让其拥有“触觉”“味觉”和“嗅觉”等“感官”，从而与外部世界进行联系。而这种联系的中介，则是各种各样的传感器。

3.3.1 传感器的认识

传感器（transducer/sensor）是一种检测装置，一般由敏感元件、转换元件、其他辅助元件三部分组成。它通过敏感元件感受（或响应）和检出被测对象的待测信息，如位移、速度、力、温度、湿度、流量、光、声、化学成分等非电学量；通过转换元件将敏感元件所感受（或响应）到的信息按一定规律转换成电信号或其他所需形式的信息，以满足信息传输、处理、存储、显示、记录和控制等方面的需求；辅助元件主要包括信号调节与转换电路及其激励电路等。传感器的结构如图3.3.1所示。

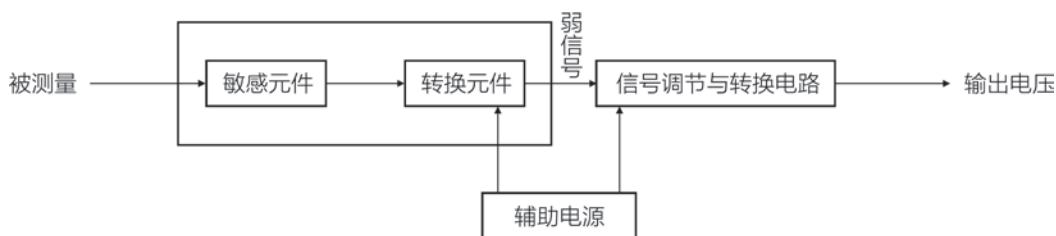


图3.3.1 传感器的结构示意图

国家标准GB/T 7665—2005对传感器的定义是：能感受被测量并按照一定的规律转换成可用输出信号的器件或装置，通常由敏感元件和转换元件组成。

例如，光敏传感器广泛应用于路灯、太阳能草坪灯、监控器、防盗钱包、光控音乐盒、音乐蜡烛、人体感应开关等电子产品光自动控制领域。光敏传感器中的光敏电阻，是利用半导体的光电效应制成的一种电阻值随入射光的强弱而改变的电阻器。如图3.3.2所示的就是一种常用的光敏电阻。电阻值的测量一般是利用串联已知阻值的电阻构成分压电路，并施加已知大小的激励电压，通过测量已知阻值的电阻上的分压值，可以计算得出被测电阻的阻值，如图3.3.3所示。设施加的激励电压为 V_{in} ，光敏电阻的阻值为 R_t ，串联电阻阻值为 R_s 。根据欧姆定律 $U_1:U_2=R_1:R_2$ ，因此该串联电阻上的分压值为：

$$V_{out} = V_{in} \times \frac{R_s}{R_t + R_s}$$



图3.3.2 光敏电阻

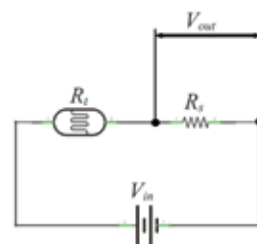


图3.3.3 串联测量法

问题与讨论

在“亮度的大小”实例中，用micro:bit的1号信号引脚来测量分压电阻的值，以间接读取光敏电阻的电压值，从而获取当前环境的光线亮度。如果用micro:bit的1号信号引脚直接读取光敏电阻的电压值，具体硬件连线应该怎么变化？实例的程序要做怎样的修改？

3.3.2 传感器的分类

根据基本感知功能的不同，通常把传感器分为温度、光线、气味、压力、磁力、湿度、声音和味觉等传感器。为了便于理解，将传感器的功能与人类的五大感觉相类比，如表3.3.1所示。

表3.3.1 传感器与人类感觉的类比

人类感觉	传感器名称
视觉	光线传感器
听觉	声音传感器
嗅觉	气味传感器
味觉	味觉传感器
触觉	温度、压力传感器

这些传感器由其对应的敏感元件组成。如图3.3.4所示的这些器材分别是组成温度传感器、气味传感器、磁力传感器和湿度传感器的敏感元件。

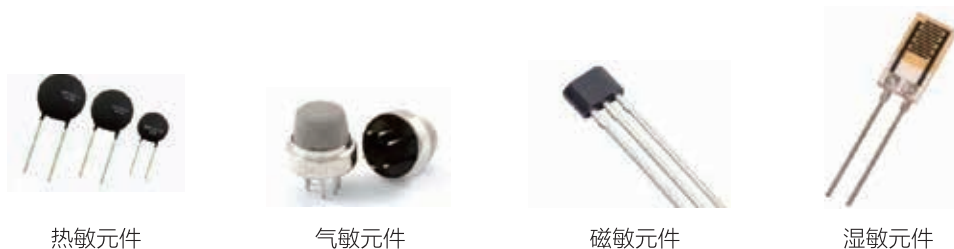


图3.3.4 各种常见的敏感元件

传感器除了根据感知功能分类以外，还可以根据输出信号的性质分为模拟传感器、数字传感器、开关传感器等，如在“亮度的大小”实例中的光线传感器就属于模拟传感器。传感器还可以按制造工艺、构成、作用形式等多种方式进行分类。

在开源硬件项目的开发中，通常将对应的传感器定义成相应的模块来实现。如表3.3.2所示的是项目开发中常用的传感器模块及其简介。

表3.3.2 常用的传感器模块及其简介

传感器	传感器模块图片示例	简介
光线传感器		光线传感器也称环境光传感器，可以用来对环境光线的强度进行检测。左图所示的这款传感器输出的是模拟信号，光线越强，数值越小
温度传感器		温度传感器可以用来对环境温度进行定量的检测，左图所示为一款数字温度传感器
声音传感器		声音传感器用来对周围环境中的声音强度进行检测。左图所示为一款模拟声音传感器，其输出的信号是模拟信号，检测到的声音强度与输出电压成正比
气体传感器		左图所示的气体传感器基于气敏元件的MQ2进行制作，可以灵敏地检测到空气中的烟雾、液化气、丁烷、丙烷、甲烷、酒精和氢气等，输出模拟信号
温湿度传感器		左图所示的温湿度传感器是一款含有已校准数字信号输出功能的温湿度复合传感器
土壤湿度传感器		土壤湿度传感器可用于检测土壤的水分含量
红外测障传感器		红外测障传感器是一种集成发射和接收于一体的反射式光电传感器。该传感器探测距离远，可通过背面的电位器调节测量范围，广泛用于机器人避障、互动媒体、工艺流水线等领域
触摸传感器		触摸传感器是一个基于电容感应的触摸开关模块。人体或金属在传感器金属面上的触碰会被感应到

续表

传感器	传感器模块图片示例	简介
倾斜传感器		倾斜传感器也叫单向倾角传感器。左图所示的传感器是基于钢球开关的数字模块，利用钢球的特性，通过重力作用使钢球向低处滚动，从而使开关闭合或断开
超声波传感器		超声波传感器是利用超声波的特性研制的传感器
心率传感器		心率传感器采用光电容积脉搏波描记法，通过测量血液中血红蛋白随心脏跳动而对氧气吸收的变化量来测量人体心率参数。该模块拥有方波和脉搏波两种信号输出模式，可以通过板载开关自由切换输出信号
按钮传感器		按钮传感器是最简单、最直观的一种传感器，有两种状态，即开或关

除了表3.3.2中的传感器以外，还有其他传感器，如碰撞、触须、霍尔磁力、加速度、指南针传感器等。

在设计开源硬件项目时，要根据测量对象与测量环境的特点，从传感器的类型、量程大小、灵敏度、精度和测量方式等角度选择合适的传感器。

3.3.3 传感器的应用

开源硬件使用的传感器一般具有标准化的接口，可以更方便与开发板或者扩展板进行连接。如图3.3.5所示的土壤湿度传感器模块共有3个接口，其中红色线代表电源正极，通常的表示方法还有“+”“V”“5V”等；黑色线代表电源接地，通常的表示符号还有“-”“G”等；蓝色线代表数据信号输出，通常的表示符号还有“A”“D”“S”，其中“A”通常用来表示模拟信号输出，“D”通常用来表示数字信号输出。在使用传感器前，首先要仔细阅读说明，了解该传感器的输出信号类型，从而确定应该连接到开发板的数字口还是模拟口。



图3.3.5 土壤湿度传感器

●●● 例6 “花盆的心情”

实现效果：利用土壤湿度传感器和开发板，测量土壤的湿度值，根据湿度值，给出适当的提示。

器材清单：① micro:bit；② 土壤湿度传感器（模拟）；③ USB线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，将土壤湿度传感器的2号接口和micro:bit的3V引脚连接，3号接口与micro:bit的GND引脚连接，1号接口与micro:bit的2号信号引脚连接。具体连接方式如图3.3.6所示。

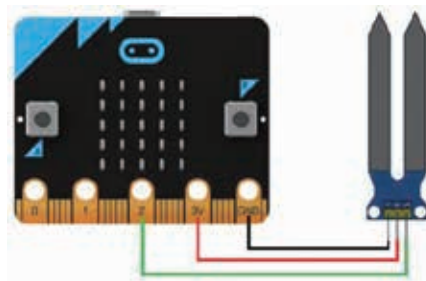


图3.3.6 “花盆的心情”硬件连接图

(2) 程序编写

micro:bit测量读取传感器的模拟值，判断出相应的土壤湿度值，并根据不同的土壤湿度值在显示屏显示不同的图案或在串口中输出相应的值。软件设计部分流程如图3.3.7所示。

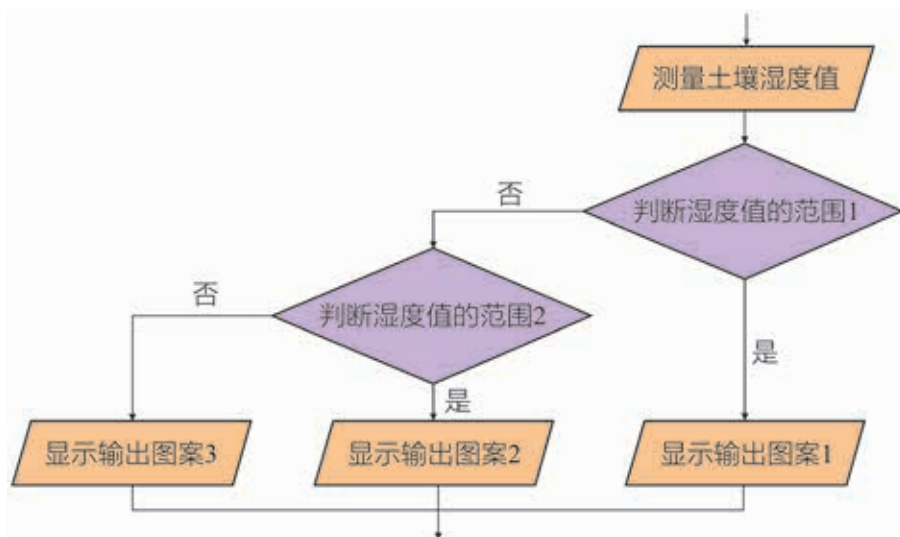


图3.3.7 “花盆的心情”程序设计部分流程图

参考程序如下：

```

from microbit import *           #引用micro:bit库
while True:                       #循环
    a = pin2.read_analog()        #读取2号引脚的模拟值
    if a > 200:                   #在LED阵列显示惊讶的图案
        display.show(Image.SURPRISED)
    elif a > 150:                 #在LED阵列显示生气的图案
        display.show(Image.ANGRY)
    elif a < 50:                 #在LED阵列显示困惑的图案
        display.show(Image.CONFUSED)
    else:                         #在LED阵列显示高兴的图案
        display.show(Image.HAPPY)
    sleep(1000)
    
```

(3) 调试并烧录

将程序烧录至 micro:bit，并将土壤湿度传感器插入花盆进行调试，如图 3.3.8 所示。micro:bit 的 LED 阵列将根据不同的土壤湿度显示不同的内置图案。



图3.3.8 “花盆的心情”调试场景

拓展链接

micro:bit 的内置传感器及对应库模块

1. 加速度传感器

micro:bit 自带三轴加速度计，即加速度传感器，其中 x 代表左右运动，y 代表前后运动，z 表示上下运动。用 `accelerometer.get_x()`、`get_y()`、`get_z()` 的方法来测量设备在 x 轴、y 轴、z 轴上的值，返回一个正数或负数。该传感器还附加手势感应功能，当沿着某个方向移动 micro:bit 时，MicroPython 可以识别出来。`accelerometer.current_gesture()` 方法可以实现对当前动作的抓取，这些手势用字符串的方式来表示，分别是 `up`、`down`、`left`、`right`、`face up`、`face down`、`freefall`。

2. 指南针传感器

micro:bit 内置一个指南针传感器，利用该传感器可以获取 micro:bit 的方向角。使用之前应该先校准，否则可能会读取错误数据。`compass.calibrate()` 可以对指南针传感器进行校准。`compass.is_calibrated()` 判断是否校准完毕，若校准，则返回 `True`；否则，返回 `False`。`compass.get_x()`、`compass.get_y()`、`compass.get_z()` 分别读取 x、y、z 轴上的值，根据方向的调整，得到正或负的整数。`compass.heading()` 返回 micro:bit 与正北方向的顺时针夹角（ $0^{\circ} \sim 360^{\circ}$ ），如返回值 90 代表 micro:bit 指向正东方向。

3. 温度传感器

micro:bit 内置环境温度传感器，提供了内置环境温度检测函数 `temperature()`。通过该函数，返回当前环境温度的浮点数值，单位为摄氏度。

III 实践与体验 III

制作“噪声测试仪”

噪声会损害身体健康。下面让我们通过实践，制作“噪声测试仪”，并使用它对环境声音进行检测。

实践内容：

利用 micro:bit 和声音传感器，根据不同的噪声分贝强度，在 LED 阵列中显示

不同的图像或字符，按下内置按钮 a，在 LED 点阵中滚动显示当前环境的噪声分贝值。

实践步骤：

1. 通过网络搜索噪声的国家规定标准，确定各级噪声的具体数值。
2. 连接声音传感器和 micro:bit。
3. 使用相应的编程工具，编写下列代码。
 - ①通过声音传感器采集环境噪声分贝强度。
 - ②根据第一步确定的噪声判定的各级数值，判断当前的分贝级别，并在 LED 阵列中显示相应的图像或字符。
 - ③按下按钮 a，在 LED 点阵中滚动显示当前环境的噪声分贝值。
4. 调试程序。
5. 使用市场上常用的噪声测试仪与自制的噪声测试仪，对各个时间、地点的环境噪声进行测试，并填写如下实验报告。

时间与地点	常用的噪声测试仪显示的分贝	自制的噪声测试仪显示的分贝	国家标准（分贝）
白天 教室			
白天 寝室			
晚上 教室			
晚上 寝室			

思考讨论：

1. 在学习小组内分享实验报告，比较各自实验报告的异同。
2. 若存在噪声超标现象，试讨论、分析噪声超标的原因。

思考与练习

查找资料，了解当前常用的智能手机中有哪些传感器，并举例说明这些传感器的应用场景。

3.4 控制信号的输出

开发开源硬件项目，不但需要传感器设备的输入，还需要控制各种信号的输出。如实例“闪烁的小星星”，通过数字信号来控制发光LED元件的开和关。根据开源硬件开发的特点，常见的输出信号有图像、光效、声音以及其他动作等。本节将从显示模块、声音模块、电动执行模块三类不同的输出信号来列举常用的输出模块。

3.4.1 显示模块

根据制造材料的不同，显示模块可分为显像管（CRT）显示器、等离子（PDP）显示器和液晶（LCD、LED）显示器等。如图3.4.1所示的是某品牌的数字记分牌，主要通过LED数码管来显示具体分数。LED数码管是开源硬件项目开发的主要显示模块，常见的LED数码管元件如图3.4.2所示。



图3.4.1 数码管显示效果图



图3.4.2 常见的数码管

如图3.4.3所示的是一款七段（也称八段）数码管。它由8个LED发光二极管组成，其中7个LED组成8字形，另一个构成小数点，通过控制不同LED的亮、灭来显示不同的字形。数码管根据驱动方式的不同分成“共阳极”和“共阴极”。“共阴极”是将8个LED的阴极连在一起组成公共端，并使其接地；而“共阳极”是将8个LED的阳极连在一起。数码管的原理如图3.4.4所示。如要在共阴数码管中显示数字“5”，应设置7、4、2、9、10号引脚为高电平，6、1、5号引脚为低电平，则a、c、d、f、g段亮，b、e、dp段不亮。



图3.4.3 七段数码管

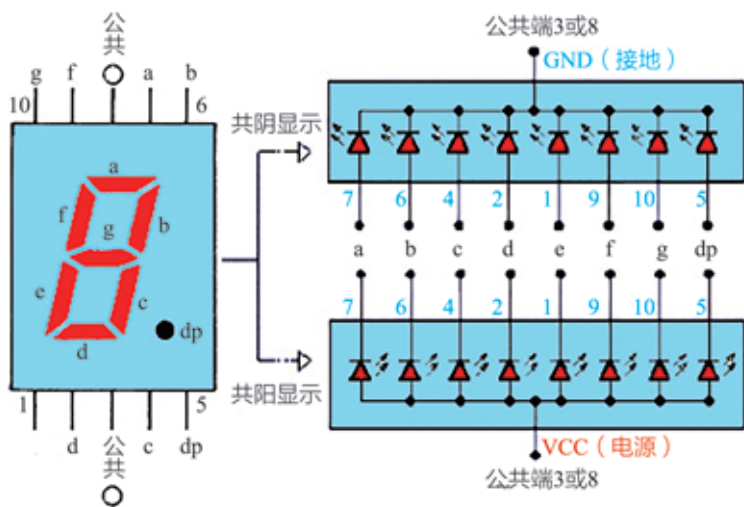


图3.4.4 七段数码管原理图

在项目开发时，将数码管的各个引脚与开发板的引脚进行连接，再通过编写程序，即可实现一个简单的定时器、计数器或时钟模块等功能。通常一个七段数码管称为一位，由几个数码管构成，就称为几位数码管，如图3.4.5所示的是一款8位数码管。



图3.4.5 8位数码管

由七段数码管原理图可知，一位数码管显示需要占用8个数字输入输出（I/O）端口，多位数码管需要的端口则更多。在许多项目中的，更多采用支持SPI总线或者I2C总线控制的显示模块。如图3.4.6所示的是WS2812B全彩LED灯带通过一条数据线实现对多个全彩LED进行颜色控制的原理图，其中“PIX”表示一个LED灯和控制电路组成的像素点， d_n 代表传送的第 n 组LED灯的颜色数据，每一组占24位（RGB三个颜色分量各占8位）。开发板通过输入输出（I/O）口，给出 n 组颜色数据，每经过一个像素点时，都将提取送过来的第一个24位数据，并将剩余的数据转发给下一个像素点。因此每经过一个像素点的传输，数据减少24位。

