



广东省义务教育教学用书



信息技术

信息技术

八年级 下册

广东教育出版社课程教材研发中心 编著

 广东教育出版社

广东省义务教育教学用书



信息技术

信息技术

八年级 下册

广东教育出版社课程教材研发中心 编著

穆 肃 黄国洪◎主编

 广东教育出版社

· 广州 ·



致同学们

亲爱的同学们，很高兴和你们相会在信息世界！

信息技术很有用。你想成为信息技术的能手、网络文明的使者吗？在这里，同学们通过学习、探索信息技术，梦想就能实现啦！

在这本教材里，同学们将以制作有趣的Scratch作品为案例学习运用计算机解决问题的方法和步骤，掌握程序设计的基本控制结构，学会用模块化程序设计方法编写程序；通过学习常见的智能控制器、传感器等开源硬件，开展项目创作活动，学会自主、探究和协作学习，共同完成项目任务。

爱因斯坦曾说过：“兴趣是最好的老师。”本教材以“完成学习任务”的方式或“问题解决”方式引导同学们展开学习过程，并设置了“学习任务”“合作探究”“自主探究”“小组讨论”“阅读理解”“巩固练习”“知识链接”“小提示”等栏目，以及丰富多样的主题式综合活动，以激发同学们学习和应用信息技术的兴趣，在参与多种多样的活动中提高信息素养。

同学们，让我们一起走进轻松、愉快的信息技术课堂，携手度过一段美好的时光！

编者





目 录

第一单元 程序设计初步 / 1



- 第一课 计算机解决问题的基本过程 / 2
- 第二课 程序与程序设计 / 7
- 第三课 选择结构的程序设计 / 12
- 第四课 多重条件选择结构程序的实现 / 19
- 第五课 循环结构的程序设计 / 25
- 第六课 嵌套循环程序的实现 / 33
- 第七课 结构化程序设计 / 42
- 第八课 综合活动：无人驾驶汽车自控系统 / 48
- 单元扼要回顾 / 56
- 单元学习评价 / 57

第二单元 开源硬件设计 / 59



第九课 认识开源硬件 / 60

第十课 软件与硬件的结合使用 / 67

第十一课 数字信号与模拟信号 / 78

第十二课 使用传感器采集信息 / 86

第十三课 控制输出模块工作 / 96

第十四课 电动机与舵机 / 104

第十五课 数码显示与无线通讯的实现 / 113

第十六课 综合活动 / 123

单元扼要回顾 / 135

单元学习评价 / 136

附录一 常用开源硬件说明 / 138

附录二 程序模块说明 / 143

第一单元

程序设计初步

知识导图

程序设计初步

计算机解决问题的基本过程

程序与程序设计

选择结构的程序设计

多重条件选择结构程序的实现

循环结构的程序设计

嵌套循环程序的实现

结构化程序设计

综合活动：
无人驾驶汽车自控系统



自1946年第一代电子计算机问世以来，计算机不仅自身朝着超大型化、微型化和智能化方向持续发展，而且在应用方面已经遍及人类社会生产和生活的每个角落，可谓到了“无孔不入”的地步。

在计算机被普遍使用的今天，是否具备一定的编程能力已经成为衡量一名青少年是否胜任未来工作的重要标准。在未来的工作场合中，通过编程，可以利用计算机超强的计算能力解决很多问题，如计算机自动控制、计算机辅助设计、计算机辅助制造、计算机决策、新武器模拟试验、计算机辅助教学、计算机辅助医疗、计算机控制交通等，都少不了计算机的帮忙！那么，计算机是怎样解决问题的呢？

本单元将通过几个典型问题，用Scratch编程工具，展示运用计算机解决问题的方法与步骤，包括问题分析、算法设计、程序设计和上机调试等。

第一课 计算机解决问题的基本过程



学习目标

- 了解算法、编写程序和调试运行程序的概念。
- 了解计算机解决问题的基本过程。

知识导图



现代社会，无论是科学研究、国防军工还是工农业生产、学习生活，计算机的影响已无处不在，人们已经习惯于运用计算机来帮助解决问题。在很多领域，计算机甚至已经能代替人的工作，而且效果好、效率高。如近几年来，我国农业生产中逐渐采用智能操控无人机进行农业喷洒作业，大大提高了生产效率和效果。

无人机是通过计算机程序来实现智能操控的，下面我们一起来模拟运用计算机程序实现无人机智能操控的过程。

学习任务

编写程序，模拟实现植保无人机在指定农田区域内自动操控飞行，完成喷洒作业。

一、分析问题

求解任何问题，首先要明确问题的内容。植保无人机通过导航飞控系统，自主适应多种地形环境，对农作物开展全覆盖施药施肥喷洒作业。而现实生活中，农田的地形环境往往是不规则的。如图1-1所示，红线框内是要进行喷洒作业的农田区域，植保无人机在装填基地加装好药液后，自

提示

小

RTK (Real-time kinematic, 实时动态) 载波相位差分技术, 是一种新的、常用的GPS测量方法, 能够在野外实时得到厘米级定位精度。

动操控从喷洒作业起点a出发, 沿着程序设定的航线(黄色线)依次经过b、c、d等转向点。由于农田形状不规则, 因此转向点的坐标也没有规律性。目前有一种卫星测量技术RTK技术, 能够精确测出这些转向点的卫星定位坐标。

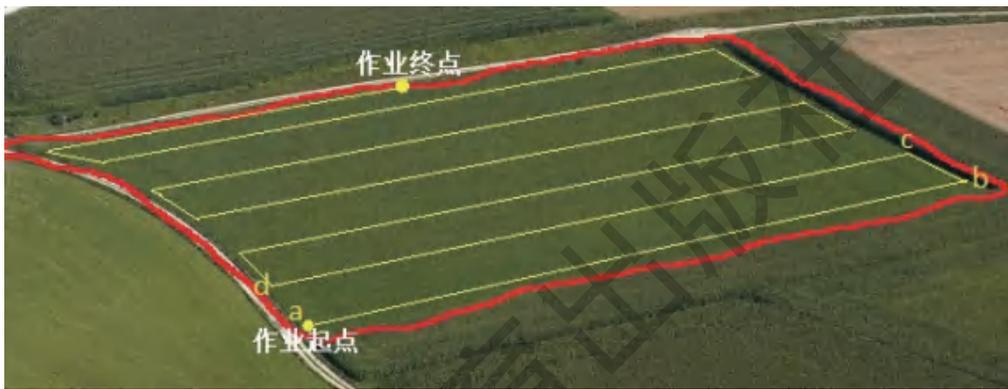


图1-1 喷洒作业仿真图

二、设计算法

按照上面的问题分析, 我们把具体的操作步骤写出来:

第一步: 使用RTK技术测量出无人机航线中各转向点的卫星坐标(包括起点和终点), 并使用链表来记录其坐标值。

第二步: 装填农药, 起航。

第三步: 植保无人机通过卫星坐标控制转向, 从a点(喷洒作业起点)开始依次经过b、c、d等转向点, 自控飞行喷洒作业, 直到作业终点。

第四步: 喷洒作业完毕, 返航到装填基地。

以上就是控制植保无人机在指定农田区域内自动操控飞行完成喷洒作业的方法和步骤, 我们称之为“算法”。

三、编写程序

确定算法以后, 我们要选定一种计算机可以执行的程序设计语言来描述这种算法, 这个过程就是编写程序。同学们还记得以前学过的Scratch编

程语言吗？本单元我们通过Scratch编程来学习程序设计的知识。

探究练习

用Scratch编程工具编写程序，模拟植保无人机自动操控飞行喷洒作业的过程。程序代码如图1-2和图1-3所示。



图1-2 无人机代码



图1-3 输入转向点坐标代码

四、调试程序

搭建好脚本指令后，要通过运行程序的方法对指令进行适当的调试修改、完善和美化。如果脚本程序有逻辑错误或计算方法错误，可以从程序执行的结果去判断，对程序存在的问题进行相应的修改和进一步完善，直到得到满意的结果。

拓展思考

在现实农业生产中，除了地形环境不规则之外，农田中可能还会有电线杆等障碍物，喷洒区域中还可能存在池塘等非农田区域或喷洒作业中药量用完等状况，我们该如何编程处理？

以上就是运用计算机来实现无人机智能操控喷洒作业的过程。

利用计算机解决问题，需要人们完成一系列的程序设计任务，把所要解决的问题转化为计算机程序，然后让计算机来执行这个程序，最终达到利用计算机解决问题的目的。这个解决问题的过程一般分为以下几个步骤：

（1）分析问题。

任何一个问题必须弄清楚其内容、性质和规模，才能找到解决问题的方法。在程序设计开始时，必须收集与问题相关的资料，分析该问题所涉及的输入数据和结果要求，确定该问题的功能要求、性能要求及其他要求。

（2）设计算法。

确定了程序应该“做什么”之后，就要为它设计一个“如何做”的合适的算法。一个算法给出一个求解某一个问题的方法和步骤，是解决该问题的一系列清晰的、精确的指令。由于计算机本身是不能进行逻辑思维的，任何时候都是按照人们预先安排的指令机械地完成各种操作，因此人们运用计算机解题时，要先设计算法并用适当的方式把它准确地描述出来。

（3）编写程序。

编写程序就是用一种计算机能接受的程序设计语言来描述问题求解的算法。因此，在编写程序前要先选定一种程序设计语言。一个好的、适用于解决具体问题的语言可以使程序的结构清晰、简洁，有利于正确地记述待解决的问题，同时还可以正确地表示过程。

（4）调试程序。

程序编写完成后要进行检查和调试，调试程序的目的是查找和改正程序中存在的错误，使程序能顺利地执行，得出正确的结果。

小组讨论

小组交流探讨用计算机解决问题与人工解决问题的异同，并把讨论的结果记录在表1-1中。

表1-1 关于解决问题方式的讨论记录

解决问题的方式	相同点	不同点
人工解决问题		
用计算机解决问题		



反思评价

对自己在本课中的学习进行评价，对照表1-2，在相应的○里画√，在“其他收获”栏里留言。

表1-2 学习评价表

序号	学习内容	能否掌握
1	了解算法、编写程序和调试运行程序的概念。	能○ 不能○
2	了解计算机解决问题的基本过程。	能○ 不能○
其他收获：		

第二课 程序与程序设计



学习目标

- 了解程序和程序设计。
- 知道程序的三种基本控制结构。
- 了解程序顺序结构的执行流程。
- 能够用顺序结构流程图描述算法。
- 能够编写简单的顺序结构程序。

知识导图



计算机本身是不能进行逻辑思维的，它任何时候都是按照人们预先编写好的程序机械地完成各种操作，本课我们一起来了解程序与程序设计。

一、程序和程序设计

程序实际上是一组计算机操作的指令或语句的序列，是算法面向计算机的一种描述。计算机是不会自动解决问题的。如果要运用计算机解决问题，程序员必须把问题逐步变换成程序，这个过程包括分析问题、设计算法、编写程序、调试程序，保证程序准确地解决问题。程序员把需要计算机做的工作写成一种计算机能够接受的程序的过程就是程序设计。

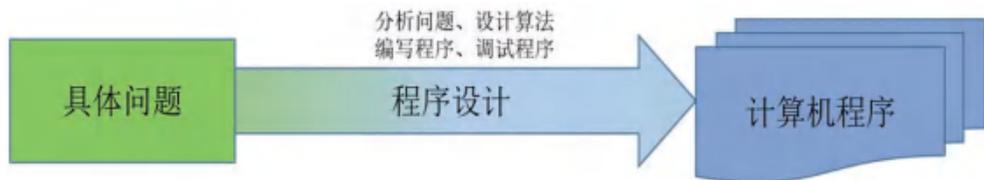


图2-1 程序设计过程

二、算法的描述

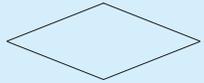
算法是人们求解某一个问题的方法和步骤，是对解题过程的精确描述。程序设计中的算法是指用计算机解决问题的一系列清晰、精确的指令。在运用计算机程序解决问题的过程中，算法设计的作用举足轻重，不同的算法会产生不同的运行效果，算法的好坏直接影响问题解决的效率。

在程序设计中，表示算法的语言主要有自然语言、伪代码、流程图等。上节课关于无人机自控飞行算法的描述，使用的是自然语言，就是人们日常所用的语言。

伪代码是用介于自然语言和计算机语言之间的文字和符号来描述算法的工具。它回避了程序设计语言严格的书写格式，方便且易于理解，缺点是语句不规范，容易产生误解。

流程图是用程序框图来描述算法的一种常用工具，它用表2-1所列的一组图形符号来表示算法。流程图便于体现程序结构，它不依赖于任何具体的计算机和计算机程序设计语言，能清晰、简洁地描述算法，从而有利于不同环境的程序设计。

表2-1 流程图的基本图形及其功能

图 形	名 称	功 能
	开始 / 结束框	表示算法的开始或结束。
	处理框	表示算法中的计算或处理等。
	判断框	表示算法中的条件判断。
	流程线	表示程序的执行顺序。

三、程序的基本结构

程序是由若干个基本结构组成的，一个基本结构可以包含一个或多个语句，每个语句对应一个或多个操作。一般地，程序有三种最基本的结构：顺序结构、选择结构和循环结构。任何复杂的算法都可以用这三种基本结构组合而成的程序表示，所以这三种结构称为程序的三种基本控制结构。

我们可以用如图2-2所示的算法流程图（其中，S和 S_1 、 S_2 … S_n 表示按顺序执行的语句；C表示判断条件）来表示这三种基本控制结构的一般形式：

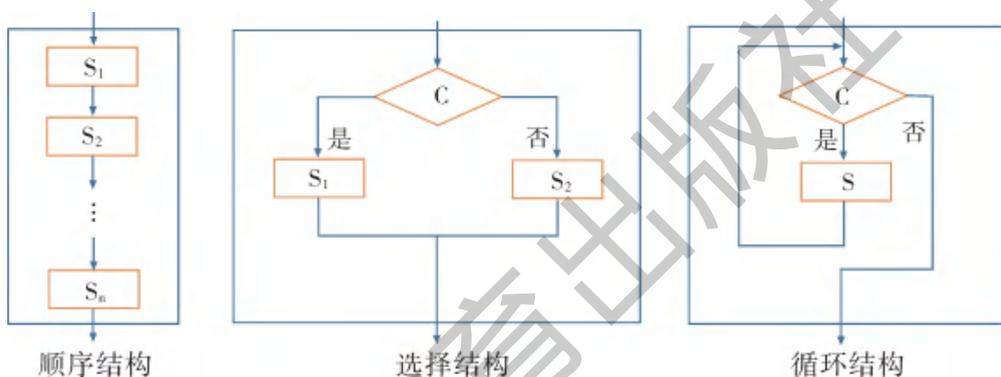


图2-2 程序的三种基本控制结构

使用三种基本控制结构的组合来表现程序，可以改善程序的清晰度，提高程序的可读性。

四、顺序结构的程序设计

阅读图2-3中的Scratch程序脚本后不难发现，程序执行时是按语句出现的先后顺序执行的。即当执行第一条语句后，必然从第二条语句继续，顺序执行所有的语句，直到执行完最后一条语句为止。像这样的程序控制结构称为程序的顺序结构，是最简单、最基本的程序结构，也是结构化程序设计的基础。

程序顺序结构的执行流程如图2-4所示。程序中的“语句1，语句2，……，语句N”，便是一组顺序结构的语句。



图2-3 顺序结构的程序

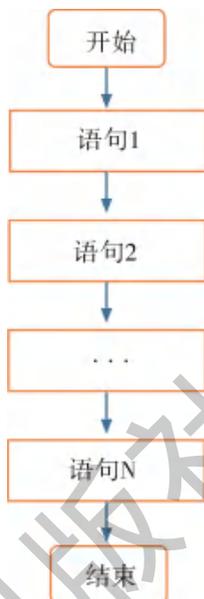


图2-4 顺序结构的程序执行流程

运行图2-3中的Scratch程序脚本，观察顺序结构程序的执行流程。

自主探究

1. 运用本课学习的程序顺序结构的知识，编写一个简单的Scratch程序，程序设计过程中要求用顺序结构流程图来描述算法。

2. 参考范例：编写一个顺序结构的Scratch程序，制作快递小哥派送货物场景的作品。

(1) 分析问题。

快递小哥派送货物的过程按时间顺序至少可分为接单、送货上门和顾客签收三个阶段，因此可以设计三个场景来描绘。

(2) 设计算法。

用顺序结构流程图描述本算法，如图2-5所示。



图2-5 快递派送算法流程图

(3) 编写程序。

根据上述设计的算法，在Scratch中自主编写程序脚本。

(4) 调试程序。

运行程序，观察顺序结构程序的执行流程，对程序存在的问题进行相应的修改和进一步完善。

巩固练习

自己动手制作或者上网搜集素材，深度加工、完善作品。

反思评价

对自己在本课中的学习进行评价，对照表2-2，在相应的○里画√，在“其他收获”栏里留言。

表2-2 学习评价表

序号	学习内容	能否掌握
1	了解程序和程序设计。	能○ 不能○
2	知道程序的三种基本控制结构。	能○ 不能○
3	了解程序顺序结构的执行流程。	能○ 不能○
4	能够用顺序结构流程图描述算法。	能○ 不能○
5	能够编写简单的顺序结构程序。	能○ 不能○
其他收获：		

第三课 选择结构的程序设计



学习目标

- 了解程序中选择结构的执行流程。
- 能够用选择结构流程图描述算法。
- 能够编写选择结构程序。

知识导图



顺序结构程序执行时是按指令出现的顺序一条一条地执行。但在现实世界中，不可避免地会遇到需要进行选择、判断的情况。比如汽车在行驶中，如果前方遇到车辆或障碍物，那么驾驶员一般都会采取减速停下或变道避让的处理方法。在这种情况下，采取顺序程序结构显然难以处理，这时就需要使用选择结构来实现程序的设计。

利用程序设计实现识别与障碍物的距离并自动避让，是无人驾驶汽车应用选择结构程序设计的实例。

一、无人驾驶汽车

目前，世界上已有多个国家研发并测试了无人驾驶汽车，最先进的无人驾驶汽车已经测试行驶几十万公里，这是未来汽车发展的一种方向。

无人驾驶汽车在行驶过程中，当正前方有车辆时，能够自主识别与前方车辆的距离。如果两车距离小于安全值（比较接近）时，无人驾驶汽车会暂停行驶；如果前方车辆远离，无人驾驶汽车则重新开始前进，如图 3-1 所示。

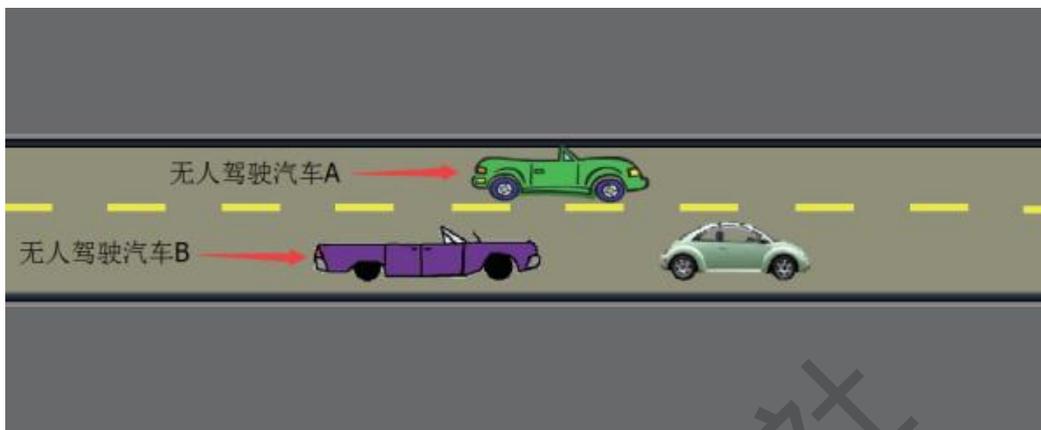


图3-1 无人驾驶汽车行驶情景

学习任务

请你根据以上描述，编写程序，实现无人驾驶汽车自动避让前方车辆的功能。

1. 问题分析

无人驾驶汽车能够自主识别并调整与前方车辆的距离：当在同一车道上时，与前车的距离应大于安全距离；当在不同车道上时，则汽车继续行驶。

2. 设计算法

第一步：初始化无人驾驶汽车，设定大小、位置等；

第二步：如果距离（前车的x坐标-无人驾驶汽车的x坐标）的绝对值大于90或者偏移（前车的y坐标-无人驾驶汽车的y坐标）的绝对值大于18，则向前移动0.5~2步，否则移动0步；

第三步：重复执行第二步；

第四步：结束。

算法设计流程图如图3-2所示。

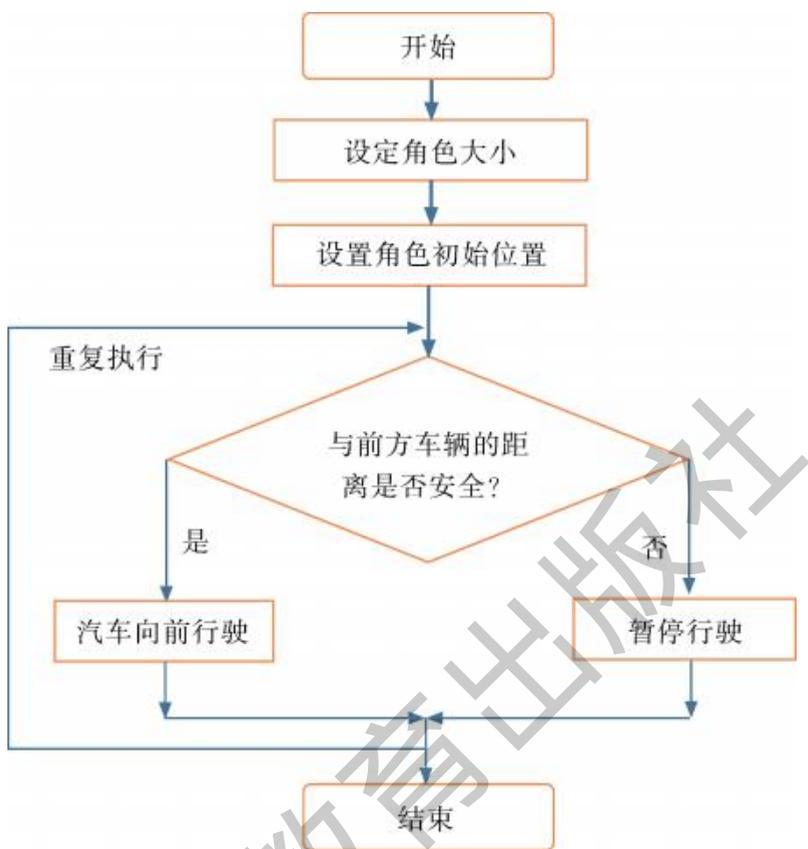


图3-2 无人驾驶汽车角色的算法设计流程图

3. 设计程序

根据算法，设计无人驾驶汽车角色的程序代码如图3-3所示。

图3-3 无人驾驶汽车角色的程序代码

4. 调试运行程序

单击“绿旗”运行程序，运行效果如图3-4所示。



图3-4 程序运行效果

运行结果显示，当前方有车辆时，无人驾驶汽车会根据识别距离做出相应判断：如果与前方距离较近时，自动暂停向前行驶，否则继续行驶。

二、条件语句

在无人驾驶汽车自动避让车辆的程序中所使用的程序代码（如图3-5所示）被称为“条件语句”。



图3-5 条件语句

Scratch中条件语句的两种不同形式，如表3-1所示。

表3-1 两种条件语句指令代码

Scratch指令代码	流程图	代码解释
		如果条件为真，则执行代码块。
		如果条件为真，则执行代码块1，否则执行代码块2。

其中，条件是关系表达式或逻辑表达式，将作为“如果”的条件，其结果为“真”或“假”；“那么”和“否则”之后存放需要执行的指令代码块。

三、程序的选择结构

在日常生活中，常常会碰到这样的例子：（1）如果下雨，我就带雨伞出门；（2）如果明天不下雨，学校就组织我们去郊游，否则就去看电影。

类似这种需要进行选择的情况，在结构化程序中我们称之为“选择结构”。选择结构的特点是它根据给定的条件判断进行选择，做出不同的决定。无论条件是否成立，不同的处理结束以后都执行其后续语句。

例如，当手机“电量”参数小于10时提示“电量不足，请及时充电”的程序如图3-6所示。



图3-6 电量提示程序代码

自主探究

编写一个程序：用户输入三个数字，输出其中的最大值。

（1）问题分析。

（2）算法描述。

（3）程序设计。

（4）调试运行。

巩固练习

(1) 完善本节课小汽车角色的脚本代码, 实现用键盘控制小汽车的行驶;

(2) 在本节课所设计的程序的基础上进行修改, 让无人驾驶汽车识别判断红绿灯。

反思评价

对自己在本课中的学习进行评价, 对照表3-2, 在相应的○里画√, 在“其他收获”栏里留言。

表3-2 学习评价表

序号	学习内容	能否掌握
1	了解程序中选择结构的执行流程。	能○ 不能○
2	能够用选择结构流程图描述算法。	能○ 不能○
3	能够编写选择结构程序。	能○ 不能○
其他收获:		

