WatchBP Office Blood Pressure Monitor API Documentation

Content

1. WatchBPOfficeHid class

Enables PC to communicate with WatchBP Office devices over USB.

1.1. Inheritance Hierarchy

WatchBPOffice.WatchBPOfficeHid

Namespace: WatchBPOffice

1.2. Properties

Name	Description
CanCommunication	Gets a value indicating whether the device can communicate or not.
MsgType	Gets the type of the message which is return by the device.

1.3. Methods

Name	Parameters	Description
RegisterHandle (IntPtr)	*Handle* Type: System.IntPtr	Registers the application to let it be notified for the device events.
ParseMessages (ref Message)	*m* Type: System.Windows.Forms.Message	Filters the messages that are passed for the device change messages only. And parse them to take appropriate action.
InitWatchBPSDK(string)	*key* Type: System.String	Registers this SDK to let all APIs be available. If the input key is not valid, an exception will occur.
WriteCmd (CmdEnum)	*commandType* Type: WatchBPOfficeHid.CmdEnum The type of command you want to write to the device.	Writes command to the device.

1.4. Events

Name	Description
SpecifiedDeviceRemoved	Occurs when a usbhid device with vid: 0x04B4 and pid: 0x5500 is removed.
SpecifiedDeviceArrived	Occurs when a usbhid device with vid: 0x04B4 and pid: 0x5500 is plugged in.
FullDataReceived	Occurs when the device returns memory records.
MsgReceived	Occurs when the device returns information, including error message, measurement type, processing status or the device name.

1.5. Enumerations

1.5.1. CmdEnum Enumeration

Name	Value	Description
None	0	
RequestDeviceVer	1	Request for the version of the BPM (3Eh)
SelectLeftArm	2	Select left arm to take a measurement (Follow up mode) (30h)
SelectRightArm	3	Select right arm to take a measurement (Follow up mode) (31h)
SelectScreen	4	Select two arms to take a measurement (First Visit mode) (32h)
SelectABI	5	Select arm and leg to take a measurement (ABI mode) (35h)
Stop	6	Stop any actions of BPM (21h)
Start	7	Start to take a measurement (25h)
HideData	8	Hide data function on (33h)
CancelHideData	9	Hide data function off (34h)

1.5.2. MsgTypeEnum Enumeration

Name	Value
none	0
Err	1
deviceName	2
measureType	3
deviceStatus	4

1.6. Fields

Name	Description
dataTransferHid	Need to be initialized according to current device name for receiving the blood pressure raw data correctly.

1.7. Examples

1.7.1 Create WatchBPOfficeHid class

The following example shows how to Create **WatchBPOfficeHid** and handle the SpecifiedDeviceRemoved, SpecifiedDeviceArrived, MsgReceived, and FullDataReceived events.

```
...
using WatchBPOffice;
class MainFrm:Form
{
    private WatchBPOfficeHid watchBPOfficeHid;
    public MainFrm()
    {
        ...
        watchBPOfficeHid = new WatchBPOfficeHid();

        // key: a specific string to register this SDK
        watchBPOfficeHid.InitWatchBPSDK("key");

        watchBPOfficeHid.SpecifiedDeviceRemoved+=watchBPOffice_Onspeci
fiedDeviceRemoved;
        watchBPOfficeHid.SpecifiedDeviceArrived+=watchBPOffice_OnSpeci
fiedDeviceArrived;
        watchBPOfficeHid.MsgReceived += WatchBPOfficeHid_OnMsgReceived;
        watchBPOfficeHid.FullDataReceived +=
WatchBPOfficeHid_OnFullDataReceived;
    }
    protected override void OnHandleCreated(EventArgs e)
    {
        base.OnHandleCreated(e);
        watchBPOfficeHid.RegisterHandle(Handle);
    }
    protected override void WndProc(ref Message m)
    {
        base.WndProc(ref m);
        watchBPOfficeHid.ParseMessages(ref m);
    }
```

```
    private void watchBPOffice_OnspecifiedDeviceRemoved(object sender,
EventArgs e)
    {
        ...
    }
    private void watchBPOffice_OnSpecifiedDeviceArrived(object sender,
EventArgs e)
    {
        ...
    }
    private void watchBPOffice_OnMsgReceived(object sender, MsgEventArgs
e)
    {
        ...
    }
    private void watchBPOffice_OnFullDataReceived(object sender,
FullDataEventArgs e)
    {
        ...
    }

}
```

