技术文件

完成时间：2018.6.10

工程实践与科技创新[4A]
总体概要设计报告

组号：05

项目名称：AlphaBar 智能调酒机

成员：王浩宇（组长），周梦蝶，唐文成，黄子健，尚贤聪，曾艺涵，吴邦源，王春微，刘萌欣

摘　要：

本组创意项目旨在实现一个智能调酒系统 AlphaBar，通过一个放置于前面板的手机或平板操控整个调酒器。AlphaBar 支持菜单点酒、自定义配酒与语音交互判断心情自动推荐三种制作模式，以实现分层鸡尾酒的制作，配合灯光、音乐效果与心率传感，试图给用户以最好的品酒体验。通过点击进入管理员模式，系统管理人员可查看制作历史记录及原料余量信息，为未来的原料采购、酒品推荐提供数据信息，并在原料不足时自动提醒管理员加液。

关键词： 智能调酒 安卓 App 语音交互 蓝牙通信

ABSTRACT

This project mainly aims to implement an intelligent cocktail mixer system, AlphaBar, which is controlled by a smart phone on the front panel with a newly-designed Android Application. AlphaBar supports three order modes: order by menu, self-defined making, and automatic recommendation by mood judgment through speech input and interaction. While the cocktail is being made, the light and music provided attempt to create a supreme atmosphere for the drinker, together with heart rate abnormity detection offering a protection. By entering manager mode, the system manager can look at cocktail making history and amount of rest materials, which may serve as information for further material purchase and cocktail recommendation. Besides, when the amount of a certain material is below a threshold, the system informs the manager to add that material.

KEYWORDS

Intelligent Cocktail Mixer, Android Application, Speech Interaction, Bluetooth Communication

上海交通大学 电子信息与电气工程学院

地　址： 东川路 800 号

邮　编： 200240

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

上海交通大学 电子信息与电气工程学院
地 址：东川路 800 号
邮 编：200240

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

1. 概述

1.1 编写说明

本报告是上海交通大学电子信息与电气工程学院大三下工程科技创新[4A]课程的第五大组的总体概要设计报告。报告为我们的 AlphaBar 智能调酒机项目各子任务的开发人员提供整个系统的架构情况参考，为项目监督人员提供一个合理的开发进度参考，为项目任务的布置者提供一个比较完整的内容和时间的解决方案。

1.2 名词定义

1. AlphaBar：是一种能为用户提供三种鸡尾酒调制模式，并且允许管理员查看制作历史和原料查询的智能调酒机。三种调制模式为：菜单点酒、自定义点酒、语音点酒。在鸡尾酒配置过程中，附有音乐、灯光、心率监测效果。
2. 继电器：是一种电控制器件，是当输入量（激励量）的变化达到规定要求时，在电气输出电路中使被控量发生预定的阶跃变化的一种电器。在 Alpharbar 的设计中，由于 arduino 单片机的电驱动力不足，采用继电器对蠕动泵供电与控制。
3. 蠕动泵： 由驱动器、泵头和软管三部分组成的，通过对泵的弹性输送软管交替进行挤压和释放来泵送流体的装置。在本次设计中，为了得到需要的流量，利用马达转动来挤压软管，从而抽取液体原料。
4. arduino 单片机：arduino 是一款便捷灵活、方便上手的开源电子原型平台。包含硬件（各种型号的 arduino 板）和软件（arduino IDE)。它构建于开放原始码 simple I/O 介面版，并且具有使用类似 Java、C 语言的 Processing/Wiring 开发环境。主要包含两个主要的部分：硬件部分是可以用来做电路连接的 arduino 电路板；另外一个则是 arduino IDE，计算机中的程序开发环境。对 arduino 的编程是

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

利用 arduino 编程语言 (基于 Wiring)和 arduino 开发环境(基于 Processing)来实现的。

5. 压感：压力传感器（pressure transducer）指能感受压力信号并转换成可用输出信号的传感器。
6. 心率传感器：能通过光来检测血液容积变化进而检测心率。
7. 蓝牙发送接收：无线方式连接手机与 arduino 以实现手机 App 对系统的控制。
8. 亚克力板：acrylic，是一种开发较早的重要可塑性高分子材料，具有较好的透明性、稳定性，易加工、外观优美。
9. CAD 制图：computer aided design，计算机辅助设计，本项目采用 CAD 制图，对调酒机外观进行设计，并使用 CAD 文件加工、切割亚克力板。
10. SK6812 幻彩灯条：256 色的 SMD 5050 LED。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

2. 项目概述

2.1 项目名称

AlphaBar 智能调酒机

2.2 项目简介

本组创意项目旨在实现一个智能调酒系统 AlphaBar，通过一个放置于前面板的手机或平板操控整个调酒器。AlphaBar 支持菜单点酒、自定义配酒与语音交互判断心情自动推荐三种制作模式，以实现分层鸡尾酒的制作，配合灯光、音乐效果与心率传感，试图给用户以最好的品酒体验。通过点击进入管理员模式，系统管理人员可查看制作历史记录及原料余量信息，为未来的原料采购、酒品推荐提供数据信息，并在原料不足时自动提醒管理员加液。

2.3 项目组成员

表 2.1 小组任务分配情况表

小组	组长	成员
手机 App 模块	曾艺涵	黄子健、王春微
酒品调制模块	吴邦源	周梦蝶
传感器模块	刘萌欣	唐文成
外观与板材模块	王浩宇	尚贤聪、吴邦源

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

3. 系统详细设计

3.1 系统总体性能要求

1.用户友好

手机 App 需要能够清晰地向用户展示系统提供的功能，如有用户指南、酒品介绍查看页面等。通过语音输入，系统可与用户实现对话交互，并向用户根据心情的判断推荐相应的酒品。当出液口未放置杯子时，系统应提醒用户放置杯子。

2.响应时间要求

App 应能实时响应用户发起的请求。

3.可靠性

尽可能避免系统出现 App 闪退、电机停止运转、不能出液等问题。蓝牙的串口信令交互应尽量避免差错。

4. 可维护性

液体余量不足应向管理员发出提醒消息，并可通过查看制作历史记录统计商品销量，调整未来的酒品种类与推荐。

5. 可维护性

与用户实现对话交互中涉及的语音识别应能做到准确、快速，系统给出的反应与推荐也是如此。

3.2 系统设计

系统分为以下四个模块：

模块名称	模块工作
手机 App 模块	App 的制作与手机控制消息的编写、音乐播放
酒品制作模块	arduino 的编程控制继电器、蠕动泵工作以实现对液体的抽取
传感器模块	心率传感器测量心率、压力传感器测量杯子是否放置
外观与板材模块	灯光效果的实现、亚克力板的切割与拼装、原料配比选择

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

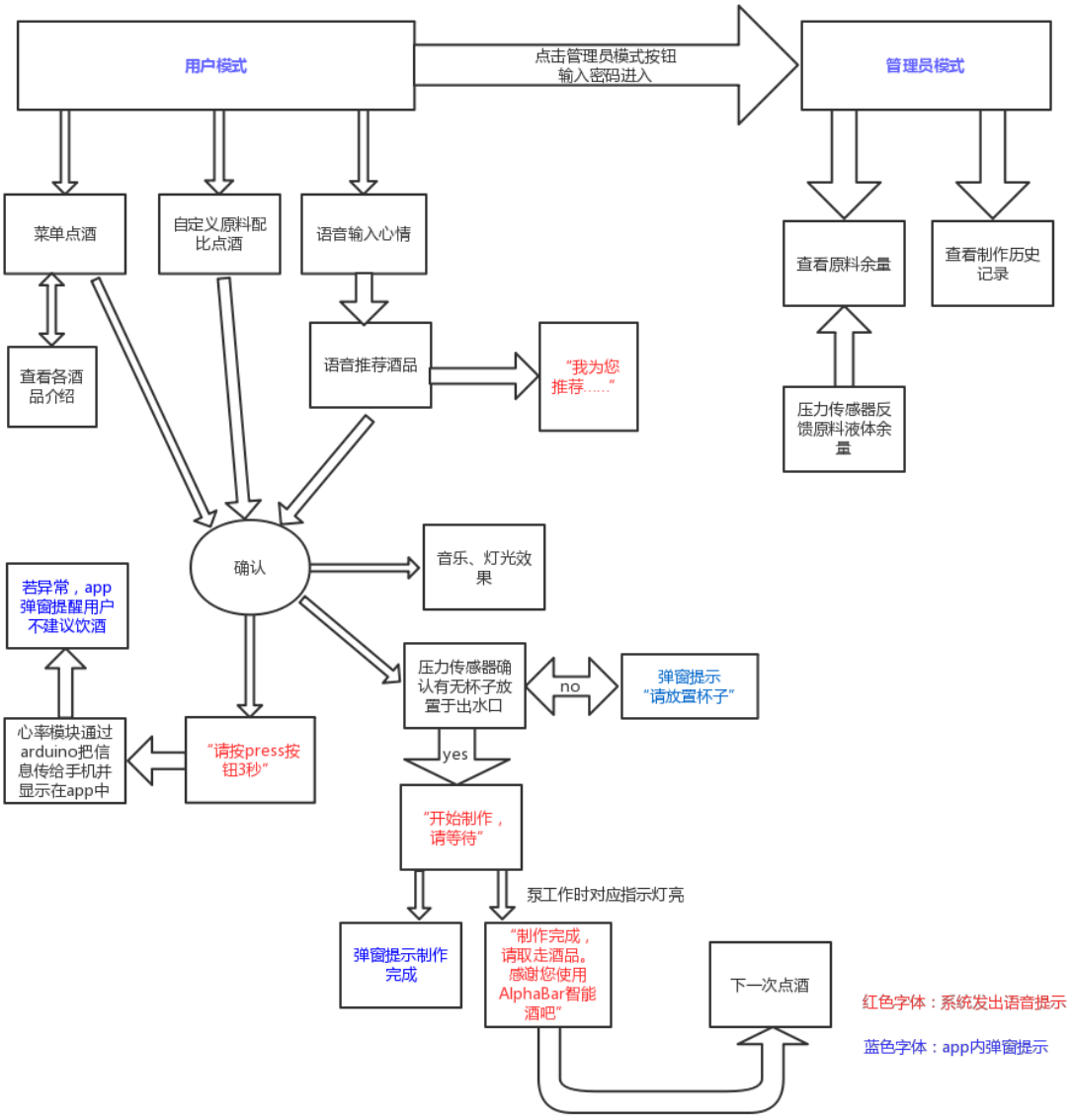


图 3.2.1 系统设计总体框图

上海交通大学 电子信息与电气工程学院

地 址：东川路 800 号

邮 编：200240

4. 手机 APP 模块

4.1 主要功能

本模块作为自助调酒机的人机交互界面，将以操作面板的形式提供用户服务和管理员服务。提供的服务内容如表 4.1 所示。

表 4.1 人机交互界面服务

用户服务	管理员服务
三种点酒方式（菜单、自定义、语音）	销量查看
心率检测	余量查看
压感检测	重置余量

4.2 功能模块

4.2.1 菜单点酒

菜单点酒是由 AlphaBar 系统提供配比确定的酒品推荐，已方便希望有明确口味倾向、想选择已有酒品的用户。在本系统中，我们设计了 6 种推荐酒品，酒品的配比和口感可以在菜单界面中查看，以方便用户能够更好的了解推荐酒品和配料信息，从而选择最适合自己的推荐酒品。