第四课 多重条件选择结构程序的实现



学习目标

- 学会创建变量及其赋值。
- 了解程序的多重条件选择结构。
- 能够用多重条件选择结构流程图描述算法。
- 学会编写简单的多重条件选择结构程序。

知识导图



在Scratch控制指令中有如图4-1所示的两种条件选择语句，用于处理两种情况的判断和选择问题。



图4-1 两种条件选择语句

一个条件语句可以解决简单选择问题的，可是生活中面临的很多问题是一个条件语句无法解决的，如本课例子中的区域物流中心负责中转发往八个城市的货物，分拣系统需要判断选择的情况就有八种，那么应该如何编写计算机程序来解决这类问题呢？

一、无人仓智能分拣系统

某物流公司一个区域物流中心负责中转香港、澳门、广州、深圳、珠海、佛山、中山、东莞八个城市的货物。为了应付日益庞大的货物中转需求，提高物流中心的运行效率，物流中心采用了无人仓智能分拣系统，以数百台智能分拣机器人代替人工分拣货物，货物处理量实现数倍增长。

学习任务

请根据以上介绍，编写程序实现分拣机器人根据目的地智能分拣处理货物的过程。

1. 分析问题

物流中心处理中转货物的流程大致可以分为接收货物入仓、分拣货物和出仓发货到目的城市三个部分：分拣机器人在货物入仓车间接收运送带送来的货物，在自动分拣中心根据目的地把货物分拣到运送目的地对应的出口，然后在出仓发货车间把货物投送到配送运输车中，运往目的城市。

中转目的地有多少个，货物、配送运输车就有多少个造型，我们用变量“目的地”来控制货物和运输车的造型。为了使分拣机器人与其他设备同步操作，需要设置“接货”“发车”和“发货”等状态变量来控制程序各角色的同步。

2. 设计算法

根据上述分析，设计解决问题的算法。

场景a：接收货物入仓（参考场景效果如图4-2所示）

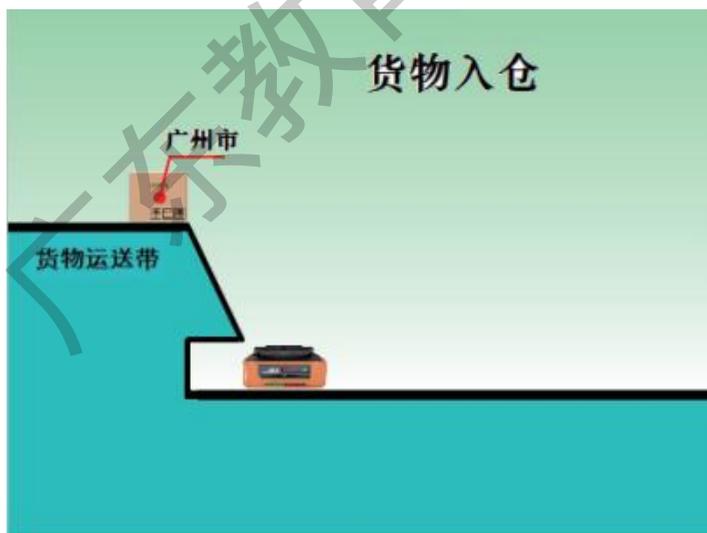


图4-2 货物入仓场景

(1) 分拣机器人运行到接收货物的位置等待，将变量“接货”赋值为1。

(2) 当“接货”=1时，设定变量“目的地”的值为1~8之间的随机数，根据变量的值来选择货物的造型：

如果“目的地”=1, 那么选择货物造型为“广州”;
 如果“目的地”=2, 那么选择货物造型为“深圳”;
 如果“目的地”=3, 那么选择货物造型为“珠海”;
 如果“目的地”=4, 那么选择货物造型为“香港”;
 如果“目的地”=5, 那么选择货物造型为“澳门”;
 如果“目的地”=6, 那么选择货物造型为“佛山”;
 如果“目的地”=7, 那么选择货物造型为“中山”;
 如果“目的地”=8, 那么选择货物造型为“东莞”。

(3) 分拣机器人接收运送带送来的货物, 将变量“接货”赋值为0。

(4) 分拣机器人运送货物离开, 切换到场景b。

场景b: 自动分拣中心(参考场景效果如图4-3所示)



图4-3 自动分拣场景

(5) 分拣机器人根据变量“目的地”的值选择运送货物的路径(目的地出口):

请你把这里的多重选择结构算法补充完整。

(6) 切换到场景c。

探究练习

用多重条件选择结构流程图来描述场景b的算法。

场景c: 出仓发货 (参考场景效果如图4-4所示)

(7) 将变量“发车”赋值为0, 运输车等待。

(8) 分拣机器人运载货物到投送口, 把货物投送给配送运输车, 机器人原地等待。

(9) 货物进入车厢后, 将变量“发车”赋值为1。

(10) 当“发车”=1时, 运输车出发; 当运输车驶出场景时, 变量“发货”赋值为1。

(11) 当“发货”=1时, 分拣机器人隐藏, 切换到场景a, 变量“发车”赋值为0。



图4-4 出仓发货场景

3. 编写程序

根据上面的算法用Scratch编写程序。

4. 调试程序

运行程序，观看运行效果，对程序存在的问题进行相应的修改和进一步完善。

二、变量及其赋值

变量是程序用来存放数据的，在编写程序时，少不了要使用变量。在本课的程序中，根据需要我们设置了“目的地”“接货”“发车”和“发货”等多个变量。

在程序处理数据时，通常把输入的数据、参加运算的数据、运行结果等临时数据暂时存储在计算机的内存中。变量就是被命名的内存单元位置，我们可以把它比喻为一个存放数据的“盒子”。当我们在程序中需要使用这个“盒子”中的数据时，只要用该变量名指代就行了。

那么，使用Scratch编程时，怎样构建变量以及往变量中存放数据呢？

参照图4-5的步骤我们可以新建变量，使用图4-6中的指令可以给变量赋值。



图4-5 新建变量



图4-6 给变量赋值

三、多重条件选择结构

在Scratch中，对于多种情况的选择问题，可以用多个条件选择语句来解决，本程序就使用了8个“如果…那么…”来处理8个目的地货物的分拣问题。

小组讨论

小组交流探讨用



来编写多重条件选择结构程序的区别。

反思评价

对自己在本课中的学习进行评价，对照表4-1，在相应的○里画√，在“其他收获”栏里留言。

表4-1 学习评价表

序号	学习内容	能否掌握
1	学会创建变量及其赋值。	能○ 不能○
2	了解程序的多重条件选择结构。	能○ 不能○
3	能够用多重条件选择结构流程图描述算法。	能○ 不能○
4	能够编写简单的多重条件选择结构程序。	能○ 不能○
其他收获：		

第五课 循环结构的程序设计



学习目标

- 认识三种类型的循环结构。
- 学会用循环结构流程图描述算法。
- 能够编写循环结构程序。

知识导图



不管是顺序结构，还是选择结构，程序的语句都只执行一次。如果要重复处理一些问题，则程序中需要编写很多重复的语句。在实际应用中，处理规律性重复问题时，使用循环结构语句，可以避免语句的重复编写。

本节课，我们将通过编写模拟分拣机器人自动返回充电的程序，来学习程序的循环结构。

一、分拣机器人

在智能无人仓中，分拣机器人负责成千上万个货物的分拣和搬运等工作。分拣机器人处于工作状态时，如果检测到电量低于10%，会暂停当前工作自动搜寻充电装置并前往充电，实现自主回充的功能。

学习任务

请根据以上说明编写程序，模拟实现分拣机器人自动回充的功能。

1. 问题分析

分拣机器人工作时，其电量随着时间推进会逐渐减少。当电量低于10%时，机器人会暂停工作状态并自动发出电量不足的信号；接收到电量不足的信号后，机器人则自动搜寻充电装置前往充电。

2. 设计算法

第一步：处于工作状态，初始化电量；

第二步：电量逐渐减少，直到电量为0或回到充电装置后，电量不再减少；

第三步：在指定范围内工作，直到电量低于10；

第四步：当电量低于10时，发出电量不足的信号；

第五步：当接收到电量不足的信号后，移动到充电装置开始充电；

第六步：开始充电，电量逐渐增加直到100。

算法设计如图5-1至图5-3所示。

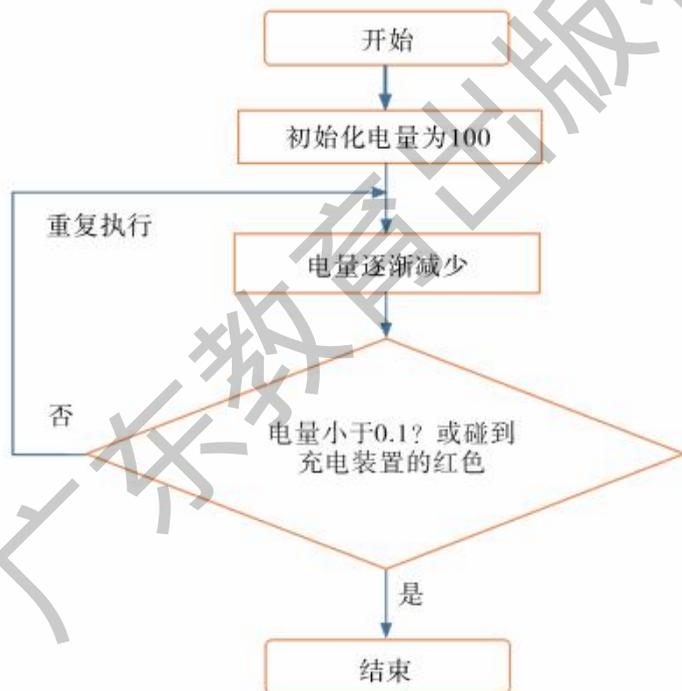


图5-1 电量逐渐减少的算法设计流程图

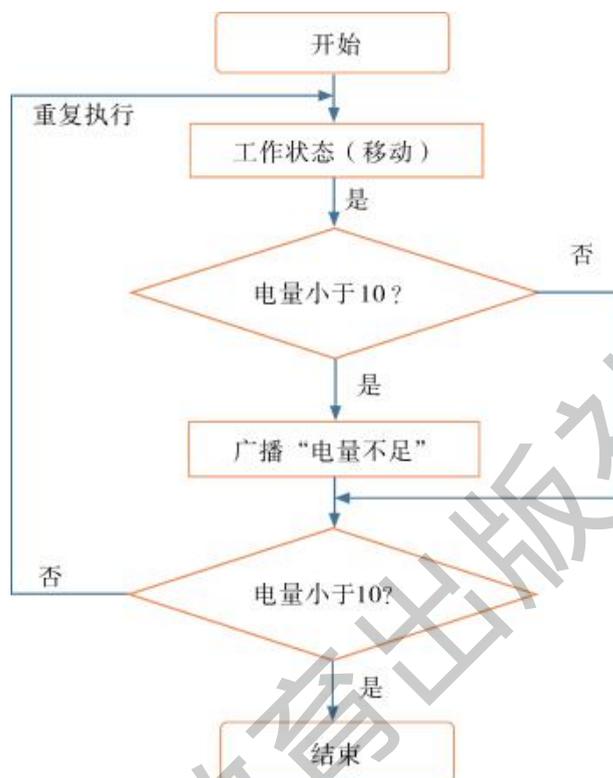


图5-2 工作状态的算法设计流程图

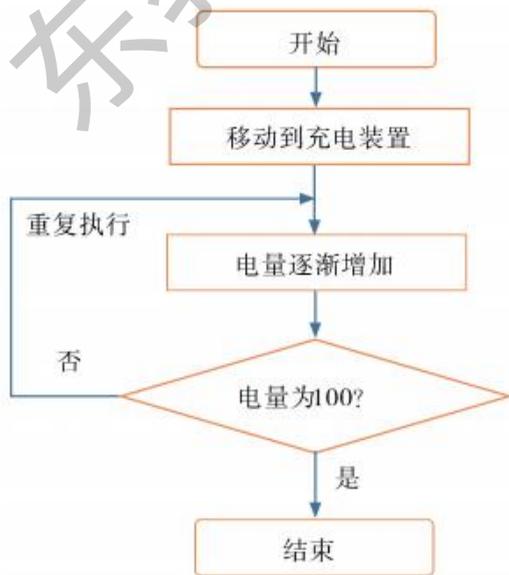


图5-3 电量逐渐增加的算法设计流程图

3. 设计程序

根据算法，设计分拣机器人角色的程序代码如图5-4至图5-6所示。



图5-4 电量逐渐减少的程序代码



图5-5 工作状态的程序代码



图5-6 电量逐渐增加的程序代码

4. 程序运行

单击“绿旗”运行程序，运行效果如图5-7所示。



图5-7 程序运行效果

运行结果显示，分拣机器人处于工作状态（移动），直到电量低于10%时，暂停工作状态并发出电量不足的信号；接收到电量不足信号后，机器人自动搜寻充电装置前往充电。

二、循环语句

上述分拣机器人的程序中运用到的程序代码（如图5-8所示）被称为“循环语句”。



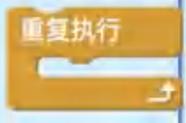
图5-8 循环语句

三、程序的循环结构

在处理重复性任务时，计算机突显出了自己的优势。循环结构让计算机重复地执行一条或多条指令代码。在前面章节的学习过程中，我们已经使用过“重复执行”循环结构，此外还有“重复执行…次”和“重复执行直到…”两种循环结构。

Scratch中循环结构的三种指令代码，如表5-1所示。

表5-1 三种循环结构的指令代码

Scratch指令代码	流程图	代码解释
		<p>“重复执行”是指重复执行内层指令代码，执行无限次。</p>
		<p>“重复执行…次”是计数循环结构，属于确定型循环，重复执行指定次数（如10次）内层指令代码。</p>
		<p>“重复执行直到‘条件’”是条件式循环结构，属于不确定型循环。直到“条件”成立时，跳出循环执行循环体外的指令代码，否则重复执行循环体指令代码。</p>

自主探究

某商场设有商品售卖机器人，顾客可通过机器人自助购买商品。在自助购买时，顾客输入的数据需要经过有效性验证。

假设某自助商品售卖机器人有10种商品，顾客通过输入1~10之间的数字来实现商品的选择，正确输入后，机器人提示“您选择的是：第XX号商品。”如果用户输入其他数字或字符，机器人将判断为无效输入，则显示“输入无效，请重新输入！”错误提示。

(1) 问题分析。

(2) 算法描述。

(3) 程序设计。

(4) 调试运行。

巩固练习

(1) 编写一个密码验证的程序，询问用户密码，密码正确则通过验证；如果密码错误，显示提示并要求重新输入，如果用户连续三次输错密码，将拒绝访问。

(2) 编写一个求最大值和最小值的程序。

反思评价

对自己在本课中的学习进行评价，对照表5-2，在相应的○里画√，在“其他收获”栏里留言。

表5-2 学习评价表

序号	学习内容	能否掌握
1	认识三种类型的循环结构。	能○ 不能○
2	学会用循环结构流程图描述算法。	能○ 不能○
3	能够编写循环结构程序。	能○ 不能○
其他收获：		

广东教育出版社

第六课 嵌套循环程序的实现



学习目标

- 认识嵌套循环的程序设计。
- 学会用嵌套循环结构流程图描述算法。
- 能够编写简单的嵌套循环结构程序。

知识导图



在Scratch控制指令中有如图6-1所示的三种循环结构语句，让计算机重复地执行一条或多条指令代码。



图6-1 三种循环结构语句

通过使用以上语句，已经能够处理很多重复性工作，极大地提高效率。然而在实际应用中，有些问题需要在循环体中再使用循环语句形成嵌套循环，才能更好地解决问题。

本节课，我们将通过Scratch编写模拟无人超市自助售货的程序，来深入认识嵌套循环结构。

一、无人超市

24小时营业，没有一个店员，刷脸进店，随意挑选商品，不管购买多少商品，无需排长队，离开超市的过程中同时刷脸结账，自动按价格由高至低显示所购商品及价格清单……在这种充满“黑科技”的无人超市里完

成一次购物体验，是不是感觉很新奇，很高科技？

学习任务

请根据以上描述，编写一个模拟无人超市自助售货的程序。

1. 问题分析

在无人超市中，顾客拿起货物放进购物车，系统自动识别货物并计算价钱。顾客带着货物离开超市，顾客手机自动付款并按价格高低显示商品的消费数量和金额。

2. 设计算法

第一步：初始化场景及商品。添加室内、室外两个场景，并对应添加两个人物角色、购物车及若干商品。

第二步：选购商品。当鼠标碰到商品时，商品自动移动到人物角色手中，随后移动到购物车里。

第三步：商品结算。采取累加的方式计算购物车中商品的数量及价格，当有商品进入到购物车时，就将商品数量增加1，同时将商品总价格加上当前商品的价格，一直循环重复。当离开超市时，商品数量和总价格的值就是加上最后一个商品数量及价格之后的值，即为我们要求的商品总数量和总价格。

计算商品总数量及价格的累加过程：

(1) 将商品数量设定为0，商品总价格设定为0；

(2) 第1个商品进入购物车时，商品数量=(原)商品数量+1，商品总价格=(原)商品总价格+商品1的价格；

(3) 第2个商品进入购物车时，商品数量=(原)商品数量+1，商品总价格=(原)商品总价格+商品2的价格；

(4) 第3个商品进入购物车时，_____；

(5) ……

(6) 第n个商品进入购物车时，_____。

第四步：离开室内场景。当鼠标碰到人物时，表示选购结束，人物角色展示动画效果。程序运行之后，人物一直在等待鼠标的碰触，这里需要

用到“重复执行”循环指令，这是外层循环。而人物本身动画效果的实现，采取不断变换造型的方法，这是内层循环。

算法设计流程图如图6-2所示。

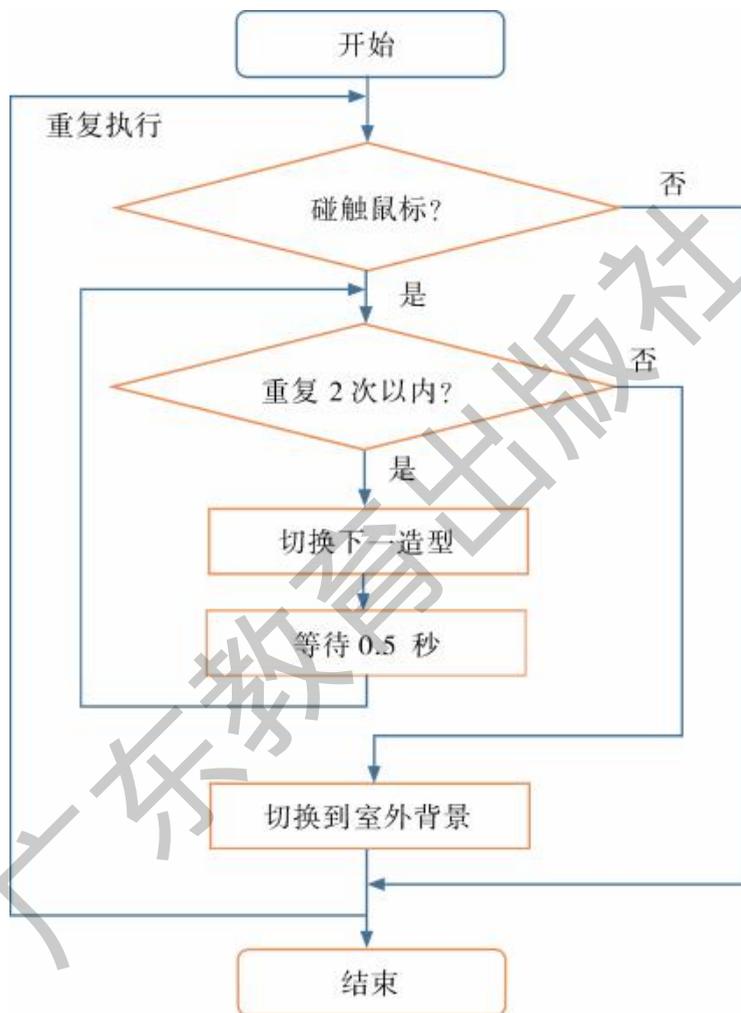


图6-2 人物离开室内场景算法设计流程图

第五步：购买结束，离开室内超市，切换到室外场景，显示购买商品数量及总金额。

3. 设计程序

根据上面的分析，编写程序如下：

室内人物角色的指令代码，如图6-3所示。



图6-3 室内人物角色的指令代码

商品帽子角色的指令代码如图6-4所示，其他商品角色类似。



图6-4 商品角色的指令代码

室外人物角色的指令代码，如图6-5所示。



图6-5 室外人物角色的指令代码

4. 程序运行结果



图6-6 程序运行效果

二、程序的嵌套循环结构

在上例无人超市的程序中，图6-7的指令代码就是嵌套循环结构的指令。



图6-7 嵌套循环指令代码

所谓嵌套循环结构，就是一个循环结构里嵌套着另一个（或多个）循环结构。如图6-7的嵌套循环结构的外层循环是一个“重复执行”循环结构，里面嵌套着一个计数型循环结构作为内层循环。内层循环结构实现人物角色的不断变化，外层循环结构实现侦测人物被鼠标碰触的功能。循环的嵌套编程技术非常重要，它可以用来解决大量的编程问题。

例如，如何编写程序实现如图6-8所示的绘画效果？通过分析，我们发现图形由一个12边形和12个正方形组成。12边形可以通过如图6-9所示的指令代码实现：

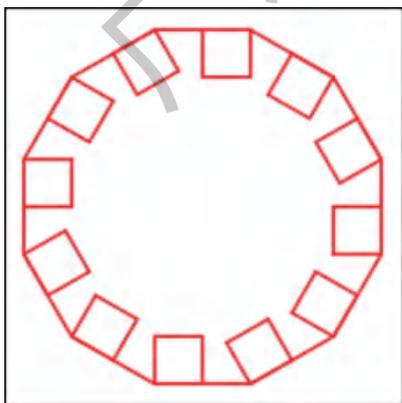


图6-8 用嵌套循环结构绘图



图6-9 12边形指令代码

一个正方形可以通过如图6-10所示的指令代码实现。



图6-10 正方形指令代码

通过将以上两个循环指令代码进行组合，形成嵌套循环结构，将绘制里面正方形的循环指令作为内层循环，将绘制外面12边形的指令作为外层循环，程序如图6-11所示。



图6-11 绘画的程序指令代码

自主探究

编写一个自动绘制如图6-12所示效果图形的程序。（提示：每个格子的边长为60步长）

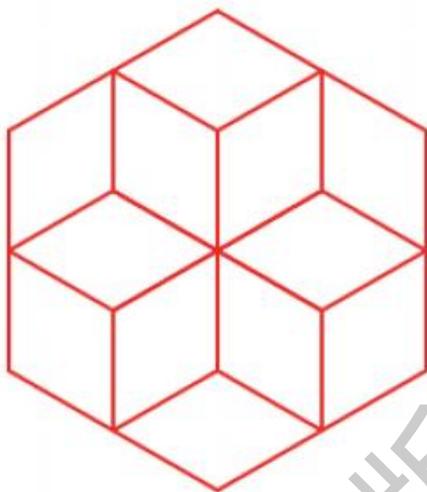


图6-12 效果图

(1) 问题分析。

(2) 算法描述。

(3) 程序设计。

(4) 调试运行。

巩固练习

(1) 编写一个绘制旋转的正方形的程序，运行效果如图6-13所示。

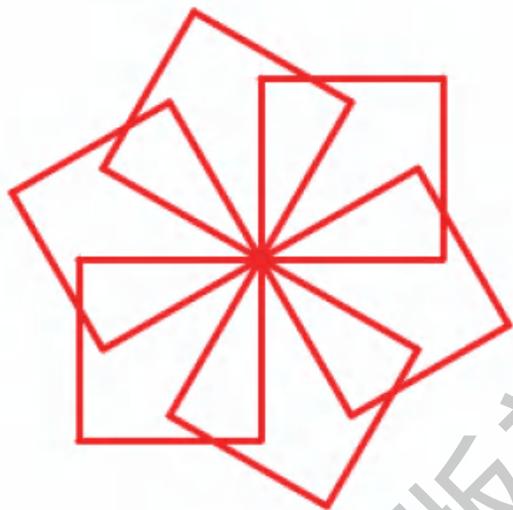


图6-13 旋转的正方形运行效果图

(2) 尝试结合链表的知识, 完善本课无人超市的程序, 实现顾客购买结束时将所购买商品的价格从高到低进行排序。



反思评价

对自己在本课中的学习进行评价, 对照表6-1, 在相应的○里画√, 在“其他收获”栏里留言。

表6-1 学习评价表

序号	学习内容	能否掌握
1	认识嵌套循环的程序设计。	能○ 不能○
2	学会用嵌套循环结构流程图描述算法。	能○ 不能○
3	能够编写简单的嵌套循环结构程序。	能○ 不能○
其他收获:		