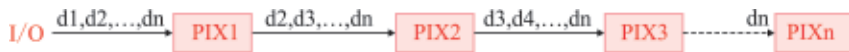



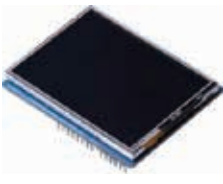


图3.4.6 WS2812B单数据线控制原理图

开源硬件项目开发中常用的一些显示模块如表3.4.1所示。

表3.4.1 常用的显示模块及其简介

显示模块	显示模块图片示例	简介
LED灯模块		LED有白、蓝、黄、红等多种颜色，但一般只能显示一种颜色。全彩LED灯模块可以通过控制内置的三种颜色灯珠亮度来调制出多种颜色。在选择LED灯模块时，需要注意该模块是高电平点亮还是低电平点亮。
8×8 LED点阵显示矩阵模块		8×8 LED点阵显示矩阵模块是利用封装8×8的模块组合点元板形成模块，常用于户外门头单红屏、户外全彩屏，室内全彩屏等。LED点阵显示模块可显示汉字、图形、动画及英文字符等
16×16 RGB全彩LED柔性点阵屏		16×16 RGB全彩LED柔性点阵屏支持单总线控制，仅需一个引脚即可控制所有LED
LCD液晶屏		LCD液晶屏是在两片平行的玻璃当中放置液态的晶体，两片玻璃中间有许多垂直和水平的细小电线，通过通电与否来控制杆状水晶分子改变方向，将光线折射出来以产生画面
OLED显示屏		OLED显示屏是利用有机电致发光二极管制成的显示屏。由于同时具备自发光有机电致发光二极管，以及不需背光源、对比度高、厚度薄、视角广、反应速度快、可用于挠曲性面板、使用温度范围广、构造及制程较简单等优异特性，被认为是下一代的平面显示器新兴应用技术
TFT屏幕		TFT屏幕属于有源矩阵液晶显示器，在技术上采用“主动式矩阵”的方式来驱动，方法是利用薄膜技术所制成的电晶体电极，利用扫描的方法“主动”地控制任意一个显示点的开与关，光源照射时先通过下偏光板向上透出，借助液晶分子传导光线，通过遮光和透光来达到显示的目的
WS2812B全彩LED灯带		WS2812B全彩LED灯带是一个集控制电路与发光电路于一体的智能外控LED光源。灯带中一个IC控制一颗LED组成一个回路(一组)，控制器通过控制IC对，从而控制LED灯带变化出不同的效果

在项目设计中，利用显示模块可将温度、数量、重量、日期、时间等显示出来，让项目的人机交互方式更直观、更多元。

●●● 例7 音乐的溢彩

实现效果：利用彩色LED灯带、模拟声音传感器与开发板，随着音乐的律动，灯带展现炫彩灯光。

器材清单：①micro:bit；②WS2812B彩色LED灯带（含8个像素点）；③模拟声音传感器；④micro:bit扩展板；⑤USB线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，通过micro:bit扩展板将模拟声音传感器与micro:bit的0号引脚连接，再将灯带通过扩展板与micro:bit的1号引脚连接（灯带对电压和电流有要求，因此需要在扩展板上外接电源）。具体连接方式如图3.4.7所示。

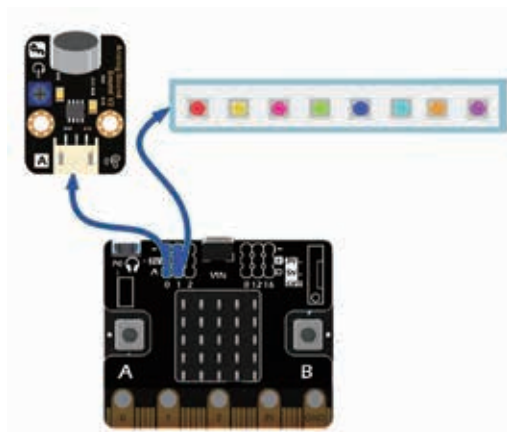


图3.4.7 “音乐的溢彩”硬件连接图

(2) 程序编写

在程序编写前，参照实例“亮度的大小”，利用模拟声音传感器读取音乐播放时声音的大小，并记录希望灯带点亮的声音最小值 s_{min} 和声音最大值 s_{max} 。再结合灯带的像素点个数，计算出亮灯的等级差 s_{step} ， $s_{step} = (s_{max} - s_{min}) \div \text{像素点个数}$ 。

音乐响起，先读取当前模拟声音传感器的值给 a 。当 a 的值大于声音最小值，根据等级差，计算要显示的像素点个数 b ，再随机产生前 b 个像素点的RGB值并显示。音乐分贝越高，则灯带上显示的像素点数越多。当音乐分贝小于声音最小值，则关闭彩色灯带显示。部分程流程如图3.4.8所示。

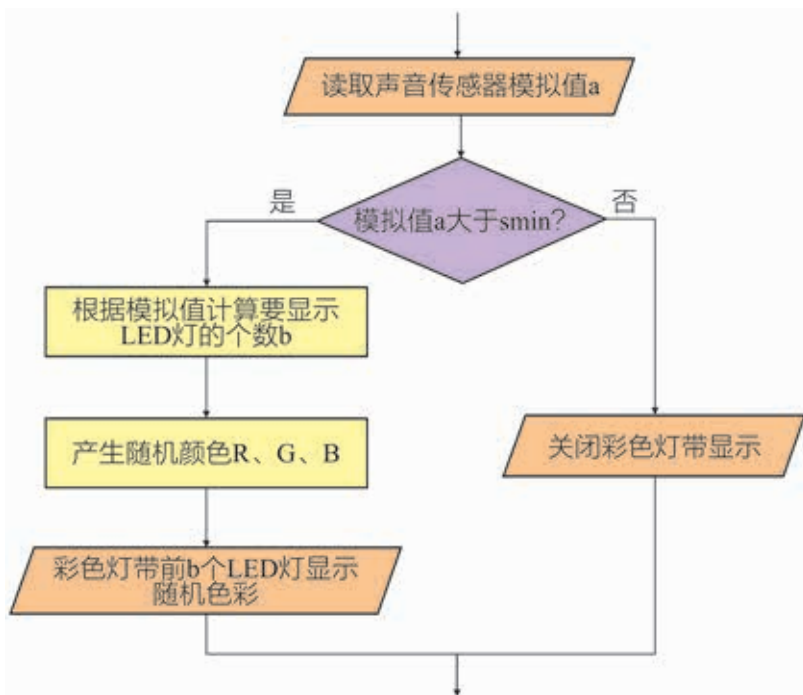


图3.4.8 “音乐的溢彩”部分程序设计流程图

参考程序如下：

```

from microbit import *
import neopixel          #引用多色可编程LED模块 ( neopixel )
from random import randint #引入随机函数

np = neopixel.NeoPixel(pin1,8)
#创建一个使用引脚pin1控制的neopixel对象, LED彩灯数为8个
smax=250                #设置声音最大值smax
smin=100                #设置声音最小值smin
sstep=int((smax-smin)/8) #计算等级差
while True:
    a = pin0.read_analog() #读取0号引脚的模拟值给a
    np.clear()            #关闭Neopixel条显示
    b = int((a-smin)/sstep)+1
    #计算要显示的像素点个数
    if b>8:
        b = 8
    for id in range(0,b):
        rc = randint(0,255) #随机产生颜色
        gc = randint(0,255)
        bc = randint(0,255)
        np[id] = (rc,gc,bc) #给索引设置像素
    np.show() #显示Neopixel条
    sleep(100)

```

(3) 调试并烧录

将程序烧录至 micro:bit，打开音乐进行调试。彩色灯带将根据音乐的律动，显示不同数量、不同颜色的LED灯。

设计开源硬件项目时，不同的显示模块与开发板有不同的连接方法，具体参考相应显示模块的说明。另外，不同的显示模块的输出代码也不相同，需要参考相应的库函数说明及示例代码。

问题与讨论

为了更好地突显彩色灯带的炫彩效果，如果在for循环中也加上sleep(100)，能否实现彩色灯带中的每一个LED依次显示？这种显示方法是否适用于以上实例，根据实例要求展开讨论。

拓展链接

neopixel库函数

neopixel库用于控制全彩LED条带（ws2812芯片）的显示。

1. import neopixel

使用neopixel模块时，需要先单独导入neopixel库。

2. NeoPixel(pinN,n)

创建一个列表，表示从引脚pinN控制的'n'个串联的neopixel灯珠。

3. clear()

清除所有的像素。

4. show()

显示像素。

5. np[0] = (255, 0, 0)

将np列表的neopixel对象的第一个灯珠设置为全亮度红色。以0~255之间的RGB（红色、绿色、蓝色）值作为元组。

拓展链接

micro:bit板载5×5 LED阵列

micro:bit主板中间搭配了5×5可编程LED阵列，该阵列可以用来显示数字、英文、图形及动画等，显示方式有静态、横向滚动等。但其显示信息有限，且不支持汉字显示。该阵列显示函数如表3.4.2所示。

表3.4.2 micro:bit内置LED阵列相应函数

函数	描述
display.scroll(string)	在内置LED阵列中滚动显示string。可以通过delay参数控制字符串滚动的速度
display.show(x)	显示字符或内置图像'x'，如果'x'是图像列表，它们将以动画方式显示
display.set_pixel(x,y,b)	将LED阵列的(x,y)处显示设置亮度'b'，参数'b'可以在0（关闭）至9（全亮）之间设置
display.on()	打开LED阵列
display.off()	关闭LED阵列
display.is_on()	判断LED阵列的显示是否打开
display.clear()	清空LED阵列显示，也就是关闭全部LED灯

III 实践与体验 III

利用彩色灯带制作跑马灯

跑马灯在古代属于灯笼的一种，灯的各个面上绘有将士骑马的画面，由火焰推动空气，使灯笼转动，从而达到将士你追我赶的效果，故名跑马灯。现在编写程序，控制一组彩灯忽熄忽亮，实现各种炫彩变化效果。

实践内容：

利用WS2812B全彩LED灯带和micro:bit，编写全彩LED灯带中的灯珠忽熄忽亮的程序，实现各种炫彩变化效果。

实践步骤：

1. 连接全彩LED灯带和micro:bit。
2. 分组讨论要实现的炫彩变化效果，并完成下表。

序号	效果名称	效果具体描述
1		
2		
3		
4		
5		

3. 绘制各种效果对应的流程图，并利用MicroPython开发环境编写程序。
4. 调试并烧录各种效果。

思考讨论：

如何结合跑马灯效果，实现更炫彩的音乐灯光秀？

3.4.2 声音模块

声音模块可以通过声音进行人机交互，其中蜂鸣器是一种常见的声音模块，广泛应用于计算机、打印机、复印机、报警器、电子玩具、汽车电子设备、电话机、定时器等产品。按其驱动方式的原理，蜂鸣器可分为有源蜂鸣器与无源蜂鸣器。这里的“源”不是指电源，而是指振荡源。

如图3.4.9所示的是一个无源蜂鸣器元件。无源蜂鸣器内



图3.4.9 无源蜂鸣器元件

部不带振荡源，因此价格便宜。通过周期变化的信号驱动，无源蜂鸣器可以发出不同频率的声音。




而有源蜂鸣器控制更简单、更方便，因为内部有振荡源，所以只需通电，蜂鸣器就会通过内部的振荡电路，将恒定的直流电转化成一定频率的脉冲信号，带动钼片振动发音。如图3.4.10所示的是一个有源蜂鸣器模块。



图3.4.10 有源蜂鸣器模块

除了蜂鸣器以外，常见的声音模块如表3.4.3所示。

表3.4.3 常见的声音模块及其简介

声音模块	声音模块图片示例	简介
扬声器		扬声器又称“喇叭”，是一种把电信号转变为声信号的器件
语音播放模块		语音播放模块在生活中应用广泛，如洗衣机、电冰箱、电子玩具、倒车雷达、超市迎宾器等
语音合成模块		语音播放模块一般采用播放存储卡中的声音文件实现语音播放；而语音合成模块能读出输入的中英文字符和数字，无须预先录音就能发出真人的声音

拓展链接

语音识别

语音识别技术，也称为自动语音识别 (Automatic Speech Recognition, 简称ASR), 其目标是将人类语音中的词汇内容转换成计算机可读的输入内容，如按键、二进制编码或字符序列。与说话人识别及说话人确认不同，语音识别技术尝试识别或确认发出语音的说话人，而非其中所包含的词汇内容。

●●● 例8 音乐的节拍

实现效果：开发板输出不同频率的方波，使无源蜂鸣器发出两种不同节拍的聲音。

器材清单：① micro:bit；② 无源蜂鸣器模块；③ micro:bit 扩展板；④ USB 线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，将无源蜂鸣器通过扩展板与micro:bit的2号引脚连接，如图3.4.11所示。

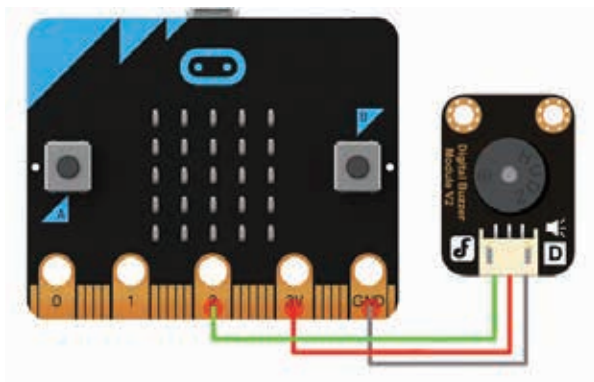


图3.4.11 无源蜂鸣器组件与micro:bit硬件连接图

(2) 程序编写

参考代码如下：

```
from microbit import *
while True:
    for i in range(0,80):                #输出一个频率的声音
        pin2.write_digital(1)           #发声音
        sleep(1)                         #延时1ms
        pin2.write_digital(0)           #不发声音
        sleep(1)
    for i in range(0,100):
        #输出另一个频率的声音，这里的100与前面的80一样，用来控制频率
        pin2.write_digital(1)
        sleep(2)
        pin2.write_digital(0)
        sleep(2)
```

从上面的代码可以看出，for语句中的80、100控制频率，sleep控制时长，类似音乐中的节拍。

(3) 调试并烧录

将程序烧录至micro:bit，蜂鸣器就会发出两个不同频率、不同节拍的音符。

采用其他声音模块与开发板连接时，需要查找相应模块的资料，并参考一些实例，从而总结出具体的连接方式以及相应的代码。

拓展链接

micro:bit 内置音乐模块、语音模块及对应的库函数解析

1. `music.set_tempo(number,bpm)`

设置音乐播放的速度，每分钟播放 bpm 个节拍。每个节拍分为 number 个音调。number 或 bpm 越大，播放越快。

2. `music.play(music,pin=micro:bit.pin0,wait=True,loop=False)`

播放指定音乐，其中 music 表示音符列表，pin 参数用于指定 micro:bit 的输出引脚 micro:bit.pin0。若 wait 设置 False，则在程序继续时音乐将在后台播放。若 loop 设置为 True，则音乐将重复播放。

3. `music.pitch(freq,length=-1,pin=micro:bit.pin0,wait=True)`

使 micro:bit 在 freq 频率下播放 length 毫秒。如果 length 是负数，音调将连续播放。例如，`music.pitch(440,1000)`，表示发出（freq 为 440）的音调一秒钟。

4. `music.stop(pin=micro:bit.pin0)`

停止 pin 针脚上的音乐播放。

5. `speech.translate(words)`

返回一个由英文单词 words 的音素组成的字符串。

6. `speech.pronounce(phonemes)`

读字符串 phonemes 中的每一个音位。

7. `speech.say(words)`

在扬声器中说出字符串 words 中的英文单词。与以下语句等同：

```
speech.pronounce(speech.translate(words))
```

8. `speech.sing(phonemes)`

在扬声器中唱出字符串中包含的音素 phonemes。`speech.say()` 语句能够实现英文字符和数字的英文发音。

III 实践与体验 III

用蜂鸣器演奏歌曲《两只老虎》

从例8可以看出，如果能够控制好频率和节拍，就有可能演奏出动听的音乐。要想演奏出歌曲《两只老虎》，先要了解各音调的频率。

实践内容：

1. 了解各种音调对应的频率。
2. 能够使用 micro:bit 和无源蜂鸣器编写歌曲。

实践步骤：

1. 在网络搜索各种音调对应的频率，并填写下表。

音符	音调						
	1	2	3	4	5	6	7
A							
B							
C							
D							
E							
F							
G							

2. 使用micro:bit和无源蜂鸣器编写程序，播放各种音调。
3. 搜索歌曲《两只老虎》的简谱，并编写程序，使无源蜂鸣器播放音乐。

思考讨论：

1. 休止符就是乐理中记录停顿的符号，如何让micro:bit演奏音乐时实现休止符的效果？
2. 如何改进程序，使得micro:bit支持更长、更复杂的音乐演奏？

3.4.3 电动执行模块

除了显示和声音以外，还有很多方式可以表达信息。例如，手机可以用振动电机进行来电提醒，智能家居系统用电机来打开和关闭自动窗帘。这种将电能转换成机械能的元件叫作电动执行元件。

如果把传感器模块类比成人的感觉器官，那么电动执行元件相当于人的四肢。常用的电动执行模块有继电器、电机、舵机等。

1. 继电器

继电器是一种电控制器件，是一种用小电流去控制大电流的自动开关。继电器的应用范围非常广泛，如通信、机床、家电、汽车等。如图3.4.12所示的是一款继电器模块。



图3.4.12 继电器

拓展链接

继电器原理

如图3.4.13所示，继电器一般由铁芯、线圈、衔铁、触点簧片等组成。只要在线圈两端加上一定的电压，线圈中就会流过一定的电流，从而产生电磁效应，衔铁就会在电磁力吸引的作用下克服返回弹簧的拉力吸向铁芯，从而带动衔铁的动触点与静触点（常开触点）吸合。当线圈断电后，电磁的吸力也随之消失，衔铁就会在弹簧的反作用力下返回到原来的位置，使动触点与原来的静触点（常闭触点）释放。这样吸合、释放达到了在电路中导通、切断的目的。对于继电器的“常开、常闭”触点，可以这样来区分：继电器线圈未通电时处于断开状态的静触点，称为“常开触点”；处于接通状态的静触点称为“常闭触点”。继电器一般有两股电路，分别为低压控制电路和高压工作电路。