1.7.2. Request the device's version

The following example shows how to get the raw data of the 3Eh command (request the device's version).

```
private void watchBPOffice_OnMsgReceived(object sender, MsgEventArgs e)
{
    if (InvokeRequired)     // to make a thread-safe calls
    {
        try
        {
            Invoke(new MsgEventHandler(WatchBPOfficeHid_OnMsgReceived), new
object[] { sender, e });
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
     }
    else
    {
         byte[] deviceNameByteAry = new byte[]{};
         if (watchBPOfficeHid.MsgType ==
WatchBPOfficeHid.MsgTypeEnum.deviceName)
              deviceNameByteAry = e.rawData;  // raw data from the device
    }
}
```

1.7.3. Read error message

The following example shows how to get the raw data received from the device when a measurement error occurs.

```
private void watchBPOffice_OnMsgReceived(object sender, MsgEventArgs e)
{
    if (InvokeRequired)     // to make a thread-safe calls
    {
        try
        {
            Invoke(new MsgEventHandler(WatchBPOfficeHid_OnMsgReceived), new
object[] { sender, e });
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
     }
    else
    {
         byte[] errMsgByteAry = new byte[]{};
         if (watchBPOfficeHid.MsgType == WatchBPOfficeHid.MsgTypeEnum.Err)
              errMsgByteAry = e.rawData;  // raw data from the device
    }
}
```

1.7.4. Select Routine/Central mode

The following example shows how to select Routine/Central mode (0x30 or 0x31).

```
...
// Select Routine/Central mode with the left arm. (0x30)
if (watchBPOfficeHid.CanCommunication)
    watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.SelectLeftArm);

...
// Select Routine/Central mode with the right arm. (0x31)
if (watchBPOfficeHid.CanCommunication)
    watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.SelectRightArm);
```

1.7.5. Select Screen mode

The following example shows how to select Screen mode (0x32).

```
...
if (watchBPOfficeHid.CanCommunication)
    watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.SelectScreen);
```

1.7.6. Select ABI mode

The following example shows how to select ABI mode (0x35).

```
...
if (watchBPOfficeHid.CanCommunication)
    watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.SelectABI);
```

1.7.7. Choode Hide data or not

The following example shows how to hide (0x33) / show (0x34) data.

```
...
if (watchBPOfficeHid.CanCommunication)
    watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.HideData);
...
if (watchBPOfficeHid.CanCommunication)
    watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.CancelHideData);
```

1.7.8. Start measurement

The following example shows how to start measurement (0x25).

```
...
if (watchBPOfficeHid.CanCommunication)
     watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.Start);
```

1.7.9. Stop measurement

The following example shows how to stop measurement (0x21).

```
...
if (watchBPOfficeHid.CanCommunication)
     watchBPOfficeHid.WriteCmd(WatchBPOfficeHid.CmdEnum.Stop);
```

1.7.10. Read current measure mode

The following example shows how to get the raw data of the 30h (select left arm to take a measurement), 31h (select right arm to take a measurement), 32h (select two arms to take a measurement), or 35h (select ABI mode) command.

```
private void watchBPOffice_OnMsgReceived(object sender, MsgEventArgs e)
{
    if (InvokeRequired)     // to make a thread-safe calls
    {
        try
        {
            Invoke(new MsgEventHandler(WatchBPOfficeHid_OnMsgReceived), new
object[] { sender, e });
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
     }
    else
    {
         byte[] measureTypeByteAry = new byte[]{};
         if (watchBPOfficeHid.MsgType ==
WatchBPOfficeHid.MsgTypeEnum.measureType)
             measureTypeByteAry = e.rawData; // raw data from the device
    }
}
```

1.7.11. Read current device status

The following example shows how to get the raw data received from the device during the measurement.

```
private void watchBPOffice_OnMsgReceived(object sender, MsgEventArgs e)
{
    if (InvokeRequired)     // to make a thread-safe calls
    {
        try
        {
            Invoke(new MsgEventHandler(WatchBPOfficeHid_OnMsgReceived), new
object[] { sender, e });
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
     }
    else
    {
         byte[] deviceStatusByteAry = new byte[]{};
         if (watchBPOfficeHid.MsgType ==
WatchBPOfficeHid.MsgTypeEnum.deviceStatus)
              deviceStatusByteAry = e.rawData;     // raw data from the device
    }
}
```