第七课 结构化程序设计



学习目标

- 了解结构化程序设计的基本思想。
- 学会用模块结构图划分模块。
- 能够用模块化程序设计方法编写程序。

知识导图



在社会活动中，人们在对一些大型的、复杂的问题进行求解时，常常会把它们分解为更小、更简单和更容易处理的子问题。当这些子问题被求解以后，原来的问题就可以随着这些子问题的解决迎刃而解。这就是“自顶而下、逐步求精”的模块化程序设计思想。本课我们通过一个实例来学习模块化程序设计方法。

智能停车场为了提高运作效率、减少人员管理，采用了停车场泊位管理系统。该管理系统能自动统计空闲车位并在出入口信息屏告示、车辆出入自动控制栏杆放行。

学习任务

编写“停车场泊位管理系统”程序，实现如图7-1所示的停车场管理功能。



图7-1 停车场泊位管理系统场景

1. 分析问题

如图7-1所示的停车场管理系统能自动统计停车场空闲车位数量，当没有空车位时告示牌显示“车位已满”，并拒绝车辆进入，直到出现空闲车位。该系统由车辆出库管理、车辆入库管理和车位统计组成。其中，车辆出库管理包括出库车辆与出口栏杆两个管理模块；车辆入库管理包括入库车辆与入口栏杆两个管理模块。我们将“停车场泊位管理系统”划分为如图7-2所示的出库车辆、入库车辆、出口栏杆、入口栏杆、车位统计五个相对独立而又有内部联系的管理模块，分别设计算法和编写子程序。

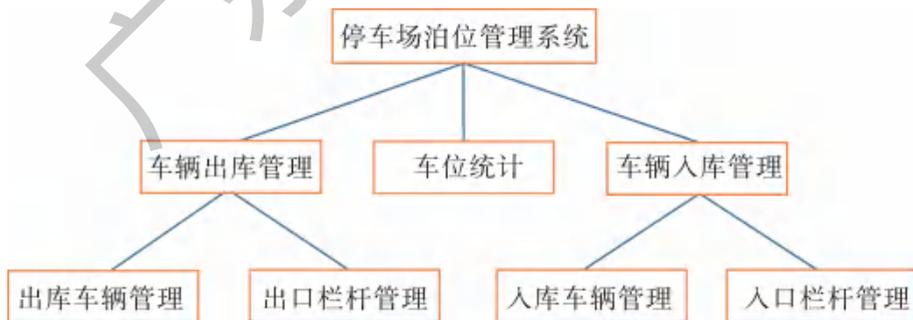


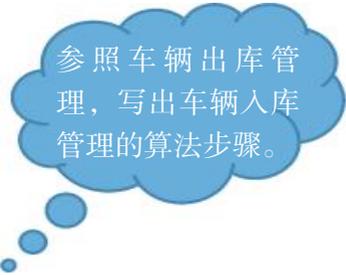
图7-2 模块结构图

2. 设计算法

车辆出库管理：

第一步：如果空车位小于100，执行下面步骤，否则重复本步骤；

- 第二步：出库车辆来到出口栏杆前停下等待；
 - 第三步：出口栏杆打开；
 - 第四步：车辆开走；
 - 第五步：出口栏杆关闭，空车位加1，返回第一步。
- 车辆入库管理：



车位统计：

- 第一步：初始化车位；
- 第二步：如果空车位=0，则告示“车位已满”，拒绝车辆入库，否则隐藏“车位已满”；
- 第三步：返回第二步。

出库车、出口栏杆、车位统计的算法流程图如图7-3所示。

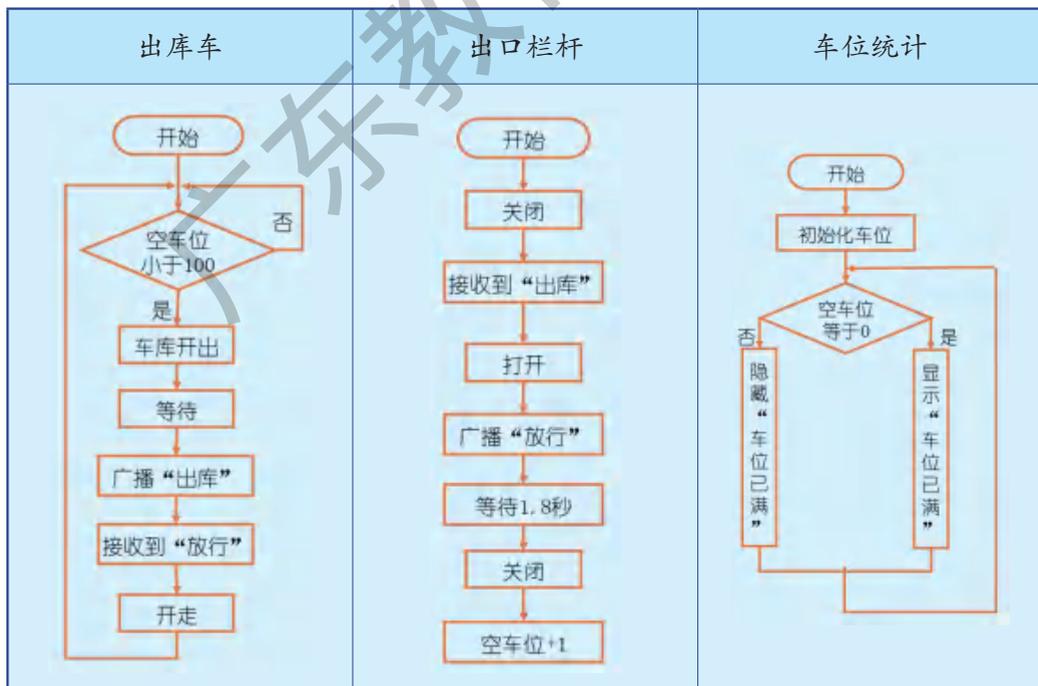


图7-3 算法流程图一

 自主探究

在图7-4中补充入口栏杆、入库车的算法流程图。

入口栏杆	入库车

图7-4 算法流程图二

3. 编写程序

根据上面的算法用Scratch编程，得到出库车、出口栏杆和车位统计模块的程序代码如图7-5所示。

出库车	出口栏杆	车位统计

图7-5 程序代码一

在图7-6中编写入口栏杆和入库车的程序代码。

入口栏杆	入库车
Empty space for coding	Empty space for coding

图7-6 程序代码二

4. 调试程序

运行程序，观看运行效果，对程序存在的问题进行相应的修改和进一步完善。

通过上述任务的求解过程，我们得到如下启示：

在用计算机解决一个复杂的问题时，不要急于用计算机指令、数字和逻辑符号来表示问题，应该先从总体角度进行分析，将其分解成几个有机的组成部分，如果某些部分还是比较复杂可以进一步分解。经过逐步分解和细化，将一个大而复杂的问题，由总体到局部，逐步分解为若干小的、可解的基本问题，再通过求解这些基本问题最终求得原问题的解。



反思评价

对自己在本课中的学习进行评价，对照表7-1，在相应的○里画√，在“其他收获”栏里留言。

表7-1 学习评价表

序号	学习内容	能否掌握
1	了解结构化程序设计的基本思想。	能○ 不能○
2	学会用模块结构图划分模块。	能○ 不能○
3	能够用模块化程序设计方法编写程序。	能○ 不能○
其他收获：		

第八课 综合活动：无人驾驶汽车 自控系统



学习目标

- 能够根据任务需求，按照结构化程序设计思想和方法进行程序设计。
- 学会运用算法和程序设计知识来解决问题。

知识导图



在第三课中，我们用条件选择指令编写程序，实现了无人驾驶汽车避让前方车辆的功能。在现实道路行驶中，无人车面对的状况要复杂很多，它不但要实时识别和应对交通灯信号、前方近距离低速障碍物等，还要具备紧急避让行人闯红灯或横穿马路的能力。本课我们对城市道路交通场景进行综合分析描绘，应用前面学习的程序设计知识，编写程序模拟无人驾驶汽车在城市道路中自动驾驶。

一、活动任务

用Scratch编程工具编写程序，模拟无人驾驶汽车在城市道路中自动驾驶的动画场景。

任务描述：模拟无人汽车在城市道路上自控行驶的状况。程序作品要求实现：作品运行时的交通状况随机呈现，自动驾驶汽车不但能够实时识别和应对交通灯信号、前方近距离低速障碍物等，还具备紧急避让行人闯红灯或横穿马路的能力。

二、活动过程

1. 分析问题

(1) 收集道路交通状况的相关资料,按照“自顶而下,逐步求精”的程序设计思想,对与自动驾驶相关的情景问题从整体的角度进行分析,将其分解为若干单独场景,用模块结构图描述它们之间的联系,如图8-1所示。

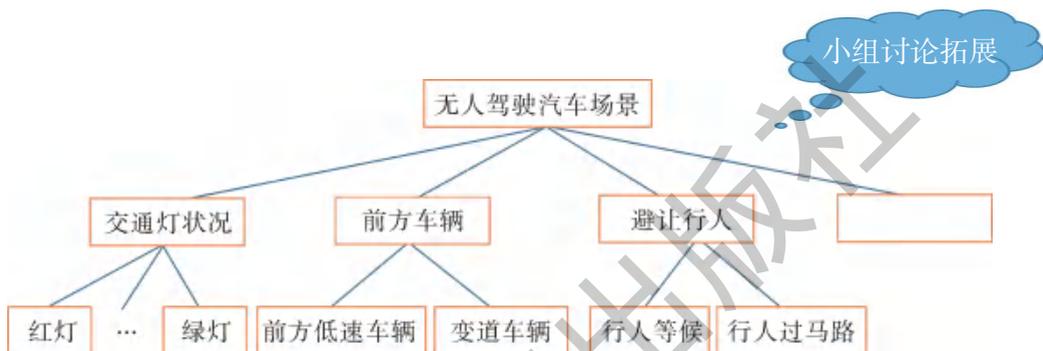


图8-1 “无人驾驶汽车自控系统”模块结构图

现实道路交通中还有哪些状况会出现?小组展开讨论,并在图8-1的空白处补充扩展。

(2) 分析作品涉及的角色,确定角色可能发生的状态、自动驾驶汽车的应对策略,补充在表8-1中。

表8-1 分析角色可能发生的状态

角色	可发生状态	无人驾驶汽车应对策略
交通灯	红灯/绿灯/箭头指示绿灯	红灯停绿灯走,按箭头指示灯行走
前方车辆	接近	当与前车距离过近时,减速避让
变道车辆	前方旁道车辆变线并道	减速,保持安全距离行驶
行人	路边等候/横过马路	停车让行

2. 算法设计

根据以上角色任务分析，设计算法，参考流程图如图8-2、图8-3所示。

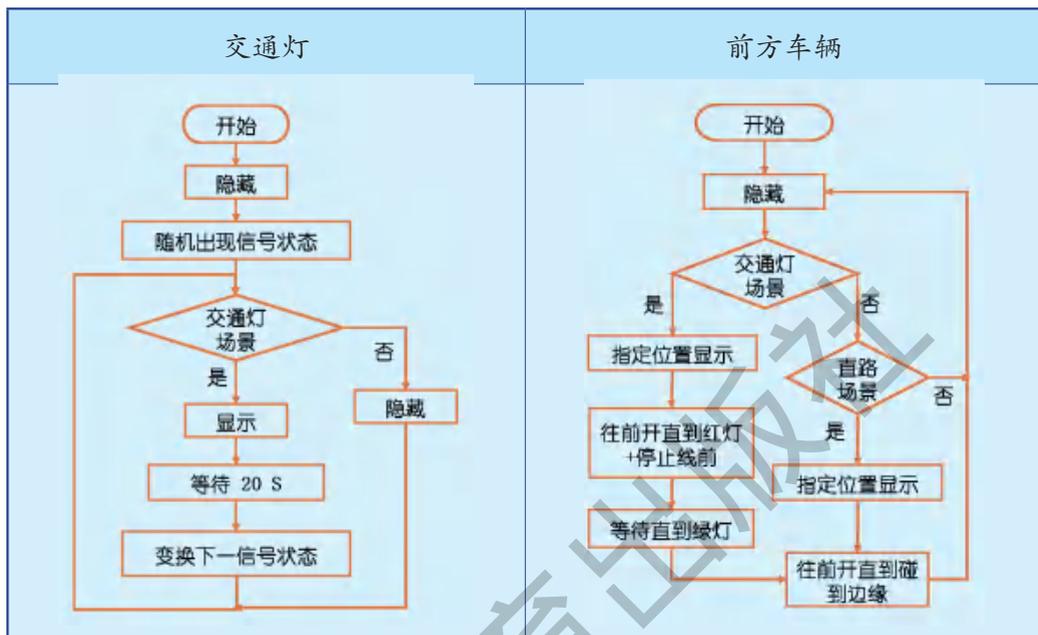


图8-2 算法流程图一

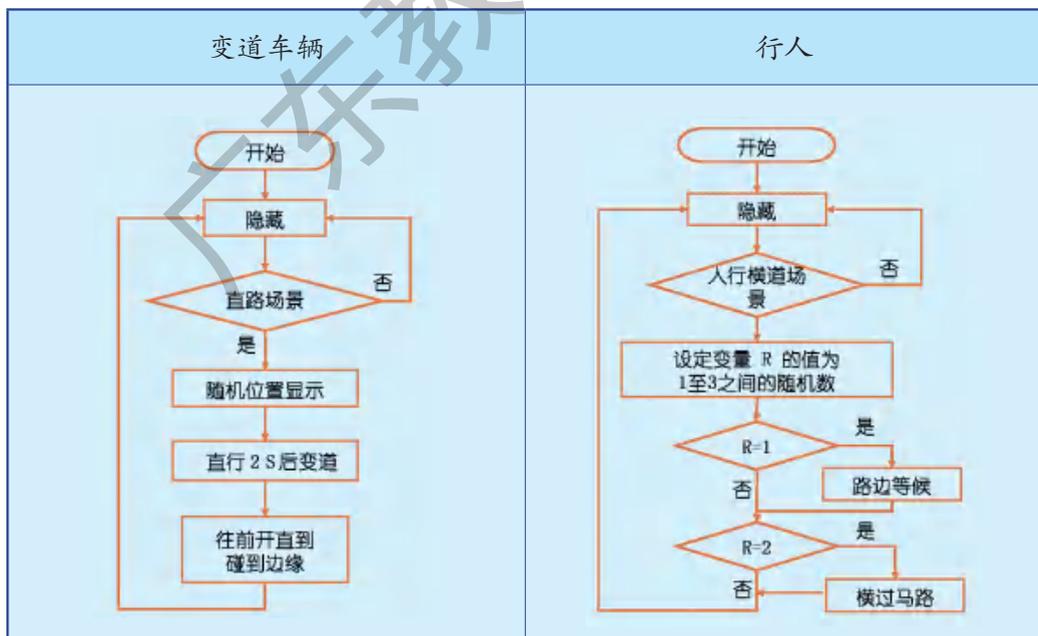


图8-3 算法流程图二

自主探究

在图8-4中完成无人驾驶汽车和其他扩展角色的算法流程图。

无人驾驶汽车	扩展角色

图8-4 算法流程图三

3. 编写程序

根据算法描述，编写程序代码，完成表8-2的填写。

表8-2 各角色的算法流程图及代码

角色	算法流程图	程序代码
交通灯	<pre> graph TD Start([开始]) --> Hide[隐藏] Hide --> Random[随机出现信号状态] Random --> Decision{交通灯场景} Decision -- 是 --> Display[显示] Display --> Wait[等待 20 S] Wait --> Change[变换下一信号状态] Change --> Decision Decision -- 否 --> Hide2[隐藏] Hide2 --> Decision </pre>	
前方车辆	<pre> graph TD Start([开始]) --> Hide[隐藏] Hide --> Decision1{交通灯场景} Decision1 -- 是 --> Display1[指定位置显示] Display1 --> Drive1[往前开直到红灯+停止线前] Drive1 --> Wait[等待直到绿灯] Wait --> Drive2[往前开直到碰到边缘] Decision1 -- 否 --> Decision2{直路场景} Decision2 -- 是 --> Display2[指定位置显示] Display2 --> Drive3[往前开直到碰到边缘] Decision2 -- 否 --> Hide2[隐藏] Hide2 --> Decision1 Drive2 --> Hide2 </pre>	

(续表)

角色	算法流程图	程序代码
变道车辆	<pre> graph TD Start([开始]) --> Hide[隐藏] Hide --> Decision1{直路场景} Decision1 -- 否 --> Hide Decision1 -- 是 --> Display[随机位置显示] Display --> Action1[直行 2 S后变道] Action1 --> Action2[往前开直到碰到边缘] Action2 --> Hide </pre>	
行人	<pre> graph TD Start([开始]) --> Hide[隐藏] Hide --> Decision1{人行横道场景} Decision1 -- 否 --> Hide Decision1 -- 是 --> SetR[设定变量 R 的值为 1至3之间的随机数] SetR --> Decision2{R=1} Decision2 -- 是 --> Action1[路边等候] Action1 --> Decision3{R=2} Decision3 -- 是 --> Action2[横过马路] Action2 --> Hide Decision2 -- 否 --> Decision3 Decision3 -- 否 --> Hide </pre>	

(续表)

角色	算法流程图	程序代码
无人驾驶汽车		
扩展角色		

4. 调试运行

运行程序，观看运行效果，对程序存在的问题进行相应的修改和进一步完善。

展示交流

- (1) 以小组为单位，展示程序作品并介绍作品的内容、技术路线。
- (2) 观摩、评价、交流。

反思评价

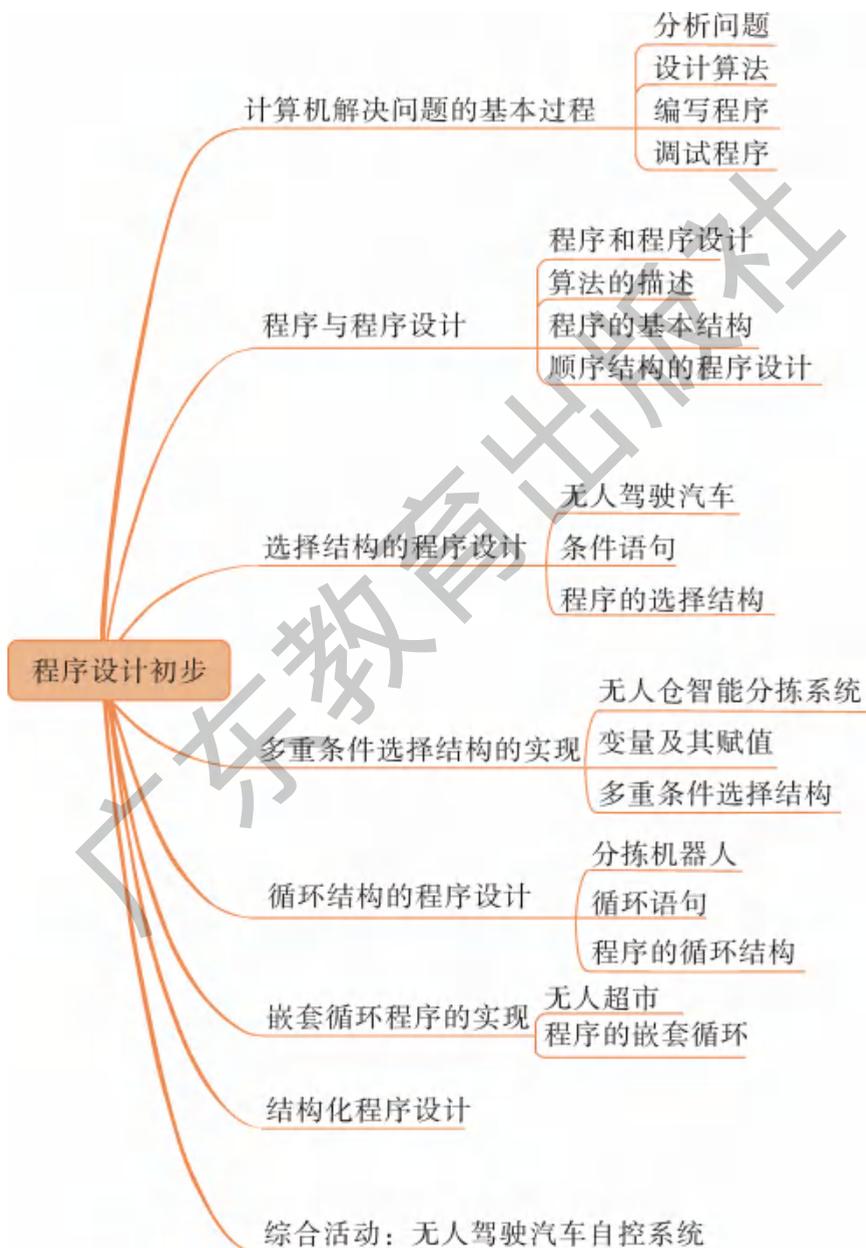
对自己在本课中的学习进行评价，对照表8-3，在相应的○里画√，在“其他收获”栏里留言。

表8-3 学习评价表

序号	学习内容	能否掌握
1	学会用模块化程序设计思想分解任务。	能○ 不能○
2	掌握程序的编写和调试的方法。	能○ 不能○
3	能够用模块化程序设计方法编写程序。	能○ 不能○
其他收获：		

广东教育出版社

单元扼要回顾



单 元 学 习 评 价

一、学习评价

请根据本单元各环节的学习情况，填写单元学习评价表，从知识与技能、方法与过程以及情感态度与价值观等方面，综合评价自己是否达到了本单元的学习目标。

单元学习评价表

学习目标		是否达到目标	
		是	否
知识与技能	了解计算机解决问题的基本过程。		
	了解算法、编写程序和调试运行程序的概念。		
	了解程序顺序结构、选择结构、循环结构的执行流程，能够用流程图描述算法，编写简单的程序。		
	了解程序的多重条件选择结构，认识嵌套循环的程序设计。		
	学会用模块结构图划分模块，能够用模块化程序设计方法编写程序。		
	学会运用算法和程序设计知识来解决实际问题。		
过程与方法	在学习过程中，能根据本章的实践、观察、练习等环节自主学习、动手操作，开动脑筋积极思考。		
	能与同学交流，勇于尝试不同的操作方法，总结归纳不同操作方法的特点，从而构建自己的知识体系。		

(续表)

学习目标		是否达到目标	
		是	否
情感态度与价值观	在本单元的探究、讨论环节能积极主动地参与小组合作学习，勇于表达自己的想法。		
	当其他同学碰到学习困难时，能主动帮助他们，在分组探究的过程中，形成小组合作意识。		

二、学习体会

请从以下几方面谈谈自己在本单元学习过程中的成长体会。

本单元对我启发最大的是：_____

我不太理解的学习内容有：_____

我还学会了：_____

我还想学习：_____

第二单元

开源硬件设计

知识导图

开源硬件设计

认识开源硬件

软件与硬件的结合使用

数字与模拟信号

使用传感器采集信息

控制输出模块工作

电动机和舵机

数码显示与无线通讯的实现

综合活动



开源硬件与开源软件中所指的“开源”，是计算机科学领域的一种特殊的文化现象，是人类对智慧成果共享、自由的追求。所谓开源软件是指其源代码可以被公众使用的软件，并且此软件的使用、修改和分发也不受许可证的限制。而开源硬件是指使用与开源软件相同的方式设计的计算机和电子硬件。开源硬件设计者通常会公布详细的硬件设计资料，如机械图、电路图、物料清单、PCB版图、HDL源码和IC版图，以及驱动开源硬件的软件开发工具包等。

由于开源硬件从硬件设计到驱动以及软件开发工具包均是开放授权并可免费获得的，除生产、销售过程中可能产生的物料、人工、仓储等成本外，其他成本接近于零。因此，开源硬件特别能满足青少年探究客观世界的好奇心，激发创作的欲望和兴趣，培养动脑动手的能力。

在本单元中，我们将介绍常见的智能控制器、传感器等开源设备，通过项目创作活动，引领大家自主、探究和协作学习，共同完成项目任务。

第九课 认识开源硬件



学习目标

- 知道什么是开源硬件。
- 了解开源硬件的基本用途。
- 能够让Arduino板与计算机正确连接。
- 能实现对Arduino板的简单控制。

知识导图



在我们的生活中，各种各样的用电器给我们带来了便利，而计算机则改变了我们的生活和工作方式。随着科技的发展，未来将会有更多的电子设备改变我们的生活，而开源硬件可以让每个人根据自己的想法设计电子设备。

一、开源硬件

开源硬件是指使用与开源软件相同的方式设计的计算机和电子硬件。开源硬件不仅设计图纸是公开的，而且还可以让人们对其进行修改开发，从而使得电子设备的设计变得简单。由于开源硬件采用模块化设计，因此我们不需要对复杂电子元件进行组装，而只需要关注各种模块的功能，像搭积木一样将这些硬件模块连接在一起即可使用。

开源硬件能够让每个人按照自己的想法设计出各种电子设备，比如根据光线强弱控制的台灯，根据温度控制转速的风扇，还有会自动躲避障碍物的小车，等等。开源硬件的引入，使我们学习电子制作变得简易，硬件的标准化、模块化，技术资料的公开，都令其得到了十分广泛的使用。

