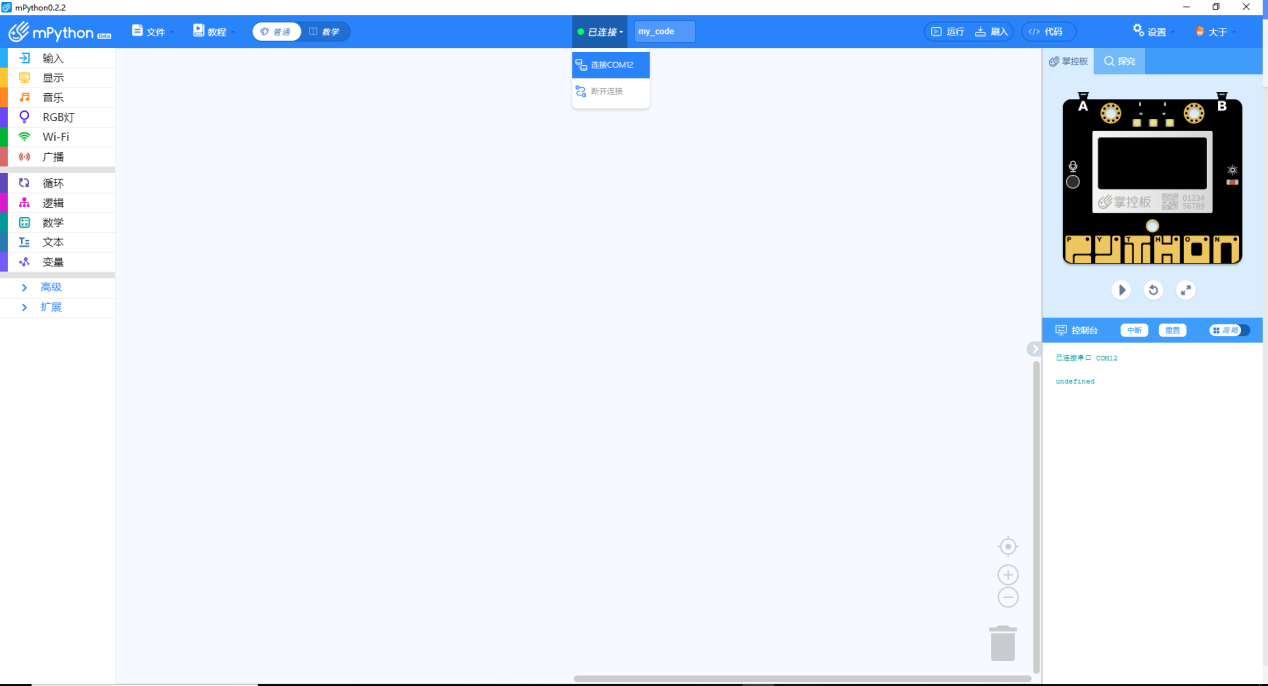
**一、基本信息**

|  |  |
| --- | --- |
| **作品名称** | 我爱游戏\_掌控板\_赛车 |
| **作者姓名** | 于飞 |
| **作者单位** | 嘉善信息技术工程学校 |
| **教材名称** | 《慧编程》 |
| **教材类型** | 自编教材 |
| **教材简介** | 自编校本教材，作为学生自由选修课使用，丰富学生课程生活，打造“四力融合，课程改革”，符合其中创造力的学习。 |
| **执教年级** | 职业高中 |
| **课时长度** | 2课时 |
| **涉及器材** | 掌控板 |
| **涉及软件** | mPython |
| **文件清单** | 教学设计、配套视频、配套代码 |

