NOLO VR Windows SDK Interfaces Documentation

LYRobotix Co., Ltd

July 2017

Directory

1. Introduction

NOLO VR Windows SDK is the interfaces description which is provided by LYRobotix used for NOLO CV1, It is convenient for the developers to integrate the SDK to get NOLO device data。

2. SDK Interfaces Description

2.1 Interfaces Detail

NOLO VR Windows SDK has 19 interfaces as below:

1) Interface of opening NOLO ZeroMQ client.
2) Interface of closing NOLO ZeroMQ clien.
3) Interface of connection successful notification between client and NOLO server.
4) Interface of disconnection notification between client and NOLO server.
5) Interface of getting all NOLO devices data.
6) Interface of getting data from leftcontroller of NOLO device.
7) Interface of getting data from rightcontroller of NOLO device.
8) Interface of getting data from headset marker of NOLO device.
9) Interface of getting expanded data from NOLO device.
10) Interface of getting NOLO device headset initial position.
11) Interface of getting NOLO device status.
12) Interface of getting NOLO device electricity quantity.
13) Interface of getting NOLO device headset calibration value.
14) Interface of getting NOLO device version.
15) Interface of getting data from controllers of NOLO device.
16) Interface of getting NOLO device position and attitude.
17) Interface of setting vibration data to controllers of NOLO device.

18) Interface of Double-click the menu key or the system key to notify in real time.

19) Interface of NoloRuntime Received data from the NOLO PC software immediately callback notification.

The name，function，functionality，parameter and return value of each interfaces are as follows.

Name	Description	
Interface of opening NOLO ZeroMQ client	Function	Bool open_Nolo_ZeroMQ()
	Functionality	Open NOLO ZeroMQ client
	Parameter	
	Return value	Return opening status: false; true
Interface of closing NOLO ZeroMQ client	Function	void close_Nolo_ZeroMQ()
	Functionality	Close the communication between SDK and NOLO
	Parameter	
	Return value	
Interface of connection successful notification between client and NOLO server	Function	Bool connectSuccess_FunCallBack(funcCallBack func)
	Functionality	Call the registered func function when the connection between client and Nolo_driver_for_windows software server is successful
	Parameter	Parameter func, custom function pointer, typedef void(*funcCallBack)();
	Return value	Returns the function registration status: false; true
Interface of disconnection notification between client and NOLO server	Function	Bool disConnenct_FunCallBack(funcCallBack func)
	Functionality	Call the registered func function when the connection between client and Nolo_driver_for_windows software server is disconnected
	Parameter	Parameter func, custom function pointer, typedef void(*funcCallBack)();

	Return value	Returns the function registration status: false; true
Interface of getting all NOLO devices data	Function	NoloData get_Nolo_NoloData()
	Functionality	Get all data of NOLO devices, such as the data of headset marker, controllers and base station.
	Parameter	
	Return value	Returns the NoloData structure data, see nolo_api for properties
Interface of getting data from leftcontroller of NOLO device	Function	Controller get_Nolo_LeftControllerData()
	Functionality	Get data from leftcontroller of NOLO device
	Parameter	
	Return value	Returns the Controller structure data, see nolo_api for properties
Interface of getting data from rightcontroller of NOLO device	Function	Controller get_Nolo_RightControllerData()
	Functionality	Get data from rightcontroller of NOLO device
	Parameter	
	Return value	Returns the Controller structure data, see nolo_api for properties
Interface of getting data from headset marker of NOLO device	Function	HMD get_Nolo_HMDData()
	Functionality	Get data from headset marker of NOLO device
	Parameter	
	Return value	Returns the HMD structure data, see nolo_api for properties
Interface of getting expanded data from NOLO device	Function	BYTE* get_Nolo_ExpandData()
	Functionality	Get expanded data from NOLO device, such as double-click system button, double-click menu button
	Parameter	
	Return value	Returns the packet address of BYTE data [64]