1.7.12. Read memory records

The following example shows how to get the raw data received from the device after a measurement finished.

```
private void watchBPOffice_OnFullDataReceived(object sender,
FullDataEventArgs e)
{
    if (InvokeRequired)     // to make a thread-safe calls
    {
        try
        {
            Invoke(new
FullDataEventHandler(WatchBPOfficeHid_OnFullDataReceived), new object[]
{ sender, e });
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
     }
    else
    {
         byte[] bpDataByteAry = e.rawData;    // raw data from the device
    }
}
```

2. MsgEventArgs class

Provides data for the MsgReceived event.

2.1. Inheritance Hierarchy

WatchBPOffice.MsgEventArgs

Namespace: WatchBPOffice

2.2. Properties

Name	Description
Msg	Gets a message about device status, device measure type or error.
rawData	Gets the raw data of the message.

3. FormatTransfer class

Converts the byte representation of raw data to string.

3.1. Inheritance Hierarchy

WatchBPOffice.Decode.FormatTransfer

Namespace: WatchBPOffice.Decode

3.2. Extension Methods

Name	Parameters	Description
HexToASC(byte[], int)	*rawData* Type: System.Byte[] *size* Type: System.Int32	Converts the byte array to the string and returns it.

3.3. Examples

3.3.1. Parse the raw data from MsgReceived event

The following example shows how to parse the raw data from MsgReceived event.

```
using WatchBPOffice.Decode;
...
    byte[] dataAry;       // raw data received from MsgReceived event
    ...
    string result = FomatTransfer.HexToASC(dataAry,dataAry.Length);
```

4. FullDataEventArgs class

Provides data for the FullDataReceived event.

4.1. Inheritance Hierarchy

WatchBPOffice.FullDataEventArgs

Namespace: WatchBPOffice

4.2. Properties

Name	Description
data	Gets the blood pressure records (Data class).
rawData	Gets the raw data of the blood pressure records.

5. Data class

Stores whole records of data.

5.1. Inheritance Hierarchy

WatchBPOffice.Decode.Data

Namespace: WatchBPOffice.Decode

5.2. Properties

Name	Descriptions
ID	Gets or sets the ID of the data.
Date	Gets or sets the measurement date and time of the data.
Pulse	Gets or sets the pulse of the data.
Avg_Pulse	Gets or sets the average pulse of the data.
sIAD	Gets or sets the difference in systole between arms.
Avg_sIAD	Gets or sets the average sIAD of the data.
dIAD	Gets or sets the difference in diastole between arms.
Avg_dIAD	Gets or sets the average dIAD of the data.
RecommendedArm	Gets or sets the recommended arm of the data.
MeasureType	Gets or sets the measure type of the data.
ABI	Gets or sets the ABI index of the data.
Avg_ABI	Gets or sets the average ABI index of the data.
L_Sys	Gets or sets the systole of the left side.
L_Dia	Gets or sets the diastole of the left side.
L_MAP	Gets or sets the MAP of the left side.
L_PP	Gets or sets the PP of the left side.
L_CBP	Gets or sets the CBP of the left side.
L_CPP	Gets or sets the CPP of the left side.
L_cDia	Gets or sets the cDia of the left side.
Avg_L_Sys	Gets or sets the average systole of the left side.
Avg_L_Dia	Gets or sets the average diastole of the left side.
Avg_L_MAP	Gets or sets the average MAP of the left side.
Avg_L_PP	Gets or sets the average PP of the left side.
Avg_L_CBP	Gets or sets the average CBP of the left side.
Avg_L_CPP	Gets or sets the average CPP of the left side.