开源硬件由于是开放的，所以人们可以在原有硬件的基础上对其进行改造，派生出许多新的硬件。最常见的开源硬件包括Arduino、树莓派和Micro:bit等。

1. Arduino

Arduino是一块电路控制板，可以连接其他电子模块，如图9-1所示。通过编写专门的程序，可以让Arduino控制这些电子模块工作，把我们的想法变成现实的电子产品。



图9-1 Arduino UNO 控制板

2. 树莓派

树莓派更像一台小型计算机，不仅有视频信号接口、USB接口、网络接口，而且如果将显示器和鼠标键盘连接到树莓派上，就可以像使用普通计算机一样使用各种软件，如图9-2所示。虽然具备较强的计算能力，但树莓派的体积仅一张银行卡大小，非常方便携带。

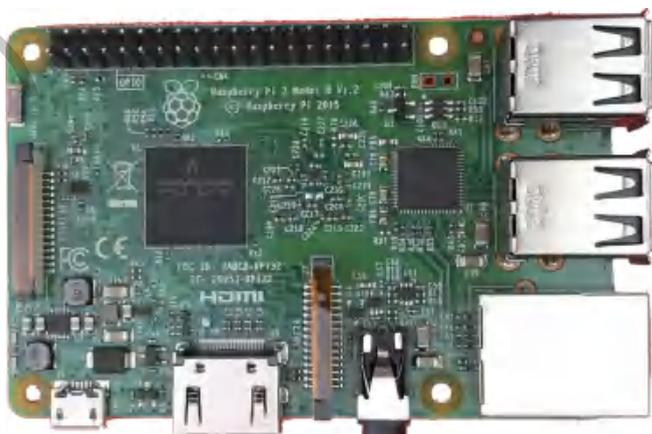


图9-2 树莓派

3. Micro:bit

Micro:bit是专门为青少年学习设计的控制板，适合用来学习电子制作与编程，由于控制板上集成了LED、按钮和几种传感器，所以不需要连接其他模块就能实现很多功能，非常适合用来学习硬件的设计。

二、认识Arduino

Arduino主要指的是控制板硬件，其有多种型号，最常见的也是我们将要使用的型号为Arduino UNO控制板。本单元全部使用Arduino UNO控制板作为主控硬件，所以后面提到Arduino板指的都是Arduino UNO控制板。如图9-3所示，Arduino板主要接口包括：连接电脑的USB接口；专门的电源输入接口；数字接口和模拟接口，主要用来连接其他硬件模块；等等。

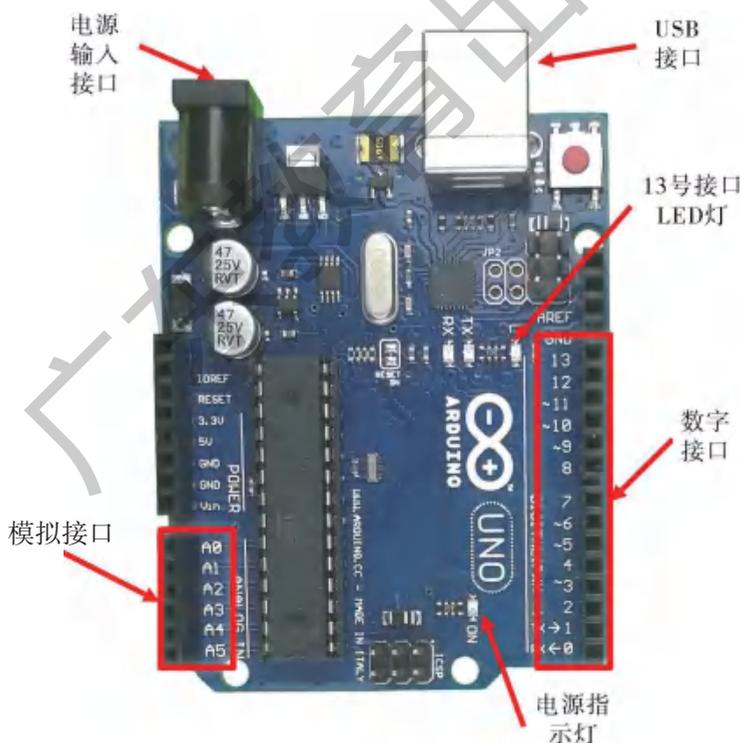


图9-3 Arduino UNO控制板的主要接口

三、计算机与Arduino的连接

通过USB数据线可以将Arduino板与计算机连接，再通过计算机中编写控制指令，然后把这些指令上传到Arduino板上即可让其按照我们的要求工作。

学习任务

在开始学习任务之前，我们需要准备如表9-1所示的硬件材料。

表9-1 学习任务使用硬件列表

硬件名称	Arduino板	USB数据线
硬件外观		
硬件功能	Arduino UNO 控制板，控制硬件工作的核心。	USB 打印数据线，用来连接电脑和Arduino板。
数量	1	1
备注	本单元每节都会用到的硬件，后面不再清单中列出。	

(1) 使用USB数据线把Arduino板与计算机连接，如果第一次连接计算机还需要安装驱动程序，否则连接后即可识别硬件，然后查看串口编号。

(2) 在设备管理器中查看“端口”，找到“Aruido UNO”设备，并查看括号中的串口编号，串口编号为“COM”字母后的数字。

(3) 你找到的串口编号是_____。

四、简单硬件控制体验

要Arduino板按照我们的要求工作，就需要编写相应的指令，在计算机中编写的这些指令，需要上传到Arduino板中，才能让其按照指令工作。可以编写这些指令的软件有很多种，我们选择Mixly（米思齐）软件。接下来我们完成第一个控制Arduino板工作的指令，打开Mixly软件，使用打开功能，打开文件“闪烁LED.xml”，可以看到如图9-4所示的代码块。



图9-4 闪烁LED程序

这些组合在一起的模块含义是让Arduino板上自带的13号引脚（又称管脚）LED灯闪烁，“高”表示亮，“低”表示灭，延时确定亮或灭持续的时间。而这段代码会不停循环执行，所以会产生LED灯不停闪烁的效果。

学习任务

让Arduino板上自带的13号接口LED灯有规律地闪烁。

1. 问题分析

灯的闪烁，就是控制LED灯的亮度变化，通过控制输出为“高”或“低”，相应的LED灯就会亮或不亮。由于这些拼接在一起的模块会从上而下一个一个执行，完成后会再从头开始，最后会不停地循环执行，所以只需要写一次灯的闪烁命令，就会不停闪烁。我们只需要把Arduino板与电脑用USB数据线连接即可体验LED灯闪烁的效果。

2. 设计算法

使用方框与箭头可以直观表示整个LED灯闪烁控制实现的过程。



图9-5 闪烁LED流程图

流程图中的方框表示执行一项工作，连接方框之间的箭头表示执行的顺序。如图9-5所示，第4个方框执行完成后，根据箭头指示，又会从头开始执行。

3. 编写程序

打开Mixly软件，单击“打开”按钮，找到文件“闪烁LED.xml”，然后可以看到程序模块，如图9-4所示。该模板文件中共有4个模块，这些模块按照顺序拼接在一起，4个模块会按顺序执行，并且不断循环。

4. 测试运行

使用USB数据线连接Arduino板与电脑的USB接口，单击“编译”按钮，检查程序是否有错误，确认无误后，选择正确的串口COM编号（通常为非COM1的编号），然后单击“上传”按钮，即可把程序指令发送给Arduino板。之后查看Arduino板上13号引脚的LED是否有闪烁。最后使用“另存为”功能，保存这个程序文件，这个文件储存的就是可以控制Arduino板工作的指令。

自主探究

学习任务中LED灯按照1秒钟的间隔闪烁，但是如何确定LED灯的闪烁就是由我们上传的指令控制的修改间隔时间，把延时模块中的1000修改为其他的数字，就可以改变间隔时间。为了直观，最好选择较大的时间间隔，但也要避免等待时间过长。改完之后，再次观察LED灯闪烁的频率，

是否按照你的指令工作。

在计算机中编写的程序指令，上传到Arduino板后，就可以按照指令工作。即使断开与计算机的USB数据线，只需要有电源可为其供电，Arduino板也可独立工作。如图9-6所示，将安装9 V充电电池的电池盒与Arduino板连接，Arduino板就按照存储的指令控制LED灯的闪烁，此时Arduino板就像一台计算机一样可以独立工作。



图9-6 Arduino板独立工作



反思评价

对自己在本课中的学习进行评价，对照表9-2，在相应的○里画√，在“其他收获”栏里留言。

表9-2 学习评价表

序号	学习内容	能否掌握
1	知道什么是开源硬件。	能○ 不能○
2	了解开源硬件的基本特征和用途。	能○ 不能○
3	会将Arduino板与计算机连接。	能○ 不能○
4	能实现对Arduino板的简单控制。	能○ 不能○
其他收获：		

第十课 软件与硬件的结合使用



学习目标

- 了解软件与硬件的基本关系。
- 了解Arduino板的基本使用方法。
- 会使用Mixly软件。
- 学会单色LED模块硬件的使用。
- 学会全彩LED模块硬件的使用。

知识导图



计算机在我们的生活中有广泛的应用，而开源硬件Arduino UNO控制板也属于计算机，只是与我们常见的计算机外观不同，但是基本结构都是一样的。计算机主要由运算器、控制器、存储器、输入设备、输出设备五大部分组成，其中运算器与控制器合称为中央处理器。计算机工作就是依靠程序控制这些硬件工作。

一、软件与硬件

硬件是组成计算机的各种物理设备，计算机工作不仅需要硬件，还需要软件。软件是计算机工作用到的程序与数据的集合，而程序就是控制计算机工作的指令的集合。计算机的硬件工作需要要有对应的程序，所以使用计算机为我们工作，必须编写程序（简称编程）。

对于开源硬件而言，常用的硬件种类比普通计算机使用的种类更多，编写控制这些硬件的程序前，需要熟悉硬件的基本工作方式，这样才能编写合适的指令。

Arduino板属于硬件，具体接口可参见图9-1。

由于Arduino板上的电源接口和地线接口较少，为了方便连接多个设

备，我们利用扩展板扩张引脚的数量。如图10-1所示为两种常见的扩展板，使用这些扩展板可以极大方便连接其他硬件模块，使得使用Arduino板就像搭积木一样简单。



图10-1 Arduino板的两种扩展板

如图10-2所示，以4、5、6、7号引脚为例，每个编号的引脚为3个一组，最上面一行黑色底座标注为G的引脚为接地接口，中间红色一行底座标注为V的引脚表示电源接口，最下面黄色一行底座标注为S的引脚为数据接口。以4号引脚为例，本单元我们将其分别命名为4G、4V、4S，分别表示接地、电源、4号引脚接口。黑色一行接地引脚与红色一行电源引脚，功能是一样的，只是为了提供更多接口方便使用。A0~A5的模拟接口也将类似标记，如A0接口一组引脚分别命名为A0G、A0V、A0S。

在连接各种硬件模块时，要正确连接电源和接地引脚，还要注意模块对应的引脚编号。在编写程序控制硬件工作时，主要是通过引脚的编号实现的，所以要留意对应的硬件连接Arduino板的哪个引脚。

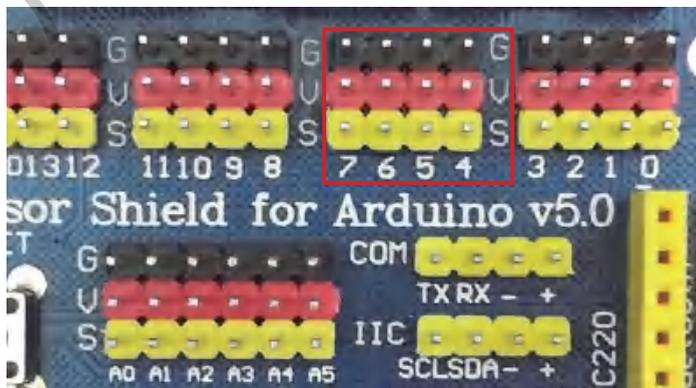


图10-2 Arduino Uno扩展板部分引脚结构

二、Mixly软件的使用

要使Arduino板按照我们的要求工作，就必须在计算机中编写程序并上传到Arduino板中。我们使用Mixly软件编写程序，其界面如图10-3所示。

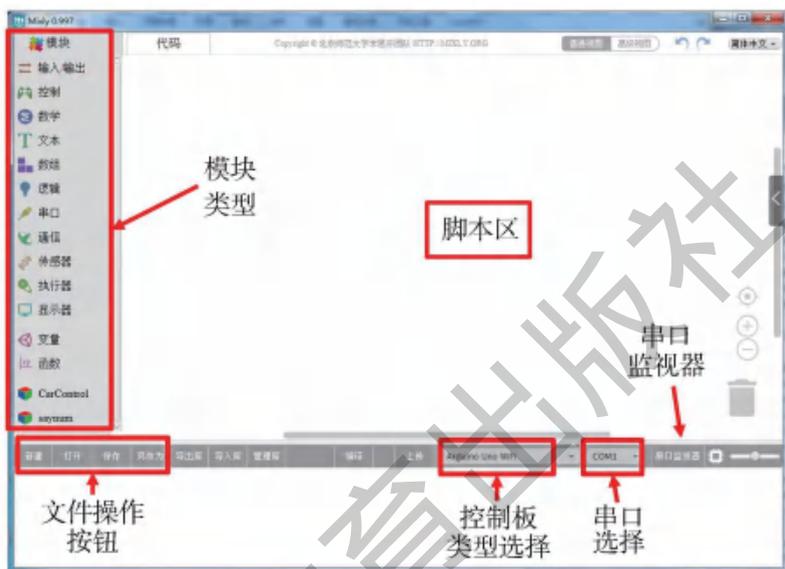


图10-3 Mixly软件界面

界面左边为模块类型区，单击左边各种类型模块的名称，右边就弹出一个“抽屉”，里面有该类型的全部模块，可以自由拖动到模块脚本区。拖动到脚本区的模块，可以按照需要进行拼接，也可使用右键进行复制和删除。

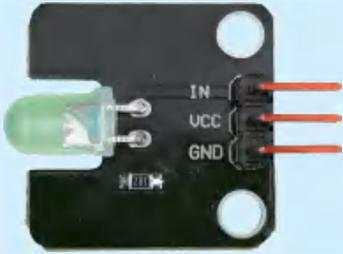
界面下面灰色底纹区域的按钮和下拉菜单提供了丰富的功能，除了包括新建、打开、保存等按钮，还有编译、上传、串口监视器按钮。两个下拉菜单可以分别选择Arduino板的类型名称，这里使用Arduino UNO类型即可。串口编号可以在使用USB数据线Arduino板连接后，再选择非COM1就可以建立。

三、LED模块的使用

LED是发光二极管的简称。发光二极管使用时，需要正确连接电源的

正负极。LED模块中有三个引脚，分别为GND引脚（有的模块缩写为G或“-”）为接地，VCC引脚（有的模块缩写为V）为电源正极，IN引脚（有的模块缩写为S）为信号输入，如表10-1所示。

表10-1 LED模块硬件接口功能表

LED模块	引脚	引脚功能
	IN	控制信号输入
	VCC	电源输入
	GND	接地

学习任务

控制车辆通行的交通信号灯，有红、黄、绿三种颜色，各种颜色的灯按照规则顺序亮或灭，以此控制车流。完成通过Arduino板连接3个LED模块，模拟路口的红绿灯工作过程。

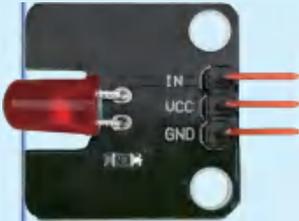
1. 问题分析

红绿灯的变化主要是三种颜色的灯依次亮，每个灯亮一定时间后就灭了，所以依次控制三个灯即可模拟红绿灯的工作。通常黄灯作为红绿灯的过度，亮的时间较短，所以设置红灯和绿灯亮的时间为2秒（2000毫秒），黄灯亮的时间为0.5秒（500毫秒），然后一直循环。

2. 硬件清单与连接。

本任务需要使用的硬件有的为每课必备的，如Arduino板与USB数据线不再每次详细说明，本课使用的其他硬件主要有扩展板、杜邦线和LED模块。扩展板选择彩色接口的便于连接；杜邦线用来连接各种硬件的引脚，颜色可以任意选择；LED模块选择红、黄、绿三种颜色各1个。具体硬件清单如表10-2所示。

表10-2 模拟红绿灯硬件清单表

硬件名称	扩展板	杜邦线	LED模块
硬件外观			
硬件功能	扩展Arduino板的接口数量，主要增加电源和接地引脚的数量。	硬件之间相互连接使用，本单元使用母对母20厘米的连接线。	可以发光的LED模块，本任务需要红色、黄色、绿色的LED模块共3个。
数量	1	9	3种颜色各1个
备注	本单元后面每课都会用到的硬件，将不再每次在清单中列出。		

由于每课的任务都需要连接Arduino板与扩展板，其他硬件模块都是与扩展板连接，所以全部硬件连接图，不再展示Arduino板，直接展示扩展板与硬件模块。连接3个LED模块与扩展板，每个模块使用3条杜邦线，可以将杜邦线分为3个一组，而且连接的时候，必须电源、接地和数据引脚必须对应。

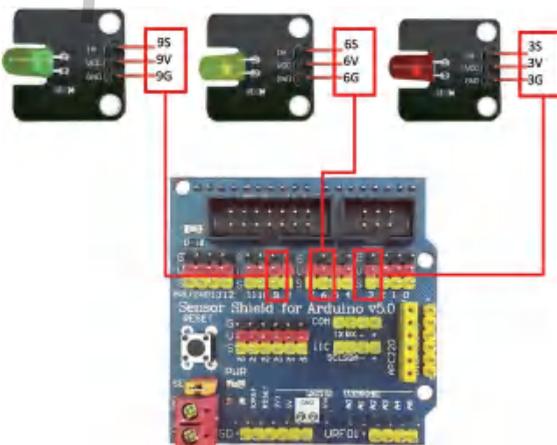


图10-4 模拟红绿灯硬件连接图

为了方便连接，使用扩展板增加Arduino板的引脚数量，3个LED模块分别连接在数字接口3、6、9引脚上（也可以选择其他引脚，由于0、1引脚用作与电脑实现串口通讯，所以不要使用），红色LED模块连接3号引脚，黄色LED模块连接6号引脚，绿色LED模块连接9号引脚，如图10-4所示。

3. 算法设计

依次控制3个LED模块的亮与灭，就是分别设置3个模块LED灯亮，延时后再设置LED灯灭。通常黄色灯亮的时间较短，所以红色与绿色LED亮2秒，而黄色LED亮0.5秒。模拟红绿灯的算法流程图如图10-5所示。



图10-5 模拟红绿灯流程图



图10-6 模拟红绿灯程序模块图

4. 编写程序

根据流程图使用“输入/输出”类型模块中的“数字输出”模块，设置为“高”，表示输出高电平，LED发光。如果设置为“低”，表示输出低电平，LED不发光。按照时间间隔，以此设置3个LED模块的亮灭，“延时”模块为“控制”类型模块，模拟绿红程序模块如图10-6所示。

5. 测试运行

通过USB数据线连接Arduino板，选择正确的COM端口号，然后单击“上传”按钮，即可把程序指令发送给Arduino板。最后检查3个LED灯是否按照交通信号灯的顺序依次亮灭。

四、全彩LED模块的使用

单色LED模块只能产生一种颜色光的亮度变化，全彩LED模块又称为三色LED模块。虽然与单色的LED模块很像，但是引脚数量却为4个，其内部有3个LED可分别发出红光、绿光、蓝光，并且可以通过红色、绿色、蓝色3种颜色光组合产生各种颜色光，其产生的颜色按照RGB颜色系统定义，如表10-3所示。

表10-3 全彩LED模块硬件接口功能表

全彩LED模块	引脚	引脚功能
	-	3个LED公共的接地，有的标注为GND
	R	红色LED正极
	G	绿色LED正极
	B	蓝色LED正极

知识链接

RGB颜色模型

全彩LED模块可以发出各种颜色的光，我们可以控制颜色和亮度。手机上广泛使用的呼吸灯，其实就是控制LED的亮度和颜色变化，以此给用户传递信息。全彩LED模块，可以分别设置每一种颜色，如果显示的颜色太多，所需的模块数量非常多，但使用循环可以减少模块的使用量。如图10-7所示，使用数字引脚3连接全彩LED模块红色引脚，输出从1至255的数值，以此表示LED的亮度变化，延时10毫秒，整个循环2550毫秒即可完

成一次。为了区分不同的LED颜色，将循环的名称改为“红”，再从“变量”模块中得到“红”变量给引脚3模拟输出赋值。



图10-7 循环模块的使用

学习任务

夜晚的街道上，可以看到很多霓虹灯，现在的霓虹灯很多都是用LED作为光源。我们可以使用全彩LED模块，产生霓虹灯变化的效果。

1. 问题分析

霓虹灯的颜色变化很多，而Arduino板输出的数值可以从1~255变化，因此可以通过循环的方式，让LED依次以此亮度发光，并且可以通过颜色的叠加展示出更加丰富的颜色。

2. 硬件连接

全彩LED模块除了3个颜色的3个引脚，还有1个公共引脚，该模块公共引脚为公共接地，可以接在任何一个数字接口的接地引脚上，我们选择3G引脚。

3个LED模块分别连接在接口3、6、9引脚上（也可以选择其他引脚，不可使用0、1引脚），红色LED模块连接3号引脚，绿色LED模块连接6号引脚，蓝色LED模块连接9号引脚，如图10-8所示。

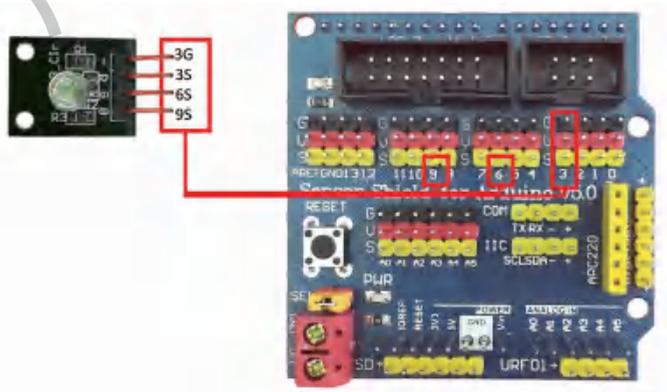


图10-8 模拟霓虹灯硬件连接图

3. 算法设计

模拟霓虹灯流程图如图10-9所示。

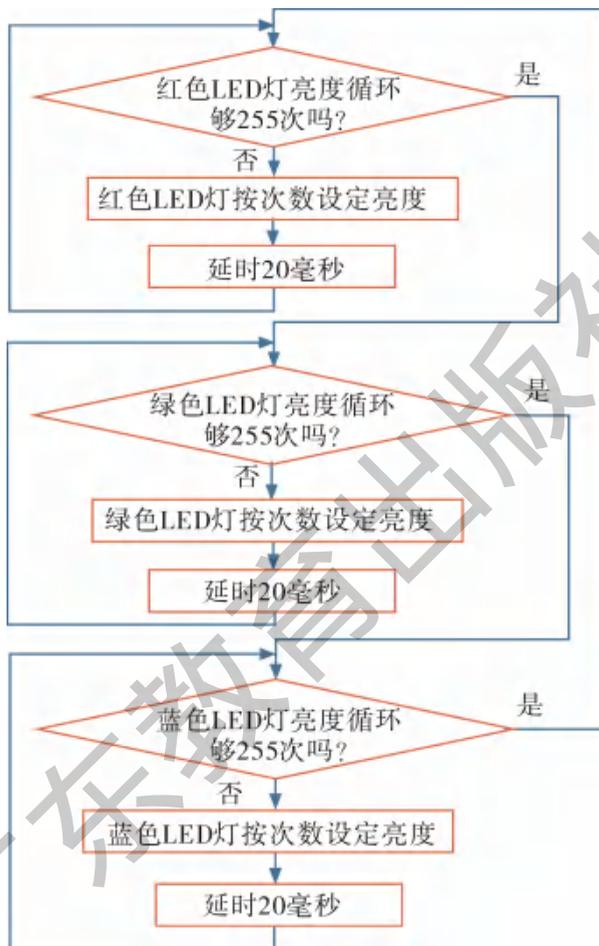


图10-9 模拟霓虹灯流程图

4. 编写程序

根据流程图使用“输入/输出”类型模块中的“模拟输出”模块，设置为“高”，表示输出高电平，LED发光；如果设置为“低”，表示输出低电平，LED不发光。按照时间间隔，以此设置3个LED模块的亮灭，“延时”模块为“控制”类型模块，模拟霓虹灯程序模块如图10-10所示。



图10-10 模拟霓虹灯程序模块图

5. 测试运行

通过USB数据线连接Arduino板，选择正确的COM端口号，然后单击“上传”按钮，即可把程序指令发送给Arduino板，最后观察LED灯颜色的变化（为避免灯光刺眼，可将LED灯光对着白纸观察灯光颜色）。

