

```
1: //-----
2: // Interface for Usb-API for PIC 18F4550 Demo board
3: // Author: Gerhard Burger
4: // E-Mail: gCoolfire@yahoo.de
5: // Web: http://members.aon.at/geburger/
6: // Last Update: 2nd of June 2005
7: // Version 1.0
8: // Compiler: Borland Delphi, Version 7.0
9: // This software is free for non commercial use, for commercial use, please contact
10: // the author
11: //
12: // THIS SOFTWARE IS PROVIDED IN AN "AS IS" CONDITION. NO WARRANTIES,
13: // WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED
14: // TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
15: // PARTICULAR PURPOSE APPLY TO THIS SOFTWARE. THE COMPANY SHALL NOT,
16: // IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL OR
17: // CONSEQUENTIAL DAMAGES, FOR ANY REASON WHATSOEVER.
18: //-----
19: Unit UsbAPI;
20: Interface
21: Uses Windows, SysUtils;
22:
23: Const MPUSB_FAIL           = 0;
24:         MPUSB_SUCCESS       = 1;
25:         MP_WRITE            = 0;
26:         MP_READ             = 1;
27:         MAX_NUM_MPUSB_DEV   = 127;
28:
29: // MAX_NUM_MPUSB_DEV is an abstract limitation.
30: // It is very unlikely that a computer system will have more
31: // then 127 USB devices attached to it. (single or multiple USB hosts)
32:
33: Function MPUSBGetDeviceCount(pVID_PID: PCHAR) : DWORD; cdecl;
34:
35: Function MPUSBOpen(instance : DWORD ; pVID_PID :PCHAR; pEP : PCHAR;
36:                   dwDir : DWORD; dwReserved : DWORD): THANDLE;cdecl;
37:
38: Function MPUSBGetDLLVersion():DWORD;cdecl;
39:
40: // Reads a data package to the USB device
41: // HANDLE: Handle to the device
42: // pData: Pointer to the input buffer
43: // dwDataLen: expected count of bytes to read
44: // dwDataLenRead: actually count read data
45: // waiting time
46: Function MPUSBRead(HANDLE: THANDLE; pData :Pointer; dwDataLen :DWORD;
47:                   pDataLenRead :PDWORD; dwMilliseconds :DWORD):DWORD; cdecl;
48:
49: // Writes a data package to the USB device
50: // HANDLE: Handle to the device
51: // pData: Pointer to the output buffer
52: // dwDataLen: expected count of bytes to send
53: // dwDataLenSent: actually count of sent data
54: // waiting time
55: Function MPUSBWrite(HANDLE :THANDLE; pData :Pointer; dwDataLen : DWORD;
56:                   pLengthSent :PDWORD; dwMilliseconds :DWORD):DWORD; cdecl;
57:
58: // see microchip documentation
59: Function MPUSBReadInt(HANDLE :THANDLE; pData :Pointer; dwDataLen : DWORD;
60:                      pLengthReceive :PDWORD; dwMilliseconds :DWORD):DWORD; cdecl;
61:
62: Function MPUSBClose(HANDLE : THANDLE ): Boolean;cdecl;
63:
64: Implementation
65:
```

```
66: Function MPUSBGetDLLVersion;      cdecl; external 'MPUSBAPI.Dll' index 1;
67: Function MPUSBGetDeviceCount;     cdecl; external 'MPUSBAPI.Dll' index 2;
68: Function MPUSBOpen;                cdecl; external 'MPUSBAPI.Dll' index 3;
69: Function MPUSBRead;                cdecl; external 'MPUSBAPI.Dll' index 4;
70: Function MPUSBWrite;               cdecl; external 'MPUSBAPI.Dll' index 5;
71: Function MPUSBReadInt;             cdecl; external 'MPUSBAPI.Dll' index 6;
72: Function MPUSBClose;              cdecl; external 'MPUSBAPI.Dll' index 7;
73:
74: end.
```