

串口SDK

GetSerialSDK获取串口SDK配置

```
<!--请求-->
<?xml version='1.0' encoding='utf-8'?>
<sdk guid="##GUID">
  <in method="GetSerialSDK"/>
</sdk>

<!--反馈
##name 串口接口名 {"sensor", "gps"}
##enable 开启串口SDK功能 {"true", "false"}
-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="0dda9f5d2f5243075256923b9a8dcde5">
  <out result="kSuccess" method="GetSerialSDK">
    <config name="##name" enable="##enable"/>
  </out>
</sdk>
```

SetSerialSDK设置串口SDK配置

```
<!--请求
##name 串口接口名 {"sensor", "gps"}
##enable 开启串口SDK功能 {"true", "false"}
-->
<?xml version='1.0' encoding='utf-8'?>
<sdk guid="##GUID">
  <in method="SetSerialSDK">
    <config name="##name" enable="##enable"/>
  </in>
</sdk>

<!--反馈-->
<?xml version="1.0" encoding="utf-8"?>
<sdk guid="0dda9f5d2f5243075256923b9a8dcde5">
  <out result="kSuccess" method="SetSerialSDK"/>
</sdk>
```

串口SDK发送和接收格式

字段	命令包长度 (len)	命令 (cmd)	xml总长度 (total)	xml当前位置 (index)	xml crc32	xml
字节 数	2字节	2字节	4字节	4字节	4字节	n字 节

串口SDK数据每包要求不大于9000字节, 超过9000字节进行分包处理

```
#pragma pack(1)
typedef struct HHeader
{
    huint16 len;      ///< 命令包长度
    huint16 cmd;      ///< 命令值
} HHeader;

typedef struct HSerialSDKAsk
{
    HHeader header;
    huint32 total;    ///< xml总长度
    huint32 index;    ///< xml在总xml的位置，首包为0，下一包就是切包的位置
    huint32 crc32;    ///< 当前包的xml数据的crc32校验值
    hint8   data[1];
} HSerialSDKAsk, HSerialSDKAnswer;
#pragma pack()
```