目录

[Micro:bit创客与编程 2](#_Toc24801984)

[一、课程概述 2](#_Toc24801985)

[二、课程使用的开源硬件Micro:bit简介 3](#_Toc24801986)

[三、课程设计中的核心理念介绍 4](#_Toc24801987)

[四、单元设计框架 5](#_Toc24801988)

[第一单元 MakeCode与Micro:bit 5](#_Toc24801989)

[第一课 让我们建立规则与顺序 5](#_Toc24801990)

[第二课 不插电计算机科学活动 7](#_Toc24801991)

[第三课：MakeCode和Micro:bit的简单操作 8](#_Toc24801992)

[第四课：小项目 15](#_Toc24801993)

[第二单元 小牛试刀 17](#_Toc24801994)

[项目一 神秘的Micro:bit 17](#_Toc24801995)

[项目二 闪烁的 LED 21](#_Toc24801996)

[项目三 流动的沙子 27](#_Toc24801997)

[项目四 呼吸灯 29](#_Toc24801998)

[第三单元 如虎添翼 35](#_Toc24801999)

[项目一：变速风扇 35](#_Toc24802000)

[项目二 音乐盒 42](#_Toc24802001)

[项目三 炫彩灯带 46](#_Toc24802002)

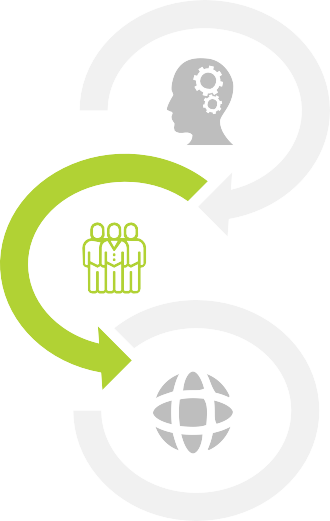
[第四单元 大展拳脚 55](#_Toc24802003)

# Micro:bit创客与编程

## 一、课程概述

**1、人工智能普及与教育**

随着信息时代飞速发展，特别是以人工智能为主的超新科技不断发展，对教育产生了重大的影响。国家出台的一系列政策，也是着眼于未来，我们如何通过教育，使现在的学生能够适应未来的社会变革带来的冲击。2017年7月，国务院颁布的《新一代人工智能发展规划》中指出，逐步开展全民智能教育项目，在中小学阶段设置人工智能相关课程、逐步推广编程教育，鼓励社会力量参与寓教于乐的编程教学软件、游戏的开发和推广。2018年4月教育部《教育信息化 2.0行动计划》中指出“完善课程方案和课程标准，充实适应信息时代、智能时代发展需要的人工智能和编程课程内容。2017年10月教育部《中小学综合实践活动课程指导纲要》指出，在中小学各学段的目标中均包含运用信息技术解决实际问题的描述。2018年1月教育部《普通高中课程标准（2017年版）》把信息技术学习内容分为必修、选择性必修和选修三类，包括“数据与信息”“数据处理与应用”“算法与程序实现”三部分内容。



**实现人才储备**

**基础教育阶段人工智能普及教育的意义**

**培养科学素养**

• 理性思维

• 批判质疑

• 勇于探究

• 扩大人工智能相关人才基数

• 形成多层次的阶梯化人才体系

**增强国家竞争力**

• 促进社会经济发展

• 提升国家竞争力和国际地位

人才培养是系统性工程，从基础教育阶段就有必要引入人工智能普及教育，从小培养学生适应产业革命的科学素养，形成必要的知识储备，激发学生对人工智能领域的学习兴趣，与高等教育阶段的人工智能教育形成有效衔接，构建多层次、阶梯化的人才体系。

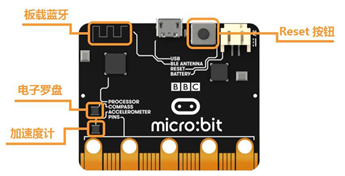
**2、本课程学习内容与方法**

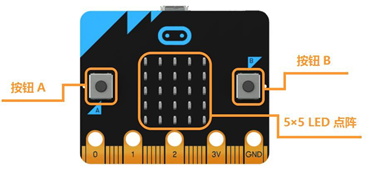
本课程为小学阶段三至五年级中，对信息技术与创客活动有兴趣的学生而设计。适合学校提高型兴趣小组，开展拓展型与探究型课程。在学习过程中，注重学生4C能力的培养，即：批判性思维、沟通能力、团队合作能力、创造性解决问题能力。

在课程教学中，不同于以往的课程模式，我们可以尝试使用全新的教学形式，例如PBL、设计思维等，注重学生对于真实场景的观察与分析，注重真实问题的解决能力，注重创新创造出真实产品，注重团队合作与表达能力。

## 二、课程使用的开源硬件Micro:bit简介

micro:bit 是一款由英国 BBC 设计的 ARM 架构的单片机，板载蓝牙，加速度计，电子罗盘，三个按 钮，5×5LED 点整，主要用于青少年的编程教育。利用 micro:bit 可以实现任何酷炫的小发明，无论是机器人还是乐器。micro:bit 拥有一系列新颖的功能，例如 25 个可显示消息的红色 LED 灯；有两个可编程按钮，可以用于控制游戏操作或者暂停/播放 一首音乐。micro:bit 可以检测动作并且告知使用者动作进行的方向，同时它也可以通过低功耗蓝牙模块与其它设备或因特网互联。





micro:bit 还自带光照度传感器和温度传感器等常见的感应装置，这意味着它自己本身也可以制作很多生活中常见的智能化产品。

## 三、课程设计中的核心理念介绍

* 计算机科学
* 设计思维
* 计算思维
* Steam与跨学科

**学习方法：**

**讨论方法：**

1、 圆桌会议（一种带有目的的自由讨论，常用于小组设计阶段，主要是认识周围学习伙伴对于同一事物的认识和各自观点的差异，在设计思维活动中会引发同理心思考）

2、 金鱼缸会议（为展示自己而进行的一个讨论会议，展示者将自己最好的一面展示出来，观众必须指出自己最欣赏的部分，可以提出建议性的观点）

3、 智囊团会议（为一些创造性想法提出而进行的头脑风暴）

**授课方法：**

1、 主题讲解（讲座形式，介绍一些知识点内容）

2、 自由探索（给出核心概念，并且在核心概念的理解上，进行自由操作，可以给出具体的内容和操作目标，让学生自己尝试自己的解决方法）

3、制作展示（组织集体展示活动，每个学生都需要制作展板、演讲，展示自己的作品和设计意图等）

**思维工具：**

1、 设计思维（项目的制作遵从设计思维的同理心、明确问题、构思创意、原型实践、验证调整来进行）

2、 计算思维（分为分解问题、模式识别、抽象、算法开发五个步骤，通过项目学习使用到计算思维的概念和方法）

## 四、单元设计框架

一学期由几个单元组成，在这些单元里，通过与真实世界的链接，让学生了解信息技术为我们的生活带来的便利与乐趣。通过可编程智能硬件学习，让学生掌握创客技能、计算思维与信息素养。通过项目制学习的形式，提升学生综合知识的运用能力以及跨学科学习带来的学习能力提升。

单元学习基本框架

1、麦克的故事

2、编程学习

3、能力提升（展示交流）

# 第一单元 MakeCode与Micro:bit

## 第一课 让我们建立规则与顺序

程序是由许多驱动硬件的指令组成，这节课，我们将开始进行编程与驱动硬件的学习。通过一个不插电活动，让学生进入学习的状态，感受指令驱动硬件执行的乐趣，哪怕驱动的是他的同学。

**一、课程目标**

通过这一课的学习，学生将能够

1. 理解程序是一个个指令组成的序列；
2. 应用makecode进行简单的程序编程，并且驱动led实现简单的动画；
3. 建立自己的学习小组，建立学习小组的学习制度；
4. 进行一次有效的团队讨论会。

**二、课程结构**

第一课建议2课时（每课时建议时长1小时），如果每课时是40分钟，建议分成3次。

1. 认识自己的学习小组；
2. 不插电活动：挑战挖宝藏
3. Micro:bit的认识和操作的学习；
4. 设计团队动态徽章

材料清单：略

**第一部分：认识自己的学习小组**

根据自己学校的情况，将所有的学生进行分组。这种分组可以是随意性的，也可以有组织性的。以下是几种选组的建议：

1. 教师先以报名的形式，挑选出各个小组的组长，然后由组长挑选组员或者让学生自己报名参加哪个组；
2. 根据年龄或者能力层次，进行分组；
3. 以班级为单位进行分组。让熟悉的孩子在一个小组，有利于今后一些课后项目的推进；
4. 每个小组建议是不超过3人-5人；

**小组圆桌会议**，确定小组的组名和吉祥物。

教师需要先介绍圆桌会议的规则，可以参考罗伯特议事制度，建议介绍并让学生遵守以下规则：

1. 举手发言，由会议主持人选择发言优先顺序；
2. 提出自己的议点，其他人可以同意、反对，并进行讨论；
3. 主持人可以通过表决形式来确定某个议点的通过；

在创客活动中，会由许多集体合作项目，而一个团队进行有效合作，小组的讨论和有效沟通是非常关键和必要的，因此在课程实施中，需要反复练习团队会议。

**展示自己的团队**：各个小组由队长上来展示自己的小组。

**建议**：

各个团队都有一本记录本，记录团队的会议情况。一些表决的内容，一些小组规则等。

## 第二课 不插电计算机科学活动

这个活动需要用到方格地砖的空地，建议10\*10左右的空间，设定好范围和起点位置和方向。一块地砖作为一个移动单元，随机在里面设置一些障碍物，例如书本、三角锥、球、或者任何什么东西。当然，某一些物品里面是藏有“宝物”的。教师需要制作这样的迷宫寻宝图。每个组的宝藏埋藏的位置都是不一样的。

每个小组都有一个“机器人”，只能接受指令和遵守指令去执行挖宝行动。其余成员为程序员，根据地图编辑指令。机器人和程序员不允许正常语言交流，教师事先可以将他们隔离开来。

**活动流程**：

1. 机器人设计一个开始按钮，例如按一下鼻子就是开始。
2. 程序员拿到地图后，有两分钟的程序编辑时间，程序员会用符号或者简单文字，做成一条条指令，程序条尽可能大，会在活动结束后进行公开展示；
3. 教师收走程序纸条，并且让机器人依次站在起点，并且阅读指令，并且让程序员按他的“开始”按钮开始。教师对比设计图，评判是否正确挖取了宝藏；
4. 机器人行动过程中，所有程序员都不允许遥控机器人。所有行动都是机器人对程序的解读；
5. 成功的小组将得到小组奖励。

**反思**：

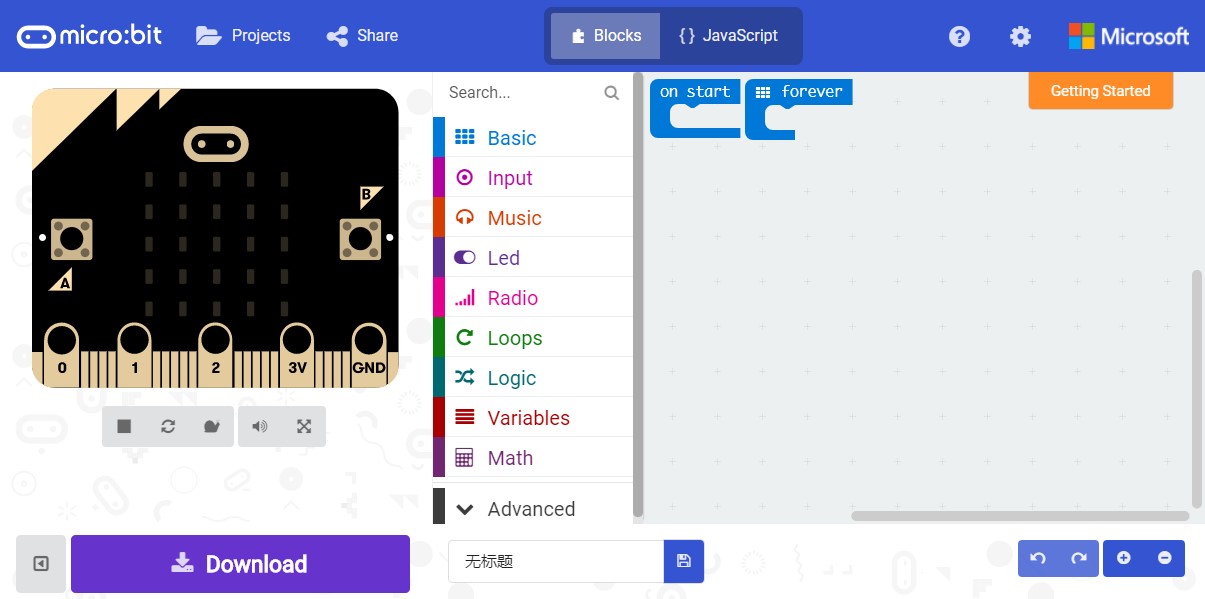
活动完成后，教师展示这些程序指令条。集体讨论并反思以下问题：

1. 失败的小组，失败的地方在哪里？“机器人”是否误解了程序员的指令或者根本就没看懂这些指令？
2. 程序员和机器人之间，是通过什么来沟通的？沟通成功的关键是什么？
3. 真实世界中，机器人是如何理解程序员的指令的？通过什么?
4. 程序是什么？

