

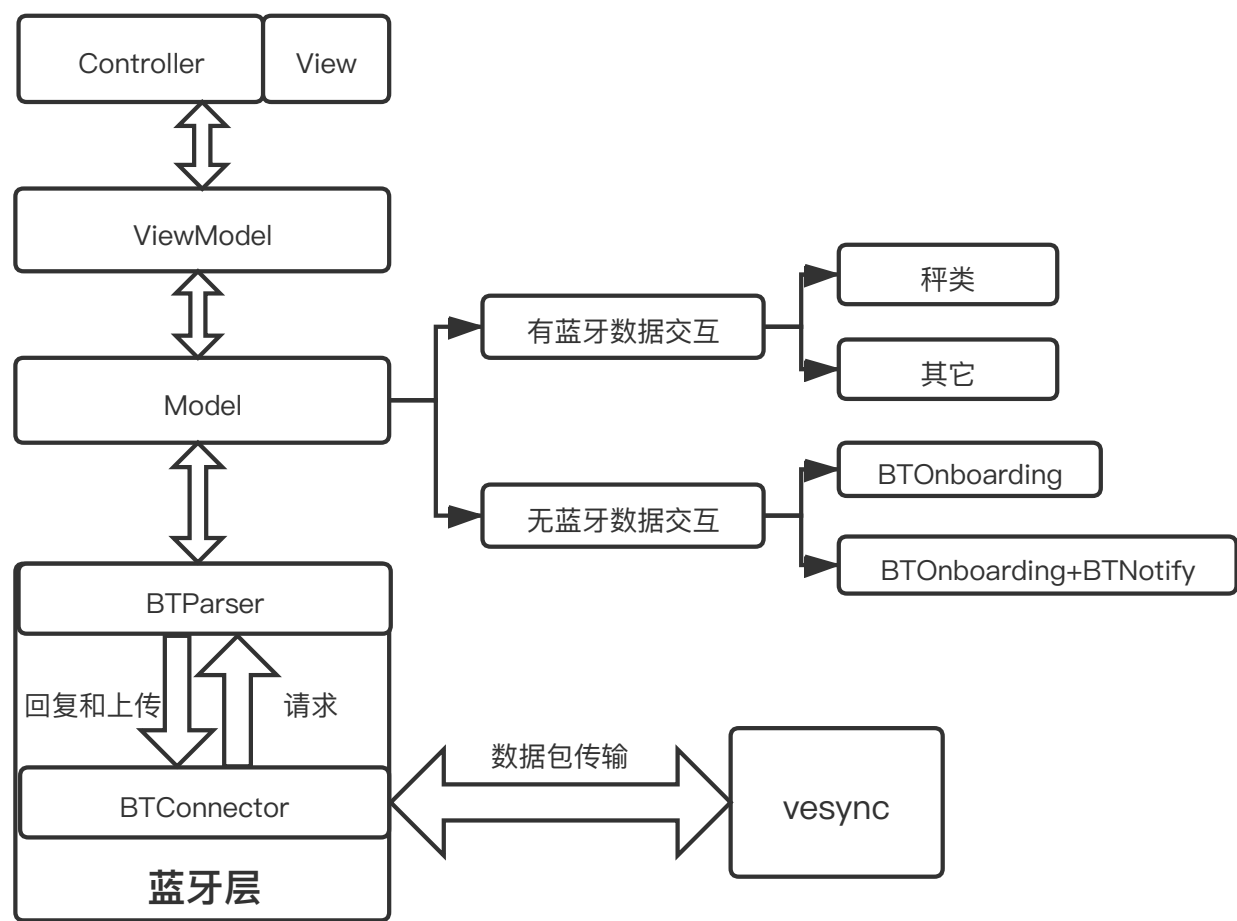
# 模拟项目第二版概要设计-郭韬

## 1. 需求描述

针对开发过程中有时缺少真实硬件外设测试的场景，开发一款测试APP，基于ios蓝牙外设开发模式，可任意选取设备类型并进行模拟，可以发出蓝牙广播并和vesync配网成功，针对用蓝牙进行数据交互的外设类型（如体脂秤），需要完整模拟其与vesync的数据交互，如上报当前称重数据和vesync同步设置时间到体脂秤等。

## 2. 整体架构

项目整体采用MVVM架构，其中底层为通用的蓝牙工具类，只负责发送蓝牙广播和建立蓝牙连接，此外还需要控制数据收发的解析类，用于解析vesync端下发数据和拼装本地上发数据包，再往上到model层，根据是否有蓝牙数据传输和产品功能分为几个大类，再往上的viewmodel和viewController也根据model大类来分别设计，数据的双向绑定通过RxSwift实现，具体结构如下图：



