

Megawin

USB DFU Library for VB

User Manual

Index

DFU_Library for VB	1
1. Introduction	3
2. Files Needed	3
3. How to Use DFU DLL	3
3.1. Debug Mode:	3
3.2. Execution Mode:	3
4. How to use the Internal Built Function	4
4.1. DFU_Reset_To_ISP	4
4.1.1. Application with HID interface.....	4
4.1.2. Application without HID interface.....	4
4.2. DFU_Reset_To_AP	4
4.3. DFU_Download	4
4.4. DFU_Get_ProcessCount	5
5. Example	5
6. Revision History	6

1. Introduction

This document explains how to use DFU DLL in the Visual Basic 6.0 environment for device firmware upgrade.

2. Files Needed

- 2.1. DFU.DLL
- 2.2. DFU.bas

3. How to Use DFU DLL

Activate the Visual Basic Software. Use Visual Basic 6.0 to develop the application program, please be guided by the following steps:

3.1. Debug Mode:

- 3.1.1. Please copy DFU.DLL mentioned in section #2, into your hard disk. For example, if your operating system is Windows XP, please copy the file in C:\Windows directory.
- 3.1.2. Create a New Project. Go to the project menu then choose Add Module. Using the Add Module menu, add the DFU.Bas file.

3.2. Execution Mode:

Copy the DFU.DLL into the same directory where the execution file is located.

4. How to use the Internal Built Function

The DLL functions are described as following:

4.1. DFU_Reset_To_ISP

The function will trigger device to re-boot into ISP code from firmware AP region. Then, the device will execute USB DFU code in ISP region to receive new firmware to upgrade AP firmware code. This command is packaged through USB HID request. So, it can be applied for HID device, such as "Megawin EasyPOD", "USB Keyboard", or a USB composite device with HID interface. The function format is described as following:

Function : Public Declare Function DFU_Reset_To_ISP Lib "DFU.DLL" (ByVal VID As Long, ByVal PID as Long) As Long
Return : TRUE (1) means reset to ISP mode is successful. FALSE (0) means fails.
Parameter : VID, Vendor ID.
PID, Product ID.

If the device behaves without HID interface, it should need a user specific command to trigger device firmware AP to re-boot in ISP region and perform USB DFU progress

4.1.1. Application with HID interface

This application can use DFU_Reset_To_ISP function call in this DLL. Host AP only call this function to trigger device to re-boot in ISP.

```
DFU_Reset_To_ISP(VID, PID) ;
```

For device side, firmware only execute following procedure to re-boot in ISP region.

```
Receive RESET command (0x44) /* data stage (RXDAT) in HID Set_Feature Request */  
/* RXDAT [0] =0x44, RXDAT [1] ~ RXDAT[n] = Don't care */  
Write ISPCR = 0x68; /* sfr ISPCR = 0xE7 that is define in "REG_MG84FL54.H" */
```

4.1.2. Application without HID interface

This application can not use DFU_Reset_To_ISP function call. User should define a specific vendor command (**RESET**) in host AP to inform device to re-boot into ISP region.

```
Handle = Create_Device();  
Set_Reset_Command( Handle . RESET,.. ); /* This function define by user... */  
Delay 5s; /* Wait device reset. */  
Close_Handle ( Handle );
```

For device side, firmware only execute following procedure to re-boot in ISP region.

```
Receive RESET from PC side AP /*specific vendor command */  
Write ISPCR = 0x68; /* sfr ISPCR = 0xE7 that is define in "REG_MG84FL54.H" */
```

4.2. DFU_Reset_To_AP

If firmware AP upgrade has been finished, host DFU AP issue this command to trigger device re-boot into AP region. After this trigger, device will execute user firmware AP to perform new function.

Function : Public Declare Function DFU_Reset_To_AP Lib "DFU.DLL" () As Long
Return : TRUE (1) means reset to AP mode is successful. FALSE (0) means fails.
Parameter : None

4.3. DFU_Download

This function call will transmit new firmware code to device and upgrade to AP region.

Function : Public Declare Function DFU_Download Lib "DFU.DLL" (lpFW As Byte, ByVal Addr As Long, ByVal Size as Long, ByVal CheckSum As Long) as Long
Return : TRUE (1) means download is successful. FALSE (0) means fails.
Parameter : lpFW, pointer to the buffer containing the data to write to the device.

Addr, the offset of start address.

Size, the size of buffer to be downloaded.

CheckSum, the checksum of the buffer. The checksum is calculated by summing the values of all bytes in the buffer modulo 65536.

4.4 DFU_Get_ProcessCount

This function call will return the percentage in download process.

Function : Public Declare Function DFU_Get_ProcessCount Lib "DFU.DLL" () As Integer

Return : An Integer value represent the percentage in download process.

Parameter : None.

5. Example

```
Private Sub Command1_Click ()
    Dim VID As Long
    Dim PID As Long
    Dim Size As Long
    Dim Addr As Long
    Dim CheckSum As Long
    Dim result As Long
    Dim intFileNum As Integer
    Dim strFilePath As String
    Dim bytSendData () As Byte
    Dim intProcessCount as Integer
    ReDim bytSendData (10)
    VID = &HE6A&
    PID = &H317&
    Addr = 0
    result = DFU_Reset_To_ISP(VID, PID)
    If result = 1 Then
        MsgBox "OK"
        intFileNum = FreeFile(1)
        strFilePath = "D:\Temp\DFU\VBTest\EasyPOD.BIN"
        Open strFilePath For Binary Access Read As intFileNum
        CheckSum = &HC82C&
        Size = LOF (intFileNum)
        ReDim bytSendData (Size - 1)
        Get intFileNum, , bytSendData
        Close intFileNum
        intProcessCount = DFU_Get_ProcessCount
        result = DFU_Download(bytSendData(0), Addr, Size, CheckSum)
        result = DFU_Reset_To_AP
        intProcessCount = DFU_Get_ProcessCount
    Else
        MsgBox ("err")
    End If
End Sub
```

6. Revision History

Revision	Description	Date
V1.00	Release version	2008/06/24
V1.01	Add new function of "DFU_Get_ProcessCount"	2008/11/14