## 第三课：MakeCode和Micro:bit的简单操作

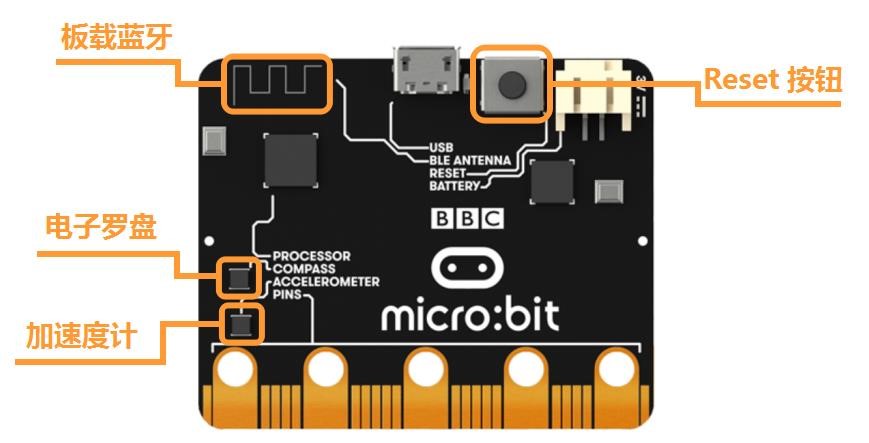
**一、MakeCode 平台介绍**

MakeCode for micro:bit 是 micro:bit 官网上使用最为广泛的图形化编程环境，它基于微软开源项目 MakeCode 开发的图形化编程环境。

 **二、micro:bit 简介**

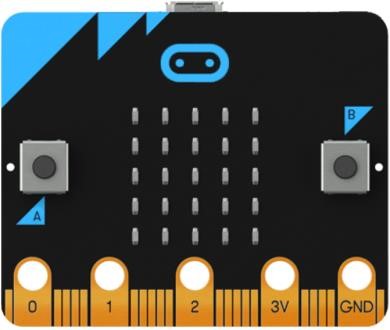
Micro:bit 是一款由英国 BBC 设计的 ARM 架构的单片机，板载蓝牙，加速度计，电子罗盘，三个按钮，5×5LED 点整，主要用于青少年的编程教育。

利用 micro:bit 可以实现任何酷炫的小发明，无论是机器人还是乐器。micro:bit 拥有一系列新颖的功能，例如 25 个可显示消息的红色 LED 灯；有两个可编程按钮，可以用于控制游戏操作或者暂停/播放一首音乐。micro:bit 可以检测动作并且告知使用者动作进行的方向，同时它也可以通过低功耗蓝牙模块与其它设备或因特网互联。





Micro:bit 还自带光照度传感器和温度传感器等常见的感应装置，这意味着它自己本身也可以制作很多生活中常见的智能化产品。如果你是第一次接触 Micro:bit，同样可以通过以下零基础教程学习如何使用 MakeCode 在线编程平台，以及如何为Micro:bit 编写程序。开始之前，请确认你手边有如下物品：



Micro:bit 主板 microUSB 连接线

除此之外，你还需要一台运行 Windows/Mac OS/Linux 操作系统并且有网络连接的电脑。

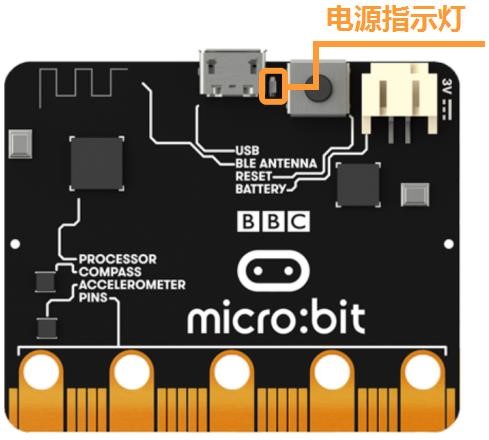
**STEP1: 打开 MakeCode 在线编程平台**（以下的步骤说明基于 Windows 操作系统，如果你使用的是其他操作系统，可以将其作为参考。）

需要登录网页： https://makecode.microbit.org/#

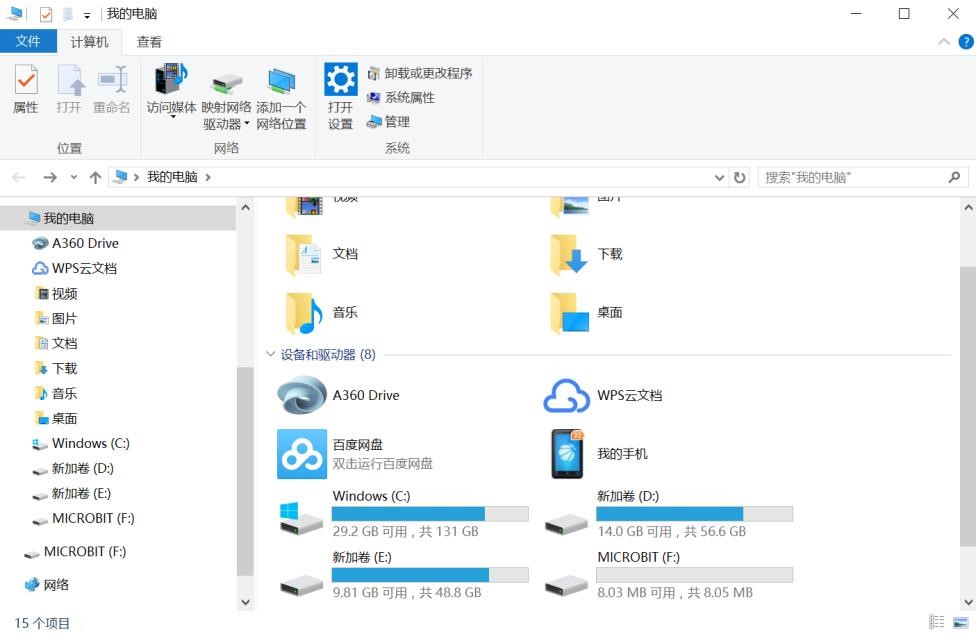


**STEP2: 连接 micro:bit 至电脑**。

将 micro:bit主板通过 MicroUSB 数据线连接至电脑。正确连接时 micro:bit 主板背面的电源指示灯会亮起。

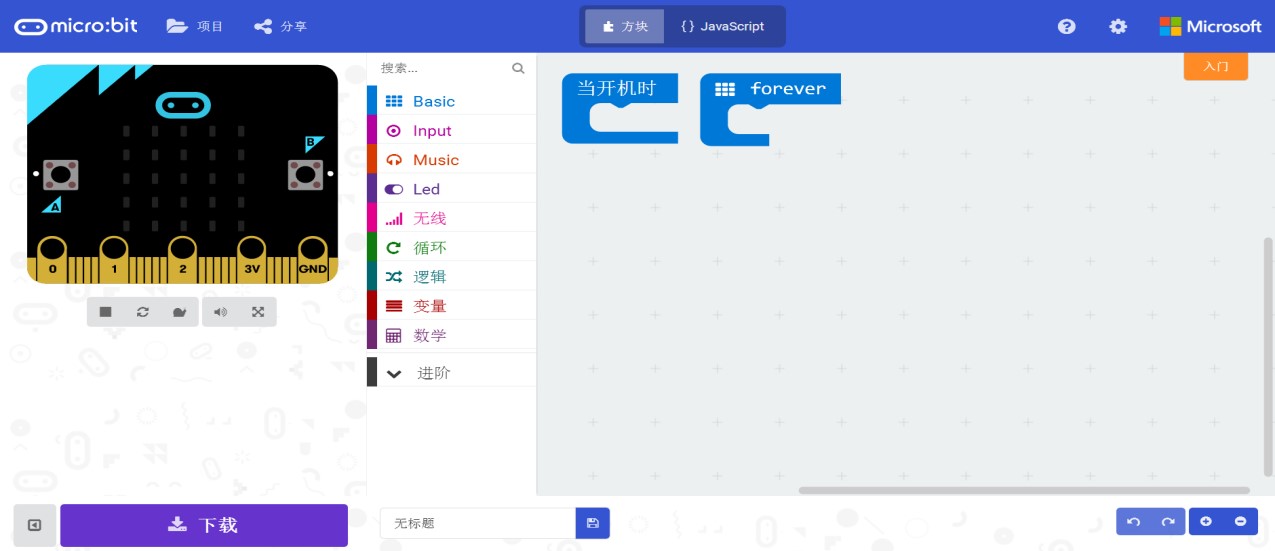


在编程之前，我们需要确保主板被电脑识别。 打开“我的电脑”。接上 micro:bit 主板的就会在磁盘列表中显示磁盘MICROBIT。



**STEP3: 新建项目**

现在让我们仔细看下 makecode 在线编程界面。



模拟

窗口



指令区



脚本区



设置



命名

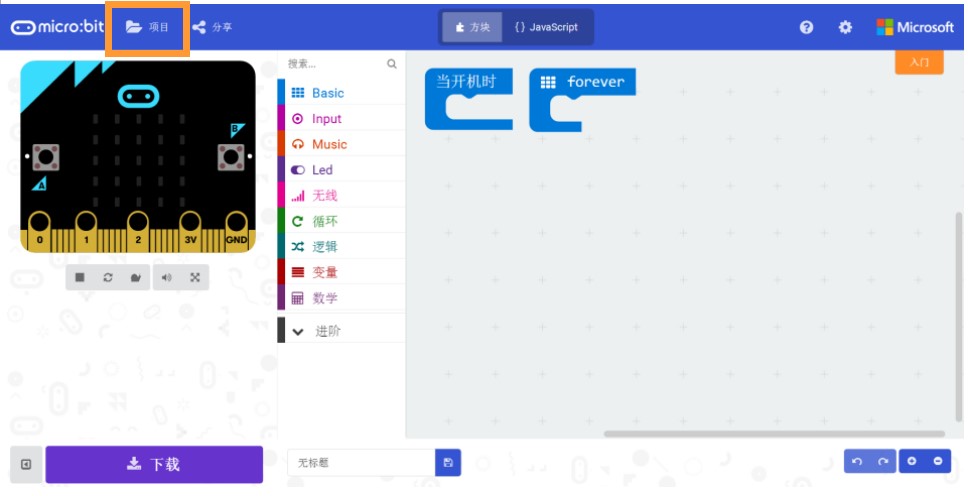
**模拟窗口：**模拟 micro:bit 工作的状态，在编程过程中，可以通过观察模拟窗口，观看初步的运行结果。

