

ROBOTPEN 鲁伯特

PPWrite iOS SDK

参考手册

2.3.2 版

目 录

目 录.....	2
简介.....	3
PPWrite.....	3
一. SDK 包说明.....	4
二. SDK 配置.....	4
三. Demo 演示.....	5
四. API.....	6

简介

PPWrite

写写是依托电磁本、智能本两个硬件产品及其附件产品的配合针对教学中的微课、直播活动提供解决方案。目前电磁本产品分为 P1、P7、Elite、Elite Plus 四款产品，P1 款产品为 USB 线连接方式，P7 款产品为蓝牙连接方式，Elite 款和 Elite Plus 款产品都为蓝牙连接方式且支持设备本地离线笔记的存储。书写时，智能笔前端发射器以每秒钟 100-200 个上报点信息，电磁板会把这些点连接并还原原始笔迹。

SDK 根据客户的实际使用需要，分为 4 大功能：

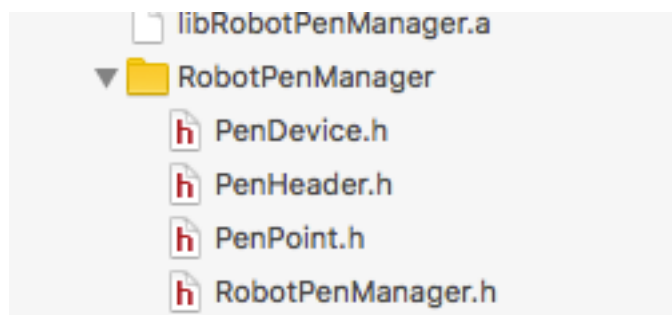
- 笔迹坐标点
- 微课功能
- 直播功能
- 设备交互（蓝牙版的连接、升级、离线笔记的同步等）



一. SDK 包说明

SDK 包开发环境：Xcode, Mac

设备系统要求： iOS9.0 及以上系统



其中包括 libRobotPenManager.a, PenDevice.h, PenHeader.h, PenPoint.h, RobotPenManager.h 五个文件。

PenDevice.h : 设备相关的属性

PenPoint.h : 蓝牙板点的属性

RobotPenManager.h : 笔服务相关的代理以及 manager 相关方法

二. SDK 配置

将上面所述的五个文件导入工程中即可使用。

三. Demo 演示

```
[[RobotPenManager sharePenManager] setPenDelegate:self];
```

将 penmanager 初始化并将 RobotPenDelegate 挂上

```
-(void)blueToothButtonPressed:(UIButton *)sender{
    NSLog(@"%s",__func__);
    if (isConnect == NO) {
        sender.selected = YES;
        // [[RobotPenManager sharePenManager] scanDevice];
        [[RobotPenManager sharePenManager] scanDevice:self];

    } else{
        sender.selected = NO;
        [self.deviceArray removeAllObjects];
        [_tableView reloadData];
        [[RobotPenManager sharePenManager] disconnectDevice];
    }
}
/**
 获取点的信息

```

在搜索设备的按钮方法中 执行 scanDevice 需要断开连接就执行 disconnectDevice 方法

```
-(void)getBufferDevice:(PenDevice *)device{
    [self.deviceArray addObject:device];
    [self.tableView reloadData];
}
```

将发现的设备放到数组中

```

99 -(void)getDeviceState:(DeviceState)State{
100     switch (State) {
101         case DISCONNECTED:
102             NSLog(@"disconnect");
103             isConnect = NO;
104             _blueToothButton.selected = NO;
105             break;
106         case CONNECTED:
107             NSLog(@"CONNECTED");
108             isConnect = YES;
109             self.device = [[RobotPenManager sharePenManager] getConnectDevice];
110             self.deviceName.text = [NSString stringWithFormat:@"%@",[self.device getName]];
111             self.deviceUUID.text = [NSString stringWithFormat:@"%@",self.device.uuID];
112             [[RobotPenManager sharePenManager] stopScanDevice];
113             break;
114         case CONNECTING:
115             NSLog(@"connecting");
116             break;
117         default:
118             break;
119     }
120 }
121 }
122 }
123 }

```

此代理方法是返回不同的 DeviceState，根据不同的 state 来进行状态的切换

```

146 }
147 -(void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath{
148     NSLog(@"%s",__func__);
149     PenDevice *selectItem = [self.deviceArray objectAtIndex:indexPath.row];
150     [[RobotPenManager sharePenManager] connectDevice:selectItem :self];
151 }
152 }
153 }
154 @end
155 }

```

连接设备的代理方法，连接需要的外部设备

四. API

RobotPenDelegate 相关方法

Api	解释	备注
-(void) getDeviceState: (DeviceState) State;	获取设备状态	
-(void) getPointOnfo: (PenPoint *) point;	获取点数据	
-(void) getDeviceVersion: (NSDictionary *) infos;	获取设备信息	
-(void) AutoConnectingDevice: (PenDevice) device;	自动连接连接过的设备	

- (void) OTAUpdateState: (OTAState) state;	OTA 升级状态 (state)	
- (void) SyncState: (SYNCStates) state;	SYNCNoteState	
- (void) getSyncData: (TralisModel *) trails;	获取笔记数据	
- (void) getStorageNum: (int) num;	获取笔记数量	
- (void) getBufferDevice: (PenDevice *) device;	发现设备	

RobotPenManager 相关方法

Api	解释	备注
+ (RobotPenManager *) sharePenManager;	单例初始化	
- (BOOL) checkDeviceConnect;	检查设备是否连接	
- (PenDevice *) getConnectDevice;	获取当前链接的设备	
- (void) connectDevice: (PenDevice *) penDevice : (id<RobotPenDelegate>) delegate;	连接设备	
- (void) scanDevice: (id<RobotPenDelegate>) delegate;	扫描设备	
- (void) disconnectDevice;	断开设备	
- (NSDictionary *) getDeviceVersion;	获取版本号	
- (DeviceType) getConnectDeviceType;	获取设备类型	
- (void) setPenDelegate: (id<RobotPenDelegate>) delegate;	设置回调代理	
- (void) stopScanDevice;	停止扫描	
- (void) setAutoConnet: (int) isOpen;	设置自动连接	
- (void) startOTA: (id<RobotPenDelegate>) delegate;	开始 OTA 升级	

- (void) startSyncNote: (id<RobotPenDelegate>) delegate;	同步笔记 开始	
- (void) stopSyncNote;	同步笔记 停止	