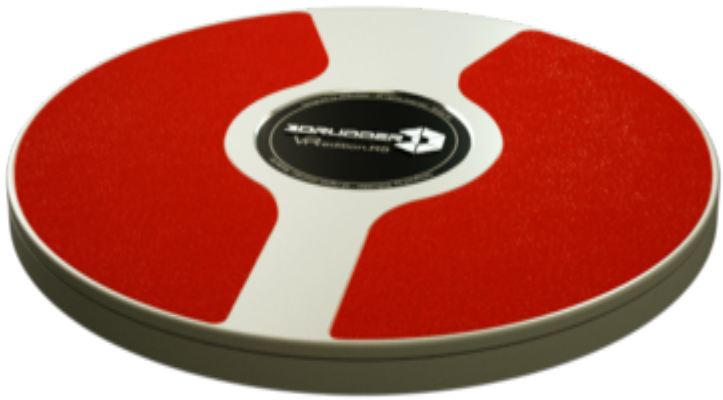
3DRUDDER
Python Module



9/26/2016

Version 0.4 for Windows

This is the pre-release of the 3DRudder Python Module

3DRudder Module

ER ...)

Module Organization

This module is based on two files :

- ns3DRudder.py
- _ns3DRudder.pyd

Module Information

- Available in 32 and 64 bits for Python 3.5.2
- Based on the 3DRudder C++ SDK

Module Usage

```
from  ns3DRudder import *

sdk=GetSDK()
```

SDK Reference

All the SDK is defined in the class `ns3DRudder.CSdk` With this SDK it's possible to manage up to four 3DRudder `_3DRUDDER_SDK_MAX_DEVICE` define the maximum number of port.

Get the sdk version

```
ns3DRudder.CSdk.GetSDKVersion()
```

Return the SDK version of the library. The version is a fixed point unsigned short in hexadecimal: 0x0040 mean version 0.4.

Get the number of connected 3DRudder

```
ns3DRudder.CSdk.GetNumberOfConnectedDevice()
```

Return the number of 3DRudder currently connected to the computer.

Check if a 3DRudder is connected to the port

```
ns3DRudder.CSdk.IsDeviceConnected(nPortNumber)
```

Return true if a 3DRudder is connected to the `nPortNumber` port.

Get the Firmware version of a 3DRudder

```
ns3DRudder.CSdk.GetFirmwareVersion(nPortNumber)
```

Return version number of the firmware of the 3DRudder connected to the `nPortNumber` port. The version is a fixed point unsigned short in hexadecimal: 0x1152 mean version 1.1.5.2

Return 0xFFFF in case of error.

Play a sound on a 3DRudder

```
ns3DRudder.CSdk.PlaySnd(nPortNumber,nFrequency,nDuration)
```

It's possible to play a sound on a 3DRudder connected to the `nPortNumber` port.

`nFrequency` define the frequency of the sound in Hz (440 is a A).

`nDuration` define the duration of the sound in ms.

Get the 3DRudder State

`ns3DRudder.CSdk.Get3DRudderState(nPortNumber,ns3DRudder.State)`

This method fill the class `State` with the value of the 3DRudder connecter to the `nPortNumber` port.

`if this method return 0 this mean the values in State are correct.`

The class `State` is defined like this :

aX is the X Axis (you can use `ns3DRudder.State.GetXAxis()` to read it)
aY is the Y Axis (you can use `ns3DRudder.State.GetYAxis()` to read it)
aZ is the Z Axis (you can use `ns3DRudder.State.GetZAxis()` to read it)
rZ is the Z Rotation (you can use `ns3DRudder.State.GetZRotation()` to read it)
s1 to **s6** are the the six sensor value (you can use `ns3DRudder.State.GetSensor(nIndex) to` read it)
status give the current status of the 3DRudder. (you can use `ns3DRudder.State.GetStatus()` to access to this value)
This status could have the values :

`1:` Puts the 3DRudder on the floor, curved side below, without putting your feet on the device. The user waits for 2 seconds for the 3DRudder to boot up until 3 short beeps are heard.
`2:` The 3DRudder initialize for about 2 seconds. Once done a long beep will be heard from the device. The 3DRudder is then operational.
`3:` Put your first feet on the 3DRudder.
`4:` Put your second Foot on the 3DRudder.
`5:` The user must wait still for half a second for calibration until a last short beep is heard from the device. The 3DRudder is ready to be used.
`6:` The 3DRudder is in use.
`7:` The 3DRudder is in use and is fully operational with all the features enabled.