**基于rt-thread+springboot+uniapp+mssql的IOT小车设计报告**

**2022年9月21日**

目 录

[一.系统概览 1](#_Toc73620013)

[1.1 系统简介 1](#_Toc73620014)

[1.2 系统框图 1](#_Toc73620015)

[1.3 系统整体框架概要...........................................1](#_Toc73620015)

[二.手机APP部分 2](#_Toc73620016)

[2.1 手机APP展示 2](#_Toc73620017)

[2.2 手机APP技术概要 6](#_Toc73620018)

[三. 嵌入式小车部分 7](#_Toc73620020)

[3.1 嵌入式小车展示 7](#_Toc73620021)

[3.2 嵌入式小车技术概要 8](#_Toc73620022)

[四. 后台服务端部分 9](#_Toc73620024)

[4.1 服务端概览 9](#_Toc73620025)

[4.2 服务端技术概要 9](#_Toc73620026)

[五. 数据库部分 1](#_Toc73620029)0

[5.1 数据库概览 1](#_Toc73620030)0

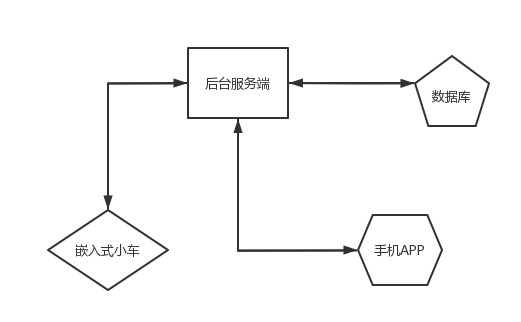
[5.2 数据库技术概要 1](#_Toc73620031)0

**一、系统概览**

**1.1系统简介**

本系统是使用stm32l475vet6正点原子潘多拉物联网开发板+rt-thread嵌入式实时操作系统开发嵌入式小车，springboot开发后台服务端，uniapp开发前端手机APP以及mssql作为数据库系统的IOT（物联网）系统

**1.2系统框图**

****

**1.3系统整体框架概要**

本系统以后台服务端+数据库为核心，将手机APP与嵌入式小车都作为了前端部分。其运行的大体过程如下：

小车上线后，向服务端发送请求，服务端注册该小车信息到数据库。用户通过手机APP注册、登录、更改个人信息、绑定小车、操控小车时，手机APP也会向服务端发送请求，而后服务端对数据库进行相应的操作。后端服务端会进行自动检测，若小车没有定时发送过数据（心跳包）到服务端，则判断小车离线。

