

Documentation for

Win95IO.DLL

Version 1.4

WinIO and Win95IO © 1994-1997 Vincent Himpe
All rights reserved. Released as Public Domain /Shareware

What is Win95IO.

WinIO is a DLL or Dynamic Linkable Library for Windows 95 and later. It allows you to perform very low level i/o functions which windows locks out under normal circumstances.

Most programming languages for Windows will not let you read or write physical ports of your machine directly.(Pascal / Visual Basic / C / C++). This DLL gives any programming language the capability of performing these IO actions.

Note that this DLL is a native 32 bit core. This means that you cannot use it to run on Win3.11 .

Furthermore you need a native 32 bit development tool like VB4.0 Professional or VB5 and higher. Since this is still a DLL and not a VXD you can NOT use this to perform hardware I-O on Windows NT.

Why ?

Win95IO was originally created because somebody needed it (I did). I wanted to control an electronic circuit connected to the printerport. Therefore i needed a way to send some data to the port and read back data from the port. This can easily be done from within DOS but not from within Windows.

After experimenting with several tricks (eg: run a dos program from within windows that saves data to file and then read this file from within windows) I decided there was only one neat solution : a DLL.

How ?

The DLL was created using Turbo Pascal 2.0 for Windows and Turbo Assembler 3.5. The actual engine of the DLL is written entirely in Assembler. This allows for very high throughput and access to low level stuff. The Core is then compiled to an .OBJ file and the functions exported. The rest of the work is done by Turbo Pascal . This links the .OBJ file to the necessary hull to create a DLL.

Description of the functions.

The DLL restores the following functions :

INP	ADRESS	reads a byte (8 bit) from port ADRESS
OUT	ADRESS,DATA	writes a byte to port ADRESS
INPW	ADRESS	reads a word (16 bit) from port ADRESS
OUTW	ADRESS,DATA	write a word (16 bit) to port ADRESS

INP Reads a byte

Example : A = INP (889)
 A = INP (&H379)

Both commands read a byte from the same address. The first line uses decimal notation. The second uses hexadecimal notation. The resulting value will be put in variable A in decimal form.

Note : reading port &h 379 gives you the status of the printer on LPT1.

OUT Writes A Byte

Example OUT &H378 , 65

This sends a 65 to port &h378 which is the printer port LPT1. 65 will represent a capital a (A) on the printer. Just as with INP it does not matter what form the numbers are in. They can be DECIMAL or hexadecimal or even octal or binary provided the language you are using can handle these systems.

INPW (input Wide) reads a word

Example: A = INPW(&H379)

This will read the word on port 889. Since the printerport only uses the lower 8 bit (low Byte) only the these 8 bits will be of meaning. The high Byte or upper 8bit will be meaningless and could contain anything. Therefore you can expect INP (&h379) and INPW (&h379) to give completely different readings.

OUTW (output wide) writes a word

Example: OUTW &H378,65

This will send also character 65 (capital a) to the printer, but it could as well screw up your computer.(it will not damage the computer but it could be needed to reboot). This is due to the fact that you are actually sending 0065. Where 65 goes to the 8 bit in use by the printerport and the 00 goes to the upper 8 bits or the high byte.

Notes on the use of 16 bit operations

Some care should be taken when performing 16 bit IO functions. While you can use them to access 16 bit cards you should keep in mind that the upper 8 bits might be used for some other purpose. Some cards 'steal' upper bytes which are not used. I know of one card that uses these techniques. A Number Nine 64GXE-pro Video Card !.

The High byte of COM2 ,which is never used, serves as the top address lines to my video memory. (4Mbyte on board). Usually this is not used since all communication goes through the video processor. But if you want to grab the video memory you can use these locations to access the memory directly. Therefore one golden rule : use these WIDE commands only on your own cards or on cards of which you know how they work.

General Notes

Keep in mind that , when using these routines , you are not allowed to use :

- numbers below 0
- numbers above 65535 for the target address .
- numbers above 255 for the data in the OUT routine
- numbers above 65535 for the data in OUTW routine

Actually the DLL will accept them but will 'rollover'.

What I mean is : If you send 65537 as an address the DLL will actually write to Address 1

65537 in Binary looks like this :
100000000000000001
you will write 0000000000000001 which equals to 1

Using the DLL from within Visual Basic

Select Project and then Add Module . Then select existing and specify the path to the file WINIO.INC

This file includes the directives for VB on how to use the DLL and where the DLL can be found.