**二、作品内容**

【硬件搭建】所需材料掌控板和数据线，用数据线将掌控板和电脑连接，选择已经连接COM接口，点击连接成功。

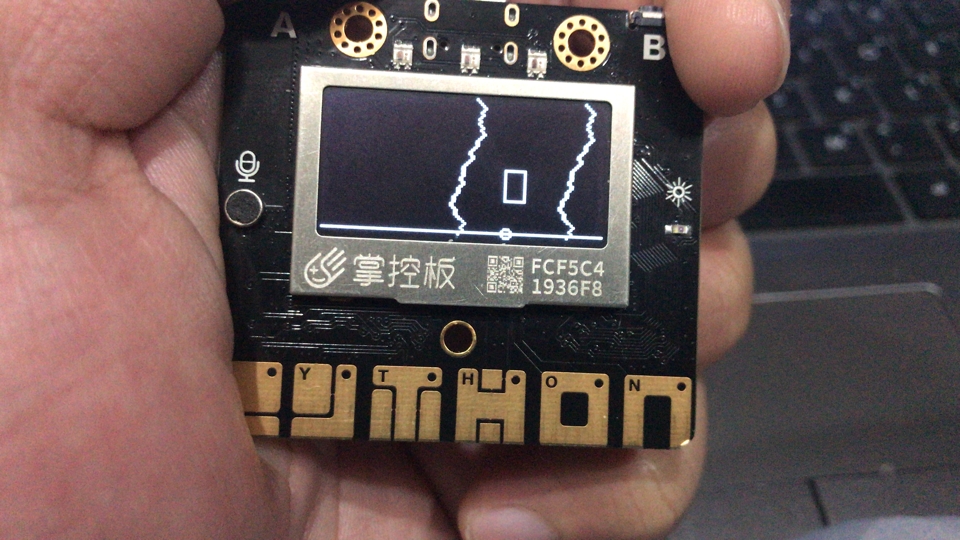


【作品功能】点击“运行”将程序上传到掌控版，左侧和右侧变动的线是赛道的边界，实际上他是由64个点排列组成的，左右各64个点，中间空心矩形是我们控制的小车，下边缘的一条线和线上的空心圆是用来标识当前状态下掌控板的倾斜角度（小车的移动受到掌控板倾斜角的控制）。

屏幕下边缘的空心圆用于反馈掌控板的倾斜程度；

空心矩形为主角小车，y轴不变，x轴的值受到有y轴倾斜角的控制；

左右跑道边界，实时更改点位，操作者需要根据变化调整自己的倾斜角度，避免撞到边界，如果撞到边界游戏结束。



【教材与学情分析】：本课是《慧编程》的第三节课，这些学生在上一学期已经完成了scratch的基础学习，对于逻辑语句的使用已经没有障碍，但是没有接触过市场上的编程外设，所以带来这样一个基础外设-掌控板，我使用盛思的配套软件mPython作为教学软件，期中的数据类型是学生在scratch中没有接触过的，这里为了减少变量的使用会增加字典和列表的学习，这次课制作一个赛车的小游戏，使用“描点”功能绘制赛道两侧的边界，通过传感器控制小车的左右移动，随着赛道的不断变化，需要控制好自己小车，如此游戏，激发了学生学习的欲望。不仅让学生熟悉了这款外设，更能提高学生的学习兴趣。

本课授课对象是中职一年级编程兴趣班学生，学生人数40人以上。中职类学生相对普高来说数学和英语的基础素质较差，所以选用这样一款图形化编程软件，上学期有过scratch的基础，搭配掌控板可以实体操作，之前的两节课，相信已经对图形绘制和变量的使用已经熟练。这些学生可以通过自己的操作，真正的做出一款小游戏，重拾自信。

【学习目标】：1.学会使用“描点功能”。

2.掌握使用列表功能。

3.熟练掌握映射功能。

【学习重难点】

重点：1. 掌握描点功能。

1. 使用列表存储更多变量。
2. 列表元素的遍历判断。

难点：1.使用描点功能绘制出赛道两侧的边界，实现赛道的变换（左转/右转）； 2.列表元素的遍历使用（对比判断）。

【教学资源】：学生机房、电脑一人一机、掌控板六人一块、 学生机与教师机局域互联并安装mPython软件；成品体验。

【预设流程】

**环节一、绘制赛道**

教师活动：演示demo，赛道不断在左右变换，需要调整掌控板的倾斜角度及时修正位置，防止碰触到边界。

教师一边操作一边口述运行要领和对应的知识点，让学生有大概的了解。

教师端广播屏幕演示：

绘制左右各一条线作为跑道的边界。



由于跑道是y轴的值为0~63，跑道不能是一直都是垂直状态，所以要一直在变动，这样就能营造出弯道的效果。那么我们使用列表来实现这么多坐标的存储。新建listx列表，用于存储每一行的x值（赛道左侧），至于赛道右侧就是x+赛道宽度即可（我设置的是50）。