所有端与端的通信，都是通过响应式的http api实现的，从而实现了前后端分离，并达到数据高效组织，信息高效率传输的目的。

**二、手机APP部分**

**2.1手机APP展示**

（1）用户登录注册功能。

可实现不同用户的登录以及注册，以方便不同用户控制不同的小车。成功登录后便可进入控制页面。

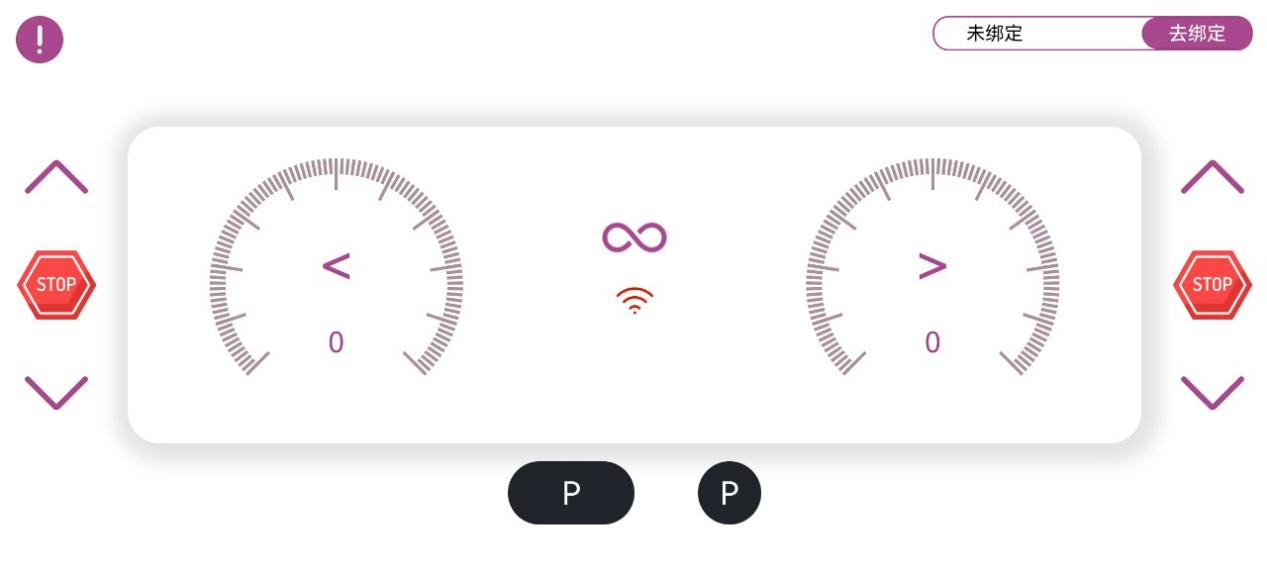




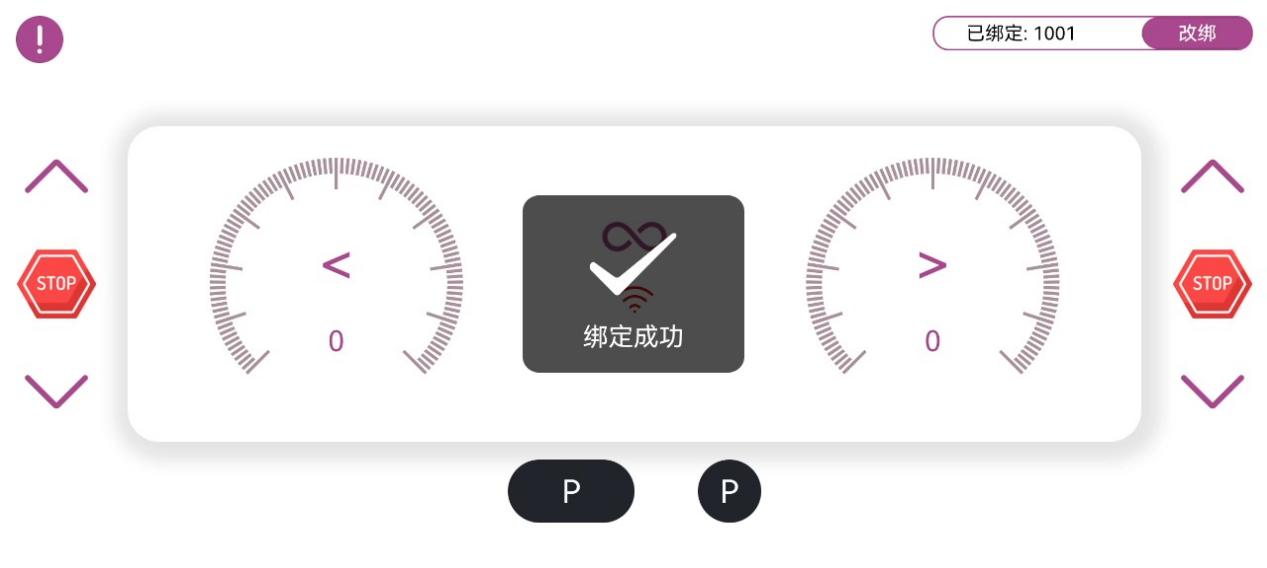
