

# Introduction

---

## HID Class DLL

HID Class DLL packages up the calls and links to unmanaged code into one simple to use DLL. The result is just a few simple calls. The DLL even handles unplug/replug events seamlessly.

---

### Introduction

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#)

## SW License Agreement

---

Copyright © 2009-2010 released Microchip Technology Inc. All rights reserved.

Microchip licenses to you the right to use, modify, copy and distribute Software only when embedded on a Microchip microcontroller or digital signal controller that is integrated into your product or third party product (pursuant to the sublicense terms in the accompanying license agreement).

You should refer to the license agreement accompanying this Software for additional information regarding your rights and obligations.

SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY, TITLE, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL MICROCHIP OR ITS LICENSORS BE LIABLE OR OBLIGATED UNDER CONTRACT, NEGLIGENCE, STRICT LIABILITY, CONTRIBUTION, BREACH OF WARRANTY, OR OTHER LEGAL EQUITABLE THEORY ANY DIRECT OR INDIRECT DAMAGES OR EXPENSES INCLUDING BUT NOT LIMITED TO ANY INCIDENTAL, SPECIAL, INDIRECT, PUNITIVE OR CONSEQUENTIAL DAMAGES, LOST PROFITS OR LOST DATA, COST OF PROCUREMENT OF SUBSTITUTE GOODS, TECHNOLOGY, SERVICES, OR ANY CLAIMS BY THIRD PARTIES

(INCLUDING BUT NOT LIMITED TO ANY DEFENSE  
THEREOF), OR OTHER SIMILAR COSTS.

---

## [SW License Agreement](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# Release Notes

---

## HID Class DLL

Version 1.10, 15 April 2010

This version simplifies the naming scheme, though the old names, though deprecated, will continue to be supported. Also it adds an overloaded HIDClassInit that allows specifying the timeout duration. The original defaults to 1000mS.

---

## [Release Notes](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## Running the Demos

---

See "[↗ How to run the Custom HID demos](#)" for running the Demos. This DLL is specifically used in the "Generic HID - HID DLL - PC Software" application.

---

### [Running the Demos](#)

---


Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# HIDClass Namespace

This is namespace HIDClass.

## Classes

	Name	Description
	<a href="#">MCHPHIDClass</a>	This is class HIDClass::MCHPHIDClass.

Symbol Reference > [HIDClass Namespace](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# MCHPHIDClass Class

---

**C**

```
ref class MCHPHIDClass;
```

## Description

---

This is class HIDClass::MCHPHIDClass.

## Class Hierarchy

---

HIDClass::MCHPHIDClass

Symbol Reference > [HIDClass Namespace](#) > [MCHPHIDClass Class](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# MCHPHIDClass Methods

---

Symbol Reference > [HIDClass Namespace](#) > [MCHPHIDClass Class](#) > [MCHPHIDClass Methods](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)



# Files

---

## Files

---

Name	Description
<a href="#">HID class.h</a>	The HID Class DLL provides a simple interface to HID Class Devices.

---

## Symbol Reference > [Files](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# HID class.h

---

## HID Class DLL

The firmware on the corresponding device must have a generic report descriptor. See USB Device HID Custom Demos Firmware project. This hides all the low level details of bridging to unmanaged functions leaving four simple calls.

## Namespaces

---

Name	Description
<a href="#">HIDClass</a>	This is namespace HIDClass.

