扫雷游戏项目说明文档

作者：

黄程远

目录：

1.游戏功能说明

2.游戏设计过程

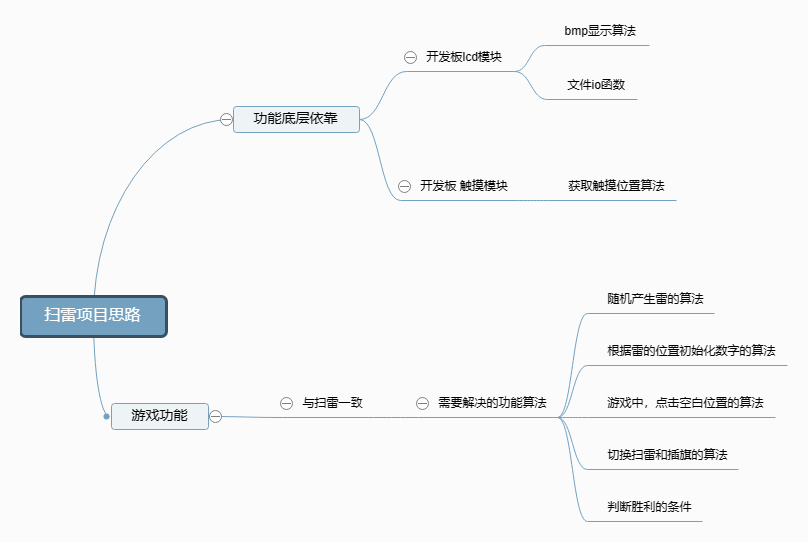
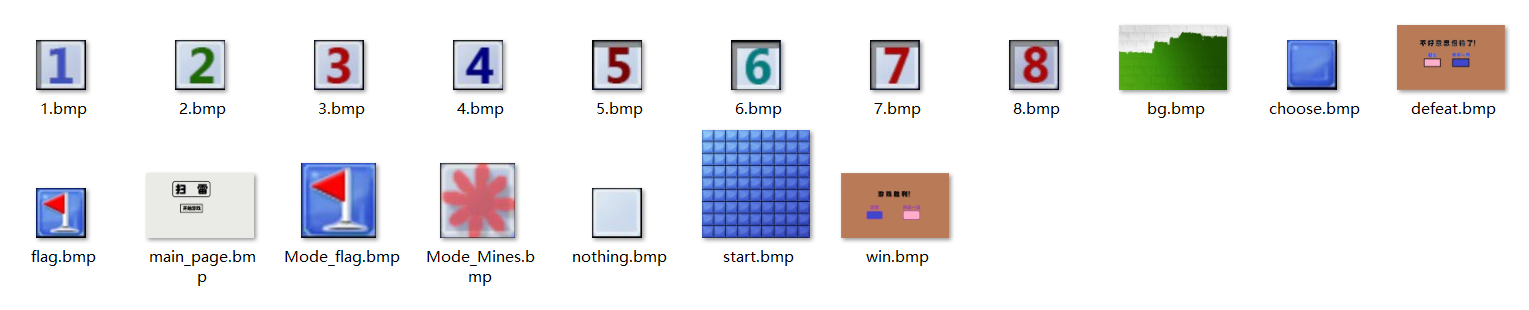
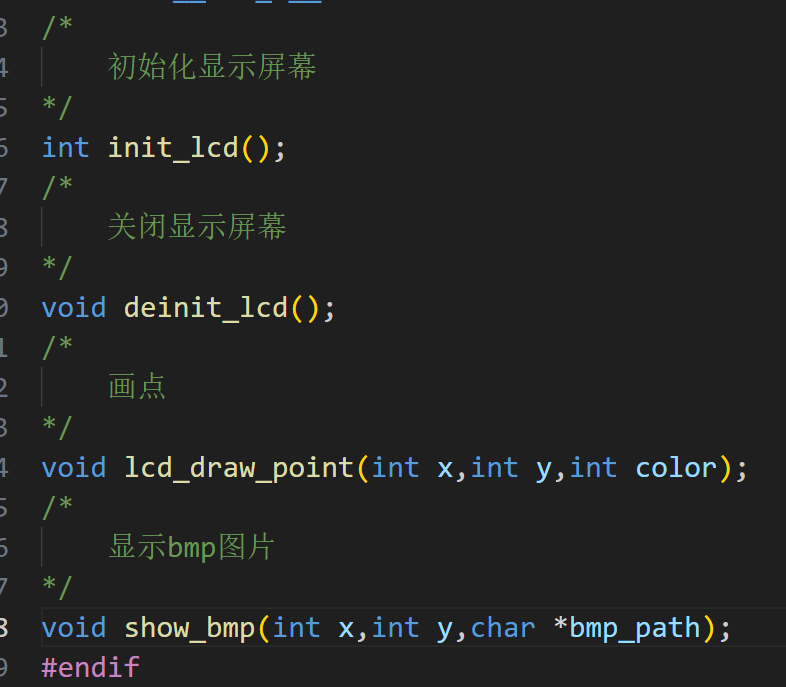
3.设计遇到的问题与解决办法