（2）用户扫码绑定功能

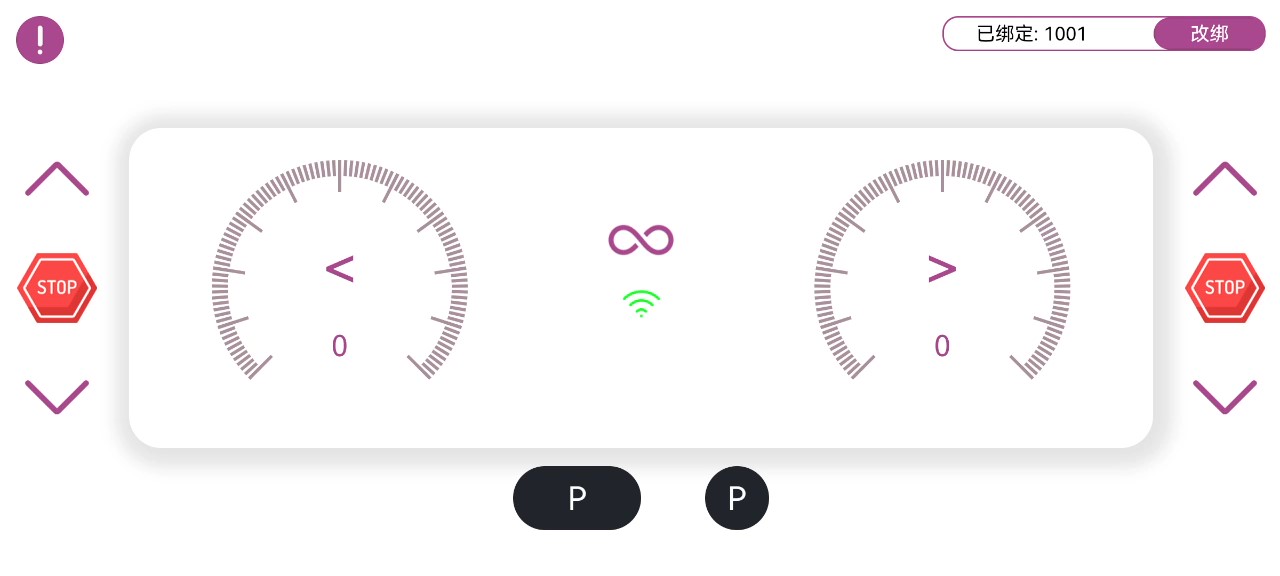
在控制界面，初始情况下，右上角绑定状态栏会显示未绑定，则此时无法控制小车。点击右上角去绑定按键则可进行扫码绑定。扫描小车上lcd屏幕二维码即可绑定对应小车。绑定完成后会显示绑定成功图标，并显示已绑定的小车id，去绑定也变为了改绑。稍微等待几秒后，屏幕中间信号图标由红变绿，则可开始控制小车。