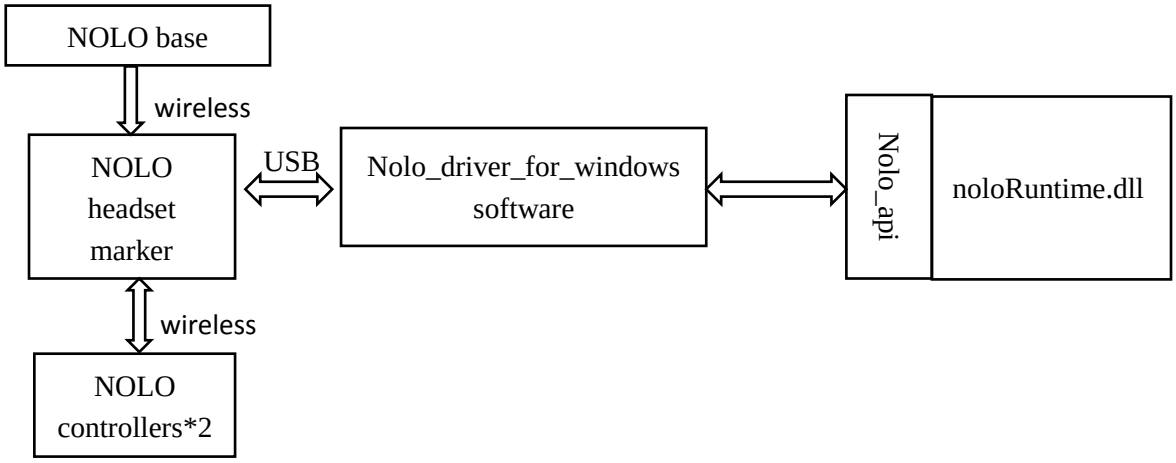
		(data[0]>>0)==1 :Double click Menu (data[0]>>1)==1 :Double click System (data[1]:1 or 0):Double click Menu
Interface of getting NOLO device headset initial position	Function	Vector3 get_Nolo_HMDInitPosition()
	Functionality	Get the initial position of NOLO device
	Parameter	
	Return value	Returns the Vector3 structure data, see nolo_api for properties
Interface of getting NOLO device status	Function	int get_Nolo_StateByDeviceType(NoloDeviceType type)
	Functionality	Get status data from NOLO device
	Parameter	Parameter type is an enumeration type, see nolo_api for properties
	Return value	Returns int type data: 0: blocked; 1: normal
Interface of getting NOLO device electricity quantity	Function	int get_Nolo_Battery(NoloDeviceType deviceType)
	Functionality	Get the data of NOLO device electricity quantity
	Parameter	Parameter deviceType is an enumeration type, see nolo_api for properties
	Return value	Returns int type data: 0-100: the percentage of electricity quantity; 255: turn the power off
Interface of getting NOLO device headset calibration value	Function	int get_Nolo_HMDTwoPointDriftAngle()
	Functionality	Get the calibration value between two points (This interface is valid only for the DK2 protocol of NOLO device)
	Parameter	
	Return value	The calibration value between two points
Interface of getting NOLO device version	Function	int get_Nolo_VersionID(NoloDeviceType devicetype)

	Functionality	Get NOLO device version
	Parameter	Parameter deviceType is an enumeration type, see nolo_api for properties
	Return value	Returns int type data: device version
Interface of getting data from controllers of NOLO device	Function	ControllerStates get_Nolo_ControllerStates(NoloDeviceType type)
	Functionality	Get data from controllers of NOLO device, such as the data of buttons, touch and Axis
	Parameter	Parameter type is an enumeration type, see nolo_api for properties
	Return value	Returns the ControllerStates structure data, see nolo_api for propertie
Interface of getting NOLO device position and attitude	Function	Nolo_Pose get_Nolo_Pose(NoloDeviceType devicetype)
	Functionality	Get NOLO device position and attitude information
	Parameter	Parameter deviceType is an enumeration type, see nolo_api for properties
	Return value	Returns the Nolo_Pose structure data, see nolo_api for propertie
Interface of setting vibration data to controllers of NOLO device	Function	Void set_Nolo_TriggerHapticPulse(NoloDeviceType type,int intensity)
	Functionality	Set vibration data to controllers of NOLO device
	Parameter	Parameter deviceType is an enumeration type, see nolo_api for properties Parameter intensity means vibration intensity, in the range (0~100), larger is more intense
	Return value	
Interface of	Function	Bool expandDataNotify_FuncCallBack(expandMsg_

Double-click the menu key or the system key to notify in real time		FuncCallBack func)
	Functionality	Notification double click menu or system
	Parameter	Parameter func,custom function pointer,see nolo_api for properties
	Return value	Returns the function registration status: false; true
Interface of NoloRuntime Received data from the NOLO PC software immediately callback notification	Function	Bool noloDataNotify_FuncCallBack(noloData_FuncCallBackNotify func)
	Functionality	NoloRuntime receives a data callback notification immediately
	Parameter	Parameter func,custom function pointer,see nolo_api for properties
	Return value	Returns the function registration status: false; true

2.2 Communication Process

As shown in the figure below, the NOLO device consists of a base station, a headset marker and two controllers, the base station and the controllers interact with the headset marker in a wireless communication. The headset marker gather the data and communicate with computer in two-way through the USB protocol. Computer-side Nolo_driver_for_windows software can get the data information of NOLO device, and transfer data to nolo_api in two-way.


NOLO base
wireless
NOLO
headset
marker
USB
Nolo_driver_for_windows
software
Nolo_api
noloRuntime.dll
wireless
NOLO
controllers*2