**指令区：**拖动指令区的指令可以对 micro:bit 进行编程控制。

**脚本区：**程序的编写区域，拖拽指令区的指令在此编写程序。

**设置：**常用功能为切换语言 。

单击模拟窗口正上方的“项目”。在跳转出来的页面单击“新建项目”即可。





**STEP4: 下载程序至 micro:bit**

程序编写完成之后要下载到 micro:bit 主板里面，点击指令区正下方的“无标题”框，给程序命名。



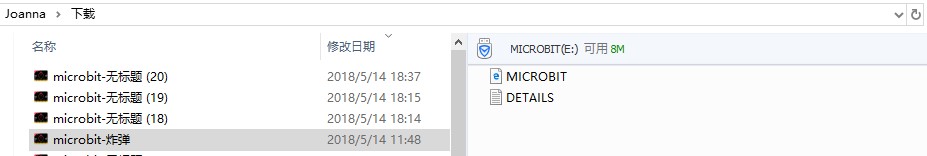
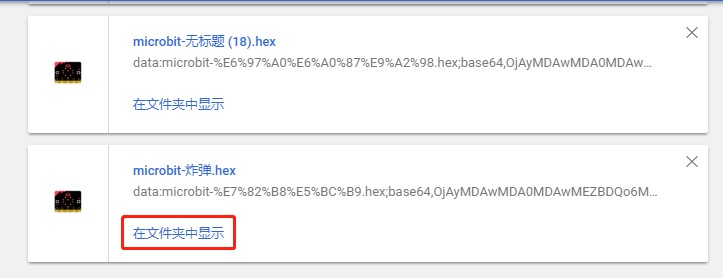
在下载之前，可以在模拟窗口看到程序运行之后的模拟动作，通过模拟窗口下方的按钮可以控制模拟器。

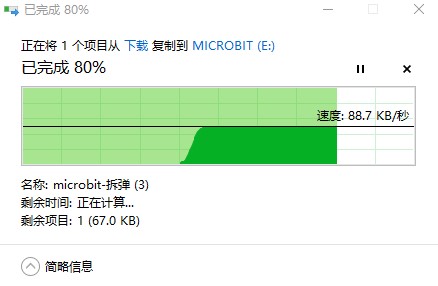
 点击模拟窗口正下方的“下载”，弹出的对话框下选择“另存为”。



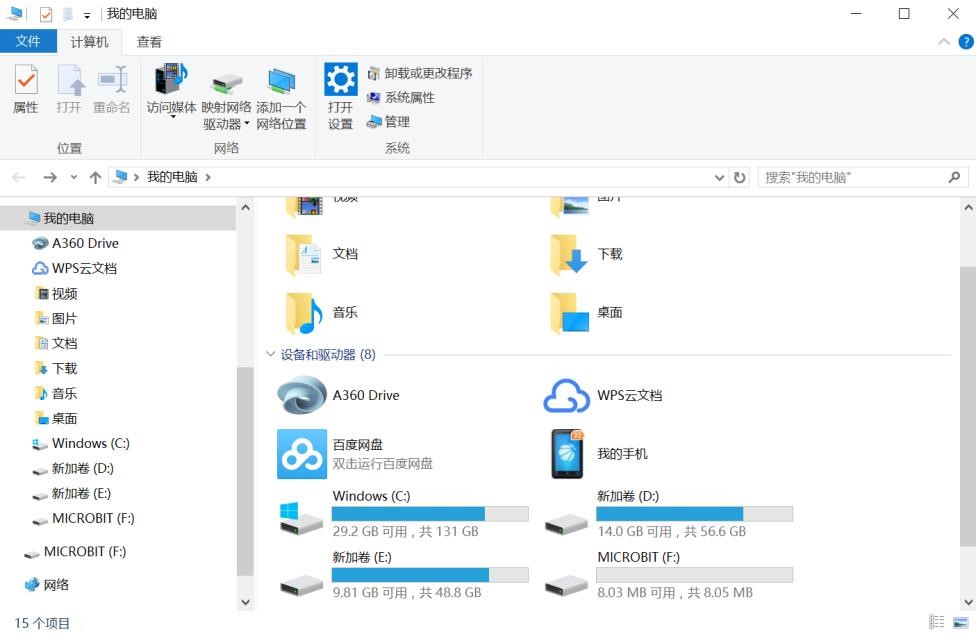
在弹出的对话框中，选择把.hex 文件保存到“MICROBIT 磁盘”，再点击“保存”。

如果没有出现另存为，需要你找到下载的.hex 文件，复制或者拖入到“MICROBIT 磁盘”。





下载过程中 micro:bit 主板背面的电源信号灯会闪烁，当下载完成后电源信号灯停止闪烁，保持长亮。



**三、Micro:bit 扩展板与 Micro:bit 的结合**

**（一）Micro:bit 的使用限制**

Micro:bit 需要通过金手指插槽来连接更多外部设备。因其自身的驱动能力较差，无法直接驱动电机，舵机等大电流设备。

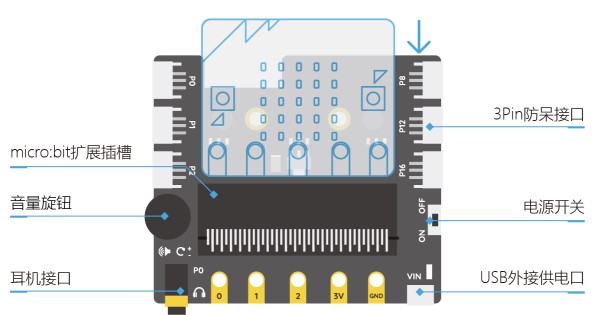
另外，micro:bit 本身的供电和信号电压为 3.3V，无法直接和常用的 5V 电子元件一起使用。

**（二）Micro:bit 扩展板的好处**

1. micro:bit Boson 扩展板能轻松将 micro:bit 额外的引脚引出，即插即用，简单快捷；
2. 扩展接口，连接稳定性更强；
3. 外接的 USB 供电口 VIN 能为外部元件提供额外的供电，更好地支持灯带，舵机等大电流设备；

（程序下载时，还是需要连接在 Micro:bit 上）

1. 扩展板还附带了 3.5mm 耳机接口和音量旋钮，方便耳机直插。



扩展板最大额定电流为 500mA，请在使用前确定元件的最大功率。请勿将扩展板与大型舵机一起使用，可能引起板载元件损坏！

## 第四课：小项目

这是第一个项目，各个小组将制作一个自己小组的特有动态徽章。动态徽章的设计，需要小组成员进行有效沟通和讨论达成一致。徽章需要能够固定在身体上，教师需要提醒学生，如何保护micro:bit板和电池。

1. 教师提出小项目要求，这个要求最好是能够打印，并且贴在小组记录本上。
   1. 制作小组动态徽章；能够和小组名字或者吉祥物有对应；
   2. 徽章能够固定在衣服上，并且可以展示；
   3. 需要小组成员通过会议达成一致；
   4. 需要制作一些小装置，能够将micro:bit和电池固定在一起，并且固定在衣服上。
2. 组织各个小组进行会议，并且开始进行制作；
3. 做一个展示活动，让各个小组进行公开展示和宣讲；
4. 可以组织一次评选活动，给高分小组奖励。

**活动建议：**

1. 作为第一个项目，教师关注点在于有效的团队沟通和合作，防止组长有过多的决策权；
2. 每个组员都有机会编写自己的程序，虽然程序是相同的。
3. 教师事先可以准备一些手工材料，甚至一个徽章固定的框架。但是徽章正面可以让孩子们自己设计和制作。让学生直到和看到你准备的材料，让学生们可以自己选择。
4. 任何小组行动，希望都是先确认步骤和方法，再去行动。

**本课反思**

让学生写一个约200-300个字的反思，包括以下内容：

1. 用你的理解，描述：程序是什么？程序能够用在哪些地方？
2. Makecode的编程和你之前学过的其他编程有哪些区别？如果你之前没有学过编程，可以跳过。
3. 在游戏环节，你如何理解程序员和机器人之间的关系？
4. 小组合作中，你为小组合作主要有哪些贡献？你更愿意在哪些方面为小组合作提供你的贡献？

# 第二单元 小牛试刀

经过前面的知识介绍，你是不是跃跃欲试了呢？现在，我们就开始 Boson for micro:bit 的神奇之旅吧！

## 项目一 神秘的Micro:bit

**一、麦克的故事**

麦克会在周末的时候会跟小朋友用网络沟通，常常在手机上或者电脑上面会见到一些萌萌的，很可爱的表情包，比如这样的：

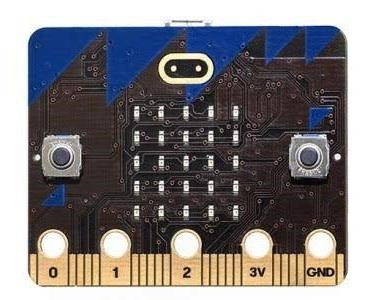


麦克问小米老师，可不可以自己设计表情包，随时随地给其他小朋友展示呢？

小米老师说：带你认识一位新的小朋友吧，它的名字叫 micro:bit。当看到这个小小的很精致的 micro:bit 的小板子的时候，麦克很好奇，这个小板子真的可以做出来表情包么？麦克已经迫不及待的动手制作表情包啦，接下来，由小米老师带着大家一起玩起来。