我们的推荐菜单包括六种酒品：玛格丽特、新加坡司令、长岛冰茶、金菲士、海岸、曼哈顿。其配料比例都是精心选择的，能够满足大众的口味。

如图 4.2.1 所示，菜单点酒的流程为，系统在侧栏菜单中点击进入菜单界面，查看各酒品具体信息，在主界面中通过选项框选择点酒酒品，点击确认即可点酒。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

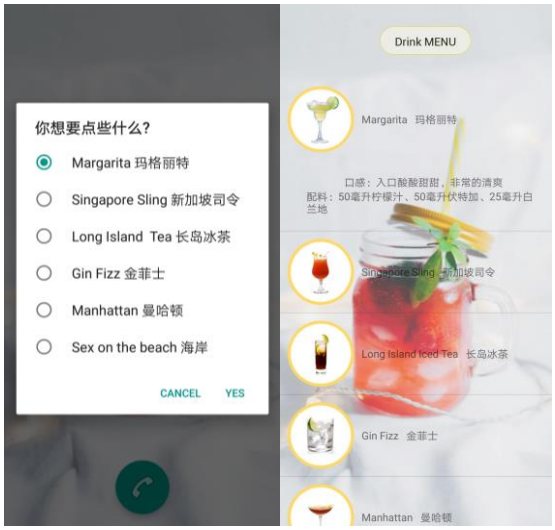


图 4.2.1 菜单点酒界面及菜单界面

用户点酒成功后，将进入心率检测模块，当通过心率测试，酒品制作完毕后，酒品的销售信息和余料信息将会在管理员仓库界面进行更新。

4.2.2 自定义点酒

对于用户而言，饮酒是一种享受的过程，调酒也不应是调酒师的个人乐趣。用户可能存在尝试调酒的意愿，按照自己的口味，调出自己的心仪酒品；也可能存在少部分用户不满意推荐酒品，想要按照自己的口味选择配料。

因此，AlphaBar 为用户提供了自定义点酒的功能。我们一共提供了 4 种配料供用户选择，分别是柠檬水、龙舌兰、白兰地、伏特加。

自定义点酒过程如图 4.2.2 所示，用户在主界面进入自定义调酒界面，通过拖动每样配料的进度条，选择自己想要的配料用量，调出心仪的口味。我们为每种原料提供 0-100Ml 的配比选择。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

图 4.2.2 自定义点酒界面

用户点酒成功后，将进入心率检测模块，当通过心率测试，酒品制作完毕后，酒品的销售信息和余料信息将会在管理员仓库界面进行更新。

4.2.3 语音交互

对于新用户而言，用户在选择酒品时，经常会出现不知道选择何种酒品；当用户心情或者心境不同时，不同的酒品会有不同的饮酒体验。为更好的满足用户的需求，AlphaBar 提供了“听你的心情，为你推荐”的语音交互点酒模块。

如图 4.2.3 所示，用户通过主页进入语音交互界面，向对话框说出此刻的心情，AlphaBar 会通过语音分析识别出用户此刻的心情，由系统机型进行语音回答，为用户推荐符合相应心情的酒品。我们将用户心情分为 6 大类，分别与 6 种推荐酒品进行对应，具体的对应信息如下表 4.2.1 所示。

表 4.2.1：推荐酒品与心情对应表

心情	**开心**	**难过**	**兴奋**	**困惑**	**纠结**	**生气**
酒品	曼哈顿	长岛冰茶	海岸	金菲士	新加坡司令	玛格丽特

若用户不满意推荐，则可以选择退出，我们将持续录音交互，继续搜索配对。若用

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

户满意推荐并点酒后，将会进入心率检测模块，当酒品制作后，酒品的销售信息和余料信息将会在管理员仓库界面进行更新。

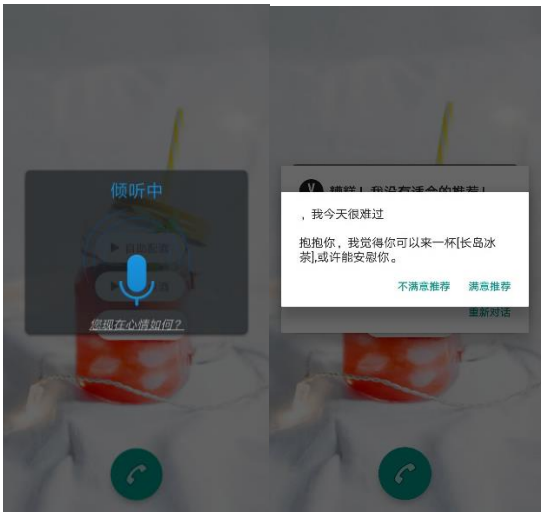


图 4.2.3 语音交互

此外我们在点酒流程的各环节都加入了语音播报，以提供更好的使用体验。

表 4.2.2 语音反馈场景

播报场景	播报内容
心率检测	“请将大拇指放在 press 按钮上“； ”请查看心率检测结果“
制酒成功	“您的酒品制作成功”
压感提醒	“未正确放置杯子，请重新放置”
余量提醒	“xxx 余量不足，请管理员注意添加”
语音交互	语音交互的智能识别回答

4.2.4 音乐播放

为提供调节饮酒氛围，当用户选择不同类型的酒品点酒后，将会播放不同类型的音乐为用户助兴，以更好的提高用户满意度。该模块一共为用户提供了 6 种音乐，音乐与酒品的对应信息如下表 4.2.3 所示。

上海交通大学 电子信息与电气工程学院

地 址：东川路 800 号

邮 编：200240

表 4.2.3：推荐酒品与音乐对应表

音乐	我相信	把悲伤留给自己	红日	光辉岁月	我的未来不是梦	海阔天空
酒品	曼哈顿	长岛冰茶	海岸	金菲士	新加坡司令	玛格丽特

4.2.5 管理员仓库模式

AlphaBar 提供了管理员仓库模式，可知直接通过 App 掌握目前的各类酒品的销售记录以及各种配料的余量。

如图 3.6 所示，本界面有管理员权限隐私设置，只有当管理员输入管理员用户与密码信息，才可进入管理员仓库模式，防止普通用户查看信息。

图 4.2.4 仓库信息登录界面及查看界面

管理员点击 Sale 按钮可进入销量统计界面，可以查看 6 种酒品的当前销量情况，通过对销量的分析，可以进一步掌握酒品的欢迎程度，并做出适当的酒品调整策略或者配料购买策略。

管理员点击 Rest 按钮可计入余量统计界面，可以查看 4 种配料的目前余量。

酒品制作前，若配料的余量低于警戒线时，系统将向管理员发送警告，提醒管理员补充配料。当管理员补充完配料后，进入管理员仓库模式点击 Add 按钮，选择相应配

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

料并点击 full 之后，可以更新当前配料余量。

4.2.6 心率检测

为避免用户饮酒过多，AlphaBar 提供了心率检测模块，以进行健康监督。

用户在每次点酒后，都会先进入心率检测模块为用户检测心率。用户通过将大拇指放在吧台面板的心率传感器上，如图 3.7 所示，点击开始检测按钮后，心率模块开始检测心率，一段时间后，App 界面会上显示用户的心率，并根据心率状态，提醒用户是否适合饮酒。

我们将不推荐饮酒的心率阈值设为 100。

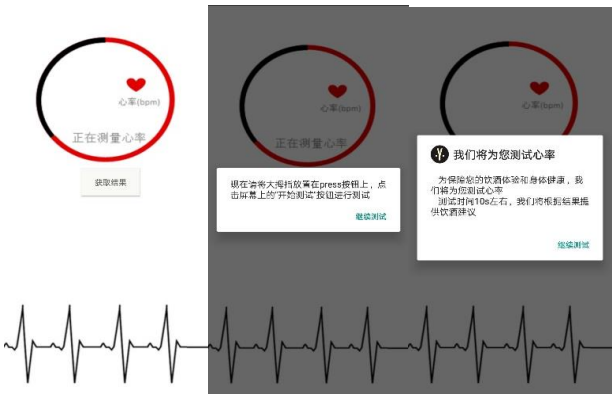


图 4.2.5 心率检测

若用户仍然想要继续，可以点击继续点酒按钮进入配酒状态，若最终配酒成功，则会以弹窗形式提醒用户酒品制作完成。

4.2.7 压感警报

为了识别用户在配酒时是否放置酒杯，以在用户忘记放置时提供警报警告和禁止注酒的防意外服务，App 中包含了压感警报环节。

当用户通过心率检测，进入酒品配置时，若没有正确放置杯子，系统将提供警报，并且机器将停止注酒。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

4.3 功能实现

4.3.1 语音交互

本次语音交互的实现基于讯飞开放平台。讯飞开放平台是科大讯飞推出的以语音交互技术为核心的人工智能开放平台。为开发者免费提供语音识别、语音合成等语音技术 SDK。

MSC-SDK 的功能从调用开始到结果返回，大多使用接口回调（Interface Callback）的方式返回结果和状态。其主要功能接口如图 3.8 所示。

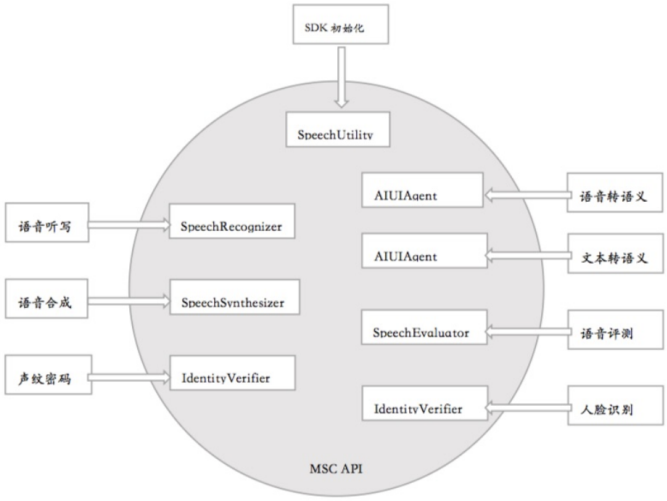


图 4.3.1 MSC-SDK 主要功能接口

在本次项目中，我们使用集成科大讯飞 MSC（Mobile Speech Client，移动语音终端）Android 版 SDK，运用了其中的语音识别和语音合成的功能来实现本次语音互动。

我们按照以下步骤将 SDK 用于在 Android 系统中：

1、导入 SDK

我们在讯飞平台上创建了 Appid，通过审核后进入应用接入流程，可在平台上获取应用功能对应的 SDK 服务文件支持，根据工程需求选取适当的平台库文件进行集成。我们将 android 设备 cpu 对应指令集的服务文件配置在系统库中。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