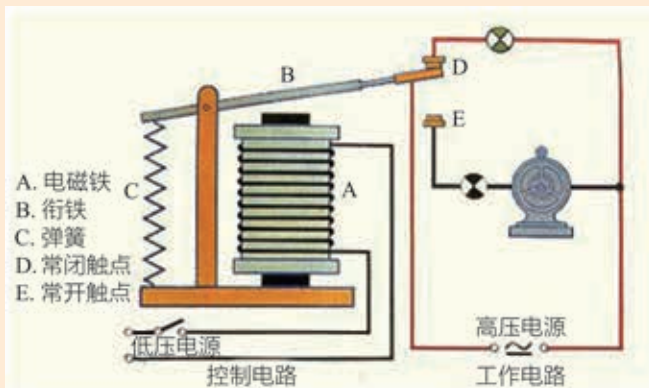


图3.4.13 继电器原理图

2. 电机

电机俗称“马达”，它的作用是将电能转化为机械能，从而产生驱动转矩，可以作为电器或各种机械的动力源。如图3.4.14所示的是一款直流电机风扇模块。

除了直流电机以外，还有许多其他电动执行模块，如无刷电机、步进电机等。在开发开源硬件项目时，要选择合适的电动执行模块，首先要了解它们的特点和用途。如无刷电机因为不需要电刷和换向器，所以可以达到很高的转速，主要用于无人机旋翼、函道风扇中。步进电机可以精确控制转动的角度与速度，主要应用在数控机床制造领域，也可以用在其他的机械上，比如自动送料器、软盘驱动器、打印机和绘图仪等。

电机驱动器又叫电机控制器，是开发板和电机之间的驱动电路，如图3.4.15所示。根据电机的不同，电机驱动器可以分为步进电机驱动器、伺

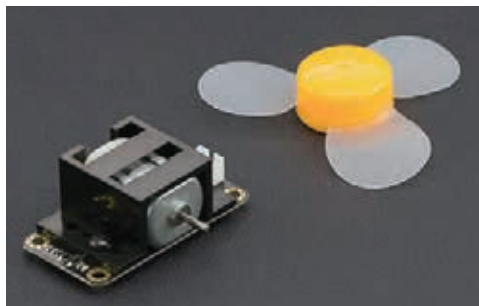


图3.4.14 直流电机风扇模块



图3.4.15 电机控制器

服电机驱动器、无刷直流电机驱动器等。

3. 舵机

舵机常用于控制角度，通过内部的电位器（或其他角度传感器）检测输出轴角度，并结合输入的信号，控制板就能比较精确地控制和保持输出轴的角度，在车模、船模、遥控机器人等项目中广泛应用。如图3.4.16所示的是一款舵机模块。



图3.4.16 舵机模块

电动执行元件在现实生活中应用非常普遍，比如高速公路入口闸机的打开和关闭、发现火情后逃生通道的打开、手环震动提醒等。在一些智能控制中，更是离不开电动执行元件。如智能控制遮阳板会根据当时的天气状况、太阳光照的强度和方向等，自动调整遮阳板的角，既确保室内有充足的光线，又能把强烈的阳光挡在外面。

●●● 例9 智能遮阳板

实现效果：当光线传感器读取的值在一定的范围时，将舵机向某个方向旋转；当读取的值在另一个范围时，将舵机向另一个方向旋转。

器材清单：① micro:bit；② 光线传感器；③ 舵机；④ micro:bit 扩展板；⑤ USB 线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，通过micro:bit扩展板将光线传感器与micro:bit的0号引脚连接，再将舵机与micro:bit的1号引脚连接。具体连接方式如图3.4.17所示。

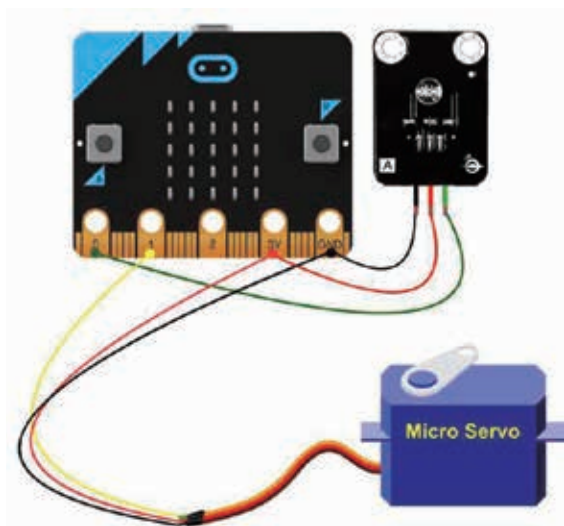


图3.4.17 “智能遮阳板”硬件连接示意图

(2) 程序编写

首先要在开发板读取光线传感器的值，当该值在某个范围时，驱动舵机（代表遮阳板）旋转至某个角度，当该值在另一个范围时，驱动舵机旋转至另一个角度。程序设计部分流程图如图 3.4.18 所示。

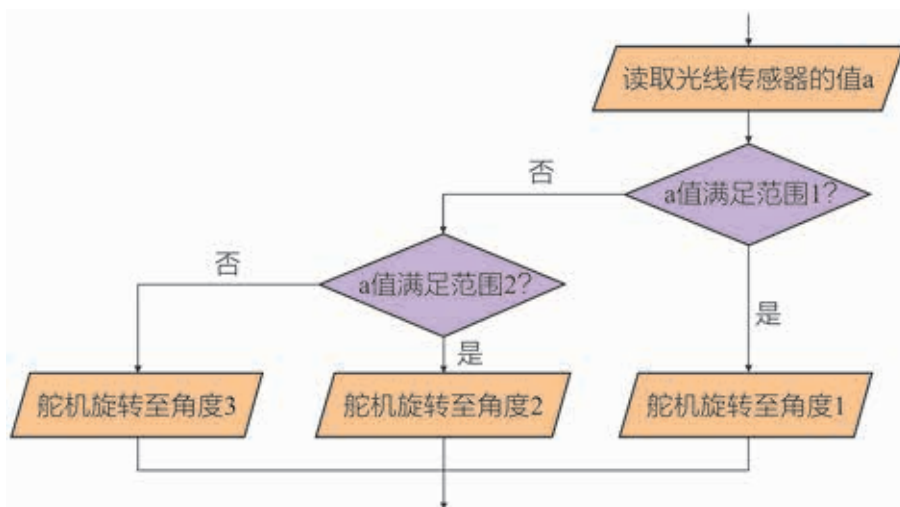


图3.4.18 智能遮阳板部分流程图

参考程序如下：

```

from microbit import *
import Servo                                #引入Servo库
sv= Servo(pin1)                             #建立舵机对象并设置舵机数据接口在pin1
while True:
    a = pin0.read_analog()                  #读取0号引脚的模拟值（光线）给变量a
    if a >= 100:                            #设置舵机转至90度
        sv.angle(90)
    elif a <= 50:                           #设置舵机转至0度
        sv.angle(0)
    else:                                    #设置舵机转至45度
        sv.angle(45)
  
```

（注：Servo库已经内嵌至BXY编程工具中，如果采用其他MicroPython开发工具，需要先引入Servo的类文件或编写的相应Servo类代码。）

(3) 调试并烧录

将程序烧录至micro:bit，测试当光线传感器感应到相应范围的光线后，舵机是否旋转至相应的角度。

拓展链接

扩展模块接线的安全提示

从安全角度考虑，开发板与扩展模块的连接必须在断电的情况下进行。接线时必须遵循相应接线的要求，注意两端标识是否一致。很多扩展模块不同厂商的命名规范并不统一，因此连接前需要先仔细阅读模块的引脚说明，并参考具体实例后再进行连接。特别要注意输入电压是否在扩展模块的额定值以内。连接后还要再次查看接线是否正确，以及各个针脚是否短路，检查清楚后才能进行通电。鉴于接线出错，轻则导致模块无法工作，重则造成模块元器件的损坏，造成不必要的损失。

安装接线时，应注意勿使螺钉、垫圈、接线头等零件落入扩展模块或开发板中，造成短路现象。在身体出汗、衣服潮湿时，也要防止水滴滴入模块或开发板中，造成元器件损坏。日常生活中人体所带的静电也能导致元器件的损坏，因此要求佩戴防静电手腕带或采取其他防静电措施后再进行操作。

在使用动力器件或工具时，应将长头发扎起来，并将宽松的衣服和首饰固定住。防止衣服或头发卷入动力器件造成意外人身伤亡事故。

不同的开源硬件驱动电动执行元件，往往需要不同的控制语句，电动执行元件厂商一般会提供范例或者库文件。在开发开源硬件项目时，要根据选择的电动执行元件或电机驱动器，自行查找相应的资料，并参考对应实例，连接电路并编写代码。

III 实践与体验 III

农业大棚智能控制系统

要实现农业大棚的智能控制，需要通过传感器采集数据，再通过开源硬件中的程序代码进行判断，最后借助输出设备来进行环境控制。下面利用 micro:bit、温度传感器、电机和继电器来模拟制作农业大棚智能控制系统。

实践内容：

通过 micro:bit 的温度传感器采集环境温度，判断温度值是否超过预设的最大值，若超过，智能控制系统会启动电机实现通风，启动继电器 A 实现加湿；如果判断温度低于预设的最小值，那么控制系统会启动另一个继电器 B 实现加热升温；如果判断温度在正常范围内，输出设备暂停工作。

实践步骤：

1. 确定一种大棚植物，并通过网络搜索该植物生长的最适宜温度，并确定启

动控制的最高温度和最低温度，以及适合的温度范围。

2. 将电机、继电器 A 和继电器 B 与 micro:bit 连接，温度传感器可以采用 micro:bit 内置的温度传感器，也可以外接其他温度传感器。

3. 使用 USB 线将 micro:bit 与计算机连接。

4. 使用相应的编程工具，编写下列代码。

①每隔一定时间读取温度传感器的值。

②如果温度值超过最大值，启动电机和继电器 A。

③如果温度值低于最小值，启动继电器 B。

④如果温度值在正常范围内，关闭电机和相应的继电器。

5. 通过人工模拟温度的变化，对制作好的系统进行调试，并填写下面的实验报告。

植物名称	
适宜的温度范围	
最高温度值	
最低温度值	

思考讨论：

在小组内展示实验报告并演示实验装置，了解更多适宜植物生长的温度范围。

3.5 无线通信的实现

在开源硬件项目的开发中，越来越多的设备通信不再局限于通过数据线的有线通信方式。借助无线通信技术，多个设备之间可以进行非接触式的点对点或点对多的数据传输。常见的无线通信技术包括FM、红外、2.4GHz等。特别值得一提的是2.4GHz无线技术，常用的蓝牙和Wi-Fi都是基于2.4GHz技术的无线传输协议。

3.5.1 蓝牙通信

进行串口通信时，需要用传输线进行通信，也就是说串口通信是一种有线通信方式。而在无线通信领域，蓝牙已成为目前的一个新热点，比如手机、耳机、音箱、鼠标等产品都支持蓝牙功能。目前市场上蓝牙模块版本有许多，在开发开源硬件项目时，应该选择合适版本的蓝牙模块。

自20世纪90年代发展到今天，蓝牙标准经历了多个版本的演进，连接速度和传输效率得到大幅度的提高，功耗也越来越低。目前，蓝牙最常用的是2.0和4.0两个版本，如图3.5.1和图3.5.2所示。从4.1版本开始，蓝牙开始支持同时多设备连接。



图3.5.1 蓝牙模块2.0



图3.5.2 蓝牙模块4.0

设计项目时，要选择合适的蓝牙版本。一般来说，高版本的蓝牙能够兼容低版本的蓝牙，但要使数据通信更加稳定，应尽量采用相同版本的蓝牙模块。

●●● 例10 micro:bit与手机交互

实现效果：在手机上输入大写字母，在micro:bit内置LED矩阵显示相应的字母，同时按下micro:bit内置按钮a，手机上输出“Hello”。

器材清单：①micro:bit；②蓝牙2.0模块；③micro:bit扩展板；④安卓手机；⑤USB线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，通过micro:bit扩展板将蓝牙模块的3v3、GND、RxD、TxD接口分别与micro:bit的VCC、GND、0号和1号引脚连接。具体连接方式如图3.5.3所示。

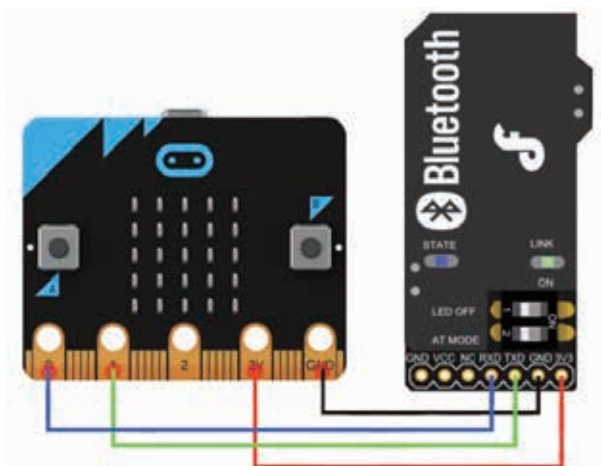


图3.5.3 “micro:bit与手机交互” 硬件连接示意图

(2) 程序编写

先进行蓝牙连接，设置蓝牙无线广播参数，再写入接收和发送广播的判断，其中“uart.any()”用于判断是否接收到数据。参考程序如下：

```

from microbit import *
uart.init(baudrate=9600, bits=8, parity=None, stop=1, tx=pin1, rx=pin0)
#串口的初始化，设置波特率、数据位、奇偶校验、停止位及串口引脚
while True:
    if uart.any():
        incoming = str(uart.readall(), "UTF-8")
        incoming=incoming.strip('\n')
        if ord(incoming)>=65 and ord(incoming)<=90:
            display.show(incoming)
    if button_a.was_pressed():
        uart.write("Hello")
    
```

(3) 调试并烧录

将程序烧录至micro:bit。开启手机蓝牙功能，搜索附近可配对的蓝牙设备。选择相当的蓝牙模块进行配对，配对码默认为“1234”。在手机的应用商城选择合适的“蓝牙串口助手”应用程序并安装运行，再点击“助手”上的“连接”按钮，选择正确的蓝牙设备进行连接，如图3.5.4所示。



图3.5.4 蓝牙串口助手连接蓝牙设备

在手机上输入消息字母“A”，在micro:bit的内置LED矩阵上显示字母“A”；按下micro:bit内置按钮a，手机上输出“Hello”。

利用App Inventor（一款安卓APP的开发工具），就可以编写相应的手机APP程序，实现如用手机APP获取温度传感器值、控制灯的开和关等。

3.5.2 Wi-Fi 模块通信

很多开发板（如Arduino、micro:bit等）都不具备网络访问的功能，但只需使用Wi-Fi模块，就可轻松接入互联网，从而实现无线智能家居、M2M等应用。如图3.5.5所示的是一块Wi-Fi通信模块。

相比蓝牙模块，Wi-Fi模块传输速率更快，但最大的缺点在于功耗较高，不适合可穿戴等需要低功耗传输的设备。不过在家用领域，如无线音箱、视频同步传输等方面，Wi-Fi显然更具优势。针对不同项目，应该根据不同的要求，选择适合的无线通信技术和模块。



图3.5.5 Wi-Fi通信模块

●●● 例11 室外环境温度实时监测

实现效果：利用温度传感器监测室外环境温度，并通过Wi-Fi通信模块将数据传送给服务器，服务器根据传送的数据记录进行分析，并自动生成曲线图。

器材清单：①micro:bit；②温度传感器；③Wi-Fi通信模块；④micro:bit扩展板；⑤USB线。

操作步骤：

(1) 硬件连接

通过USB线将micro:bit连接到计算机，通过micro:bit扩展板将温度传感器与micro:bit的3V、GND和0号引脚连接；再将Wi-Fi通信模块的“T”“R”“+”“-”端口分别与micro:bit的1号、2号、3V和GND引脚连接。具体连接方式如图3.5.6所示。

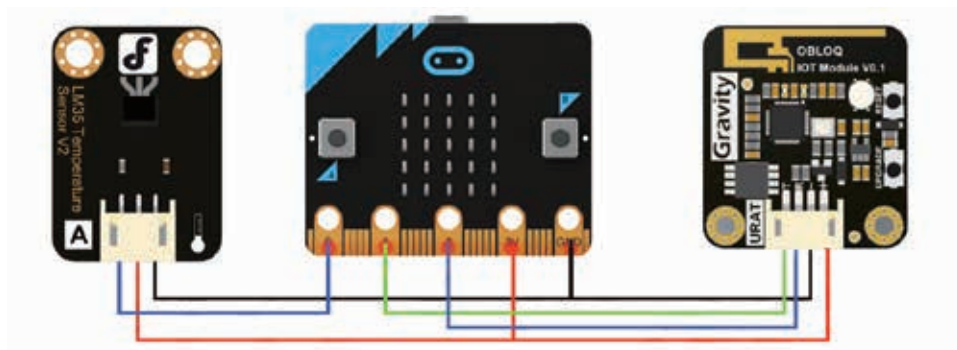


图3.5.6 “室外环境温度实时监测”硬件连接示意图

(2) 程序编写

客户端代码如下所示：

```

from microbit import *
import Obloq                                     #导入Obloq包

IP="192.168.2.13"                                #定义Web服务器IP地址
PORT="8080"                                     #设置Web服务器端口号
SSID="wifiname"                                 #设置Wi-Fi的ID
PASSWORD="12345678"                             #设置Wi-Fi的密码

uart.init(baudrate=9600, bits=8, parity=None, stop=1, tx=pin2, rx=pin1)
#串口的初始化，设置波特率、数据位、奇偶校验、停止位及串口引脚

while Obloq.connectWifi(SSID,PASSWORD,1000)!= True:
#循环判断是否连上Wi-Fi
    display.show(".")                            #在LED矩阵中显示 “.”

display.scroll(Obloq.ifconfig())                 #在LED矩阵中滚动显示micro:bit的IP地址
Obloq.httpConfig(IP,PORT)                       #设置Obloq连接的服务端IP地址

while True:                                     #循环
    temp = (pin0.read_analog()/4096)*3300/10.24 #读取引脚0
    errno,resp=Obloq.get("inputs?id=1&val="+str("%.1f"%temp),10000)
#服务器发送温度信息，并得到返回状态值
    if errno == 200:                             #如果返回状态值为200
        display.scroll(resp)                     #显示状态信息
    else:
        display.scroll(str(errno))               #显示状态码
    sleep(1)
    
```

