

# **NOLO VR Android SDK FOR C**

## **接口说明**

北京凌宇智控科技有限公司

2017 年 5 月

## 目录

一、简介.....	1
二、SDK 接口说明.....	1
1. 接口详细说明.....	1
附录：头文件.....	4

## 一、简介

NOLO VR Android SDK 是北京凌宇智控科技有限公司针对其 NOLO CV1 产品对外提供的接口说明，便于 APP 接入方通过该 SDK 获取 NOLO 设备数据。

## 二、SDK 接口说明

### 1. 接口详细说明

NOLO VR Android SDK FOR C 共有 9 个接口，每个接口的名称、原型、功能、参数和返回值如下表所示。

接口名称	详细说明	
SDK 连接 NOLO 设备接口	原型	bool NoloDevice_conn()
	功能	连接 NOLO 设备接口
	参数	无
	返回值	返回连接状态码， true: 连接成功，false: 连接失败
SDK 向 NOLO 设备发送数据接口	原型	bool NoloDevice_sendData(int data[],int len)
	功能	向 NOLO 设备发送数据
	参数	data: 待发送的数据，例如： int[4]: [ 0xAA(帧头第一位), 0x66(帧头第二位), 0x00(左手柄的震动强度，范围在(0x00~0x64)), 0x00(右手柄的震动强度，范围在(0x00~0x64)) ] len:发送数据长度，震动长度为 4
	返回值	true: 发送成功，false: 发送失败

SDK 断开 NOLO 设备接口	原型	bool NoloDevice_finish()
	功能	断开与 NOLO 设备的连接
	参数	无
	返回值	true: 断开成功 false: 断开失败
获取 NOLO 设备版本号接口	原型	int GetVersionByDeviceType(int type)
	功能	该接口获取 NOLO 设备的版本信息
	参数	参数 type 表示设备类型, 0: 表示头盔; 1: 表示左手柄; 2: 表示右手柄; 3: 表示基站
	返回值	设备的版本号 返回值 1: DK2 返回值 2: CV1
获取 NOLO 设备电量接口	原型	int GetElectricityByDeviceType(int type)
	功能	获取 NOLO 设备的电量
	参数	参数 type 表示设备类型, 0: 表示头盔; 1: 表示左手柄; 2: 表示右手柄; 3: 表示基站
	返回值	NOLO 设备电量
获取 NOLO 设备连接状态接口	原型	int GetDeviceTrackingStatus(int type)
	功能	获取 NOLO 设备的连接状态
	参数	参数 type 表示设备类型, 0: 表示头盔; 1: 表示左手柄; 2: 表示右手柄; 3: 表示基站
	返回值	NOLO 设备的连接状态 0: 未连接或遮挡, 1: 正常
获取 NOLO 设备位置与姿态接口	原型	Nolo_Pose GetPoseByDeviceType(int type)
	功能	获取 NOLO 设备的位置与姿态信息
	参数	参数 type 表示设备类型, 0: 表示头盔; 1: 表示左手柄; 2: 表示右手柄; 3: 表示基站
	返回值	NOLO 设备的位置与姿态信息, Nolo_Pose 的属性详见附录

获取 NOLO 设备的反馈接口	原型	Nolo_ControllerStates GetControllerStatesByDeviceType(int type)
	功能	获取 NOLO 设备的反馈信息
	参数	参数 type 表示设备类型， 0：表示头盔；1：表示左手柄；2：表示右手柄； 3：表示基站
	返回值	NOLO 设备的反馈信息，Nolo_ControllerStates 的属性详见附录
获取 NOLO 设备头盔初始位置接口	原型	Nolo_Vector3 GetHmdInitPosition()
	功能	获取头盔标定高度时地面点的位置坐标
	参数	无
	返回值	返回头盔标定高度时地面点的位置坐标

## 附录：头文件

```
#pragma once

#ifndef NOLO_API
#define NOLO_API extern "C"
#endif

#include <windows.h>
#include <math.h>

namespace NOLO
{
    typedef struct Nolo_Vector2
    {
        float x;
        float y;
    }Nolo_Vector2;

    typedef struct Nolo_Vector3
    {
        float x;
        float y;
        float z;
    }Nolo_Vector3;

    typedef struct Nolo_Quaternion
    {
        float x;
        float y;
        float z;
        float w;
    }Nolo_Quaternion;

    typedef struct Nolo_Pose
    {
        Nolo_Vector3 pos;
        Nolo_Quaternion rot;
    }Nolo_Pose;

    /*
        buttons:
        TouchPad = 1 << 0;
        Trigger = 1 << 1;
    */
}
```

```

        Menu = 1 << 2;
        System = 1 << 3;
        Grip = 1 << 4;
    */
typedef struct Nolo_ControllerStates
{
    UINT buttons;
    UINT touches;
    Nolo_Vector2 touchpadAxis;
}Nolo_ControllerStates;

/**
*****
***
    * Function description: After the application gets the usb permission, call this
method to establish a connection
    * Parameter: null
    * Return Value: bool
    * 【应用获取到 usb 读取权限之后，调用此方法与建立连接】
*****
***
    */
NOL_API bool _cdecl NoloDevice_conn();

/**
*****
***
    * Function description: Disconnect the application from the usb device
    * Parameter: null
    * Return Value: bool
    * 【断开应用与 usb 设备的连接】
*****
***
    */
NOL_API bool _cdecl NoloDevice_finish();

/**
*****
***
    * Function description: Get device tracking status
    * Parameter: int
    0:hmd
    1:controller one
    2:controller two

```

```

3:basestation
* Return Value: int
* 【获取设备跟踪状态】
*****
***
*/
NOLO_API int _cdecl GetDeviceTrackingStatus(int type);

/**
*****
***
* Function description: Get the device hardware version
* Parameter: int
0:hmd
1:conroller one
2:controller two
3:basestation
* Return Value: int
* 【获取设备硬件版本号】
*****
***
*/
NOLO_API int _cdecl GetVersionByDeviceType(int type);

/**
*****
***
* Function description: Get NOLO device electricity quantity
* Parameter: int
0:hmd
1:conroller one
2:controller two
3:basestation
* Return Value: int
* 【获取设备电量信息】
*****
***
*/
NOLO_API int _cdecl GetElectricityByDeviceType(int type);

/**
*****
***
* Function description: Get device tracking information

```



---

```

    * Parameter: int
    0:hmd
    1:conroller one
    2:controller two
    3:basestation
    * Return Value: Nolo_Pose
    * 【获取设备定位信息】
    *****
***

    */
    NOLO_API Nolo_Pose _cdecl GetPoseByDeviceType(int type);

    /**
    *****
***

    * Function description: Get the controller status information
    * Parameter: int
    0:hmd
    1:conroller one
    2:controller two
    3:basestation
    * Return Value: Nolo_ControllerStates
    * 【获取手柄状态信息】
    *****
***

    */
    NOLO_API Nolo_ControllerStates _cdecl GetControllerStatesByDeviceType(int
type);

    /**
    *****
***

    * Function description: Get the headset marker when calibrating the position
    * Parameter: NULL
    * Return Value: Nolo_Vector3
    * 【获取头盔标定时位置】
    *****
***

    */
    NOLO_API Nolo_Vector3 _cdecl GetHmdInitPosition();

    /**
    *****
***

```

---

```
* Function description: Send data to NOL0 device
* Parameter: int data[] int len
* Return Value: bool
* 【震动数据接口】
*****
***
*/
NOL0_API bool _cdecl NoloDevice_sendData(int[] data, int len);
}
```