图 4.3.2 讯飞应用接入流程

2、添加用户权限

MSC-SDK 有两种引擎类型。分别为：

- 在线引擎（TYPE_CLOUD），又称为云端模式，需要使用网络，速度稍慢，并产生一定流量，但有更好的识别和合成的效果，如更高的识别匹配度，更多的发音人等。
- 离线引擎（TYPE_LOCAL），又称为本地模式，不需要使用网络，且识别和合成的速度更快，但同时要求购买并使用对应的离线资源。

出于成本的合理化考虑，我们选择云端模式进行语音交互的开发。

基于云端模式的使用权限范围，我们需要连接网络权限，执行云端语音能力，并开启读取网络信息和 wifi 状态的权限，允许程序改变网络连接状态；为了实现听写和识别，我们需要获取手机录音使用权限，并开启其他语音功能权限。

如图 4.3.3 所示，根据以上需求，我们对 Android 工程的权限进行相关配置。

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<!--连接网络权限，用于执行云端语音能力 -->
<uses-permission android:name="android.permission.INTERNET"></uses-permission>
<!--获取手机录音机使用权限，听写、识别、语义理解需要用到此权限 -->
<uses-permission android:name="android.permission.RECORD_AUDIO"></uses-permission>
<!--读取网络信息状态 -->
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"></uses-permission>
<!--获取当前wifi状态 -->
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"></uses-permission>
<!--允许程序改变网络连接状态 -->
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE"></uses-permission>
<!--读取手机信息权限 -->
<uses-permission android:name="android.permission.READ_PHONE_STATE"></uses-permission>
<!--读取联系人权限，上传联系人需要用到此权限 -->
<uses-permission android:name="android.permission.READ_CONTACTS"></uses-permission>
```

```
<!-- 语音所需必要的权限 -->
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_SETTINGS" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
```

图 4.3.3 语音交互的用户权限

3、初始化

初始化是用来创建语音配置对象，如图~所示，只有初始化后才可以使用 MSC 的各项服务。我们将初始化过程放在程序入口处，使用我们审核通过的 Appid 进行语音对

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

象的配置初始化。参数需要以键值对的形式存储在字符串中传入 createUtility 方法，以配置初始化对象的参数。

```
//speaking set
SpeechUtility.createUtility( context: this,  s: SpeechConstant.APPID +"=5ad1cbcb");
```

图 4.3.4 语音对象初始化

4.3.1.1 语音识别

语音识别技术(Auto Speech Recognize，简称 ASR)所要解决的问题是让机器能够“听懂”人类的语音，将语音中包含的文字信息“提取”出来，相当于给机器安装上“耳朵”，使其具备“能听”的功能。语音识别包括听写、语法识别功能。语音识别技术(Auto Speech Recognize，简称 ASR)即把人的自然语言音频数据转换成文本数据。

我们预计使用 API 实现识别并记录用户对话，以方便分析用户话语信息。因此我们使用语音听写记录用户对话内容。语音听写是基于自然语言处理，将自然语言音频转换为文本输出的技术。

此外，语音听写得到的是 JSON 文本流，为了对语音听写的文本进行匹配处理，以实现互动反馈，我们需要将 JSON 格式转换为 string。

因此，我们的语音听写处理过程步骤如下：

1、创建识别对象

在本工程中，我们使用标准的识别控件 RecognizerDialog 进行识别对象的初始化。而后，可以利用 SDK 封装的配置函数方便地配置识别参数。我们将识别对象配置为中文普通话，并以前端点超时 3s 作为语音识别的结束符。

```
RecognizerDialog mDialog = new RecognizerDialog(context,  initListener: null);
mDialog.setParameter(SpeechConstant.DOMAIN,  s1: "iat");
mDialog.setParameter(SpeechConstant.LANGUAGE,  s1: "zh_cn");
mDialog.setParameter(SpeechConstant.ACCENT,  s1: "mandarin");
mDialog.setParameter(SpeechConstant.VAD_BOS,  s1: "3000");
```

图 4.3.5 创建识别对象

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

2、 文本转换

在监听器进程中，识别对象得到的识别结果为 JSON 格式的文本流，这对我们的分析处理造成了不便。因此我们决定采用 JSON 库进行文本的解析转换。

JSON 是谷歌提供的开源库，可以用来解析 JSON 格式的数据。在 Android 项目中使用时，需要在库目录下添加 Gson 的库文件，以构建路径。

在本次工程中，我们利用 Gson 包将 API 返回的 JSON 格式转换为 String，对 JSON 文本进行逐个解析，并设计 VoiceClass 封装存储解析结果，最后将解析内容依次拼接，即得到最终的语音听写字符串。

```
public String parseVoice(String resultString) {
    Gson gson = new Gson();
    VoiceClass voiceBean = gson.fromJson(resultString, VoiceClass.class);

    StringBuffer sb = new StringBuffer();
    ArrayList<VoiceClass.WSBean> ws = voiceBean.ws;
    for (VoiceClass.WSBean wsBean : ws) {
        String word = wsBean.cw.get(0).w;
        sb.append(word);
    }
    return sb.toString();
}
```

```
public class VoiceClass {

    public ArrayList<WSBean> ws;

    public class WSBean {
        public ArrayList<CWBean> cw;
    }

    public class CWBean {
        public String w;
    }
}
```

图 4.3.6 JSON 文本格式转换

3、文本匹配处理

我们在监听进程中，利用以上的解析方法对识别文本做类型转换，因此可以方便地对字符串结果进行关键词搜索和匹配，以进行对应的互动反馈。

如图 4.3.7 所示，我们设计了关键词字典，尽可能覆盖大多数描述形容词，以向用户提供较全面的反馈效果。我们使用字符串关键词搜索方法进行匹配。根据匹配结果进行语音播报反应，并推荐适合的酒品供选择。

```
private String instructionhappy = "开心";
private String instructionexcited = "兴奋";
private String instructionsad = "难过";
private String instructionangry = "生气";
private String instructiontangle = "纠结";
private String instructionconfused = "困惑";
```

图 4.3.7 部分关键词字典

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

图 4.3.8 识别反馈

4.3.1.2 语音合成

语音合成，又称文语转换（Text to Speech，TTS）技术，涉及声学、语言学、数字信号处理、计算机科学等多个学科技术，是中文信息处理领域的一项前沿技术。与语音听写相反，语音合成是将文字信息转化为可听的声音信息，也即“让机器像人一样开口说话”。

讯飞的大规模语音语料库采用 HMM 统计建模框架，进行有针对性的模型优化，其预测的升学参数相当贴近真是的语音，因此能供提供非常良好的语音合成服务。

我们采用了在线语音合成服务，基于云端处理，将文本信息转换为声音信息，音量、语速、音高等参数支持动态调整，方便快速构建个性化声音。

与语音听写类似，语音合成也需要初始化合成对象。如图~所示，我们初始化语音合成对象，并对声音信息做相关配置。

在本次工程中，我们将合成的语音设定为国语普通话，发音人为女中音，以创造较缓和亲切的服务体验氛围。经过我们的测试体验，我们对声音进行了更详细的设定，语速设定为 60，语调设定为 50，音量设定为 100。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

```
public void make_voice(String content){
    SpeechSynthesizer speaker = SpeechSynthesizer.createSynthesizer( context: MainActivity.this,  initListener: null);
    speaker.setParameter(SpeechConstant.VOICE_NAME, s1: "xiaoyan");
    //初始化语音合成相关设置
    speaker.setParameter(SpeechConstant.SPEED, s1: "60");
    speaker.setParameter(SpeechConstant.PITCH, s1: "50");
    speaker.setParameter(SpeechConstant.VOLUME, s1: "100");
    speaker.setParameter(SpeechConstant.STREAM_TYPE, s1: "3");
    speaker.setParameter(SpeechConstant.KEY_REQUEST_FOCUS,  s1: "true");

    speaker.startSpeaking(content,  synthesizerListener: null);
}
```

图 4.3.9 创建并配置语音合成对象

4.3.2 蓝牙通信

在本次工程中，我们需要与作为控制主板的 arduino 板进行通信。由手机端发送客户请求的菜单配酒或自定义配酒命令，以控制 Ardiuno 板按照用户指定的方式控制四种原料的注入时间，以生成不同配比量；同时由 arduino 板发送配酒结束的通知指令，以及关于心率传感和压感传感器的传感状态信息，在手机端上接受信息并处理反馈给用户。

Android 用户界面和 Arduino 控制主板间的通讯信息如下表 4.3.1 所示。

表 4.3.1 Android 与 Arduino 蓝牙通讯信息

	发送信息	接收信息
Android 端	指定酒品指令； 自定义原料配比指令	实时心率数据； 压感传感器状态；
Ardiuno 端	实时心率数据； 压感传感器状态；	指定酒品指令； 自定义原料配比指令

4.3.2.1 Android 蓝牙开发

本次工程使用手机作为 Client，HC06 作为 Service，在 Android 与进行通信。

如表 4.3.2 所示，本次蓝牙开发中我们将用到以下建立蓝牙需要的基本类。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

表 4.3.2 蓝牙通信基本类

类名称	类功能
BluetoothAdapter	代表了一个本地的蓝牙适配器。用来发现、查询绑定设备，使用已知的 MAC 地址实例化一个蓝牙设备和建立一个 BluetoothServerSocket 作为服务器端来监听来自其他设备的连接。
BluetoothDevice	代表了一个远端的蓝牙设备。用来请求远端蓝牙设备连接或者获取 远端蓝牙设备的名称、地址、种类和绑定状态。
Bluetoothsocket	代表了一个蓝牙套接字的接口。是应用程序通过输入、输出流与其他蓝牙设备通信的连接点。

1、开启权限

Android 平台支持蓝牙网络协议栈，实现蓝牙设备之间数据的无线传输。

为了工程中使用蓝牙功能，我们需要先声明蓝牙权限蓝牙。此权限用来执行任何蓝牙通信，如请求一个连接、接受一个连接和传输数据。为了让应用启动设备发现或操纵蓝牙设置，必须申报 BLUETOOTH_ADMIN 许可。

```
<!--蓝牙通信权限-->
<uses-permission android:name="android.permission.BLUETOOTH"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
```

图 4.3.10 开启蓝牙权限

2、设置蓝牙

在 Android 蓝牙进行通信之前，需要验证是否设备是否能够开启蓝牙支持，并确保它已启用。如果不支持蓝牙，则应禁用任何蓝牙功能。如果蓝牙是支持的，但禁用，那么则要求用户启用蓝牙，且无需离开应用程序。

这个设置是在两个步骤来完成，如图 4.3.11 所示。首先，使用蓝牙适配器 BluetoothAdapter；下一步，需要确保蓝牙功能已启用。我们使用蓝牙连接基本类中

上海交通大学 电子信息与电气工程学院

地 址：东川路 800 号

邮 编：200240

的函数 isEnable 检查蓝牙目前是否启用，如果如果此方法返回错误，则禁用蓝牙。要求蓝牙启用将发出一个请求，使蓝牙通过系统设置，且不阻止应用程序进程。