自主探究

全彩LED模块可以实现模拟霓虹灯的绚丽效果，如果更换RGB颜色的组合还可以产生多种颜色。尝试修改图10-10所示模拟霓虹灯的程序，产生其他的灯光效果。

反思评价

对自己在本课中的学习进行评价，对照表10-4，在相应的○里画√，在“其他收获”栏里留言。

表10-4 学习评价表

序号	学习内容	能否掌握
1	了解软件与硬件的基本关系。	能○ 不能○
2	了解Arduino板的基本使用方法。	能○ 不能○
3	会使用Mixly软件。	能○ 不能○
4	会使用单色LED模块硬件。	能○ 不能○
5	会使用全彩LED模块硬件。	能○ 不能○
其他收获：		

广东教育出版社

第十一课 数字信号与模拟信号



学习目标

- 了解什么是数字信号与模拟信号。
- 掌握使用串口监视功能查看硬件反馈的信息。
- 学会开关模块与滑杆的使用。

知识导图

数字信号与模拟信号

- 数字信号与模拟信号的区别
- 数字信号的输入与输出
- 模拟信号的输入与输出

自然界有很多信息都可以用具体的物理量表示，比如温度的高低、光亮的强度、声音的大小等，这些光、声、电等承载信息的载体我们称为信号。信号对人们完成控制有重要价值。在电子控制系统中，通过传感器可以把其他信号转换成电信号，电信号可分为数字信号和模拟信号两种。

一、数字信号与模拟信号的区别

模拟信号为连续的信号，数字信号为离散不连续的信号。Arduino板可以接收两种信号。在Arduino板的模拟信号输入输出中，模拟信号的数据范围为0~1023。

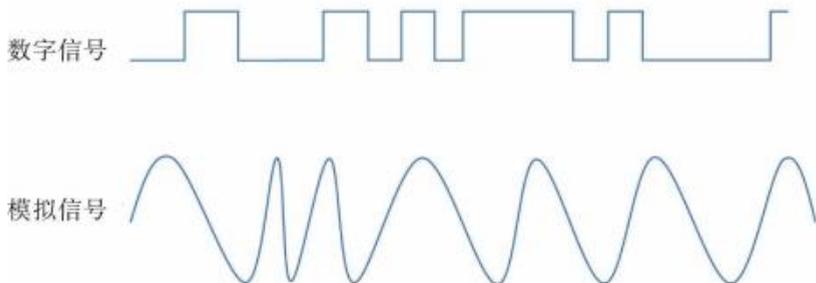


图11-1 数字信号与模拟信号对比

二、数字信号的输入与输出

Arduino板可以接收其他硬件模块输入的电信号，也可以输出电信号给其他硬件模块。Arduino板为完成信号的输入和输出，提供了丰富的接口，并对这些接口进行了编号。Arduino板信号接口对应表11-1所示。

表11-1 Arduino信号接口对应表

	数字信号	模拟信号
常用输出信号的引脚	0~13, A0~A5	3、5、6、9、10、11
常用输入信号的引脚	0~13, A0~A5	A0~A5
备注	0、1为串口通讯使用，Arduino板与电脑通讯使用，通常不适用这2个引脚连接硬件模块	

Arduino板可以通过数字引脚输出数字信号控制LED模块发光。接下来，我们将认识一种数字输入硬件——按键开关模块，通过其可以实现数字信号的输入。按键开关模块共3个引脚，分别接电源、接地和数字输入引脚，当按下开关，如表11-2所示。模块即可输出高电平信号；当不按开关时，输出为低电平信号。

表11-2 按键开关模块功能表

按键开关模块	引脚	引脚功能
	VCC	电源
	GND	接地
	OUT	信号输出引脚

学习任务

使用串口监控器查看数字输入

1. 问题分析

为了查看Arduino板是否收到按键开关模块输入的数字信号，我们可以使用串口监视器显示输入的信息，即可验证数字信号是否完成输入。

2. 硬件连接

串口监视查看数字输入的连接图如图11-2所示。

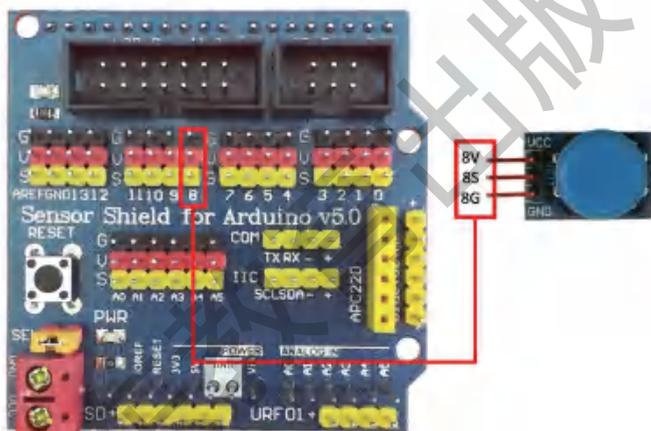


图11-2 串口监视器查看数字输入连接图

3. 设计算法

串口监视器查看数字输入流程图如图11-3所示。



图11-3 串口监视器查看数字输入流程图

4. 编写程序

使用串口类型中的“Serial打印（自动换行）”模块输出数字信号，数字输入模块在“输入/输出”类型中，设置引脚编号，设置延时1000毫秒实现1秒采集一次信息，如图11-4所示。



图11-4 串口监视器查看数字输入程序

5. 测试运行

通过USB数据线连接Arduino板，然后单击“编译”按钮，检查程序是否有错误。确认无误后，选择正确的COM端口号（通常为非COM1的编号），然后单击“上传”按钮把程序指令发送给Arduino板。打开串口监视器窗口，反复按键，即可查看监视器窗口显示的数字变化情况。

三、模拟信号的输入与输出

Arduino板可以输入模拟信号的有A0~A5共6个引脚，输出模拟信号的有3、5、6、9、10、11共6个引脚。Arduino板获取其他模块输入的模拟信号，其数值范围为0~1023；而输出给其他模块的模拟信号，其数值范围为0~255。

可调电阻可以输出模拟信号，其上有一个轨道和滑竿，其功能如表11-3所示。滑竿可以左右滑动，相应输出的模拟信号数值也会发生变化。此硬件模块有2组输出信号，我们只使用其中一组读取模拟信号。

表11-3 可调电阻功能模块表

按键开关模块	引脚	引脚功能
	VCC	电源
	GND	接地
	OTA	第1组模拟信号输出引脚
	OTB	第2组模拟信号输出引脚

学习任务

可调光小夜灯的制作

在晚上休息的时候，有时需要开灯，但是打开灯后灯光很刺眼。如果灯可以根据需要调节亮度，既能满足灯光需求又不刺眼。

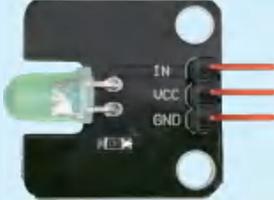
1. 问题分析

使用可调电阻模块可以给Arduino板输入模拟信号，信号数值范围从0~1023。使用模拟输出功能控制LED模块灯的亮度，信号数值范围从0~255。由于输入和输出的范围不同，所以还需要对数据进行转换。使用数字类型中的映射模块，将可调电阻变化范围0~1023映射为模拟输出的范围0~255。

2. 硬件清单和连接

制作可调光小夜灯的硬件清单如表11-4所示。

表11-4 可调光小夜灯硬件清单表

硬件名称	可调电阻模块	LED模块
硬件外观		
硬件功能	可调电阻可以左右滑动黄色的滑竿，实现输出从0~1023的模拟信号	全彩LED模块，通过模拟信号控制亮度，实现LED灯亮度的渐变
数量	1	1

可调光小夜灯硬件连接图如图11-5所示。

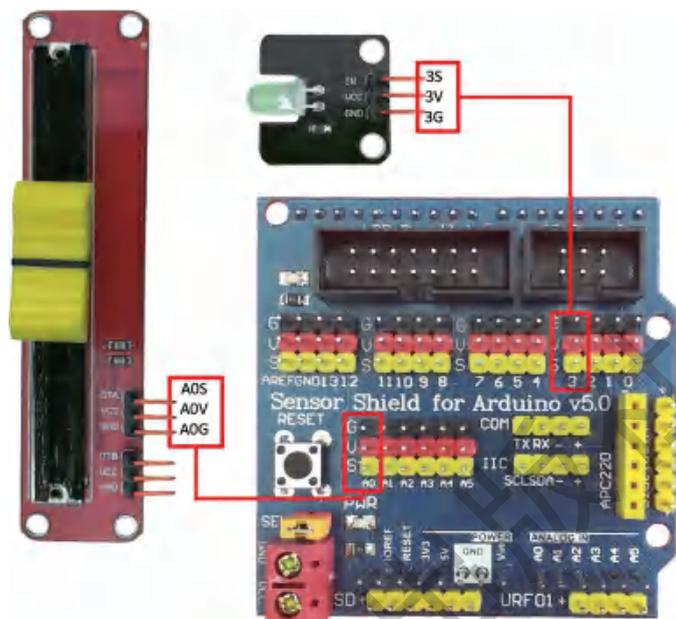


图11-5 可调光小夜灯硬件连接图

3. 设计算法

可调光小夜灯流程图如图11-6所示。

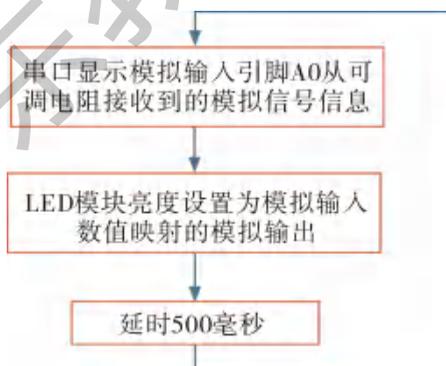


图11-6 可调光小夜灯流程图

4. 编写程序

串口类型中的打印模块输出可调电阻的数值，以便检查可调电阻模拟输入是否正确；模拟输出引脚3，模拟输入模块，数字类型中的映射模块完成模拟输入和模拟输出数值的转换。

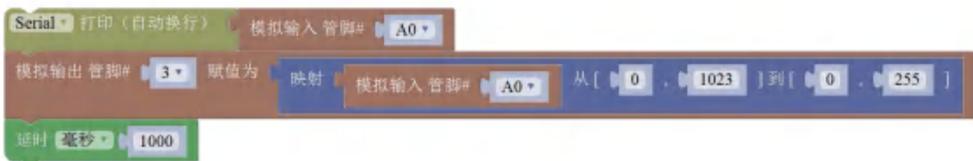


图11-7 可调光小夜灯程序

5. 测试运行

上传程序后，打开串口监视器窗口，然后拖动可调电阻的滑竿，查看监视器窗口显示的数字变化情况，也可使用绘图模式查看模拟输入数值的变化，最后观察LED灯的亮度是否随着可调电阻的调节而发生变化。

自主探究

日全食是太阳被月亮遮挡的天文现象。使用LED模块模拟太阳光，使用可调电阻滑竿的运动模拟月亮的运动。由于模拟月亮遮挡太阳光有两个过程，开始随着遮挡区域越来越大，LED灯的亮度越来越小，当可调电阻滑竿到达导轨中央时，达到全部遮挡LED灯，LED熄灭。继续拖动滑竿，模拟月亮遮挡区域越来越小，LED灯亮度最越来越大，最后LED灯为全亮状态。为了实现此效果，需要用到控制类型中的如果模块，对LED灯的亮度进行控制，可参考如图11-8所示的程序。



图11-8 模拟日全食程序

反思评价

对自己本课中的学习情况进行评价，对照表11-5，在相应的○里画√，在“其他收获”栏里留言。

表11-5 学习评价表

序号	学习内容	能否掌握
1	了解什么是数字信号与模拟信号。	能○ 不能○
2	掌握使用串口监视功能查看硬件反馈的信息。	能○ 不能○
3	学会开关模块的使用。	能○ 不能○
4	学会可调电阻的使用。	能○ 不能○
其他收获：		

广东教育出版社

第十二课 使用传感器采集信息



学习目标

- 知道什么是传感器，传感器的主要作用。
- 会使用传感器获取外界信息。

知识导图



在一些公共场所，当人接近门的时候，门会自动打开。这种自动门为什么会知道有人接近了门，又是如何控制门打开的呢？原来是在自动门上方安装有人体感应传感器，当人接近时会检测到人体信息，并将这些信息传递给门的控制器，并控制设备把门打开。这就是利用传感器实现了门的自动开关。

一、传感器

人可以用眼睛获取光线信息，可以用耳朵获取声音信息，当外界信息传递给人的大脑，人能根据这些信息做出回应。传感器就像电子设备的感觉器官一样，可以采集外界信息，并且将这些信息转换为电信号或其他信号。而这些电信号传递的信息，经过Arduino控制器处理后，又可以改变一些装置的状态作为回应，如图12-1所示。



图12-1 传感器与Arduino的工作流程

传感器可以将外界信息转换为电信号，电信号的大小变化，即反映外界信息的变化。比如光敏传感器，可以将光线的强弱变化，转换为电信号的大小变化。

小组讨论

除了光敏传感器，小组讨论还有其他种类的传感器，并填写表12-1。

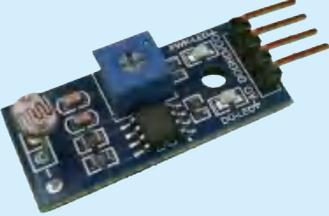
表12-1 常见传感器

传感器	对应人体感觉器官	获取哪些外部信息	生活中的应用举例
光敏传感器	眼睛	光线	楼梯光控灯
温敏传感器			
热敏传感器			
声敏传感器			
气敏传感器			

二、光敏传感器模块

光敏传感器模块可以将光信号转换为电信号，其功能如表12-2所示，使用模拟输入读取的值范围为0~1023。当光线强时，电信号数值较小；当光线弱时，电信号数值较大。光线的强弱信息即可转换为电信号，并用数值的大小表示光线强弱的变化。

表12-2 光敏传感器模块功能表

光敏传感器模块	引脚	引脚功能
	VCC	电源
	GND	接地
	DO	数字信号输出
	AO	模拟信号输出

学习任务

测量光敏传感器输入的信号值

光敏传感器模块可以把光线的强度转换为电信号，Arduino板获取电信号后，可以使用这些电信号数据进行相应的控制。

1. 问题分析

光敏传感器模块输入模拟信号，Arduino板获取模拟信号，并使用串口打印功能显示测量的数值。通过自然光状态和手指遮挡状态的转换，查看模拟信号的变化。

2. 硬件连接

本案例中需要用的硬件清单如表12-3所示。

表12-3 测量光敏传感器输出信号的值所需硬件清单

硬件名	光敏传感器
硬件外观	
硬件功能	用于检测光线的强弱
数量	1

取出光敏传感器，将GND引脚接入扩展板上模拟信号A0中的G，将VCC引脚接入模拟信号A0中的V，将A0引脚接入模拟信号A0中的S，D0引脚留空，如图12-2所示。

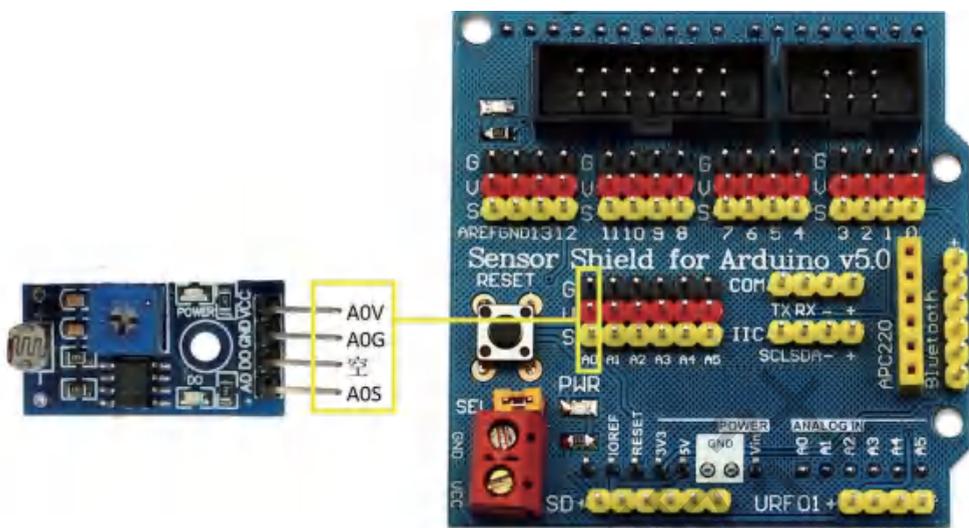


图12-2 测量光敏传感器输入的信号值硬件连接图

3. 设计算法

测量光敏传感器输入的信号值流程图如图12-3所示。

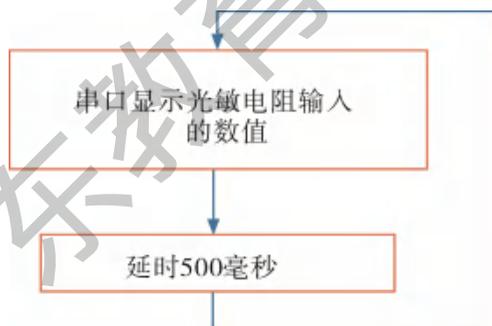


图12-3 测量光敏传感器输入的信号值流程图

4. 编写程序

测量光敏传感器输入的信号值流程图的程序如图12-4所示。



图12-4 测量光敏传感器输入的信号值程序

5. 测试运行

上传程序后，打开串口监视器窗口，在自然光状态下，查看光敏传感器模拟输入的数值，再通过手指遮挡光敏传感器的感光部位，查看串口监视器的变化。自然光状态，串口输出的电信号数值大约为_____；手指遮住感光部位后，串口输出的电信号数值大约为_____。通过对这些模拟信号进行判断，就可以实现对其他硬件模块的控制。

三、倾斜传感器模块

倾斜传感器模块可以通过改变模块的角度获取倾斜的信息，其功能如表12-4所示。当模块水平时，DO引脚输出数字信号为高电平；当模块倾斜超过15°后，DO引脚输出数字信号为低电平，这样就可以检测模块是否有倾斜的情况。

表12-4 倾斜传感器模块功能表

倾斜传感器模块	引脚	引脚功能
	VCC	电源
	GND	接地
	DO	数字信号输出

学习任务

自行车头盔转向灯

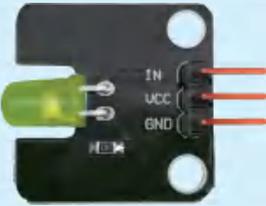
自行车是非常方便的代步工具，我们可以设计一款具有转向灯功能的头盔，让骑行者在转向时更好地保护自己。

1. 问题分析

自行车安全头盔具备转向灯的功能，最方便的控制方法就是当头部倾斜，转向灯可以直接亮起来。即当头盔向左倾斜时，左边的转向灯亮；向右倾斜时，右边的转向灯亮。

2. 硬件清单和连接

表12-5 自行车头盔转向灯所需硬件清单

硬件名称	黄色LED灯模块	倾斜传感器
硬件外观		
硬件功能	亮灯表示转向	用于检测倾斜状态
数量	2	2

(1) 取出一个黄色LED灯模块，将GND引脚接入扩展板上8 G，VCC引脚接入8 V，IN引脚接入8 S。取出另一个黄色LED灯模块，将GND引脚接入扩展板上6 G，VCC引脚接入6 V，IN引脚接入6 S。

(2) 取出一个倾斜传感器，将GND引脚接入扩展板上11 G，VCC引脚接入11 V，DO引脚接入11 S。取出另一个倾斜传感器，将GND引脚接入扩展板上3 G，VCC引脚接入3 V，DO引脚接入3 S，如图12-5所示。

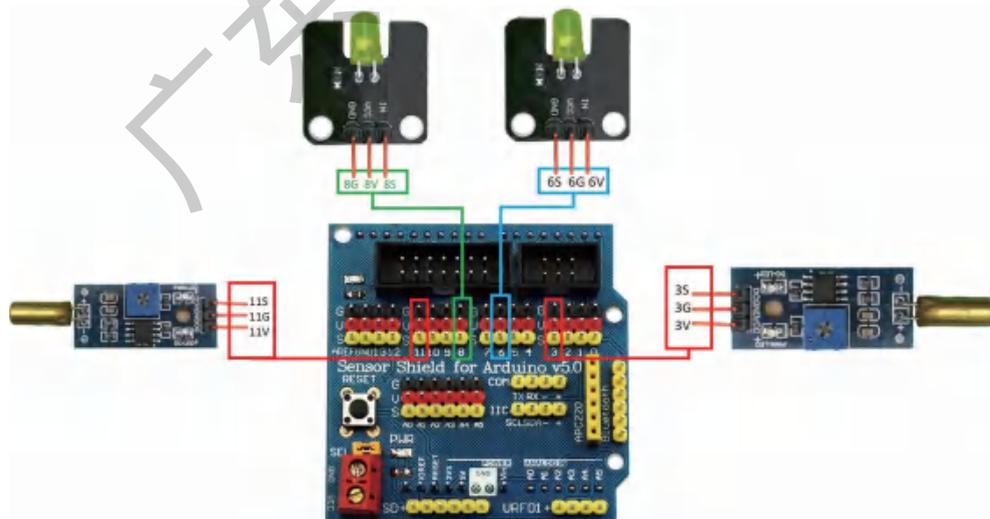


图12-5 自行车头盔转向灯硬件连接图

3. 设计算法

自行车头盔转向灯流程图如图12-6所示。

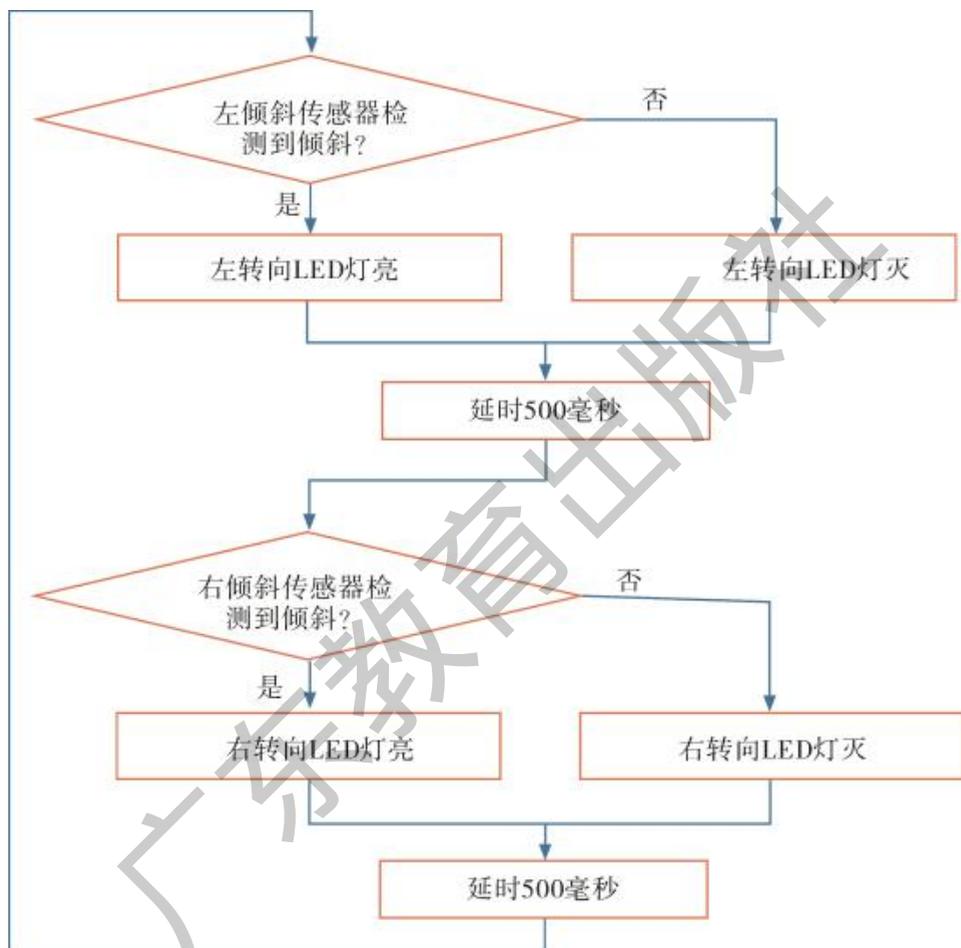


图12-6 自行车头盔转向灯流程图

4. 编写程序

自行车头盔转向灯程序如图12-7所示。



图12-7 自行车头盔转向灯程序

5. 测试运行

上传程序后，左倾斜传感器倾斜 15° 。因为倾斜传感器模块的倾斜开关要向上倾斜 15° ，所以左倾斜传感器应该对着右边，然后整体向左倾斜测试，测试可以让左边的LED灯亮起来，即可起到转向灯的作用。右倾斜可以使用同样的方法测试。