**二、所需元件**

1.1× micro:bit 主控板



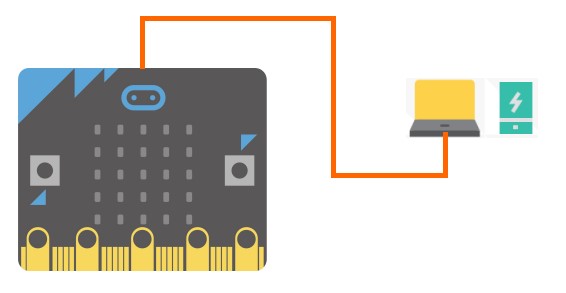
1× MicroUSB 连接线



**三、实现功能**

micro:bit 板子上面有 5\*5（25 个）板载的小灯，我们可以通过控制小灯的亮暗，制作小灯亮出不同的造型，展示表情包的效果。

**四、硬件连接**



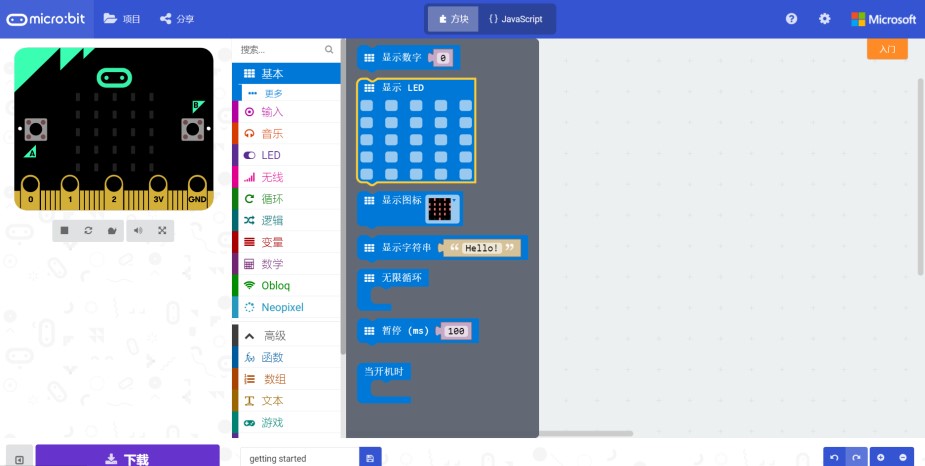
**五、编程学习**

1.将 micro:bit 主控板用 USB 连接线连接在电脑上。

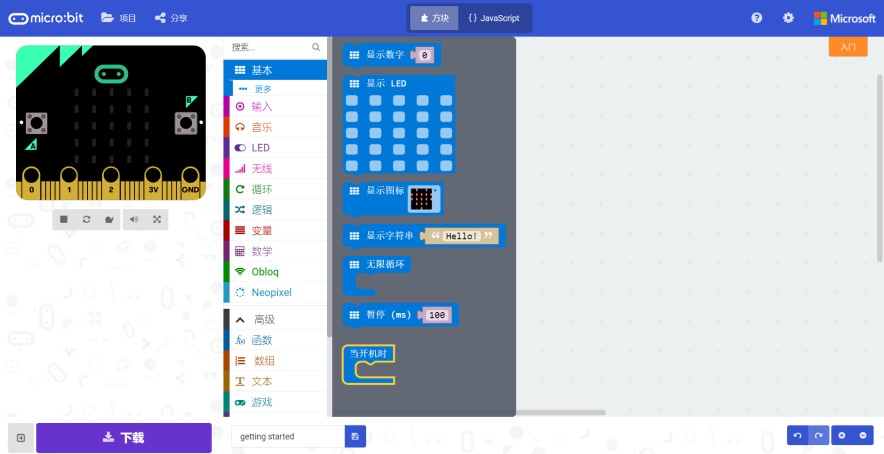
2.打开 MakeCode 在线编程界面。

STEP1：新建项目。

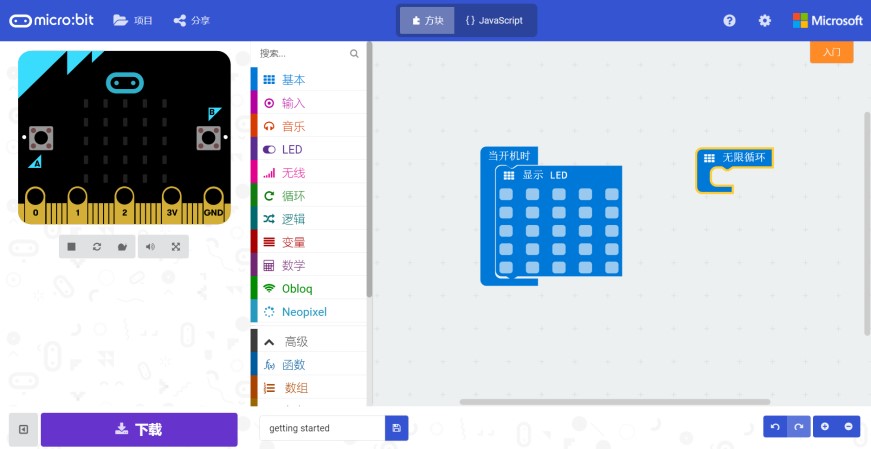
STEP2：单击指令区最上面的“基本”，在指令区的右侧会出现一排模块，在右侧模块中找到“显示LED”模块。



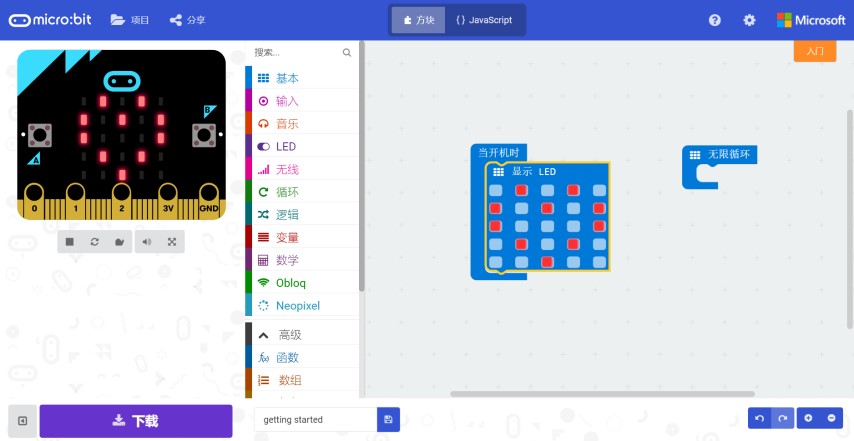
STEP3：打开 MakeCode 在线网站之后，界面上会自动出现“当开机时”模块，或者在“基本”右侧模块的最底部可以找到“当开机时”模块，将“当开机时”模块拖至脚本区。



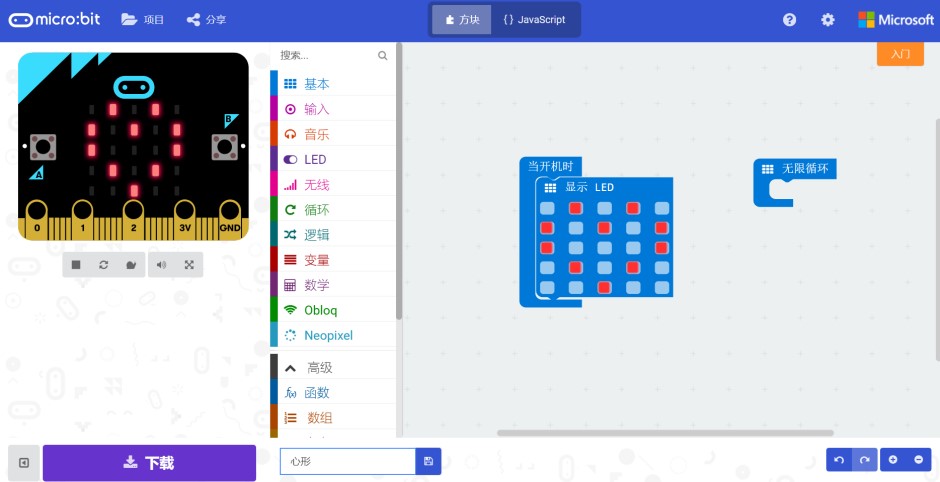
STEP4：将“显示 LED”模块拖动至脚本区，并放入“当开机时”模块内。



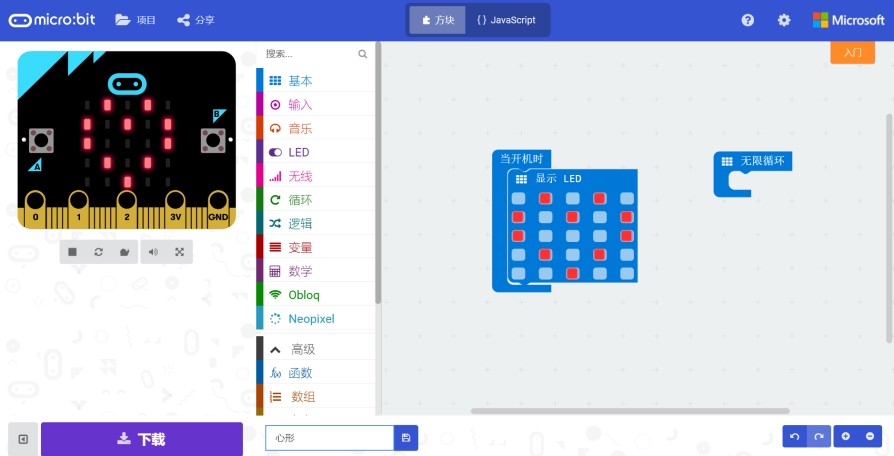
在脚本区可以看到已经拖动的“显示 LED”模块，单击浅蓝色的小方块，可以开始设计表情包，首先，先来绘制一个萌萌的“心型”，可以在左边的模拟窗口可以看到绘制的模拟效果。



STEP5：在下方红框内，编辑为“心形”（可以自己命名）。



STEP6：单击“下载”，将文件保存在电脑上“micro:bit”硬盘上，即可完成保存程序，下载至micro:bit 主板。这样，micro:bit 板子上就会显示“心形”。（具体下载步骤课参考第一话“初次使用 micro:bit”的 STEP4）



**六、能力提升与展示交流**

试着制作可以切换两个表情的表情包，并在班级内进行展示交流。

## 项目二 闪烁的 LED

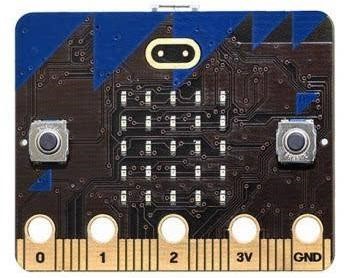
**一、麦克的故事**



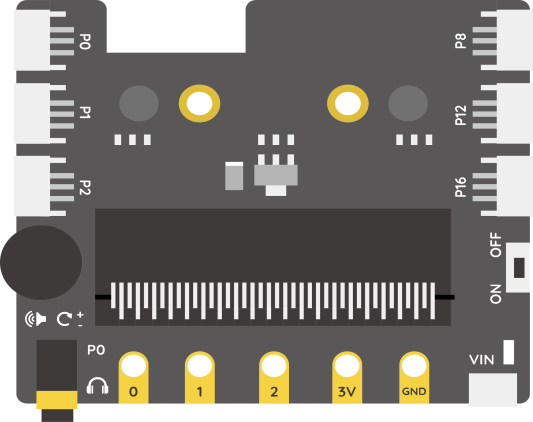
一周的时间很快就过去啦。在周五放学的时候，麦克收到了新朋友 micro:bit 的邀请去家中做客， micro:bit 来自于遥远的 DF 星球。为了感谢好朋友的盛情邀约，麦克决定在出发前一天做一个小礼物送给 micro:bit，到底做什么好呢？麦克不由得纠结了起来...麦克看着大街上五颜六色的小彩灯，心中突然有了灵感。不如就做一个闪闪发光的星星灯吧。

**二、所需元件**

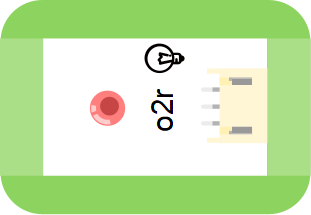
1.1× micro:bit 主控板



2.1× micro:bit 扩展板



3.1× LED 灯模块



4.1× MicroUSB 连接线



**三、完成任务一：点亮 LED 灯**

**1.实现功能**

当在Makecode 在线网站编写程序后，烧录至 micro:bit 小板子，点亮连接在 micro:bit 小板子上的外接 LED 灯。

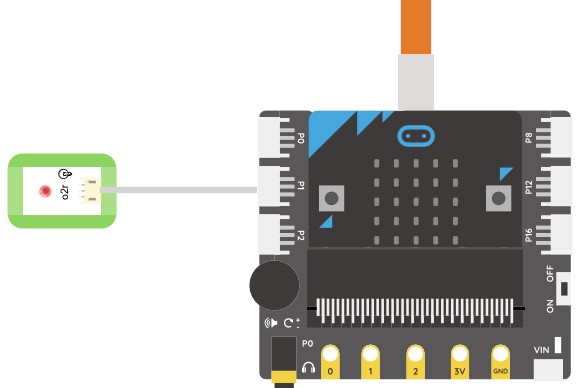
**2.硬件连接**

我们需要连接以下部分：

micro:bit 主控板与 micro:bit 扩展板连接



LED灯模块连接到扩展板的P1号接口



**3.编程学习：打开 MakeCode 在线编程界面。**

**STEP1：**新建项目——点亮 LED 灯，具体操作可参考第一话“初次使用 micro:bit”的 STEP3。

**STEP2：**硬件连接中运用到了外接 LED 小灯，编程过程中，通过设置 micro:bit 的引脚值（连接 LED 小灯的引脚）来控制小灯的亮暗。控制 LED 小灯呈现“亮”或“暗”两种状态。

LED 小灯在这个实验中属于**数字输出**，单击“高级”，会出现“引脚”，调用指令的流程为：引脚（高级指令）---- “向 引脚 P0 数字写入值 0”。（拖动后，将 P0 改为 P1）



**STEP3：**将其指令拖动入“无限循环”中，当烧录代码至 micro:bit 小板子时，可以点亮连接在 P1 号引脚的外接 LED 小灯。（拖动后，将 P0 改为 P1）引脚值只有 0 和 1 两种情况，当值为 1 时，表示小灯被点亮；当值为 0 时，表示小灯呈现熄灭状态。