---

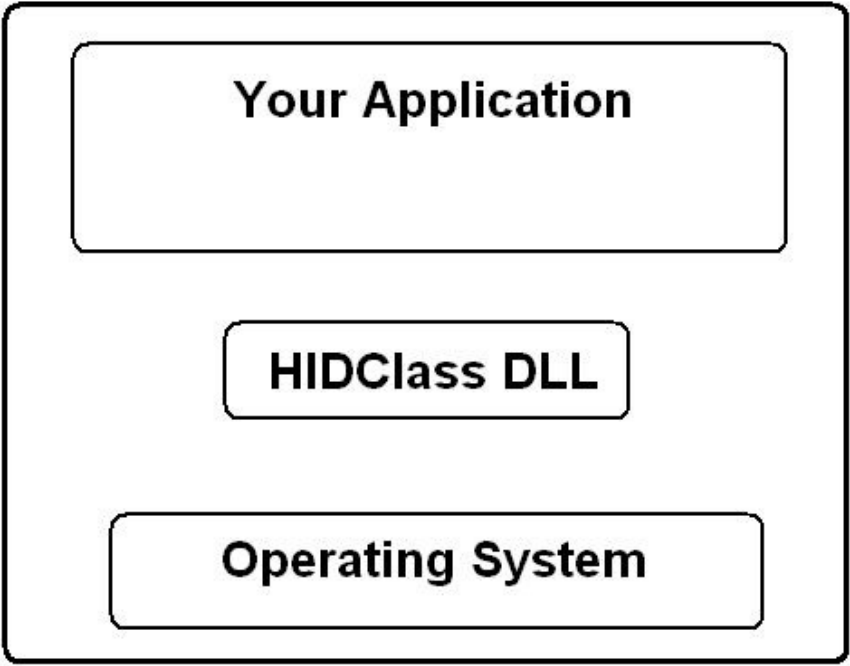
Symbol Reference > [Files](#) > [HID class.h](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# Using the Library

## Topics

Name	Description
<a href="#">Library Architecture</a>	<p>HID Class DLL provides a interface to the HID class device.</p>  <p>The diagram illustrates the architecture of the HID Class DLL. It consists of a large outer box containing three smaller boxes stacked vertically. The top box is labeled 'Your Application', the middle box is labeled 'HIDClass DLL', and the bottom box is labeled 'Operating System'. This indicates that the application interacts with the HIDClass DLL, which in turn interacts with the operating system.</p> <p>The DLL encapsulates the bridge to unmanaged code and interfacing to the operating system from your program.</p>
<a href="#">How the Library Works</a>	<p>This DLL assumes the device we are attaching to enumerates as a generic HID device. That is the data is passed as an array of bytes (usually 64). See the firmware in the Microchip Applications Library, "USB Device - HID - Custom Demos".</p> <p>The DLL encapsulates the windows OS calls necessary</p>

to search through the list of connected devices and find the device with the specified Vendor and Product ID. Then once it's found it opens a pipeline to the device. It also automatically tries to reopen the device in case it has been disconnected and reconnected. This makes recovering... [more](#)

## [Getting Started](#)

The DLL is simple and easy to use. Associate the DLL with the project, declare the name space, initialize the DLL and then read and write as necessary:  
First Associate the DLL with the application. In Visual C++ 2008 Express Edition, select Project | Properties (Alt-F7). "Framework and References" tab. Add New Reference button, Browse Tab. Then navigate to the DLL location and add it to the project.  
Next, Add "using namespace [HIDClass](#);" to the namespace section at the top of your code.  
Then you only need three calls to use the library. HIDClassInit, HIDWriteReport and HIDReadReport.  
Other calls... [more](#)