```
BluetoothAdapter mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
if (mBluetoothAdapter == null){
    show_toast( msg: "Bluetooth not supported");
    return;
}

if(! mBluetoothAdapter.isEnabled()){
    //请求开启设备蓝牙
    Intent openBT = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
    //REQUEST_OPEN_BT_CODE = 1
    startActivityForResult(openBT,  requestCode: 1);
}
```

图 4.3.11 设置蓝牙

3、查询配对设备

使用蓝牙适配器 BluetoothAdapter 可以通过查询名单发现远程蓝牙设备配对。

设备发现是一个扫描程序，然而，局域网内的蓝牙设备将响应发现请求只有在能够被发现。如果一个设备可以被发现，它会响应发现请求并共享一些信息，比如设备名称、类别，并以其独特的 MAC 地址。使用此信息，执行发现的设备可以选择启动一个连接到所发现的设备。

getBondedDevices 获取与本机蓝牙所有绑定的远程蓝牙信息，以 BluetoothDevice 类实例返回。本次工程使用名称配对的方法，向设备指定了蓝牙配对对象 HC-06。

```
Set<BluetoothDevice> pairedDevices = mBluetoothAdapter.getBondedDevices();
for (BluetoothDevice i : pairedDevices)
{
    if (i.getName().equals("HC-06")) {
        mBluetoothDevice = i;

        show_toast( msg: "find bluetooth!");
        break;
    }
}
```

图 4.3.12 获取与本机蓝牙所有绑定的远程蓝牙信息

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

4、创建接口

使用已知的 MAC 地址实例化一个蓝牙设备和建立一个 BluetoothServerSocket 作为服务器端来监听来自其他设备的连接，如图~所示，作为应用程序通过输入、输出流与其他蓝牙设备通信的连接点。

```
try {
    mBluetoothSocket = mBluetoothDevice.createInsecureRfcommSocketToServiceRecord(MY_UUID);
} catch (IOException e) {
    e.printStackTrace();
}
```

```
try {
    mOutputStream = mBluetoothSocket.getOutputStream();
    mInputStream = mBluetoothSocket.getInputStream();
    show_toast( msg: "ready1!");
} catch (IOException e) {
    e.printStackTrace();
}

mConnectedThread = new ConnectedThread(mBluetoothSocket);
mConnectedThread.start();
```

图 4.3.13 创建蓝牙通信连接点

4.3.2.2 Android 蓝牙发送

通过服务器端建立的数据输出流接口，我们可以方便地传输任意数据。传输过程如下：

首先，获取 InputStream 和 OutputStream，二者分别通过套接字以及 getInputStream() 和 getOutputStream() 来处理数据传输。

而后，使用 read(byte[]) 和 write(byte[]) 读取数据并写入到流式传输。

```
byte[] buffer = msg.getBytes();
mOutputStream.write(buffer);
```

图 4.3.14 Android 蓝牙发送数据

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

4.3.2.3 Android 蓝牙接收

蓝牙的接收过程是被动的，因此我们设计另一线程进行无间断的循环监听，如图 4.3.15 所示。

为了避免不同信号的接收串扰，我们设计了标识符作为启动监听的标识，在 Handler 中进行主线程和子线程的信息传递。由于接收信号有心率信号和压感信号两类，因此两类信号将分别在两个 Handler 中记录和传递接收信息，并以不同的标识符作为启动标志。

```
// 监听输入流以备获取数据
while (true) {
    try {
        bytes = mInputStream.read(buffer);
        // 将接受数据发回UI处理
        //接收心率信号
        if(bytes!=-1&&allowRec){
            bluetoothIn.obtainMessage(RECIEVE_MESSAGE, bytes,  arg2: -1, buffer).sendToTarget();

        }
        //接收压感信号
        if(bytes!=-1&&allowRec2) {
            pressIn.obtainMessage(RECIEVE_MESSAGE, bytes,  arg2: -1, buffer).sendToTarget();
        }
```

图 4.3.15 监听启动信号接收

为了防止线程频繁工作造成的数据处理不及时而导致的阻塞，我们设计了线程睡眠时间。如图 4.3.16 所示

```
try {
    //线程睡眠20ms以避免过于频繁工作  50ms->20ms
    //导致UI处理发回的数据不及时而阻塞
    Thread.sleep( time: 20);
} catch (InterruptedException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
```

图 4.3.16 线程休眠防阻塞

在 Handler 进程中，我们对接收到的输入流进行转换处理，并对转换后的字符串信息做相关判断反馈。对于心率数据，我们将判断心率数值，作出正常范围和过高

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

范围的判断，对应不同的反馈；对于压感数据，我们将判断压感状态，作出有放置杯子和无杯子防止的判断。

```
public void get_value(String heart_data){
    //判断心率信号
    geth=heart_data;
    show_toast(geth);
    allowRec=false;

    if (geth.length()<5  ){
        go_head_guide(geth);
    }
    else{
        stop_guide(geth);
    }
}
```

```
public void get_press(String press){
    //判断压感状态
    if(press.compareTo( string: "b")<= 0){
        show_toast( msg: "press");
        make_drink();
    }
    else{
        stop_press();
    }
}
```

图 4.3.17 心率和压感的判断反馈

4.3.3 音乐播放

为了配合不同点酒动作，与灯光共同营造氛围效果，我们设计音乐播放服务作为配酒时的背景音乐。

如图 4.4.1 所示，我们采用 Service 进行后台运行，即通过不可见的无界面程序，在不影响主程序运行的状态下提供后台的多媒体服务。我们接受用户自行关闭音乐，因此设计了判断音乐暂停和重启的监听器。由主程序传递给 Service 的信息存在 intent 中，以判断应该调用哪个对应的音乐资源进行播放。

```
public class PlayingMusicServices extends Service {
    //用于播放音乐等媒体资源
    private MediaPlayer mediaPlayer;
    //标志判断播放歌曲是否是停止之后重新播放，还是继续播放
    private boolean isStop=true;

    @Nullable
    @Override
    public IBinder onBind(Intent intent) { return null; }
    //在此方法中服务被创建
    @Override
    public void onCreate() {
        super.onCreate();
        if (mediaPlayer==null){
            mediaPlayer=new MediaPlayer();

            //为播放器添加播放完成时的监听器
            mediaPlayer.setOnCompletionListener((mp) -> {
                    //发送广播到MainActivity
                    Intent intent=new Intent();
                    intent.setAction("com.complete");
                    sendBroadcast(intent);
            });
        }
    }
```

图 4.4.1 创建音乐播放多媒体资源

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

如图 4.4.2 所示，我们在主程序中设计 PLAY_MUSIC1~PLAY_MUSIC6 作为调用六首不同音乐资源的指令符，通过提取和判断匹配主程序中使用的指令符，即可在 Service 中开启相对应的音乐播放，以实现不同点酒动作对应不同的音乐播放。