设置引脚为高电平，点亮

**LED**

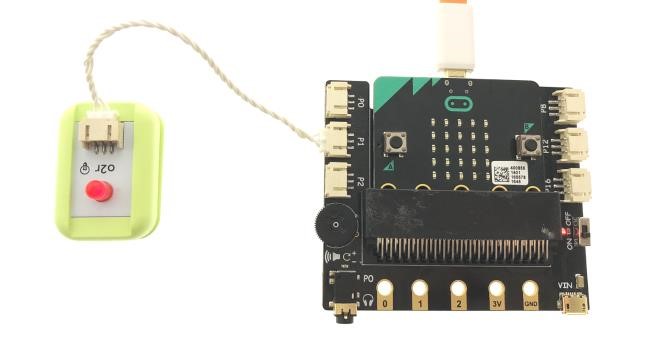
设置引脚为低电平，熄灭

**LED**

最终程序，下载到 micro:bit 板子中。



**4.实现效果**



常亮

**四、完成任务二：闪烁 LED 灯**

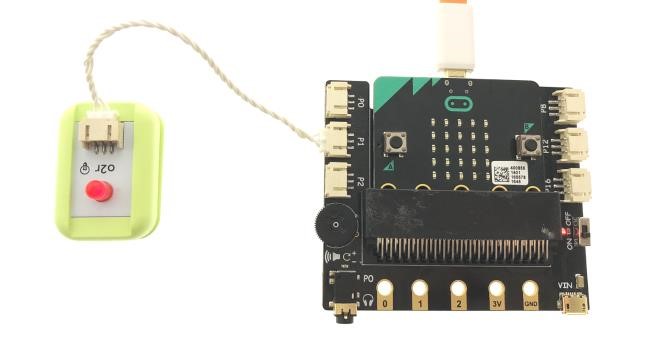
**1.实现功能**

当在 Makecode 在线网站编写程序后，烧录至micro:bit小板子，外接LED小灯呈现亮一秒，暗一秒的闪烁效果。

**2.编写程序**

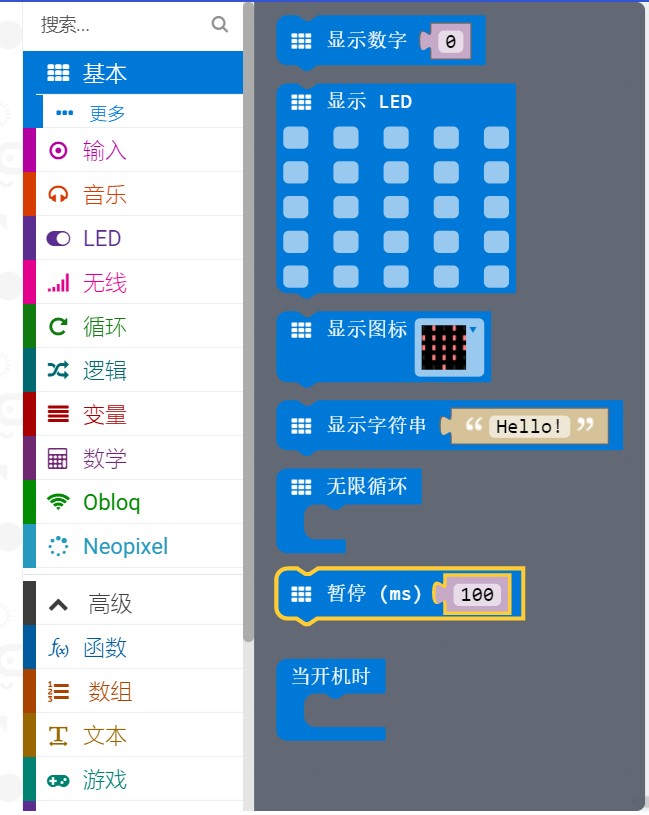
STEP1：新建项目——闪烁LED灯，具体操作可参考第一话“初次使用 micro:bit”的STEP3。

STEP2：设置 micro:bit 的引脚值来控制小灯的亮暗，控制 LED 小灯呈现“亮”或“暗”两种状态。



STEP3：怎么实现闪烁呢，需要让外接 LED 小灯保持亮的状态 1 秒，保持熄灭的状态 1 秒，呈现亮 1 秒，暗 1 秒相交替的状态，需要调用指令的流程为：基本-----暂停（ms）（100）。

“暂停”指令可以通过改变后面的参数，控制 LED 小灯呈现亮或灭的状态保持一定时间。



STEP4：将“暂停”指令拖入“无限循环”中，调整数字后，最终代码见下图。表示：P1 引脚为高电平（点亮状态）保持 1000ms，P1 引脚为低电平（熄灭状态）保持 1000ms，循环呈现闪烁效果。



**五、能力提升与展示交流**

通过查找资料，了解一下 SOS 紧急救援信号灯的原理，与同学分享，尝试制作 SOS 信号灯吧！

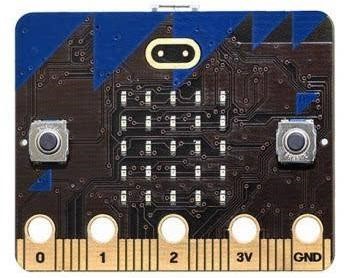
## 项目三 流动的沙子

**一、麦克的故事**

麦克看到好朋友们在玩一个玩具，可好玩呢！它里面的液体能跟着手振动的方向一起移动。麦克很羡慕，打算利用Micro:bit也来实现这个功能。

**二、所需元件**

1.1× micro:bit 主控板



2.1× MicroUSB 连接线



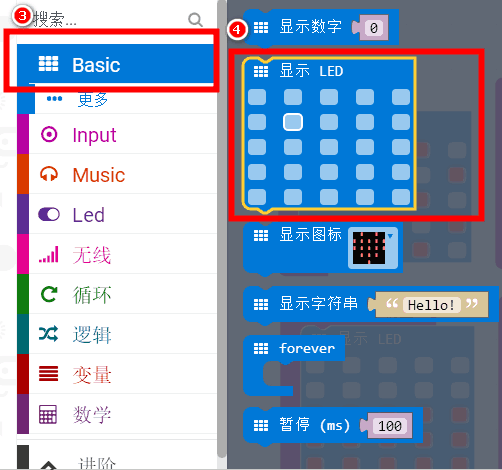
**三、实现功能**

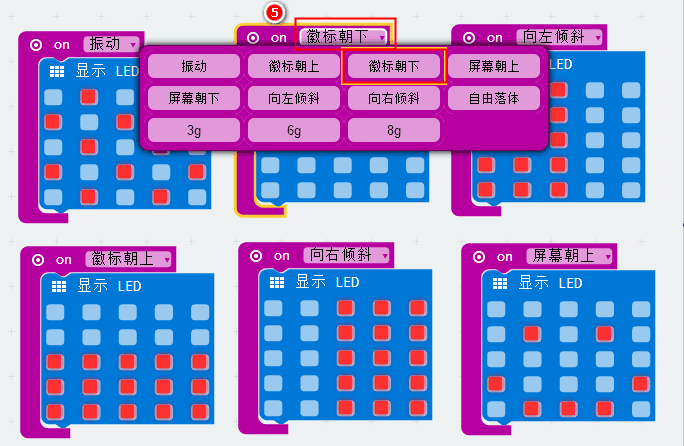
首先正着放（点阵面向上）时micro:bit板子会显示一个微笑、摇一摇可以显示一盘散沙，向左倾斜沙子沉到左边，向右倾斜沙子沉到右边，向下倾斜沙子沉到下面，向上倾斜沙子沉到上面，大家看一看，这像不像一盘流动的沙子呢？

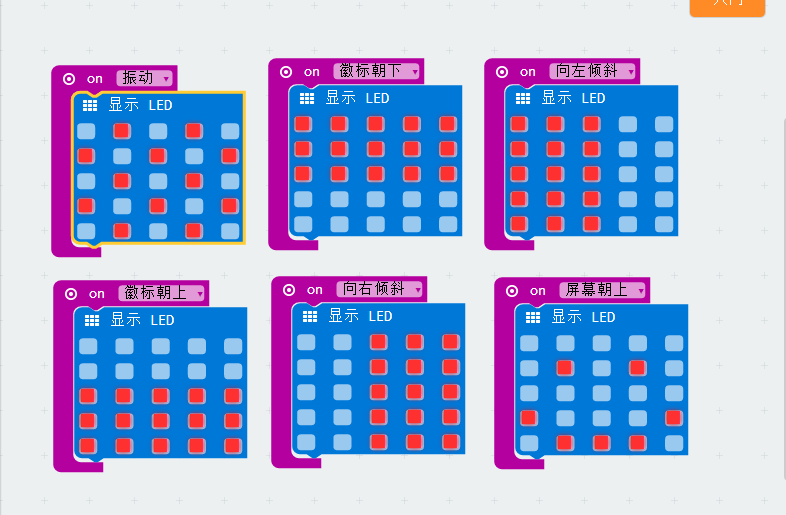
**四、编程学习**

通过input模块—on---实现，当徽标朝上时，LED显示向上的图标；当徽标朝下时，

显示向下的图标等。







**五、能力提升与展示交流**

大家现在已经学会了如何使用micro:bit中的陀螺仪了，但是大家知道陀螺仪是什么吗？陀螺仪是用来测量物理量的偏转、比如倾斜倒置等方位偏转。可以精确的分析判断出使用者的实际动作，通过他收集的这些动作给micro:bit下达一些指令。涉及到陀螺仪的地方很多,例如我们可以利用陀螺仪制作一些小游戏，例如摇骰子小游戏，贪吃蛇小游戏，都是通过micro:bit的陀螺仪实现的，请你来利用陀螺仪来实现吧！

## 项目四 呼吸灯

**一、麦克的故事**

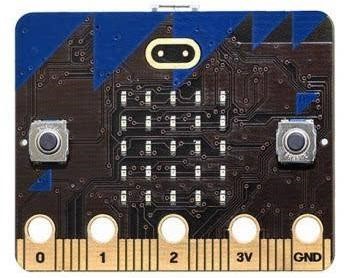
周六早上到了约定好的出发时间，麦克跟着 micro:bit 登上了前往 DF 星球的星际飞船，通过空间跃迁预计需要几个小时才能到达。

过了大约一个小时左右，突然飞船内响起了呜呜呜的警报声，原来是空间跃迁仪出现了故障，飞船偏离了航向，在茫茫星海中迷了路。麦克急得满头大汗，倒是 micro:bit 还算镇定，对麦克说，这种时候千万不要慌，我们先想办法让别的飞船注意到我们，这样就容易获救啦。

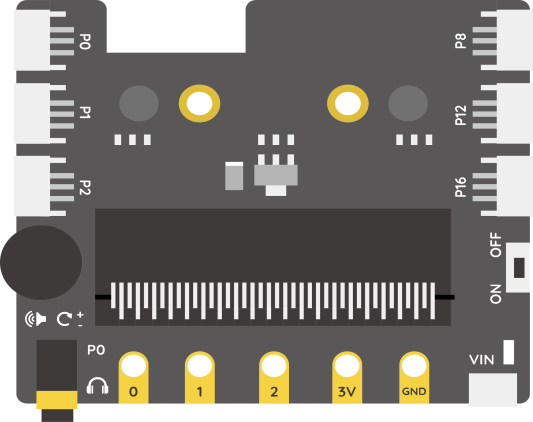
听了 micro:bit 的话，麦克也慢慢镇定了下来，他想起有时候晚上会看到天上的飞机有一亮一暗的灯光，在夜空中十分醒目，于是对 micro:bit 说：“让我们来做一个呼吸灯吧。”

