

linkon温控器

版本号V002R001

林肯温控器/linkon设计文档

**深圳市银河风云网络系统股份有限公司**

**Shenzhen GALAXYWIND Network Systems Co., Ltd.**

地址：深圳市高新技术产业园区北区新西路5号银河风云大厦

Address: GALAXYWIND building, No.5 Xinxi road, Shenzhen High-Tech Industry Park, NanShan district, China

邮编 P.C.: 518055

电话 Tel: +86-755-83400088

传真 FAX: +86-755-86139063

网站 Web: [http://www.galaxywind.com](http://www.galaxywind.com/)

客服 Hotline: 400 000 9879

企业QQ: 4000009879

微信公众号 WeChat:



版权声明

版权所有©深圳市银河风云网络系统股份有限公司2016。深圳市银河风云网络系统股份有限公司（以下简称"银河风云"）对本资料进行版权声明，未经银河风云书面许可，任何单位及个人不得以任何方式或理由对本资料的任何内容进行复制、修改、抄录、传播。

商标声明

"银河风云"，"GALAXYWIND"等是深圳市风云实业有限公司的注册商标，本文档提及的其他商标、服务标志、注册商标及注册服务标志均为其所有者拥有。

目录

**1** 简介 2

1.1 整体目标 2

1.2 主要功能 2

1.3 名称与取值范围 2

**2** 用例设计 2

2.1 设备管理 2

2.2 设备控制 3

2.3 定时任务 3

**3** 枚举设计 3

3.1 设备连接状态 3

3.2 设备运行状态 3

3.3 林肯温控器风速 3

3.4 林肯温控器模式 4

3.5 林肯温控器情景 4

3.6 林肯温控器定时器周期重复 4

3.7 林肯温控器定时器类别 4

3.8 林肯温控器属性组 4

**4** 实例类设计 5

4.1 林肯温控器设备类 5

4.2 林肯温控器定时器类 5

4.3 林肯温控器定时器周期时间类 6

**5** 管理类设计 6

5.1 设备管理器类 6

5.2 通知实例类 7

5.3 温度单位管理类 8

5.4 反馈管理类 8

**6** 算法 8

6.1 定时器冲突检测算法 8

**7** 实现 9

7.1 多语言 9

7.2 屏幕适配 9

7.3 统一简单界面元素 9

7.4 列表为空时的提示界面 9

7.5 列表侧滑菜单 9

7.6 搜索设备界面的波纹扩散按钮 9

7.7 温控器遥控面板状态参数展示界面 9

7.8 温控器遥控面板温度设置界面 9

7.9 选择控制器 10

7.10 周期选择 10

7.11 其他细节问题 10

# 简介

## 整体目标

通过手机App管理林肯温控器。

## 主要功能

1. 管理温控器设备列表，包含设备的读取，搜索，新增，编辑，删除。
2. 温控器当前状态的显示。
3. 温控器控制，包含开关，切换情景，模式，风速，调节温度，儿童锁，延时开关。
4. 温控器定时任务列表控制，包含定时任务的读取，新增，编辑，删除。