同时支持多用户绑定不同的小车，以便不同用户控制各自不同的小车。也支持用户绑定小车后改变绑定的小车。



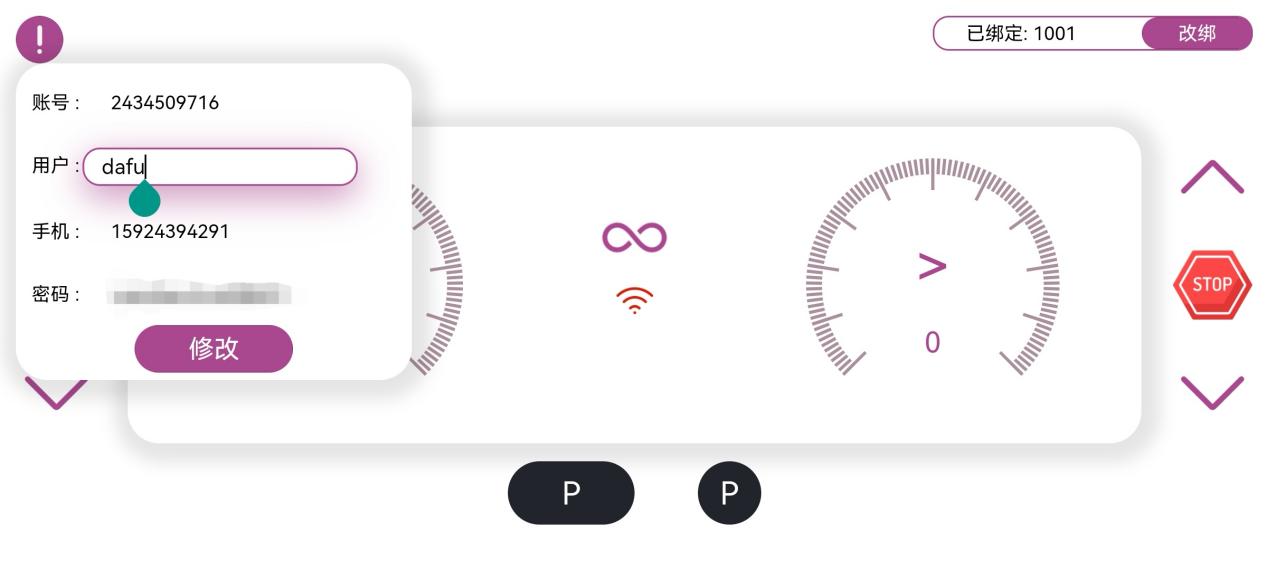






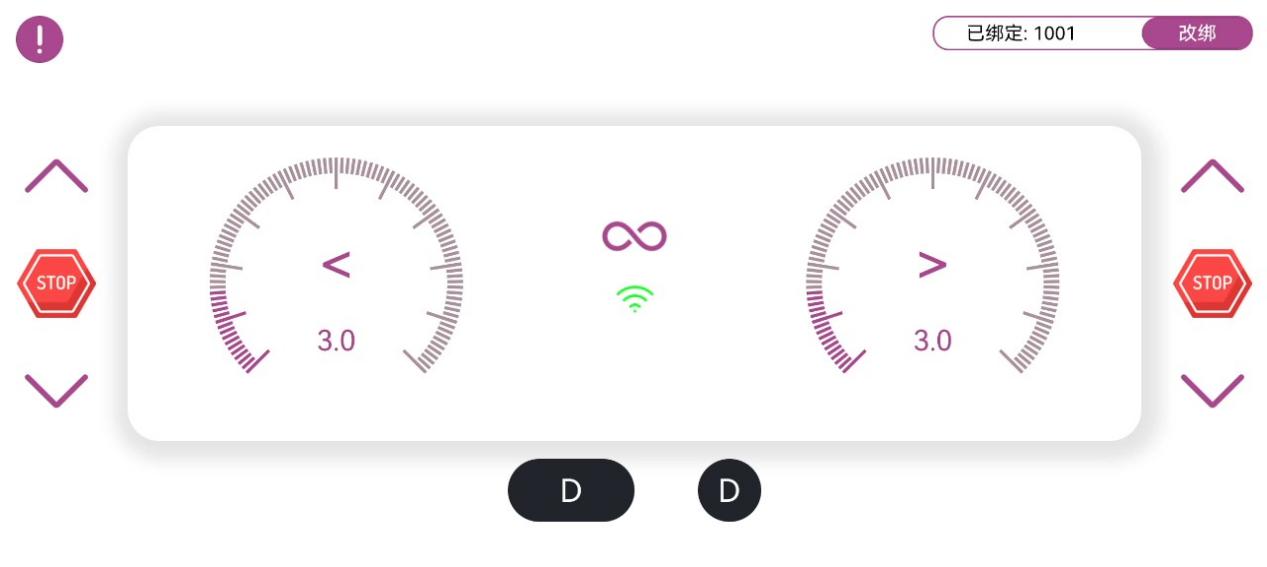
（3）用户信息更改功能

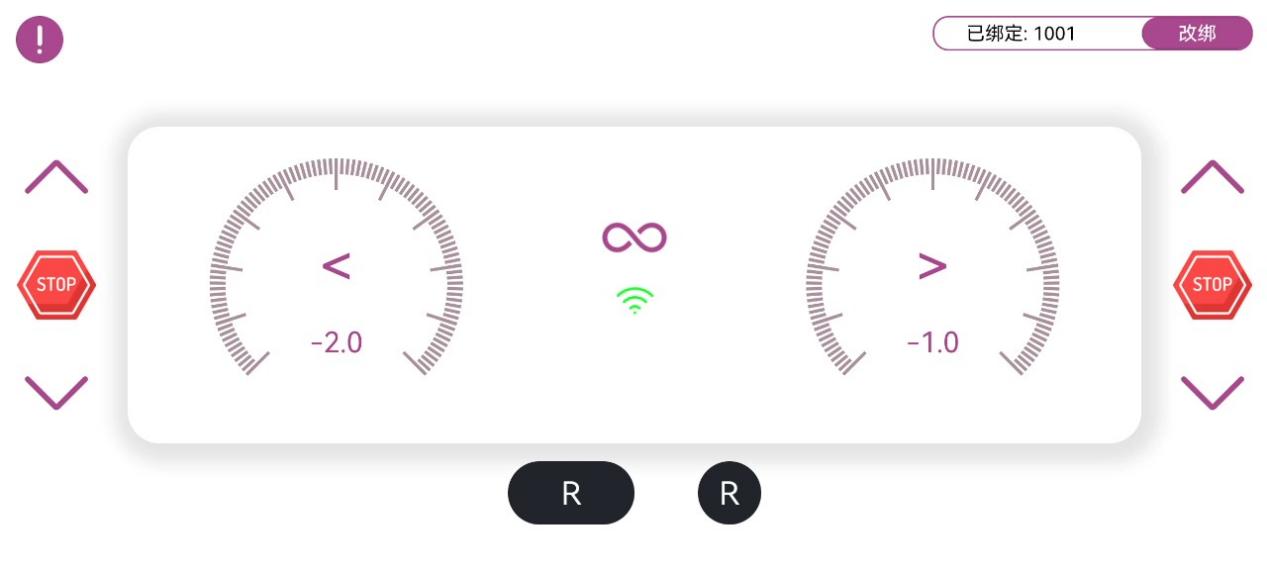
点击左上角！按键，可显示用户信息，点击可更改。

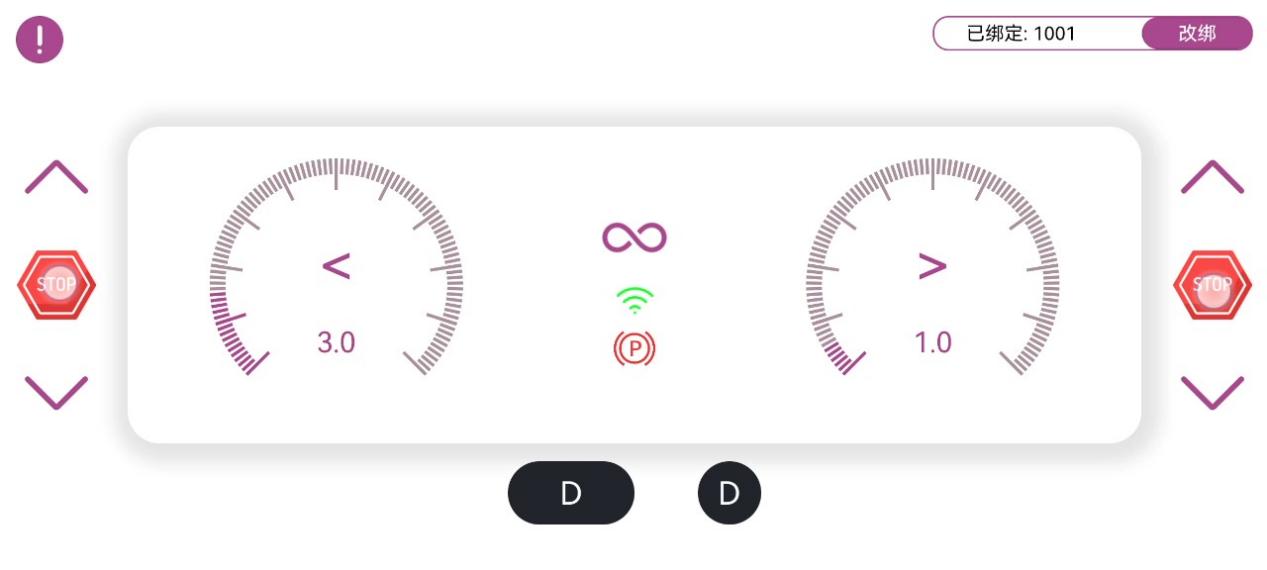
****

（4）小车操控功能

支持对左右履带的加减速操作，以实现各种运动。支持刹车功能，刹车时制动灯会亮起。同时左右履带速度会实时反馈到仪表盘上。