---

## [Using the Library](#)

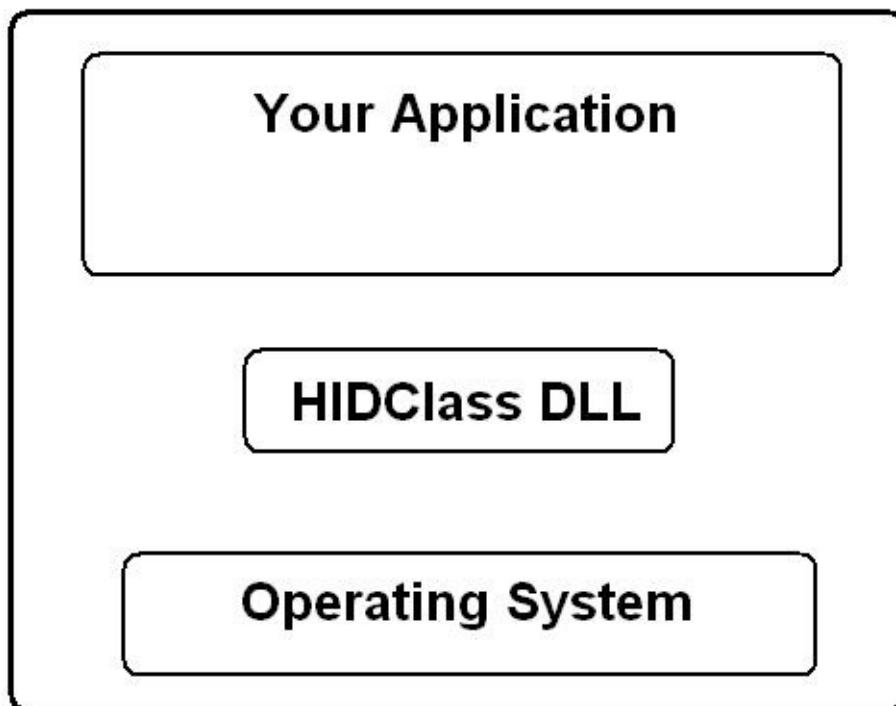
Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## Library Architecture

---

HID Class DLL provides a interface to the HID class device.



The DLL encapsulates the bridge to unmanaged code and interfacing to the operating system from your program.

---

[Using the Library](#) > [Library Architecture](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## How the Library Works

---

This DLL assumes the device we are attaching to enumerates as a generic HID device. That is the data is passed as an array of bytes (usually 64). See the firmware in the Microchip Applications Library, "USB Device - HID - Custom Demos".

The DLL encapsulates the windows OS calls necessary to search though the list of connected devices and find the device with the specified Vendor and Product ID. Then once it's found it opens a pipeline to the device. It also automatically tries to reopen the device in case it has been disconnected and reconnected. This makes recovering from a disconnect/reconnect event seamless to the application.

The compilation setting must be /clr, /clr:pure or /clr:safe. This dll is a .NET assembly, so it can't be used with a compilation setting that produces pure unmanaged code.

---

### [Using the Library](#) > [How the Library Works](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## Getting Started

The DLL is simple and easy to use. Associate the DLL with the project, declare the name space, initialize the DLL and then read and write as necessary:

First Associate the DLL with the application. In Visual C++ 2008 Express Edition, select Project | Properties (Alt-F7). "Framework and References" tab. Add New Reference button, Browse Tab. Then navigate to the DLL location and add it to the project.

Next, Add "using namespace [HIDClass](#);" to the namespace section at the top of your code.

Then you only need three calls to use the library. HIDClassInit, HIDWriteReport and HIDReadReport. Other calls are provided but are not required to communicate with device.

HID Class Init configures the DLL with the needed information that does not change: Vendor ID, Product ID, Buffer Size and Timeout (optional). This can be called from your application's constructor.

Both Read and Write functions first check to see if the device has been opened before attempting to communication with the device. If it not it opens a channel and then completes the communication. This allows seamless recovery if a device has been disconnected then reconnected.

The read and write functions both return a boolean indicating if the transfer was successful or not. Suggest testing these to verify the transfer and recover gracefully if not.

```
HIDClassInit (0x04D8, 0x0032, 64);
```

```
OutBuffer[0] = SOME_COMMAND;
OutBuffer[1] = SOME_COMMAND_PARAMETER_1;
OutBuffer[2] = SOME_COMMAND_PARAMETER_2;
OutBuffer[3] = SOME_COMMAND_PARAMETER_3;
if (MCHPHIDCLASS::HIDWriteReport (OutBuffer,
{
    if (MCHPHIDCLASS::HIDReadReport (InBuffer
    {
        // process InBuffer
    }
    else
    {
        // appropriate reaction to not receive
    }
}
else
{
    MessageBox ("Is the device connected?");
}
```

---

## [Using the Library](#) > [Getting Started](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)



## Library API

---

This section describes the API of the library.