Avg_L_cDia	Gets or sets the average cDia of the left side.
R_Sys	Gets or sets the systole of the right side.
R_Dia	Gets or sets the diastole of the right side.
R_MAP	Gets or sets the MAP of the right side.
R_PP	Gets or sets the PP of the right side.
R_CBP	Gets or sets the CBP of the right side.
R_CPP	Gets or sets the CPP of the right side.
R_cDia	Gets or sets the cDia of the right side.
Avg_R_Sys	Gets or sets the average systole of the right side.
Avg_R_Dia	Gets or sets the average diastole of the right side.
Avg_R_MAP	Gets or sets the average MAP of the right side.
Avg_R_PP	Gets or sets the average PP of the right side.
Avg_R_CBP	Gets or sets the average CBP of the right t side.
Avg_R_CPP	Gets or sets the average CPP of the right side.
Avg_R_cDia	Gets or sets the average cDia of the right side.

5.3. Enumerations

5.3.1. RecomArmEnum Enumeration

Name	Value
none	0
overCriterion	1
leftArm	2
rightArm	3

5.3.2. MeasureTypesEnum Enumeration

Name	Value
none	0
RightAnkleRightBrachial	1
LeftAnkleRightBrachial	2
LeftAnkleLeftBrachial	3
RightAnkleLeftBrachial	4
Screen	5
CentralLeft	6
CentralRight	7

5.4. Structs

5.4.1. IndexStruct Struct

Name	Description
Mode	A string indicating measurement mode of the data. “1”: Screen mode, “2”: Central/Routine mode, “3”: ABI mode
Set_SYM	
Afib	A string indicating whether there is Afib or not. “1”: True, “0”: False.
PAD	A string indicating whether there is PAD or not. “1”: True, “0”: False.
MeasureTimes	The measure times of the data.

5.5. Fields

Name	Description
Index	Shows index of the data
Avg_Index	Shows index of the average data.

6. DataParser class

Converts the byte representation of raw data to their real data (Data class).

6.1. Inheritance Hierarchy

WatchBPOffice.Decode.DataParser
Namespace: WatchBPOffice.Decode

6.2. Extension Methods

Name	Parameters	Description
ParseData(byte[], DeviceNameEnum, out Data)	*BPDataByteAry* Type: System.Byte[] *deviceName* Type: WatchBPOffice.Decode.DeviceNameEnum *result* Type: WatchBPOffice.Decode.Data	Converts the byte array to the Data and returns it. Returns a value indicating whether the checksum is correct or not.
ParseDeviceName(byte[], out string)	*dataAry* Type: System.Byte[] *deviceName* Type: System.String	Converts the byte array to the string and returns it. Returns a value indicating whether the checksum is correct or not.

6.3. Global Enumerations (WatchBPOffice.Decode)

6.3.1. DeviceNameEnum Enumeration

Name	Value
BT040	0
BT0c0	1
BT0c2	2

6.4. Examples

6.4.1. Parse the raw data of device's blood pressure data

The following example shows how to parse the raw data of device's blood pressure data.

```
using WatchBPOffice.Decode;
...
    byte[] dataAry;       // raw data received from the device
    DeviceNameEnum currentDeviceName;
    ...
    Data result;
    if (DataParser.ParseData(dataAry, currentDeviceName, out result))
    {
        MessageBox.Show(
        string.Format("L_Sys:{0}, L_Dia:{1}, L_MAP:{2}, L_PP:{3}, R_Sys:{4},
    R_Dia:{5}, R_MAP:{6}, R_PP:{7}, Pulse:{8}, L_Avg_Sys:{9}, L_Avg_Dia:{10},
    L_Avg_MAP:{11},L_Avg_PP:{12}, R_Avg_Sys:{13}, R_Avg_Dia:{14},
    R_Avg_MAP:{15},R_Avg_PP:{16}, Avg_Pulse:{17}, L_CBP:{18}, L_CPP:{19},
    R_CBP:{20}, R_CPP:{21}, L_Avg_CBP:{22}, L_Avg_CPP:{23}, R_Avg_CBP:{24},
    R_Avg_CPP:{25}", result.L_Sys, result.L_Dia, result.L_MAP, result.L_PP,
    result.R_Sys, result.R_Dia, result.R_MAP, result.R_PP, result.Pulse,
    result.Avg_L_Sys, result.Avg_L_Dia, result.Avg_L_MAP, result.Avg_L_PP,
    result.Avg_R_Sys, result.Avg_R_Dia, result.Avg_R_MAP, result.Avg_R_PP,
    result.Avg_Pulse,result.L_CBP,result.L_CPP,result.R_CBP,result.R_CPP,re
    sult.Avg_L_CBP,result.Avg_L_CPP,result.Avg_R_CBP,result.Avg_R_CPP)
        );
    }
```

6.4.2. Parse the raw data of device's Name

The following example shows how to parse the raw data of device's name.

```
using WatchBPOffice.Decode;
...
    byte[] dataAry;       // raw data received from the device
    ...
    string deviceName;
    if (DataParser.ParseDeviceName(dataAry, out deviceName))
        MessageBox.Show("The device's version is {0}",deviceName);
```

7. IDataCommunication interface

Provides functionality to receive blood pressure raw data from the device.

