**武汉纺织大学**

**毕业设计（论文）任务书**

**课题名称： 基于Python的体感游戏设计**

**完成期限： 2019 年12月26日至2020年5月30日**

**院系名称：机械工程与自动化 指导教师：马双宝**

**专业班级：自动化11601 指导教师职称：副教授**

**学生姓名： 钟成**

**院系毕业设计（论文）工作领导小组组长签字**

**一、课题训练内容**

1. 与Python游戏设计相关技术资料的收集、查询以及文献料综述，完成开题报告；

2. 学习Python编程基础知识，并训练用函数设计与调用来解决实际问题的能力；

3. 训练传感器与单片机，单片机与电脑通信的能力；

4. 游戏规则分析设计能力；

5. 单片机配置IO口等编程能力；

6. 针对编程遇到的各种问题，设计并评价解决方案；

7. 论文撰写能力、计算机应用能力、外文阅读与翻译能力；

**二、设计（论文）任务和要求（包括说明书、论文、译文、计算程序、图纸、作品等数量和质量等具体要求）**

1. 了解Python基础知识，熟悉各种Python库，学习如何调用pygame、re、pyserial等Python库；

2. 设计游戏规则、游戏框架、硬件电路图等；

3. 编写游戏函数，游戏代码，编写单片机软件，搭建硬件电路。

4. 测试游戏运行，调试游戏参数。

5. 按照学校统一要求，完成开题报告和毕业设计论文等要求。

**三、毕业设计（论文）主要参数及主要参考资料**

**1. 毕业设计主要参数**

游戏帧率：60

可以自由设定游戏角色移动速度

实时显示游戏角色所得奖励果实

**2. 毕业设计主要参考资料**

[1]谭浩强.C程序设计（第三版）[M].北京:清华大学出版社,2005.

[2]Eric Matthes .Python编程:从入门到实践[M].北京人民邮电出版社,2016.

[3]Bradbury.树莓派Python编程指南[M].机械工业出版社,2015.

[4]刘火良.杨森.STM32库开发实战指南[M].机械工业出版社,2013.

[5]小甲鱼. 零基础入门学习Python[M]. 北京:清华大学出版社, 2016.

[6]黄焕林.丁昊. 从零开始学习Arduino电子设计[M].机械工业出版社，2018.

**四、毕业设计（论文）进度表**

**武汉纺织大学毕业设计（论文）进度表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序  号 | 起止日期 | 计划完成内容 | 实际完成情况 | 检查人签名  检查日期 |
| 1 | 2019.12.26---  2020.3.10 | 收集与本课题相关的各种资料，并消化吸收，撰写开题报告。 |  |  |
| 2 | 3.11-3.24 | 需求与设计总体解决方案 |  |  |
| 3 | 3.25-4.7 | 理论分析与设计 |  |  |
| 4 | 4.8-4.21 | 电路与程序设计 |  |  |
| 5 | 4.22-5.12 | 总体实验运行/调试，并评价解决方案的可行性 |  |  |
| 6 | 5.13-5.30 | 撰写毕业设计，准备答辩 |  |  |

注：1.本任务书一式两份，一份院（系）留存，一份发给学生，任务完成后附在说明书内。

2.“实际完成情况”和“检查人签名”由教师用笔填写，其余各项均要求打印，打印字体和字号按照《武汉纺织大学毕业设计（论文）规范》执行。

武汉纺织大学毕业设计（论文）开题报告

|  |  |  |  |
| --- | --- | --- | --- |
| 课题名称 | 基于Python编程的体感跑酷游戏设计 | | |
| 学院名称 | 机械工程及自动化 | 专 业 | 自动化专业 |
| 班级 | 自动化11601班 | 学生姓名 | 钟成 |
| 一、本课题研究背景和意义  随着社会的不断发展进步，越来越多人走入了城市，在寸土寸金的城市里，人们环境受限，喜欢宅在家里，很少出门运动，经常以游戏作为工作学习之余的消遣娱乐活动。专家表示，现代人普遍缺乏锻炼，身体素质出现滑坡。所以，为了迎合人们喜欢游戏的需求以及身体锻炼的需求，本人运用所学习到的Python编程技术和单片机技术，开发出这一款体感Python跑酷游戏。研究步骤如下，先用pygame库写出完整用键盘操纵的跑酷游戏，然后用超声波测距模块或者人体红外检测模块检测玩家是否跳起和跳跃时间，向单片机传递响应数据，单片机接收并处理数据后，通过串口线传递数据给电脑，Python程序检测串口数据，紧接着调整游戏中人物的是否跳起和跳起高度。这样，就可以让玩家玩游戏的同时锻炼了身体。  二、该领域发展状况  随着AR、VR技术的不断发展，体感游戏也不断进步，目前比较火热的switch游戏机就是体感游戏的一个巅峰，玩家通过游戏手柄上的各种传感器来控制游戏角色，结合任天堂品类繁多的游戏，这款游戏机可以模拟现实打乒乓球，模拟现实格斗等。除了这种传感器来控制游戏，还有一种模式体感游戏则另辟蹊径，比如微软出的XBOX游戏机，根据摄像头加图像捕捉来操纵游戏，玩家可以无束缚玩虚拟现实游戏。这些都是目前体感游戏领域的大成之作。  三**、**课题研究内容  本课题可以简单分为三个部分，游戏设计部分、硬件设计部分、调试测试部分。游戏设计部分主要研究pythony关于游戏设计相关库函数的应用与调试。硬件设计部分主要研究设备之间通信技术以及单片机的编程技术。  四、本课题研究方法和手段  主要研究方法是先跟着网上学习资料学习编程基础，模仿做出小游戏，然后自己设计游戏规则，自己设计游戏框架，一步步编写代码，调试代码。期间出现问题如果自己解决不了就在网络上寻找资料，寻找类似问题及相应的解决方案。  五、研究步骤  **1.**研究相关编程技术及基本原理  **2**.构建游戏框架  **3.**编写游戏代码  **4.**运行游戏，观察游戏结果，分析  **5.**得出结论并撰写论文  参考书目:  [1]谭浩强.C程序设计（第三版）[M].北京:清华大学出版社,2005.  [2]Eric Matthes .Python编程:从入门到实践[M].北京人民邮电出版社,2016.  [3]Bradbury.树莓派Python编程指南[M].机械工业出版社,2015.  [4]刘火良.杨森.STM32库开发实战指南[M].机械工业出版社,2013.  [5]小甲鱼. 零基础入门学习Python[M]. 北京:清华大学出版社, 2016.  [6]黄焕林.丁昊. 从零开始学习Arduino电子设计[M].机械工业出版社，2018. | | | |