### Topics

---

Name	Description
<a href="#">HID Class</a>	<p>HID Class DLL simplifies the interface to a HID class USB device. It assumes the firmware is configured to send an array of bytes. See the project in Microchip Solutions\USB Device - Custom Demos\Generic HID - Firmware.</p> <p>While there are quite a number of functions, only three are necessary: HIDClassInit, HIDWriteReport, HIDReadReport. The rest are optional or aliases for backwards compatibility.</p>

---

### [Library API](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## HID Class

HID Class DLL simplifies the interface to a HID class USB device. It assumes the firmware is configured to send an array of bytes. See the project in Microchip Solutions\USB Device - Custom Demos\Generic HID - Firmware.

While there are quite a number of functions, only three are necessary: `HIDClassInit`, `HIDWriteReport`, `HIDReadReport`. The rest are optional or aliases for backwards compatibility.

### Public Methods

	Name	Description
◆	<a href="#"><u>HIDClassInit</u></a>	This is the overview for the <code>HIDClassInit</code> method overload.
◆	<a href="#"><u>HIDWriteReport</u></a>	Transfers data to the end device.
◆	<a href="#"><u>HIDReadReport</u></a>	Transfers a data from the end device.
◆	<a href="#"><u>HIDClassVersion</u></a>	Returns the version of the DLL
◆	<a href="#"><u>HIDIsConnected</u></a>	OS inquiry to find out of the previously specified VID & PID is currently attached.
◆	<a href="#"><u>HIDCloseReport</u></a>	Closes the communication channel.
◆	<a href="#"><u>USBHIDClassInit</u></a>	Initializes the DLL with the VID, PID and buffersize of the device.
◆	<a href="#"><u>USBHIDIsConnected</u></a>	OS inquiry to find out of the previously specified VID & PID is currently

		attached.
	<a href="#"><u>USBHIDReadReport</u></a>	Transfers data from the end device.
	<a href="#"><u>USBHIDWriteReport</u></a>	Transfers data to the end device.

---



## [Library API](#) > [HID Class](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## HIDClassInit Method

	Name	Description
	<a href="#"><u>MCHPHIDClass::HIDClassInit (unsigned int, unsigned int, unsigned int)</u></a>	Initializes the DLL with the VID, PID and buffersize of the device.
	<a href="#"><u>MCHPHIDClass::HIDClassInit (unsigned int, unsigned int, unsigned int, unsigned int)</u></a>	Initializes the DLL with the VID, PID, buffersize and timeout of the device.

[Library API](#) > [HID Class](#) > [HIDClassInit Method](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::HIDClassInit Method (unsigned int, unsigned int, unsigned int)

```
C
static void HIDClassInit(
    unsigned int VendorID,
    unsigned int ProductID,
    unsigned int BuffSize
);
```

### Description

Initializes the DLL with the VID, PID, and Buffersize of the device. This function should be called before attempting to communicate with the device.

### Preconditions

None

### Parameters

Parameters	Description
VendorID	16 bit unsigned integer.
ProductID	16 bit unsigned integer
BufferSize	16 bit unsigned integer. Usually 64

### Returns

Nothing

### Remarks

This function, or an overloaded variant, should be called prior to attempting to communicate with the device.