其中，代码的第1行和第2行：导入micro:bit中的所有模块和Obloq包。

第4行到第7行：定义IP地址、端口号、WiFi的ID和密码。

第9行：串口的初始化，设置波特率、数据位、奇偶校验、停止位及串口引脚。

第11行和第12行：循环连接Wi-Fi，如果没有连接上，会在LED矩阵中显示“.”，直到连接成功执行下一步，如果在10秒内没有连接成功，视为超时。

第14行和第15行：Wi-Fi连接成功后，在LED矩阵中滚动显示micro:bit的IP地址，并设置要连接的服务端的IP地址。

第18行到第24行：读取温度信息，并向服务器发送温度信息，然后得到返回值：状态码和信息。若状态码为200，则显示状态信息，否则显示状态码。

该实例包含客户端和服务端两部分，实现客户端和服务端数据传输，基于HTTP协议进行传送。客户端通过GET传送数据。如“/inputs?id=1&val=32.5”，表示传送温度值为32.5，若发送成功，则在micro:bit内置的LED阵列显示“success”，否则显示“error”。服务器端通过“request.args.get("val")”获取温度值，并用“f.write()”将温度值及相应参数写入文件“data.txt”中，具体的写入格式如图3.5.7所示。

18-07-26,20:23:38	1	23.4
18-07-26,20:23:38	1	23.6
18-07-26,20:23:38	1	23.5

图3.5.7 温度值及相应参数写入格式参考

服务器端部分代码如下所示：

```
from flask import Flask
from flask import request
from datetime import datetime

app = Flask(__name__)

@app.route('/inputs', methods=['GET'])
def inputs():
    id=request.args.get("id")
    val=request.args.get("val")
    now_dt=str(datetime.now().strftime("%y-%m-%d,%H:%M:%S"))
    instr=now_dt+"\t"+str(id)+"\t"+str(val)+"\n"
    try:
        f=open("data.txt","a+")
        f.write(instr)
        return "success"
    except Exception:
        return "error"
    finally:
        f.close()

#main
if(__name__ == '__main__'):
    app.run(host="192.168.2.13",port=8080,threaded=True,debug=True)
```


服务器端可以将存储文件中的所有记录生成曲线图，以便查看温度变化趋势，如图3.5.8所示。

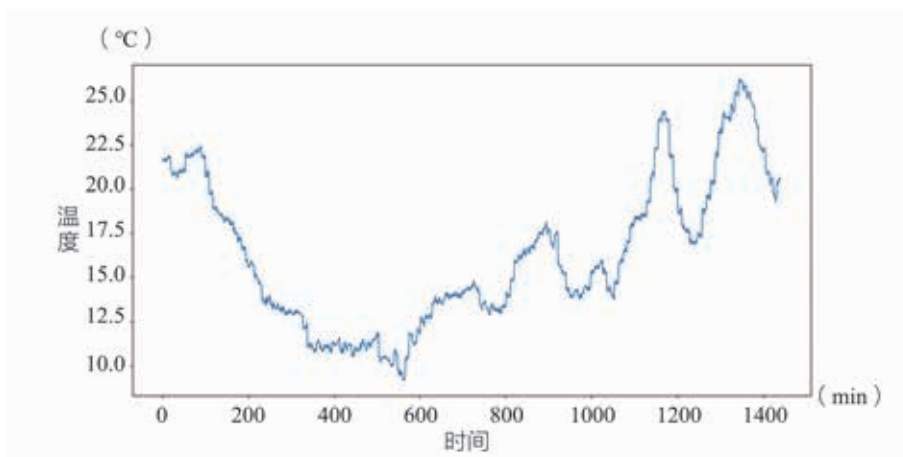


图3.5.8 室外环境温度曲线图

III 实践与体验 III

制作“发报机”

micro:bit内置了2.4GHz芯片，通过简单的代码，就能实现多个micro:bit间的通信，实现简单的发报功能。

实践内容：

两个micro:bit分别用A、B来代表。当按下A款micro:bit上的按钮a时，B款micro:bit上显示相应的图像或字符；当按下A款micro:bit上的按钮b时，B款micro:bit上显示另一个图像或字符。

实践步骤：

1. 导入radio无线模块库，并设置相应的参数。

```
import radio
radio.on()           #打开广播
radio.rest()        #将设置重置为默认值
radio.config(power=7, channel=98) #设置无线广播的参数
```

2. 编写A款micro:bit发送代码。

```
if button_a.is_pressed():
    radio.send("1") #当按钮a被按下
                    #发送广播消息字符串“1”
```

3. 编写B款 micro:bit 接收和输出反馈代码。

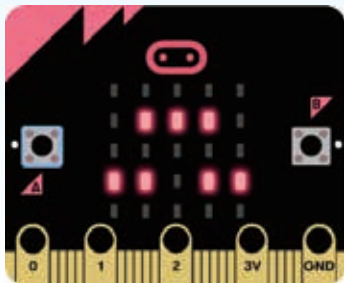
```
received=radio.received()           #接受广播消息
if received=='1':                   #判断如果接收到广播消息“1”
    display.show(Image.ARROW_W)     #显示向左（西）的箭头
```

4. 调试并烧录程序。

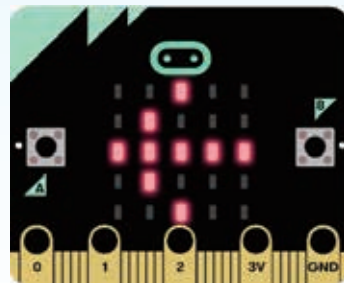
5. 测试 micro:bit 的通信距离，并完成下表。

距离	1 米	2 米			
连接情况					

结果呈现：



A款micro:bit



B款micro:bit

巩固与提高

1. 设计一个现代化智能农业大棚实现下述功能：

- 自动检测泥土湿度情况，并进行自动灌溉。
- 自动检测环境亮度，对大棚遮阳情况进行调整。
- 检测泥土肥沃情况，并对不同情况下的绿植状况进行记录。

根据以上功能要求，列举在现代化智能农业大棚中要用到的传感器模块、输出模块和通信模块，并思考具体设计方案。

模块类型	模块名称	对应功能	设计方案

2. 调查所在城市的彩灯现状，了解彩灯常见的交互功能及其对应的各种变化规律，并尝试写出实现方法。

交互功能	变化规律	设计方案

项目挑战

创客训练营——初步开发项目

充满创意的开源硬件项目原型已经产生，而理想与现实之间还有很长的一段距离。现在到了具体开发项目的时候了。

▶ 项目任务

利用本章学习的内容，初步开发自己的开源硬件项目。

▶ 过程与建议

1. 确定要开发的功能

请利用下表梳理当前开源硬件项目中各个功能的实现可能性：在第一列中写下预想功能；在第二列中用五角星来标示该项功能实现的可能性，五星是完全有可能，四星是比较有可能，三星是较有难度，二星是非常有难度，一星是完全不可能；在第三列中说明要实现这一功能需要使用到的技术或传感器。

创意中涉及的功能	利用现有技术可实现的可能性	使用何种技术/传感器
功能1:	☆☆☆☆☆	
功能2:	☆☆☆☆☆	
功能3:	☆☆☆☆☆	
功能4:	☆☆☆☆☆	

确定下来优先开发的功能是：_____。

2. 开发中的小组分工与时间进度

面对开发多个功能时，小组成员要协同合作，分工时既要考虑每个人的特长，也要考虑时间进度。

请试着填写下面的甘特图，确定小组成员的分工与时间进度。

注：甘特图于1917年由美国工程师和社会学家亨利·甘特（Henry L. Gantt）发明。一个典型的甘特图由横轴与纵轴构成，纵轴称为“任务栏”，表示构成整个项目的任务（或活动）；横轴表示项目的时间，并按照特定的时间刻度（如小时、天、周、月、季度、年等）呈现。

任务	负责人	1	2	3	4	5	6	7

3. 测试与修订

对开源硬件项目已实现的功能进行调试和修订，保证流畅运行。

▶ 评价标准

请根据项目实施的过程、效果以及成果展示交流的结果，对自己完成项目的情况进行客观的评价，并思考后续完善的方向。把评价结果和完善方案填写在下面的表格中。

评价条目	说明	评分（1~10分）	评分主要依据阐述	后续完善方向
小组合作	小组分工合理、协作有效			
功能开发	已开发的功能技术含量高			
运行情况	已开发的功能运行流畅			

▶ 拓展项目

1. 风能是一种清洁的可再生能源，利用风力发电非常环保，而且能够产生的电能非常巨大。我们校园的环境是否适合风力发电？所发电量能否起到节省成本的作用？选择合适的开源硬件和传感器，设计跟踪监测校园环境的系统。根据监测的结果，分析校园内是否适合风力发电。

2. 利用开源硬件和各种传感器、显示模块、电动执行元件，设计一个人灯互动的彩灯世界。

3. 设计一个智能的盆栽花盆，让用户能够随时掌握盆栽的生长状态。列举出需要使用的传感器模块和输出模块，并分析需要的具体参数与具体的选择情况。

开源硬件项目的开发实践



在世界各地的Maker Faire和创客嘉年华活动中，常常可以看到创客们用开源硬件开发人机互动、多机互动和物联网等方面的项目。在掌握开源硬件的基础知识后，我们已经具备一定的“造物”能力。不断实践，可以使我们的“造物”能力得到快速提高。

参考别人的创意项目进行模仿并改造是学习开源硬件项目开发的常用方法。每一次模仿和改造，都是有意义的学习实践。在实践中，新的创意会不断涌现。

问题与挑战

- 自从有了计算机，人们就开始不断探索如何与它进行更有效的沟通，并努力使沟通体验更愉悦，于是有了“人机交互”这一学科。虽然鼠标和键盘一直是人与计算机沟通的主要输入设备，屏幕是计算机信息最主要的输出设备，但这种方式始终不尽如人意。为了减少这种冷漠感，工程师们不断探索更接近人类行为的新人机交互方式。那么，如何利用开源硬件设计这类自然、直观的人机交互方式？

- 做科学实验离不开实验数据的采集。无线数据采集（Wireless Data Acquisition）是指利用无线数据采集模块或设备，将传感器输出的电压、电流等物理量通过无线形式进行远程传输。而在具体应用中，分布式数据采集能够有效降低成本和难度。近几年，支持分布式、远程通信的开源硬件不断涌现。如何利用开源硬件，实现同时与多个传感器终端连接？

- 智能硬件的普及让物联网的发展不断加速。通俗地讲，物联网就是“物物相连的因特网”，其目标是让万物沟通对话。试想一下这样的“智慧生活”：早晨，客厅及卧房的窗帘因户外的亮度而自动打开；当开启冰箱准备早餐时，冰箱马上显示哪些食物即将过期；搭乘地铁时，智能手表显示要搭乘列车的到达时间；回到家，空调已经自动开启并调好室内温度。如何基于开源硬件，设计一些与物联网、智能家居相关的应用项目？

学习目标

1. 根据设计方案，选择恰当的开源硬件，搜索相关的使用说明资料，审查与优化项目设计方案。
2. 了解项目开发过程中的各种设备与组件的安全使用规则和方法，根据设计方案，选择合适的开源硬件、相关组件与材料。
3. 根据设计方案，利用开源硬件的设计工具或编程语言，编制项目的各种功能模块。
4. 根据设计方案，测试、运行项目的数据采集、运算处理、数据输出、调控执行等各项功能，优化设计方案。

★ 内容总览



4.1

人机互动的项目开发

鼠标和键盘一直是人们与计算机沟通的主要输入设备，在程序设计中经常会使用“鼠标经过”“鼠标单击”“键盘按下”等事件来判断用户的行为与指令。不借助鼠标键盘，而通过其他输入设备实现与计算机互动的项​​目，一般称为“人机互动”项目。Makey Makey可以让任何能导电的物体变成鼠标和键盘。很多人基于Makey Makey设计了有趣的交互作品，如用橡皮泥做的游戏柄，用草莓、香蕉等水果做的水果钢琴（图4.1.1所示）等。

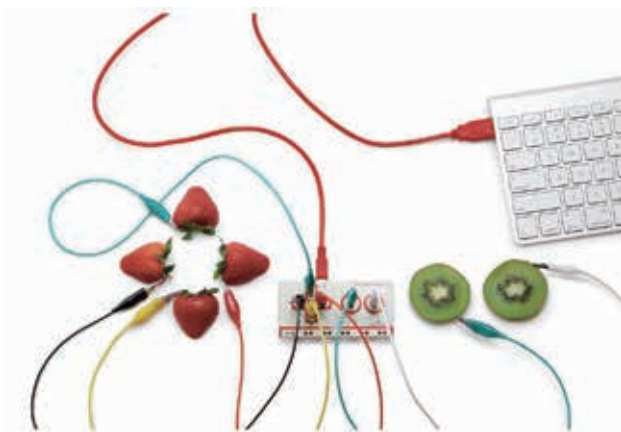


图4.1.1 Makey Makey水果手柄

4.1.1 人机互动的项目特点

一个典型的人机互动项目一般分为动作采集、智能处理、反馈输出三个关键环节，如图4.1.2所示。人机互动过程实际上是一个典型的输入和输出过程，用户通过各种技术手段向计算机输入动作指令（如触摸香蕉），经过计算机的特定程序智能处理后，把输出结果（如发出声音）呈现给用户。

人机互动项目的设计者为了摆脱鼠标和键盘的限制，会不断尝试各种新技术。这些技术最初的应用场景往往出现在科幻小说或科幻电影中，如在科幻电影《少数派报告》中，男主角用手势控制计算机屏幕，让观众震撼不已，如图4.1.3所示。除了手势控制以外，姿态控制、眼球控制、脑波控制等技术都是人机互动



图4.1.2 人机互动项目的运行流程



图4.1.3 《少数派报告》男主角用手势控制计算机屏幕

项目的研究热点。

人机互动项目强调用户和计算机之间能够做到自然交互。自然交互是指用手势或身体状态发出操作指令，像操作真实世界的事物一样操作计算机，如手掌平移就能移动计算机屏幕里的窗口。这种手势或动作的采集，一般采用图像识别和运动识别两种方式。图像识别是使用摄像头、体感传感器之类的设备，如用Kinect和Leap Motion来识别操作者的手势或身体状态。运动识别是用加速度、陀螺仪等传感器，根据传感器的数值变化，判断操作者的手势。相对而言，第一种方式需要专用的设备和使用场景，限制较多；第二种方式虽然识别的准确度会低一些，但成本也较低，应用灵活。

人机互动项目的设计，除了能够满足人们对新奇的计算机控制方式的追求以外，还有很多更为实际的积极意义，如康复训练、VR控制等。一些肢体先天发育不良或者后天受过伤害的残疾人士，不方便使用传统的输入设备，需要设计特定的人机互动设备来帮助他们控制计算机。如中风后的身体康复训练很枯燥，一些患者往往难以坚持，而康复训练游戏通过用手握操作杆玩游戏的方式锻炼患者的神经和肌肉功能，增加患者康复锻炼的兴趣。

问题与讨论

除了鼠标和键盘以外，你还见过哪些可以控制或操作计算机的设备？请分析这些设备的使用场景。

4.1.2 人机互动项目的开发实例

用户在演讲的时候，有时为控制PPT翻页而跑到计算机前操作键盘，这样容易打断演讲者的思路，而小巧的PPT翻页笔有效解决了这个问题。PPT翻页笔是人机互动项目中最常见的一种产品，如图4.1.4所示。

PPT翻页笔一般只能控制PPT的播放，功能十分有限。如果这支翻页笔的按键可以自定义，那么不仅可以控制PPT播放，还可以控制视频的播放，甚至可以玩一些简单的桌面游戏。这就是电脑魔法控制器项目的开发背景，即做一款能够灵活控制计算机里各种应用软件的手持设备。



图4.1.4 PPT翻页笔

1. 电脑魔法控制器的功能特点

电脑魔法控制器的开发需要在使用方式、产品外观、基本功能和扩展功能等方面进行考虑。项目分析如表4.1.1所示，其中遥控和模拟按键是核心功能。

表4.1.1 电脑魔法控制器的项目分析

需求	描述	理由
键盘模拟	能模拟常用软件的快捷键	能够适用于更多的应用场景，如控制PPT的播放、控制桌面弹球之类的小游戏
自然交互	能用手势或身体状态发出操作指令	能够体现出一种比较自然的交互
远程控制	摆脱数据线的束缚，实现无线控制	电脑魔法控制器本来就是为了解放用户，让他们不受鼠标键盘的限制，那自然不能再被“线”所束缚。如演讲的时候，用户一般会站在投影或者大屏幕前
小巧便携	一只手就能携带、操作	控制器如果体积过大，既不易携带又容易转移观众的注意力

用思维导图来描述电脑魔法控制器的项目功能，如图4.1.5所示。

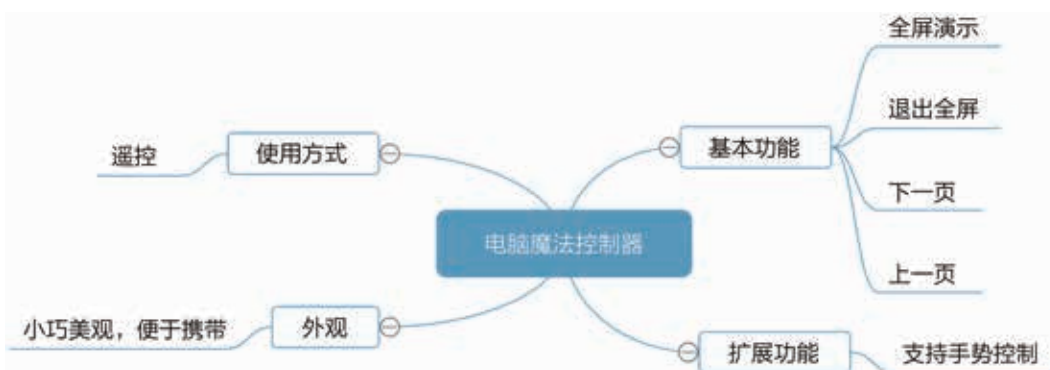


图4.1.5 电脑魔法控制器项目功能

2. 电脑魔法控制器的硬件组成

硬件方面，选用了2块micro:bit分别作为这个项目中的控制器和接收器。micro:bit自带无线通信功能，可以实现简单的信息互动；内置加速度传感器，可以识别简单的姿态动作；板上a、b两个按钮，刚好可以作为PPT“上一页”和“下一页”的按钮。

micro:bit板子的体积小，加上一个小型的锂电池作为电源，因此，控制器外壳可以做得很小，方便携带。如果借助3D打印或者激光切割技术，为接收端和控制端做一个精致的外壳，佩戴在手上效果会更好，如图4.1.6所示。

电脑魔法控制器分为控制器和接收器两个部分。其中控制器负责将用户的控制指令发送给接收器，接收器负责将控制指令翻译为控制计算机操作的按键信息。系统框架如图4.1.7所示。



图4.1.6 戴在手上的micro:bit



图4.1.7 电脑魔法控制器的系统框架

控制计算机需要模拟鼠标和键盘的功能。有些设备能够将自己虚拟成一个HID设备（Human Interface Device，人机接口设备），按照鼠标和键盘的规范发送相应的脉冲信号，计算机会将这个设备看成鼠标或键盘，如Makey Makey就是采用这个原理实现的。micro:bit不支持HID功能，但可以采用软件的方式实现模拟按键。