学生活动：学生小组内传递观看掌控板，并体验小程序。

观看教师端广播视频演示脚本和运行状态，思考为什么不能用直线来制作赛道的边界，

效仿教师制作列表，用于存储赛道边界的描点坐标。

**环节二、修整赛道**

教师活动：烧录环节一中的脚本，整个屏幕都是乱套的，本来是使用随机数来左右赛道的转弯变化，效果实在不忍直视，改为如下的方法，使得转弯变得更加圆滑。

使用随机数来决定y=0的x坐标值，后面的y=1，y=2····依次继承上一行的x坐标值。形成不断更新的滚动效果。演示结束后，机房巡视，观察学生制作。



学生活动：观看教师演示整个demo制作过程和烧录过程，在可以操作时，自己操作制作烧录，及时反馈自己遇到的问题。求助教师，协助解决问题。

**环节三、制作水平提示**

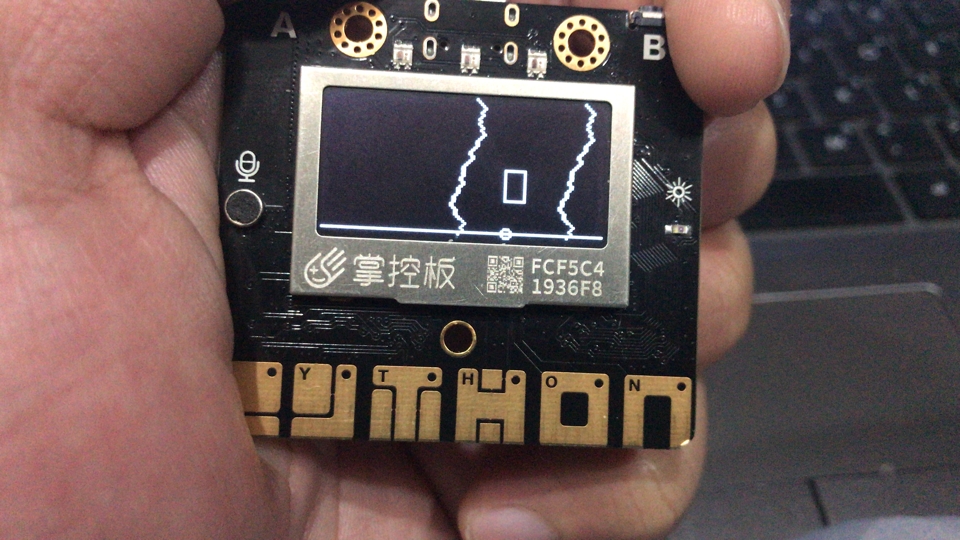
教师活动：

在屏幕底边绘制水平直线，y=61的位置，在y=61的位置绘制空心圆，并将其x坐标值与传感器（y轴倾斜角）相关联，实时提示使用者掌控板的倾斜程度。



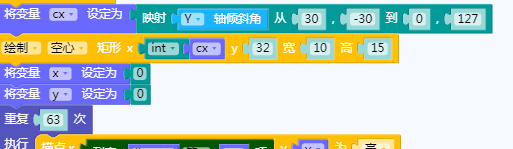
注意屏幕下边缘的空心圆和水平的横线。效果展示。

布置课堂临时作业，添加一个矩形作为主角“赛车”，同样使用y轴倾斜角控制，可以任意发挥。效果和“水平提示球”类似即可。



学生活动：学生在机位上观看学生端教师演示，观看结束后自主制作，编辑脚本，上传到掌控板，实现使用映射来制作倾斜角的提示图。

制作课堂作业，完成矩形的制作。



设计意图：复习映射概念，直线、圆形、矩形的绘制，再次记忆各种图形的坐标点，重点是传感器的使用，第二节课中有使用过，这次课再次强化。

**环节四、碰撞逻辑（矩形和赛道边界）**

教师活动：播放demo视频，提问学生要如何判断矩形和边界发生了碰撞？

由于赛道是由64\*2，两条描点组成的“曲线”，每个点的x坐标都存储在列表list\_x中，只需要将列表中的全部点与矩形范围内点进行比较即可（提示：赛道宽为50）。教师演示制作，学生观看学习。



学生活动：学生观看学习教师屏幕广播，结束后自行制作。有问题向老师求助解决。

【拓展探究】

教师活动：我们对列表的遍历到底有没有必要全部遍历？

解决方案：对于列表元素遍历判断的范围取决于矩形的高，绘制的矩形越高，那么需要判断的碰撞范围就越大，需要遍历的元素也就越多。（例如，生活中，大一些的车不容易转弯，而小车转弯相对灵活许多）。

学生活动：查找自己的矩形高度值和坐标，修改遍历范围，减少遍历时长。

【交流总结】

教师活动：请每一组出一个代表，展示并说明自己的作品。

学生活动：运行并展示自己的小项目，说明自己的优势。

设计意图：小组之间互相对比，各有不同，相互学习。通过每位代表的介绍，可以判断学生的掌握情况，梳理了整节课的知识重点。

【教学反思】

不足：关于列表的遍历，判断速度很快，学生理解起来有点问题，实际上应该做一个元素少一点的列表，便于讲解，64个元素的类表理解起来有点麻烦。

优点：吸取了前两次课的不足，增加了课堂项目的趣味性，课堂粘度更高，学生学起来积极性更高。