How to compile the open source version of Winrad

The distributed open source of Winrad is meant to be compiled with the Borland (now Embarcadero) compiler named CodeGear RAD Studio 2007 (also known as RAD Studio V5). There is the possibility that other compilers will suit, but I haven't tested any other compiler apart from the above. The fact that the visual widgets on the Winrad screen are implementations of VCL objects must be taken into account.

In the following notes I will assume that you have just installed from fresh the CodeGear compiler, on a disk of your choice, into a main directory named CodeGear.

The installation of the compiler will have produced a directory tree like this :
x:\CodeGear\RAD Studio\Projects where x is the letter of the disk you did choose for the installation.

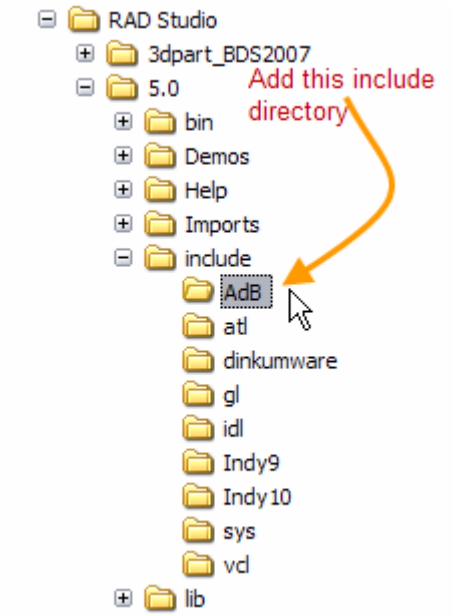
Probably the installer will try to convince you to put everything under the "Program Files" subdirectory. Let it learn who is the master in charge, and place the CodeGear directory in the root of the chosen hard disk.
If the subdirectory Projects is not there, create it as a subdirectory of "RAD Studio".

Please add a subdirectory of Projects named Winrad_OS (that stands for Winrad Open Source)
Then copy all the source code distributed with the first release into this Winrad_OS directory.
Copy the following files from the second release into this same directory :

- Winrad_OS.cbproj
- Winrad_OS.cbproj.local
- Winrad.res
- WinRadPX.lib

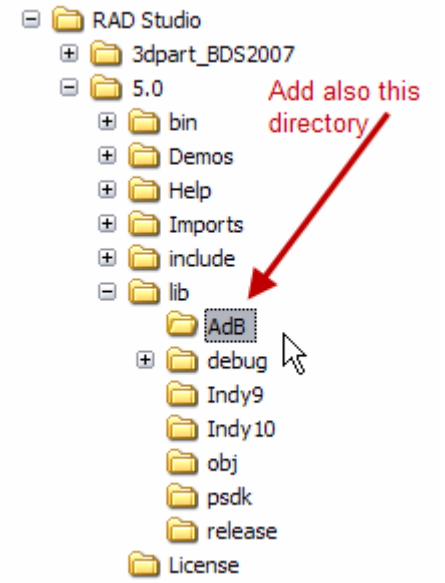
Now add to the Winrad_OS directory two subdirectories, named Release_Build and Debug_Build
Then copy the file WinRadPX.dll from the second release into both these two subdirectories.

At this point add a subdirectory named AdB to the include directory of the compiler, like this :



And copy into this subdirectory the contents of the folder named extra_includes of the distributed ZIP file of the second release.

Similarly, add a subdirectory named AdB to the lib directory of the compiler, like this :

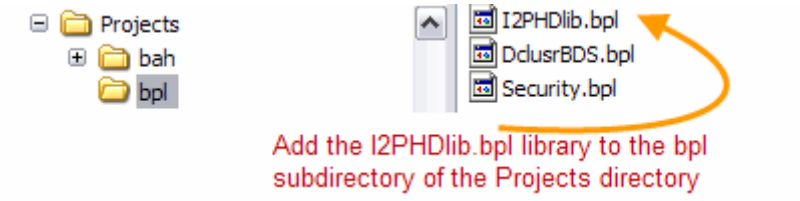


And copy into this subdirectory the contents of the folder named extra_lib of the distributed ZIP file of the second release.

If you have already done this with the previous release of the second part of the Winrad code, please do it again, as the contents of the extra_includes and the extra_lib files have changed.

Now create a subdirectory of the RAD Studio directory, and name it ASIO_SDK. Then copy into it the contents of the folder ASIO_SDK contained into the distributes ZIP file of the second release. Please keep intact the structure of the subdirectories in it.