拓展链接

HID 设备

HID 自带USB设备，能够将自己虚拟为HID设备。开源硬件Arduino中有一个叫作Leonardo的型号，采用了ATMega32 U4芯片，下载特定的程序后，计算机就会把它识别成一个鼠标或键盘，甚至是游戏柄。Makey Makey上使用的也是ATMega32 U4芯片。

3. 软件设计

电脑魔法控制器涉及接收器、控制器和应用程序三个模块，三者之间的通信需要确定一个通信“协议”。这个控制器需要发出的指令并不复杂，以PPT控制为例，基本操作仅“上一页”和“下一页”两个指令。

表4.1.2 接收器、控制器和应用程序的通信协议描述

编号	指令	控制器	接收器	应用程序（PPT版）
1	上一页	按下a，发送字符A	收到字符A，串口发送A	收到字符A，模拟按键“左移键”
2	下一页	按下b，发送字符B	收到字符B，串口发送B	收到字符B，模拟按键“右移键”

（1）控制器代码

控制器负责将用户的按键信息转化为字符，通过无线的方式发送给接收器。

```
if button_a.is_pressed():
    radio.send("A")
```

（2）接收器代码

接收器的作用是将无线发过来的信息，通过串口转发给计算机。考虑到信息干扰，代码中只针对处理A、B这两种字符串，其他的全部过滤。

```
incoming=radio.receive()
if not(incoming==None) and (incoming=='A' or incoming=='B'):
    print(incoming)
```

(3) 串口应用程序代码

串口应用程序的作用是读取串口信息，然后将读取的串口信息转化为模拟的按键信息。键盘上的按键，都有一个对应的代码，代码和ASCII码是一致的，如图4.1.8所示。

Esc	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	Print	Scr	Pause	Number Lock	Caps Lock	
27	112	113	114	115	116	117	118	119	120	121	122	123	44	145	19			
192	49	50	51	52	53	54	55	56	57	48	189	187	220	8	45	36	33	
9	TAB	Q	W	E	R	T	Y	U	I	O	P	[]	Enter	46	35	34	
20	Caps L	A	S	D	F	G	H	J	K	L	;	'	13	144	111	106	109	
16	Shift	Z	X	C	V	B	N	M	,	.	/	16	38	103	104	105	107	
17	Ctrl	Win	Alt	Space					Alt	Win	RightK	Ctrl	37	40	39	96	110	13

图4.1.8 键盘键值表

计算机软件一般都有快捷键，先研究要控制的软件，找到相应的功能按键，再从“键盘键值表”中找到对应的键值。以PPT控制为例，PowerPoint和WPS都是用左移键和右移键来控制PPT播放的。从键盘键值表可以得知，左移键和右移键的键位码分别是37和39。F5是116，ESC键是27。整理这些信息，如表4.1.3所示。

表4.1.3 控制PPT播放的按键分析

功能	按键	键值
上一页	左移键	37
下一页	右移键	39
进入演示模式	F5	116
退出演示模式	ESC	27

以控制桌面游戏“太空弹球”为例，常见的操作其实只有“移动左挡板”和“移动右挡板”两种，分别用“Z”和“/”键控制，如表4.1.4所示。

表4.1.4 控制“太空弹球”的按键分析

功能	按键	键值
移动左挡板	z	90
移动右挡板	/	191

在Python中实现模拟按键功能，要导入win32api和win32con两个库，参考代码如下：

```
win32api.keybd_event(37,0,0,0) #按下左移键，键位码是37
win32api.keybd_event(37,0,win32con.KEYEVENTF_KEYUP,0) #释放左移键
```

电脑魔法控制器项目的运行界面如图4.1.9所示，这个程序支持PPT播放和太空弹球游戏两种应用。



图4.1.9 程序运行状态

问题与讨论

1. 电脑魔法控制器要实现对应用程序的控制，需要研究这些应用程序的快捷键。不同的应用程序的快捷键是不同的，串口应用程序应该预设这些常见的按键信息。如何让这个串口应用程序同时支持多个应用程序？如视频播放和太空弹球游戏。
2. 使用两个按键控制PPT的上翻和下翻仅仅是基础功能，能否结合micro:bit内置的加速度传感器、电子罗盘等实现用手势来控制PPT的播放？

实践与体验

用micro:bit按键控制PPT的播放

电脑魔法控制器是一款能够远程控制计算机进行常见应用软件操作（如PPT播放、游戏控制等）的人机互动项目。micro:bit自带了无线通信模块，micro:bit之间的广播通信很容易实现。这个项目的难度在于如何将串口的信息转化为模拟按键的命令。

实践内容：

1. 编写代码实现接收器将控制器的按键信息通过串口发送给计算机。
2. 用控制器的a、b按键远程控制计算机PPT的播放。

实践步骤：

1. 准备器材：2块micro:bit、2根USB线和1个充电宝。

2. 给控制器编写代码。打开BXY，输入如下代码。

```
from microbit import *
while True:
    if button_a.was_pressed():
        print("A")
        display.scroll("A")
    if button_b.was_pressed():
        print("B")
        display.scroll("B")
```

烧录程序后，打开“串口调试”，按下 micro:bit 的 a 键或 b 键，窗口中将显示信息，如图 4.1.10 所示。



图4.1.10 程序运行状态

3. 串口应用程序接收信息并控制 PPT 播放。新建 Python 代码，内容如下：

```
import win32api, win32con, serial
ser = serial.Serial()
#设置通信波特率
ser.baudrate = 115200
#设置串口号
ser.port = 'COM3'
ser.open()
while(1):
    incoming=ser.readline()
    incoming=incoming.strip()
    incoming=incoming.decode("gb2312")
    if incoming=='A':
        win32api.keybd_event(37,0,0,0) #左移键的键位码是37
```

```
win32api.keybd_event(37,0,win32con.KEYEVENTF_KEYUP,0) #释放按键
if incoming=='B':
    win32api.keybd_event(39,0,0,0) #右移键的键位码是39
    win32api.keybd_event(39,0,win32con.KEYEVENTF_KEYUP,0) #释放按键
```

注意：ser.port = 'COM3'处，要根据当前计算机的micro:bit设备串口号进行修改。在不同的计算机中，micro:bit设备的串口号是不一样的。

运行这段代码，然后打开任意一个PPT文件，设置为“幻灯片演示”状态。然后按下micro:bit的a、b键，测试PPT是否前后翻页。

思考讨论：

在一个区域内，若有多人使用该PPT遥控器，就会造成干扰。有哪些办法可以解决干扰问题？

4.2 多机通信的项目开发

多机通信是指通过有线或无线的方式，多个子系统之间进行数据通信，让信息系统能够实现比较复杂的功能。如图4.2.1所示的汽车胎压实时监测系统，需要通过四个胎压传感器来获取每个轮胎的实时胎压，并将胎压数据实时发送给行车计算机，在仪表盘上显示出来。又如农场里的温室环境监测管理系统，也需要连接很多的环境采集节点，来管理各温室大棚的环境数据。



图4.2.1 胎压实时监测系统

4.2.1 多机通信的项目特点

多机通信的应用非常广泛，简单的多机通信应用有多节点数据采集、远程集群控制等，如图4.2.2所示。而复杂的多机通信应用则包括了分布式存储、云计算等领域。

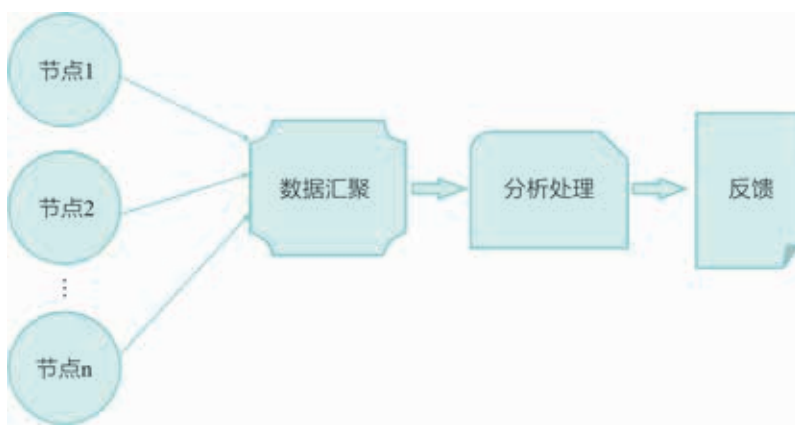


图4.2.2 简单的多节点数据采集系统模型

在多节点数据采集系统中，一般会利用多个硬件装置分别采集不同位置的数据，并将这些数据发送至同一个数据汇聚端，并进行统一分析处理。而实际应用中，需要监测的位置常常因为相距较远或其他客观原因不容易布线，如在胎压监测系统中，车轮处于运动状态无法使用有线方式进行通信，因此在多机通信项目中往往会采用无线数据传输方式进行多机通信。

在多机通信中，设备之间的数据传输模式有广播模式、点对点模式、中继模式等。在广播模式下，数据的发送端只需要负责向空中发送无线电信号，接收方监听收到的信号即可，一般适合用于小规模、近距离、对通信安全要求不高的场景；点对点模式，则采用一些底层协议，让两个设备之间建立较为稳定的连接，比较适合传输数据规模较大、对通信

安全要求较高的场景；中继模式，是通过中继手段让两个相距较远的设备进行数据通信的方式，适合远距离、大规模的组网。zigBee是常用于物联网项目的一种短距离、低功耗的无线通信技术，采用的就是中继模式。在zigBee网络中，每一个节点不仅用于自身收发数据，还用于其他节点的数据中继，如图4.2.3所示。

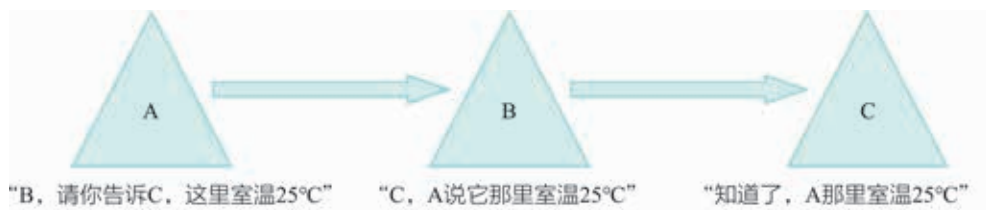


图4.2.3 中继模式下的数据传输

4.2.2 开发实例

在上课过程中，教师常常会设计课堂练习来巩固某一知识点，根据学生答题情况了解学生的掌握情况。有些学校通过在课堂上使用计算机和手机之类的智能终端，向教师及时反馈练习的结果。

目前，平板电脑价格较为昂贵，对许多学校来说还无法普及。我们可以利用开源硬件开发一个课堂答题器，用低成本的方式实现课堂练习的实时反馈。

问题与讨论

除了课堂的练习反馈，还有哪些应用场景需要收集反馈信息并且快速进行精确统计？结合你的生活实际，以小组形式讨论一个可行的解决方案。

1. 功能特点

“课堂答题器”又称“课堂即时互动答题系统”，是一种能够帮助教师快速收集并分析学生课堂答题情况的教学工具。该系统使用时，先由教师向学生提问，学生思考问题后使用手持式无线答题器提交答案，计算机上的配套软件会生成图表，即时显示学生答案的分布情况，然后教师根据反馈信息确定课堂进程。

通过对教师实际需求的调研，得到了如表4.2.1所示的项目分析表。

表4.2.1 课堂答题器的项目分析

需求	描述	分析
客观题作答	可以在某硬件上进行客观题的作答	学生通过终端答题，客观题的答题结果容易统计
多终端区分	能区分不同学生的答案	有很多学生会同时参与答题，区分不同学生的答案有助于学情跟踪

续表

需求	描述	分析
实时传输数据	可以及时收集所有学生的作答情况	教师需要快速了解学生的作答情况
记录保存	可以保存学生的答题记录	保存学生的答题记录，既有助于教师后续统计，又有助于跟踪每个学生的学习情况
统计结果呈现	可以统计学生的答题情况并用直观的方式呈现	教师不仅需要了解每个学生的答题情况，还需要从宏观上了解整个班级的情况，若能用图表方式呈现结果则更加直观
较低的成本	以较低的硬件成本实现该项目	成本必须控制，否则就失去该项目的意义

2. 项目规划

课堂答题器主要由答题终端、接收机和计算机服务端三大部分组成。答题终端接受学生输入的答案，并将身份标识信息和答案一起通过无线方式发送给接收机；接收机负责将收到的数据通过串口转发给计算机服务端；计算机服务端负责将数据存入数据库，同时运行一个网页服务器供教师访问以便查看答题结果统计。系统整体框架如下图4.2.4所示。

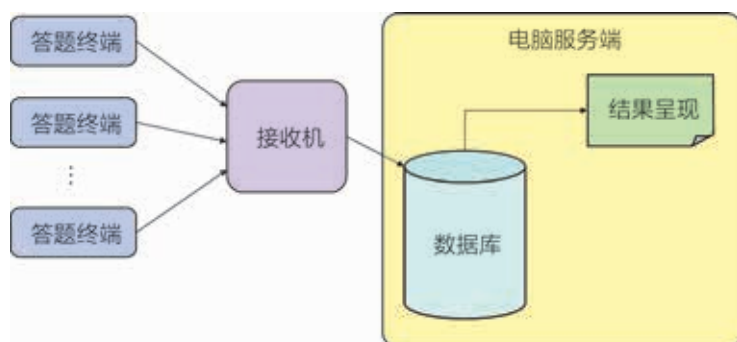


图4.2.4 课堂答题器系统整体框架

3. 硬件组成

考虑硬件的低成本、便携性以及无线通信的需求，因此选择micro:bit作为课堂答题器项目的主要支持硬件。micro:bit自带的两个按钮和5×5 LED阵列可以用于人机交互。答题终端只需一块micro:bit主控板和供电底板，适合手持操作，也容易携带，如图4.2.5所示。

为了方便实现与答题终端的数据传输，接收机依然采用micro:bit。接收机与教师计算机通过USB连接，以串口方式传输数据。



图4.2.5 加了供电底板的micro:bit

此外，还可以利用3D打印或者激光切割为答题终端和接收机各自设计制作一个外壳。外壳的设计并不复杂，只需把micro:bit的两个按钮位置、LED阵列以及用于充电的mini USB口位置留出来即可。

4. 软件设计

在课堂答题器项目中，软件设计是重点，大部分功能都需要软件来支撑。

“多终端区分”显然是一个重要的功能需求。有很多方式可以实现用户身份标识，例如，获取某一设备的序列号，并将序列号作为唯一标识；给设备进行编号，保证编号是不重复的，便可以作为标识。

拓展链接

设备序列号

设备序列号是某些设备在出厂时厂家预先写入的一段编号。该编号通常具有唯一性，即任意两个设备的序列号是不同的。硬盘、CPU、网卡、蓝牙模块等设备都有自己的序列号。

因为唯一性，所以序列号被广泛应用于各个场景。

- 软件的授权使用。某些软件被激活时，往往会根据计算机中某个硬件的序列号生成一个激活码，而通过该激活码申请的软件密钥只能用于该计算机。
- 身份验证。一些网络应用软件会检测当前用于登录的设备是否与之前登录过的为同一设备。若是新设备，为避免盗号等现象需要进行二次身份验证。
- 产品保修服务。有些设备若在官网查询产品的序列号就可以得知出厂时间和保修期限。

micro:bit的编程工具没有提供获取设备序列号的方法，因此需要事先写入一个包含唯一编号的文件。此外，在答题终端与接收机的通信中，还需要设计传输数据的格式。定义的数据格式如下：

```
u[user_id]t[answer]
```

其中，user_id为用户标识，answer为提交的答案。比如“us001tA”表示：用户s001提交了A答案。

为了方便代码编写，先给micro:bit写入唯一的标识文件。具体做法是准备一个文本文件“SN.txt”，它包含一行唯一的标识文本，比如“s001”“s002”……并将这个文本文件上传到micro:bit闪存中。使用读文件的方式，可以得到这一标识文本。

(1) 答题终端部分

答题终端的答案选择，可以借鉴空调遥控器对于空调的常见控制方式。空调的运行模式和风量控制都是通过一个按钮来循环切换不同的模式，因此可以利用micro:bit的两个按钮，一个按钮进行答案选择，另一个按钮表示确定，如图4.2.6所示。

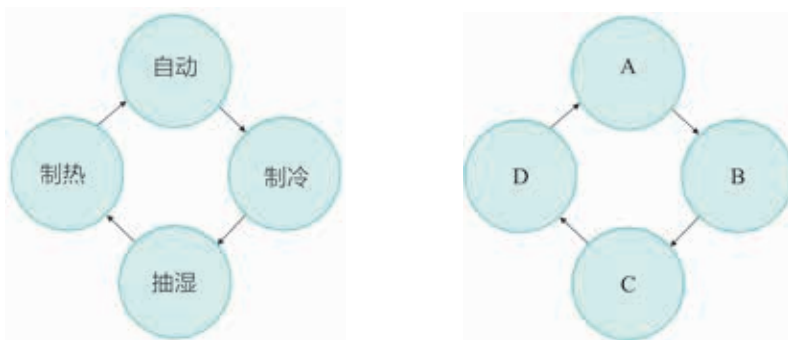


图4.2.6 空调遥控器循环切换运行模式和课堂答题器答案选择模式

通过以下代码可以读取保存在 SN.txt 文件中的设备编号，并保存到 usr 变量中。

```
import os
with open('SN.txt','r') as rf:
    usr=rf.read()
```

当选定答案后，可以通过如下语句将设备编号和答案一起以无线广播的方式发送出去。

```
radio.send("u" + usr + "t" + ans)
```

(2) 接收机部分

接收机的任务是接收由答题终端发来的数据。接收机首先将蓝牙配置成与发送端相同的信道，并不断地监听来自蓝牙的数据，若有数据则转发给计算机的串口。接收机部分的代码参考如下：

```
while True:
    receiver=radio.receive()
    if receiver!=None:
        uart.write(receiver)
        uart.write("\n")
```

(3) 数据存储部分

将学生的身份信息和答题信息传输给教师计算机后，这些信息要先保存到教师计算机上的数据库中。使用数据库保存数据的好处是方便检索，因为接收的数据量规模不大，且并发性也不高，选择 SQLite3 数据库即可。存储身份和答题信息需要两个类型为 STRING 的字段，如图 4.2.7 所示。

Table name: STU_D		<input type="checkbox"/> WITHOUT ROWID						
名称	Data type	Primary Key	Foreign Key	唯一	条件	Not NULL	排序规则	Default value
1	USR	STRING						NULL
2	ANS	STRING						NULL