## Example

---

[Copy Code](#)

```
MCHPHIDCLASS::HIDClassInit(0x4D8, 0x0F00, 64);
```

---

[Library API](#) > [HID Class](#) > [HIDClassInit Method](#) > [MCHPHIDClass::HIDClassInit Method \(unsigned int, unsigned int, unsigned int\)](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::HIDClassInit Method (unsigned int, unsigned int, unsigned int, unsigned int)

C

```
static void HIDClassInit(  
    unsigned int VendorID,  
    unsigned int ProductID,  
    unsigned int BuffSize,  
    unsigned int Timeout  
);
```

### Description

Initializes the DLL with the VID, PID, and Buffersize of the device. This function should be called before attempting to communicate with the device. The reads and write use overlapped transfers. This version allows specifying how long to wait before timing out. The default is 1000mS.

### Preconditions

None

### Parameters

Parameters	Description
VendorID	16 bit unsigned integer.
ProductID	16 bit unsigned integer
BufferSize	16 bit unsigned integer. Usually 64
TimeOut	16 bit unsigned integer. Time in mS to wait for a read or write to timeout before

|| returning.

## Returns

---

Nothing

## Remarks

---

This function, or an overloaded variant, should be called prior to attempting to communicate with the device.

## Example

---

```
|| Copy Code  
MCHPHIDCLASS::HIDClassInit(0x4D8, 0x0F00, 64, 250);
```

---

[Library API](#) > [HID Class](#) > [HIDClassInit Method](#) > [MCHPHIDClass::HIDClassInit Method \(unsigned int, unsigned int, unsigned int, unsigned int\)](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)



## MCHPHIDClass::HIDWriteReport Method

```
C
static bool HIDWriteReport(
    unsigned char buffer[],
    unsigned int nBytes
);
```

### Description

Transfers a buffer to the end device. Returns true if successful, false if the transfer fails.

### Preconditions

The DLL should be initialized via a call to [HIDClassInit](#) or variant.

### Parameters

Parameters	Description
buffer	up to 64 byte array
nBytes	unsigned int. Number of valid bytes in the buffer.

### Returns

true if the transfer is successful, false if not. A failure to transfer usually means the device has been disconnected.

### Remarks

The full buffer size is always transferred, regardless of how many are valid

## Example

---

[Copy Code](#)

```
ToSendBuffer [0] = 'H';  
ToSendBuffer [1] = 'i';  
ToSendBuffer [2] = ' ';  
ToSendBuffer [3] = 'M';  
ToSendBuffer [4] = 'o';  
ToSendBuffer [5] = 'm';  
ToSendBuffer [6] = '!';  
MCHPHIDCLASS::HIDWriteReport (ToSendBuffer, 7);
```

---

[Library API](#) > [HID Class](#) > [MCHPHIDClass::HIDWriteReport Method](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::HIDReadReport Method

```
C
static bool HIDReadReport(
    unsigned char buffer[]
);
```

### Description

Transfers a buffer from the end device. Returns true if successful, false if the transfer fails.

### Preconditions

The DLL should be initialized via a call to [HIDClassInit](#).

### Parameters

Parameters	Description
buffer	A byte array up to 64 bytes long.

### Returns

true if the transfer is successful, false if not. A transfer failure may mean the device is no longer connected or it could just mean the end device didn't send any data before timeout.

### Remarks

### Example

```
Copy Code
if (MCHPHIDCLASS::HIDReadReport (ReadBuffer))
{
```

```
// Process buffer if arrived
switch (ReadBuffer[0])
{
}
}
else
{
    // do other stuff if no data
}
}
```

---

[Library API](#) > [HID Class](#) > [MCHPHIDClass::HIDReadReport Method](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::HIDClassVersion Method

**C**

```
static unsigned int HIDClassVersion();
```

### Description

Reports back the DLL, Major, Minor and Dot release numbers of the bill.

