

USB2CAN-C#-API

Introduce

This is InnoMaker Usb2Can C# API Document, You can download our demo to see how to use api

Doc Version:

Date: 2021-07-16

Version: 1.2.0

Structure

UsbCanMode

```
public enum UsbCanMode
{
    UsbCanModeNormal,
    UsbCanModeLoopback,
    UsbCanModeListenOnly,
}
```

-Summary: UsbCanMode Enum

innomaker_device_bittming

```
public struct innomaker_device_bittming
{
    public UInt32 prop_seg;
    public UInt32 phase_seg1;
    public UInt32 phase_seg2;
    public UInt32 sjw;
    public UInt32 brp;
}
```

-Summary: Device Bittming Structure

Callback

addDeviceDelegate

```
public delegate void AddDeviceNotifyDelegate();  
-Summary: If Device Plug In, it will call the delegate
```

removeDeviceDelegate

```
public delegate void RemoveDeviceNotifyDelegate();  
-Summary: If Device Plug Out, it will call the delegate
```

Functions

GetDllVersion

```
public String GetDllVersion()  
-Summary: Return Current Dll Version  
-Return: Current Dll Version
```

scanInnoMakerDevices

```
public bool scanInnoMakerDevices()  
-Summary: Scann Inno Maker Devices  
-Return: Scan success return true , else return false
```

getInnoMakerDeviceCount

```
public int getInnoMakerDeviceCount()  
-Summary: Get Device Count  
-Return: Device count
```

getInnoMakerDevice

```
public InnoMakerDevice getInnoMakerDevice(int devIndex)  
-Summary: Get Inno Maker device by device index  
-devIndex: Device index
```

-return: Inno Maker Device Instance

openInnoMakerDevice

```
public bool openInnoMakerDevice(InnoMakerDevice device)
```

-Summary: Open Device

-param: device

-return: if open success return true,else return false

closeInnoMakerDevice

```
public bool closeInnoMakerDevice(InnoMakerDevice device)
```

-Summary: Close Device

-param: device

-return: if Close success return true, else return false

asyncGetInnoMakerDeviceBuf

```
public bool asyncGetInnoMakerDeviceBuf(InnoMakerDevice device, Byte[]
```

```
buf, int size, int transferredIn, int timeout)
```

-Summary: Read buffer from device async

-param: device

-param: buf, buffer reads in

-param: size, buffer size

-param: transferredIn, actually buffer length reads

-param: timeout, read buffer timeout

-return: if read device success,return true,else return false

syncGetInnoMakerDeviceBuf

```
public bool syncGetInnoMakerDeviceBuf(InnoMakerDevice device, Byte[]
```

```
buf, int size, int transferredIn, int timeout)
```

-Summary: Read buffer from device sync

-param: device

-param: buf, buffer reads in

-param: size, buffer size

-param: transferredIn, actually buffer length reads

-param: timeout, read buffer timeout

-return: if read device success,return true,else return false

asyncSendInnoMakerDeviceBuf

```
public bool asyncGetInnoMakerDeviceBuf(InnoMakerDevice device, Byte[]
buf, int size, int timeout,int transferredOut)
-Summary: write buffer to device async
-param: device
-param: buf, buffer writes out
-param: size, buffer size
-param: transferredOut, actually buffer length writes
-param: timeout, write buffer timeout
-return: if write device success,return true,else return false
```

syncSendInnoMakerDeviceBuf

```
public bool syncGetInnoMakerDeviceBuf(InnoMakerDevice device, Byte[]
buf, int size, int timeout,int transferredOut)
-Summary: write buffer to device sync
-param: device
-param: buf, buffer writes out
-param: size, buffer size
-param: transferredOut, actually buffer length writes
-param: timeout, write buffer timeout
-return: if write device success,return true,else return false
```

urbResetDevice

```
public bool UrbResetDevice(InnoMakerDevice device)
-Summary: Reset Device
-param: Device Instance
-return: If reset device success return true,else return false
```

urbSetupDevice

```
public bool UrbSetupDevice(InnoMakerDevice device, UsbCanMode
canMode, innomaker_device_bittming bittming)
-Summary: Setup device
-param: Device Instance
-param: canMode, usbCanMode
-param: bittming, usb bittming params
-return: if setup device success return true,else return false
```