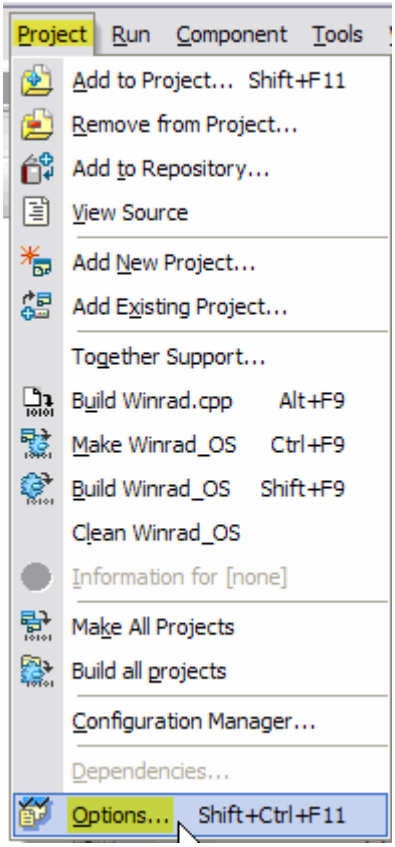
Check if the Projects directory has a subdirectory named bpl. If it doesn't, create it.
Next step, extract the file I2PHDlib.bpl from the distributed ZIP file, and place it into the bpl subdirectory of the Projects directory :



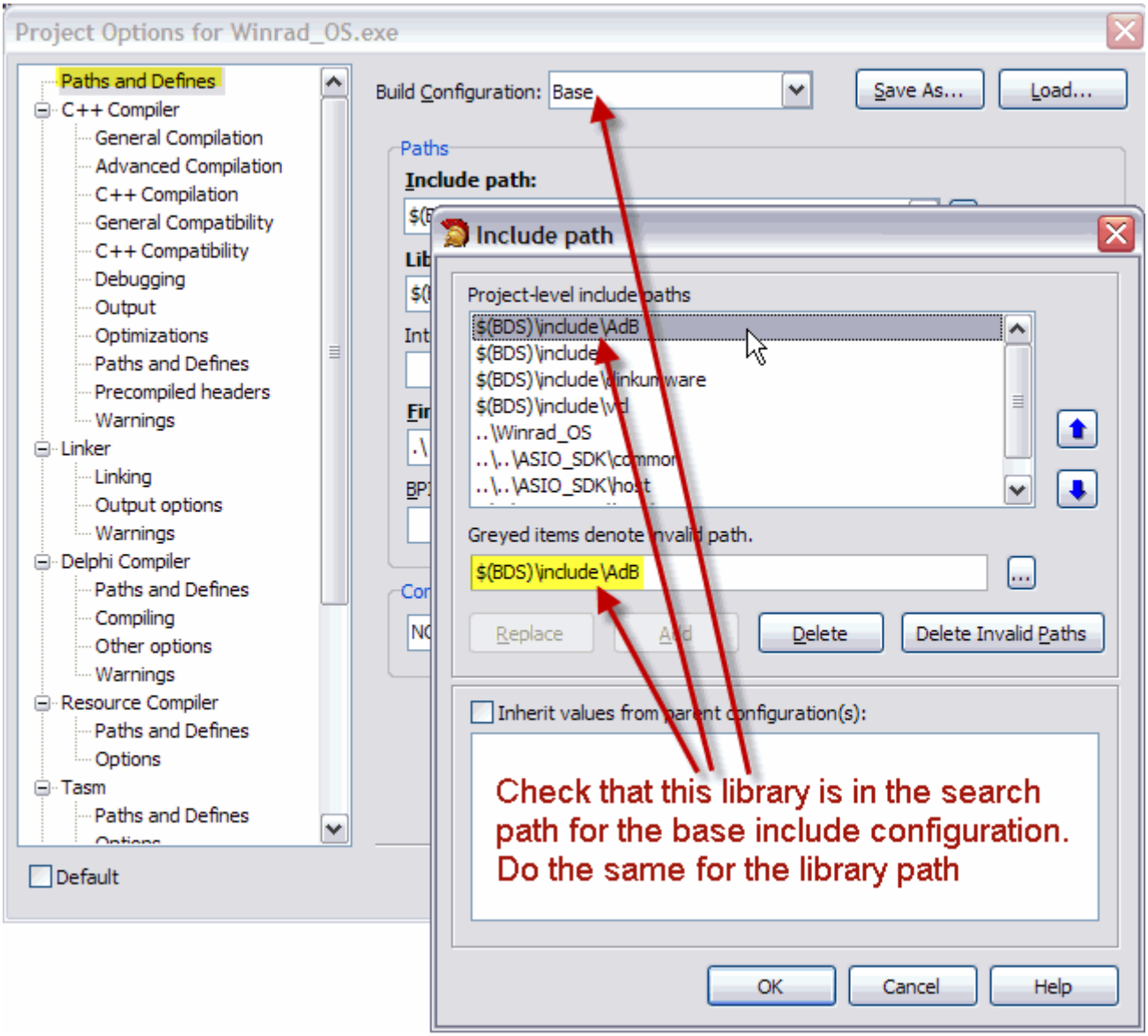
Well, at this point, if you haven't done any mistakes, and, more important, if **I** haven't done mistakes in describing the steps to be done, you should be ready for the first compilation.