```
public int onStartCommand(Intent intent, int flags, int startId) {
    switch (intent.getIntExtra( name: "type", defaultValue: -1)){
        case MainActivity.PLAT_MUSIC1:
            if (isStop){
                //重置mediaplayer
                mediaPlayer.reset();
                //将需要播放的资源与之绑定
                mediaPlayer=MediaPlayer.create( context: this,R.raw.happy);
                //开始播放
                mediaPlayer.start();
                //是否循环播放
                mediaPlayer.setLooping(false);
                isStop=false;
            }else if (!isStop&&mediaPlayer.isPlaying()&&mediaPlayer!=null){
                mediaPlayer.start();
            }
            break;
```

图 4.4.2 判断启动音乐资源

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

5. 酒品调制模块

5.1 主要功能

此模块通过蓝牙桥接手机客户端和 Arduino，使用手机端发送信号，Arduino 主板处理信号，控制蠕动泵的开关，通过标定实现定量控制，从而实现自定义或菜单饮品的注入和中断。

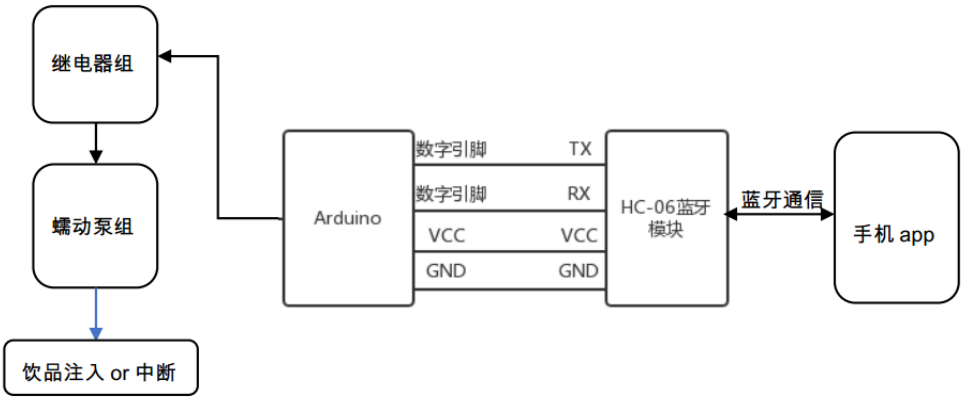
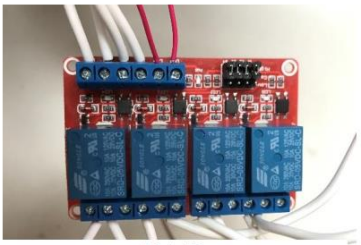


图 5.1.1 Ardiuno 控制流程图

蓝牙 HC06



继电器

蠕动泵

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

图 5.1.2 控制模块元件

5.2 功能模块

1.蓝牙模块

蓝牙核心模块使用 HC-06 从模块，引出接口包括 VCC,GND,TXD,RXD,预留 LED 状态输出脚，单片机可通过该脚状态判断蓝牙是否已经连接。led 指示蓝牙连接状态，闪烁表示没有蓝牙连接，常亮表示蓝牙已连接并打开了端口。输入电压 3.6~6V，未配对时电流约 30mA，配对后约 10mA，输入电压禁止超过 7V。

2.单片机

本实验采用 arduino 单片机编写代码接受蓝牙信号，并对继电器做对应处理，从而实现实验目的。arduino 是一款便捷灵活、方便上手的开源电子原型平台。它构建于开放原始码 simple I/O 介面版，并且具有使用类似 Java、C 语言的 Processing/Wiring 开发环境。主要包含两个主要的部分：硬件部分是可以用来做电路连接的 arduino 电路板；另外一个则是 arduino IDE，你的计算机中的程序开发环境。你只要在 IDE 中编写程序代码，将程序上传到 arduino 电路板后，程序便会告诉 arduino 电路板要做些什么了。arduino 能通过各种各样的传感器来感知环境，通过控制灯光、马达和其他的装置来反馈、影响环境。板子上的微控制器可以通过 arduino 的编程语言来编写程序，编译成二进制文件，烧录进微控制器。对 arduino 的编程是通过 arduino 编程语言 (基于 Wiring)和 arduino 开发环境(基于 Processing)来实现的。基于 arduino 的项目，可以只包含 arduino，也可以包含 arduino 和其他一些在 PC 上运行的软件，他们之间进行通信 (比如 Flash, Processing, MaxMSP)来实现。

3.蠕动泵

本实验采用蠕动泵型号为 karmoer 12V 直流蠕动泵。电机长度 48.5mm,整机长度为 65mm。经实验标定流速为 100ml/min。蠕动泵就像用手指夹挤一根充满流体的软管，随着手指向前滑动管内流体向前移动。蠕动泵也是这个原理只是由滚轮取代了手指。通

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

过对泵的弹性输送软管交替进行挤压和释放来泵送流体。就像用两根手指夹挤软管一样,随着手指的移动,管内形成负压,液体随之流动。蠕动泵就是在两个转辊子之间的一段泵管形成“枕”形流体。“枕”的体积取决于泵管的内径和转子的几何特征。流量取决于泵头的转速与“枕”的尺寸、转子每转一圈产生的“枕”的个数这三项参数之乘积。“枕”的尺寸一般为常量（泵送粘性特别大的流体时除外）。

4.继电器

继电器连接 arduino 数字引脚，为蠕动泵提供 12V 直流电压。

5.3 设计细节

5.3.1 各模块与 arduino 引脚连接方法

蓝牙 VCC：接 arduino 的 5V。

蓝牙 GND：接 arduino 的 GND。

蓝牙 TXD：发送端，一般表示为自己的发送端，接 arduino 的 RX。

蓝牙 RXD：接收端，一般表示为自己的接收端，接 arduino 的 TX

蓝牙 KEY：接 arduino 的 GND

继电器组：输入端分别连接 arduino 数字 4,5,6,7，输出端分别连接对应蠕动泵 1,2,3,4,。

5.3.2 器材选择

不同于工科创以往项目，本实验的实验器材完全是由本组成员自由选择，我们深知器材是否合适对实验结果会有非常大的影响。 为达到最佳实验效果，我们在开始实验前搜集了实验所需器材的相关资料，并详细分析其性能后选取了以上器材。

5.3.3 汲液装置

最初我们考虑将器皿倒置，并使用继电器直接控制出口的开关来实现液体的输出。经讨论后我们发现，随着液体的减少，器皿内液面上气压势必会减少，导致相同开关

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

时间输出液体体积不同，甚至会导致液体无法下降，若采取此方法，在计算液体余量和输出量时会非常复杂。而后我们考虑加入类似输液瓶的排气管，使液面气压始终一致，该方法解决了上述问题，保证相同开关时间输出液体体积始终一致。但是，倒置液体不利于更换器皿和添加液体，因此我们更换思路，将液体正置。而蠕动泵作为一个汲取液体装置，本身不依靠外界压强汲取液体，而是通过挤压管内空气和释放来泵送液体，很好地符合正放液体的汲液要求。

5.3.4 蠕动泵标定

不同液体密度不同，粘稠度不同，导致蠕动泵对不同液体抽取量不同。为达到精准释放各类酒品的需求，我们标定了蠕动泵对不同液体的抽取量：粉色液体和绿色液体 100ml/min 蓝色液体和红色（添加糖浆）90ml/min。

5.3.5 实验遇到的问题和解决方法

蓝牙和手机 App 端串口通信：

一开始对 App 发送的字符串进行处理时，我们认为串口通信蓝牙端接受到的是分离的字符，因此可以一次接受中对字符串进行分割处理。但实验发现发送“AABD””ABD”的结果均是 A 饮品输出一次即结束运行，与预期 ABD 饮品均有输出不一致。经分析发现，蓝牙串口通信接收到的是完整字符串，并不会有时间上的分离。因此，我们对接受到的字符串另做单独处理，得到字符串数组，再对该数组做对应饮品释放的处理。

5.4 功能实现

5.4.1 代码总体结构和功能

通过判断 app 向蓝牙模块发送的信息来控制泵的运转，从而控制调酒的量的多少。结构上，在压感模块和心率模块正常工作后，通过分支结构判断收到的字符，区分不同的调酒模式。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

5.4.2 代码设计思路

为调制不同酒品不同比例的饮品，手机 App 向蓝牙模块发送一串字符串，arduino 处理字符串，并控制继电器开关时间来操作蠕动泵实现不同酒品不同比例的饮品制作：

1. 若为单个字符‘X’，‘Y’，‘Z’，‘W’，‘U’，‘V’，对应菜单上 6 种既定饮品，程序控制继电器组开关时间，实现既定饮品配比。

2. 若收到字符串组如“ABCD”：其中 A、B、C、D 分别对应四种原料的配比量，范围为 0-100 mL,跳变区间为 10 mL。如”ABCD”意为四种原料分别为 0mL、10mL、20 mL、30 mL。

5.4.3 流程逻辑

模块的逻辑流程如图 5.4.1 所示。首先程序判断压感与心率模块是否正常工作。若没有正常工作，则不会启动调酒流程，如用户心率过快、酒杯未正常放置等情况；若正常工作，继而判断是否接收到蓝牙信息。根据蓝牙信息指定的调酒要求，驱动泵开始调酒。

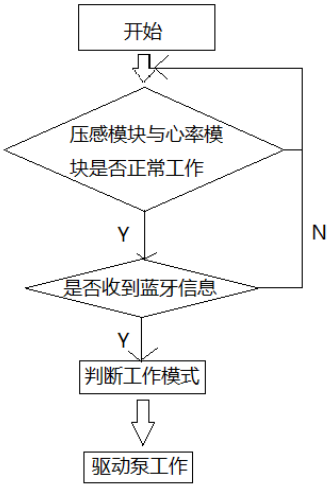


图 5.4.1 控制模块流程图

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

5.4.4 调用函数说明

```
void pour1(int pin,int tm)
{
  digitalWrite(pin,HIGH);
  for (j=0;j<tm;j++)
    delay(1000);

  digitalWrite(pin,LOW);
  delay(200);
}
```

硬件模块不同于软件设计，digitalWrite（）控制管脚输出为瞬时操作，因此需要delay（）函数来维持设定时间的管脚输出状态。但由于 arduino 中自带的 delay()函数不支持超过 15300ms 的延迟设计，因此采用循环结构处理泵来设计泵转动的时间，从而达到准确控制的效果。

`void serialOutputWhenBeatHappens()` ：这是心率模块函数。

`void Detectcup()`：这是压感检测函数。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

6. 传感器

6.1 心率模块

6.1.1 设计目标

本模块主要是利用心率传感器，通过探测指尖，得到用户的心率，从而对用户提供反馈和建议。如果心率过高，则不建议用户继续饮酒。

6.1.2 模块元件

PulseSensor 是一款用于脉搏心率测量的光电反射式模拟传感器。将其佩戴于 手指或耳垂等处，通过导线连接可将采集到的模拟信号传输给 arduino 等单片机用来转换为数字信号，再通过 arduino 单片机简单计算后就可以得到心率数值，此外还可将脉搏波形上传到电脑上显示波形。

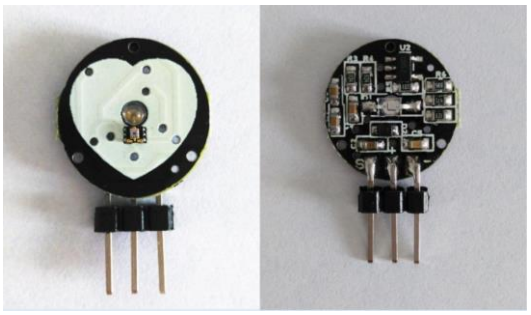


图 6.1.1 正面（手指接触面） 反面（非手指接触面）

PulseSensor 与 arduino 的连接方式如下图所示：

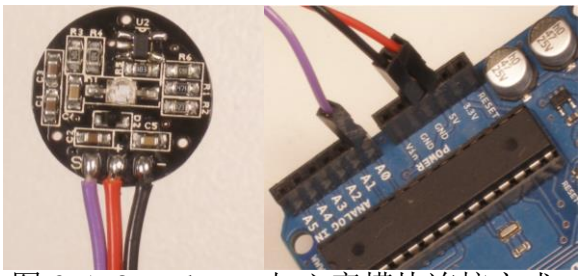


图 6.1.2 arduino 与心率模块连接方式

上海交通大学 电子信息与电气工程学院

地 址：东川路 800 号

邮 编：200240

6.1.3 模块原理

传统的脉搏测量的方法主要有三种：一是从心电信号中提取；二是从测量血压时压力传感器测到的波动来计算脉率，三是是光电容积法。前两种方法提取信号都会限制用户的活动，如果长时间使用会增加用户生理和心理上的不舒适感。而光电容积法脉搏测量作为监护测最普遍的方法之一，其具有方法简单、佩戴方便、可靠性高等特点。

光电容积法的基本原理是利用人体组织在血管搏动时造成透光率不同来进行脉搏测量的。其使用的传感器由光源和光电变换器两部分组成，通过绑带或夹子固定在病人的手指或耳垂上。光源一般采用对动脉血中氧和血红蛋白有选择性的一定波长（500nm、700nm）的发光极管。当光束透过人体外周血管，由于动脉搏动充血容积变化导致这束光的透光率发生改变，此时由光电变换器接收经人体组织反射的光线，转变为电信号并将其放大和输出。由于脉搏是随心脏的搏动而周期性变化的信号，动脉血管容积也周期性变化，因此光电变换器的电信号变化周期就是脉搏率。

根据相关文献和实验结果，560m 波长左右的波可以反映皮肤浅部微动脉信息，适合用来提取脉搏信号。本传感器采用了峰值波长为 515nm 的绿光 LED，型号为 AM2520，而光接收器采用了 APDS9008，这是一款环境光感受器，感受峰值波长为 565nm，两者的峰值波长相近，灵敏度较高。此外，由于脉搏信号的频带一般在 0.05 到 200Hz 之间，信号幅度均很小，一般在毫伏级水平，容易受到各种信号干扰。在传感器后面使用了低通滤波器和由运放 MCP6001 构成的放大器，将信号放大了 330 倍，同时采用分压电阻设置直流偏置电压为电源电压的 1/2，使放大后的信号可以很好地被单片机的 AD 采集到。

整个心率传感器的结构如下图：

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

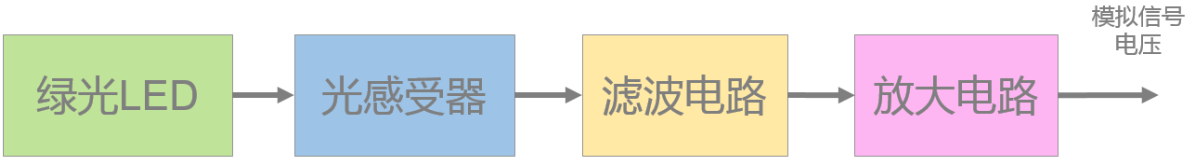


图 6.1.3 心率传感器的结构

检测到的心率数据可由 processing 软件显示，结果如下图所示。

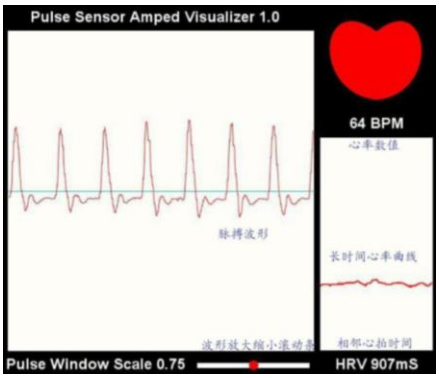


图 6.1.4 心率显示

6.1.4 应用说明

基于 PulseSensor 传感器搭建心率脉搏测量系统主要有两种方法：一、有线传输方式，将信息通过 USB 先传输到电脑端显示；二、无线传输方式，通过蓝牙传输。在本实验中，采用第二种无线传输方式，通过蓝牙将 arduino 中计算得到的心率传送到手机。

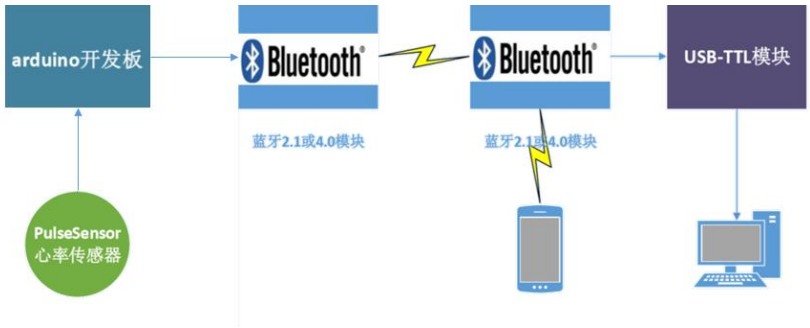


图 6.1.5 系统蓝牙连接

上海交通大学 电子信息与电气工程学院

地 址：东川路 800 号

邮 编：200240

6.1.5 功能实现

代码实现分为两大部分：心率监测和蓝牙通信。

1、心率检测

检测到心率变化，并储存两次心跳之间的时间间隔。

