**夏季学期课程设计报告**

**AVR微控制器应用实践**

设计题目： flappy bird小游戏实现

专 业： 智能、光电

姓 名： 张明昕、刘逸伦

学 号： 1611251

南开大学电子信息与光学工程学院

一、**设计目的**

掌握Atmel SAM D20系列的 ARM Cortex-M0+微控制器原理与应用和熟悉对Atmel SAMD21J18A开发板的基本操作,了解OLED显示屏的基本方法与器件原理,熟悉图片、字符显示的方法，实战进行跨平台移植代码。

**二、 设计任务**

利用已有的SAMD21J18A开发板、OLED显示屏以及所学知识，实现flappy bird 小游戏。

本系统需要实现的功能如下：

1. 将游戏图片取模，显示到OLED上。
2. 完成小鸟的动画效果，控制每一帧的绘图。
3. 实现通过外部手柄控制小鸟升降。
4. 完成flappy bird游戏逻辑，优化游戏控制。

**三、系统构成与原理**

**1.OLED显示原理**

OLED，即有机发光二极管。OLED 由于同时具备自发光，不需背光源、对比度高、厚度薄、视角广、反应速度快、可用于挠曲性面板、使用温度范围广、构造及制程较简单等优异之特性，被认为是下一代的平面显示器新兴应用技术。

OLED显示屏采用二极管阵列显示，左右有128个像素点，上下有64个像素点。我们采用的SSD1306驱动芯片把上下每8个像素点并成一页，共八页。能在OLED上显示的图片必须取模为16进制数组。数组的每一个元素是一个16进制数，转化为八位二进制数后，控制纵向的八个点的亮灭。

**2.游戏逻辑及显示原理**

按照游戏基本框架，逻辑实现将分为初始化模块、用户控制器、事件响应器、绘图模块和结束退出模块。其中绘图模块需要注意适应显示设备的硬件条件。总的来说有以下几个难点：

1. 防止闪屏

要想实现连续动画，帧与帧之间必须有切换模块。常用的一种方式是利用清屏函数完全清除上一帧，再绘制下一帧。但是，由于从清屏函数执行到下一帧播放需要一定的时间，常导致闪屏。在其它平台上开发游戏，常采用放置背景图片的方式来减小人眼感觉到的色差，从而尽可能消除闪屏。

但对于OLED来说，其本身只能显示两色，因此没法放置背景图片。为了解决闪屏的问题，我们不采用全屏清零。先找到帧与帧之间图像的差异，然后局部清除上一帧的内容，以减小闪屏。

1. 纵向连续滚动

Ssd1306芯片把纵向的像素分成了八页，纵向绘制也只支持以页为单位的坐标。也就是说其本身是不支持一个像素点一个像素点向上滚动的。考虑到游戏必须实现小鸟的连续上下移动，我们必须实现纵向连续滚动。

