上位机模块设计报告

姓名： 学号：

目录

[一、UI设计 2](#_Toc104585302)

[二、通信协议编码 3](#_Toc104585303)

[三、通信实现 3](#_Toc104585304)

[四、维护模式 4](#_Toc104585305)

[五、线程管理 6](#_Toc104585306)

[六、导出发布 7](#_Toc104585307)

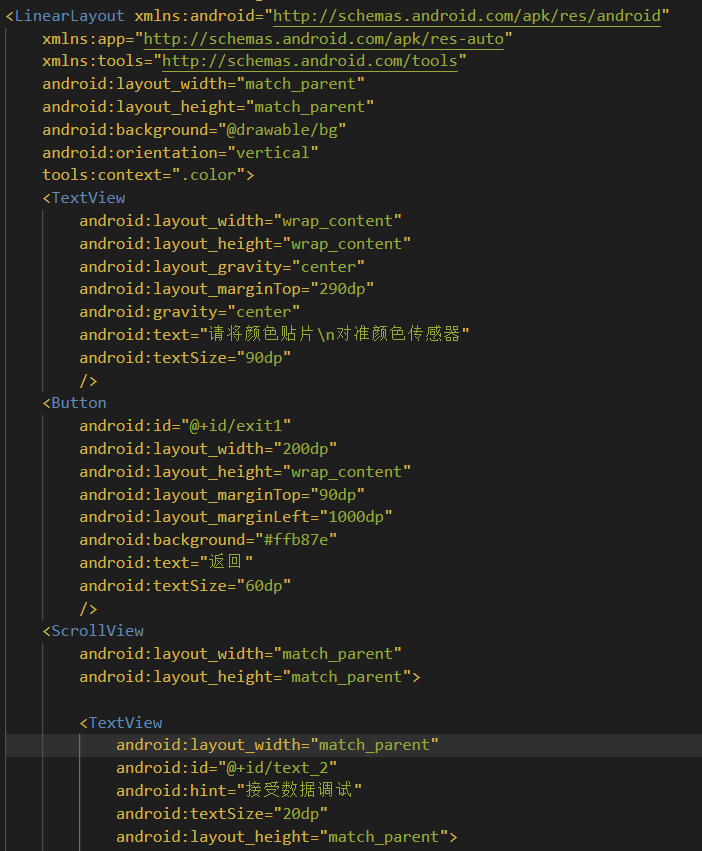
## 一、UI设计

饮料机的UI设计使用简洁明朗的设计风格，融入莫兰迪色，视觉效果舒适，同时突出强调功能的暗示和引导，使操作简单易上手，几乎没有使用门槛。



主界面UI

以简单的线性布局实现简洁的界面：



界面代码实现

## 二、通信协议编码

为了规范上位机与MCU之间的通信，需要对通信的内容进行编码。其中有12个指令通信、2种数据通信。指令通信可以使用一位字符编码，保留一位用于区分数据通信；数据通信则根据数据格式编码。最终使用灵活1位16进制进行编码如下：

**第一位**

|  |  |
| --- | --- |
| **编码** | **内容** |
| 0 | 上位机收：保留位，数据通信的开头位 |
| 1 | 上位机发：告知上位机需要可乐 |
| 2 | 上位机发：告知上位机需要雪碧 |
| 3 | 上位机发：告知上位机需要芬达 |
| 4 | 上位机收：表示未进入合法运行范围（10cm） |
| 5 | 上位机收：表示进入合法运行范围（10cm） |
| 6 | 上位机发：距离传感器测试，MCU接收后返回传感器数据 |
| 7 | 上位机发：颜色传感器测试，MCU接收后返回传感器数据 |
| 8 | 上位机发：舵机1测试 |
| 9 | 上位机发：舵机2测试 |
| a | 上位机发：舵机3测试 |
| b | 上位机发：初始化请求，MCU接收后判断是否进入10cm并告知上位机 |
| c | 上位机发：初始化成功，MCU接收后进入待命 |

**第二位**

|  |  |
| --- | --- |
| **编码** | **内容** |
| 1 | 距离传感器接受数据头 |
| 2 | 颜色传感器接受数据头 |

**距离传感器编码**：01xxx

例：01004 表示4mm，01233表示233mm，01020表示20mm

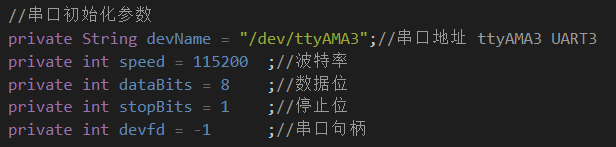
**颜色传感器编码**：02rrrgggbbb

例：02255255255 表示R255,G255,B255，02042152010表示R42,G152,B10

## 三、通信实现

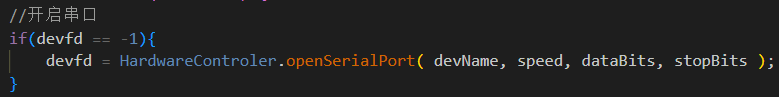
Friendly Elec公司生产的AOSP on NanoPi2提供了外置各种接口，借助提供的安卓例程中的封装的HardwareControl可以实现对接口的调用。本工程利用串口3与MCU进行通信。