4.总结

游戏功能说明

1. **主界面，点击开始游戏进入游戏界面**
2. **游戏界面，左侧显示游戏内容，右侧则是游戏ui，目前有一个按键，用于切换扫雷和插旗模式**
3. **扫雷模式下点击游戏区的任意格子，根据格子的内容显示不同结果，是雷就直接游戏失败，是旗子则无法有效点击，是数字会直接显示，是空白格则会显示周围所有的非雷非旗格**
4. **游戏胜利或者游戏失败均可选择回到主界面或者再次进行游戏**

游戏设计过程

1. **我首先画出了思维导图，确定了项目的难点和必须的硬件软件功能，并且创建了相应的文件。**
2. **我手动绘制了最基本的游戏素材，比如游戏背景图和必须的雷 旗子 空白格之类的素材**
3. **我将基本的素材都上传到了开发板，并且将游戏必须的底层功能完成，比如触摸获取坐标，显示bmp图片等**
4. **我将所有的底层函数封装起来，以此调用完成了主界面的创建，这样我就可以通过触摸进入游戏界面了。**
5. **正式开始游戏功能的搭建，我首先完成了生成雷的函数，然后初步通过输出函数确定了函数的功能正确性**
6. **雷生成完成后，我通过雷的位置来完成了数字的生成，原理是把雷的周围遍历一遍，将所有非雷的区域数字加一，这样叠加起来就完成了游戏的初始化。**
7. **因为还没有写游戏交互，所以我仍然通过输出函数确定了我写的功能是正确的**
8. **开始完成交互功能，当我点击游戏区域时，要根据雷 数字 空白的不同来显示不同的效果，主要难点是解决点击空白的问题。**
9. **因为点击空白的块时，需要显示它周围的所有非雷块，所以我用上了递归的方式解决，当空白块的周围有空白块时，我就递归调用函数**
10. **基本功能完成了，接下来就完成胜利和失败的函数编写。失败很容易，点击到雷就判定为失败。胜利需要判断剩余的格子数目，当剩余格子数目和雷的数目一致时，则判断游戏胜利，在这里我用了一个全局变量，每点击一个块它就自减**
11. **完成所有功能，这里可以开始添加其余功能了……**
12. **增加了游戏中显示时间的功能，增加了游戏胜利后将时间作为成绩显示在下方的功能**

问题和解决办法

**问题1. 一开始的时候，我在显示图片时就产生了问题，一些非全屏的图片只会显示一些零星的像素点，在经过排查后我确定了是自己的lcd显示bmp算法出了问题**

**解决办法：**

**经过排查，我发现是自己遗漏了bmp一行像素点必须是4的倍数的问题，所以会出现一些图片能正常显示，一些不能的问题。**

**问题2. 在写出产生雷的算法后，我迟迟想不出产生数字的算法，一开始我的解决方案是遍历保存格子的数组，一个一个格子来确定数字**

**解决办法：**

**我在写出这个算法后发现这个算法产生的时间复杂度太高了，所以我换了一种算法，通过扫描雷的周围来确定数字，这样通过空间换时间的方式解决了问题**

**问题3. 因为在电脑上是鼠标左右键来进行游戏，而在开发板上只有触摸屏，只能完成一种事件，所以不能很好的切换扫雷和插旗的选项**

**解决办法：**

**我通过在屏幕上设置一个按键，来完成了扫雷和插旗的切换，也就是点击按键，就可以从插旗（扫雷）模式转换为扫雷（插旗）模式。**

总结

**通过本次项目经验，我收获了许多。首先，我对于项目的规划和制作流程有了一定了解，并且加深了对于文件io c语言linux系统 多系统开发之类的知识理解。**