OneNETDemo_iOS 使用指南

一：SDK 静态库和头文件

静态库和头文件是通过整理官方源码编译而得到，如下图 1.1 所示，开发者可以将静态库和头文件拷贝到新建项目或者已有项目中直接使用，在新的项目中使用时需要导入如图 1.2 所示的 libz.1.2.8.tbd，否则无法进行编译。

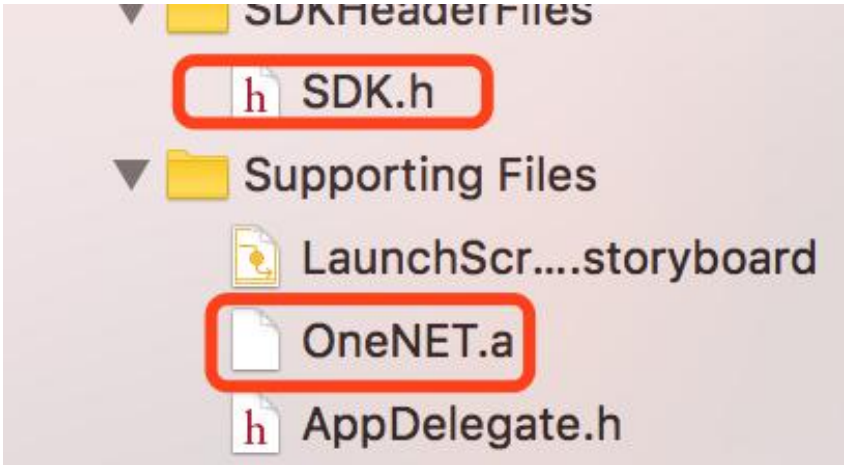

图 1.1

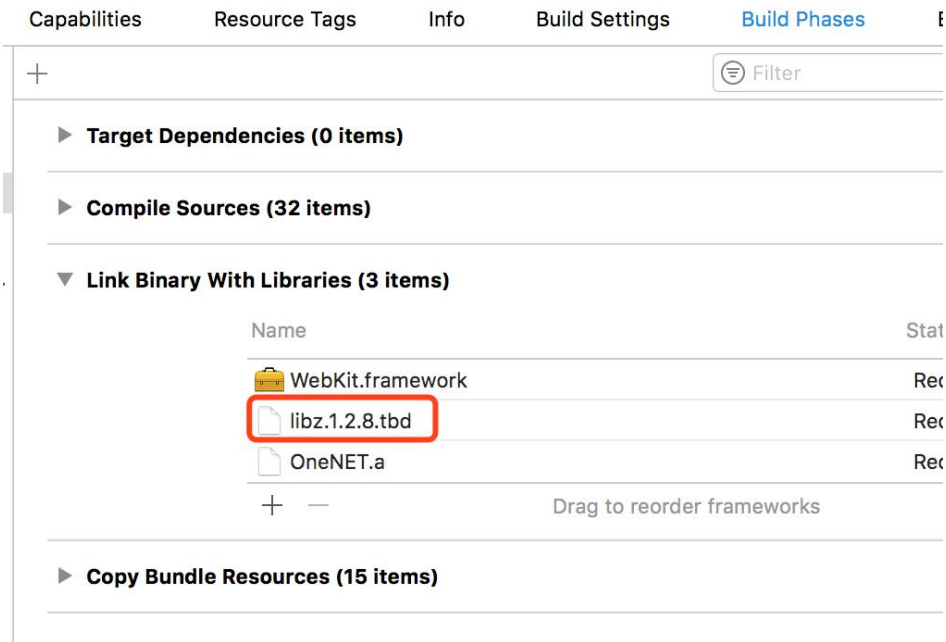

图 1.2

二：获取 OneNET 平台的 MasterAPIKey

由于 OneNET 平台没有开放登录和注册接口，开发者需要自己到 OneNET 官方平台网站注册并登录。登录之后就可以进行产品创建，产品的创建可以参照 OneNET 官方文档进行操作，这里创建产品时使用的协议是 EDP 协议。创建产品之后，点击创建的产品详情，找到产品的 MasterAPIKey 如图 2.1，记住或者拷贝 MasterAPIKey 到 APP 中使用。之后的一切操作都将基于这个 MasterAPIKey，所以这个是很重要的。

图 2.1

三：获取设备列表

调用头文件 SDK.h 中的方法

-(NSDictionary *)getMoreDeviceRequestKey:(NSString *)apikey andPage:(int)page andPerPage:(int)perpage andRequestParam:(NSString *)param;