图4.2.7 数据库的表结构

计算机服务端监听串口数据，若接收到的一段数据的首字符为“u”，则认为有效数据。首先以用户标识作为查询条件在数据表中统计该用户标识的记录数，若记录数为0，说明该用户首次提交答案，于是插入一行答题记录；否则，执行更新操作，确保数据表中记录的答题结果是该用户最后一次提交的数据。以下代码可以从串口读到数据并存储在 incoming 变量中。

```
import serial
ser=serial.Serial()
ser.baudrate=115200
ser.port='COM3'
ser.open()
incoming=ser.readline()
incoming=incoming.strip()
incoming=incoming.decode("gb2312")
```

Python 访问 SQLite 数据库，需要先导入 SQLite3 库，并通过 cursor 对象进行数据库操作。如下代码实现对 stu_data.db 数据库的“STU_D”表进行操作。

```
import sqlite3
conn=sqlite3.connect("stu_data.db")
c=conn.cursor()
sqlstr="INSERT INTO STU_D (USR,ANS) VALUES ('%s','%s')" %(usr,ans)
c.execute(sqlstr)
conn.commit()
```

(4) 数据呈现部分

为了方便教师及时查看答题情况，需要一个界面来呈现数据反馈结果。这部分代码相对比较简单，只需要读取 stu_data.db 数据库，查询各个答案的作答人数，并进行统计即可。查询选“A”的用户并统计人数可以参考如下代码：

```
sqlstr="SELECT USR FROM STU_D WHERE ANS='A'"
cursor=c.execute(sqlstr)
tmpdata=cursor.fetchall()
s_a="%s(%d)人" %(s_a,len(tmpdata))
for i in tmpdata:
    u_a=u_a+"选A\n"
print("%s\n%s\n-----\n" %(s_a,u_a))
```

问题与讨论

该项目还需要进一步完善，如学生答题后并不知道接收机是否已经收到信息，在无线通信中有数据发送与接收出现故障的情况。能否实现提交答案后，答题终端显示发送成功或者失败的提示？

III 实践与体验 III

课堂答题器的编程实现

课堂答题器项目需要实现客观题作答、答题结果统计等功能。请利用 micro:bit，通过实践完成这一项目。

实践内容：

1. 实现答题端从 A、B、C、D 中选择合适的答案并通过蓝牙发送到接收机。
2. 实现接收机将数据通过串口转发给计算机。
3. 实现服务端从串口获取数据并存储到数据库。
4. 实现教师端从数据库中读取结果并呈现出来。
5. 调试项目。

实践步骤：

1. 答题端、接收机的程序编写。
2. 服务端数据库操作的程序实现。
3. 实现教师端的结果呈现。
4. 项目调试。

在课堂答题器项目中，主要包括数据传输、存储、呈现三个环节。其中，数据传输环节可以通过计算机上串口监视器进行调试。将接收机通过 USB 线连接至计算机，并打开计算机的串口监视器，在答题终端通过 A 按钮选择不同的答案并按下 B 按钮后，可以看到如图 4.2.8 所示的字符信息，这表示数据传输环节正常。

关闭计算机的串口监视器，并运行计算机服务端的监听程序，继续通过答题终端答题，正常则会显示如图 4.2.9 所示的信息。



```
us001tD
us001tD
us001tD
us001tD
us001tD
us001tD
us001tD
us001tD
us001tD
us001tD
us001tD
us001tA
us001tB
us001tC
us001tD
```

图4.2.8



```
IPython console
Console 1/A
In [85]: runfile('/Users/james/mu_code/
receiver_to_sqlite.py', wdir='/Users/james/mu_code')
s001D
s001A
s001B
s001C
s001C
```

图4.2.9

通过SQLiteStudio软件打开数据库文件，可以看到数据表中被插入了记录。如图4.2.10所示，最后一次的记录是用户s001发送的C答案。

	USR	ANS
1	s004	A
2	s002	B
3	s006	A
4	s003	A
5	s005	B
6	s001	C

图4.2.10

通过运行教师端程序，可以看到答题结果呈现页面，如图4.2.11所示。

```
选A:(3)人  
s004  
s006  
s003  
-----  
选B:(2)人  
s002  
s005  
-----  
选C:(1)人  
s001  
-----  
选D:(0)人  
-----
```

图4.2.11

思考讨论:

如何通过修改程序代码，实现对答题者答题时间的记录?

4.3 基于物联网的项目开发

基于物联网的开源硬件项目能够实现物与物或者物与人之间的互联，并充分利用互联网技术，实现对各种物品的识别、管理和控制。当电饭煲被加入Wi-Fi功能后，就可以使用手机对其进行预热和加温等的控制，如图4.3.1所示；当空调与互联网连接之后，就可以在回家途中远程打开家中的空调，到家就能享受舒适的室内环境。



图4.3.1 支持物联网功能的电饭煲

4.3.1 基于物联网项目的特点

在很多物联网项目的模型中，除了包含人机互动、多机通信以外，还有一个很重要的因素是互联网。正是因为有互联网的支持，才可以突破空间的局限，实现任意距离的人与物、物与物的连接。

在项目里，大多数物联网设备本身不具有直接访问互联网的能力，它们通过多机通信方式相互通信，并通过一个类似于网关设备访问互联网。而用户对设备发送的信息也是通过互联网传递给网关，并由网关转发给各个设备。这种方式适用于智能家居等场景，其系统拓扑结构如图4.3.2所示。



图4.3.2 智能家居中的物联网

有些物联网应用因为要控制的设备较多，设备之间的间距较远，或者本身需要在移动中使用，所以设备本身需要具备访问互联网的能力。例如，共享单车上安装了一把智能锁，里面内置了移动网络通信模块，每把智能锁都可以通过GPRS、4G或NB-IoT等移动

网络连接到互联网。当用户扫码骑车时，通过移动网络向共享单车运营商发起开锁请求，而共享单车运营商也通过移动网络对特定的智能锁设备发送开锁命令，其系统拓扑结构如图4.3.3所示。

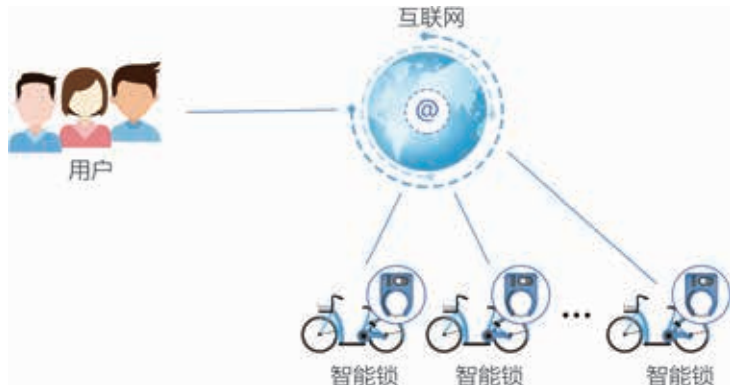


图4.3.3 共享单车中的物联网

4.3.2 物联网项目开发实例

市场上有出售智能花盆或智能盆栽等产品。如图4.3.4所示，这些产品大多可以通过网络获取花盆的环境信息，并对植物进行远程管理。



图4.3.4 智能花盆

智能花盆产品自带的花盆一般都很小，不能种体积较大的植物。利用开源硬件，结合土壤湿度传感器和光线传感器等传感器，将家庭中常见的盆栽改造成智能“盆栽伴侣”，是一项有意思的工作。

1. 功能特点

“盆栽伴侣”是一款能实时监测盆栽植物生长状态并实现智能浇灌的系统。这个项目可以和任意一个花盆结合，让普通的花盆变身为智能花盆，用户还可以通过互联网实时了解花盆的状态并进行远程浇水控制。项目的功能需求分析如表4.3.1所示。

表4.3.1 “盆栽伴侣”的项目分析

需求	描述	分析
实时监控	温度、光线等环境的监测，能够实时上传到Web服务器，通过Web平台能够实时查看植物的生长状态	采集这些数据，可以帮助用户科学地养护盆栽
智能浇灌	能够自动根据土壤湿度进行智能浇灌，也可以通过Web平台远程给植物手动浇灌	自动浇灌能够避免用户因为疏忽而“伤害”盆栽的现象，手动浇灌则能促进用户和盆栽之间的互动
感情互动	通过LED，以表情的形式显示植物当前的状态，当生长环境状态出现异常时能够实时报警	与植物互动不仅有趣，而且有意义
通用兼容	能够安装在常见的花盆中，操作方便	通用性强是项目最重要的优势

2. 项目规划

“盆栽伴侣”主要由智能终端、Web服务器两大模块组成。智能终端实时采集环境光线、环境温度和土壤湿度等信息，通过网络上传到Web服务器。Web服务器将数据存入数据库，并通过网页实时显示，系统框架如图4.3.5所示。

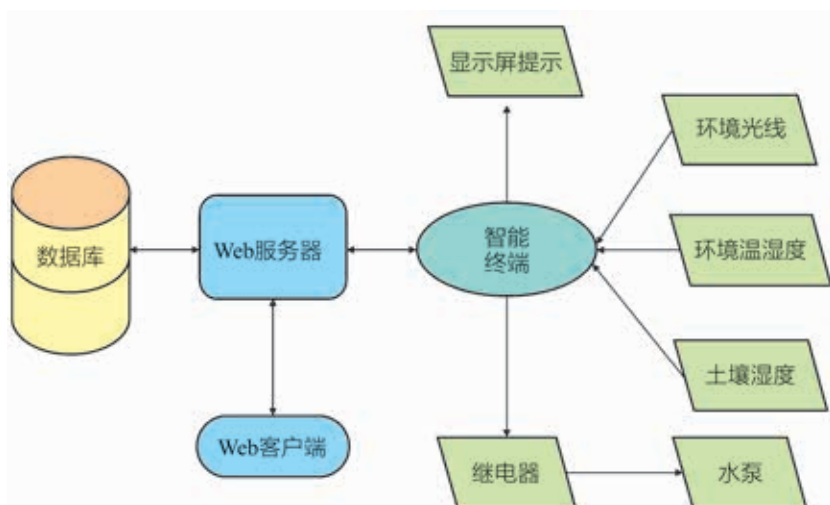


图4.3.5 “盆栽伴侣”整体框架图

3. 硬件组成

micro:bit自带LED阵列，可以用来做花盆的信息提示器。为了获取土壤湿度信息，需要外接一个土壤湿度传感器。micro:bit虽然自带温度传感器，但精度不高，所以需要使用第三方的温度传感器。为了实现互联网访问，需要Wi-Fi模块。为了实现浇水，必须有直流电水泵，但水泵需要较大的电压和电流驱动，因此还需要配备一个继电器模块，以实现用小电流控制大电流设备。

表4.3.2中列出了该项目所需的器材。

表4.3.2 器材清单

序号	器材名称	数量
1	micro:bit	1
2	micro USB线	1
3	温度传感器	1
4	土壤湿度传感器	1
5	光线传感器	1
6	micro:bit扩展板	1
7	Wi-Fi模块	1
8	潜水泵	1
9	潜水泵电源连接线	1
10	继电器模块	1
11	橡胶水管	1



图4.3.6 项目实物展示图

如图4.3.6所示为项目实物展示图。

3. 软件设计

“盆栽伴侣”涉及智能终端和Web服务器两大模块。二者之间的数据传输根据HTTP协议，通过GET或POST方法传送数据给Web服务器，Web服务器不仅实时记录数据以备后期分析，还会将当前数据显示出来。数据交换遵循如表4.3.3所示的格式。

表4.3.3 数据交换描述

数据交互动作	格式	返回值
智能终端发送数据及获取指令	/input?val=[环境温度],[环境湿度],[环境光线],[花盆湿度]	2, 浇水指令 1, 默认信息 0, 数据库出错
Web发送浇灌指令	/set?open=[0/1]	1, 表示进行浇灌 其他信息表示系统异常

智能终端每一次发送数据后，Web服务器返回的信息中包含了是否浇水的指令，即返回2则浇水，默认情况下是1。当智能终端获取了Web发送的浇灌指令后，启动水泵进行浇灌，持续1~2秒后停止。

因为需要存储的数据量不大，可以采用Python自带的数据库SQLite3。数据库中有一张sensorlog表，用于存储各种传感器数据。表中各字段设置如表4.3.4所示。

表 4.3.4 数据库设置

字段	数据类型	作用
logid	INTEGER, 自动编号	记录编号, 主键
sensor1	float	环境温度
sensor2	float	环境光线
sensor3	float	花盆湿度
updatetime	time	上传时间

Web 服务器采用 Python 的 Flask 框架编写。首先为 Web 服务器编写一个主页模版 index.html, 并在模版中插入一个 table 标签。

```
<table border="1" width="80%" cellspacing="0" >
  <tr height="40" align="center" width="30%">
    <td><b>环境温度</b></td>
    <td>{{sensor1}}</td>
    <td rowspan="3">
      <h2 align="center"><a href="/">刷新</a></h2>
      <h2 align="center"><a href="/set?open=1">浇水</a></h2>
    </td>
  </tr>
  <tr height="40" align="center" width="30%">
    <td><b>环境光线</b></td>
    <td>{{sensor2}}</td>
  </tr>
  <tr height="40" align="center" width="40%">
    <td><b>花盆湿度</b></td>
    <td>{{sensor3}}</td>
  </tr>
</table>
```

在 Flask 框架中, 通过 render_template 绑定该模板, 并将从数据库中查询到的环境数据显示到主页中。参考代码如下:

```
@app.route("/")
def hello():
    db = sqlite3.connect(DATABASE)
    cur = db.cursor()
    cur.execute("SELECT * FROM sensorlog")
    data = cur.fetchall()
    cur.close()
    db.close()
    temp1 = data[len(data)-1]
    return render_template('index.html', sensor1=temp1[1], sensor2=temp1[2], sensor3=temp1[3],
    sensor4=temp1[4])
```

Web 服务器获取智能终端上传的环境数据, 且保存到数据库中。参考代码如下:

```

#用于接收智能终端发出的http数据，并储存至文件data.db中。同时根据浇水变量的值
#返回相应的浇灌指令
@app.route("/input",methods=['POST','GET'])
def add_data():
    try:
        global watering
        if request.method == 'POST':
            sensorvalue = request.form.get('val')
        else:
            sensorvalue = request.args.get('val')
        ss=sensorvalue.split(',') #
        if len(ss)==4: #
            nowtime = datetime.datetime.now()
            nowtime = nowtime.strftime('%Y-%m-%d %H:%M:%S')
            db = sqlite3.connect(DATABASE)
            cur = db.cursor()
            cur.execute("INSERT INTO sensorlog(sensor1,sensor2,sensor3,sensor4,updateime)
VALUES(%f,%f,%f,%f,'%s')"%(float(ss[0]),float(ss[1]),float(ss[2]),float(ss[3]),nowtime))
            db.commit()
            db.close()
            if watering == 1:
                watering = 0
                return '2'
                #当浇灌变量为1时，返回字符串“2”，使其启动浇灌2秒钟，同时将浇灌变量重置为0
            else:
                return '1'
                #当浇灌变量不为1时，返回字符串“1”，使其保持停止浇灌状态

        else:
            return '0'          #指令无法被识别或通信出错时，返回字符串“0”
    except:
        return '0'          #指令无法被识别或通信出错时，返回字符串“0”

```

作为智能终端的 micro:bit 通过串口与 Wi-Fi 模块通信，定时向 Web 端传递温度、湿度等环境数据，同时根据 Web 端的返回值确定是否浇水。当 Wi-Fi 模块配置好网络参数后，通过 GET 方式向 Web 端上传一组环境数据，根据返回的信息判断是否浇水。实现上述功能的程序代码参考如下，其中继电器模块接在 pin8 口。

```

temp=pin0.read_analog()
light=pin1.read_analog()
moisture=pin2.read_analog()
errno,resp = Obloq.get("input?id=1&val="+str(temp)+' '+str(light)+' '+str(moisture)+' '+str(wateri
ng),500)
if error==200:
    if resp=='2':
        pin8.write_digital(1)
    else:
        pin8.write_digital(0)

```

```

sleep(2000)
else:
    pin8.write_digital(0)
    display.scroll("error")

```

III 实践与体验 III

“盆栽伴侣”的编程实现

“盆栽伴侣”是一个综合性比较强的项目，不仅要实时获取花盆的环境信息，还要实现远程给植物浇水。该项目最大的困难在于对各个模块的编程实现。通过编写代码，可以更好地理解这一项目各个模块是如何协同工作的。

实践内容：

1. 编写 Web 服务器代码，实现将花盆环境的信息存储在数据库。
2. 实现智能终端通过 Wi-Fi 模块上传环境数据，并获取 Web 端的返回值。
3. 通过 Web 页面实时显示温度传感器、光线传感器、土壤湿度传感器的值，并实现对继电器模块的控制。
4. 调试项目。

实践步骤：

1. 在 Python 环境下，利用 Flask 框架编写 Web 服务器代码，使得服务器端可以正确接收 GET 或 POST 方式的传值，“盆栽伴侣”物联网系统中温度、湿度显示的 Web 页面代码如图 4.3.7 所示。在编写代码过程中，可以先通过 Chrome 浏览器模拟给 Web 服务器发起 GET 请求。

“盆栽伴侣”物联网系统			
V0.1			
刷新			
环境温度	{{sensor1}}	环境光线	{{sensor3}}
环境湿度	{{sensor2}}	花盆湿度	{{sensor4}}

图4.3.7 Web页面代码示例

2. 连接 micro:bit 和 Wi-Fi 模块，编写 Web 服务器通信代码，实现将测试数据发送给 Web 服务器，并从 Web 服务器中获取返回值。

3. 将 DHT11 温湿度传感器、光线传感器、土壤湿度传感器与 micro:bit 连接，编写代码获取这些传感器的值；通过 micro:bit 的数字口控制继电器动作，实现对水泵的控制。

4. 项目整体调试。

这个项目的调试难点是micro:bit通过Wi-Fi模块与Web服务器进行连接。因为micro:bit和Wi-Fi模块的通信是使用UART功能，如果发生错误，难以判断问题在Web服务器还是micro:bit端，建议参考下面两种方法来调试。

方法1：利用面包板，将连接Wi-Fi的引脚并联出来，连接到一个串口模块上，同步将信息发送到计算机串口。

方法2：利用无线功能，将UART的所有信息同步发送到其他micro:bit板子。作为接收信息的micro:bit板子，再用print的方式发送到电脑查看。如果在教室里多人同时使用会造成信息干扰，要加地址码进行区分。

思考讨论：

1. 如果通过互联网来控制花盆的浇水，这个信息系统的网络应该如何搭建？
2. 如果花盆处于严重干旱的情况下，要想智能终端即使没有接收到浇水指令，也能自行给花盆浇水，应该如何实现？