To avoid problems later when installing your programs on other computers you should not specify a path to the WIN95IO.DLL file.

When a DLL needs to be loaded windows will search first the programs installation directory , the Windows directory and the Windows/System directory. If the file cannot be found there the Operating system will scan in the PATH statement of the Autoexec.BAT file. On failure to find the Win95IO.DLL you will get a runtime error.

Distributing programs that use WIN95IO.DLL

To make sure that the library is installed on the target machine you should let the installation wizard from VB do the work for you. It will scan your project and automatically include Win95IO.DLL in your distribution set. To make sure the wizard can find it you should store a copy of the DLL in your Windows\SYSTEM directory.

Note : This DLL is protected by a copyright .

You can use and distribute as many copies of the DLL and possible accompanying material provided that :

- You do NOT SELL the DLL
- You do NOT SELL your program

This means that you cannot make a program that uses the DLL and SELL this program. You can make Freeware that uses the DLL , but you cannot make Shareware or commercial programs that use the DLL.

In case that you want to sell programs that use this DLL you owe me money !.

A commercial user license is available for purchase. Contact me for more information.

Demo's : Pmon95

As demonstration there is a little Windows95 program included that monitors the 2 printerports continuously and displays what is happening on the ports. I included the sourcecode so you can modify it as you want.

Demo's : HIT95

HIT95 is a small program that allows you to monitor a hardware port of your computer.

This tool allows programmers and hardware developers to monitor card activity on /O ports from within windows. The program may be started a number of times. Doing so lots of different ports can be read and written (both 8 and 16 bit ports).

The window is split in 2 portions. The Address and data section. You can select a number of standard addresses with the list box on the left side. If you select user there then you can type in your own address on the textbox below the list box.

The button 'Immediate' does the following: When activated the data you enter in the data field is being transmitted on the spot. When deactivated you must press the GO button to transmit the data.

On the menu there are 4 functions

Quit :	quits the program
Decimal/Hexadecimal :	toggles the readout mode
8 Bit/16 bit :	toggles bus width.
About :	Ego screen

Click on the lower row of selection boxes to code the bits. The upper row of selection boxes shows the current data on the address selected.

Copyright notice and users licence.:

WinIO.DLL (c)1994-1997 Vincent Himpe All rights reserved.

You are allowed to distribute the DLL package provided that you do not modify it in any way. If you distribute it you must make sure that the distribution media contains following files.

- Win95IO.DLL	The DLL itself
- Win95IO.DOC	This File in unmodified form
- Win95IO.BAS	Include file for Visual Basic
- PMON95.BAS	Printmonitor Source in Visual Basic
- PMON95.FRM	Printmonitor form
- PMON95.MAK	Printmonitor project file for VB
- PMONABT.FRM	About form for PMON
- HIT95.BAS	Sourcecode for HiT
- HIT95.FRM	HiT Main form
- HIT95.MAK	Project file for HiT
- ABOUTHIT.FRM:	About from for HiT
- HIT95.DOC	Textfile describing HiT

You are allowed to use the DLL for private purposes only. You are NOT allowed to sell the DLL , neither alone or as part of a program you wrote. If you write a commercial program that will use this DLL then you owe me money !.

As a non-commercial user you can use the DLL for free.

However i would like to ask you to send me the registration form or an e-mail message.(you do not have to send any money , to know that my DLL is beeing used somewhere out there is enough to keep me satisfied)

If you write software that uses the DLL, and you want to sell this software you MUST register and obtain a distribution licence.

Distribution:

This may be distributed without further notice. Any media (floppy, tape, modem, CD) is allowed. This package(WIN95IO.ZIP) may be included in collections like a CD-rom or spread on the internet , BBS , modem etc without written agreement.

Registration Form for Win95IO.DLL

Name : _____

Adress : _____ Nr: _____

City : _____ Zip : _____

State : _____ Country : _____

Uses the Win95IO DLL Version 1.4 for non-commercial purposes only

Orders :

Distribution licence rated at 75 US \$ (contact me by mail for instructions)

Comments :

.....
.....
.....
.....
.....

Send by E-Mail : Vincent.Himpe@ping.be