正确的返回如图 3.1

图 3.1

四：设备的添加、删除和修改

设备的添加和修改可以对更多的设备属性进行操作，开发者可以按照实际的需求增加请求的参数。（注：APP 中限制对默认设备的操作）

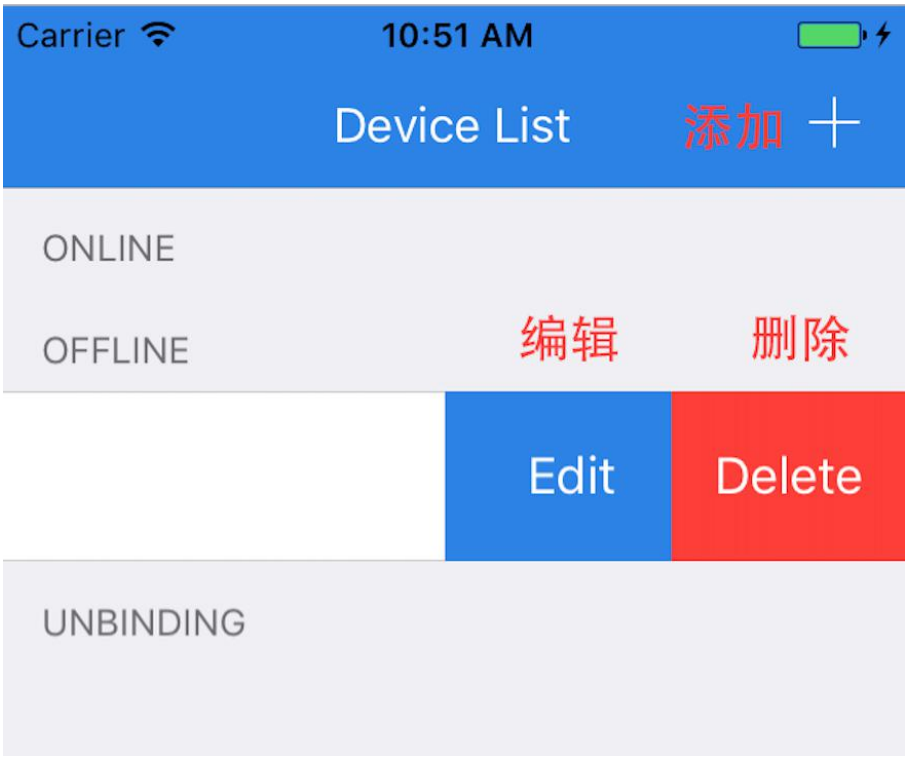

图 4.1

（1） 设备的添加

调用方法

-(NSDictionary *)addDeviceRequestKey:(NSString *)apikey andRequestParam:(NSString *)param;

添加成功返回

```
{
    data =     {
        "device_id" = 4756530;
    };
    errno = 0;
    error = succ;
}
```

图 4.2

（2） 设备的删除

调用方法

-(NSDictionary *)deleteDeviceRequsetKey:(NSString *)apikey andDeviceId:(NSString *)deviceId;

删除成功返回：

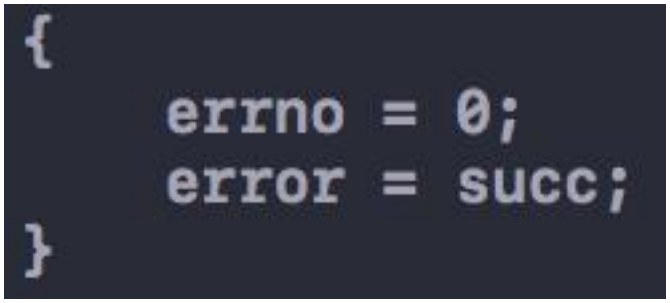

图 4.3

（3） 设备的编辑

调用方法

-(NSDictionary *)editDeviceRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andRequestParam:(NSString *)param;

编辑成功返回：

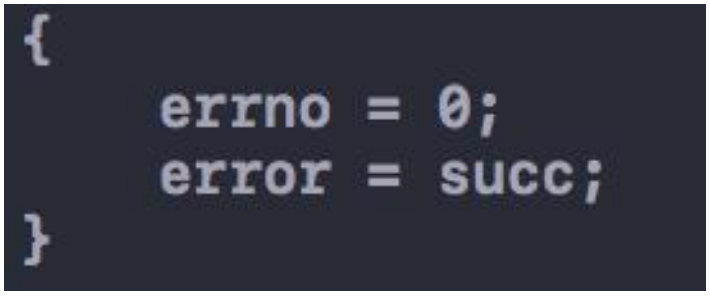

图 4.4

五：获取数据流（读取数据点的值）

（1） 获取所有的数据点（读取所有数据点的值）

调用方法

-(NSDictionary *)readMoreDataPointRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andParam:(NSString *)param;