## 名称与取值范围

1. 设备

指Linkon温控器，主要由序列号，密码，昵称以及其他运行状态参数组成。

1. 连接状态

在线，离线，指代设备与服务器间的通讯状态。

1. 运行状态

待机，开机

1. 情景

恒温，节能，离家

1. 模式

制冷，制热，换气

1. 风速

低风，中风，高风

1. 温度

5℃至35℃（间隔0.5℃）

1. 周期

周一，二，三，四，五，六，日

1. 时间

00：00 至 23：59（间隔1分钟）

1. 定时任务

开关定时，阶段定时

# 用例设计

## 设备管理

* 1. 获取设备列表

本地读取设备列表。

* 1. 搜索新设备

通过广播SSID与Wifi密码，完成设备的自动匹配与自动添加。

* 1. 添加已配置过的设备

手动输入序列号和密码完成设备添加。

* 1. 修改设备昵称

重新设置设备昵称。

* 1. 修改设备密码

须提供旧密码与新密码完成密码修改。

* 1. 删除设备

删除设备本地记录。

## 设备控制

* 1. 显示当前设备状态

包含运行状态，当前室内湿度，当前室内温度，设定温度，模式，情景，风速，延时倒计时，儿童锁。

* 1. 切换设备运行状态
  2. 切换情景

循环切换三种情景。

* 1. 切换模式

循环切换三种模式。

* 1. 切换风速

循环切换三种风速。

* 1. 设定温度

手动选择设定温度。

* 1. 儿童锁

切换儿童锁开关。

* 1. 延时开关机

当前设备处于待机状态时，延时开机。当前设备处于开机状态时，延时关机。设备运行状态更改时，清空延时控制。延时开关每次增加30分钟，超过8小时则清零。

## 定时任务

* 1. 获取定时任务列表
  2. 新增定时任务
  3. 编辑定时任务
  4. 删除定时任务

# 枚举设计

## 设备连接状态

|  |  |  |
| --- | --- | --- |
| DeviceConnectionState | 取值 | 说明 |
| DeviceConnectionStateOffLine | 1 | 离线 |
| DeviceConnectionStateOnLine | 2 | 在线 |

## 设备运行状态

|  |  |  |
| --- | --- | --- |
| DeviceRunningState | 取值 | 说明 |
| DeviceRunningStateTurnOFF | 1 | 待机 |
| DeviceRunningStateTurnON | 2 | 开机 |

## 林肯温控器风速

|  |  |  |
| --- | --- | --- |
| LinKonWind | 取值 | 说明 |
| LinKonWindLow | 1 | 低风 |
| LinKonWindMedium | 2 | 中风 |
| LinKonWindHigh | 3 | 高风 |

## 林肯温控器模式

|  |  |  |
| --- | --- | --- |
| LinKonMode | 取值 | 说明 |
| LinKonModeCool | 1 | 制冷 |
| LinKonModeHot | 2 | 制热 |
| LinKonModeAir | 3 | 换气 |

## 林肯温控器情景

|  |  |  |
| --- | --- | --- |
| LinKonScene | 取值 | 说明 |
| LinKonSceneConstant | 1 | 恒温 |
| LinKonSceneGreen | 2 | 节能 |
| LinKonSceneLeave | 3 | 离家 |

## 林肯温控器定时器周期重复

|  |  |  |
| --- | --- | --- |
| TimerRepeat | 取值 | 说明 |
| TimerRepeatNone | 0 | 无 |
| TimerRepeatSunday | 1<<0 | 周日 |
| TimerRepeatMonday | 1<<1 | 周一 |
| TimerRepeatTuesday | 1<<2 | 周二 |
| TimerRepeatWednesday | 1<<3 | 周三 |
| TimerRepeatThursday | 1<<4 | 周四 |
| TimerRepeatFriday | 1<<5 | 周五 |
| TimerRepeatSaturday | 1<<6 | 周六 |
| TimerRepeatEveryDay | 127 | 每天 |

## 林肯温控器定时器类别

|  |  |  |
| --- | --- | --- |
| LinKonTimerTaskType | 取值 | 说明 |
| LinKonTimerTaskTypeSwitch | 1 | 开关定时 |
| LinKonTimerTaskTypeStage | 2 | 阶段定时 |

## 林肯温控器属性组

|  |  |  |
| --- | --- | --- |
| LinKonPropertyGroup | 取值 | 说明 |
| LinKonPropertyGroupNone | 0 | 无 |
| LinKonPropertyGroupBinding | 1 << 0 | 绑定属性，包括昵称与密码 |
| LinKonPropertyGroupState | 1 << 1 | 状态属性，包括连接状态与运行状态 |
| LinKonPropertyGroupSetting | 1 << 2 | 设定属性，包括当前温度，湿度，设定温度，情景，模式，风速，儿童锁，延时开关机 |
| LinKonPropertyGroupTimer | 1 << 3 | 定时器属性，包含新增定时器，修改定时器，删除定时器 |