### Preconditions

None

### Returns

unsigned int representing the version of the DLL. Formatted as three 8 bit unsigned integers, major release.minor release.dot release. For example: Version 1.23.45 would be encoded as 0x00011739

### Example

[Copy Code](#)

```
Version = MCHPHIDCLASS::HIDClassVersion ()
Major = (Version & 0x00FF0000) >> 16;
Minor = (Version & 0x0000FF00) >> 8;
Dot    = Version & 0x000000FF;
```

[Library API](#) > [HID Class](#) > [MCHPHIDClass::HIDClassVersion Method](#)

## MCHPHIDClass::HIDIsConnected Method

```
C
static bool HIDIsConnected();
```

### Description

Polls the OS to find out if the previously specified VID & PID is currently attached. This is an OS inquiry only. No bus traffic is generated.

### Preconditions

None

### Returns

Nothing

### Example

```
MCHPHIDCLASS::HIDIsConnected ()
```

[Copy Code](#)

[Library API](#) > [HID Class](#) > [MCHPHIDClass::HIDIsConnected Method](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::HIDCloseReport Method

```
C
```

```
static void HIDCloseReport();
```

### Description

Closes the communication channel. The next attempt to read from or write to the device will re-establish communications

### Preconditions

The DLL should be initialized via a call to [HIDClassInit](#) or variant.

### Returns

True if the device is connected, False if not.

### Example

[Copy Code](#)

```
if (MCHPHIDCLASS::HIDIsConnected ())  
{  
    MCHPHIDClass::HIDCloseReport ();  
}
```

[Library API](#) > [HID Class](#) > [MCHPHIDClass::HIDCloseReport Method](#)

## MCHPHIDClass::USBHIDClassInit Method

```
C
static void USBHIDClassInit(
    unsigned int VendorID,
    unsigned int ProductID,
    unsigned int BuffSize
);
```

### Description

Initializes the DLL with the VID, PID, and Buffersize of the device. This function should be called before attempting to communicate with the device.

### Preconditions

None

### Parameters

Parameters	Description
VendorID	16 bit unsigned integer.
ProductID	16 bit unsigned integer
BufferSize	16 bit unsigned integer. Usually 64

### Returns

Nothing

### Remarks



This function, or an overloaded variant, should be called prior to attempting to communicate with the device.

## Example

---

[Copy Code](#)

```
MCHPHIDCLASS::USBHIDClassInit(0x4D8, 0x0F00, 64);
```

---

[Library API](#) > [HID Class](#) > [MCHPHIDClass::USBHIDClassInit Method](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::USBHIDIsConnected Method

```
C
```

```
static bool USBHIDIsConnected();
```

### Description

Polls the OS to find out if the previously specified VID & PID is currently attached. This is an OS inquiry only. No bus traffic is generated.

### Preconditions

None

### Returns

Nothing

### Example

[Copy Code](#)

```
MCHPHIDCLASS::USBHIDIsConnected ()
```

[Library API](#) > [HID Class](#) > [MCHPHIDClass::USBHIDIsConnected Method](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::USBHIDReadReport Method

```
C
static bool USBHIDReadReport(
    unsigned char buffer[]
);
```

### Description

Transfers a buffer from the end device. Returns true if successful, false if the transfer fails.

### Preconditions

The DLL should be initialized via a call to [HIDClassInit](#) or variant.

### Parameters

Parameters	Description
buffer	A byte array up to 64 bytes long.

### Returns

true if the transfer is successful, false if not. A transfer failure may mean the device is no longer connected or it could just mean the end device didn't send any data before timeout.

### Remarks

### Example

```
Copy Code
if (MCHPHIDCLASS::USBHIDReadReport (ReadBuffer))
{
```

```
// Process buffer if arrived
switch (ReadBuffer[0])
{
}
}
else
{
    // do other stuff if no data
}
}
```

---

[Library API](#) > [HID Class](#) > [MCHPHIDClass::USBHIDReadReport Method](#)

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## MCHPHIDClass::USBHIDWriteReport Method

```
C
static bool USBHIDWriteReport(
    unsigned char buffer[],
    unsigned int nBytes
);
```

### Description

Transfers the buffer to the end device. Returns true if successful, false if the transfer fails.