Double click on the file Winrad_OS.cbproj in the directory Winrad_OS, and the compiler should start and open the source code of Winrad for inspections and/or modifications.

Now click on Project | Options in the action bar of the IDE :

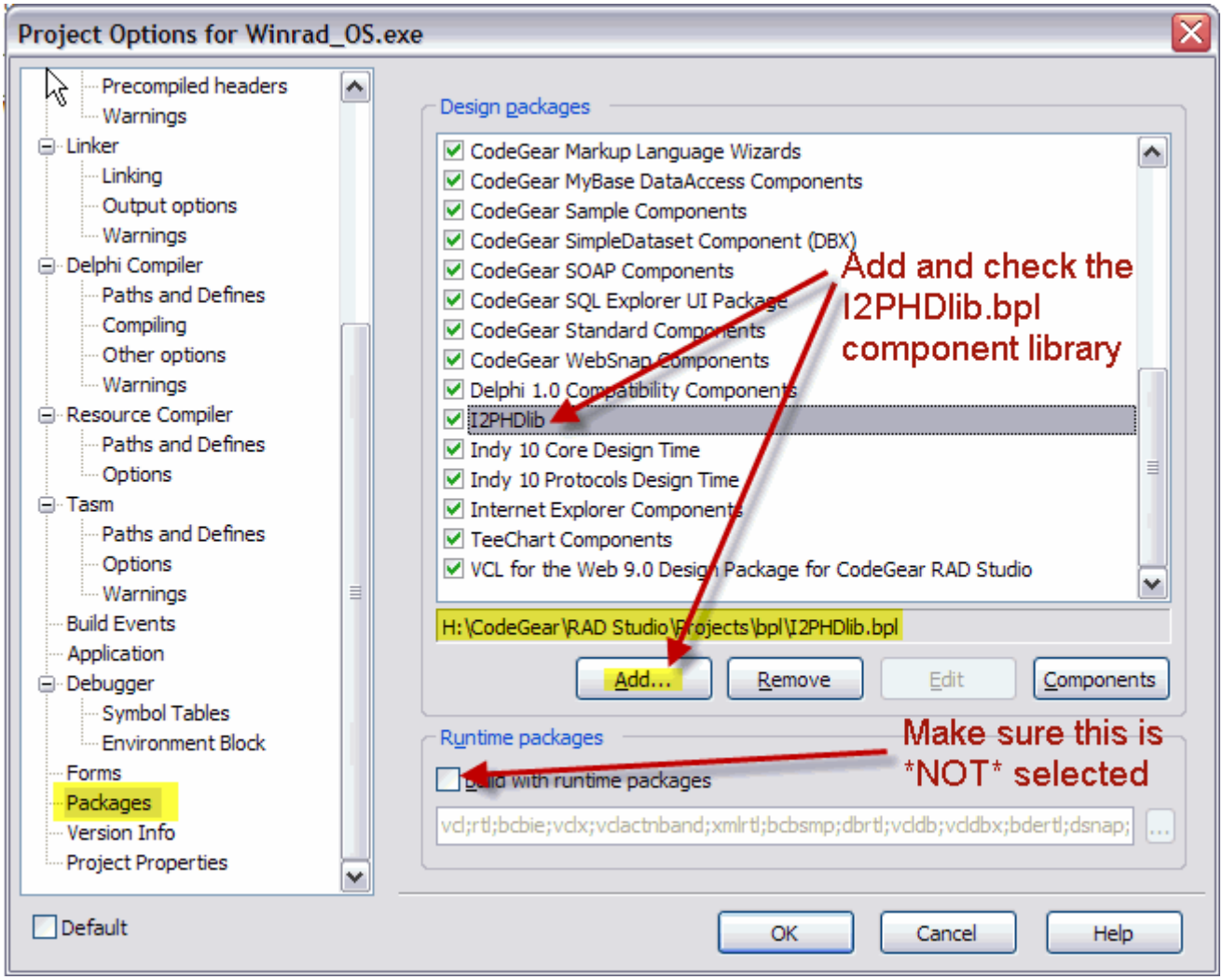


The pop down menu at the left should open, and clicking on Options will open the Options menu, where you will perform the operations described in the next page



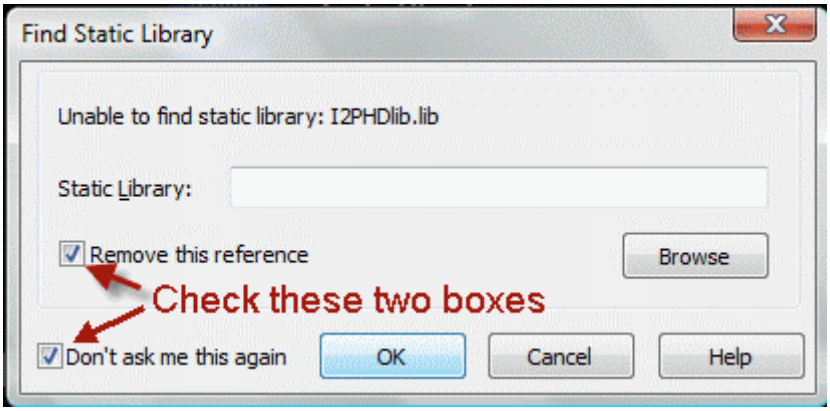
Make sure that the $(BDS)\include\AdB and the $(BDS)\lib\AdB are at the top of the include and the lib paths, in the “Base” build configuration (that will be inherited by both the Release and the Debug build configurations).

Then you must add the I2PHDlib.bpl library to the list of the component libraries used by the project,
With the Options menu still open, click on Packages and add this library (see next page)



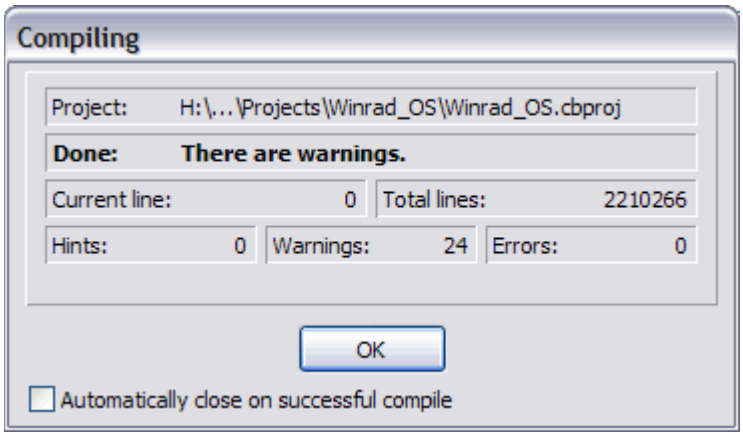
Probably in your installation the list of the libraries will be different from that shown here. The important thing is that the I2PHDlib is present and checked. Control also that the “Runtime packages” checkmark is not selected.

Now you can try your first compilation, clicking on the action bar on Project | Build Winrad_OS.
With almost 100% probability you will be greeted by this error.



Check the two boxes as shown in the figure, and try again to compile. This time the error should not appear.
I still have to understand why it looks for that lib file, which is not present on my PC, and not referenced in the package file (and actually not needed).

Now, retrying the compilation, after some time, depending on how fast is your computer, you should hopefully see something like this



Disregard the 24 warnings, they are not important. At this point, if you haven't changed anything from the distributed package, you should find into the Release_Build subdirectory the just compiled Winrad executable, ready to be launched. Make sure you see there also the WinRadPX.dll file that you should have placed there in a previous step.

If you change the options of the compiler, and produce a debug version of the executable, that can accept breakpoints and step-by-step execution, you will find it into the Debug_Build directory.

If something goes wrong, probably I did forget some steps in this description, so please contact me directly or through the Winrad Yahoo group.

Good luck and enjoy yourself !

73 Alberto I2PHD