1. 串口初始化参数



1. 串口开启

Devfd=-1表示串口开启失败

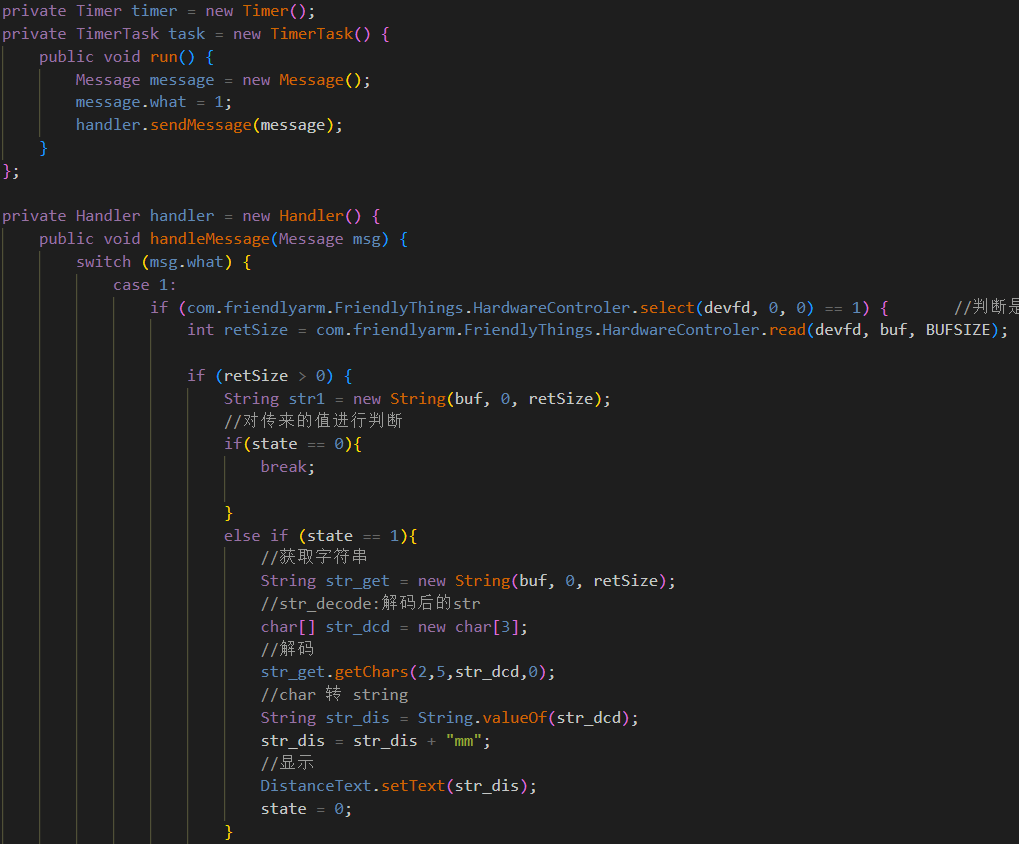


1. 串口发送



1. 串口接受

开辟一个计时器，对串口数据进行每500ms轮询一次，如果串口有数据，则进入串口数据处理。随后根据通信协议编码来解码数据。



## 四、维护模式

1．管理员登录

正确输入管理员账号、密码后方可登录进入维护模式。



代码实现



管理员登录UI界面

2．机器测试

在维护模式中，管理员可以对传感器和舵机进行测试，查看传感器是否正常连接、获取数据，检查舵机是否能正常运行。



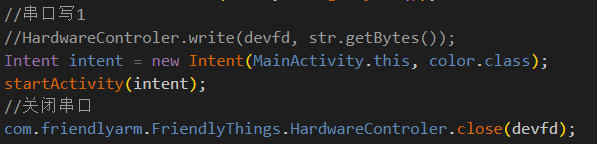
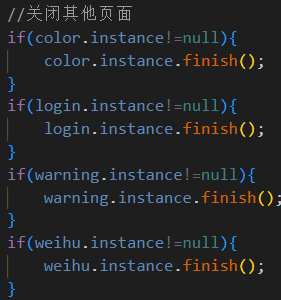
颜色传感器数据解码



维护模式UI

## 五、线程管理

在安卓开发中，开启的线程不会自动关闭，随着程序的运行，如果不及时关闭线程，会占用大量内存、处理器资源，拖慢APP运行，甚至导致崩溃。为了规避这种问题，在每个界面关闭上上一个界面的程序，并在每次串口发送结束之后关闭串口，灵活管理进程的生命周期，使通信收发更加灵敏、可用。



代码实现例子

## 六、导出发布

App在设计、调试完成后，就可以导出为.apk文件安装到上位机了。由于设计的app调用了系统的串口权限，因此必须经过签名打包才能正常运行。