摘 要

随着社会的不断发展进步，越来越多的人走出了农村，走进了城市。在寸土寸金的城市里，人们受环境所限，很少出门运动。越来越多的人喜欢在休息时间宅在家中打游戏。专家表示，现代人普遍缺乏锻炼，身体素质已经出现大面积滑坡，这必须尽早引起人们关注。所有，为了迎合人们对游戏的热爱以及对锻炼的需求。本次毕业设计开发了这样一款基于python的体感跑酷游戏。该设计分为两大部分，硬件检测部分和软件游戏部分，软件游戏部分又分为游戏界面设计与游戏功能设计。本次设计的软件游戏主要是运用Python中的pygame库来进行开发。硬件检测部分主要是运用Arduino单片机和红外传感器进行开发。

关键词 ： 跑酷，游戏，Python，pygame，Arduino，单片机

**ABSTRACT**

With the continuous development and progress of society, more and more people have walked out of the countryside and into the city. In an inch of city, people are limited by the environment and seldom go out to exercise. More and more people like to play games at home during rest time. Experts say the general lack of exercise in modern people,the body is getting worse.which must be early attention. all, in order to cater to people's love of games and the need for exercise. And this graduation project developed such a body sense parkour game based on python. The design is divided into two parts, hardware detection part and software game part, software game part is divided into game interface design and game function design. The design of the software game is mainly using the pygame library in the Python to develop. Hardware detection part is mainly developed using Arduino Microcontroller and infrared detection module.

**Keywords:** parkour; game; Python; pygame; Arduino; Microcontroller

目 录

[1 概述 1](#_Toc28422)

[1.1 国产游戏发展现状 1](#_Toc21820)

[1.2 国产游戏的未来发展远景 1](#_Toc31352)

[2 开发工具的介绍 2](#_Toc9493)

[2.1 Python语言简介 2](#_Toc12412)

[2.2 pygame库简介 2](#_Toc7680)

[2.3 Pycharm软件介绍 3](#_Toc20990)

[2.4 Arduino软件介绍 3](#_Toc32085)

[3 需求分析 3](#_Toc11917)

[3.1 可行性分析 3](#_Toc9627)

[3.2 系统平台要求 3](#_Toc29441)

[3.3 硬件要求 3](#_Toc15117)

[4 概要设计 3](#_Toc14948)

[4.1 功能介绍 4](#_Toc15710)

[4.2 软件概念设计图 4](#_Toc23332)

[4.2.1 游戏元素设计 5](#_Toc20644)

[4.2.2 游戏界面设计 7](#_Toc31448)

[4.2.3 游戏框架设计 9](#_Toc13969)

[4.2.4 游戏逻辑设计 1](#_Toc1305)2

[5 详细设计 1](#_Toc16985)5

[5.1 主要类及模块 1](#_Toc17985)2

[5.1.1 主要模块及其作用 1](#_Toc9471)3

[5.1.2 主要类及其作用 1](#_Toc12562)4

[6 硬件电路及程序设计 1](#_Toc7394)9

[6.1 主要硬件介绍 1](#_Toc20106)9

[6.2 单片机及其程序介绍 1](#_Toc10934)9

[7 测试与运行 2](#_Toc4472)0

[7.1 运行软件程序及调试 2](#_Toc16211)0

[7.2 运行硬件程序及调试 2](#_Toc21384)1

[8 总结 2](#_Toc12552)2

[8.1 遇到的问题及解决方法 22](#_Toc12256)

[8.1.1 如何制作合适的图片素材 22](#_Toc31810)

[8.1.2 如何让障碍物和奖励物随机出现 22](#_Toc20362)

[8.1.3 如何让孙悟空跳起来 23](#_Toc13013)

[8.1.4 如何读取串口数据不影响游戏运行 23](#_Toc7739)

[8.1.5 如何将提取的串口信息中的数字信息并转为整型 23](#_Toc22057)

[8.1.6 如何绘制血条 23](#_Toc21042)

[8.1.7 如何选择合适的传感器和单片机 24](#_Toc31757)

[8.2 结论 24](#_Toc18508)

参考文献

致谢

1 概述

游戏是如今世界越来越火热的娱乐方式，近些年，各种各样的游戏层出不穷，有些游戏简单易懂、老少皆宜，有些游戏错综复杂、适合团队攻关。但无论是哪一款游戏，都为人们带来了欢声笑语。在无聊休闲时间里，大多数人都会选择来一款有趣的游戏。虽然电影、音乐、文学、旅行都是很好的打发时间的娱乐方式，但显然游戏的主动性，沉浸感更强。游戏已经成为了现代社会不可或缺的一部分。

跑酷游戏自诞生起就是深受大家喜欢的一类游戏，以其简单的操作和惊险刺激的游戏设定为大家所熟知，无论是前些年火爆全国的天天跑酷还是近些年火遍全球的神庙逃亡，无不广受玩家好评。

* 1. 国产游戏发展现状

中国有着庞大的入口基数，随着人们生活条件的提高，手机电脑越来越普及，

人们的娱乐休闲方式越发多元化。游戏产业也开始在中国这片古老的土地上大放异彩。市场上专注于开发游戏的公司也有上千家。根据2019年中研网发布的报告显示。亚太地区已经成为全球游戏市场的增长引擎。2017年亚太地区游戏市场规模达到512亿美元，较2016年增长9.2%，全球市场占比47.10%。目前，中国、日本和韩国分别占据亚太地区市场规模前三名，未来几年内，还将保持稳定增长。国内最赚钱的互联网公司“腾讯”的很大一部分营收就是来自游戏。随着游戏市场的大火，游戏直播也成为了一大热门行业。国内大火的互联网独角兽公司“快手”大部分营收就是来自游戏直播时网友刷的礼物。自从2016年直播大火后，游戏直播就一直是直播行业的顶梁柱，越来越多玩家有条件时就玩游戏，没条件时就打开游戏直播看别人玩。由此可见国产游戏行业的大火。

* 1. 国产游戏的未来发展远景

众所周知，电子游戏起源于西方，距今已有将近50年的历史。但对于中国来说，国产游戏发源于改革开放初期，距今不过二十年。然而，现在谁也不敢小瞧这个才出生不久的小东西。电子游戏经过在中国发展二十余年，突飞猛进。国产游戏在这二十余年里主要经过三个阶段，从1994年国产游戏的元年的“单机

游戏时代”，到二十一世纪初期开始的“网游时代”，再到最近火爆的“端游时代”。国产游戏一步一个脚印，从无到有，从弱到强，如今，国产游戏开始走出国门，前两年，腾讯公司出品的“刺激战场”屡登日本游戏下载榜榜首便是最好的证明。