思考与练习

请找三款热销的智能花盆产品，了解具体参数并填写下面表格。

产品	智能浇灌	状态监测	通知提醒	支持手机管理	其他优势

巩固与提高

1. 跑步机是常见的室内锻炼器材。新型的跑步机能够模拟虚拟街景，让跑步者产生一种漫步森林或沙滩的错觉。结合百度街景这一应用，普通跑步机也能改造为新型的跑步机，如用加速度传感器结合左右按钮来控制街景的浏览。请使用开源硬件，设计一个跑步机的改造方案，并做出原型。

2. 某创客空间的学生曾经设计过一款睡眠检测仪，利用加速度传感器来监测用户晚上睡眠时翻身的次数，以此来衡量睡眠的质量，如图4.3.8所示。micro:bit自带了加速度传感器，并且体积较小，适合随身携带。请编写程序让micro:bit“变身”为睡眠检测仪。



图4.3.8 睡眠检测仪检测结果示例

3. “盆栽伴侣”的提示功能，目前只能通过LED阵列来实现。请用App inventor2为“盆栽伴侣”开发一款APP。APP能够实时读取Web服务器的数据，接收来自花盆的提示信息，并显示不同的表情图片，让植物和用户之间产生有趣的互动。

项目挑战

创客训练营——优化开发我们的项目

所有的开发项目都有一个持续迭代优化的过程，随着技术的发展与互联网+思维的影响，开源硬件项目的迭代周期更短、速度更快。

▶ 项目任务

在完成初步功能的开源硬件项目基础上，进一步优化开发，并形成更符合整体目标的项目原型。

▶ 过程与建议

1. 优化项目方案

根据本章所学的内容，重新审视自己当前的项目，判断实现总体项目的可行性和创新价值，如果有必要，重新修订项目方案。

2. 优化开发

在原有的开发基础上，进一步丰富和优化功能，美化外观，使项目的整体价值不断提升。

3. 理解开发的局限

项目中所使用的各种部件都存在一定的局限性。比如用于测量人体体温时要求有较高的精度，而测量锅炉温度时又要求较大的温度测量范围。同样作为动力输出部件的电机，在有些场景中要求高转速，如无人机的旋翼；而有些场景又要求较大的扭矩，如升降机中的电机。请通过咨询、文献搜索或者操作体验，深入了解自己项目中各种部件的适用场景，比较更多的同类产品，从更高的层面理解不同部件的优势与局限，更好地改进自己的项目。

部件名称及型号	规格参数	适用场景	局限性

4. 拓展学习与开发

开源硬件的开发文档是非常丰富的，绝大多数的扩展模块使用都有相应的教程，请根据需要选择学习，并进一步优化开发项目。

▶ 评价标准

请根据项目实施的过程、效果以及成果展示交流的结果，对自己完成项目的情况进行客观的评价，并思考后续完善的方向。把评价结果和完善方案填写在下面的表格中。

评价条目	说明	评分（1~10分）	评分主要依据阐述	后续完善方向
团队合作	整个团队对自己的项目充满热情，持续修订与发展项目			
项目质量	项目的完成水平具有较高的技术含量			
项目优化	对项目进行了有效的优化			
拓展提高	加分项：自学新的部件，并实现了原有项目不能实现的功能			

▶ 拓展项目

1. 借助如图4.3.9所示的红外接收头（也称红外接收模块，即IR Receiver Module），micro:bit能接收到家电遥控器的红外控制信号。请修改“电脑魔法控制器”项目，用家电遥控器来控制计算机里视频的播放。

2. 多肉植物的生长需要足够的光照，光照强度会随着时间变化而变化，而花盆是固定的。能否给“盆栽伴侣”加上轮子，让它们学会找“阳光”？检测阳光需要多个光线传感器，再通过特定的算法预测阳光移动的方向。另外，还需要借助大功率驱动板和电机让花盆动起来。请利用光线传感器、大功率驱动板和电机，做一个“移动的花盆”。



图4.3.9 红外接收模块

开源硬件项目的发布与维护



《道德经》中说：“一生二、二生三、三生万物。”利用开源硬件开发项目，就是在前人的基础上进行创新。当我们从零到一，完成了一个从创意到实现的项目后，下一步的工作就是在开源社区上继续开源，从而让创意生生不息。

项目一旦开源，其生命周期就得到延续。吸引更多志同道合的人参与到项目中并逐步完善的前提，是项目文档的友好。一个优秀的开源硬件项目从来不是一蹴而就的，而是需要通过一次次的迭代与创新使其焕发更强的生命力。

问题与挑战

- 开源不等于“免费”，开源项目的发布一般都附带相应的开源许可协议。这些协议确定了项目的开源方式、项目发布者的权益保护以及使用者的义务等内容。那么，开源项目方面有哪些协议？在开发项目时，使用的开源硬件遵循什么协议？又应该如何去履行这些开源许可协议的要求？

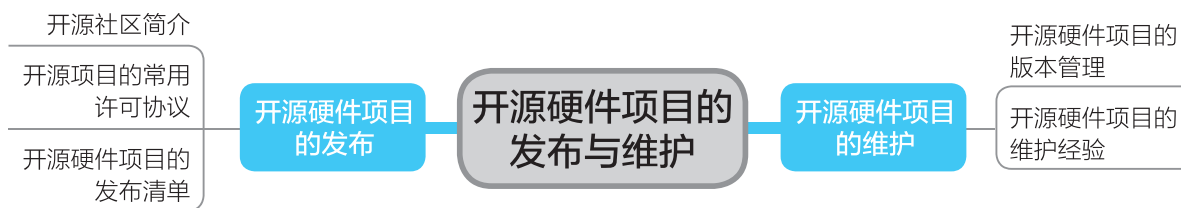
- 一个项目的开源，需要发布在相关的开源社区，才能让更多有同类爱好或者需求的用户（开发者）快速找到。国内外有多个为开源项目提供发布服务的社区，还提供了相对完善的开源许可协议。在完成一个或多个硬件项目后，如何选择适合这个开源项目的分享平台和协议？

- 一个项目的开源发布，代表着这个项目要面对全世界对它感兴趣的用户（开发者）。为了让更多的人愿意参与到这个项目中，一起完善功能，项目开发者要提供完整的开发文档、使用说明，甚至包括详细的代码注释等。那么如何整理这些项目资料？除了整理资料以外，对一个项目进行正常维护还需要注意哪些细节呢？

学习目标

1. 了解开源社区，能够在常见开源硬件社区发布自己的项目。
2. 了解各种开源协议，践行开源与知识分享的精神，理解保护知识产权的意义。
3. 完善项目作品的设计方案，理解项目维护的意义，体会开源精神的内核。

内容总览



5.1 开源硬件项目的发布

开源的精神在于分享，开源的价值在于由分享带来的迭代创新。因而，利用开源硬件去做项目，其实是一种知识的传承，不仅继承了前人的成果，而且加上自己的创新并分享给更多的人，是“从受惠到反哺”的一种体现。但是，开源项目的“开源”并不等同于随意将项目发布在某个论坛、个人博客或网盘上，而是要选择合适的开源社区，确定认可的开源协议，按照约定的格式准备资料，再发布出去。本教材所述的各种开源硬件开发，遵循开源许可协议，体现“契约精神”。

5.1.1 开源社区简介

要让更多人看到自己的开源项目，最好的方法是把项目发布到一些人气旺盛的开源社区。国外知名度较高的开源社区有GitHub、SourceForge和Arduino.cc等。国内的开源社区以分享软件代码为主，如开源中国、码云和CSDN等。在国内外的开源社区中，汇集较多硬件项目的开源社区有GitHub、Instructables、Arduino.cc、创酷网和DF创客社区等。

GitHub于2008年上线，是全球著名的开源项目协作社区。GitHub设计了多种机制让用户和开发者协同对项目进行处理，有如下特点：

① GitHub是一个跨平台的社区，可以在任何终端上，随时随地共享项目，或者评论他人的项目。

② GitHub提供了一些方便社会化共同软件开发的功能，即社区功能，包括允许用户追踪其他用户、组织、软件库的动态，对软件代码的改动和Bug提出评论等。

③ GitHub还提供了图表功能，用于显示开发者怎样在代码库上工作以及软件的开发活跃程度。

GitHub是全球最大的代码托管平台和开源社区之一。大部分开源项目，包括软件和硬件项目，都会选择在GitHub上发布，如图5.1.1所示的是开源硬件Arduino和micro:bit项目的页面。通过GitHub可以了解大家对这些开源项目的评价，以及项目的更新情况。



Arduino项目



micro:bit项目

图5.1.1 GitHub上的著名项目

Instructables 是一个创意分享平台，分享动手制作物品的社区，由美国麻省理工学院媒体实验室创办。Instructables 上有很多好玩的项目，分为科技、生活、美食、户外和娱乐等类别，如“科技”栏目中有关于做雾霾探测器、自制背包防盗报警器之类的项目，“户外”栏目中有教用户如何磨刀的内容。这些项目图文并茂，用户可以发表内容或评论，也可以进行投票。2013年，Instructables 推出了中文版，不过相对来说，中文的项目还比较少。

Hackaday，顾名思义是“Hack a Day”，即改造一整天，是技术爱好者聚集的社区。Hackaday 上提供了各种适合在创客空间、车库中开展的动手项目，是为有一定技术素养、喜欢自己动手改造技术产品的玩家准备的。需要强调的是，Hack 在这里并非“黑客”的“黑”，而是指改造、创新。

Arduino.cc 是开源硬件 Arduino 的用户社区。从2005年开始，来自全世界的 Arduino 用户在 Arduino 社区上发布了数以万计的项目。这些项目都是以 Arduino 为核心，涉及人工智能、智能家居、互动媒体、科学探究等方面。很多项目经过迭代而开始量产，真正成为进入寻常百姓家的消费品。

DF 创客社区是一个国内的线上创客学习和交流的平台，面向创客和想要成为创客的人们。用户可以在社区里分享创客项目和学习使用体验，并且得到各种技术支持。该平台积累了许多易用的学习教程和富有创意的创客项目，主要内容资源涵盖了电子制作、3D 打印、机器人和机器人教育等领域。国内几名创客在2014年开发了创客网，目标是打造中文版的“Instructables”。

问题与讨论

在学习开源硬件的过程中，难免会遇到各种各样的困难。开源社区中汇集了各种技术爱好者，他们能提供很多的帮助。请结合自己的学习经历，对开源社区在推动开源硬件普及方面所起的作用做出评价，并提出相关建议。

5.1.2 开源项目的常用许可协议

开源项目使用的许可协议详尽表述了他人获得代码后所拥有的权利。许可协议有很多种，根据 Open Source Initiative 的统计，当前全球的开源许可协议大概有上百种，常用的有 GPL、BSD、MIT、Mozilla、Apache、LGPL 和 CC 等。部分协议的权利和义务如图 5.1.2 所示。

许可证	版本	包含许可证	包含源代码	链接	状态变化	商业使用	散布	修改	专利授权	私人使用	授权转售	无担保责任	没有商标
Apache许可证	2.0	是			是	是	是	是	是	是	是	是	是
3句版BSD许可证		是				是	是	是		是	是	是	是
2句版BSD许可证		是				是	是	是		是	是	是	
GNU通用公共许可证	2.0	是	是		是	是	是	是	是	是	否	是	
GNU通用公共许可证	3.0	是	是		是	是	是	是	是	是	是	是	
GNU通用公共许可证	2.1	是	是	是		是	是	是	是	是	是	是	
GNU通用公共许可证	3.0	是	是	是		是	是	是	是	是	是	是	
MIT许可证		是				是	是	是		是	是	是	
Mozilla公共许可证	2.0	是	是			是	是	是	是	是	是	是	是
Eclipse公共许可证	1.0	是	是			是	是	是	是	是	是	是	
Affero通用公共许可证		是	是		是	是	是	是		是	是	是	
一般的著作权 [note 1]		是				是	否	否		是	否		

图5.1.2 开源软件协议比较

1. 常见的开源软件协议

(1) GNU GPL 协议

GNU General Public Licence (GNU GPL) 是开源界最常用的许可模式之一。GNU GPL 保证了所有开发者的权利，同时为使用者提供了足够的复制、分发和修改的权利。

①可以自由复制。可以将软件复制到其他计算机，或者任何媒介上。复制份数没有限制。

②可以自由分发。可以从网站下载后拷贝到U盘，或者将源代码打印出来分享。

③可以用来盈利。在分发软件的时候可以收费，但收费前必须向客户提供该软件的GNU GPL许可协议。

④可以自由修改。可以添加或删除某个功能，也可以在别的项目中使用部分代码。但使用代码的项目也必须遵循该代码的GNU GPL协议。

(2) GNU LGPL 协议

GNU 还有一种协议，即LGPL (Lesser General Public Licence)，它对产品所保留的权利比GPL少。总的来说，LGPL 适用于那些非 GPL 或非开源产品的开源类库或框架。因为GPL要求，使用了GPL代码的产品也必须使用GPL协议，开发者不允许将GPL代码用于商业产品，而LGPL规避了这一限制。

(3) BSD 协议

BSD 在软件分发方面的限制比别的开源协议要少。该协议有多种版本，最主要的版本有新BSD 协议与简单BSD 协议两种。这两种协议经过修正，都与GPL 兼容，并为开源组织所认可。

BSD 协议在软件分发方面，除了需要包含一份版权提示和免责声明以外，没有任何限制。另外，该协议还禁止在衍生产品上使用开发者的名义。

(4) MIT 协议

MIT 协议的核心条款是开源软件及其相关文档对所有人都免费，可以任意处置，包括使用、复制、修改、合并、发表、分发或再授权，以及销售。唯一的限制是，软件中必须包含上述版权和许可提示。

(5) CC 协议

CC 是 Creative Commons 的简称，在中国大陆的正式名称为“知识共享”。CC 是一种开源作品的授权方式，一般用于设计、摄影或者图片作品的开放式分享。CC 是一个相对宽松的版权协议，仅保留了原作者的如下四种权利：

- ①署名 (Attribution, 简称为 BY)：必须提到原创者。
- ②非商业用途 (Noncommercial, 简称为 NC)：不得用于盈利性目的。
- ③禁止演绎 (No Derivative Works, 简称为 ND)：不得修改原作品，不得再创作。

④相同方式共享 (Share Alike, 简称为 SA)：允许修改原作品，但必须使用相同的许可证发布。






这四种权利可以自由组合，其中是最宽松的协议版本。它可以简单表述为：只要在使用时署名，那么使用者可以对原创作品进行转载、节选、混编、二次创作以及商业目的的使用，具体的标志如表 5.1.1 所示。

表 5.1.1 CC 协议的标志说明

标志	缩写	全称	说明
	BY	姓名标示	您可以自由复制、散布、展示及演出本作品；您必须按照作者或授权人所指定的方式，保留其姓名标示
	NC	非商业性	您可以自由复制、散布、展示及演出本作品；您不得为商业目的而使用本作品
	ND	禁止改作	您可以自由复制、散布、展示及演出本作品；您不得改变、转变或改写本作品
	SA	相同方式分享	您可以自由复制、散布、展示及演出本作品；若您改变、转变或改写本作品，仅在遵守与本著作相同的授权条款下，您才能散布由本作品产生的衍生作品

常见协议的区别，如图 5.1.3 所示。由于 CC 协议不是固定不变的，所以没有放在里面进行比较。

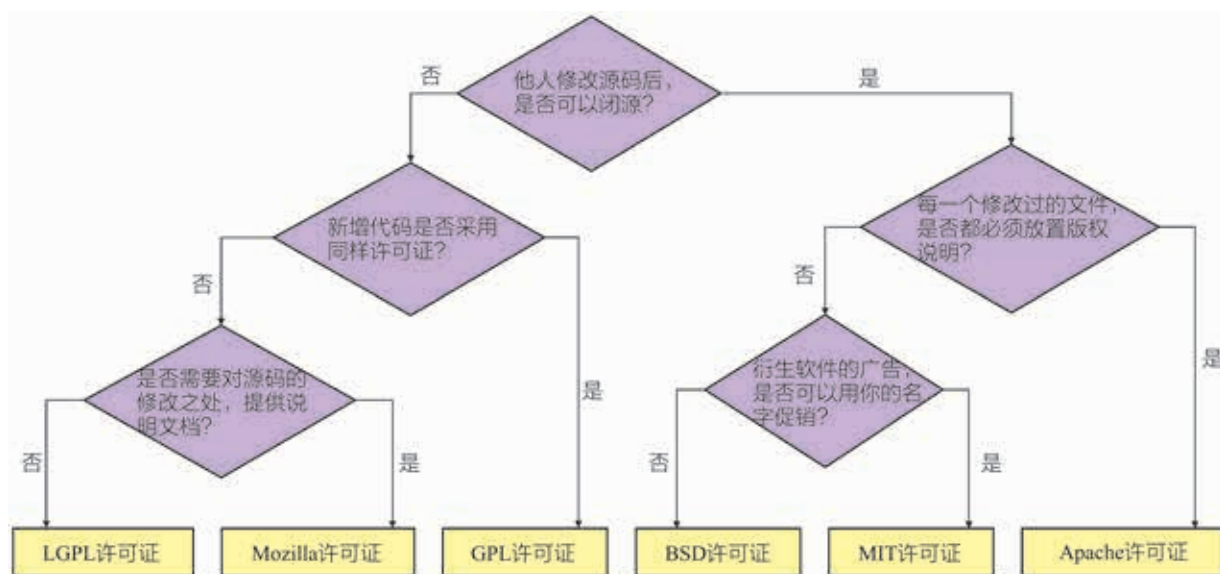


图5.1.3 常见开源协议的区别

2. 常见的开源硬件协议

相对来说，硬件开源比软件开源要复杂。一个开源硬件项目除了硬件部分的设计图纸以外，还会涉及软件代码以及和项目相关的文档等。软件代码通常由 GPL、MIT 等开源软件协议授权使用。开源硬件主要通过 TAPR、CERN 和 Solderpad 等协议授权使用。文档（包括文字、美术、音乐、摄影和多媒体创作等作品）主要通过 CC 协议授权使用。

开源硬件许可证延续着开源软件许可证的规则，把开源软件常用的 GPL、CC 等协议带到硬件领域。目前已经出现的若干开源硬件协议中，最突出的有 TAPR、CERN 和 Solderpad 三种。

(1) TAPR

TAPR 开源硬件协议是 GPL2.0 授权条款在硬件领域的衍生版本。与后者不同的是，TAPR 增加了修改者通报机制：如果设计图的开源者留下了电子邮箱，后来的参与者修改了这份以 TAPR 授权的设计文件，那么必须将修改后的设计文件发送到开源者留下的电子邮箱，或通过此电子邮箱联系开源者进行修改通报。这种义务性规定在开源软件领域里几乎是看不到的，主要是因为在开源硬件的范畴里，开源者较难知道后来的参与者对设计文件进行了修改。修改后的硬件难以通过网络平台进行共享，所以 TAPR 才刻意在授权条款中增加了修改者通报机制。