# 实例类设计

## 林肯温控器设备类

|  |  |  |  |
| --- | --- | --- | --- |
| LinKonDevice | 说明 | 取值说明 | 备注 |
| sn | 序列号 | String |  |
| password | 密码 | String |  |
| nickname | 昵称 | String |  |
| connection | 连接状态 | DeviceConnectionState |  |
| running | 运行状态 | DeviceRunningState |  |
| setting | 设定温度 | float | 取值范围5.0到35.0(间隔0.5) |
| level | 风速 | LinKonWind |  |
| mode | 模式 | LinKonMode |  |
| scene | 情景 | LinKonScene |  |
| humidity | 当前湿度 | float | 取值范围0.0到1.0 |
| temperature | 当前温度 | float | 取值范围5.0到35.0(间隔0.5) |
| lock | 儿童锁 | bool |  |
| delay | 延时截止时间 | double | 时间戳 |
| savedTimerArray | 定时器数组 | [LinKonTimerTask] |  |

* **接口：**

1. 添加定时器

输入：定时器

输出：是否添加成功（当定时器发生冲突时，添加失败）

1. 移除定时器

输入：定时器

输出：是否移除成功（当找不到指定定时器时，移除失败）

1. 修改定时器

输入：定时器

输出：是否修改成功（当找不到或者发生冲突时，修改失败）

## 林肯温控器定时器类

|  |  |  |  |
| --- | --- | --- | --- |
| LinKonTimerTask | 说明 | 取值说明 | 备注 |
| number | 标识 | String | 任务的唯一标识 |
| type | 任务类别 | LinKonTimerTaskType |  |
| validate | 是否有效 | bool |  |
| timeFrom | 起始时间 | int | 若为开关定时任务，则表示定时时间；  若为阶段定时任务，则表示起始时间。 |
| timeTo | 结束时间 | int | 阶段任务的结束时间，对开关定时任务无效。 |
| repeat | 重复 | Byte | 取值范围0到127，按位计算 |
| running | 运行状态 | DeviceRunningState | 对阶段定时任务无效。 |
| setting | 温度（设定温度） | float | 取值范围5.0到35.0(间隔0.5) |
| level | 风速 | LinKonWind |  |
| mode | 模式 | LinKonMode |  |
| scene | 情景 | LinKonScene |  |
| rangeArray | 周期时间数组 | [LinKonTimerRange] |  |

* **接口：**

1. 比较定时器是否冲突

输入：定时器

输出：是否冲突（与当前定时器比较是否冲突）

1. 重置周期时间数组

输入：无

输出：无（当定时器发生改变时，应重新计算周期时间数组）

## 林肯温控器定时器周期时间类

由于定时器类的时间可能隔天或者跨天，不便于冲突检测，因此设计此类。此类的时间均保证在当天，且起始时间必定早于结束时间。

|  |  |  |  |
| --- | --- | --- | --- |
| LinKonTimerRange | 说明 | 取值说明 | 备注 |
| timeAt | 时间点 | int | 时间点 |
| timeFrom | 起始时间 | int | 起始时间 |
| timeTo | 结束时间 | int | 结束时间 |
| repeat | 周期点 | TimerRepeat | 星期几 |

# 管理类设计

## 设备管理器类

* **目标：**

设备信息全局管理，以设备序列号作为设备的唯一标识，保证同一个设备在内存中仅有一个实例对象与之对应。界面之间直接传递设备序列号，通过序列号再找到唯一的对象，以此加强实例对象唯一的设计。

* **实现：**

|  |  |  |  |
| --- | --- | --- | --- |
| DeviceManager | 说明 | 取值 |  |
| deviceArray | 设备数组 | [LinKonDevice] | 设备实例对象列表 |
| listBlockArray | 列表监听对象数组 | [NotifyTarget] |  |
| editBlockArray | 属性监听对象数组 | [NotifyTarget] |  |