如今，随着5G的发展和VR（虚拟现实技术），AR（增强现实技术）技术的愈发成熟，未来的国产游戏势必会逐渐打破虚拟与现实之间的铜墙铁壁，虚拟游戏将会愈发真实，从国外的游戏行业新闻来看，日本早已开始了征程，switch的大火预示着体感游戏的未来前途无量。精灵可宝梦游戏引起全球玩家为之沸腾同样预示着未来增强现实类游戏的火热。随着国产游戏的高歌猛进，我们有理由相信，未来国产游戏将继续在世界游戏这个舞台大放异彩。

1. 开发工具的介绍
   1. Python语言简介

本次毕业设计的程序语言为python，版本为3.8，开发平台为window10（64位）。

Python的是一种跨平台的计算机程序语言，它结合了解释性、编译性和面向对象性等特点，深受广大程序员与编程爱好者的喜爱。目前python版本大致分为两类，一类是早期的python2+，一类是近些年推出的python3+。二者大致相同，不过python3+并不兼容老版本python2+，python3+拥有更加完善的功能，所以越来越多程序员开始弃python2+转为python3+。本次毕业设计选用的正是python3+。

Python作为一种热门的计算机编程语言，它所运用的领域也极其广泛。目前主要应用于以下几个领域：web开发，人工智能，后端开发，网络爬虫，科学计算与统计等

* 1. pygame库简介

pygame是一个利用SDL库写的游戏库。SDL是一个跨平台库，可以访问计算机的硬件组件，比如声音，视频，文件输入输出等。由此Pygame同样也可以操作计算机硬件组件。Pygame库专为游戏开发者服务，库中有众多开发游戏所必备的模块，有绘制图像的draw模块，管理事件的event模块，读取键盘操作key模块，操纵声音的mixer模块，读取鼠标的mouse模块等等。对于开发简单的2D游戏，pygame库绰绰有余。

* 1. Pycharm软件介绍

Pycharm是一款功能强大的python编辑器，其带有一整套可以帮助用户提高开发效率的工具，比如，调试，语法高亮，智能提示等。

* 1. Arduino软件介绍

Arduino IDE基于precessing IDE开发，arduino语言基于wiring语言开发，简单易学，对于初学者来说特别友好，该语言类似c语言，但简化了很多复杂的理论概念。配合arduino开发板进行开发使用，简单易学，所以在电子爱好者的世界里越来越流行。

1. 需求分析
   1. 可行性分析

该设计是以python语言作为编程语言，python语言有众多库方便开发者调用，例如本次开发所需要的pygame库和sys库以及serial库等。在现有的技术条件下，游戏主要部分的开发具有充足的可行性。其次，arduino开发板配合arduinoIDE程序开发可以轻易读取到红外检测模块所检测的数据。将数据通过下载线传递给电脑，python程序通过serial库中的函数可以轻易读取到单片机传过来的数据，进而调整游戏进程。因此，在硬件检测部分也具有充足的可行性。

* 1. 系统平台要求

Window7及以上版本系统，计算机必须安装有python3+的运行环境及编译器，必须要提前加载好pygame和pyserial包才能正常运行。

* 1. 硬件要求

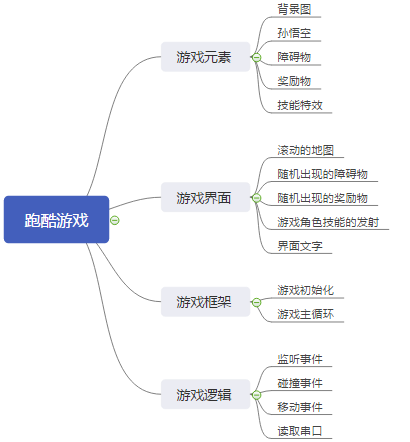
红外对管传感器一个，arduino开发板一个，下载线一个，杜邦线若干个

1. 概要设计
   1. 功能介绍

游戏规则：玩家操纵游戏角色孙悟空发射技能龟派气功波或者跳跃来避开障碍物龙，当地图上方出现奖励物时，孙悟空跳起捕获奖励物，捕获一个奖励物奖励一分，当获得分数超过200分时，屏幕上显示“恭喜你赢了”。孙悟空遇到障碍物如果既没跳起来又没发射龟派气功波，则孙悟空30%掉血。当孙悟空所获积分超过5分的时候，孙悟空便可以发射龟派气功波，每发射一次积分减少5分。孙悟空头上有一个显示生命值的横向柱状图，简称血条。当血条低于30%时，血条由绿色变成红色，当血条等于或者小于0，则游戏结束。为了增加游戏的可持续性，每隔3秒，血条恢复1%。恢复到100%则不再增加。

游戏中动画的原理。我们在游戏初始化中设置好游戏时钟，然后在游戏循环中调用游戏时钟的tick方法，配置tick方法的参数，也就是每秒绘制的次数，简称帧率。根据研究表明，人眼在时速每秒播放20张图片状况下是分不清播放的是图片还是视频。这也是电影的工作原理。游戏的帧率一般比较高，这样才能保障游戏的流畅性。本次毕业设计的游戏帧率设为60帧，也就是每秒刷新60次屏幕。

* 1. 软件概念设计图

图4-1 游戏框架

* + 1. 游戏元素设计

这款跑酷游戏的主要元素有五种：背景图，孙悟空，障碍物，奖励物，技能特效。对应五张图片，大部分图片为png格式，有利于与背景图融为一体。这五种元素的各种属性、设置，我编写了五个类用来存放。这五个类放在同一个python文件里面，该python文件名叫sprite.py。因为这五个类都继承了pygame.sprite.Sprite父类。下面介绍一下这个父类。为了简化开发过程，pygame提供了两个类，一个是存放精灵的pygame.sprite.Sprite,一个是存放精灵组的pygame.sprite.Group类。

在游戏开发里，我们一般把显示图像的对象叫做精灵Sprite。精灵一般需要设置两个主要属性：image属性和rect属性。精灵类里有一个update（）方法默认下不进行任何操作，我们一般需要自动更新精灵的位置，所有需要子类重写update（）方法。一个精灵组可以包含多个精灵对象，调用精灵族的update方法会自动将组中所有精灵更新一下。其次，精灵组对象的draw（）方法同样可以将组中每个精灵绘制一遍。当需要对大批量精灵进行统一操作的时候，精灵组对象无疑是最好的操作选择。

