

# P 宝：像素动画与智能报警器

(钟文昊 数学科学学院 21000010609)

作品介绍：由于疫情，我无法获得所需的装备，半路临时换题，重新构建想法。本作品分为两部分：一是智能报警器，让您能掌控早八、周三 DDL 等重要时间节点；二是像素动画，利用仅有的 5\*5 的像素点进行创作，其中双人芭蕾的两个舞者起舞构成“北大”是作者自认为的一个亮点。

摘要：此作品主要实现了报警器系列和像素动画两个部分的内容，让您不再栽在“时间的流逝”上，具体分为 DDL 报警器，疯狂星期四报警器，摸鱼报警器和早八报警器，同时享受一段“大道至简”的像素动画（就 25 个像素点，实在是不能不“简”），像素动画有两个不同的主题：奔跑篇和芭蕾篇。

## 一、选题及创意介绍

不借助任何外部的设备和传感器，只用 microbit 主板这样一块小小的单片机可以做什么？本作品分为两个部分，前半部分为报警器系列，后半部分为像素动画。报警器系列来源于日常的学习生活，尤其是数算课堂，让您在生活及数算的重要时间点上不再遗忘，光感系统让您拥有一个和手机闹钟不一样的报警器。像素动画挑战了在 5\*5 的像素点的极限情况下的动画创意，分为奔跑篇和芭蕾篇，芭蕾篇是作者比较满意的内容，其中双人芭蕾通过两个翩翩起舞的人为北大 124 周年校庆献上祝福。

## 二、设计方案和硬件连接

报警器简单来说分成三步：设置计时系统，表情转换，选择报警音，根据四个不同的情境，选择不同的内容。像素动画主要难在如何利用稀有的 25 个像素点展现出人物的动作变化，并让这些动作连贯起来。

报警器系列来源于日常的学习生活，尤其是数算课堂上，鉴于有些同学总是在周四凌晨才想起来周三的 DDL，microbit 会为您展示 DDL 倒计时的恐惧，当然也会提前提醒您 DDL 要到了；周三完成了一周的任务，周四怎能不来一顿肯德基疯狂星期四。待在书桌前，摸起鱼来，可谓是时光飞逝，microbit 会每隔一小时检测您是否在学习。早八起不来？将 microbit 放在阳台，利用光感，调节好亮度阈值，太阳升起天亮之时便会叫醒您，让您不再睡过头（您必须到阳台才能把它关掉，嘿嘿，具有强制性）基于此设计了 DDL 报警器，疯狂星期四报警器，摸鱼报警器和早八报警器。

像素动画奔跑篇展示了一个人向前奔跑，随后回头探视，最后和他的小孩一起奔跑。单人芭蕾展示了一个芭蕾舞者从开始的静止站立状态开始，进行转体四周的动作，最后跳跃腾空。双人芭蕾通过两块 microbit 板，两个舞者分别在两块单片机上起舞，随后向中间靠拢，共同奔赴，在两位舞者同时向中间跃起的时候，两块主板显示出‘北’，两位舞者渐渐靠拢，‘北’字渐渐转变为‘大’，献给北大 124 周年校庆。

硬件连接只需将 microbit 连接在供电装置上，控制好开机时间和单片机间的配合。

## 三、实现方案及代码分析

报警器的计时采用 microbit 自带的 time 模块，时间每停一秒剩余时间就减少一秒，定义 alarm 函数，传入小时和分钟两个参数，将其转化为秒并开始计时，剩余时间达到预设值之后，microbit 便会显示不一样的图像并发出警报声。早八唤醒器采用自带的光传感器及内置函数 display.read\_light\_level()，当光的亮度大于预设值时，便会开始响起闹铃。表情和音乐精选自 microbit 自带的库中。

像素动画主要是像素图像的绘制，作者花了很长的时间思考用 5\*5 的像素矩阵可以表

现出哪些内容，（借了一块单片机后）有两块单片机后这两块单片机怎样实现交互，以及如何控制两块单片机的图像变化速率一致，从而表现出连贯一致的动画感。以下为几个像素图的例子（像素图中 9 代表显示红点，0 代表不显示）：

例 1. 左右单片机分别显示 Image5 和 Image7，这两者图分开来看是两个跃起的舞者，合起来看就是汉字‘北’，控制左右单片机变化速率，让两个舞者以相同的速度向中间靠拢，便得到下一个例子（即例 2，一个值得注意的地方是例 1 和例 2 之间的舞者的动作是连贯的）。

```
Image5=Image("0 0 0 0 9:"  
              "0 0 9 9 9:"  
              "0 0 0 0 9:"  
              "9 9 9 9 0:"  
              "0 0 0 0 0")  
Image7=Image("9 0 0 0 0:"  
              "9 9 9 0 0:"  
              "9 0 0 0 0:"  
              "0 9 9 9 9:"  
              "0 0 0 0 0")
```

例 2. 左右单片机分别显示 Image9 和 Image10，这两者图分开来看是两个向中间靠拢的舞者，合起来看就是汉字‘大’。于上一个例子合起来，即是变化而生的“北大”。

```
Image9=Image("0 0 0 0 9:"  
              "0 0 9 9 9:"  
              "0 0 0 0 9:"  
              "0 0 0 9 0:"  
              "0 0 9 0 0")  
Image10=Image("9 0 0 0 0:"  
               "9 9 9 0 0:"  
               "9 0 0 0 0:"  
               "0 9 0 0 0:"  
               "0 0 9 0 0")
```

例 3. 左右单片机分别显示 image2 和 image3，呈现出的便是和谐家庭的图，大手牵小手，家长跟在小孩后面。

```
image2 = Image("0 0 0 0 0:"  
               "0 9 0 0 9:"  
               "9 9 9 0 9:"  
               "0 9 0 9 0:"  
               "9 0 9 0 0")  
image3 = Image("9 0 0 0 0:"  
               "9 0 0 0 0:"  
               "0 9 0 0 0:"  
               "0 9 9 0 0:"  
               "0 9 0 9 0")
```

例 4. 通过更换图片展现出一个动图，为一个舞者在舞台上旋转。这个例子仅在一块 microbit 主板上实现。（图片具体内容见代码）

# 单人芭蕾

```
while True:
    # 初始化站立的舞者
    display.show(Image6)
    time.sleep(2)
    # 转体四周
    for i in range(4):
        display.show(Image1)
        time.sleep(0.7)
        display.show(Image2)
        time.sleep(0.7)
    # 跃起
    display.show(Image3)
    time.sleep(5)
```

本次作品共创作了二十余幅简单的像素图，通过自身的变化，图像之间的转化，相邻图片的配合，两个单片机之间的配合，完成了几个场景的构建。

#### 四、后续工作展望

如果有更多的时间，更多的 microbit 主板，或者更大的像素点矩阵，像素动画可以有更生动形象的展示，单片机间的互动性可以提升，一块单片机有了个‘动作’，另几块单片机响应它的‘行为’，比如一个单片机上的小人开了一枪，另几块单片机的小人躲避。基于板间交互创作一些更加连贯，更具故事性，更加精彩的像素连环画。