获取（读取成功）返回


```
{
    data =     (
                {
            "create_time" = "2017-03-07 09:44:42";
            "current_value" = 103;
            id = r;
            "update_at" = "2017-03-08 17:13:55";
            uuid = "6e7f2992-c92d-4ff4-95b7-ef579d89bfc8";
        },
                {
            "create_time" = "2017-03-07 09:44:50";
            "current_value" = 86;
            id = b;
            "update_at" = "2017-03-08 17:13:59";
            uuid = "74d5ec23-7e46-4786-8356-46e56ea98ca6";
        },
                {
            "create_time" = "2017-03-07 09:44:45";
            "current_value" = 117;
            id = g;
            "update_at" = "2017-03-08 17:14:00";
            uuid = "ef81d23f-da13-4b41-a8af-169d25e4e30e";
        },
                {
            "create_time" = "2017-03-07 09:44:51";
            "current_value" = 1;
            id = s;
            "update_at" = "2017-03-08 17:14:06";
            uuid = "d1f0e9d3-2c2a-47e6-af73-44db8607c01b";
        }
    );
    errno = 0;
    error = succ;
}
```


图 5.1

（2） 获取单个数据点的值

调用方法

-(NSDictionary *)readSingleDataPointRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andDataStreamId:(NSString *)streamId;

获取成功返回

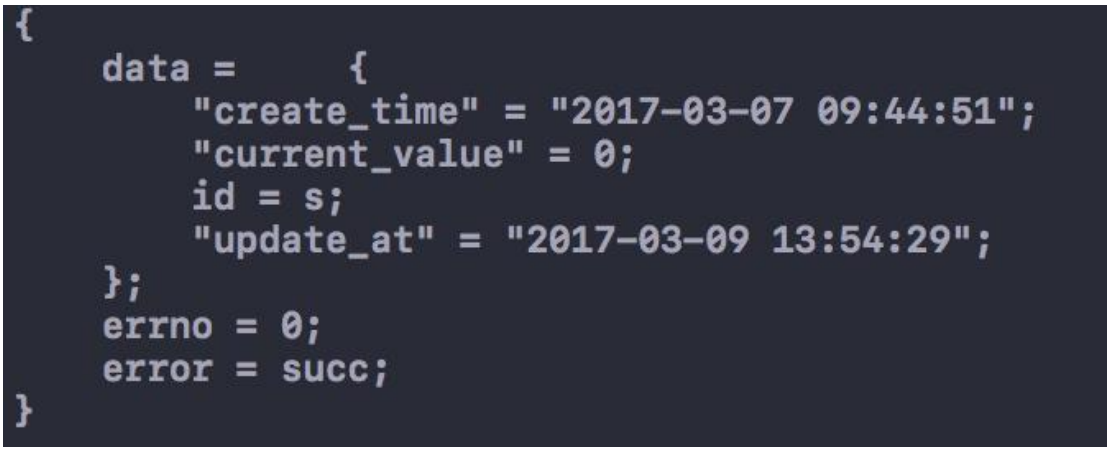

图 5.2

六：上报数据到服务器

调用方法

-(NSDictionary *)reportDataPointRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andParam:(NSString *)param;

上报成功返回

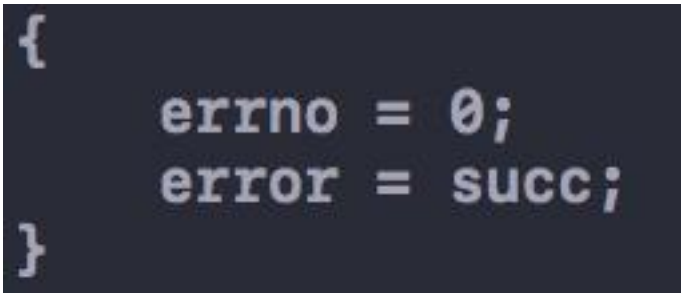

图 6.1

七：发送数据到设备

调用方法

-(NSDictionary *)sentDataReqestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andParam:(NSString *)param;

发送成功返回

图 7.1

八：附加说明

1. 可以对数据流进行操作，Demo APP 中屏蔽了这个操作。

（1） 数据流的创建

假使设备或者 APP 上传一个平台不存在的数据点和值时，平台会自动创建一个数据点并赋予相应的值。APP 可以使用下面的方法调用也实现数据点的创建：

-(NSDictionary *)addDataPointRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andParam:(NSString *)param;

（2） 数据流的编辑

可以对数据流的属性进行编辑，可编辑的属性参照 OneNET 官方网站的说明，调用以下方法可以实现数据点的编辑：

-(NSDictionary *)editDataPointRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andDataStreamId:(NSString *)streamId andParam:(NSString *)param;

（3） 数据流的删除

使用以下方法进行删除操作：

-(NSDictionary *)deleteDataPointRequestKey:(NSString *)apikey andDeviceId:(NSString *)deviceId andDataStreamId:(NSString *)streamId;

2. 发送数据分为两种，其一是上报服务器，其二是发送到真实设备，分别调用不同的方法。在 Demo APP 中使用的逻辑是：发送一个数据时，先调用发送到真实设备的方法（可以使用发送成功时返回的 cmid 查询发送的结果），再调用上报服务器的方法（发送成功时，可以在 OneNET 官方网站的设备管理中的数据展示中动态的查看发送的数据），最后调用方法读取服务器的数据（由于服务器延迟，调用需要延时 0.7s，否则读取到的数据是上一次发送的数据！）。