* + 1. 游戏界面设计

滚动的地图。首先定义一个滚动的地图类。然后在主程序游戏初始化里面实例化两个地图对象。在主程序游戏循环里调用update和rolling方法不断改变两个地图的位置并渲染到游戏界面。通过改变两个地图对象的横向坐标，让两个地图对象横向首尾连接，一个对象显示在游戏窗口，另一个地图对象在界面外预备显示，第一个地图对象显示完了又接在第二个地图后面预备显示，由此，地图开始滚动起来了，这个时候虽然游戏角色横向坐标不变，但在玩家眼里，游戏角色开始奔跑起来了。

class MyMap(pygame.sprite.Sprite):

def \_\_init\_\_(self, x, y):

self.x = x

self.y = y

self.bg = pygame.image.load("bj.jpg")#加载背景图

def map\_rolling(self):#定义让地图滚起来的函数

if self.x < -954:

self.x = 954

else:

self.x -= 6

def map\_update(self,screen):#把背景图订在游戏窗口里

screen.blit(self.bg, (self.x, self.y))

随机出现的障碍物和奖励物。先在游戏初始化中实例化奖励物和障碍物精灵组对象。要想隔一段时间随机出现障碍物和奖励物，首先要在游戏初始化里面设置一下定时器事件，以及配置相关参数，比如定时多久一次触发。经过多次测试，我设置为每3秒触发一次定时器。定时器监听代码一般放在游戏主循环里。如下：

elif event.type==CREACT:#当定时器事件触发

if energy<1:#如果生命值小于1，则每隔3秒生命值加1%

energy+=0.01

if len(fruits)<5:

#如果奖励物精灵组数量小于5，则每隔3秒增加一次奖励物

gf.add\_fruit(screen,fruits)

if len(dragons)<3:

#如果障碍物精灵组数量少于3，则每隔3秒增加一次障碍物

gf.add\_dragon(screen, dragons)

其次，为了体现生成的障碍物精灵和奖励物精灵的随机性。我们在选择生成精灵对象的位置上随机安排，这部分代码在各个精灵类里面。然后渲染到游戏界面上。生成及渲染函数代码如下：

def add\_dragon(screen,dragons):#生成障碍物精灵函数

e1=Dragon(screen)

dragons.add(e1)

# print(len(dragons))

def add\_fruit(screen,fruits):#生成奖励物精灵函数

for i in range(random.randint(1,5)):

#随机生成1-5个奖励物精灵

f1=Fruit(screen)

fruits.add(f1)

def update\_f1(screen,fruits):#把奖励物精灵渲染到游戏界面

for fruit in fruits.sprites():

fruit.blitme()

# print(fruit.rect.height)

# print(len(fruits))

fruits.update()

for fruit in fruits.copy():

if fruit.rect.right<0:

#如果奖励物精灵飞离屏幕，则销毁该精灵，以增加游戏流畅度

fruits.remove(fruit)

def update\_e1(screen,dragons):#把障碍物渲染到游戏界面

for dragon in dragons.sprites():

dragon.blitme()

dragons.update()

for dragon in dragons.copy():

if dragon.rect.right<0:

#如果障碍物精灵飞离屏幕，则销毁该精灵，以增加游戏流畅度

dragons.remove(dragon)

游戏角色技能的发射。技能波的设置跟障碍物类似。主要的特性代码也都放在其相应的精灵类里。其生成代码放在了主程序循环里的监听事件里面，监听键盘是否按下，按下时，触发其响应函数生成一个技能波精灵并渲染到游戏界面上。其生成与渲染的代码如下：

#主循环里的监听代码

elif event.type==pygame.KEYUP:

if event.key==pygame.K\_b:#监听键盘是否按下键b

if score>=5:#只有分数超过5分时才能发射技能

new\_arrow=Arrow(screen,swk)

arrows.add(new\_arrow)

score-=5#发射完减5分

#更新屏幕显示函数，该函数渲染了游戏主角孙悟空和技能波

def update\_screen(screen,swk,arrows,energy):

swk.blitme(screen,energy)

for arrow in arrows.sprites():

arrow.blitme()

arrows.update()

# print(len(arrows))

for arrow in arrows.copy():

#当龟派气功波飞出游戏界面，则销毁相应的技能波精灵，以增加游戏的流畅性

if arrow.rect.left > 954:

arrows.remove(arrow)

界面文字的设置。因为该界面文字显示分数，分数随着游戏的进行会不断变化，所以界面文字的设置放在了游戏主循环当中。代码如下：

font = pygame.font.SysFont('方正舒体', 32)

#创建一个字体文件对象，设置文字字体以及大小

text\_surface = font.render(u'你的分数：%d' % score, True, (1, 200, 200))#u表示用Unicode编码方式，第二个参数表示是否抗锯齿，最后那个参数是设置字体颜色。

text\_rect = text\_surface.get\_rect()

#获取界面文字的位置

text\_rect.left = swk.screen\_rect.left

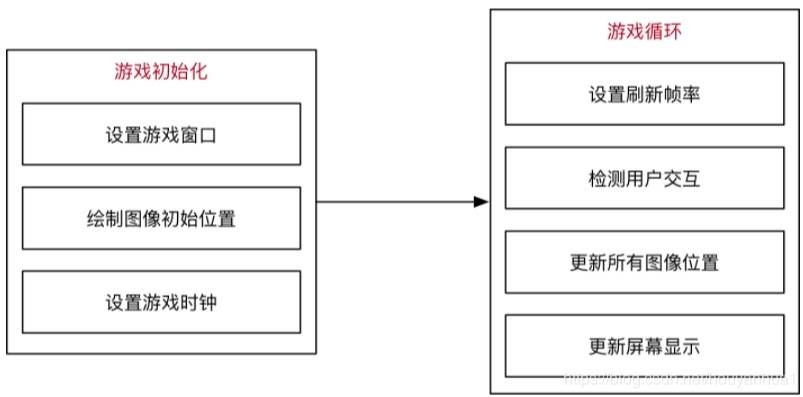
text\_rect.top = swk.screen\_rect.top

screen.blit(text\_surface, text\_rect)

#设置界面文字的位置，并将界面文字渲染到窗口。

* + 1. 游戏框架设计

游戏框架基本分为两大部分：游戏初始化部分和游戏循环部分。

图4-2 游戏主函数框架

游戏初始化部分。游戏初始化首先需要创建一个游戏窗口。Pygame有专门用于创建和管理游戏窗口的pygame.display模块。该模块提供了两个主要方法：set\_mode方法和update方法。Set\_mode方法用于初始化显示窗口，update方法是用来刷新屏幕显示内容，pygame还提供了一个类似该方法的方法flip。Flip方法主要用于局部刷新，而update方法用于全局刷新。在游戏结束时需要调quit方法。

#游戏初始化代码

pygame.init()#使用pygame所有功能之前要先初始化pygame

CREACT = pygame.USEREVENT#设置游戏定时器事件

screen=pygame.display.set\_mode((Width,Height),0,32)

