

JSB350 Class Library API Overview

The JSB350CL.DLL provides a managed API for the JSB350 Opto Input modules. It can be used by any programming languages or any applications that supports standard windows Class Libraries. On the driver CD, included is sample source code for various programming languages. They demonstrated the use of each of the functions in a simple straight-forward manner and can be used as a starting point for user applications. The following is just a quick overview for each of the functions. The class library has an .xml file that is used by Visual Studio to support IntelliSense help to make using the functions quick and easy.

1.0 Get Number of Modules

This function returns the number of JSB350 Modules attached to the computers USB port.

```
public static System::Int16 GetNumberOfModules()
```

2.0 Get the serial number of attached JSB350 Module

This function returns a string that is a unique serial number so when multiple JSB350 modules can be accessed.

```
public static System::String GetSerialNumber(System::Int32 nModuleNumber)
```

3.0 Get the version of the class library

This function returns a string that is the version of the class library being used.

```
public static System::String GetClassLibraryVersion(void)
```

4.0 Get version of driver

This function returns a string that is the version of JSB350 USB driver being used.

```
public static System::String GetDriverVersion(System::String SerialNumberOfModule)
```

5.0 Get version of firmware

This function returns a string that is the version of the firmware on the JSB350 module.

```
public static System::String GetFirmwareVersion(System::String SerialNumberOfModule)
```

6.0 Flash the module LED

This function flashes the LED on the module.

```
public System.Void FlashLed(System.String SerialNumberOfModule)
```

7.0 Check if the input is on (voltage above threshold level)

This function returns a value of true if the input on the module is above threshold otherwise returns false.

```
public System::Boolean IsInputOn(System::String SerialNumberOfModule)
```

8.0 Check if the input is off (voltage below threshold level)

This function returns a value of true if the input on the module is off, otherwise returns false.

```
public System::Boolean IsInputOff(System::String SerialNumberOfModule)
```

9.0 Get Input values

This function returns a 32 bit value with the lower 24 bits indicating the input values (1-on, 2-off).

```
public System::Int32 Inputs(System::String SerialNumberOfModule)
```

10.0 Turn Output on

This function turns the output on (driver goes to ground) for the module with the passed serial number.

```
public System::Void OutputOn(System::String SerialNumberOfModule)
```

11.0 Turn Output off

This function turns the output off for the module with the passed serial number.

```
public System::Void OutputOff(System::String SerialNumberOfModule)
```

12.0 Set the Output state

This function sets the states of the Outputs on the module with passed serial number. 1 on, 0 off.

```
public System::Void SetOutput(System::String SerialNumberOfModule, System::Int32)
```