FLYFISH
TECHNOLOGIES

Document: **hiDBLUE - APIs Software Interface Reference**, .NET

Audience:	**Partners, Designers, Developers - APIs Integrators**
Confidentiality Level:	**Public**
Document Owner:	Ivan Zilic
Version:	1.2
Date:	January 16, 2014
Number of pages:	8
File name:	Z:\Dev\Doc\hiDBLUE\hiDBLUE-APIs_SIR_NET.docx

Changes log:

Version	*Date*	*Author*	*Reason for update*
1.0	Apr 26, 2013	Peter Kelpec	Initial document
1.1	May 16, 2013	Ivan Zilic	Review, ownership change, status set to PRELIMINARY
1.2	Jan 16, 2014	Martina Krpan	Fixed codename text

Table of Contents

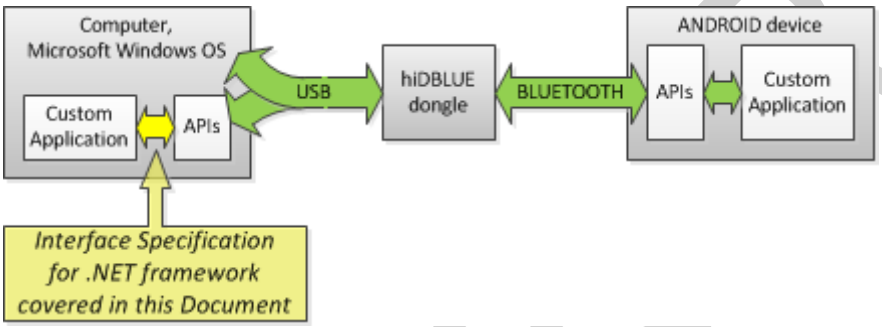
1. Scope

This is a reference document intended for Developers implementing their applications using hiDBLUE APIs, on Microsoft .NET framework platform.

Additionally, this document can give proper hiDBLUE features overview to Products Designers, during their products initial brainstorming and design phases.

2. General Overview

The APIs library is middleware between Custom Application(s) and Microsoft Windows native USB HID driver. A block scheme below points to the Interface under scope here:



APIs library wraps all hiDBLUE-specific communication details. As a result, it is sufficient to call only API functions covered in this document for mastering complete set of custom hiDBLUE features.

Be aware that this APIs library is not applicable for native USB Keyboard and Mouse functionalities. It is needed only when you have to send some custom data between Microsoft Windows Computer and Android Device.

3. Deployed Environment and Prerequisites

The APIs are packed in a single binary dll file, Assembly name is hiDBLUE. This dll is Digitally Signed.

The following table summarizes requirements:

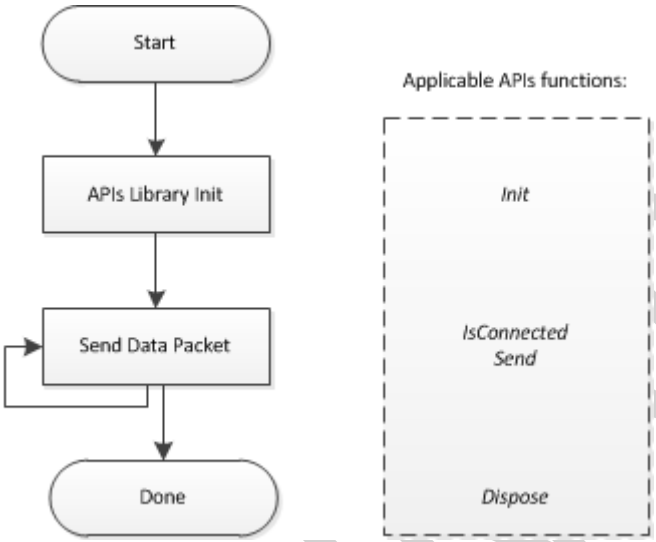
Parameter	Value
.NET Framework version	2.0 or later
Additional References	None

4. APIs Usage

Make sure that single instance of APIs is simultaneously run in all your applications.

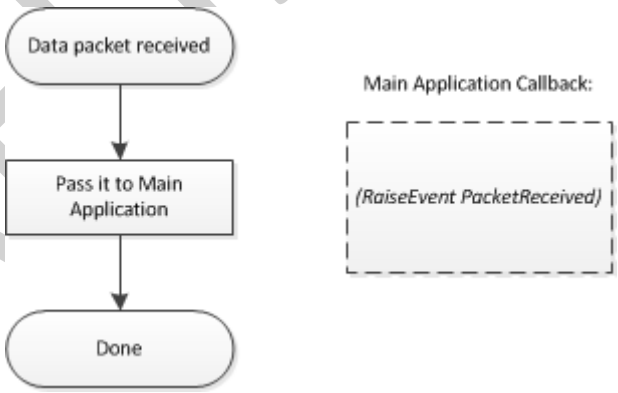
4.1 Initial and Data Send procedure

Initial and send procedure is straight-forward. The workflow is the following:



4.2 Data Receive

After successfully performed Initial procedure, the workflow when Android device sends Data Packet is the following:



5. API Functions and Properties

5.1 Init

Syntax	`Init()`
Parameters	None
Return value	Boolean, indicates success/fail result of the function
Discussion	This function performs various initial internal actions.

