**接口Library包概要设计**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 修改编号 | 日期 | 修改人 | 版本号 | 更新内容 |
| 1 | 2017.10.13 | JQJ | 1.00 | 初稿 |
|  |  |  |  |  |
|  |  |  |  |  |

目录

1 文档概述

1.1 文档目的

2 模块概述

2.1 通信模块

2.2 业务模块

3 模块说明

3.1 通信模块说明

3.2 业务模块说明

1 文档概述

1.1 文档目的

该文档简要描述接口Library包各模块的概要设计思想。

2 模块概述

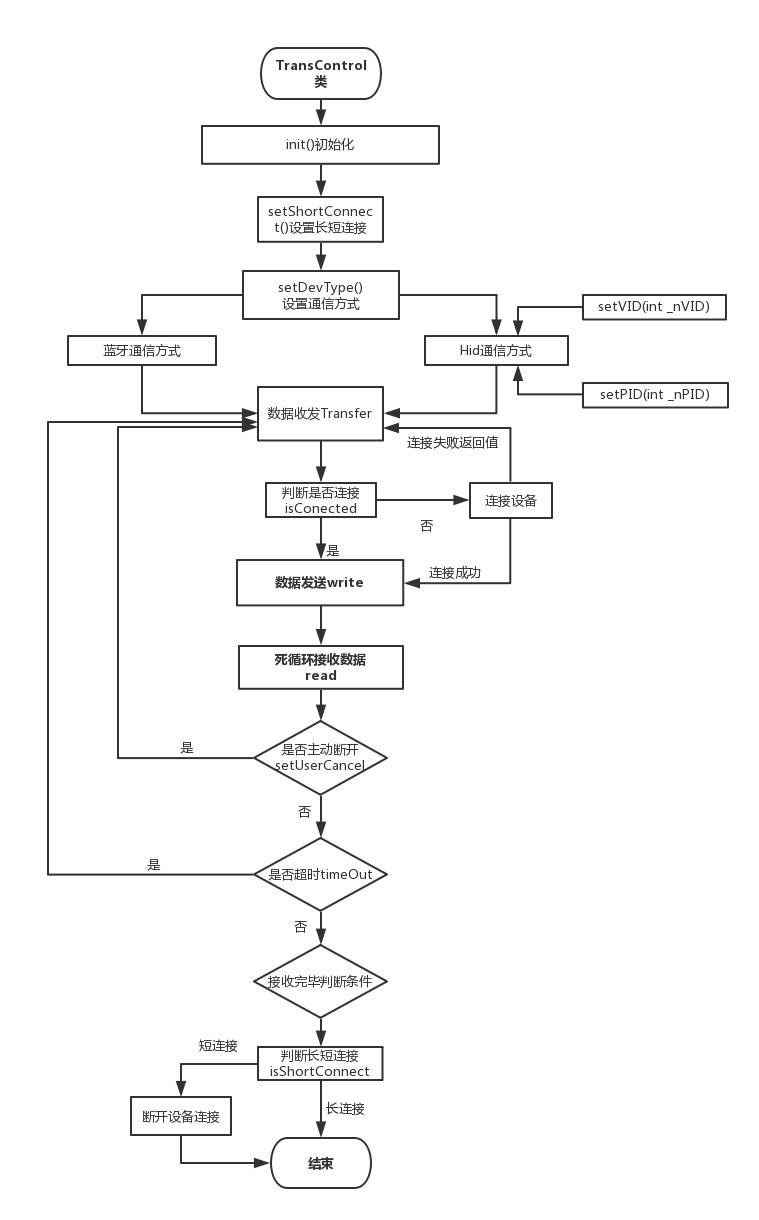
1.通信模块:该模块主要功能为实现客户端与服务端两端数据收发。现支持蓝牙通信与Hid通信

2.业务模块：该模块主要功能为实现调用T5设备端各个业务模块，支持IC卡，ID卡，磁条卡，密码键盘，电子签名以及指纹仪（串口）。

3 模块说明

1. 通信模块说明

通信控制类TransControl，提供了大部分通信相关api。简要流程图如下



1. 通信初始化

TransControl.getInstance().init(Context context, OnMesBackListener listener);

OnMesBackListener 为通信状态回调，内有两个方法

与蓝牙通信相关

1. setBluetoothStatus(String strBluetoothState)

与hid通信相关

1. setUsbStatus(String strUsbState)

strBluetoothState返回值

注册广播接收器: USB状态监听广播接收器用于监听授权、插入、拔出，蓝牙状态监听广播接收器用于监听配对请求、状态改变

1. 设置长短连接方式

TransControl.getInstance().setShortConnect(false);

默认长连接， ture 设置短连接 false 设置长连接

1. 设置当前通信方式

TransControl.getInstance().setDevType(devType);

devType可取值TransControl.DEV\_USBHID或TransControl.DEV\_BLUETOOTH

4.设置pid vid

TransControl.getInstance().setVID(int \_nVID);

TransControl.getInstance().setPID(int \_nPID);

默认vid:0x2B46 pid:0xBB01

1. 连接蓝牙

TransControl.getInstance().connectByBluetooth(remoteMac);

remoteMac为要连接设备的蓝牙mac地址

1. 连接usb hid

TransControl.getInstance().connectByHid();

1. 通信资源释放

TransControl.getInstance().unInit();

注销广播接收器：USB状态监听广播接收器、蓝牙状态监听广播接收器

1. 设置通信超时时间

TransControl.getInstance().setTimeOut(int nTimeOut);

1. 用户强制中断数据收取

TransControl.getInstance().setUserCancel(boolean userCancel)

true 是

False 否

10.自定义数据发送与接收

1.TransControl.getInstance().Transfer(byte[] byReq, int nReqLen, byte[] byRes, int nResLen, EndOfRead condition, int timeout)

byReq 发送数据字节数组

nReqLen 发送数据长度

byRes 接收数据字节数组

nResLen 接收数据长度

condition 数据接收结束条件

Timeout 超时

1. TransControl.getInstance().Transfer(byte[] byReq, int nReqLen)

byReq 发送数据字节数组

nReqLen 发送数据长度

1. 业务模块说明
2. 电子签名

SignatureFunc.getInstance().startSignature(int timeOut);

timeOut为超时时间

SignatureFunc.getInstance().getSignPic(String[] path);

path为获取成功后图片路径

SignatureFunc.getInstance().getSignXml(String[] path);

path为获取成功后轨迹路径

SignatureFunc.getInstance().getSwInfo();

调用后返回值

1. ID卡

String[] perInfo = new String[11];

IDCardFunc.getInstance().getIDCardInfoOnce(String[] perInfo);

1. 密码键盘

KeypadFunc.getInstance().startKeypad(int timeOut)

timeOut为超时时间

1. 磁条卡

MagCardInfo magCardInfo = new MagCardInfo();

MagCardFunc.getInstance().readTrack(magCardInfo);

byte[] track1 = magCardInfo.getTrack1();

byte[] track2 = magCardInfo.getTrack2();

byte[] track3 = magCardInfo.getTrack3();

1. IC卡

ICCardData data = new ICCardData();

ICCardFunc.getInstance().getICCardInfo(data)

ICCardFunc.getInstance().getICData()

data.setCardStyle(1); 设置接触式

data.setCardStyle(2); 设置非接触式

6.指纹仪

FingerFunc.getInstance().readFingerFeature(int timeOut) 读取指纹仪特征值

timeOut为超时时间

FingerFunc.getInstance().registerFingerFeature(int timeOut) 获取模板

timeOut为超时时间

FingerFunc.getInstance().getSwInfo() 获取特征值