#创建并配置游戏窗口宽度与高度

pygame.display.set\_caption("孙悟空快跑")#在标题栏中给游戏起名

clock=pygame.time.Clock()#设置游戏时钟

arrows = Group()#分别给几种需要多个一起出现的精灵设置精灵组

dragons=Group()

fruits=Group()

score=0#初始化分数

bg1 = MyMap(0, 0)#实例化两个地图对象

bg2 = MyMap(954, 0)

swk = Swk(screen)#实例化游戏角色孙悟空

pygame.time.set\_timer(CREACT, 3000)#设置定时器监听事件的时间间隔

player\_jumping=False#定义一些变量

v0=0

first = swk.screen\_rect.centery - 26#定义孙悟空的初始纵向位置

energy=1.0#定义初始生命值为1

ser=serial.Serial("COM4",115200,timeout=0.1)#实例化串口对象并配置串口基本信息。比如端口号、波特率、最多超时。

在游戏循环里，游戏中的元素不断改变，我们也不断刷新界面，游戏则不断进行下去。直到游戏角色生命值降为0，我们改变标志位退出循环，游戏结束。

while running:

clock.tick(60)#设置游戏帧率

for event in pygame.event.get():#监听各种事件

if event.type == pygame.QUIT:

pygame.quit()

sys.exit()

elif event.type==CREACT:#定时器事件

if energy<1:

energy+=0.01

if len(fruits)<5:

gf.add\_fruit(screen,fruits)

if len(dragons)<3:

gf.add\_dragon(screen, dragons)

elif event.type==pygame.KEYUP:

if event.key==pygame.K\_b:

if score>=5:

new\_arrow=Arrow(screen,swk)

arrows.add(new\_arrow)

score-=5

elif event.key==pygame.K\_ESCAPE:

#当按下ESC键，游戏暂停。

input()

elif event.key==pygame.K\_SPACE:

#当按下空格键，孙悟空跳起

if swk.rect.centery==first:

#只有当孙悟空位于地面的时候，按下空格键才会响应。此举防止连续跳。

v0 = -10.0

#设置初始速度为-10,pygame的坐标系以界面左上点为原点，向下移动，纵坐标加大，向上移动，纵坐标减小。所以初速度为负值，游戏角色向上跳。

print('在地面')

player\_jumping=True

data = ser.readline()#读取串口数据

r = str(data)

st = re.sub("\D", "", r)#用正则表达式提取串口数据中的数字

s = int(st)#将提取的数字强转为整型，然后才能比较

if s<100:当串口数据低于100时

if swk.rect.centery == first:#当孙悟空在地面时

v0 = -10.0

print('在地面')

player\_jumping = True#跳起

v0+=0.25#每秒60次刷新，每次刷新跳起速度增加0.25

if player\_jumping:#当跳起时

swk.centerY+=v0#纵向坐标不断移动

swk.rect.centery=int(swk.centerY)

if swk.rect.centery >= first:#当跳回到地面时

player\_jumping=False#禁止继续往下跳

swk.rect.centery=first

energy,score=gf.collide(swk, arrows, dragons, fruits,score,energy)#碰撞检测函数

bg1.map\_update(screen)#将地图一渲染到游戏窗口

bg2.map\_update(screen)#将地图二渲染到游戏窗口

bg1.map\_rolling()#改变地图一 横向坐标，让地图呈现滚动状

bg2.map\_rolling()#改变地图二横向坐标，让地图滚起来

font = pygame.font.SysFont('方正舒体', 32)

text\_surface = font.render(u'你的分数：%d' % score, True, (1, 200, 200))

text\_rect = text\_surface.get\_rect()

text\_rect.left = swk.screen\_rect.left

text\_rect.top = swk.screen\_rect.top

screen.blit(text\_surface, text\_rect)

gf.update\_screen(screen,swk, arrows,energy)

#将孙悟空、技能波、血条渲染到游戏界面

gf.update\_e1(screen,dragons)

#将障碍物龙渲染到游戏界面

gf.update\_f1(screen,fruits)

#将奖励物渲染到游戏界面

if energy<=0:

#当生命值为0时，标志位改为False，跳出游戏循环，游戏结束

running=False

# pygame.quit()

if score>200:

#当分数超过200分，游戏界面的正上方显示“恭喜你赢了！”

print('you win!')

tex= font.render(u'恭喜你赢了！', True, (250, 100, 100))

tex\_rect = text\_surface.get\_rect()

tex\_rect.centerx = swk.screen\_rect.centerx

tex\_rect.top = swk.screen\_rect.top

screen.blit(tex, tex\_rect)

pygame.display.update()#刷新游戏界面

* + 1. 游戏逻辑设计

监听事件。本游戏的监听主要分为四个部分：监听键盘，监听鼠标，监听定时器，监听串口。监听事件是玩家与游戏之间重要的桥梁，方便玩家更好操纵游戏角色。为了及时响应玩家的操作，监听事件代码一般放在游戏主循环当中。

碰撞事件。为了让主程序尽可能简单易读，本游戏的碰撞事件函数代码放在另一个gf.py程序当中，该程序还包括很多移动事件等可以放在外部程序的大多数操作函数。Pygame提供了三种碰撞检测方法。第一种，精灵与精灵之间的碰撞检测：pygame.sprite.colide\_rect(first,second)。该方法输入的两个参数为两个需要检测碰撞的精灵对象。返回值是一个布尔值，如果两者发生了碰撞，则返回值为True，如果两者没有发生碰撞，则返回值为False。第二种，精灵与精灵组之间的碰撞检测：pygame.sprite.spritecollide(sprite,group,True)。该方法第一个输入参数是精灵对象 ，第二个输入参数是精灵组对象。第三个输入参数必须是一个布尔值，当第三个输入参数为True时，精灵与精灵组碰撞后，精灵组中发生了碰撞的精灵对象立马消失。如果第三个输入参数为False时，精灵组中发生碰撞的精灵不会消失。该检测方法的返回值为包含组中所有被碰撞的精灵对象的一个列表。第三种，精灵组与精灵族之间的碰撞检测：pygame.sprite.groupcollide(group1,group2,True,True)。该方法第一个输入参数是第一个精灵组对象，第二个输入参数是第二个精灵组对象，第三个输入参数表示碰撞后第一个精灵组组中相应对象是否消失，第三个输入参数表示碰撞后第二个精灵组组中对应对象是否消失。该方法的返回值是一个字典，键为第一个精灵组每个对象，值为第二个精灵组组中与之发生碰撞的对象。

def collide(swk,arrows,dragons,fruits,score,energy):

#定义碰撞函数，并导入所需参数

pygame.sprite.groupcollide(arrows, dragons, True, True)

#技能波与障碍物发生碰撞后，两者都立马消失。属于第三种碰撞。

sprite\_list=pygame.sprite.spritecollide(swk,fruits,True)

