

API 使用说明

1. 概述

本文档描述如何通过 API 和 odexservice 服务通讯。

2. 文件列表

- a. oudexservice.1.3.apk -- 支持 Leke VMC 的应用程序。
- b. 接口包 mylibrary.aar -- 和 odexservice 通讯的 JAVA 包，详细见接口说明。

3. 接口说明

1) VmcTransData: vmc 数据通讯类，使用者需要创建该类对象，该接口对象实包含：请求和回调。

2) 请求接口

- 1. openSerialPort(String serialPort): 设置需要打开的串口
- 2. reqButtonBindChannel(CBindKey cBindKey): 设置按键绑定货道策略
- 3. reqTempConfig(TempConfig config): 设置温度配置
- 4. reqDeliver(DeliverData data): 执行出货通知

3) 回调接口

- 1. onVmOnline(boolean online): vmc 在线信息（是否在线）
- 2. onKeyModeDown(): 系统 mode 按键按下（）
- 3. onChannelUp(int cab, int strChannel): 货道按键上报货道（货柜号， 货道号）
- 4. onButtonUp(int buttonNum): 货道按键上报按键号（按键号）
- 5. onStockUp(int[] stockUp): 库存上报（库存数组:0 售空，1 有货）
- 6. onDeliverResp(DeliverData deliverData): 出货结果（出货数据）

4) 数据结构

- 1. CBindKey: 按键货道绑定策略

```
public HashMap<Integer, Integer> keyChannels;
```

<按键号，货道号>（每个货道最多设置 5 个按键，每个按键只能设置一个货道号）。

```
List<String> getVals();
```

获取序列化对象（上层用户不用关心）

```
CBindKey(List<String> vals)
```

通过序列化创建指定对象（上层用户不用关心）

```
CBindKey()
```

创建对象

- 2. TempConfig: 温度配置数据

```
public static final int off = 0;    //关闭
```

```
public static final int cold = 1;   //冷
```

```
public static final int hot = 2;    //热
```

```
public final int address = 1;      //指定地址
```

```
public int temp1;    //第一腔温度(5~55)
```

```
public int temp2;    //第二腔温度(5~55)
```

```
public int temp3;    //第三腔温度(5~55)
```

```
public int temp4;    //第四腔温度(5~55)
```

```
public int mode1;    //第一腔模式
```

```
public int mode2;    //第二腔模式
```

```
public int mode3;    //第三腔模式
```

```
public int mode4;    //第四腔模式
```

```
int[] getVals()
```

获取序列化对象（上层用户不用关心）

```
TempConfig(int[] vals)
```

通过序列化创建指定对象（上层用户不用关心）

```
TempConfig()
```

创建对象

3. DeliverData: 出货数据

```
public int channelNo;    //货道编号(合并后偶数为主货道)
```

```
public int cabinetNo;    //单柜默认为 1
```

```
public int mergeNo;    //合并的货道号(奇数货道)
```

```
public int fastDeliver;    //快速出货 0/1
```

```
public int deliverResult;    //出货结果 0 出货中, 1 出货成功, 2 出货失败,
```

3 机器繁忙, 4 出货超时

```
int[] getVals()
```

获取序列化对象（上层用户不用关心）

```
DeliverData(int[] vals)
```

通过序列化创建指定对象（上层用户不用关心）

```
DeliverData()
```

创建对象