

Interface of Direct IO DLL to Microsoft® Visual C ++ 6.0

These are the programming guide lines required to interface the Module in VC++ as follows:

1. **Directory Setup** : How to copy the files.?

The files required for a project to use the DirectIO functionalities are provided in the "SetupPacks\VCApp" folder. These files need to be copied to the project directory and some minor settings have to be done to include the DirectIO functionality in the project. The following are the files and corresponding target copy locations:

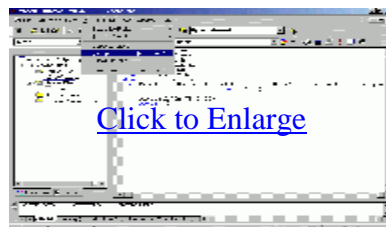
\SetupPacks\VCApp\Directio.h	\{your project folder}\Directio.h
\SetupPacks\VCApp\Directio.lib	\{your project folder}\Directio.lib
\SetupPacks\VCApp\Debug\Directio.dll	\{your project folder}\Debug\Directio.dll
\SetupPacks\VCApp\Debug\Porttalk.sys	\{your project folder}\Debug\porttalk.sys

NOTE: wherever the project's exe is placed the last two files Directio.dll and porttalk.sys must also be placed for the proper functioning of the program.

2. **Compiler Setup** : How to configure the compiler and build settings?

This is required in order that the particular DLL functionalities are inherited by the program during compile time. Here in case of DirectIO DLL the .lib and the .h files require to be instantiated. The "Directio.h" file must be included in the program where the IO functions are being used. To compile the usability of the DLL these are the build settings :

Step 1.



Step2.



After all this settings changes are done the project is ready to compile and run . This setup is already done in the VC ++ Sample App . You can compile and see it.

Functions Provided and Descriptions :

1. void _stdcall dap(void);

This is a test function to check just that if the DLL is working or not. It will show a dialogue box with some message like "Hello I am being called" and on the Title Bar "Direct IO Dll by Bose".

2. void _stdcall outportb(unsigned short,unsigned char);

This function is the main 'output to port function' at a particular address . There are two parameters : the First one is the Address and the Second one is the DATA to be sent. The parameter ranges are :

Address : 0x0000 [or &H0000 or 0000H] to
0x0FFF [or &H0FFF or 0FFFH] in Hexadecimal.
{12 bit Parameter}

DATA : 0x000 [or &H000 or 000H] to
0x0FF [or &H0FF or 0FFH] in Hexadecimal.
{ 8 bit Parameter}

3. unsigned char _stdcall inportb(unsigned short);

This function is the main 'input from port function' at a give port Address. There is only one parameter that is the address of the port. The DATA read is returned back by the function. The parameter range is:

Address : 0x0000 [or &H0000 or 0000H] to
0x0FFF [or &H0FFF or 0FFFH] in Hexadecimal.
{12 bit Parameter}

The returned value ranges are :

DATA : 0x000 [or &H000 or 000H] to
0x0FF [or &H0FF or 0FFH] in Hexadecimal.
{ 8 bit Parameter}

4. void _stdcall uninstalldriver();

This function is to uninstall the system Kernel Mode driver porttalk.sys from the system and deregister the service. This function is required to be executed for uninstall in case if the OS is an NT/2000/XP .

5. unsigned char _stdcall whatOS();

This Function return the Version of any MS OS in terms of numbers as follows :

{31 - win3.11},{35 - WinNT 3.5},{98 - win 98 or 95} ,
{40 - WinNT 4.0},{200 - Windows2000},{204 - WindowsXP}
,{255 - for any new OS like LongHorn} or { 0 - For any error}



- Set Active Project
- Add To Project
- Dependencies...
- Settings... Alt+F7**
- Export Makefile...
- Insert Project into Workspace...

BytesReturned

(Globals)

Workspace 'VcApp': 1

- VcApp files
 - Source Files
 - VcApp.cpp
 - Header Files
 - Directio.h
 - Resource Files
 - External Dependencies

nMain

```

...ows.h>
...owsx.h>
...h>
...ib.h>
...ng.h>
...o.h>
#include "directio.h"
int PASCAL WinMain( HINSTANCE hInst, HINSTANCE hPrevInst, LPSTR lpsz
                    int nCmd )
{
    outportb(0x378,0x00);
    return 0;
}

```

ClassView FileView

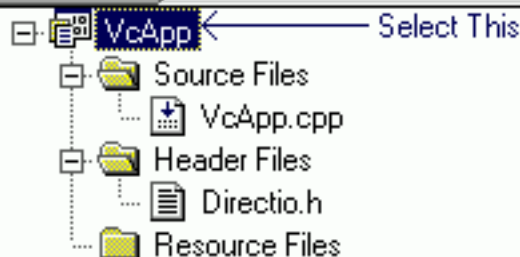
VcApp.exe - 0 error(s), 0 warning(s)

Build Debug Find in Files 1 Find in Files 2 Results

Project Settings



Settings For: Win32 Debug



Add the Entry For the Directio.lib in this box with a space in front and Accordingly the Project Options Will show the change.

General Debug C/C++ Link Resource

Category: General [Reset]

Output file name:
Debug/VcApp.exe

Object/library modules:
kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib ad

- Generate debug info
- Ignore all default libraries
- Link incrementally
- Generate mapfile
- Enable profiling

Project Options:
kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib
odbc32.lib odbccp32.lib directio.lib /nologo

Addition

OK Cancel