```
 if (N > 250){                                   // avoid high frequency noise
   if ( (Signal > thresh) && (Pulse == false) && (N > (IBI/5)*3) ){
     Pulse = true;                               // set the Pulse flag when we think there is a pulse
//     digitalWrite(blinkPin,HIGH);                // turn on pin 13 LED
     IBI = sampleCounter - lastBeatTime;         // measure time between beats in mS
     lastBeatTime = sampleCounter;               // keep track of time for next pulse

     if(secondBeat){                        // if this is the second beat, if secondBeat == TRUE
       secondBeat = false;                  // clear secondBeat flag
       for(int i=0; i<=9; i++){             // seed the running total to get a realisitic BPM at startup
         rate[i] = IBI;
       }
     }

     if(firstBeat){                         // if it's the first time we found a beat, if firstBeat == TRUE
       firstBeat = false;                   // clear firstBeat flag
       secondBeat = true;                   // set the second beat flag
       sei();                               // enable interrupts again
       return;                              // IBI value is unreliable so discard it
     }
```

每取十次心率时间间隔，就对检测到的心率间隔取平均，并通过心率间隔计算出每分钟的心跳次数，也就是心率。

```
  for(int i=0; i<=8; i++){                // shift data in the rate array
    rate[i] = rate[i+1];                  // and drop the oldest IBI value
    runningTotal += rate[i];              // add up the 9 oldest IBI values
  }

  rate[9] = IBI;                          // add the latest IBI to the rate array
  runningTotal += rate[9];                // add the latest IBI to runningTotal
  runningTotal /= 10;                     // average the last 10 IBI values
  BPM = 60000/runningTotal;               // how many beats can fit into a minute? that's BPM!
```

2、蓝牙通信

因为心率传输速率较高，数据量大因此我们对得到的心率取平均，并做一定的延时，从而减小信息传输的压力。利用蓝牙将心率数据传输过去。

```
void serialOutputWhenBeatHappens(){
 if (serialVisual == true){            //  Code to Make the Serial Monitor Visualizer Work
      num=num+1;
    BPMs=BPMs+BPM;
    if(num == 5){
      arvBPM=BPMs/5;
    }
    if(num==10){
    Serial.print("BPM: ");
    Serial.println(arvBPM);
    BT.println(arvBPM);
    BPMs=0;
    num=0;
    }
```


上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

6.2 压力检测模块

6.2.1 设计目标

将压力传感器置于放置杯子处，若未放置杯子时，则 App 弹出弹框并停止制酒进程，放置酒空流，若检测到有杯子，则开始制酒。

6.2.2 模块元件

FSR402 压力传感器是将施加在 FSR 传感器薄膜区域的压力转换成电阻值的变化，从而获得压力信息。压力越大，电阻越低。其允许用在压力 0g-10kg 的场合。

该传感器的机械参数如下：

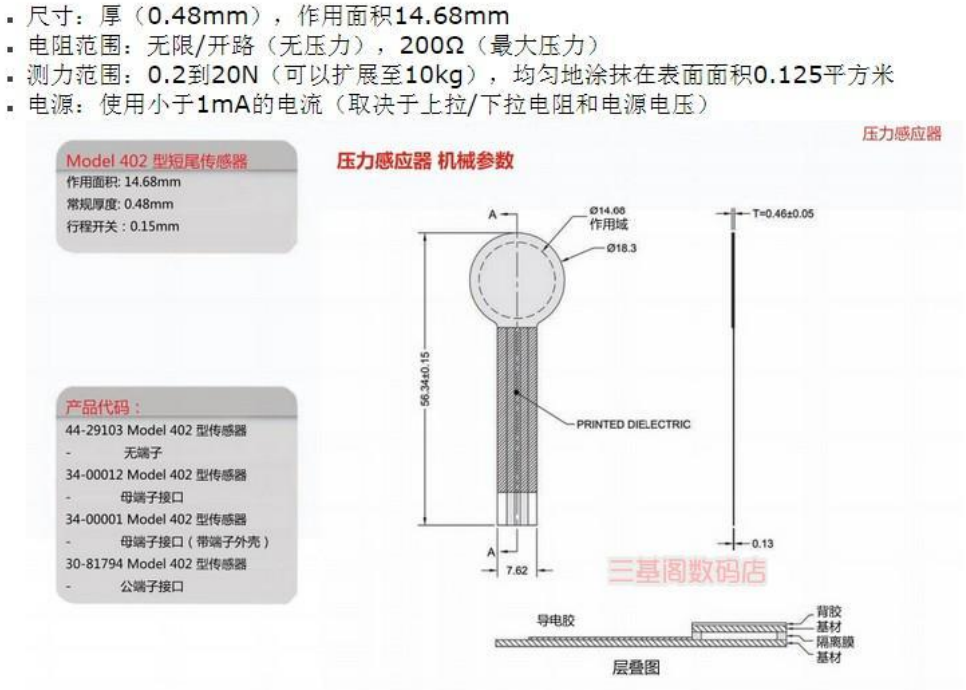


图 6.2.1 传感器机械参数

在实际使用中 FSR 传感器实物如图 6.2.2 所示：

上海交通大学 电子信息与电气工程学院

地 址：东川路 800 号

邮 编：200240

图 6.2.2 压力传感器

6.2.3 模块原理

FSR 随着压力使电阻不断变化。当压力为 0 时，传感器相当于一个开路，随压力增大电阻不断减小。需要注意的是，电阻与压力关系图不是标准的线性关系图，尤其在低测力的时候迅速从无限到 100kΩ。压力传感器压力与电阻的关系图：

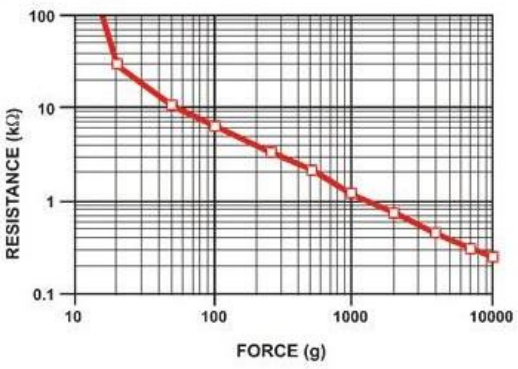


图 6.2.3 压力传感器压力与电阻的关系图

6.2.4 电路图设计

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

在该项目中，为了将压力传感器集成入 arduino 中，设计如下电路：

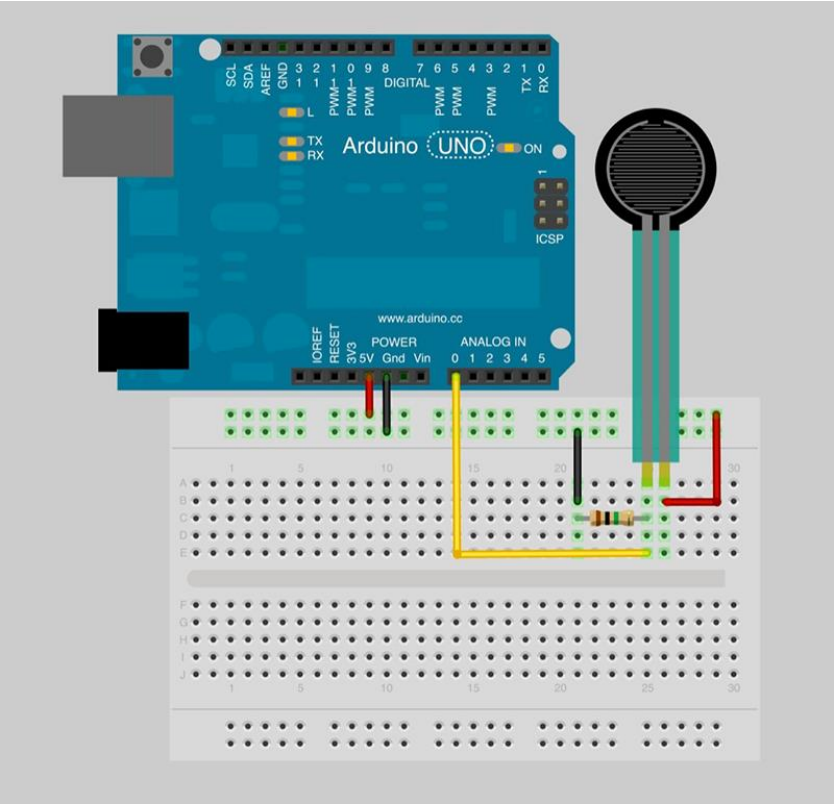


图 6.2.4 压感电路连接

其中，电阻值经过试验后，确定为 10kΩ，判断标准为可以将有无杯子的重力与噪声十分清晰的分辨出。

6.2.5 功能实现

通过上述原理分析，可以得知，可以读取此时电阻上的电压值（由于传感器阻值变化故此电压值会变化），找到有无杯子时的电压值分界点，从而判断是否有杯子。

故代码实现如下：

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