### Preconditions

The DLL should be initialized via a call to [HIDClassInit](#) or variant.

### Parameters

Parameters	Description
buffer	up to 64 byte array
nBytes	unsigned int. Number of valid bytes in the buffer.

### Returns

true if the transfer is successful, false if not. A failure to transfer usually means the device has been disconnected.

### Remarks

The full buffer size is always transferred, regardless of how many are valid

## Example

---

[Copy Code](#)

```
ToSendBuffer [0] = 'H';  
ToSendBuffer [1] = 'i';  
ToSendBuffer [2] = ' ';  
ToSendBuffer [3] = 'M';  
ToSendBuffer [4] = 'o';  
ToSendBuffer [5] = 'm';  
ToSendBuffer [6] = '!';  
MCHPHIDCLASS::USBHIDWriteReport (ToSendBuffer, 7);
```

---

[Library API](#) > [HID Class](#) > [MCHPHIDClass::USBHIDWriteReport Method](#)

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

# Library Migration

---

This release is 100% backwards compatible.. no migration necessary

---

## Library Migration

---

Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## Contents

---

Introduction

SW License Agreement

Release Notes

Running the Demos

*Symbol Reference*

### **HIDClass Namespace**

#### **MCHPHIDClass Class**

MCHPHIDClass Methods

### **Files**

HID class.h

### **Using the Library**

Library Architecture

How the Library Works

Getting Started

### **Library API**

#### **HID Class**

##### **HIDClassInit Method**

MCHPHIDClass::HIDClassInit Method (unsigned int, unsigned int)

MCHPHIDClass::HIDClassInit Method (unsigned int, unsigned int)

MCHPHIDClass::HIDWriteReport Method

MCHPHIDClass::HIDReadReport Method

MCHPHIDClass::HIDClassVersion Method

MCHPHIDClass::HIDIsConnected Method

MCHPHIDClass::HIDCloseReport Method

MCHPHIDClass::USBHIDClassInit Method

MCHPHIDClass::USBHIDIsConnected Method

MCHPHIDClass::USBHIDReadReport Method

MCHPHIDClass::USBHIDWriteReport Method

Library Migration

---



Microchip HID Class DLL 1.10 - [15 April 2010]  
Copyright © 2009, 2010 Microchip Technology, Inc. All rights reserved.

[Contents](#) | [Index](#) | [Home](#)

## Index

[F](#) | [G](#) | [H](#) | [I](#) | [L](#) | [M](#) | [R](#) | [S](#) | [U](#)

### F

[Files](#)

### G

[Getting Started](#)

### H

[HID Class](#)

[HID class.h](#)

[HIDClass namespace](#)

[HIDClassInit method](#)

[How the Library Works](#)

### I

[Introduction](#)

### L

[Library API](#)

[Library Architecture](#)

[Library Migration](#)

### M

[MCHPHIDClass class](#)

[about MCHPHIDClass class](#)

[MCHPHIDClass methods](#)

[MCHPHIDClass::HIDClassInit method](#)

[MCHPHIDClass::HIDClassInit method](#)

[MCHPHIDClass::HIDClassVersion method](#)

[MCHPHIDClass::HIDCloseReport method](#)

[MCHPHIDClass::HIDIsConnected method](#)

[MCHPHIDClass::HIDReadReport method](#)

[MCHPHIDClass::HIDWriteReport method](#)

[MCHPHIDClass::USBHIDClassInit method](#)

[MCHPHIDClass::USBHIDIsConnected method](#)

[MCHPHIDClass::USBHIDReadReport method](#)

[MCHPHIDClass::USBHIDWriteReport method](#)

### R

[Release Notes](#)

[Running the Demos](#)

### S

[SW License Agreement](#)

### U

[Using the Library](#)

[Contents](#) | [Index](#) | [Home](#)