* **接口——设备管理：**

1. 获取单个设备（参数：设备序列号）。

遍历设备数组，找到对应的设备实例对象。

1. 新增设备。（参数：设备实例对象）

向设备数组添加记录。

注：添加之前，应先判断是否有旧的设备记录。若有旧设备记录，则先删除旧设备，再添加新设备。

1. 删除设备。（参数：设备序列号）

遍历设备数组，删除对应的记录。最后要记得清空通知中心关于该设备的通知记录。

* **接口——列表监听：**

1. 监听设备列表。（参数：监听者，监听回调）

添加监听时，会立即回调一次，以替代获取设备列表的功能。

* **接口——属性监听：**

1. 监听单个设备属性。（参数：监听者，设备sn，属性组，监听回调）

设备sn可以为空，表示监听所有设备。属性组可以组合。添加监听时，会立即回调一次，以返回最新的设备信息。

1. 编辑设备属性。（参数：设备sn，属性名称，编辑后的值）

修改单个设备属性，触发通知回调。

* **接口——定时器：**

1. 获取单个定时器。（参数：定时器标识，设备sn）

遍历设备定时器列表，返回指定标识的定时器。

1. 添加定时器。（参数：定时器，设备sn）

向指定设备添加新的定时器。若发生定时器冲突则添加失败。

1. 移除定时器。（参数：定时器，设备sn）

移除指定设备已有的指定定时器。

1. 修改定时器。（参数：定时器，设备sn）

修改指定设备已有的定时器。若发生定时器冲突则修改失败。

## 通知实例类

* **目标：**

服务于设备管理类，一个监听者可以监听设备列表，可以监听特定的一个设备属性组组合，也可以监听全部设备的属性组组合。

* **实现：**

|  |  |  |  |
| --- | --- | --- | --- |
| NotifyTarget | 说明 | 取值 | 备注 |
| listener | 通知接收体 | id |  |
| sign | 监听对象唯一标识 | String | 可以为空，表示全部。 |
| listBlock | 列表监听回调 | Block |  |
| propertyGroup | 监听属性组组合 | Byte |  |
| groupBlock | 属性监听回调 | Block |  |

注意：此处object使用弱引用，避免内存泄露。

## 温度单位管理类

整个项目中，所涉及的温度均按摄氏度数值存储，仅在显示时，通过该类计算出实际的显示数值与单位。具体用户设置储存在NSUserDefaults中。

## 反馈管理类

反馈包括声音反馈与振动反馈，均通过播放系统声音实现。具体用户设置储存在NSUserDefaults中。

# 算法

## 定时器冲突检测算法

* **算法面对的实际情况：**

1. 按类型区分的话，定时器分为开关定时器与阶段定时器。开关定时器只有一个时间点，表示在某个时间点进行开机或者关机操作。阶段定时器有起止时间，表示由起始时间到结束时间执行特定的设置，当结束时间早于起始时间时，表示第二天的结束时间，即定时器可以跨天。
2. 按周期性区分的话，定时器又分为，周期性定时器与非周期性定时器。周期是按星期计算，即每周一、二、三、四、五、六、日执行。非周期性定时器类似系统闹钟，总是在下一次设定时间执行。当设定时间晚于当前时间时，表示当天执行，当设定时间早于当前时间时，表示第二天执行。

特别的，当非周期性阶段定时器的起始时间早于当前时间，且结束时间早于起始时间时，表示从明天（起始时间）到后天（结束时间）执行。

* **定时器冲突的定义：**

同一设备下，同一类型的有效定时器，时间点不能重复（针对开关定时器）或者时间段不能有交叉（针对阶段定时器）。

* **算法的核心思想：**

1. 不论定时器是否为周期性定时器，全部统一放在一个周期（即一个星期）内做冲突检测。（在一个周期内无冲突的话，就不可能冲突了）
2. 将隔天，或者跨天的定时器时间点或者时间段，拆分成明确的周期内当天执行的时间点或者时间段。（在这个设定当中，时间将取消隔天或者跨天的情况，方便比较）比如，针对非周期性阶段定时器，起始时间早于当前时间，结束时间早于起始时间的情况，我们知道，它表示从明天（起始时间）执行到后天（结束时间）。我们将其拆分成两条当天执行的时间段记录，第一条是从起始时间（明天）到23点59分（明天最晚的时间），第二条是从0点0分（后天最早的时间）到结束时间（后天）。为了将其放入周期内检测，必须知道明天是星期几，后天是星期几，则首先应该知道今天是星期几。