**二、 所需元件**

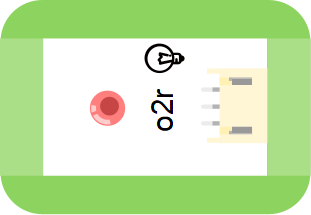
1. 1× micro:bit 主控板



2. 1× micro:bit 扩展板



3. 1× LED 灯模块



4. 1× 按钮模块



按钮就像我们家用按钮台灯的按钮一样，当按下的时候，可以传递一个命令，让我们的小灯被点亮，不过需要连接编写好程序的 micro:bit 小板子才可以哦。

5.1×旋钮模块



旋钮作为输入模块，可以通过旋转不同的角度，传递不同的信号值。旋钮与按钮的区别：按钮处于按下状态时，电路接通，旋钮旋转到一定范围内，电路处于接通状态。

6.1× MicroUSB 连接线



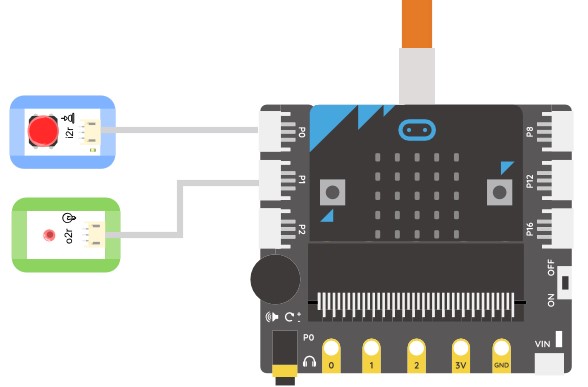
**三、完成任务一：按钮台灯**

**1.实现功能**

当按下按钮的时候，外接 LED 小灯被点亮，松开按钮以后，外接 LED 小灯熄灭。

**2.硬件连接**

把按钮模块（旋钮模块）接到扩展板 P0 号接口上；把 LED 灯模块接到扩展板 P1 号接口上。



**3.编程学习**

STEP1：通过检测 P0 号引脚的值判断按钮是否被按下。

当按钮被按下时，设置 LED 小灯引脚为高电平，LED 灯被点亮，当松开按钮时，设置 LED 小灯引脚为低电平，LED 灯熄灭。

按钮模块属于**数字输入**，单击“高级”，会出现“引脚”，调用引脚时，通常会运用到这个指令，调用指令的流程为：引脚（高级指令）----数字读取引脚 P0。



STEP2：逻辑指令中的“ = ”可以判断按钮是否被按下，如果按钮值为“1”成立，说明按钮被按下，否则，按钮没有被按下。

调用指令的流程为：逻辑----“ = ”。



将“数字读取引脚”嵌套在等式中



STEP3：条件判断——怎样知道按钮是否按下？

如果按钮被按下，外接 LED 小灯被点亮；否则，外接 LED 小灯熄灭。

调用指令的流程为：逻辑----如果为（true）—则。



STEP4：需要用到的指令结合在一起，实现：当按下按钮的时候，外接 LED 小灯被点亮，松开按钮以后，外接 LED 小灯熄灭。



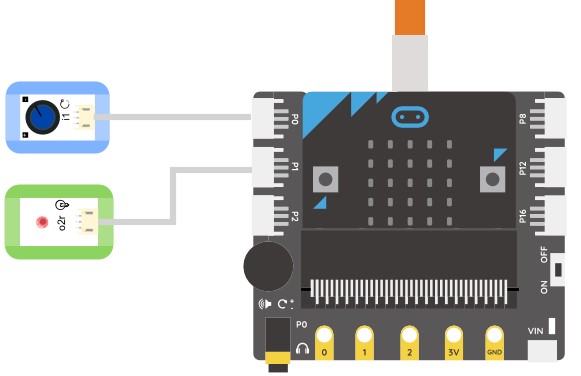
**四、任务二：旋钮控制LED灯**

**1.实现功能**

当旋钮旋至较大值时，micro:bit 控制外接小灯越亮；当旋钮旋至较小值时，micro:bit 控制外接小灯越暗。旋钮的旋转值不同，对应的小灯亮度不同，旋转时，灯的亮度连续变化。

**2.硬件连接**

把任务一按钮台灯中使用的按钮模块换为旋钮模块。



**STEP1**：通过读取旋钮的值，来决定 LED 灯的亮度。

旋钮的值为模拟值，取值范围在 0-1023 之间，将其值映射为 LED 灯对应的亮度值。

旋钮连接板子的 P0 号引脚，旋钮值调用指令的流程为：引脚（高级指令）----模拟读取引脚（P0）。



**STEP2**：LED 灯的亮度由 P0 对应的旋钮值决定，将 P0 的值赋值给 P1，可以控制 LED 灯的改变，调用指令的流程为：引脚（高级指令）----“向 引脚 P0 模拟写入 1023”，将 P0 改为 P1。 之后将获取的旋钮值（P0）放入 P1 值中。



将两个指令嵌套在一起



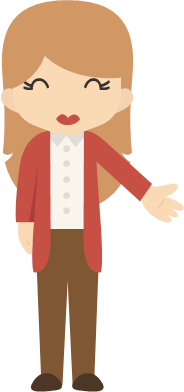
【小贴士】

这段代码可以这样理解：获取旋钮当前旋转位置的

值，这个值对应的是

LED

小灯的亮度值。



STEP3：最终程序



**3.能力提升**

利用家里的一些纸盒、纸杯、彩纸等，制作出精致的小灯外形，让它成为你自己的独一无二的作品。

# 第三单元 如虎添翼

## 项目一：变速风扇

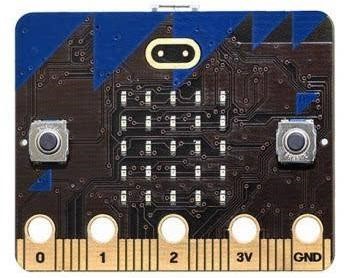
**一、麦克的故事**

星际飞船在宇宙中随着惯性飘荡着，看着舷窗外明暗交替的灯光。 船舱内的温度也慢慢升高了起来，麦克的额头上都渗出了汗珠。麦克擦了一把汗说：“我们来做个小风扇凉快凉快吧。”micro:bit 立刻表示赞成。

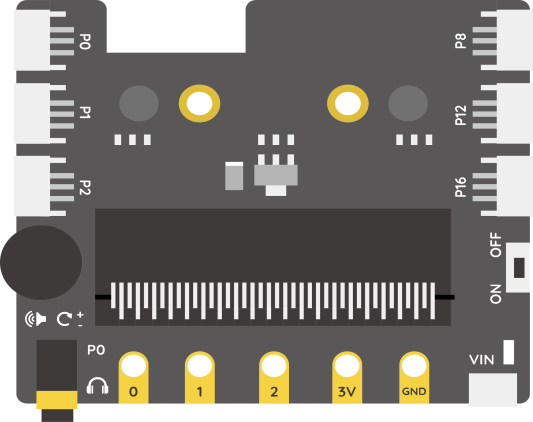
但是两人却有点无从下手，突然麦克灵机一动，想到了小米老师上节课所教给同学们的内容，呼吸灯的明暗变化不就是一个风扇的变速原理吗。于是，麦克和 micro:bit 开始了小风扇的制作。

**二、所需元件**

1.1× micro:bit 主控板



2.1× micro:bit 扩展板



3.1×风扇模块



与家用风扇的原理相似，内部有马达，扇叶可以旋转，转速可以调整。

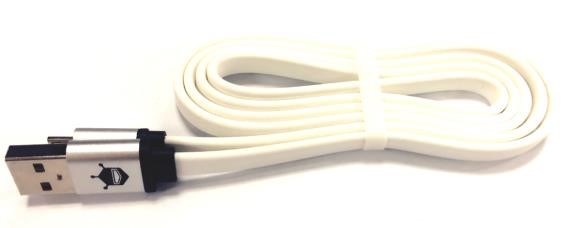
4.1×按钮模块



5.1×旋钮模块



6.1×MicroUSB 连接线



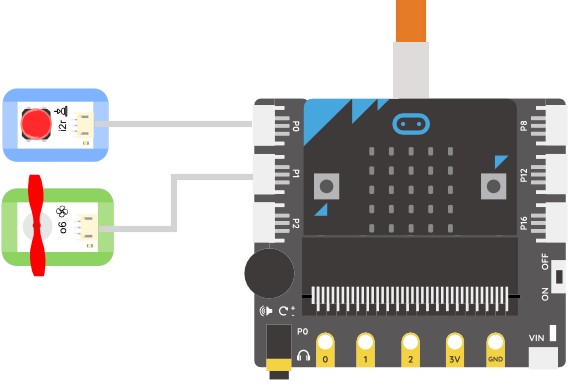
**三、完成任务一：按钮风扇**

**1.实现功能**

当按下按钮的时候，外接风扇转动，松开按钮以后，风扇停止转动。

**2.硬件连接**

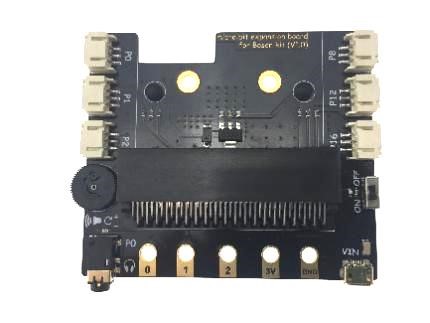
把按钮模块（旋钮模块）接到扩展板 P0 号接口上；把风扇模块接到扩展板 P1 号接口上。



**3.风扇模块的使用**

硬件连接时，我们将 micro:bit 与电脑相连接，进行程序调试和下载；

实验中，由于风扇或舵机等传感器的耗电量较大，所以，需要外接电源（或电脑 USB 口）独立供电。外接电源接在扩展板的外接电源接口 VIN。



**4.编程学习**

**STEP1：**通过检测 P0 号引脚的值判断按钮是否被按下。

当按钮被按下时，设置风扇连接引脚为高电平，风扇转动，当松开按钮时，设置风扇连接引脚为低电平，风扇停止转动。

按钮模块属于数字输入，单击“高级”，会出现“引脚”，调用引脚时，通常会运用到这个指令，调用指令的流程为：引脚（高级指令）----数字读取引脚 P0。



**STEP2：**逻辑指令中的“=”可以判断按钮是否被按下，如果按钮值为“1”成立，说明按钮被按下，否则，按钮没有被按下。调用指令的流程为：逻辑----“=”。



将“数字读取引脚”嵌套在等式中



**STEP3：**如果按钮被按下，风扇转动；否则，风扇停止转动。调用指令的流程为：逻辑----如果为（true）----则----否则。



**STEP4：**需要用到的指令结合在一起，实现：当按下按钮的时候，外接风扇转动，松开按钮以后，风扇停止转动。



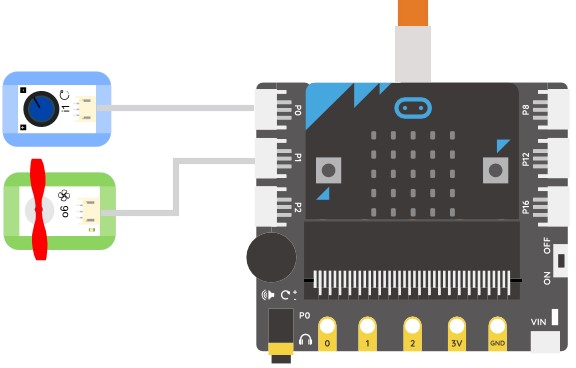
**四、完成任务二：变速风扇**

**1.实现功能**

当旋钮旋至较大值时，micro:bit 控制风扇转速增大；当旋钮旋至较小值时，micro:bit 控制风扇转速变小。旋钮的旋转值不同，对应的风扇转速不同。

**2.硬件连接**

将任务一按钮风扇用到的按钮模块换成旋钮模块。



**3.编程学习**

**STEP1：**通过读取旋钮的值，来决定风扇的转速。

旋钮的值为模拟值，取值范围在 0-1023 之间，将其值映射为风扇对应的转速。

旋钮连接板子的 P0 号引脚，旋钮值调用指令的流程为：引脚（高级指令）----模拟读取引脚（P0）。



**STEP2：**风扇的转速由 P0 对应的旋钮值决定，将 P0 的值赋值给 P1，可以控制风扇转速的改变。

调用指令的流程为：引脚（高级指令）----“向 引脚 P0 模拟写入（1023）”，将 P0 改为 P1。之后将获取的旋钮值（P0）放入 P1 值中。



将两个指令相互嵌套



**STEP3：**最终程序。



## 项目二 音乐盒

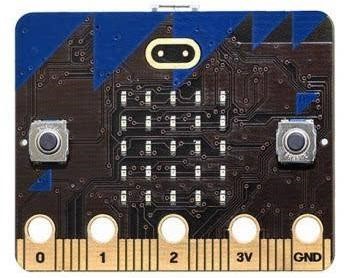
**一、麦克的故事**



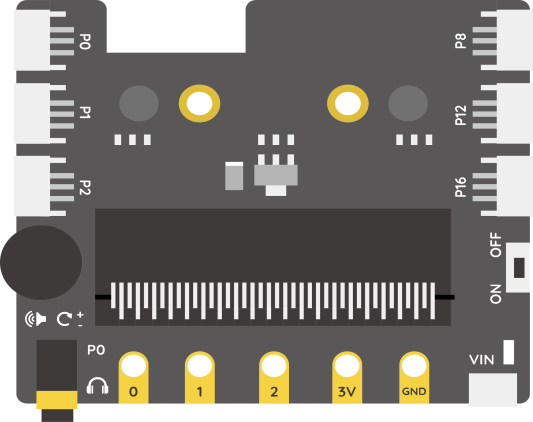
“一闪一闪亮晶晶，满天都是小星星......”麦克在学校最喜欢的就是音乐课啦！这天刚好上完音乐课学到了小星星这首歌，回到家中，麦克突然想起之前学习 micro:bit 的时候听小米老师讲通过编程也能让 micro:bit 板演奏出动听的音乐，这是真的吗？麦克开始认真研究起来，让 micro:bit 板也学会唱小星星。