```
void Detectcup(){

  value = analogRead(sensorPin);       //Read and save analog value from potentiometer
  Serial.print("Analog reading = ");
  Serial.println(value);
  if (value < 10) {
    vall = 1;
    BT.println("cccc");
  } else {
    vall =0;
    BT.println("aaaa");
    Serial.println("cup ok ");
  }
  delay(500);                          //Big delay

}
```

此函数出现于蓝牙组接收到 App 传送来的“开始制作”消息之后，其中 vall 值用于判断之后的制作流程是否继续进行。Delay 用于在蓝牙传送时将该信息与后续心率监测的信息区分开。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

7. 外观与板材模块

7.1 灯光效果

7.1.1 硬件参数

型号：SK6812 幻彩灯条

光源：SMD 5050 LED;

板材：FPCB;

芯片：台湾晶元芯片;

IC 型号：WS2811

灰度等级：256 级

LED 数量：120 个

7.1.2 模块功能

在三种不同点酒模式下用不同颜色的呼吸灯闪烁，达到高辨识度的效果，更加吸引顾客使用该设施。

7.1.3 接口定义

由于设计需要 LED 显示为呼吸灯效果，又由于 arduino 为单线程的单片机，因此在控制时考虑使用单独单片机控制。而该型号灯带不同于传统意义上的多 LED 控制，仅仅使用单独的一个信号输入即可控制任意少量个数的 LED 发出任意颜色任意模式的灯光，又该灯带需要的供电功率较小，所以方便起见直接使用单片机的电源供电，单片机的 6 号脚作为灯带的控制脚，单片机的 23 号角作为模式选择的输入角，从主控单片机中获得模式信号。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

图 7.1.1 SK6812 幻彩灯带

7.1.4 软件总体结构和功能

采用循环结构，每次循环确认输入项的值来控制选择模式，不同模式控制不同的灯光闪烁模式。

1. 重要的全局变量

表 7.1 最重要的全局变量

Mode	**控制模式变化**
Pin1	输入的控制命令
Pin2	输入的控制命令

2.流程逻辑

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

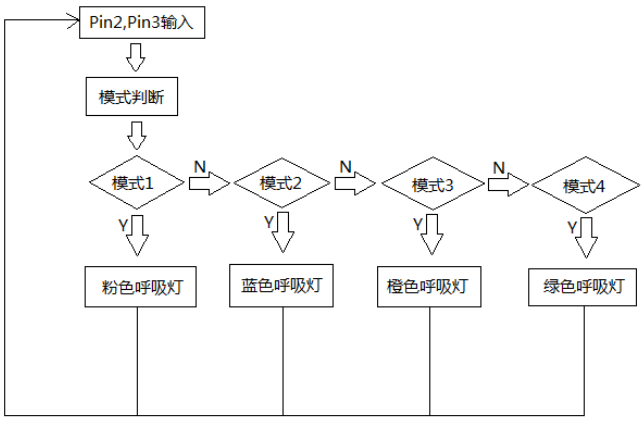


图 7.1.2 流程逻辑

3. 输入输出项描述

输入项：来自主控单片机的控制信号。

输出项：控制灯带灯光模式的控制信号。

4. 调用函数说明

本程序唯一的调用函数：rainbowfade2white();其作用是通过不同的灯光亮度来达到呼吸灯的效果，比如说所需要的颜色的 RGB 颜色为(160,130,50)，通过在前面定义的呼吸灯的数组 neopix_gamma[] 的设定数值给颜色加上权重 floor(RGB*neopix_gamma[i]/255)，从而得到需要的灯光亮度值，从而控制不同颜色下的呼吸灯。

7.2 板材制作

7.2.1 CAD 制图

结合最初的设计构想，以及参考现有的成品饮料机的整体框架结构，设计绘制了以亚克力板为基础的工程图纸。绘图软件使用 Autocad2019。调酒机的整体使用了十块亚克力板，分别为侧板、底板、前板、后板、漏液板、蠕动泵固定板、滴液板、顶板等。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

例图如下：

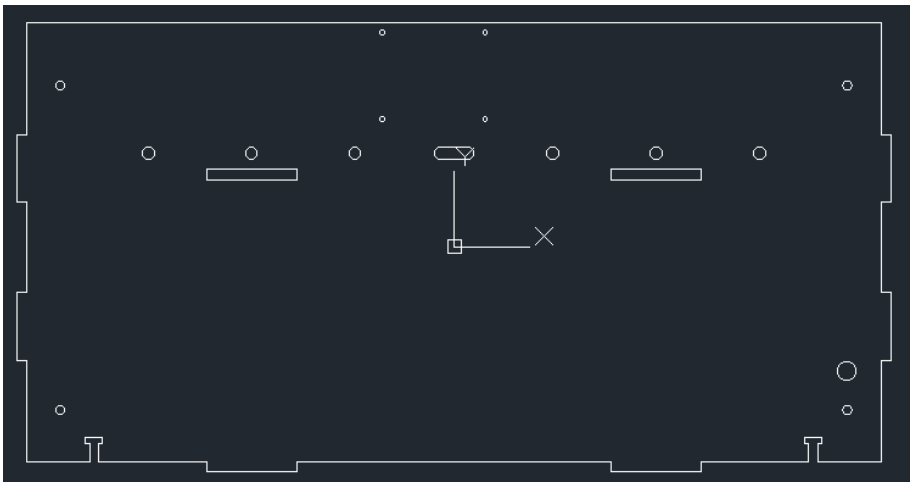


图 7.2.1 滴液板

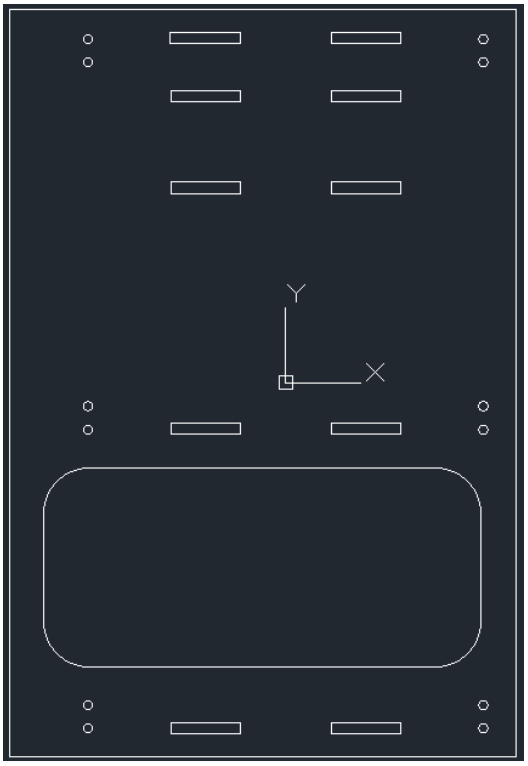


图 7.2.2 侧板

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

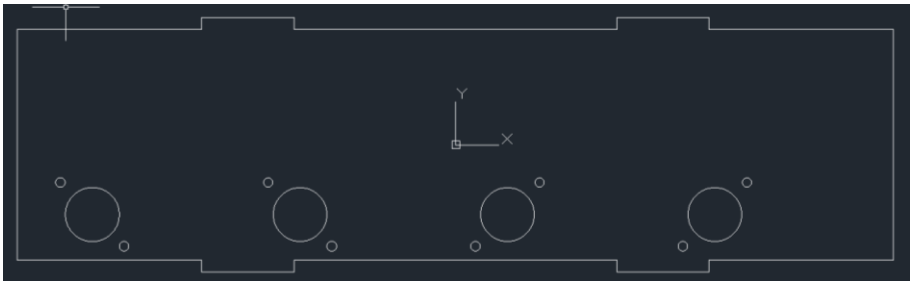


图 7.2.3 蠕动泵固定板

7.2.2 板材加工

整体使用了十块 6mm 亚克力板，分别由激光切割机、亚克力切割机和手锯完成切割工作。部分布线孔由台钻打孔。

图 7.2.4 半成品

由于团队的设计能力有限，根据需要，多次调整设计。例如，在前面板增加屏幕，在漏液板增加布线孔。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

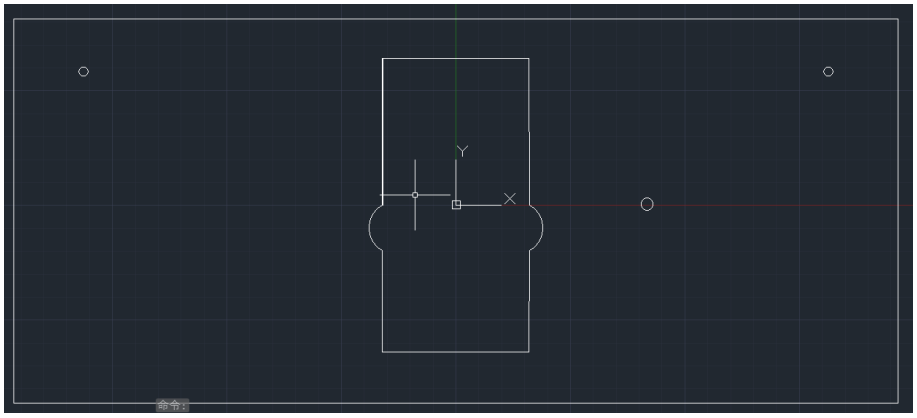


图 7.2.5 在前面板切割屏幕槽

图 7.2.6 最终展示效果

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

7.2.3 机械框架项目难点

1.元件尺寸未知

CAD 制图时，需要等待元件购买完毕后才能确定尺寸大小。在预留空间上，往往缺乏准确的估计。

2. 多段设计

本项目设有众多预期功能，板材制图、切割时需要预留足够的设计空间。项目过程中也需要一边设计，一边改进。

3.连接部分复杂

板块之间的连接，需要精准的定位及尺寸计算。

7.2.4 酒品分层

调酒器旨在对如鸡尾酒的分层效果进行实现。利用不同液体的密度差异（浓果汁、稀果汁、酒）可以实现液体的分层，实验效果如下图：

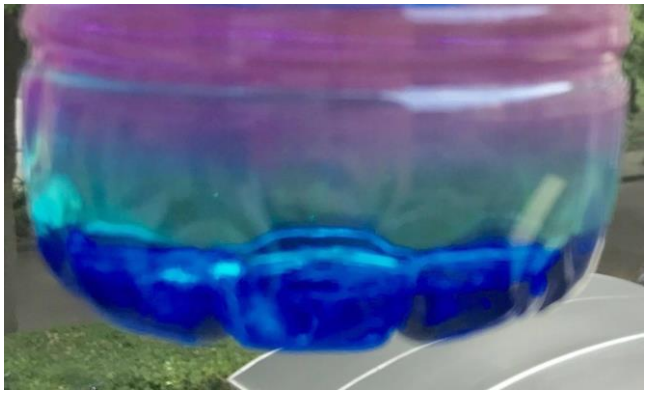


图 7.2.7 分层实验效果

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

8. 测试流程

1. 查看酒品介绍。选择“菜单点酒”模式，确认点 A 酒（出水口不放置杯子，原料 p1: 50mL，原料 p2: 50mL，原料 p3: 50mL），关闭提示弹窗，放好杯子，按住手机旁的 press 键 3~4 秒，观察 app 页面的心率显示，等待酒品制作完成，听音乐 A 并观察灯光效果 A，制作完成后观察酒品是否分三层，每层 50mL。

表 8.1“菜单点酒”模式数据（现象）记录表

酒品介绍能否正常显示：		
能否进入“菜单点酒”模式：		
是否有弹窗提示“请放置杯子”：		
轻按 press 键 app 中能否正确显示心率：		
放好杯子重新选择后是否有酒品流出：		
灯带是否为效果 A：		
音乐 A 是否正常播放：		
酒品分层效果：		
底层上液面刻度(mL)：	第二层上液面刻度(mL)：	第三层上液面刻度(mL)：

2. 选择“自定义配酒”模式。选择 3 种以下（含）原料的量（原料 p1: 50mL，原料 p2: 25mL，原料 p4: 75mL），点击确认（出水口放好杯子），按住手机旁的 press 键 3~4 秒，观察 app 页面的心率显示，等待酒品制作完成，听音乐 B 并观察灯光效果 B，制作完成后观察酒品 B 各层是否与自定义值相同。

表 8.2“自定义点酒”模式数据（现象）记录表

能否进入“自定义点酒”模式：		
能否滑动进度条确定各原料的量：		
轻按 press 键 app 中能否正确显示心率：		
灯带是否为效果 B：		
音乐 B 是否正常播放：		
酒品分层效果：		
底层上液面刻度(mL)：	第二层上液面刻度(mL)：	第三层上液面刻度(mL)：

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

3. 酒品 B 制作完成后原料 p2 余量不足，系统语音提示“余量不足，请及时加液”，点击进入管理员模式，输入密码，点击查看原料余量，加满原料，再次查看原料余量。查看制作历史记录。

表 8.3 “管理员”模式数据（现象）记录表

<table>
<tr><td colspan="2">有无语音提示余量不足：</td><td colspan="2"></td></tr>
<tr><td colspan="2">能否进入“管理员”模式：</td><td colspan="2"></td></tr>
<tr><td colspan="2">密码输入界面：</td><td colspan="2"></td></tr>