# 实现

## 多语言

使用Localization控制多语言。首次启动检测语言，支持应用内切换语言，应用名称支持多语言。

## 屏幕适配

需适配分辨率包含640\*960，640\*1136，750\*1334，1080\*1920。

## 统一简单界面元素

输入框，普通按钮，菜单按钮等具有统一样式与颜色控制的界面元素需要抽象统一编码管理。

## 列表为空时的提示界面

设备列表与定时任务列表界面有为空的情况，但是这两个列表为空时显示的并不是简单的图片文字提示，而是具有交互功能的按钮，而且各不相同。因此只能单独处理。当判断列表行数为0时，懒加载提示界面。

## 列表侧滑菜单

定时任务列表支持侧滑，并出现编辑与删除两个按钮，考虑直接用原生的tableView:editActionsForRowAtIndexPath来实现。

## 搜索设备界面的波纹扩散按钮

波纹出现的频率为1秒，扩散时间为2秒，扩散中途由细变粗，不透明度由高到低。可以考虑通过控制CALayer的borderColor和borderWidth来满足动画。Layer直接加载到按钮中，通过控制按钮的clipsToBounds属性来显示超出部分。

## 温控器遥控面板状态参数展示界面

需要实时更新界面数据，考虑使用KVO来监听设备属性，当模式为换气时，不显示设定温度。背景渐变色使用CAGradientLayer实现垂直方向渐变。外测有4个圈，里面3个颜色单一且固定使用CALayer的borderColor和borderWidth来实现，最外测的圈有旋转动画，10秒为一个周期，颜色随模式改变，并且是环形渐变，考虑直接通过旋转图片实现。

## 温控器遥控面板温度设置界面

温度设置时，左右滑动，从中心点往左右两边扩散，距离越远文字大小越小，不透明度也依次递减。有分页现象，但是分页宽度远小于显示宽度，考虑使用UIScrollView，并将其大小设置为分页大小，通过clipsToBounds属性控制超出部分的显示。超出部分的点击滑动范围通过Hit-test来修正。当模式为换气时，不可用。

## 选择控制器

选择控制器浮动在主控制器之上，点击空白区域时隐藏选择控制器。外围样式非常统一，顶部为标题，底部为取消与确定两个按钮。考虑将控制器的统一样式抽象出来。

## 周期选择

提供周一到周日的选择，并提供全选功能。周一到周日的选择，其实是个组合问题，每一天都有选中与否两种状态，合起来总共有2的7次方即128种组合可能。可以使用位运算，每一位是0还是1即表示当天是否选中。

## 其他细节问题

1. 不要缓存不重用的图片，比如设备列表顶部的大图。
2. 修改状态栏样式时需要对iOS10进行适配。
3. 当模式为换气时，不能设定温度，也不显示设定的温度。
4. 编辑设备的弹窗浮动在主控制器之上，即直接添加在UIWindow上。点击空白区域时，须隐藏弹窗。
5. 不论是手动还是自动更改设备运行状态，都需要清空延时开关。延时开关在APP重启后依然精确计算。所以需要记录延时截止时间，倒计时显示时需临时计算。
6. 定时任务列表为空时的提示，中英文显示格式不一样。中文中键有竖向分割线，英文状态下却没有。
7. 温控器智能设置虽然没有页面跳转，但是更改了标题且隐藏了返回按钮。
8. 起始时间如果晚于结束时间，结束时间显示为次日。
9. 包含计时器的对象在释放前需要手动关闭计时器，否则容易造成内存泄露。
10. 文字大小的渐变效果由adjustsFontSizeToFitWidth来控制，而非使用不同的UIFont，以免造成不必要的内存消耗。
11. 设备列表空白提示位于空白区域中心，而不是整个列表中心。
12. 延时开关图片随设备运行状态改变而改变，倒计时处一样。