7.1. Inheritance Hierarchy

WatchBPOffice.IDataCommunicaation

Namespace: WatchBPOffice

7.2. Methods

Name	Parameters	Description
CheckDataSize (byte[], out Data)	*rawData* Type: System.Byte[] *data* Type: WatchBPOffice.Decode.Data	Receives blood pressure raw data from the device. Returns an error message. If there is no error this method will return an empty string.

8. DataCommABIAfibHid class

Enables PC to receive blood pressure raw data from the **WatchBP Office ABI** or **WatchBP Office AFIB** device correctly.

8.1. Inheritance Hierarchy

WatchBPOffice.DataCommABIAfibHid
Namespace: WatchBPOffice

8.2. Constructors

Name	Parameters	Description
DataCommABIAfibHid (WatchBPOfficeHid)	*watchBPOfficeHid* Type: WatchBPOffice.WatchBPOfficeHid	Initializes a new instance of the DataCommABIAfibHid class that contains an instance of WatchBPOfficeHid class.

8.3. Explicit Interface Implementations

Name	Parameters	Description
CheckDataSize (byte[], out Data)	*rawData* Type: System.Byte[] *data* Type: WatchBPOffice.Decode.Data	Receives blood pressure raw data from the device. If there is no error this method will return an empty string; otherwise, returns an error message.

8.4. Examples

8.4.1. Enable PC to receive blood pressure raw data from WatchBP Office ABI or WatchBP Office AFIB device

The following example shows how to enable PC to receive blood pressure raw data from WatchBP Office ABI or WatchBP Office AFIB device.

```
using WatchBPOffice;
...
class MainFrm:Form
{
    private WatchBPOfficeHid watchBPOfficeHid;
    DeviceNameEnum currentDeviceName;
    ...
    public MainFrm()
    {
        // Confirm current connected device is WatchBP Office ABI or WatchBP
Office AFIB
        if (currentDeviceName == DeviceNameEnum.BT0c0 || currentDeviceName
== DeviceNameEnum.BT0c2)
            watchBPOfficeHid.dataTransferHid = new
DataCommABIAfibHid(watchBPOfficeHid);
    }
}
```

9. DataCommCentralHid class

Enables PC to receive blood pressure raw data from the **WatchBP Office Central** device correctly.

9.1. Inheritance Hierarchy

WatchBPOffice.DataCommCentralHid
Namespace: WatchBPOffice

9.2. Constructors

Name	Parameters	Description
DataCommCentralHid (WatchBPOfficeHid)	*watchBPOfficeHid* Type: WatchBPOffice.WatchBPOfficeHid	Initializes a new instance of the DataCommCentralHid class that contains an instance of WatchBPOfficeHid class.

9.3. Explicit Interface Implementations

Name	Parameters	Description
CheckDataSize (byte[], out Data)	*rawData* Type: System.Byte[] *data* Type: WatchBPOffice.Decode.Data	Receives blood pressure raw data from the device. If there is no error this method will return an empty string; otherwise, returns an error message.

9.4. Examples

9.4.1. Enable PC to receive blood pressure raw data from WatchBP Office Central device

The following example shows how to enable PC to receive blood pressure raw data from WatchBP Office Central device.

```
using WatchBPOffice;
...
class MainFrm:Form
{
    private WatchBPOfficeHid watchBPOfficeHid;
    private DeviceNameEnum currentDeviceName;
    ...
    // Confirm current connected device is WatchBP Office Central
    if (currentDeviceName == DeviceNameEnum.BT040)
        watchBPOfficeHid.dataTransferHid = new
    DataCommCentralHid(watchBPOfficeHid);
}
```