#检测孙悟空与奖励物是否碰撞，返回包含发生碰撞的奖励物对象列表

#属于第二种碰撞

score+=len(sprite\_list)

#检测被碰撞奖励物对象列表的对象数量，并以此为标准，奖励玩家相应分数

if pygame.sprite.spritecollide(swk,dragons,True):

#如果孙悟空与障碍物碰上了

print ('碰上了')

swk.rect.centerx-=10

#孙悟空后退10个像素表示受伤

energy-=0.25

#孙悟空的生命值减少25%

return energy,score

返回碰撞后剩下的生命值以及分数

读取串口。在游戏初始化里同时初始化串口，设置对应的端口号、波特率、最多超时时间。然后在游戏主循环里读取串口数据，当串口数据值低于100且孙悟空在地面时，孙悟空跳起，来避开障碍物或者捕获奖励物。

data = ser.readline()#读取串口数据

r = str(data)

st = re.sub("\D", "", r)[0,2]#用正则表达式提取三位串口数据中的数字

s = int(st)#将提取的数字强转为整型，然后才能比较

if s>50:当串口数据大于50时

if swk.rect.centery == first:#当孙悟空在地面时

v0 = -10.0

print('在地面')

player\_jumping = True#跳起

1. 详细设计
   1. 主要类及模块
      1. 主要模块及其作用

|  |  |
| --- | --- |
| 模块 | 作用 |
| pygame | 有众多用于开发游戏的函数。包含众多对图像声音处理函数 |
| sys | 包含众多与系统交互的函数 |
| random | 用于生成随机数 |
| pyserial | 用于读取串口数据 |
| re | 正则表达式模块 |
| time | 时间模块 |
| gf | 游戏功能函数 |
| sprite | 存放各种游戏元素类 |

图5-1 主要模块及作用图

* + 1. 主要类及其作用

在本次毕业设计里，我把主要包含各种元素的类统一放在sprite.py程序里面。

#孙悟空类，继承精灵类

class Swk(Sprite):

def \_\_init\_\_(self,screen):

super().\_\_init\_\_()

self.screen=screen

self.image=pygame.image.load('qiqi.png')

self.screen\_rect = screen.get\_rect()

self.rect=self.image.get\_rect()

self.rect.centerx = (2/3)\*self.screen\_rect.centerx

self.rect.centery = self.screen\_rect.centery-26

self.centerX=float(self.rect.centerx)

self.centerY=float(self.rect.centery)

#确定孙悟空的位置不变

def blitme(self,screen,energy):

self.screen.blit(self.image,self.rect)

#将孙悟空渲染到游戏界面上 pygame.draw.line(screen,(0,0,0),(self.rect.left,self.rect.top-5),(self.rect.right,self.rect.top-5),4)

#在孙悟空头顶位置画一条线表示生命槽，传入五个参数：绘制所在地，绘制线条颜色为黑色，绘制线条起始位置，绘制线条的终止位置，绘制线条的宽度

if energy>0.3:#当生命值大于30%时，

energy\_color=(0,255,0)#生命条颜色为绿

else:#否则生命条颜色为红

energy\_color=(255,0,0)

#在孙悟空头顶上再画一条直线表示生命值，线条的长度与生命值成正比 pygame.draw.line(screen,energy\_color,(self.rect.left,self.rect.top-5),(self.rect.left+self.rect.width\*energy,self.rect.top-5),4)

#障碍物类

class Dragon(Sprite):#继承精灵类

def \_\_init\_\_(self, screen):

super().\_\_init\_\_()

self.screen = screen

self.image = pygame.image.load('dragon1.png')

self.rect = self.image.get\_rect()

self.screen\_rect = screen.get\_rect()

self.rect.centery = self.screen\_rect.centery-10

#纵坐标固定

self.rect.right = random.randrange(self.screen\_rect.right,self.screen\_rect.right+5\*self.rect.width,self.rect.width)

#横坐标在游戏窗口右侧随机分布

# 不断向左移动

def update(self):

self.x = float(self.rect.centerx)

self.y = float(self.rect.centery)

self.x -= 6

self.rect.centerx = self.x

self.rect.centery = self.y

#把障碍物渲染到游戏界面

def blitme(self):

self.screen.blit(self.image, self.rect)

#奖励物类

class Fruit(Sprite):

def \_\_init\_\_(self,screen):

super().\_\_init\_\_()

self.screen=screen

self.image=pygame.image.load('fruit.bmp')

self.rect=self.image.get\_rect()

self.screen\_rect = screen.get\_rect()

self.rect.top=self.screen\_rect.top+random.randrange(2\*self.rect.height,5\*self.rect.height,self.rect.height)

#随机在窗口右方产生合适位置

self.rect.left=self.screen\_rect.right+random.randrange(0,4\*self.rect.width,self.rect.width)

# 不断向左移动

def update(self):

self.x=float(self.rect.centerx)

self.x-=6

self.rect.centerx=self.x

# 把水果画上去

def blitme(self):

self.screen.blit(self.image,self.rect)

#技能波类

class Arrow(Sprite):

def \_\_init\_\_(self,screen,swk):

super().\_\_init\_\_()

self.screen = screen

self.image = pygame.image.load('blue.png')

self.rect = self.image.get\_rect()

self.rect.left=swk.rect.right

self.rect.centery=swk.rect.centery

#确定技能波的生成位置，在孙悟空的右边中间位置

self.x=self.rect.centerx

def update(self):#更新技能波位置

self.x+=3

self.rect.centerx=self.x

def blitme(self):#将技能波渲染到游戏界面

self.screen.blit(self.image,self.rect)

1. 硬件电路及程序设计
   1. 主要硬件介绍

本次毕设用到的单片机为Arduino nano，是一款基于Microchip ATmega328P 8位微控制芯片的智能硬件开发板，尺寸虽小，功能强悍。

图6-1 Arduino单片机图

本次毕设所用到的传感器是红外对管传感器，。其具有一对红外线发射与接收管，发射管发射出一定频率的红外线，当前方遇到障碍物时，红外线反射回来被接收管接收，然后经过比较器电路处理，传感器上方绿色指示灯就会亮起来，同时输出接口输入一个低电平信号。单片机检测到该信号后经过串口把信号发射给电脑，python经过加载的pyserial模块来获取信号，进而操纵游戏。