我们采用的方式是控制16进制图像数组。欲使小鸟往上移动一个像素点，则让图像数组的每一个相关的16进制数都左移一位，而加载的页数不变。向下则反之。

1. 翅膀动画效果实现

采用交替显示动画的不同静帧实现动画效果。在这里，为了避免连续滚动时组件帧平移不同步，必须把显示和坐标分开。即，上移时所有组件帧一起上移，但显示时交替显示。

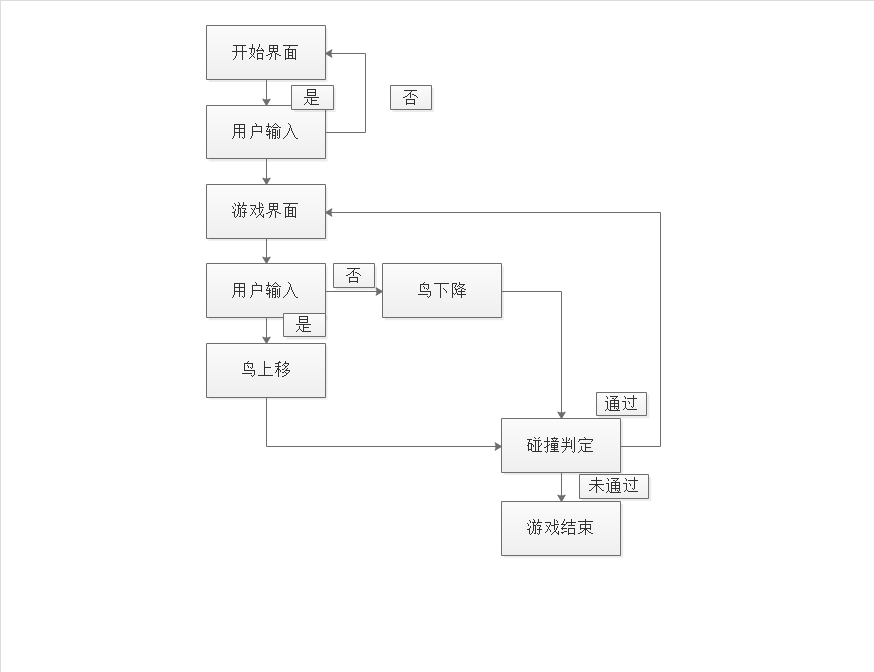
3. 模块接口定义：

1.GND 电源地  
2. VCC 电源正（3～5.5V）  
3. D0 OLED 的 D0 脚，在 SPI 和 IIC 通信中为时钟管脚  
4. D1 OLED 的 D1 脚，在 SPI 和 IIC 通信中为数据管脚  
5. RES OLED 的 RES#脚，用来复位（低电平复位）  
6. DC OLED 的 D/C#E 脚，数据和命令控制管脚