**二、所需元件**

1. 1× micro:bit 主控板



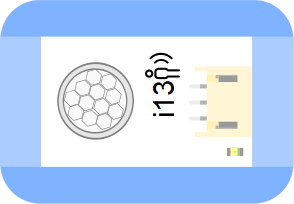
2.1× micro:bit 扩展板



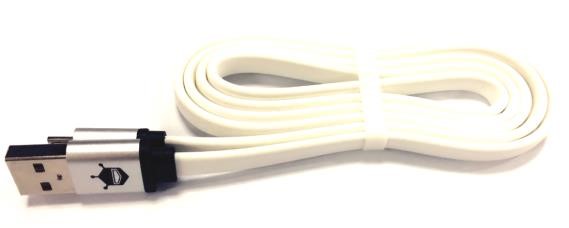
3.1× 普通耳机



4.1× 运传感器模块



5.1× MicroUSB 连接线



**三、完成任务一：电子音乐盒**

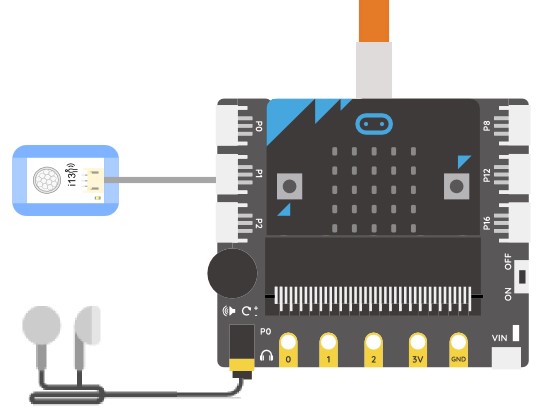
**1.实现功能**

让 micro:bit 主板播放音乐。

**2.硬件连接**

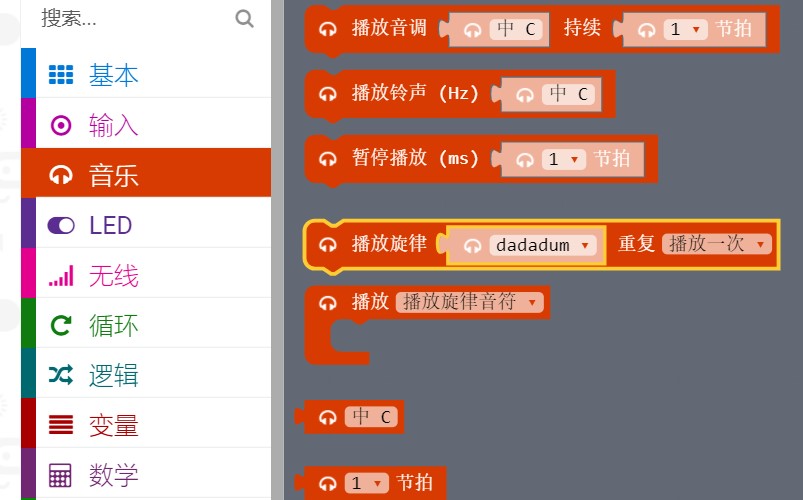
用数据线将 micro:bit 主板与电脑连接起来，将耳机插入耳机插孔；将运动传感器模块接在扩展板的 P1 号接口。

注意：播放音频的耳机插孔数据通信占用了 P0 端口，因此播放声音的时候，P0 端口不可用。

 **3.编程学习**

**STEP1：**将 music 模块集合中的播放旋律模块，拖动到无限循环中。

播放旋律模块中已经有内置的很多旋律可供选择，我们在这里选择的是第一段旋律“dadadum”重复播放一次。

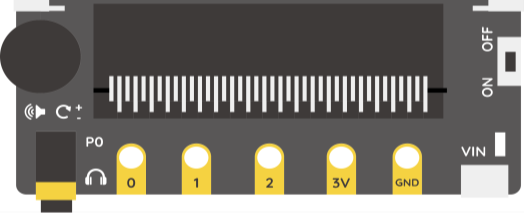


**STEP2：**将选择的旋律“dadadum”一段完整的占 4 个节拍，所以在播放旋律模块的下方需要设定旋律播放的时长为 4 个节拍。

最终程序：



**STEP3：**在扩展板耳机插孔的上方紧挨着的“小转盘”，试着旋转它，可以调节 micro:bit 板播放声音的大小。



**四、完成任务二：智能音乐盒**

**1.实现功能**

当有人靠近时，让 micro:bit 主板播放自己编的一段音乐；当没有人靠近时，不播放音乐。

**2.编程学习**

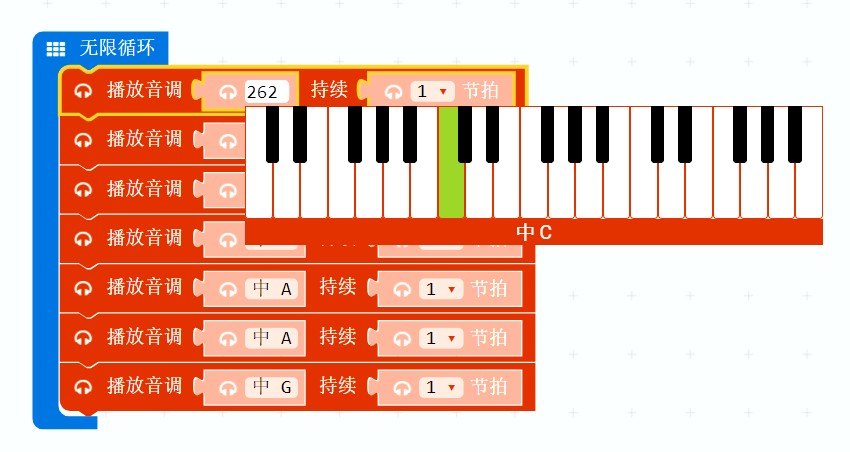
**STEP1：**编写一段歌曲《小星星》的音乐。

先尝试编写能演奏小星星第一句的代码，先看看这首歌的简谱是怎样的，音调不同，音符不同，需要的发声频率也不同。

以中音 C 调为例，发声频率与其对应的音符如下表所示：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 音符 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 字母 | C | D | E | F | G | A | B |
| 频率 | 262 | 294 | 330 | 350 | 393 | 441 | 495 |

**STEP2：**打开 music 代码集合中的代码模块，现在用到的是播放音调模块。在这里可以设置发出的音调和所占节拍。



**STEP3：**编写好小星星的曲子之后我们的 micro:bit 已经学会唱歌啦，但是如何让它变得智能呢？这时候，我们就需要用到**运动传感器**了。其实很简单，只需要添加一个我们之前已经学过的条件判断语句就可以让智能音乐盒得以实现啦！一起来看看代码是怎样的吧？



**五、展示交流**

以小组为单位，进行展示交流和演示。其他小朋友可以进行即时提问。对于比较好的建议，可以对作品进行修改和完善。

## 项目三 炫彩灯带

**一、麦克的故事**

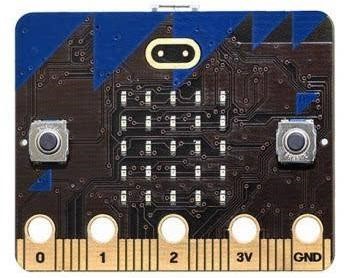
上节课结束之后麦克做出了音乐盒，能发出美妙的音乐又怎么能没有酷炫的外形呢，比如说这样的还有这样的：



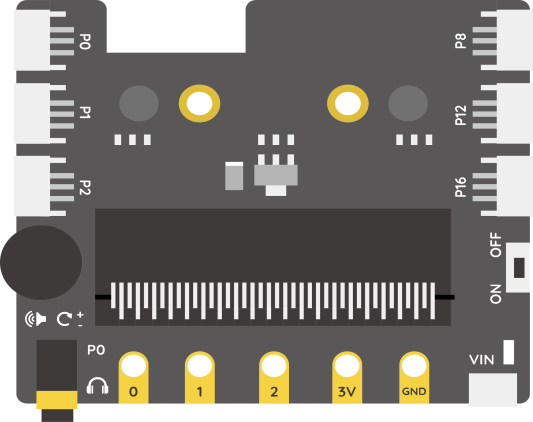
今天，麦克要给音乐盒增添一些色彩了，让 micro:bit 音乐盒变得好看起来。快来和小米老师一起动手吧。