<tr><td colspan="4">余量查看 1：</td></tr>
<tr><td>原料 p1(mL)：</td><td>原料 p2(mL)：</td><td>原料 p3(mL)：</td><td>原料 p4(mL)：</td></tr>
<tr><td colspan="4">余量查看 2：</td></tr>
<tr><td>原料 p1(mL)：</td><td>原料 p2(mL)：</td><td>原料 p3(mL)：</td><td>原料 p4(mL)：</td></tr>
<tr><td colspan="2">历史记录查看：</td><td colspan="2"></td></tr>
</table>

4. 选择“语音输入心情”模式，说“今天好难过哦”，确认点 app 为用户推荐的酒品 C（出水口放好杯子，原料 p1: 75mL，原料 p3: 50mL），按住手机旁的 press 键 3~4 秒，观察 app 页面的心率显示，等待酒品制作完成，听音乐 C 并观察灯光效果 C，制作完成后观察酒品 C 是否分两层，各 75、50mL。

表 8.4“语音输入心情”模式数据（现象）记录表

能否进入“语音输入心情”模式：	
系统能否正确识别语音：	
是否有语音提示“我为您推荐酒品 C”：	
轻按 press 键 app 中能否正确显示心率：	
灯带是否为效果 C：	
音乐 C 是否正常播放：	
酒品分层效果：	
底层上液面刻度(mL)：	第二层上液面刻度(mL)：

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

9. 项目进展完成时间表

表 9.1 项目进展完成时间表

项目进展名称	预计完成时间	具体任务以及实现功能
项目组队确认	教学周第六周	组员间互相熟悉构思方案
项目总体方案初步设定	教学周第七周	完成项目初步规划，明确小组分工
硬件采购落实	教学周第八周	采购亚克力板和单片机、传感器、灯带等器材
熟悉硬件、文献调研　搭建测试平台、初步拼装板材	教学周第九、十周	熟悉开发平台和开源平台完成相关接口的的编写
依据实际平台调试	教学周第十二周	测量流速、蓝牙串口控制测试
系统整合	教学周第十三、四周	将系统联合整合，初步调试，整合成一个系统
按测试流程进行系统测试	教学周第十五周	完善测试方案、拍摄自测视频
总结并撰写报告	教学周第十六周	总结经验，撰写实验报告

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

10. 总结与致谢

本次工科创 4A 是我们大学生涯的最后一次工科创。从 1A 一路走来，我们每学期都根据课程指导完成规定项目的学习、制作与检测，掌握了许许多多关于单片机、电路板、焊接技术等的基础知识。终于，在最后一次实践中，我们提出自己的创意项目---Alphabar，利用各种网络资源、查阅多种数据资料并结合以往工科创的基础知识，共同完成了自助调酒机的制作。这种以原有知识为基础的创意项目制作过程，十分令人难忘：一则我们的所有设计都是按照自己的想法实现；二则在各种查阅资料的自学过程中，我们接触到了更多让自己感兴趣的内容。最后的成品凝集了所有人的心血，也是属于我们的独一无二的作品，无疑让小组每一个成员都感到骄傲。

在本次项目中，手机 App 组主要负责自助调酒机的人机交互界面，提供用户模式和管理员模式，分别进行自主点酒和后台管理，是一个纯软件的部分。我们利用 Android Studio 软件进行 App 开发，主要分为前端界面的设计以及后端功能的实现。在前端设计部分，我们查阅了网上各种好看实用的前端模块，例如倒计时模块，多种浮框板式等。我们在原有代码的基础上加以改进，设计出美观简洁的操作界面进行点酒操作和后台管理。在此过程中，我们对 Android 前端开发的框架有了基本的概念，学习了如何添加和修改前端的各个模块。我们参考了许多 git 开源代码，体会到了站在巨人肩膀上的感受和开源的强大力量。另一方面，我们体会到对于前端设计而言，首选需要梳理产品的功能，然后按照功能的重要程度在界面上进行合理排版，从而设计出方便用户使用的友好人机交互界面，这类似项目的产品经理需要考虑的内容，相较于以往的工科创，本次设计方案更加充分考虑了用户使用的便捷和舒适程度。

在后端功能实现部分，我们的代码包括语音模块，蓝牙传输模块，音乐播放模块这三大部分。其中，语音交互模块的实现基于讯飞开源平台，我们学习了如何通过调用各种讯飞的 Android SDK 来实现语音识别和语音合成的功能，体验到代码封装的重要性。蓝牙模块在实现过程中，我们遇见的最大问题有两点：1. 蓝牙传输内容的选择。我们利用字符串传递信息的过程中，会出现各种接收不匹配，接收错误的情况。这需要不断核查 App 端和 arduino 单片机端两部分的程序。另外在设计字符串时，要考虑如何使用更简单，更少量的字符串来传递我们需要的内容，否则会导致传输延迟等现象的发生。2. 蓝牙切换的不连续性。由于心率模块和 arduino 单片机均通过蓝牙

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

与 App 相连接，在蓝牙切换时，常常会发生连接突然断开或者信号不稳定的情况。因此，我们在后期查询了各种优化版本的蓝牙通讯代码，分别加以改进然后测试，旨在实现更加灵活的蓝牙切换。在这过程中，我们深感自主学习能力的提升，因为我们不但需要读懂查询到的代码，更需要去理解思考还有什么改进代码的方案。有时更需要结合多个代码的优点，整合出效果更佳的代码实现方案。

在外观实现部分也遇见了一定的困难，比如说 LED 的灯带控制代码的实现过程中，刚开始发现效果几乎不受到代码的控制，除了完全纯色的以外其余模式都很难实现自己想要的，但是从代码的角度来看又找不到问题。最终发现是 LED 灯带是 RGB 灯带而不是网上流行的 RGBW 灯带，导致定义出现问题，从而达不到自己需要的效果。所以在每个部件的采购和使用过程中要细心不能漏掉任何一个细节，不然必定是事半功倍。

在本次自主创意项目中，我们自主学习的能力得到了巨大的锻炼和提升，我们学会更加主动去思考问题，思考对现有代码的优化方案。

此外，我们体验到了团队合作的力量与乐趣。在项目初期，大家齐聚一堂讨论创意项目的设计。许多新奇的脑洞例如可在家里“跑来跑去”的自主调酒机等层出不穷。但当这些脑洞被提出时，负责相应模块的成员则立即思考该脑洞实现的难度，可行性等。他们站在工艺人员的角度更容易拒绝这些想法，因为将来时他们要去实现这些脑洞；而其他扮演“产品经理”的队友则很乐意坚持这样的脑洞，因为他们只负责提出想法而不负责实现。这类似于实际项目开发中，产品经理与开发人员的博弈，我们在讨论中也感受到了其中的乐趣。但最终，所有成员都很用心地为 AlphaBar 贡献自己的力量，负责地完成了自己的模块，并在实现过程中相互沟通问题，一起思考解决方法等。此外，在这项目中我们利用 git 平台进行代码共享，方便多人协同修改代码，这对我们之后的代码开发有巨大的帮助。

总体的来说这次工科创创意项目和之前的工科创的常规项目做对比，还是收获很多的。这次更像是一个未来在工作中需要实现的一个商业项目，自己也从方案设计到方案实现到最终调试每一步都有所参加。回顾这次创意项目，我们挑战了自己的创新能力和解决问题的能力，我们学会更主动地思考问题，相互协作。希望在今后的学习中，还能有更多机会参与到科技创新实践中，也很感激学校开设这样优秀的课程，这对于工科生来说是无比宝贵的财富。

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

最后，再次感谢在这次项目进展过程中耐心指导的老师和助教。你们的指点让我们得以迅速成长，你们辛苦了！

上海交通大学 电子信息与电气工程学院

地　址：东川路 800 号

邮　编：200240

11. 参考文献

[1Arduin 蓝牙模块与手机通信
https://blog.csdn.net/weixin_37272286/article/details/78016497?locationNum=10&fps=1
[2]科大讯飞开放平台
https://www.xfyun.cn
[3]Arduino 舵机控制
https://blog.csdn.net/c20081052/article/details/78254752
[4] 基于 arduino 控制全彩 LED 灯条
https://www.arduino.cn/thread-18033-1-1.html
[5]Android 开发教程

上海交通大学 电子信息与电气工程学院
地　址：东川路 800 号
邮　编：200240