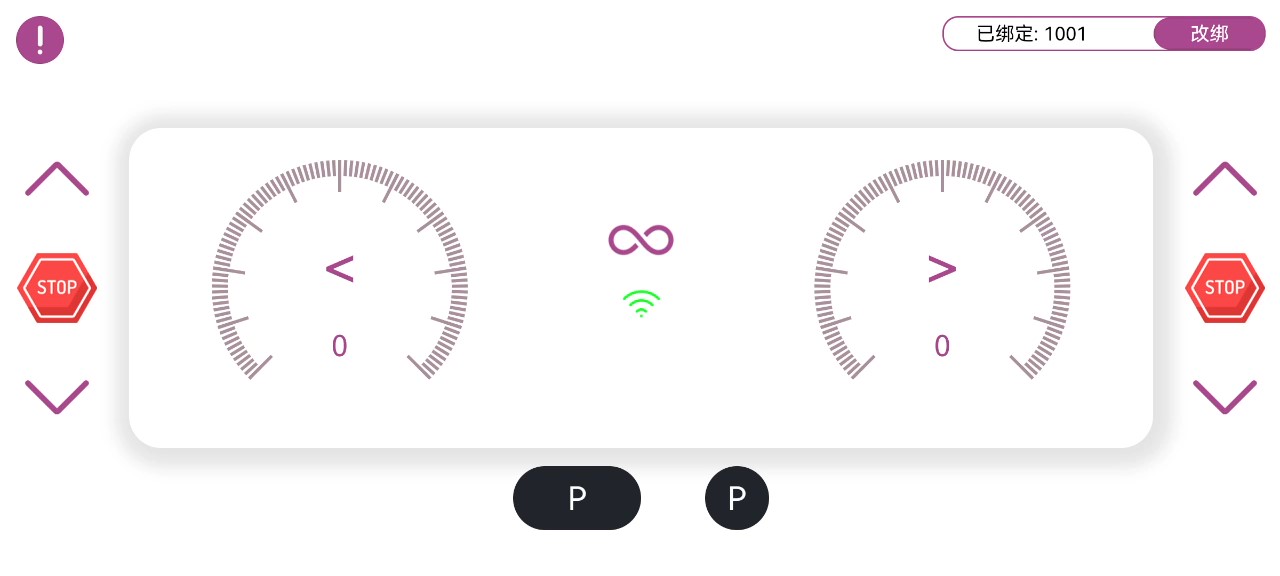
****

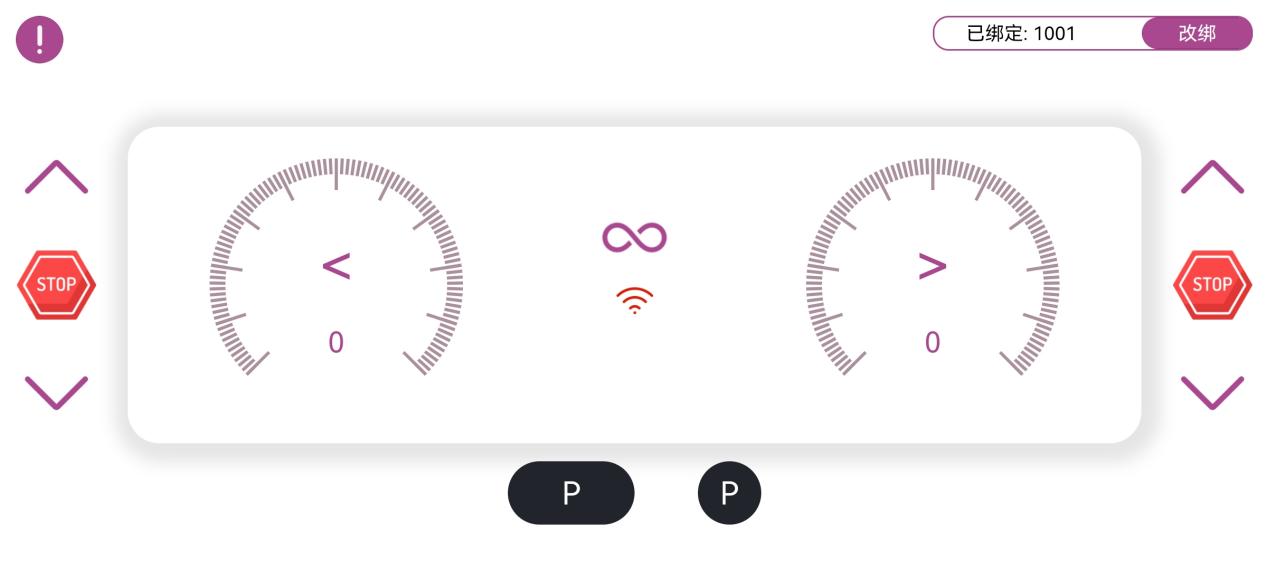
****

****

（5）小车在线离线状态显示

小车在线则信号图标为绿色，小车离线则信号图标为红色

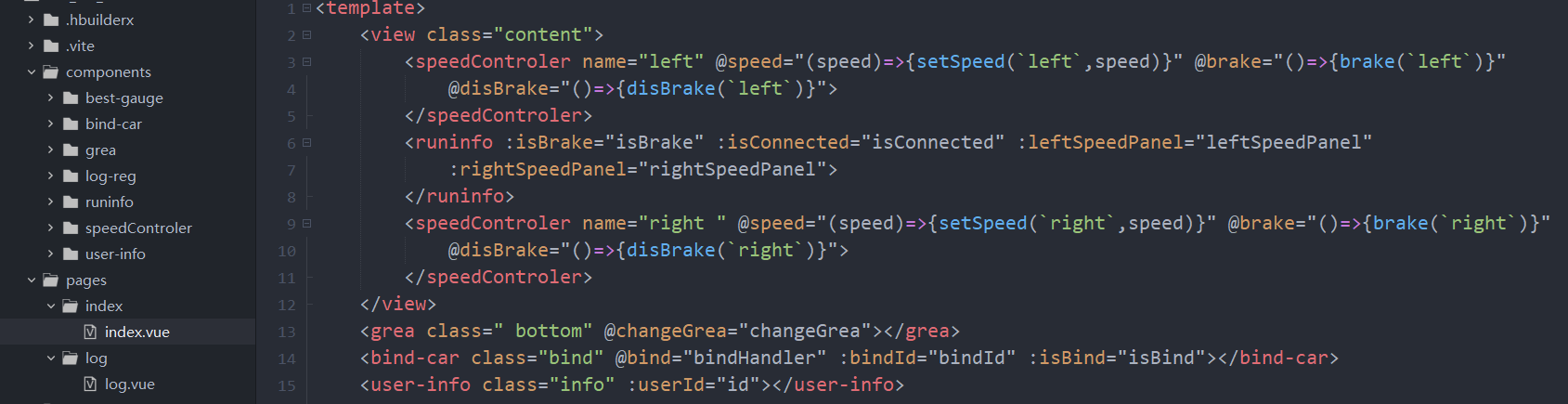




**2.2手机APP技术概要**

前端手机APP使用了uniapp开发，实现了可在ios、android、web、小程序等多端运行的目的。

使用了单文件组件的开发方法，使开发过程更加清晰。通过组件与组件的相互协调通信，最后实现了这样一个ui精美的前端可跨端的手机APP