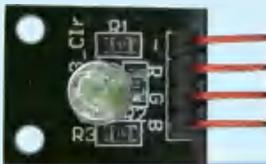
四、其他传感器模块

传感器的种类有很多，如光敏传感器可以获取光线信息；声敏传感器可以获取声音信息；倾斜传感器获取角度信息；等等。Arduino板通过传感器获取外界的信息，然后对这些信息做出判断，完成相应的控制。

自主探究

触摸传感器是可以获取人手指触摸的信息。例如，有时上课需要抢答问题，那我们使用2个触摸传感器模块和1个全彩LED模块制作一个抢答器，哪个抢答的速度快，就亮相应颜色的灯。为了不影响下次使用，LED灯可以连续亮几秒后自动熄灭，并开始准备下次抢答。制作抢答器所需材料清单如表12-6所示，连接图如图12-8所示。

表12-6 测量光敏传感器输出信号的值所需硬件清单

硬件名	全彩LED灯模块	触摸传感器
硬件外观		
硬件功能	根据程序要求亮不同颜色的灯	用于检测触摸
数量	1	2

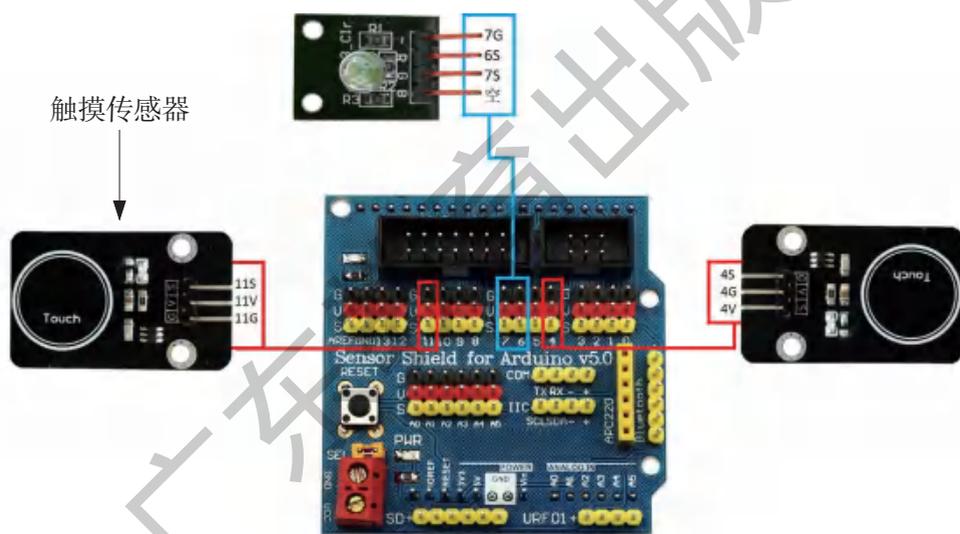


图12-8 抢答器硬件连接图

小组讨论

小组讨论传感器还可以采集哪些外界信息，并可以完成哪些控制系统的设计，同时完成表12-7。

表12-7 利用传感器设计其他

传感器	采集信息	实现控制的功能	生活中的应用举例
烟雾（可燃气 体）传感器			楼梯光控灯
温度传感器			
震动传感器			
磁敏传感器			
色彩传感器			
压力传感器			



反思评价

对自己在本课中的学习进行评价，对照表12-8，在相应的○里画√，在“其他收获”栏里留言。

表12-8 学习评价表

序号	学习内容	能否掌握
1	知道什么是传感器，传感器的主要作用。	能○ 不能○
2	会使用光敏传感器模块获取光线信息。	能○ 不能○
3	会使用倾斜传感器模块。	能○ 不能○
4	会使用触摸传感器。	能○ 不能○
5	我仔细思考了探究的问题。	能○ 不能○
其他收获：		

第十三课 控制输出模块工作



学习目标

- 了解输出模块的作用。
- 认识并学会使用蜂鸣器模块。
- 认识并学会使用扁平振动马达模块。

知识导图



通过学习我们知道Arduino板可以通过传感器获取外界世界丰富多彩的信息。Arduino板可以对这些信息处理，并做出一些反应，这就需要能执行这些反应任务的硬件设备了。

一、输出模块

Arduino板输出的数字信号或模拟信号，可以实现各种输出模块的控制。最常用的输出模块，主要有LED灯、电机、声音等。

二、蜂鸣器模块

蜂鸣器模块是一种一体化结构的电子讯响器，其功能如表13-1所示。声音是由物体振动发生的，而我们使用的蜂鸣器模块是有源蜂鸣器，它自带振荡源，一通电就会响。它被广泛地应用于电子门铃、电脑开机提示音等各类电子设备的警示提示音发声。在Mixly软件中，我们一般将蜂鸣器接

入数据端口，并采用  模块驱动它。值得注意的是，有的蜂鸣器是设置为“高”电平时发声，有的蜂鸣器则是设置为“低”电平时发声。

表13-1 蜂鸣器功能表

蜂鸣器	引脚	引脚功能
	GND	接地
	VCC	电源
	SIG	数字信号

物体在一秒钟之内振动的次数叫做频率，单位是赫兹（Hz）。蜂鸣器也可通过  模块，调节不同输出频率，驱动蜂鸣器发出类似于音乐中“do、re、me”的不同音调。表13-2所示为音调节频率的对应关系表。

表13-2 音调与频率的对应关系

音调 (低)	do	re	mi	fa	so	La	Xi
频率	131	147	165	175	196	220	247
标名	NOTE_ C3	NOTE_ D3	NOTE_ E3	NOTE_ F3	NOTE_ G3	NOTE_ A3	NOTE_ B3
音调 (中)	do	re	mi	fa	so	La	Xi
频率	262	294	330	349	392	440	494
标名	NOTE_ C4	NOTE_ D4	NOTE_ E4	NOTE_ F4	NOTE_ G4	NOTE_ A4	NOTE_ B4
音调 (高)	do	re	mi	fa	so	La	Xi
频率	523	587	659	698	784	880	988
标名	NOTE_ C5	NOTE_ D5	NOTE_ E5	NOTE_ F5	NOTE_ G5	NOTE_ A5	NOTE_ B5

三、扁平振动马达

扁平振动马达广泛应用于手机、平板电脑或需要振动的环境中实现振动功能，其可以取代声音提示使用者，避免惊扰他人或产生噪音，属于直流有刷电机，其功能如表13-3所示。马达轴上面有一个偏心轮，当马达转动的时候，偏心轮的圆心质点不在电机的转心上，从而使得马达处于不断的失去平衡状态，并由于惯性作用引起震动。

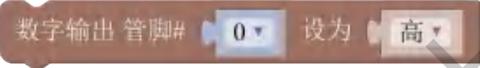
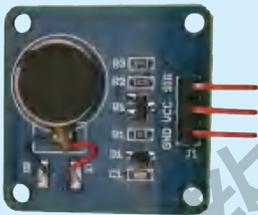
在Mixly中，我们使用  模块来驱动扁平振动马达。

表13-3 扁平振动马达功能表

扁平振动马达	引脚	引脚功能
	GND	接地
	VCC	电源
	SIG	数字信号

学习任务

光感警车闹钟

基础方案

1. 问题分析

“铃……铃……铃……！”太阳都晒到床上了，闹钟都还没将熟睡的你闹醒，因为你的潜意识里已经对这单调的闹铃声产生了“自动屏蔽”功能。如何能变个花样，让熟睡的你能立刻弹身而起呢？那就让我们一起来制作光感警车闹钟吧！

光感警车闹钟启动触发的条件是“天亮了”。如何让Arduino知道现在天亮了呢？这里我们需要选择之前给大家介绍的传感器之一——光敏传感器。要让闹钟发出警车的声音，就需要用到执行器之一——蜂鸣器。

知识链接

警车的频率，高频是1450~1550Hz，持续时间约100毫秒；低频是650~750Hz，持续时间约230毫秒。其各项参数可根据个人喜好稍做修改。

2. 设计算法

光感警车闹钟流程图如图13-1所示。

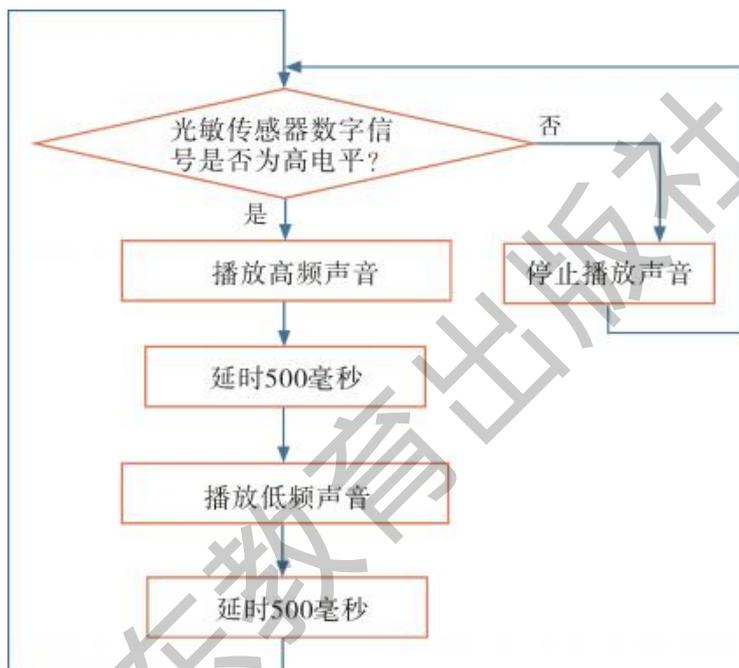


图13-1 光感警车闹钟流程图

3. 硬件清单和连接

光感警车闹钟基础方案所需硬件如表13-4所示。

表13-4 光感警车闹钟基础方案所需硬件清单

硬件名	光敏传感器	蜂鸣器
硬件外观		
硬件功能	用于检测光线强弱	闹钟提示发音
数量	1	1

- (1) 取出光敏传感器，用杜邦线将其GND接5 G，VCC接5 V，DO接5 S。
- (2) 取出蜂鸣器模块，用杜邦线将其GND接8 G，VCC接8 V，S接8 S。
- (3) 注意逐个接口对应连线。光感警车闹钟基础方案连接图如图13-2所示。

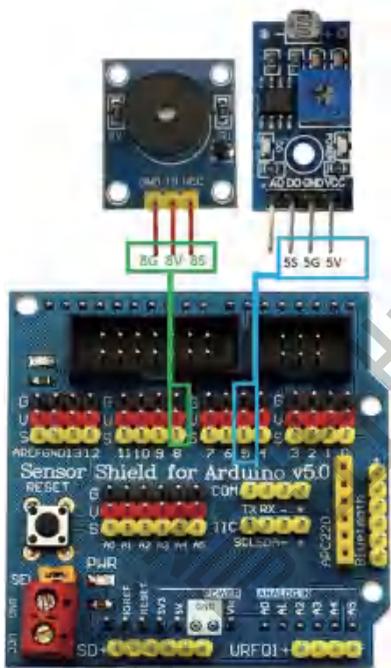


图13-2 光感警车闹钟硬件基础方案连接图

4. 编写程序

光感警车闹钟程序如图13-3所示。

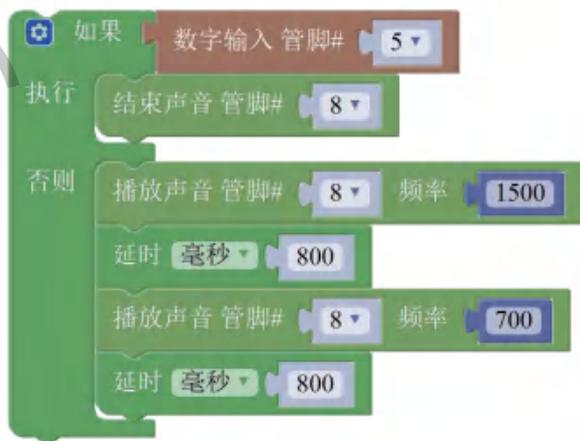


图13-3 光感警车闹钟程序

升级方案

给警车闹钟加入灯光和振动效果等，可以让作品更加生动。

1. 硬件清单和连接

光感警车闹钟升级方案所需硬件如表13-5所示。

表13-5 光感警车闹钟升级方案所需硬件清单

硬件名称	光敏传感器模块	蜂鸣器模块	全彩LED模块	扁平振动马达模块
硬件外观				
硬件功能	用于检测光线强弱	闹钟提示发音	彩灯提醒	振动提醒
数量	1	1	1	1

在基础方案的基础上，再添加全彩LED模块和振动马达模块。

(1) 使用全彩LED模块，用杜邦线将其GND接9 G，R接9 S，G接10 S，B接11 S。

(2) 使用振动马达模块，用杜邦线将其GND接12 G，VCC接12 V，SIG接12 S。光感警车闹钟升级方案硬件连接如图13-4所示。

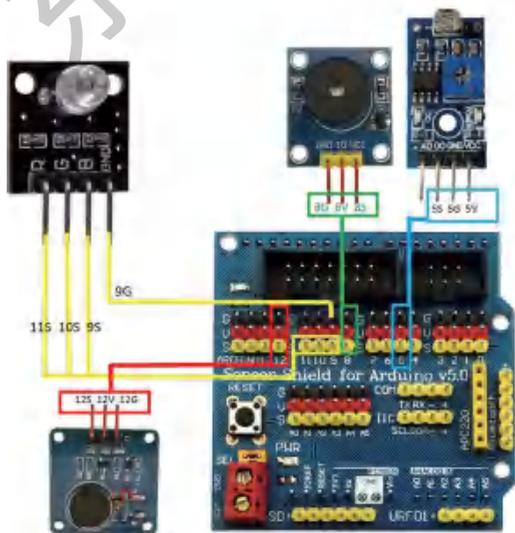


图13-4 光感警车闹钟升级方案硬件连接图

2. 编写程序

光感警车闹钟升级方案程序如图13-5所示。



图13-5 光感警车闹钟升级方案程序

自主探究

- (1) 光敏传感器上的“十”字旋转调节钮，有何作用？
- (2) 夏天，天亮得早；冬天，天亮得晚。如你能根据季节的变化调整光敏警车闹钟的启动的条件呢？同学们可以结合数字信号与模拟信号中学习的内容，思考如何将信号线接在光敏传感器的A0接口上。
- (3) 同学们还能将各类输出模块用在哪些场景中？

反思评价

对自己在本课中的学习进行评价，对照表13-6，在相应的○里画√，在“其他收获”栏里留言。

表13-6 学习评价表

序号	学习内容	能否掌握
1	了解输出模块的作用。	能○ 不能○
2	掌握执行器的使用方法。	能○ 不能○
3	蜂鸣器模块的使用。	能○ 不能○
4	能使用扁平振动马达模块。	能○ 不能○
其他收获：		

第十四课 电动机与舵机



学习目标

- 认识电动机与舵机模块。
- 掌握一些基本的机械运动模块的使用方法。

知识导图



在生活中，我们常见的电风扇、汽车飞奔的车轮、小区汽车出入的闸门等是靠什么东西动起来的呢？今天我们就来了解一下电动机与舵机。

一、电动机模块

电动机是应用电磁感应原理运行的旋转电磁机械，可产生旋转。在开源硬件中，马达就是电动机中最简单、最常见的一种。不同大小、型号的马达，转速、输出扭矩也不一样，如图14-1所示。



图14-1 各种不同型号的马达

为了能更好控制电动机的运转，电机驱动板是必不可少的。我们使用的是一种马达、驱动板合一的电动机模块，有四个引脚，分别为GND接地、VCC接电源、1A高电平时马达正转、1B高电平时马达反转，如表14-1所示。

表14-1 马达模块功能表

马达模块	引脚	引脚功能
	1B	驱动马达反转
	1A	驱动马达正转
	GND	接地
	VCC	电源

学习任务

自动降温杯垫

1. 问题分析

温开水是最健康的饮料，可是经常我们烧开了水后，由于水温太高，需要放凉后再喝。可是水温要从100度降到适宜饮用的35度左右，需要很长一段时间，长到放在那里后就忘记了喝水。

在开源硬件中，有个传感器叫热敏传感器，它可以判断温度的高低。如果我们将它安装在杯垫的底部，检测到高温时，启动杯垫自带的马达转动扇风为水降温，这样就能更快的喝到水了。同时，再安装一个蜂鸣器，当温度合适时鸣响提示，我们就知道什么时候可以喝水了。

2. 硬件清单和连接

制作自动降温杯垫所需硬件如表14-2所示。

表14-2 自动降温杯垫所需硬件清单

硬件名	马达模块	小风扇叶	热敏传感器	蜂鸣器
硬件外观				
硬件功能	产生转动，带动风扇叶	旋转产生风	检测温度的高低	发出提示声音
数量	1	1	1	1

(1) 取出马达，使用杜邦线将马达的VCC引脚接入扩展板上9 V，GND引脚接入扩展板上9 G，1A引脚接入扩展板上9 S。

(2) 取出蜂鸣器，使用杜邦线将GND引脚接入扩展板上8 G，VCC引脚接入扩展板上8 V，SIG引脚接入扩展板上9 S。

(3) 取出热敏传感器，使用杜邦线将GND引脚接入扩展板上5 G，VCC引脚接入扩展板上5 V，DO引脚接入扩展板上5 S。AO引脚是作为模拟信号接入时使用，本案例中不需接入。热敏传感器安装在杯垫的底部，与杯底接触测温。

自动降温杯垫的硬件连接如图14-2所示，外观模型示意图如图14-3所示。

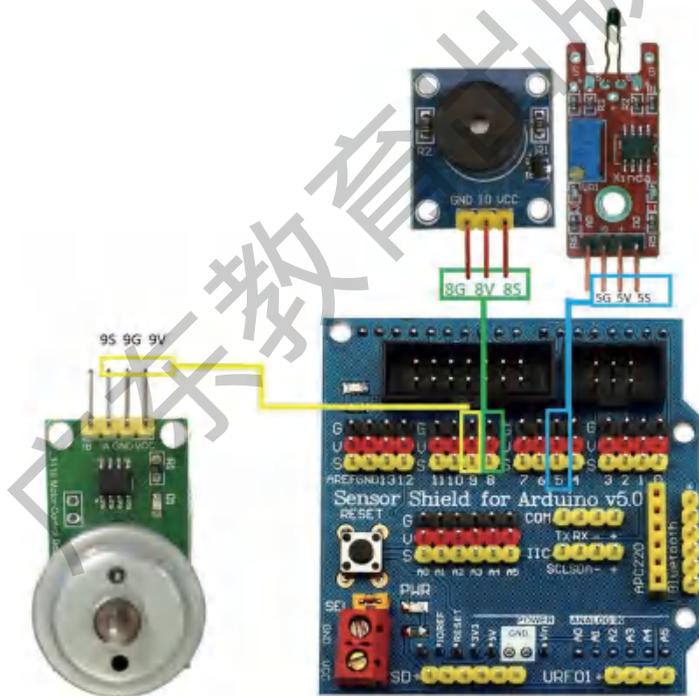


图14-2 自动降温杯垫的硬件连接图

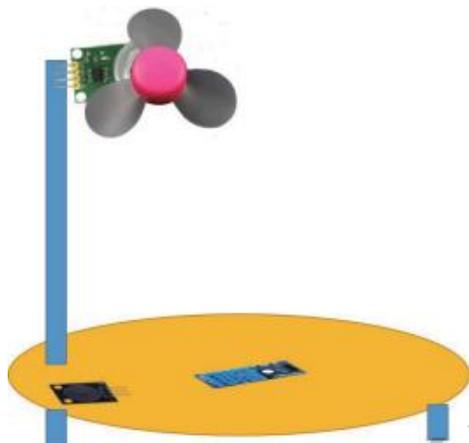


图14-3 自动降温杯垫的外观模型示意图

3. 设计算法

自动降温杯垫设计流程图如图14-4所示。

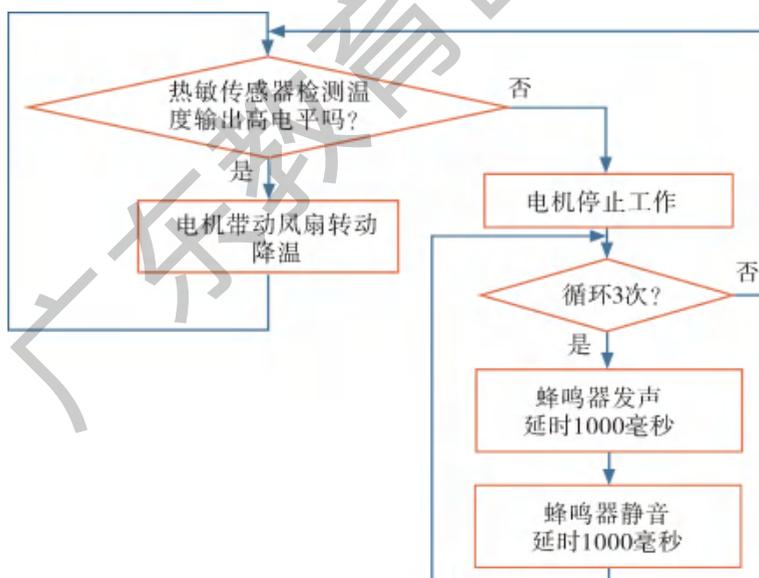


图14-4 自动降温杯垫设计流程图

4. 编写程序

自动降温杯垫程序如图14-5所示。



图14-5 自动降温杯垫程序

5. 测试运行

将热敏传感器的黑色感应头贴在装有开水的杯子底部，它能感应到透过杯壁的水的热量。当水温足够高时，马达启动，风扇转动，为杯子里的水降温；当水温降到一定温度时，马达停止转动，蜂鸣器响三声提示。

自主探究

- (1) 如何可以让温度控制得更精准？
- (2) 除了这种自带驱动板的马达模块，更多的马达是不自带驱动的，而是使用一种型号叫L298N的驱动板（图14-6）。同学们可以对此进行继续研究。

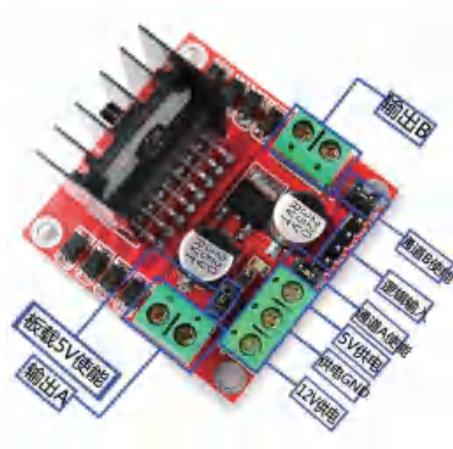


图14-6 L298N驱动板

二、舵机

舵机内部也有电动机，只不过舵机不像普通电动机那样可以360°一直旋转，而只可以在0~180°之间转动，摆动到指定角度，就像船舵控制船的前进方向，只需要一定角度即可。一般舵机都自带一条三引脚连接线，分棕、红、黄三色，其功能如表14-3所示。与之配套的，还有几个白色塑料指针，适合不同需求使用。

表14-3 舵机功能表

舵机	引脚	引脚功能
	棕色线	接地
	红色线	电源
	黄色线	数字信号

学习任务

硬币自动分拣机

1. 问题分析

储蓄罐中大大小小的硬币，如何让它自动分类出来呢？我们以分拣出1角硬币为例，使用舵机，利用它摆动产生的离心力，制作一个硬币自动分拣机。

2. 硬件清单和连接

硬币自动分拣机所需硬件如表14-4所示。

表14-4 硬币自动分拣机所需硬件清单

硬件名	舵机
硬件外观	
硬件功能	产生摆动
数量	1

- (1) 取出舵机，一般它都自带一条三合一线，将棕色线口插入扩展板上11 G，红色线口插入11 V，黄色线口插入11 S，如图14-7所示。
- (2) 取出舵机标配一字型指针插在舵机顶部。
- (3) 找一个圆形硬纸盒盖，在盒顶部按1角硬币大小挖一个洞，当作筛选硬币的筛子，然后将盒盖固定在舵机指针上，如图14-8所示。