**二、所需元件**

1.1× micro:bit 主控板



2.1× micro:bit 扩展板

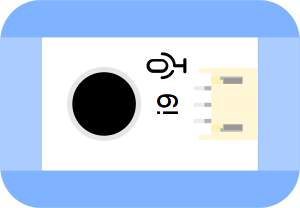


3.1× 彩虹灯带

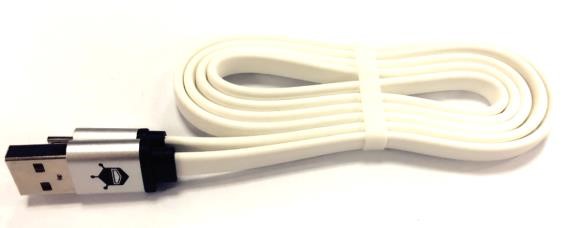


彩虹灯带可以通过调节色彩调出不同颜色的灯效

4.1×声音传感器模块



5.1× MicroUSB 连接线



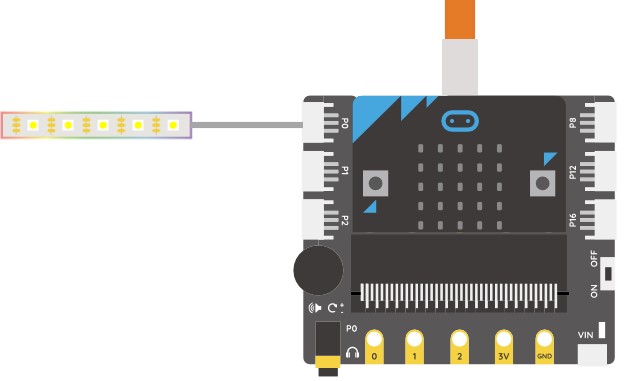
**三、完成任务一：点亮彩虹灯带**

**1.实现功能**

通过编写程序，将程序烧录至板子后，可以控制将彩虹灯带点亮并呈现七彩的效果。

**2.硬件连接**

将彩虹灯带接在扩展板的 P0 号接口。



**3.编程学习**

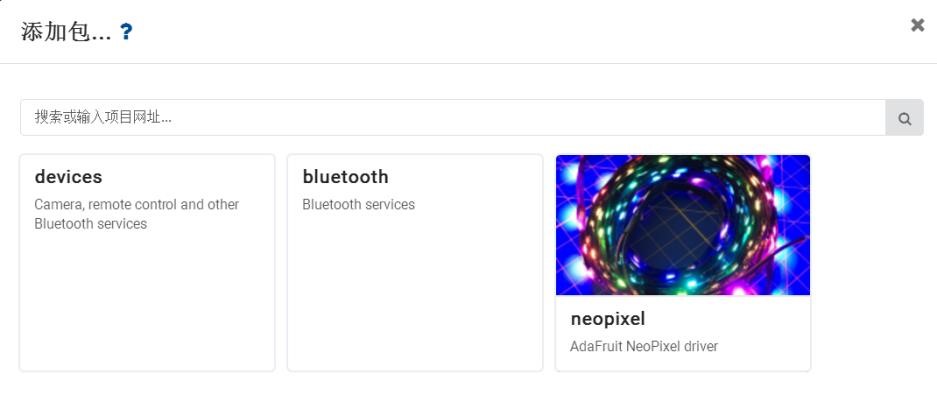
**STEP1：扩展灯带的指令库。**

使用灯带需要调用 Neopixel 指令，Neopixel 指令的调用需要滚动至指令区的最低部，单击“添加软件包”，出现“添加包”的菜单栏。

调用指令的流程为：高级----添加软件包----neopixel。单击 neopixel 会增加 neopixel 模块。



添加软件包 neopixel



**STEP2：灯带常用指令**

在使用彩虹灯带的过程中，首先选定灯带连接 micro:bit 的引脚、灯带 LED 灯的数目及颜色模式，需要调用的指令如下：



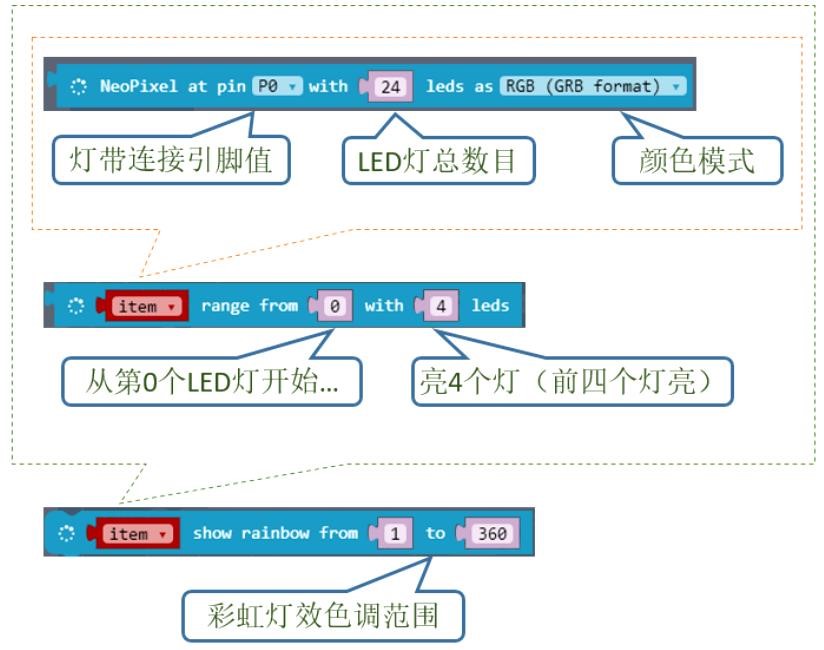
之后要确定 LED 灯使用的范围，需要调用的指令是：



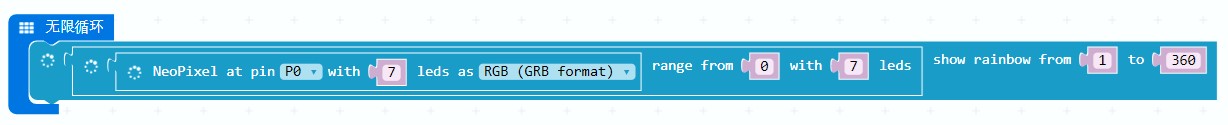
最后要确定彩虹灯效色调范围，需要调用的指令是：



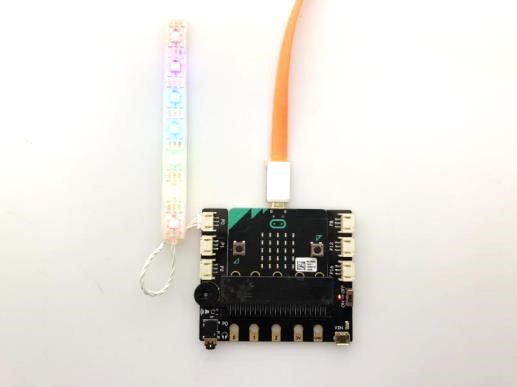
使用灯带的相关指令及说明如下：（在使用过程中，对参数会进行修改）



**STEP3：**灯带连接引脚为 P0，包含 7 个 LED 小灯，设置彩虹灯效色彩范围 1-360（红色--绿色--蓝色），彩灯呈现七彩效果。调整参数后，最终程序。



**4.实验效果**



**四、完成任务二：依次点亮彩虹灯带**

**1.实现功能**

通过编写程序，将程序烧录至板子后，可以控制彩虹灯带的 LED 小灯每隔一秒点亮一盏 LED 小灯，呈现炫彩灯效。

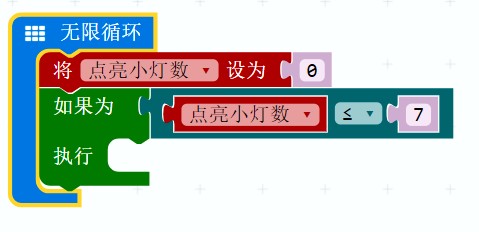
**2.编写程序**

**STEP1：**控制彩虹灯带的 LED 小灯每隔一秒点亮一盏 LED 小灯需要运用可以存放变量的指令，每次来控制小灯被点亮的具体数目，需要调用指令的流程为：变量----设置变量，任意设定变量名（如：点亮小灯数），再单击确定。

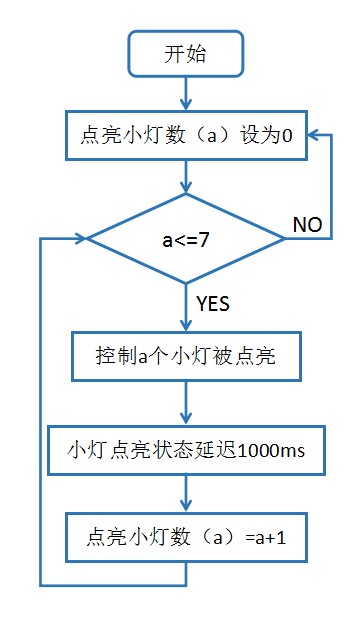


点击设置名称

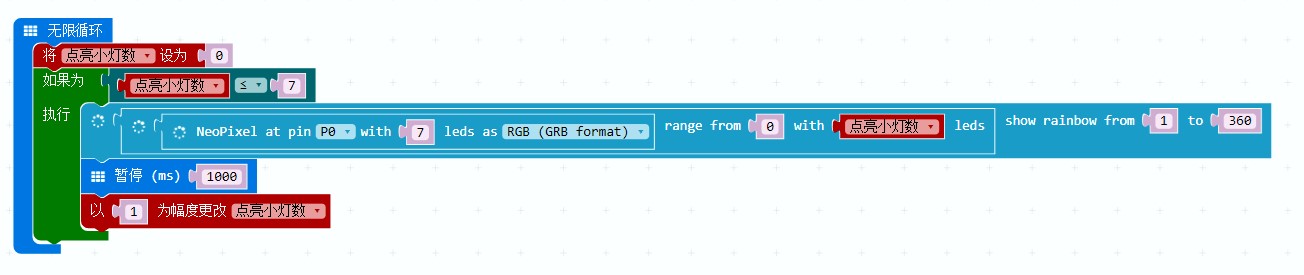
**STEP2：**“点亮小灯数”（也可以简称为“a”）最初设定变量的初值为 0（最初没有小灯被点亮），然后控制变量“点亮小灯数”在原有的基础上增加 1，需要运用到“循环”，当“点亮小灯数”不到7 个的时候，继续点亮小灯。需要调用的指令如下：



**STEP3：**当“点亮小灯数”（a）不到 7 时，可以点亮 a 个小灯，延迟 1000ms，之后让小灯数增加 1，流程图如下。



**STEP4：**将点亮小灯数（a）最初设置为 0，点亮小灯数（a）<=7 时，控制点亮 a 个小灯，并延迟1000ms，接着将点亮小灯数增加 1，流程为：变量---以 1 为幅度更改点亮小灯数。 如此循环，直到点亮小灯数大于 7，所有小灯熄灭，控制彩虹灯带的 LED 小灯每隔一秒点亮一盏 LED 小灯。



**五、能力提高：声音控制彩虹灯带实现功能**

最初灯带为熄灭状态，当发出声音时，点亮一盏 LED 灯，当没有声音时，灯带保持点亮部分小灯状态。

**1.编写程序**

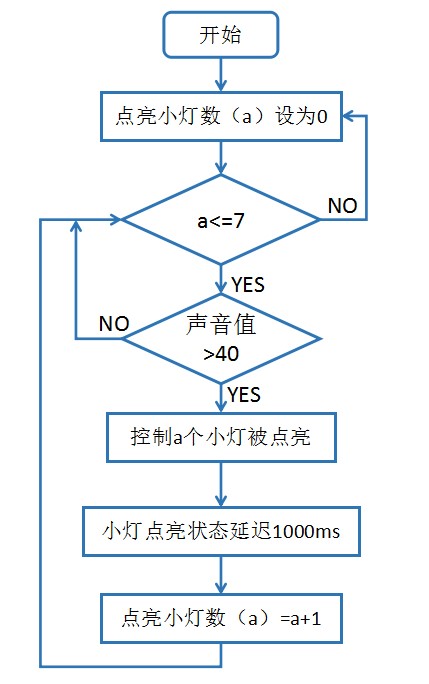
**STEP1：**这个任务需要运用到声音传感器，当发出声音时，声音传感器可以检测到，控制灯带点亮 LED 小灯。之前章节学过声音传感器的用法，我们再来巩固一下调用方法。

调用指令的流程为：引脚（高级指令）----模拟读取引脚（P0），将 P0 修改为 P1。

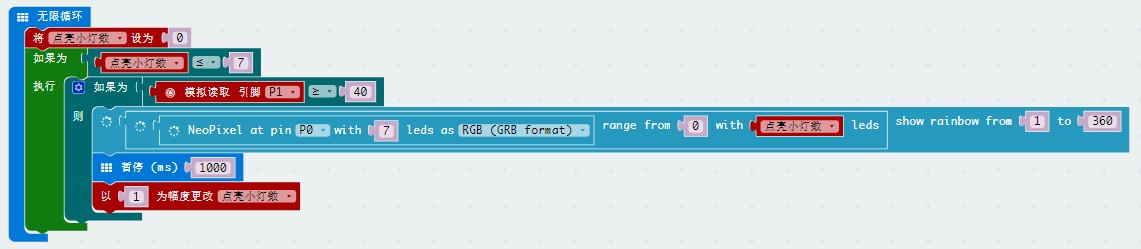


**STEP2：**当声音传感器检测到声音>40 且点亮灯的数量不超过 7 盏时，控制灯带点亮一盏 LED 灯，之后持续不断检测声音是否超过 40。

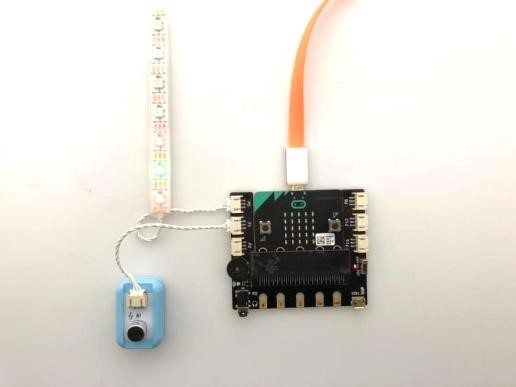
流程图：



**STEP3：**最终程序。



**2.实验效果**



# 第四单元 大展拳脚

**一、麦克的故事**

麦克在生活中发现了一些问题，如全家外出旅游时，家中的宠物和金鱼没办法饲养，盆栽植物没办法照料；体育课上跑步计数经常不准确……你是否能碰到了和麦克一样的情况？请你结合生活实际，以小组为单位，完成一个大项目。

**二、小组完成项目**

以“项目手册”为主线，周期为2-3周，通过前期调查、小组智囊团会议后，最终确定项目内容。

**三、展示交流**

以PPT为切入点，详细介绍创客作品，并进行演示和说明。通过投票的形式，最终产生优秀作品。