**三、嵌入式小车部分**

**3.1嵌入式小车展示**

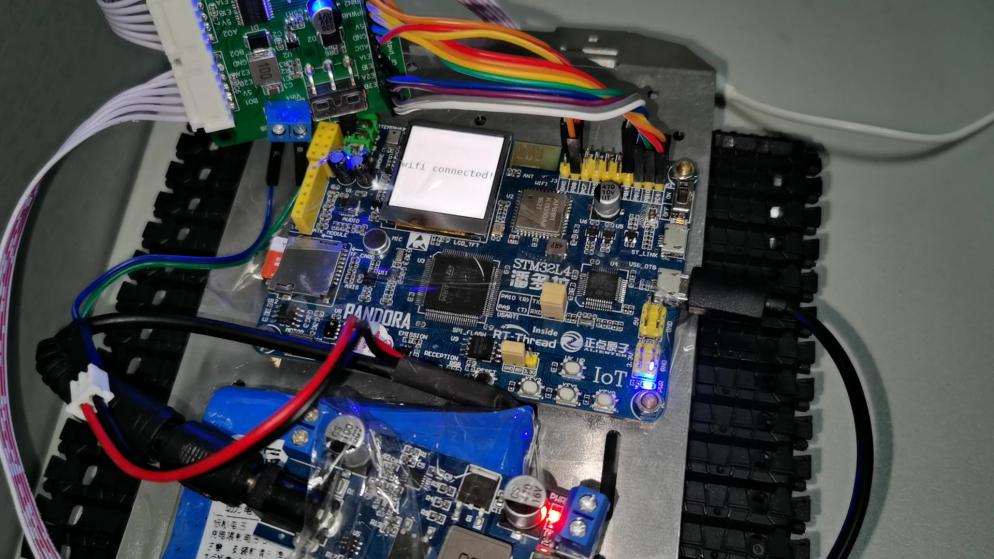
（1）开机载入

屏幕会显示loading



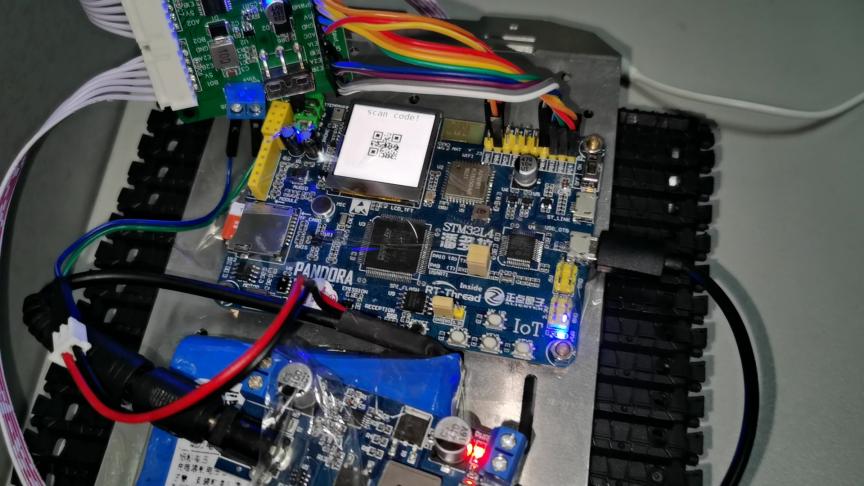
（2）网络连接

若已经成功连接过wifi热点，则会自动连接，显示wifi connected。若wifi连接未成功，或没有连接过wifi，则会提示使用串口进行网络配置（串口波特率115200），使用set\_wifi <ssid> <password> 命令进行网络连接。



（3）app绑定