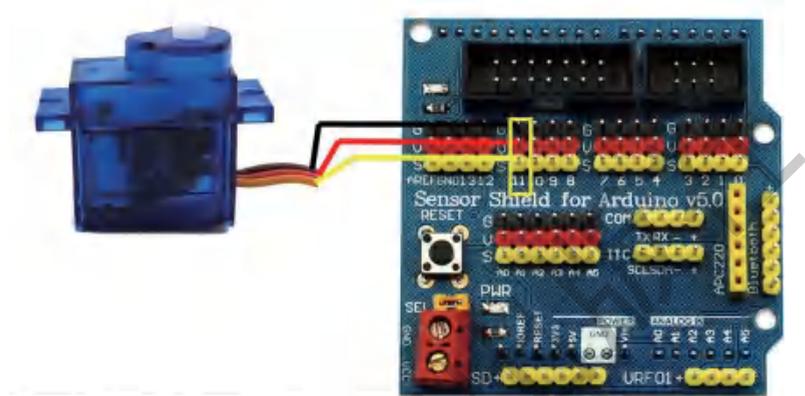


图14-7 硬币自动分拣机硬件连接图

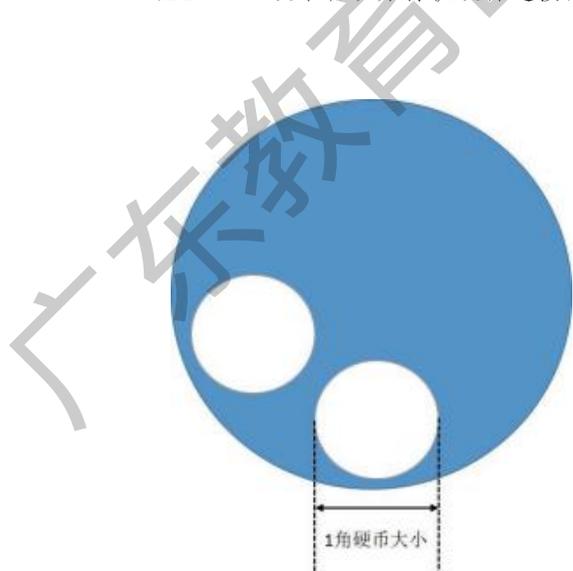


图14-8 硬币自动分拣机的筛子图示

3. 设计算法

硬币自动分拣机流程图如图14-9所示。



图14-9 硬币自动分拣机流程图

4. 编写程序

硬币自动分拣机程序如图14-10所示。



图14-10 硬币自动分拣机程序

5. 测试运行

代码上传后，舵机开始不停地左右摆动筛子，硬币被甩到刚好大小的洞时，自动掉出来。

自主探究

- (1) 尝试调节舵机旋转角度的参数或延时参数，看看会有什么效果？
- (2) 利用舵机的特性，你还能制作其他什么作品吗？

反思评价

对自己在本课中的学习进行评价，对照表14-5，在相应的○里画√，在“其他收获”栏里留言。

表14-5 学习评价表

序号	学习内容	能否掌握
1	掌握机械运动模块的使用方法。	能○ 不能○
2	掌握马达和舵机的使用方法。	能○ 不能○
其他收获：		

第十五课 数码显示与无线通讯的实现



学习目标

- 电子控制的实现。
- LCD屏幕的使用。
- 红外信号的接收与显示。

知识导图



在第三课中，我们学到可以使用串口监听功能，查看当前获取的传感器数据。这种查看数据的方式离不开电脑。我们经常需要将采集到的数据直接显示在创意作品上，这就离不开数据显示屏了。

一、LCD显示屏的使用

1602字符型液晶LCD也叫1602液晶，它是一种专门用来显示字母、数字、符号等的点阵型液晶模块。它能够同时显示16字×2行即32个字符，但不能显示中文字符。1602LCD模块通常有16个引脚。由于16个引脚连接Arduino过于繁琐，我们一般都使用转接模块将16个引脚转为包含GND、VCC、SDA、SCL等的4个引脚使用，1602LCD模块功能如表15-1所示。

表15-1 1602LCD模块功能表

1602LCD模块	引脚	引脚功能
 <p>正面</p>	GND	接地
	VCC	5V电源正极

(续表)

1602LCD模块	引脚	引脚功能
 <p>背面</p>	SDA	双向数据信息通道
	SCL	时序信号通道

需要特别注意的是，在Mixly中要驱动1602LCD模块，必须先在“初始化”模块中指定设备地址，不同厂家的设备地址不一样。本书使用的1602LCD模块，设备地址是“0x3F”，如图15-1所示。

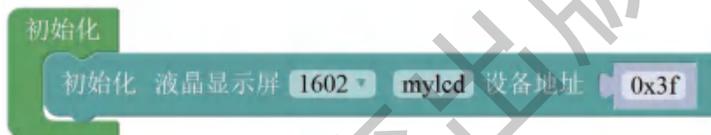


图15-1 1602LCD模块初始化命令

学习任务

智能自动计步鞋

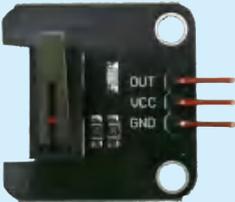
1. 问题分析

随着人们的健康意识越来越强，更多人关注自己每天走了多少步。现在市场上也有许多方便携带的具有计步功能的手机、手环等。但不同产品，它的算法不一样，导致每日计步的数据也有差别。如果将计步的传感器直接安装在鞋里，毋庸置疑，这种方式的计步将更精准。

2. 硬件清单和连接

本案例中需要使用的硬件清单如表15-2所示。

表15-2 智能自动计步鞋硬件清单

硬件名	碰撞传感器	1602LCD模块
硬件外观		
硬件功能	用于检测是否被碰撞压住，使用方法及原理与按钮模块相仿	字符液晶显示屏
数量	1	1

(1) 取出碰撞传感器，将GND引脚接入扩展板上2号口的G，VCC引脚接入2号口的V，OUT引脚接入2号的S。本模块在实际安装时，需安装在鞋内脚跟处。

(2) 取出1602LCD模块，背面的转接模块已经将原先16个引脚转化为4个引脚。扩展板上有个专用IIC区域，可接LCD屏。将GND引脚接入IIC区的-，VCC引脚接入IIC区的+，SDA引脚接入IIC区的SDA，SCL引脚接入IIC区的SCL，智能自动计步鞋的硬件连接示意图如图15-2所示。

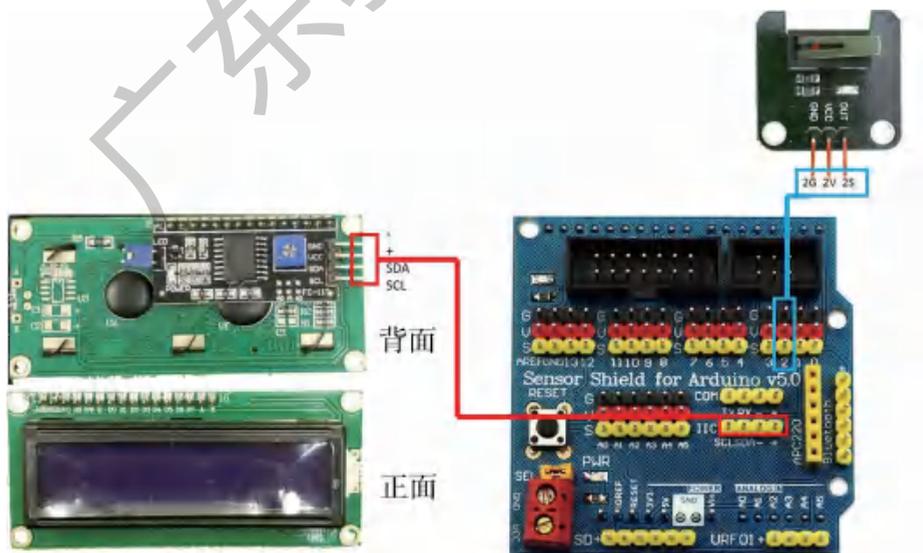


图15-2 智能自动计步鞋的硬件连接示意图

3. 设计算法

知识链接

“布尔型”变量

布尔型是编程软件中的一种数据类型，布尔型变量只有两种逻辑状态——真或假，一般用于条件判断，也可用于标识某个状态，相当于一个标记。

在本案例中，需要用到一个数值型变量和布尔型变量。数值型变量用于存储计算得到的步数，布尔型变量用于标记按钮的状态。这些变量，需要在“初始化”模块中提前设置，智能自动计步器设计流程图如图15-3所示。

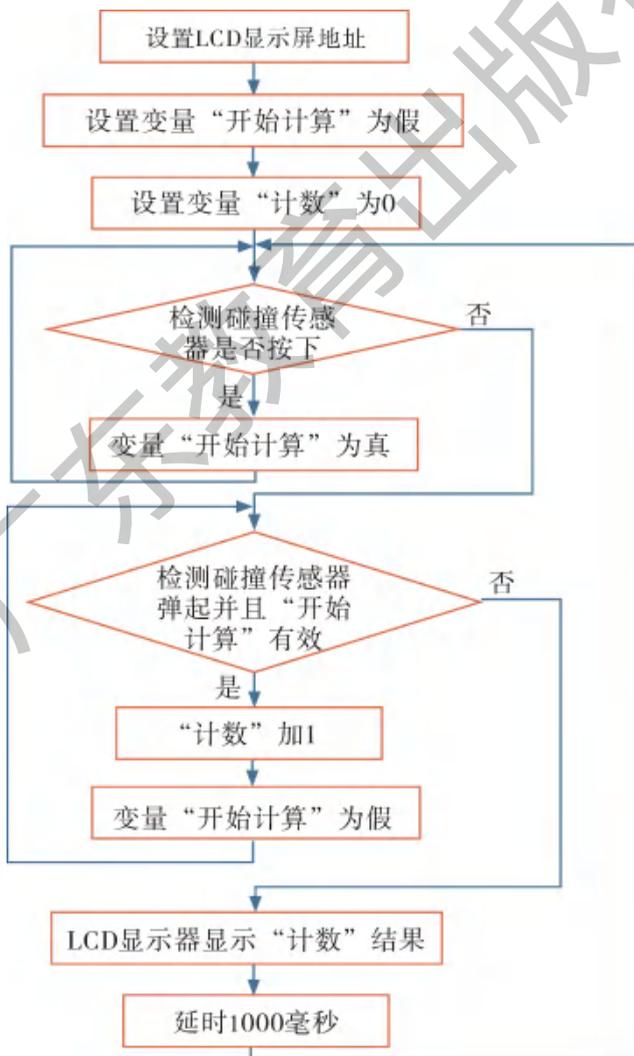


图15-3 智能自动计步器设计流程图

4. 编写程序

智能自动计步器代码如图15-4所示。



图15-4 智能自动计步器代码

5. 测试运行

通过USB数据线连接Arduino板，选择相应的端口号和主控板型号，点击“上传”。每当按下碰撞传感器的弹片，然后再弹起来后，LCD显示器的数字都会增加1。

二、红外信号的接收与显示

在生活中，我们按一下遥控器，电视机就能打开；按一下遥控器，空调就能启动；按一下遥控器，玩具车就能往前走。控制器与电视机、空调、玩具车之间建立了什么连接？它是如何实现“遥控”功能的呢？

红外线通信系统一般由红外线发射系统和接收系统组成。发射系统一般指红外遥控器，接收系统与被控电子设备连接。遥控器每个按钮被按下的时候，都会发射一个特定的16进制的代码。本书配套的遥控器发射的代码，都是以0xFF开头的4位数字和字母组合，如0xFF18E7。我们可以通过串口监视器，查出遥控器上每个按钮的发射代码，如图15-5所示。



图15-5 通过串口查看红外发射代码的程序

需要说明的，由于硬件精准度的问题，代码在传输过程中可能会出错，接收到的数据会出现以非0xFF开头的数字。此类信息不影响我们的程序正常运行，可以忽略，如图15-6所示。

```

TYPE:UNKNOWN 5182021B
TYPE:RC5 FF30CF 有效
TYPE:SHARP FF
TYPE:RC5 FF30CF
TYPE:SHARP FF
TYPE:RC5 FF18E7 有效
TYPE:RC5 FF18E7
TYPE:RC5 FFFFFFFF
TYPE:SHARP FF
TYPE:RC5 FF18E7
TYPE:UNKNOWN 5182021B
TYPE:RC5 FF7A85
TYPE:RC5 FF7A85
TYPE:UNKNOWN 5182021B
    
```

图15-6 串口监视器查看到的数据

学习任务

万能摇控器

当我们了解红外摇控器实际发射的代码数据时，就可以使用程序的条件判断语句来决定当按下哪个按钮时执行何种命令了，从而让摇控器变成“万能摇控器”。例如，我们可以命令按1时亮灯、按2时灭灯；也可以命令按1时蜂鸣器响起、按2时蜂鸣器静音等。在下面的案例中，我们将使用红外摇控器来舵机。

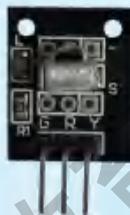
1. 问题分析

在住宅小区门口，一般都有控制汽车进出的栏杆。小区保安可以使用摇控器控制栏杆的打开与关闭。我们可以使用万能摇控器，结合舵机，设计一个智能闸门。

2. 硬件清单和连接

本案例使用的硬件清单如表15-3所示。

表15-3 万能摇控器闸门控制系统硬件清单

硬件名	红外摇控器	红外接收模块	舵机	1602LCD模块
硬件外观				
硬件功能	发送红外信号	接收红外信号	控制指针转动角度	显示提示文字
数量	1	1	1	1

(1) 取出舵机，将棕色GND引脚接入扩展板上4号口的G，红色VCC引脚接入4号口的V，黄色SIG引脚接入4号的S。在舵机上安装单向塑料指针片，并可在指针上安装模拟闸门的吸管。

(2) 取出1602LCD模块，将GND引脚接入IIC区的-，VCC引脚接入IIC区的+，SDA引脚接入IIC区的SDA，SCL引脚接入IIC区的SCL。

(3) 取出红外接收模块，将GND引脚接11号口的G，VCC引脚接11号口的V，OUT引脚接11号的S，万能摇控器闸门控制系统硬件连接示意图如图15-7所示。

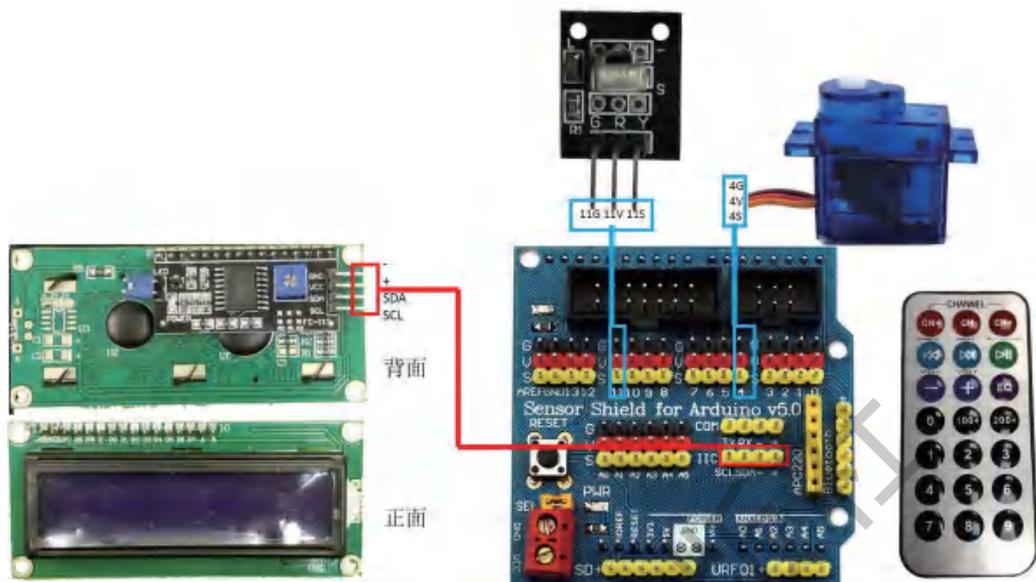


图15-7 万能摇控器闸门控制系统硬件连接示意图

3. 设计算法

万能摇控器闸门控制系统设计流程图如图15-8所示。

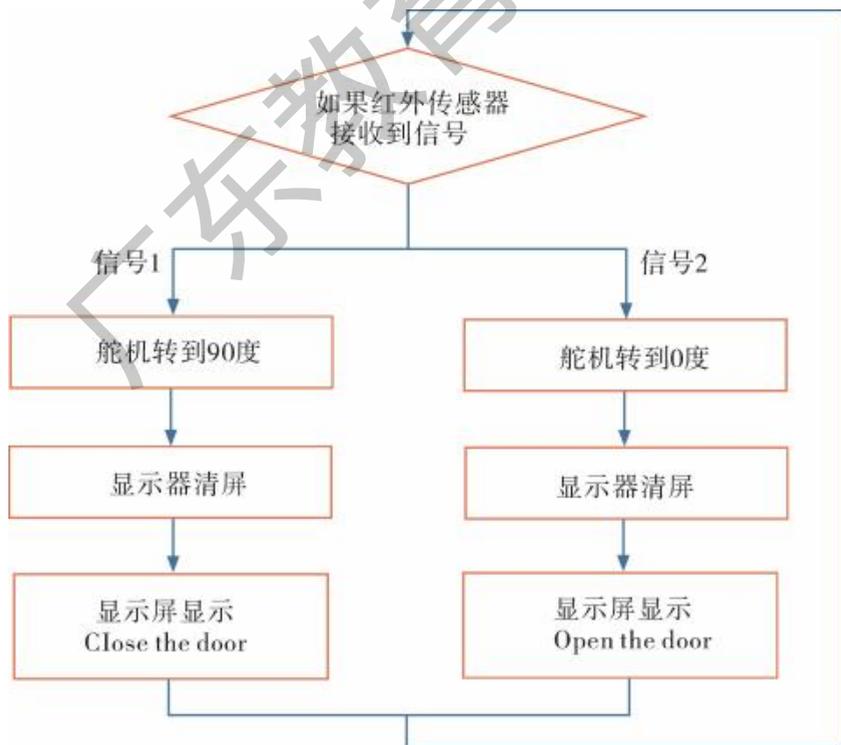


图15-8 万能摇控器闸门控制系统设计流程图

4. 编写程序

万能摇控器闸门控制系统的程序代码如图15-9所示。



图15-9 万能摇控器闸门控制系统程序代码

5. 测试运行

按下摇控器上的数字1，舵机转向90度，显示屏上显示“Close the door”；按下摇控器上的数字2，舵机转向0度，显示屏上显示“Open the door”。

自主探究

尝试使用摇控器控制RGB灯的颜色，思考如何编写代码。

将你对以上问题思考的点滴记录下来吧!



反思评价

对自己在本课中的学习进行评价，对照表15-4，在相应的○里画√，在“其他收获”栏里留言。

表15-4 学习评价表

序号	学习内容	能否掌握
1	掌握LCD屏和红外摇控器的使用方法。	能○ 不能○
2	合作完成自动计步器。	能○ 不能○
3	合作完成万能摇控器。	能○ 不能○
其他收获：		

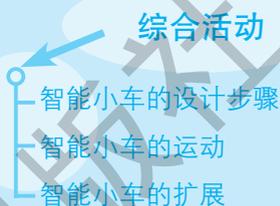
第十六课 综合活动



学习目标

- 了解开源设计的基本过程。
- 能根据实际的问题选择合适的硬件。
- 了解小车运动的基本原理。
- 能根据需要为小车增加硬件。

知识导图



随着社会的发展，我们不仅需要各种硬件设备协助人们的工作，更希望这些设备具备“智能”，这样可以给人们带来更多便利，生活中越来越多的设备具备了智能，例如，智能家居设备，智能车辆等。本课我们将制作一个智能小车，可以通过遥控器控制小车运动，也可以让其按照线路运动。

一、智能小车的设计步骤

如图16-1所示为本课将要制作的智能小车的侧视图，小车运动和转向依靠2个电机带动的轮子、1个万向轮以及硬件设备，全部安装在小车的底盘上，小车运动依靠Arduino板控制，扩展板连接传感器与电机控制模块，可以使用遥控器控制小车运动，通过地面的黑色线条，也可以让小车循迹运动。

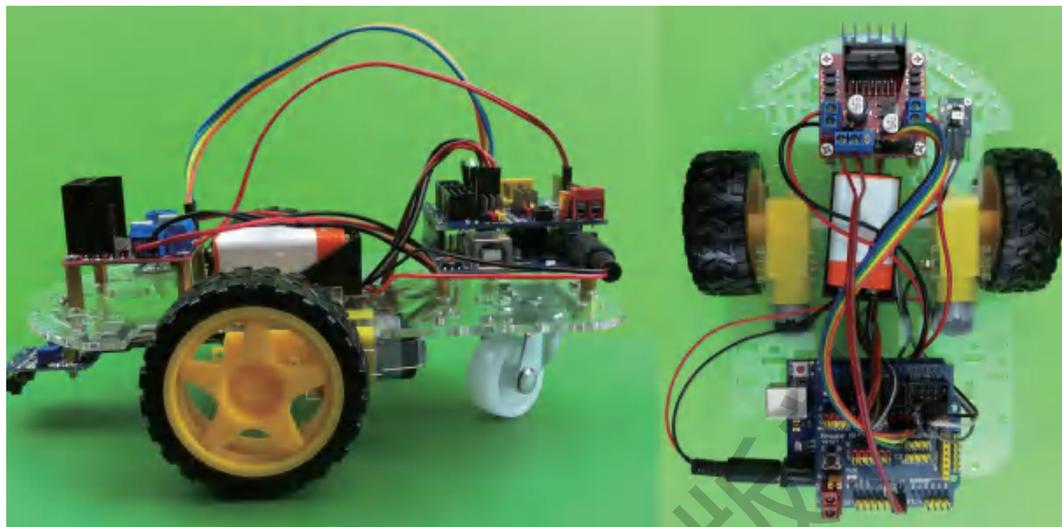


图16-1 智能小车外观图

设计智能小车首先要确定小车要实现的功能，然后选择合适的结构件与硬件，硬件电路测试完毕后，拼装在一起进行测试即可。结构件用来搭建小车的结构，只有电路而没有结构件，也无法实现小车的功能，所以制作小车离不开这些部件。

表16-1 智能小车结构件清单

硬件名	小车底盘	车轮	万向轮	其他结构件
硬件外观				电池与电池盒，六角铜柱，螺丝与螺帽，电机卡扣等。
硬件功能	作为连接各种结构和硬件的基础。	小车依靠车轮转动而运动。	保证小车可以灵活转向，并使小车稳定。	连接各种结构与硬件。
数量	1	2	1	1

表16-2 智能小车硬件清单

硬件名	电机驱动模块	直流减速电机	红外接收器模块	红外遥控器	循迹传感器模块
硬件外观					
硬件功能	控制2个电机的工作模块。	驱动车轮旋转，使小车运动，带连接线的电机会更便于连接。	接收遥控器发出的红外线信号。	发送红外信号。	判断地面黑色轨迹线，并传递信号给Arduino板。
数量	1	2	1	1	2

智能小车设计的基本步骤，也是开源硬件设计的基本步骤，主要包括以下几步骤：

- (1) 确认需求。明确设计物品能够实现的主要功能。
- (2) 规划结构。包括设计基本的外观结构和各种开源硬件连接的结构。
- (3) 软件编程。对连接好的硬件，通过编写程序，测试程序控制的功能。
- (4) 系统测试。安装好硬件与结构件，通过程序测试整体的工作情况，完成需要的功能。

二、智能小车的运动

遥控小车使用遥控器发出红外信号，小车接收红外信号，根据不同的信号，分别控制小车前进、后退、停止、左转与右转。由于小车只有2个车轮，所以各种运动的动作都需要对这两个车轮进行控制。

表16-3 智能小车运动设计清单

效果	左侧电机	右侧电机
前进	正转	正转
右转	正转	停止
左转	停止	正转
后退	反转	反转
停止	停止	停止

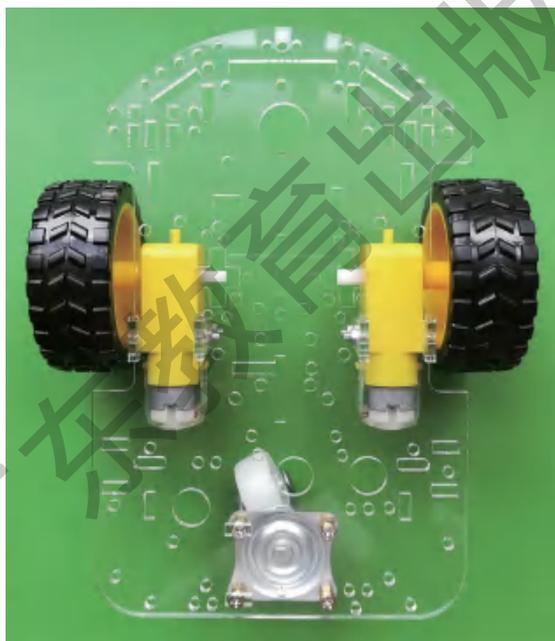


图16-2 智能小车底盘连接

由于小车主要依靠一边的轮子转动实现转向，所以转动的时间决定了小车转向的角度，左转 90° 与右转 90° ，由于电机的差异或者结构件的差异，可能两边的转速不同，所以要通过测试才能确定两边的电机转向 90° 分别需要多长时间。

学习任务

红外遥控小车的制作

通过红外接收模块收到不同的信号，根据这些信号控制电机驱动模块，分别控制2个电机工作，即可实现小车运动的控制。

1. 问题分析

控制小车运动，主要通过L298N电机驱动模块实现，通过数字引脚4、5、6、7控制2个电机的工作，4、5号引脚控制左侧电机，分别输出高电平和低电平电机正转，输出低电平与高电平电机反转。右侧电机同理，2个电机配合就是控制小车的运动，电机控制信号与结果如表16-4所示。