* 1. 单片机及其程序介绍

int analogPin=A7;#A7是arduino nano的一个模拟输入引脚

int iVal=0;#初始化捕获变量

void setup() {

Serial.begin(115200);#初始化串口

}

void loop() {#程序主循环

iVal=analogRead(analogPin);#将获取到的输入量赋给变量iVal

Serial.println(iVal);#通过串口向电脑传输数据

delay(100);#延时0.1s

}

1. 测试与运行
   1. 运行软件程序及调试

首先打开pycharm，加载程序文件以及相关图片素材。然后加载所必备的模块，比如pygame等。加载完模块就可以直接运行main.py程序。

当敲击空格键时，游戏角色孙悟空就会跳起来。如下图所示。

当接上检测模块和单片机，正常运行后，在传感器前面跳起，游戏角色孙悟空也会跳起。

当敲击字母b键，孙悟空就会发射技能波，可以消灭前方障碍物。

当敲击ESC键，游戏暂停。

当移动鼠标到关闭符号并按下，游戏结束。

调试中遇到的问题不少，比如刚开始时游戏角色不显示，只有滚动的地图显示。仔细检查代码才发现 ，主循环中渲染函数代码的先后顺序决定了谁被谁覆盖。把地图对象的渲染函数放在前面，就不会覆盖后面的游戏角色了。

调试程序的时候我一般用print函数追踪变量值的变化，这个习惯帮助我解决了不少棘手的bug。

游戏过程如下：

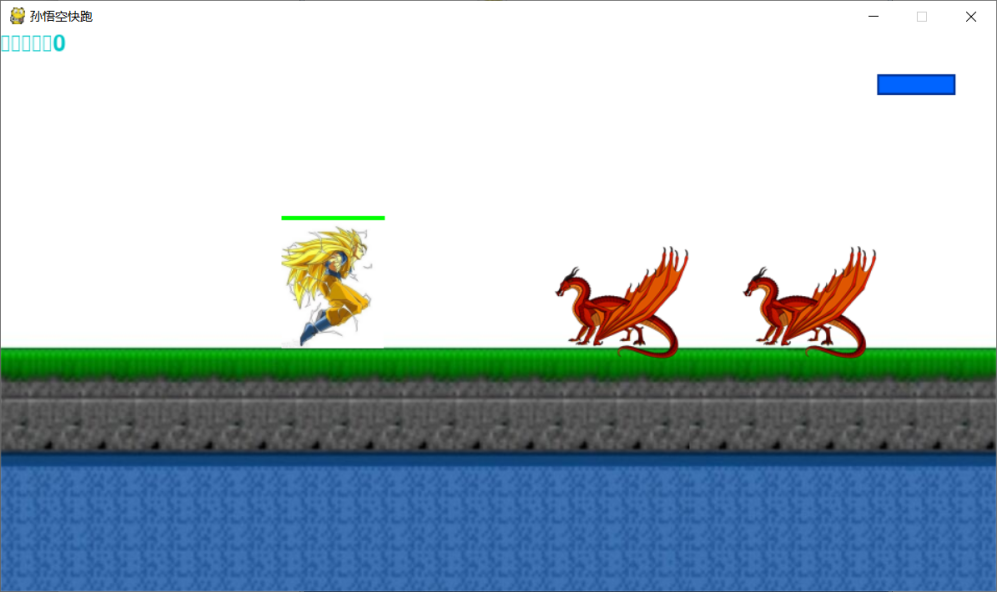


图6-2 游戏开始



图6-3 游戏角色跳跃



图6-2 游戏中

* 1. 运行硬件程序及调试

虽然硬件获取信号顺利，但传输延迟过高，每次传输时间大概1.5s。起初我试图调高波特率，然而依然不见成效。后来仔细检查代码才发现原因。我把初始化串口程序写在了python游戏主循环里面了，导致每次循环都初始化一遍串口，延迟过高，游戏界面异常卡顿。然后我把初始化串口代码放在了游戏初始化程序那一堆里面了，游戏卡顿现象立马好转了。

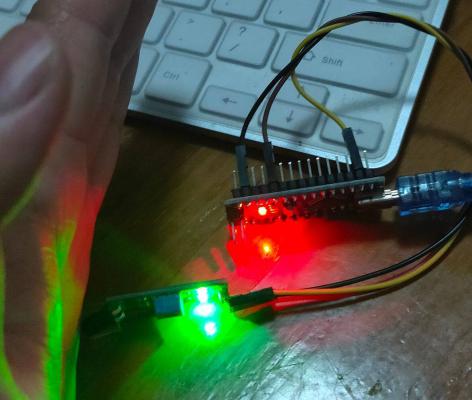


图6-2 调试图

1. 总结
   1. 遇到的问题及解决方法
      1. 如何制作合适的图片素材

在游戏制作或者PPT制作里面，我们看到的很多图片素材大部分都是png格式，png格式的图像支持透明，我们在绘制图像的时候经常为图片底色与背景图颜色不搭，显得十分突兀。而png的出现则专门为了解决这一难题。在绘制图像的时候，png图片的透明区域不会显示任何图像内容。不过如果图片下方已经有图片内容，则下方图片内容会透过透明区域显示出来，这样，层叠在一起的图片元素就能和谐共存了。然而，当我了解到png图片各种优势决定去改用png格式后，我发现我很难找到国内免费的png素材下载网站，多亏了知乎网友，我才找到了好几个国外免费的png素材下载网站。比如pngimg.com等。找到了合适的图片素材，然而这些图片素材像素太高了，直接放入游戏会超过界面尺寸。于是我又开始寻找裁剪缩小png格式图片的软件，一般的修图软件修完图会让png格式变成jpg格式，我想到用ps软件，奈何我借来的电脑实在老旧。最后翻了不少知乎网友的回答，发现美图秀秀电脑版可以对png图片进行处理，这才解决了合适的图片素材这一难题。

* + 1. 如何让障碍物和奖励物随机出现

这个问题此前困扰了很久。由于我是python新手，很多模块都没听过更没有用过。我只知道单片机有定时器函数，可以不妨碍主程序的运行自动工作。却不知道原来python也有定时器方法。查完资料后我用上了 python定时器方法，每隔3秒随机产生一次障碍物和奖励物，后期添加了自动回血功能，每隔3秒钟自动回1%的生命值。仅仅定时产生奖励物和障碍物还不够，还需要产生随机性，最好在一开始生成时位置就随机分布，所以在生成奖励物和障碍物精灵时，位置的分布都用了random函数随机产生合适的横坐标纵坐标。最好在还得在预备界面生成，这样随着地图的滚动，障碍物和奖励物也随机分布在地图上。