## 3. 模块设计

### 3.1 蓝牙连接类（BTConnector）

蓝牙连接类BTConnector是继承自NSObject类的单例，在extension中加入CBPeripheralManagerDelegate，蓝牙功能都通过类型为CBPeripheralManager的成员变量peripheralManager来实现。通过peripheralManagerDidUpdateState方法检测蓝牙状态，若可用则开启定时器发送蓝牙广播，设备链接订阅成功后回调方法中关闭广播定时器并开启心跳包定时器，在自定义的服务端口上发空数据保证不断连。将接收到的vesync写入请求都交给BTParser类处理，再接收BTPaser返回的答复包发送给vesync，设备主动上报数据和回发答复包都是走sendData方法更新notify端口。

发送广播时的广播包格式为公司id + 设备mac地址 + 设备型号，其中公司id固定，设备型号信息从上层model中获取，设备mac地址由于ios无法获取本机mac信息，可以采取随机生成和用户修改相结合的方式，将当前模拟的mac地址显示在主页，正常来说不会重复，不会影响正常的连接和数据收发。

### 3.2 蓝牙数据收发类（BTParser）

首先需要获取当前模拟设备的设备名，参照vesync主工程中的主页设备卡片和DeviceManager，在主页选取完设备后通过DeviceManager获取设备的DeviceModel，基本的DeviceModel信息为设备名、设备大类型号、设备型号和设备配网类型，根据设备型号和操作码去选择对应的viewmodel中方法来更新当前产品的model信息，这里更新的model信息是类似历史称重数据、称重单位这样的具体信息，model在选择完模拟设备时根据设备型号创建。

解析数据时采取将Data转为字符串的方式来读取整个请求包，再将字符串截取以获得关键信息，回发和上报数据时数据包拼接通过swift自带的model转字典方法即Codable协议，再将得到的字典通过JSONSerialization方法即可得到Data格式数据。查阅网上资料得知计算机默认存储是小端模式，序列化得到的Data格式应该也遵循小端模式，具体还需要测试。

### 3.3 Model类

Model类大致可以分为4个类别，首先是无蓝牙数据交互的设备（只做第一步的配网连接），往下分为只有BTOnBoarding和加了BTNotify的两种模式，这两类实现的细节区别还需要再研究，模式的信息记录在DeviceModel中的设备配网类型中，可以用0、1、2、3的数值分别指代常规蓝牙配网、加密蓝牙配网、蓝牙wifi配网和特殊的蓝牙wifi配网。在蓝牙连接订阅成功的回调方法中就加入对该值的判断，并设置只需配网的设备统一跳转到配网信息页，显示配网成功的进度。

涉及到数据包传输的模拟设备可以用一个RequestModel存储数据包各个字段，根据操作码的对应操作修改model后将修改响应到viewmodel去更新vc，之后根据请求解析的RequestModel和相关回复数据拼接回复包。

### 3.4 ViewController类

界面只涉及选择设备的主页和选择完设备后各设备的主页（配网信息或者数据传输），默认蓝牙开启，在部分设备主页中可以添加关闭蓝牙连接的按钮（如体脂秤的本地添加称重数据），只需对BTConnector类的成员变量peripheralManager赋值nil和重新初始化即可。各设备的主页尽量用一页展示数据即可，对需要传输数据的设备设置一个BaseHomeVC页，所有的体脂秤和营养秤类分别继承为BodyFatScaleHomeVC页和NutritionScaleHomeVC页，根据当前秤的具体型号显示对应功能，其它如血压计和鹅颈壶的单品都继承自BaseHomeVC页单独编写，可以做成一个大的tableview，参照vesync中SettingBaseVC页，每行显示参数信息并且可以编辑修改。

选择设备的主页参照vesync的Add Device页，左侧为产品大类，右侧为具体产品型号和图片，点击即可进入设备主页，界面上部区域用于显示和修改模拟设备的MAC地址。界面约束用SnapKit实现。

## 3.5 工具类 (Tools)

整个类由静态方法组成，包括但不限于数据展示格式的转换、UTC时间戳的转换、随机MAC地址的生成和全机型的UI界面适配，还可以添加错误处理的方法、打印调试信息等，禁止外部调用Tools工具类创建示。此外还将导入方家浩的日志抓取工具，实现脱离xcode的控制台日志显示，更加便于测试。