5.2 Dispose

Syntax	`Dispose()`
Parameters	None
Return value	Boolean, indicates success/fail result of the function
Discussion	This method must be executed at the end of the APIs usage. It performs a cleanup procedure. APIs can be re-initialized afterwards with `Init` function.

5.3 Send

Syntax	`Send(ByVal bytPacket() as byte)`
Parameters	• bytPacket – array of bytes which are to be delivered to Android device
Return value	Boolean, indicates success/fail result of the function
Description	This method is used when data packet is to be delivered to Android's Custom Application. Maximum parameter array size is 8 bytes. When larger amount of data needs to be sent, it shell be split into smaller chunks size up to 8 bytes and send in a sequence. The same chunks order appearance is guaranteed on receiver side.

5.4 IsConnected

Syntax	`IsConnected()`
Parameters	None
Return value	Boolean, indicates the current status of Bluetooth connection to the peer
Description	This function returns the status of physical Bluetooth link. It does not indicate if the software application is running on remote side.

5.5 GetLastError

Syntax	`GetLastError()`
Parameters	None
Return value	Integer, possible values are listed in Appendix below
Description	When there are no errors, this property has value `NO_ERROR`. If the error equals to `ERR_EXCEPTION`, further details are available via `GetLastExceptionMessage` property. After solving the error condition, the error status needs to be cleared with `ClearLastError` subroutine.

5.6 GetLastExceptionMessage

Syntax	`GetLastExceptionMessage()`
Parameters	None
Return value	String, this is OS-dependent string, describing error details in case of Exception occurrence.
Description	If error status differs to `ERR_EXCEPTION`, this property returns Nothing.

5.7 ClearLastError

Syntax	`ClearLastError()`
Parameters	None
Return value	None
Description	This subroutine sets error state back to `NO_ERROR`. Be aware that this call does not solve the problem itself in any way.

6. Main application skeleton

Callback to the Main application in case of received data packet is to be implemented with handler.

APIs implement public event `PacketReceived(ByVal bytPacket() As Byte)`. The main application has to implement handler, like:

```
Public Sub HandleReceivedPacket(ByVal bytPacket() As Byte) Handles API.PacketReceived
	Try
		' Parse information from bytPacket() array
	Catch ex As Exception
	End Try
End Sub
```

7. Appending

A. API Constants

Error constants:

Name	Value
NO_ERROR	0
ERR_NOT_INITIALIZED	1
ERR_PACKET_EXCEEDS_8_BYTES	2
ERR_UNABLE_TO_SEND_PACKET	3
ERR_UNABLE_TO_RECEIVE_RESPONSE	4
ERR_RESPONSE_MISSING	5
ERR_EXCEPTION	6

Codename :: Wheel

WHO INVENTED THE WHEEL ?

Nobody knows.

It is believed that the first wheels were used in Mesopotamia in the 4th millennium B.C. The wheels supposedly spread all over the world from there.

Some ascribe the invention of the wheel to prehistoric Europe.

World oldest wooden wheel with axle
City Museum Ljubljana, photo Matevž Paternoster

The oldest wooden wheel in the world, which is over 5000 years old according to the analyses, was found while researching the crannog settlement at location Stara gmajna pri Vrhniki, Slovenia.

In the spring of 2002, a team from the Slovenian Institute of Archaeology continued with the project of wood sampling at the mentioned location. A surprise awaited them in one of the drainage ditches. Besides rich findings and two dugouts, they also found the remains of a wooden wheel at the bottom of the ditch that had already been partially damaged by construction machines when they were deepening the ditch. The ditch was widened at the site of the discovery so that they also found the axle that had become separated from the wheel.

The wheel was composed of two ash wood plates that were connected by four oak wedges and had a rectangular aperture in the center, where the axle was mounted. Its diameter was 72cm (28 inch) and it had a thickness of approximately 5cm (2 inch). According to the dendrochronological research the wheel was made from ash wood that comprises both plates, a trunk with the diameter of at least 40cm (16 inch) and was made from a tree that was approximately 80 years old. The choice of ash was not coincidental, because of its strength and because it grew in the vicinity of the crannogs and because it can grow to the dimensions that were needed for large boards without any knots. The axle was constructed from one piece of oak wood and was 124cm (49 inch) long. The ending of the axle was rectangular and fitted into the opening of the wheel. The axle was attached to the wheels with oak wood wedges, which meant that the axle rotated together with the wheels.

According to the opinion of the experts, the wheel and the axle were a part of a two-wheel cart – a pushcart.

The wheel and the axle were dated on the basis of stratigraphic data with dendrochronological research and with the radiocarbon method. The wheel is approximately 5,150 years old and is contemporary with the settlement of Stara gmajna, where it was discovered. Radiocarbon dating was performed in the VERA laboratory (Vienna Environmental Research Accelerator) in Vienna, Austria.

(Source: http://www.koliscar.si)

FLYFISH TECHNOLOGIES headquarters is located about 20 kilometers (12 miles) north-east from the location where this world oldest wooden wheel has been found.