会显示一个二维码，通过手机app扫描该二维码，即可与此小车绑定，进行操控。



（4）小车运行

在配置网络以及绑定app完成后，用户便可以通过手机app操控小车的

运动。

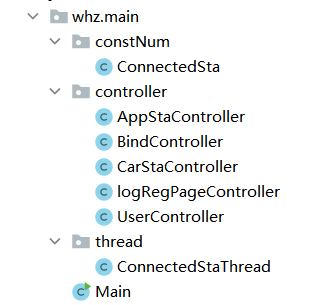


**3.2嵌入式小车技术概要**

使用了rt-thread嵌入式实时操作系统，并使用了以stm32l475vet6为主控的潘多拉物联网开发板。该开发板资源丰富，尤其搭载了ap6181 SDIO Wifi芯片，在rt-thread提供的bsp加持下，使得开发过程十分轻松。小车的难点主要集中在连接网络以及使用http api上。而通过rt-thread的网络组件以及wlan框架，这些困难便迎刃而解。

**四、后台服务端部分**

**4.1服务端概览**

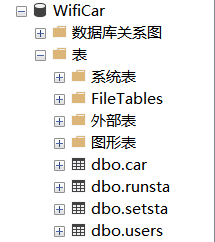


**4.2服务端技术概要**

服务端使用了springboot进行开发，从而构建了一套响应式的http api系统。由于所有api都是响应式的，因此不会出现服务端卡死而影响操控的是实时性的情况。同时也构建出了一套前后端分离的系统，方便后续进行系统的维护以及升级。

**五、数据库部分**

**5.1数据库概览**



**5.2数据库技术概要**

数据库部分使用了mssql作为数据库系统，通过jdbc与后台服务端进行相互通信。

实体ER图如下：