(2) CERN

CERN 开源硬件协议是欧洲核子研究组织发布的授权框架，是将 GPL2.0 与著佐权 (Copyleft) 相关的理念与做法，转移到开源硬件的领域。在条款内容上，CERN 也设置了类似 TAPR 中修改者主动通报的义务。但在 CERN 中，可以采用由开源者建议的任何方式来

通报，并没有限制必须以电子邮件形式。而与TAPR差异更大的是，CERN的授权条款更为全面与细致，这是因为TAPR侧重于无线电装置设计的开放应用，而CERN则关注未来有可能采用开源模式发展的其他硬件。

拓展链接

Copyright和Copyleft

Copyright（著作权）指著作权所有，即软件的一切权利归软件作者私有，其标志是“(C)”。著作权的准确定义随着司法权的不同而不同，但其本质就是：作品的作者对于作品的复制、表现等有一定的垄断性。在美国，宪法明确阐述了美国国会的任务就是制定著作权法律来“促进科学和实用艺术的进步”。和其他产权形式不同的是，著作权会立刻附加到作品上——而且不需要注册。默认情况下，所有的权力都是保留的。也就是说，没有经过作者的允许，没有人可以重新出版、表现或者修改作品。这种“允许”就是一种许可，可能还会附加一定的条件。

Copyleft（著佐权）是针对Copyright提出的，即反对软件一切权利归作者私有，保护知识共享、权利共享，其标志是“(O)”。Copyright是保留所有权利，在商用软件中常用。而自Linux出世后，为了区别就用了Copyleft来表示其自由和免费。GNU项目为Copyleft提供了简单的定义：当重新分发程序时，你不可以添加限制来否认其他人对于“自由软件”的自由权利。这可以被认为是Copyleft的权威定义。

可以用一个等式来阐述这两个概念： $\text{Copyleft} = \text{Copyright} + \text{GPL}$ 。

(3) Solderpad

Solderpad开源硬件协议与前面介绍的TAPR和CERN在授权特性上有着极大的不同，是目前市面上少有的完全不带Copyleft性质的开源硬件授权条款。TAPR与CERN都是由GPL2.0授权条款从软件领域衍生至硬件领域的，而Solderpad是由Apache-2.0授权条款衍生而来的。Solderpad在条款内容中首先便声明，它直接参考Apache-2.0授权条款的架构来进行编撰，其条款与Apache-2.0授权条款之间具有等价的转换关系。简单来说，当开源者以Solderpad公开其硬件设计文件之后，得到此设计文件的参与者，除了遵循Solderpad的授权规则来使用这份设计文件以外，还可以在需要时将设计文件的全部或部分内容，改以Apache-2.0的授权方式来进行利用。

micro:bit是一个开源项目，其中硬件设计部分使用Solderpad协议，硬件的图片使用的是CC协议中的CC BY，而软件部分（MicroPython）使用了MIT协议，如图5.1.4和5.1.5所示。



图5.1.4 micro:bit的硬件设计使用Solderpad协议

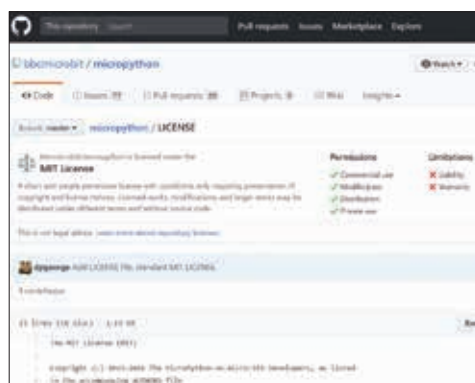


图5.1.5 micro:bit的MicroPython使用MIT协议

micro:bit的硬件设计部分使用了Solderpad协议，具体描述如下：

Copyright 2016 microbit Educational Foundation. Copyright and related rights are licensed under the Solderpad Hardware License, Version 0.51 (the “License”). You may not use this file except in compliance with the License. You may obtain a copy of the License at <http://solderpad.org/licenses/SHL-0.51>. Unless required by applicable law or agreed to in writing, software, hardware and materials distributed under this License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

5.1.3 开源硬件项目的发布清单

一个较为完整的开源硬件项目在发布前，一般需要准备四个方面的资料：相关程序源码（文件夹）、硬件文档（文件夹）、著作权许可说明文本和项目说明文档（Readme）。

相关程序源代码文件夹中应当包含该项目中涉及并且原创者愿意公布的程序源代码，该源代码可以是运行在开源硬件上的程序代码，也可以是运行在PC端的程序代码，当然还可以是运行在服务器端的脚本代码。

硬件文档文件夹中应当至少包含该开源硬件项目的硬件原理图，若该项目的硬件可以实现量产，则还可以包含其PCB（Printed Circuit Board，印制电路板）文档。

著作权许可说明文本中写明该开源硬件项目使用的是哪种许可协议，也可以详细阐述别人使用自己的项目时应当遵循的原则和规范。为了方便起见，有些文档直接提供协议的链接地址。

说明文档包含了该项目的简介、使用方法、注意事项等。开源硬件项目发布的目的是为了更方便别人使用，因此需要一个详尽的项目说明文档。

micro:bit项目的硬件部分在GitHub上提供了四个文件：使用说明、著作权许可、硬件说明、硬件清单，如图5.1.6所示。

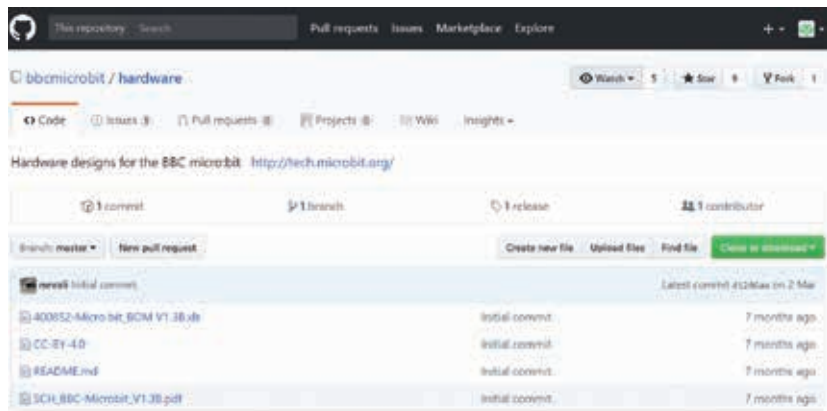


图5.1.6 micro:bit的项目截图

思考与练习

因为micro:bit的技术门槛较低，适用于中小学编程教育，所以国内外有不少企业利用micro:bit开源的资料，设计了类似的硬件。认真查阅Solderpad协议，这些企业使用micro:bit的开源资料开发的项目，怎么做才能够符合这一开源许可协议，以便既保护自己的权利，又保护micro:bit项目开发者的权利？

5.2 开源硬件项目的维护

一个项目在开源社区开源后，它的生命周期就开始得到延续和发展，包括开发者在内，每个人都有权利根据开源协议对其进行迭代和创新。项目开源后，开发者通常根据用户的反馈继续迭代更新。

5.2.1 开源硬件项目的版本管理

在项目开发过程中，常常会因为需求的变化和解决方案的改变而形成多个不同的版本。开源硬件项目因为不仅有软件代码，还有硬件选择和结构搭建，比普通的软件项目更加复杂。“牵一发而动全身”，一个地方的改动往往会引发另一些问题。所以，开发者需要保留每一次修改的版本，当发现某个改动虽然解决了眼前的问题却导致更大的问题时，可以回复到之前的版本。项目开发的管理者要有效解决问题，就必须采用一些机制，如版本管理。版本管理的任务就是对项目的历史演变过程进行记录和维护，做到统一、协调管理各个版本，有效记录不同版本的演变过程，以尽可能少的数据记录各版本，并能够快速生成新版本。

GitHub 不仅是一个开源项目的分享平台，也是一个著名的版本管理系统，被称为版本控制系统。国内也有类似的网站平台，如码云。BXY（如图 5.2.1）就是用码云来做版本管理的。



图5.2.1 BXY项目主页

拓展链接

开源分布式版本控制系统 Git

Git是一个开源的分布式版本控制系统，可以有效、高速地进行项目（从很小到非常大）版本管理。Git 本来是为帮助管理 Linux 内核开发而开发的一个开放源码的版本控制软件，但因为开源、简单、快捷、分布式、高效等特点，被软件开发者广泛接受并使用。

5.2.2 开源硬件项目的维护经验

有人在总结自己的开源之路时，提出了项目生长、运营推广、社区维护三个方面。如图5.2.2所示，这条开源之路的目标指向是“影响力”，因为开源的目的是让更多人使用，让自己的创意和成果真正发挥作用，同时得到很多人的反馈，相当于拥有一个大型的开发团队。所以开源硬件项目不仅要运营推广，还要社区维护，并对使用情况进行及时反馈。

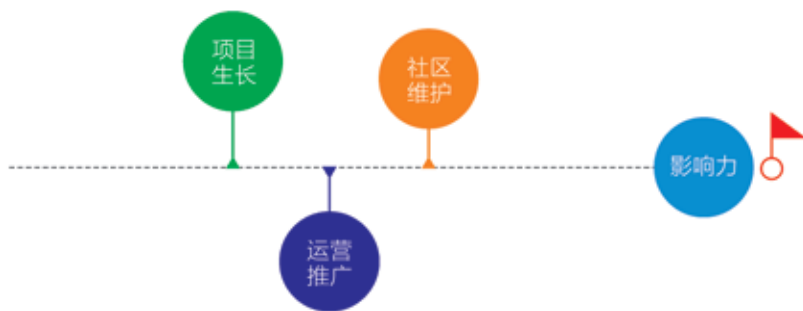


图5.2.2 对“开源之路”的经验总结

开源硬件项目的维护比开源软件项目的维护更为复杂。因为硬件容易发生一些不可预见的情况，如一个型号不对的螺丝或一个供电不稳的电池就可能导导致信息系统运行出错。因而开源硬件项目的维护是开源硬件开发中非常重要的环节。依据经验，开源硬件项目的维护一般要注重完善项目说明文档，根据反馈意见改进创意，实现项目迭代。

1. 完善项目说明文档

提供完善的使用文档非常重要，说明文档可以让用户快速了解项目内容并吸引用户产生兴趣。其次，撰写项目文档的过程，对项目来说也是一次迭代。在撰写文档时，常常会发现一些之前考虑不周到的内容，正好可以找到问题，并及时改进。

项目说明文档尽量采用英文或中英文混合的形式撰写。GitHub 上的项目文档大多是英文版本的，而中国是一个世界大国，在开源项目方面应该有更多的输出，英文版本的文档介绍能使你更有效地与更多国家的开源项目爱好者交流。

为了让项目的说明文档结构清晰、内容完善，在开始撰写时应该做好规划。图 5.2.3 所示的是一个常用的说明文档规划示例。

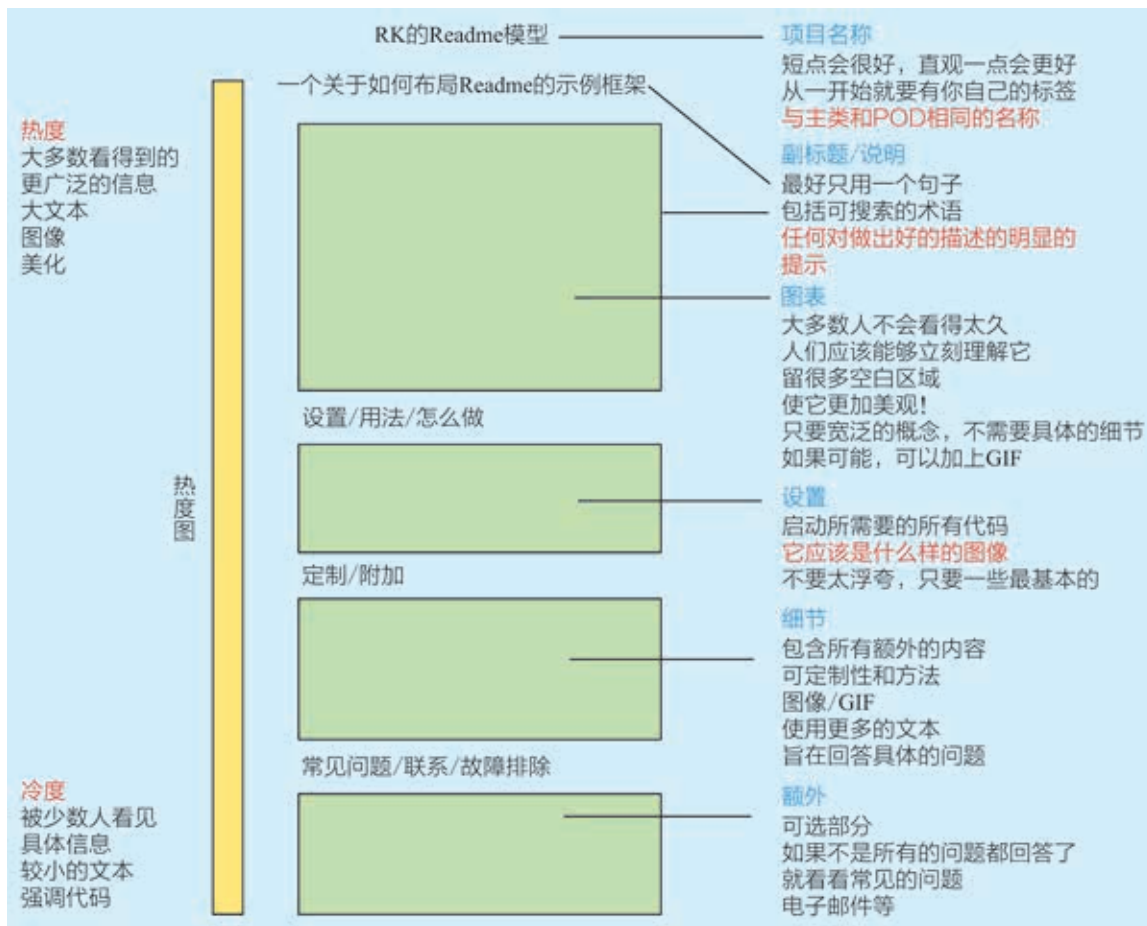


图5.2.3 常用的说明文档规划

2. 重视反馈改进创意

当一个项目在做原型开发时，开发者总是会先利用身边现有的材料，尽快做出作品。这背后隐藏了一定的风险：有些材料可能难以买到。对于不开源的硬件作品，的确可以不关注零件的来源。但开源项目面向的是全世界，如果其他用户无法快速获取硬件清单中提供的材料，肯定会影响这一项目的推广使用。因此，项目中材料的选择应该尽可能考虑使用标准件，越常见越好。

如果这个硬件项目要批量生产，那么还要考虑到供应链的问题。供应链是指产品生产和流通过程中所涉及的原材料供应商、生产商、分销商、零售商以及最终消费者等成员组成的网络结构，由物料获取、产品加工以及最终产品送到用户手中这一过程所涉及的所有企业或者企业部门组成。在这个网络中，任何一个节点出现问题，都会导致硬件项目量产的失败。供应链的问题，有时很难在“规划设计”环节发现，只能在后期维护的时候从用户的反馈中获取。



如果一个项目开源后遭受很多人批评，别灰心，这恰好说明有很多人在关注、使用这个项目产品。创意是可以慢慢改进的，谁都有犯错的时候。

总的来说，想运营好自己的开源硬件项目，要“怀揣一颗热情的心”，并保持一种激情，去迎接每一次改动。

巩固与提高

1. 整理自己项目中用到的硬件或者软件，确认这些开源项目分别使用的许可协议，研究这些协议并且为自己的项目写一个开源许可文件。
2. 整理项目文档，然后认真撰写 Readme 文件，尝试在词典和各种翻译软件的帮助下，写一份符合要求的英文版 Readme。

项目挑战

创客训练营——发布项目

通过近一个学期的学习、实践与开发，开源硬件项目已具雏形，你们对这个项目的信心如何？将采用怎样的方式来发布这一项目。

▶ 项目任务

从完成、展示、发布的角度重新审视所开发的开源硬件项目，并采用适当的方法面向用户（或假想用户）进行展示与推介。

▶ 过程与建议

1. 完善项目

在完成一个项目的时候，需要从发布的角度对自己的项目进行全面的检查，并进行完善。请利用下表自查，并根据自查结果进行修订完善。

（1）产品可用性方面的考虑：

- 如何让一个不了解此项目的人快速了解此项目的功能？
- 如何让一个初次使用的人快速知道使用方法？

（2）产品吸引力方面的考虑：

从用户的角度审视这个产品，还可以做哪些改善（外观、互动……）使这个产品更具吸引力？

（3）开源许可协议方面的考虑：

- 哪些人在这个产品中做了贡献？他们的贡献以什么形式在产品中标示？
- 这个产品应用了哪些开源软件和硬件？怎样做是遵循了开源许可协议？

2. 发布前的思考

历时半年的成果即将面对用户（或假想用户）了，需要考虑发布平台和展示内容，请从以下几个方面思考。

- （1）这个产品要解决的问题是什么？该产品从哪些方面解决了这个问题？
- （2）该产品最重要、最独特的价值是什么？
- （3）该产品目前已做到了什么？未来将要完成的理想样态是什么样的？
- （4）该产品离发布还有怎样的距离？

请将这些思考，在全班范围内进行展示和汇报。

3. 发布产品

如果你们的产品已达到可以发布的水准，可以思考以下问题：

- (1) 选择发布到哪个开源社区上？为什么选择这个社区？这个社区对发布的项目有哪些要求？还要做哪些努力才能完成发布？
- (2) 怎样才能让你的产品被更多的人知道？

▶ 评价标准

请根据项目实施的过程、结果和交流效果，对自己完成项目的情况进行客观的评价，并思考后续完善的方向。将评价结果和完善方案填写在下面的表格中。

评价条目	说明	评分（1~10分）	评分主要依据阐述	后续完善方向
项目整体质量	项目的技术含金量高，能够解决实际问题			
项目吸引力	项目创意强，且外观具有吸引力			
项目介绍	此项目的介绍对于用户而言具有吸引力			
项目发展	该项目具有发展前景			

▶ 拓展项目

1. 在DF创客社区上找一个自己感兴趣的开源硬件项目，根据项目开发人提供的各种资料，搭建硬件、编写代码，将这一项目调试成功并提出修改建议。
2. 在GitHub上注册一个用户账号，仔细阅读GitHub上发布项目的英文版教程，然后创建一个放置项目的仓库，发布自己的项目并进行项目维护，根据各种反馈进行项目迭代。