* + 1. 如何让孙悟空跳起来

起初我以为这个问题会很容易，以为就是简简单单用一个延时函数，让孙悟空横坐标不变，纵坐标随着时间改变就可以了。然而试了一下这样操作，结果就是进程发生堵塞，后面的操作都无法进行，等着孙悟空跳完。这种方法失败后，我想了很久才想到两个更加艰难的方法。一个是调用python的一个物理模块punk，来模仿现实中的跳跃。一个是用多线程的方法，当触发跳跃时调用另一个线程，主线程继续更新屏幕。然而两种方法都异常复杂，第一种方法还要设置刚体惯性量，摩擦系数等。第二种方法线程之间的通信也不容易。然后我回到起点，想到了用每次刷新1/60秒的条件，把每次刷新时间设为单位时间，加速度一定时，单位时间速度的改变是相同的，每次刷新的纵坐标减少速度与单位时间的乘积，然后把孙悟空刷新到新的位置上就能模拟出孙悟空的跳跃过程了。

* + 1. 如何读取串口数据不影响游戏运行

把串口初始化函数放在主循环外面就能减少读取数据的延迟，提高游戏的流畅度了。然而，放在外面后，串口数据提取字符串数字，准备强转为整形的时候总是失败，经过一次次测试后才发现，打开串口需要花费至少1.5秒，刚放在外面的时候没有延时函数导致串口读取的前几个数据都是空，整形转换自然总是失败，知道这个后，我在主循环外面延时两秒后才进入主循环，问题就解决了。顺便timeout时间也可以减小到0.1秒，大大提高了计算机读取串口数据的速度。

* + 1. 如何将提取的串口信息中的数字信息并转为整型

这里需要用到正则表达式。用正则表达式把所有非数字信息皆转化为空字符。查阅了正则表达式后这个问题很快就迎刃而解了。

* + 1. 如何绘制血条

绘制血条一开始就没头绪，本来没想加这一功能，让孙悟空三次碰撞后自动游戏结束。可总感觉这样做缺失了游戏的灵魂，玩家不能很好地掌握孙悟空的存亡，那这游戏就太无趣了。然后我说干就干，网上查资料后才发现，原来可以直接用一条黑色宽线代表血槽，然后在黑色宽线上面画一条绿色的线代表生命值。这种方法既简洁又实用。于是我就用pygame自带的drawline方法画上两条直线，一条表示血槽，一条表示生命值，就画好了血条。

* + 1. 如何选择合适的传感器和单片机

单片机其实我很早就想好了用arduino单片机，简单易操作，资料也多，关键开发软件还是免费的。不过传感器的挑选费了不少时间，最开始我是准备用超声波测传感器，后来发现还有更简单直接的传感器，先试了试淘宝买的人体红外检测传感器，然而这个传感器感应后自动延时一两秒，不适合。最后试了红外避障传感器，这回成了，当传感器前方出现障碍物，输出信号io口立马转为低电平，没有障碍物时则是高电平。

结论

毕业设计是一次宝贵的实战经历。是对大学生利用所学知识的一次综合运用。考验了很多大学生的发现问题、解决问题的能力。同时这也是一次针对我们实际工作能力的考察和具体培训过程。随着现代社会的不断迅速发展。计算机科学在生活中运用得也越来越广泛。而作为计算机科学的基础--编程语言也越来越重要，作为二十一世纪大学生，学会掌握一门编程语言很有必要。Python编程语言作为近年来发展速度最快的语言之一，在编程语言排行榜里一直名列前茅，深受广大程序员的欢迎。因此学好python语言就像获得了一把打开计算机科学大门的钥匙，可以帮助我们更好的运用计算机科学来为我们服务。

回顾这次毕业设计，我感慨万千。这次毕业设计一波三折，从最初的选题到最后的开始动笔写论文，每一步都不是一帆风顺。最开始我的选题并不是这个，而是我比较拿手的开关电源课题，然而因为疫情原因，我无法去实验室做实验，我果断换课题。因为从小梦想有一天自己能做一款游戏，我便开始准备用当下最火的python来编写一个游戏。由于有C语言基础，我学习python的过程不算太艰辛，一步步跟着B站小甲鱼学python。在这几个月里，我学到很多关于编程的知识。不仅初步掌握了一门新的语言，而且还慢慢领悟到了模块化编程的思想。慢慢的，我也养成了敲代码前先画一下流程图这个习惯，我对python的各种函数库也有了更多的了解，调用函数也越来越得心应手，甚至敲代码的速度越来越快了。这些都是我本次毕业设计的收获，我相信这些知识会让我现在乃至将来受益匪浅。

参考文献：

[1]谭浩强.C程序设计（第三版）[M].北京:清华大学出版社,2005.

[2]Eric Matthes .Python编程:从入门到实践[M].北京人民邮电出版社,2016.

[3]Bradbury.树莓派Python编程指南[M].机械工业出版社,2015.

[4]刘火良.杨森.STM32库开发实战指南[M].机械工业出版社,2013.

[5]小甲鱼. 零基础入门学习Python[M]. 北京:清华大学出版社, 2016.

[6]黄焕林.丁昊. 从零开始学习Arduino电子设计[M].机械工业出版社，2018.

致 谢

大学生涯渐行渐远，我即将背上行囊，告别我亲爱的的室友、同学和老师。在这离别之际，我衷心想感谢我的室友、同学和老师。感谢室友，四年来朝夕相处，我们互帮互助，从来没有过什么摩擦，回想这四年来，多少个欢声笑语的日子都是和你们度过，多少个失魂落魄的日子也有你们在旁边的安慰，在这里衷心感谢我的三位室友：谢海涛，徐东圣，吴杰。谢谢你们，包容我的过错，帮助我前行，愿我们的友谊天长地久。

感谢教导我的老师们，你们风雨无阻，每次都能准时站在讲堂上为我们讲课，虽然课上很多人可能心猿意马，不是玩手机就是睡觉，但你们每次都能激情澎湃，保证课堂质量。依稀记得教我高数下的那位女老师，演算时候算错了一个数，虽然没人看出来但她依旧自己指出自己的错误，向同学们道歉，这种治学严谨的态度令我印象深刻，还有教我近代史的老师，教我网球课的老师等，虽然你们的名字我不记得了，但你们认真负责的治学态度曾深深感染到我，谢谢。

教我电路的张海霞老师、马双宝老师是我的领路人，带我一步步走入电学的殿堂，影响了我的人生道路。如今我在实习岗位上回忆过往，所有的果实都是来源曾经那粒种子，指使我走上硬件工程师这条路，是几年前课堂上两位老师辛勤播种的结果。回想当年的我，刚从高中的炼狱生活里逃脱出来，眼里除了吃喝玩乐就不知道该干啥了，好在认真负责的张海霞老师和马老师经常语重心长地提醒我们，学习是人生大事，外面的世界远比这里复杂，唯有学习才能更好得安生立命。也正是对未来的未雨绸缪，让我走进实验室，走进电学的世界。

最后，再次对帮助、关心我的老师们和同学表示衷心地感谢！