表16-4 电机控制信号与结果

电机	左侧电机			右侧电机		
	数字引脚	数字引脚	转动效果	数字引脚	数字引脚	转动效果
	4	5		6	7	
前进	1	0	正转	1	0	正转
右转	1	0	正转	0	0	停止
左转	0	0	停止	1	0	正转
后退	0	1	反转	0	1	反转
停止	0	0	停止	0	0	停止

2. 硬件连接

先完成硬件电路的连接，首先需要把9V电池通过电池盒接到Arduino板的电源接口上，红外接收器连接3号数字引脚，电机驱动模块连接4、5、6、7号数字引脚，而电机模块电机连接OUT1连接左侧电机的正极，OUT2连接左侧电机的负极，右侧的正负极分别接在OUT3和OUT4接口上（这里需要使用小螺丝刀才能完成操作）。电机驱动模块的12V和GND分别接扩展板的Vin接口和GND接口，这里杜邦线需要去掉黑色胶壳才能连接，智能小车硬件连接图如图16-3所示。

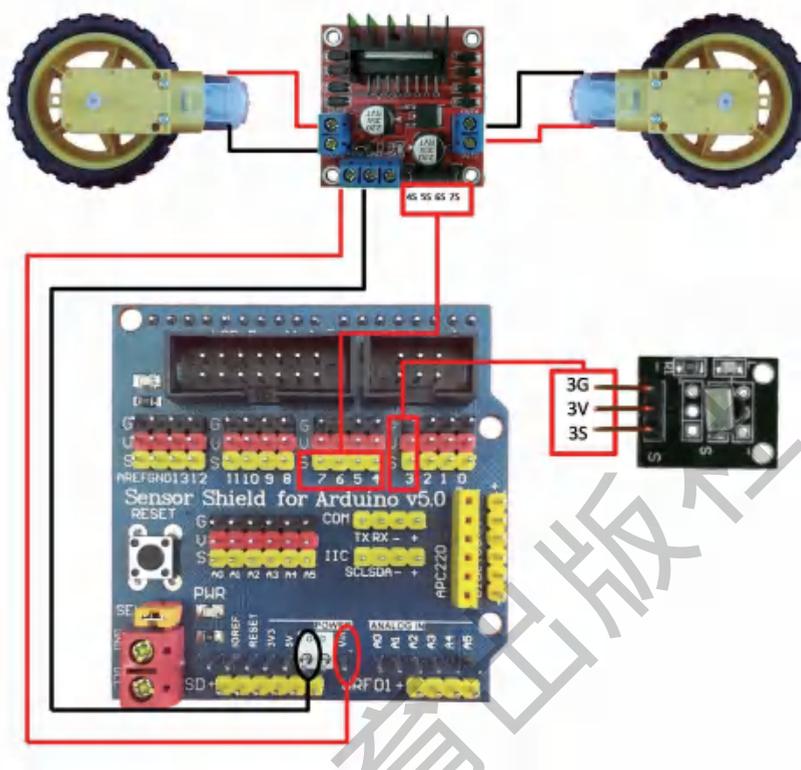


图16-3 智能小车硬件连接图

3. 设计算法

智能小车流程图如图16-4所示。

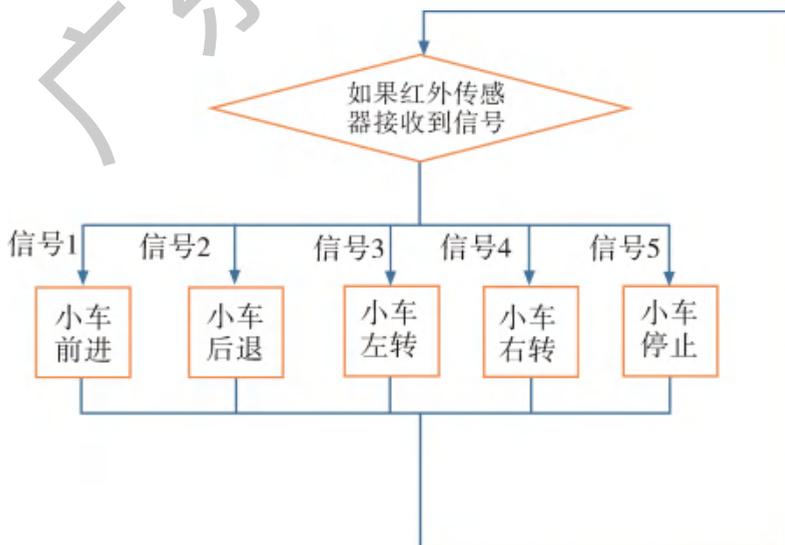


图16-4 智能小车流程图

4. 编写程序

在程序模块选择时，用到的程序模块数量非常多，为了避免程序模块拼接过多导致结构不清，通过创建函数可以使结构更加合理。在函数类型模块中选择不带输出值的函数，总共添加5个到脚本区，然后分别命名为小车运动控制5个动作，然后按照表16-4填入控制数字引脚的模块。添加通讯类型模块中的红外接收模块，然后判断红外线信号，并做出相应的控制动作，期中左转与右转需要延时，以便完成转向动作。五个控制动作对应遥控器上的上、下、左、右、中五个按键，遥控智能小车程序模块图如图16-5所示。

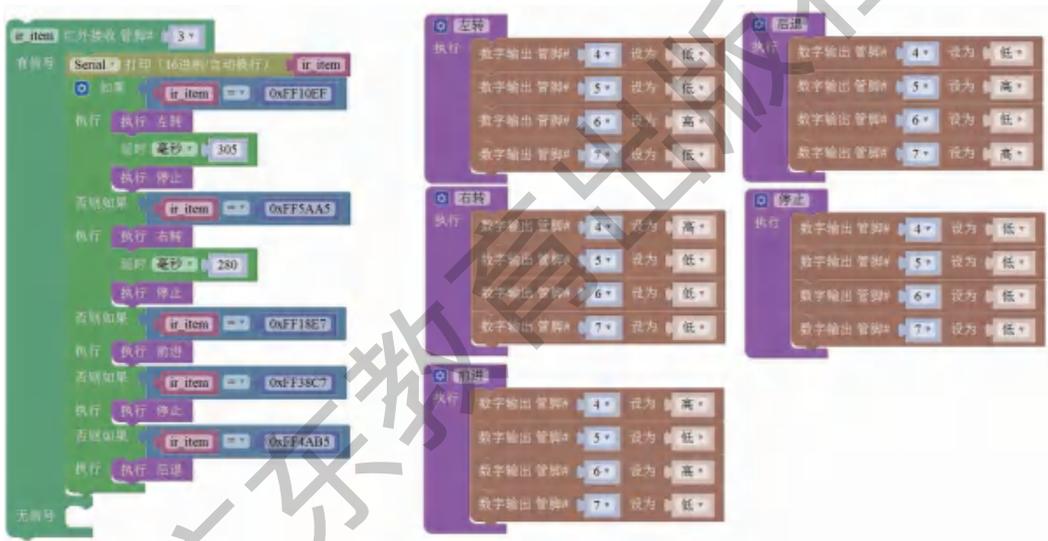


图16-5 遥控智能小车程序模块图

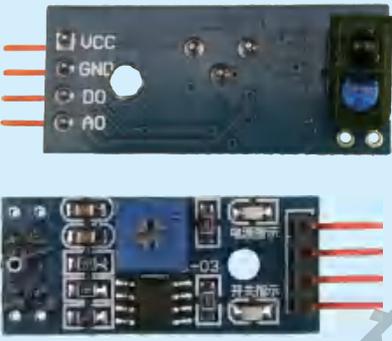
5. 测试运行

通过USB数据线连接Arduino板，选择正确的COM端口号，然后单击“上传”按钮，即可把程序指令发送给Arduino板。打开串口监视器，使用遥控器分别按下控制按钮，检查串口监视器接收到的信息是否正确，小车的车轮是否按照设计转动，测试成功后，将这些硬件安装到小车底盘上，只依靠9V电池供电，再通过遥控器进行测试，左转与右转可根据转向角度的需要修改延时时间。

三、智能小车的扩展

智能小车可以通过遥控器控制运动，也可以让其自动识别地面的线条并根据线条控制小车运动，为了判断地面的线条，需要使用循迹传感器模块。

表16-5 循迹传感器模块功能表

按键开关模块	引脚	引脚功能
	VCC	电源
	GND	接地
	DO	数字信号输出引脚
	AO	模拟信号输出引脚

循迹传感器模块，通过红外线反射来判断地面的颜色，通常在白色背景下绘制黑色线条，如果检测到白色输出数字信号0，如果检测到黑色输出数字信号1，据此判断小车是否偏离轨迹，如果偏离轨迹，做出相应的调整。

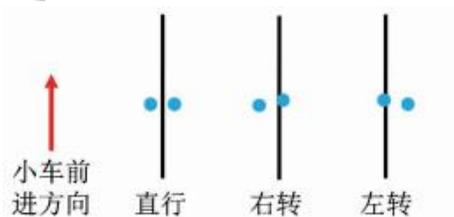


图16-6 智能小车循迹执行原理图

学习任务

循迹智能小车的制作

智能小车根据地面的黑色线条前进，如果左边循迹传感器检测到黑线，就向左转，如果右边循迹传感器检测到黑线，就向右转，如果2个传感器都没检测到黑线，就继续前进，为了保持黑线始终在2个循迹传感器检测范围内，前进和转向的幅度都应小一些。

1. 问题分析

通过对循迹的3种情况判断，控制小车运动，使小车始终沿着黑线前进。

2. 硬件连接

智能循迹小车硬件连接图如图16-7所示。

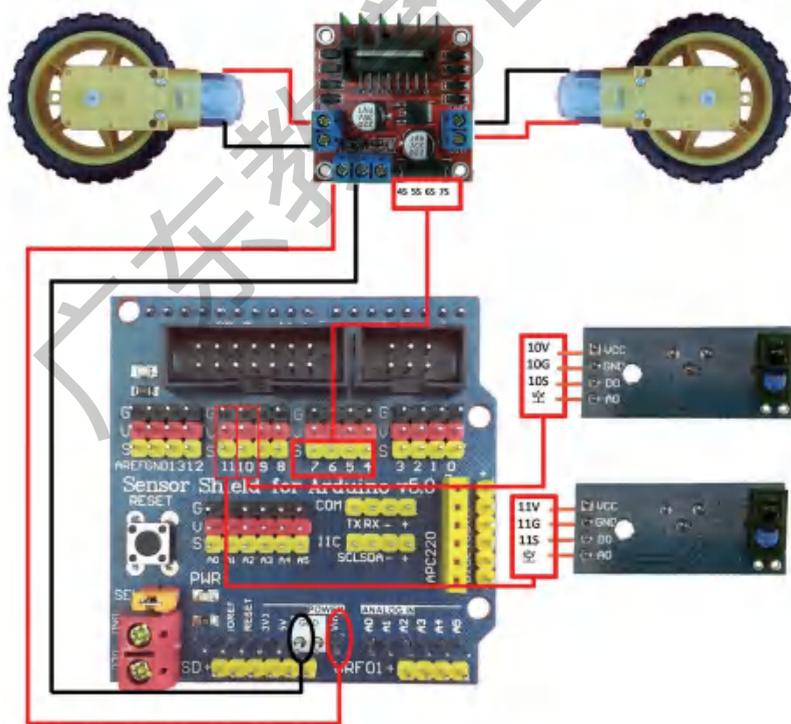


图16-7 智能循迹小车硬件连接图

3. 设计算法

智能循迹小车流程图如图16-8所示

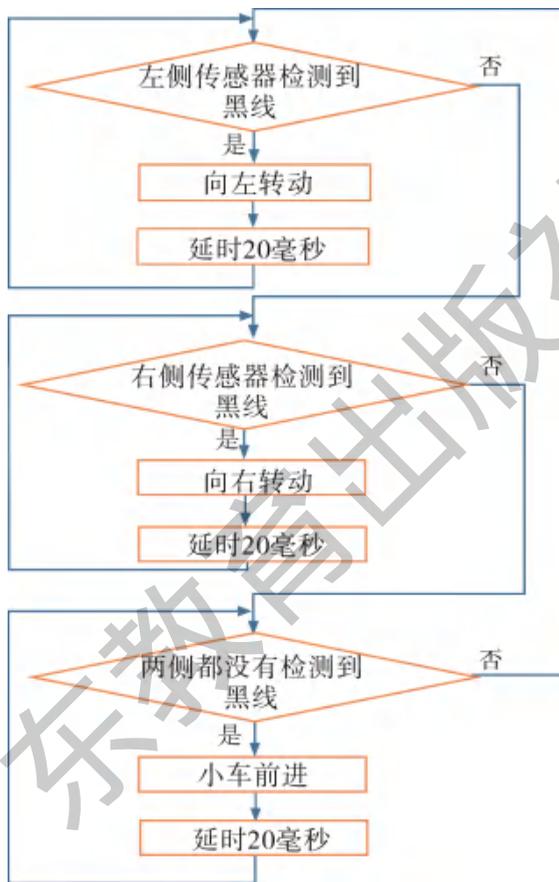


图16-8 智能循迹小车流程图

4. 编写程序

智能循迹小车程序模块图如图16-9所示。



图16-9 智能循迹小车程序模块图

5. 测试运行

上传程序后，把小车放到黑线上，保证黑线在2个循迹传感器之间，小车基本方向沿着直线前进，调整2个循迹传感器的间距稍稍大于黑线的宽度，保证小车可以一直沿着黑线前进。

自主探究

1. 智能循迹小车连接电池后，会一直工作，没有开关控制会不方便，如何增加一个开关便于控制小车的工作，通过开关控制小车是否开始循迹？
2. 智能小车处理可以通过循迹自动运动，也可以遇到障碍物自动避开障碍物，想一想如何设计一个遇到障碍物可以自动躲避的小车。

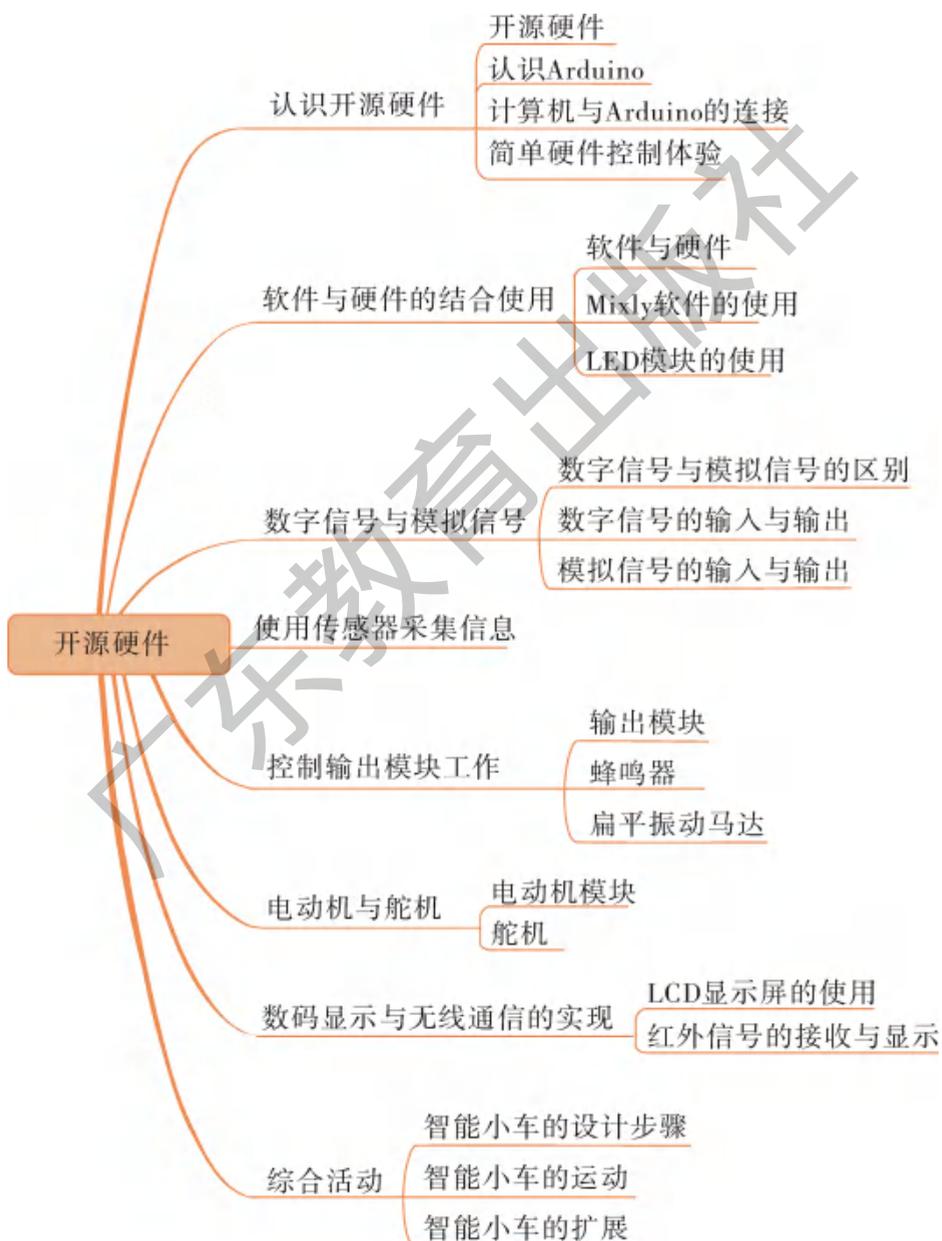
反思评价

对自己在本课中的学习进行评价，对照表16-6，在相应的○里画√，在“其他收获”栏里留言。

表16-6 学习评价表

序号	学习内容	能否掌握
1	了解开源设计的基本过程。	能○ 不能○
2	能根据实际的问题选择合适的硬件。	能○ 不能○
3	了解小车运动的基本原理。	能○ 不能○
4	能根据需要为小车增加硬件。	能○ 不能○
5	能对小车的功能进行改进。	能○ 不能○
其他收获:		

单元扼要回顾



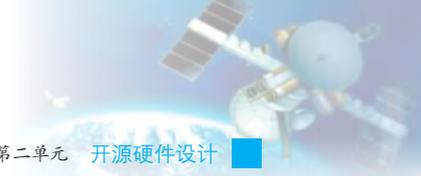
单 元 学 习 评 价

一、学习评价

请根据本单元各环节的学习情况，填写单元学习评价表，从知识与技能、方法与过程以及情感态度与价值观等方面，综合评价自己是否达到了本单元的学习目标。

单元学习评价表

学习目标		是否达到目标	
		是	否
知识与技能	了解开源硬件和软件及其用途。		
	了解Arduino板及其功能。		
	了解Mixly及其操作。		
	懂得各类传感器及其应用。		
过程与方法	懂得Arduino连接计算机和各类传感器的方法。		
	掌握数据信号的收集过程。		
	掌握利用Mixly设计编写程序的方法。		
情感态度与价值观	理解开源文化，懂得如何利用开源硬件和软件进行作品制作。		
	遵守开源文化和信息技术活动中相关的法律法规与伦理道德规范。		
	养成良好的信息活动行为习惯。		



二、学习体会

请从以下几方面谈谈自己在本单元学习过程中的成长体会。

本单元对我启发最大的是： _____

我不太理解的学习内容有： _____

我还学会了： _____

我还想学习： _____

广东教育出版社

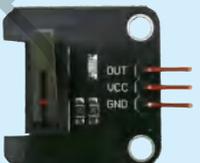
附录一 常用开源硬件说明

序号	实物图	名称	主要功能	使用该硬件的课
1		Arduino UNO控制板	具有数字接口和模拟接口，可连接其他硬件。	1~8
2		Arduino 扩展板	扩展Arduino板的引脚数量，方便连接更多的硬件。	
3		Arduino 扩展板 (彩色引脚)	彩色引脚区分电源、接地、数据接口，确保更准确的连接其他硬件的引脚。	2~8
4		单色LED 模块	可以发光的LED模块，分别有白色、红色、黄色、绿色、蓝色可选。	2、3
5		全彩LED 模块	具有红色、蓝色、绿色的LED模块，并且每种颜色亮度变化可组合成其他颜色。	2、5

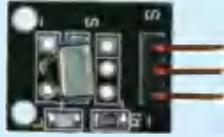
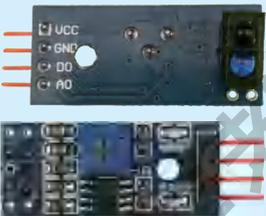
(续表)

序号	实物图	名称	主要功能	使用该硬件的课
6		可调电阻模块	通过滑竿得到不同的输出值。	3
7		按钮模块	可实现数字输出高电平或低电平。	3
8		光敏传感器模块	可以感应光线的强弱，并通过数值的大小体现。蓝色可变电阻可以调节数字信号输出的阈值。	4、5
9		触摸传感器模块	手指触摸感应区域，可将是否触摸转换为电信号。	4
10		倾斜传感器模块	倾斜角度超过 15° 后，模块输出信号会发生变化。	4
11		震动马达模块	可以产生震动效果的模块。	5
12		无源蜂鸣器模块	可以产生声音的模块。	5、6

(续表)

序号	实物图	名称	主要功能	使用该硬件的课
13		L9110 电机驱动 模块	电机可以转动，两个数据输入引脚可以控制风扇的正转与反转，输入模拟信号可以控制电机的转速。配有可安装在电机上的软扇叶小风扇叶片。	6
14		温度传感 器模块	可以检测温度变化并转换为电信号。蓝色可变电阻可以调节数字信号输出的阈值。	6
15		L298N 电机驱动 模块	可以驱动2组电机工作，具有专门的电源输入接口和控制引脚。	6
16		SG90舵机	可以在 0° 和 180° 之间转换角度。	6、7
17		碰撞模块	通过微动开关，可以检测到每次压下的操作。	7
18		1602LCD 液晶显示 屏(I ² C 接口)	可显示2行，每行16个字符。	7

(续表)

序号	实物图	名称	主要功能	使用该硬件的课
19		红外遥控器	发出红外信号。	7、8
20		红外接收模块	接收红外信号。	7、8
21		直流减速电机	最好已经焊接好电源线，红色为正极，黑色为负极。可以带动车轮转动。	8
22		循迹传感器模块	可检测地面的黑色或白色，并输出电信号实现循迹控制。	8
23		HC-SR04超声波传感器模块	超声波传感器可以发出超声波信号，并测出其与正对物体之间的距离。	
24		人体红外感应模块	可以检测人体信号。	
25		DS18B20温度传感器模块	可以检测环境温度并转换为电信号。	

(续表)

26		声音传感器模块	可以检测声音信号。	
27		USB数据连接线	连接Arduino板与电脑。	1~8
28		杜邦线	连接各种硬件，连接电源驱动模块，需要去掉黑色胶壳，并使用螺丝固定。	2~8
29		带电池的电池盒	提供9V电源供电，充电电池可反复使用。	1、8
30		收纳盒	用来装每节课所用到的各种器材。	

以上每类硬件，都有很多种可以选择，只要能实现相应的输出，外观有差异也不影响使用，本单元所有硬件均为实物图，力求直观，但都有替用，无需严格按照参考开展学习实验。

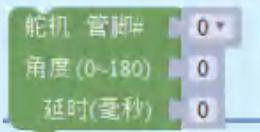
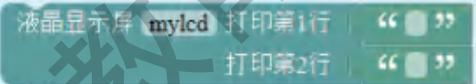
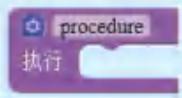
附录二 程序模块说明

序号	模块类型	模块图片	功能
1	输入输出		指定引脚数字输出高电平或低电平。
2	输入输出		指定引脚数字输入。
3	输入输出		指定引脚模拟数据输出。
4	输入输出		指定引脚模拟输入。
5	控制		只在程序开始时执行一次的程序模块。
6	控制		延时，注意单位可选毫秒或微妙。
7	控制		根据条件判断执行，蓝色按钮可以添加“否则”或“否则如果”条件。

(续表)

序号	模块类型	模块图片	功能
8	控制		从 1 到 10 循环, 按照步长为 1 执行, 变量 i 可以循环使用, 也可以修改名字。
9	数字		数字内容。
10	文本		文本内容。
11	逻辑		判断 2 个数值关系, 是否相等、大于、小于等。
12	逻辑		2 个条件的逻辑关系。
13	逻辑		逻辑取反, 非真为假, 非假为真。
14	逻辑		逻辑真与假。
15	串口		显示从 Arduino 板串口。
16	通讯		如果指定引脚接收到红外信号, 执行相应操作。

(续表)

序号	模块类型	模块图片	功能
17	执行器	 舵机 管脚# 0 角度(0~180) 0 延时(毫秒) 0	指定舵机旋转的角度和延时时间。
18	执行器	 播放声音 管脚# 0 频率 NOTE_C3	指定引脚播放声音。
19	执行器	 结束声音 管脚# 0	指定引脚停止播放声音。
20	显示器	 初始化 液晶显示屏 1602 mylcd 设备地址 0x27	设置LCD显示屏的设备地址。
21	显示器	 液晶显示屏 mylcd 打印第1行 “ ” 打印第2行 “ ”	设置LCD液晶显示屏显示的内容。
22	函数	 procedure 执行	定义函数，可以修改函数名称。