6. CS OLED 的 CS# 脚，片选管脚

7.CONTROL 用户控制器管脚

4.工作流程



**四、流程简介**

（1）OLED环境搭建

1. 静态图片载入
2. 动态效果实现

连续滚动、帧切换控制。

1. 墙体控制函数

在屏幕右边生成高度随机的墙体，并控制其往左移动。

1. 鸟控制函数

翅膀扇动效果实现、带有加速度的上下移动实现。

1. 用户控制函数

用户按一下按钮，小鸟获得一个向上的初速度，直到小鸟到达顶点之前，用户无法再次控制小鸟。

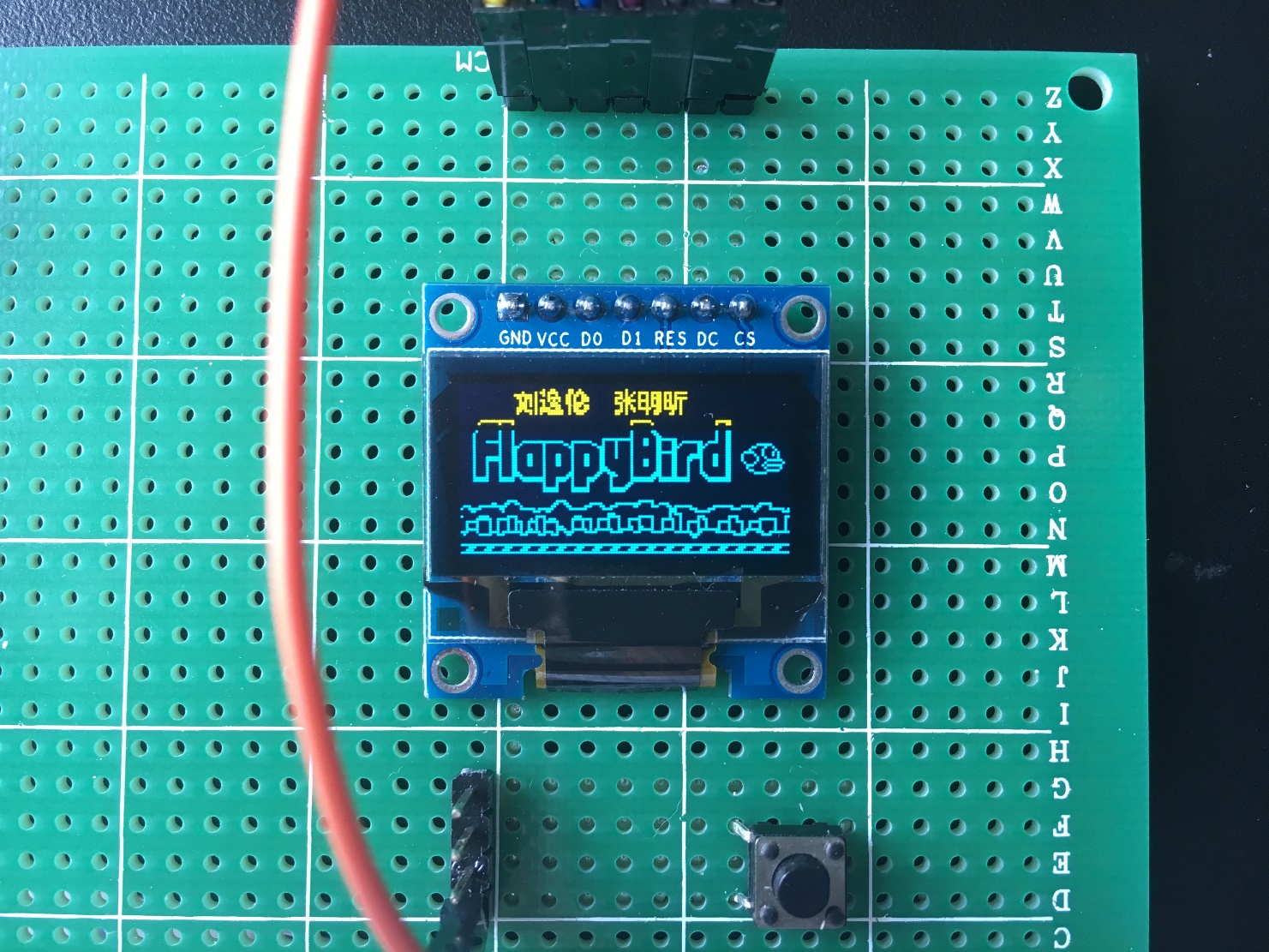
1. 事件响应器

实现碰撞判定、界面管理、记录分数等。

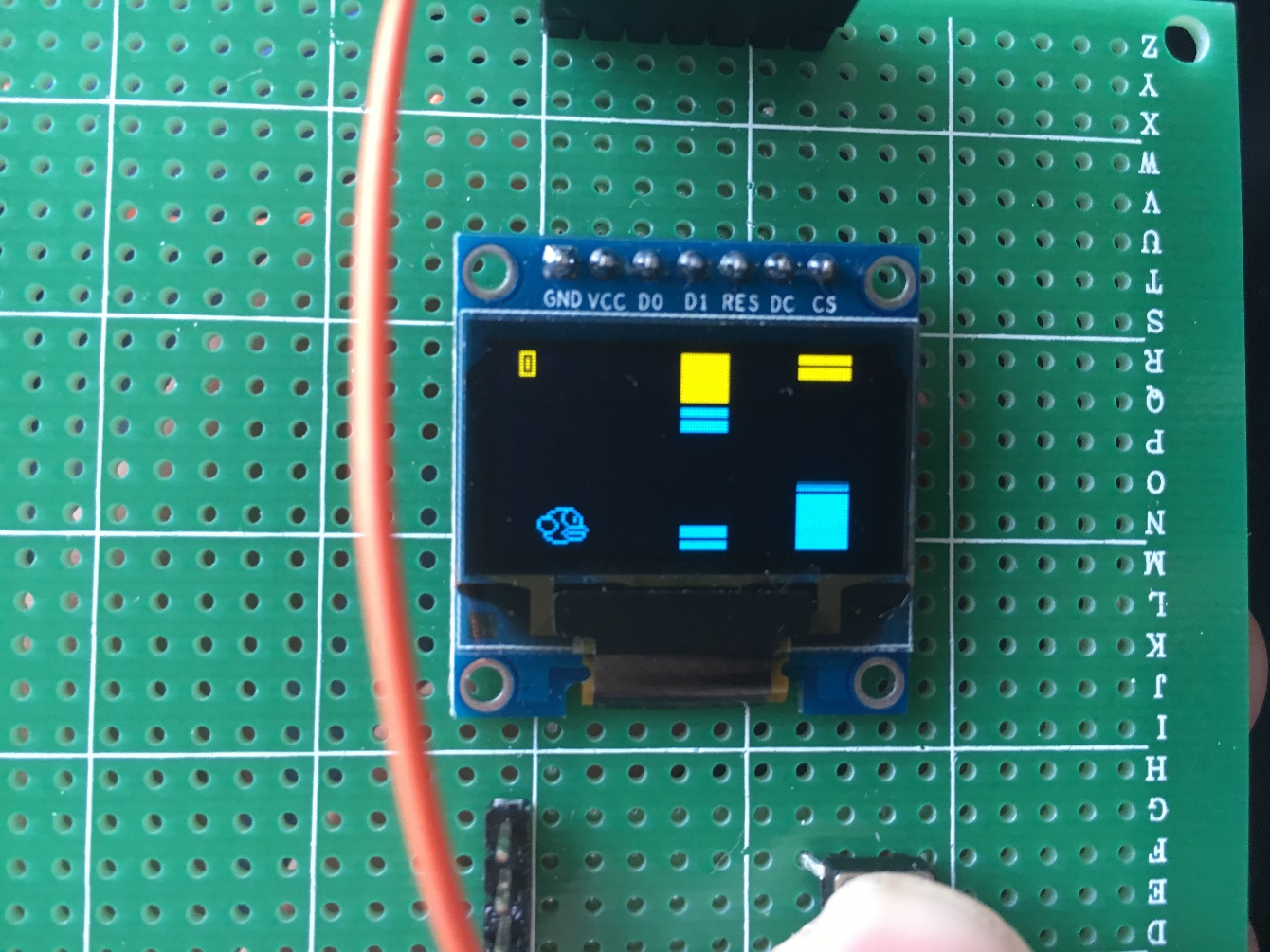
1. 开始和结束界面

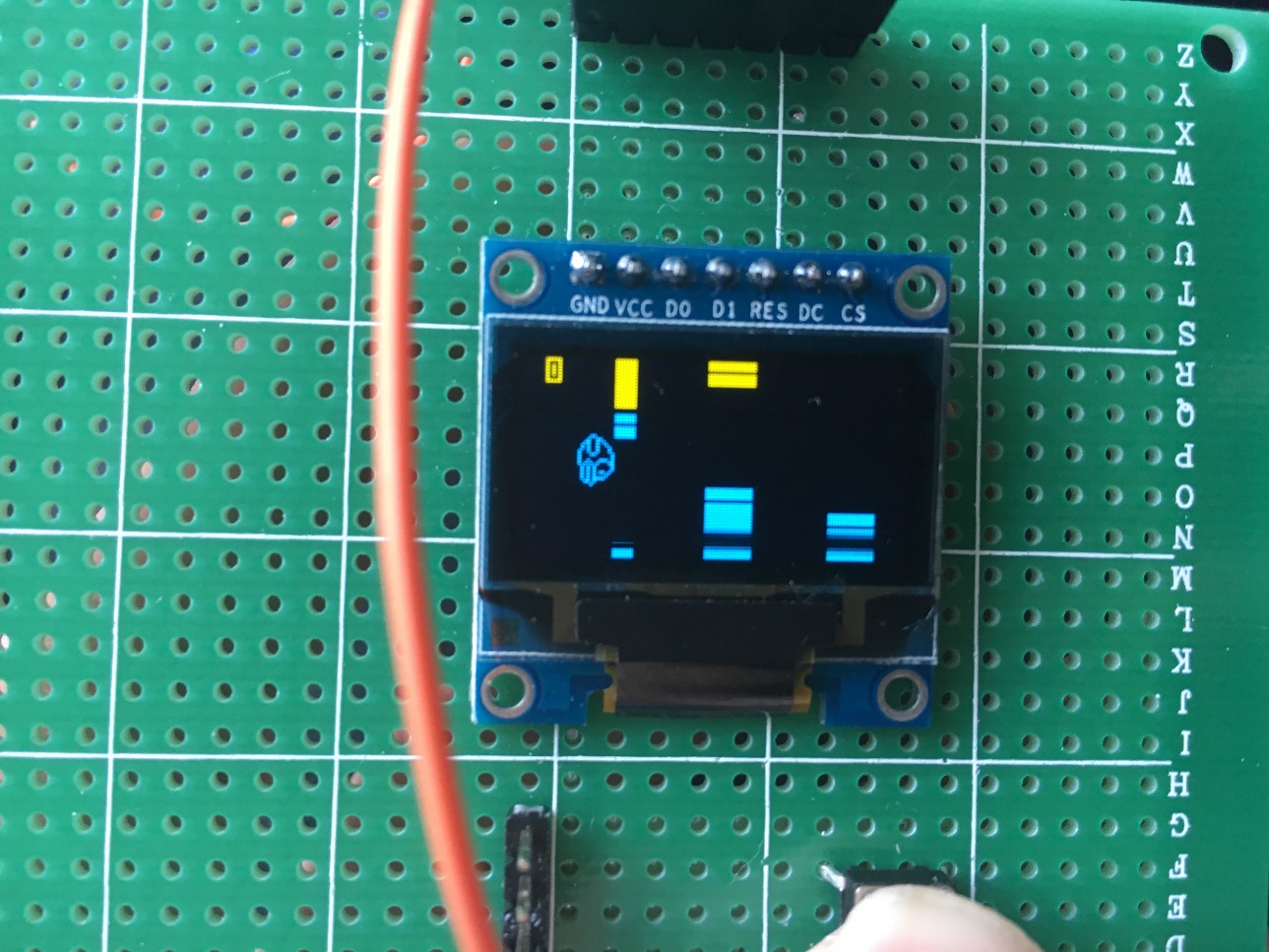
**四、运行结果**

1、开始界面



2、游戏界面





3、死亡界面

****