

## Using SIGIL with MinGW in CodeBlocks

Setting up a CodeBlocks project with SIGIL is very straightforward. First, create an **Empty project** in the Project Wizard.

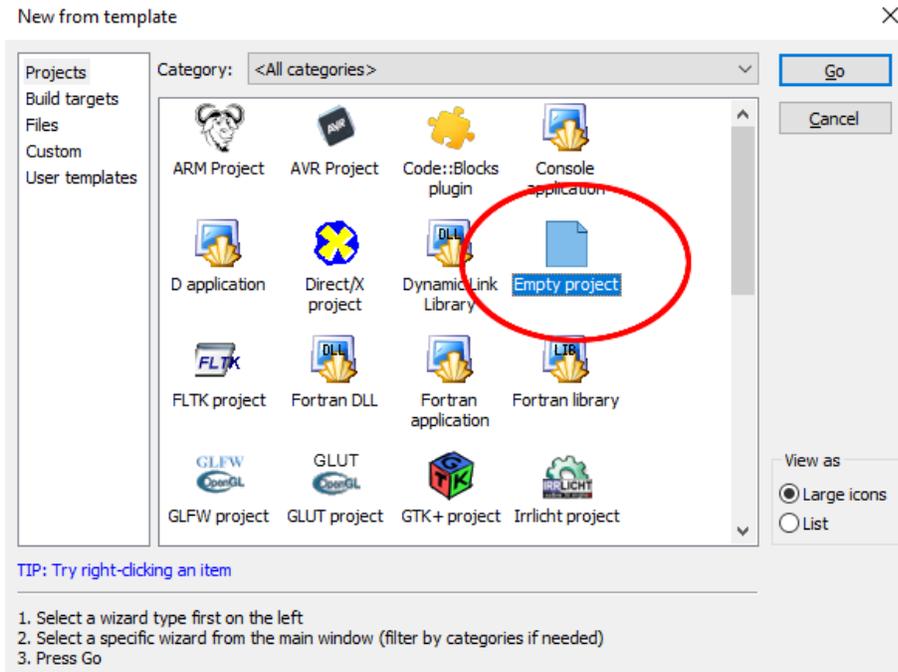


Figure 1. Select **Empty project** from the Project Wizard.

You will need to copy the relevant SIGIL files and libraries to your project folder so that CodeBlocks can find them. There are several different ways to do this, but this guide assumes that you want to keep your libraries local to your project. Copy the files inside the **sigil-mingw32** folder and the **include** folder itself to your project location like this.

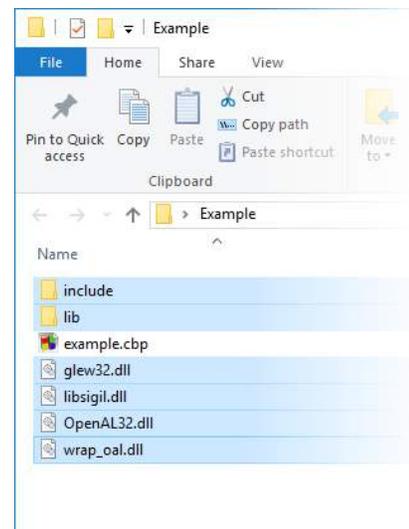
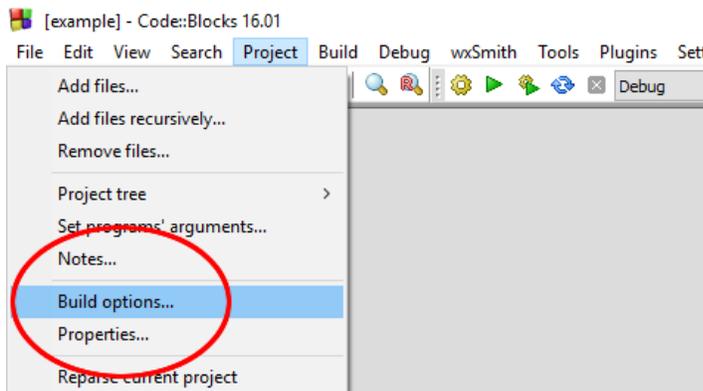


Figure 2. The **include** folder should contain **sl.h** and the **lib** folder should contain **libsigil.dll.a**.



Next, you need to tell your CodeBlocks project where to find these files. Access the **Build Options** for your project in the drop-down menu.

Figure 3. **Project -> Build options** will allow you to set linker and compiler search directories necessary for working with SIGIL.

In the **Build Options** window, select the name of your project (**example** in the picture below) from the list on the left-hand side. This will ensure that the options we input will apply to all build configurations for this project, not just the Debug or Release configurations.

You need to tell your CodeBlocks project file where to find the SIGIL library. Select the **Linker settings** tab, and click Add. Enter **lib\libsigil.dll.a** into the new dialog box and click **OK**. This will allow your project to link correctly to SIGIL. The **Project build options** window should now look like this.

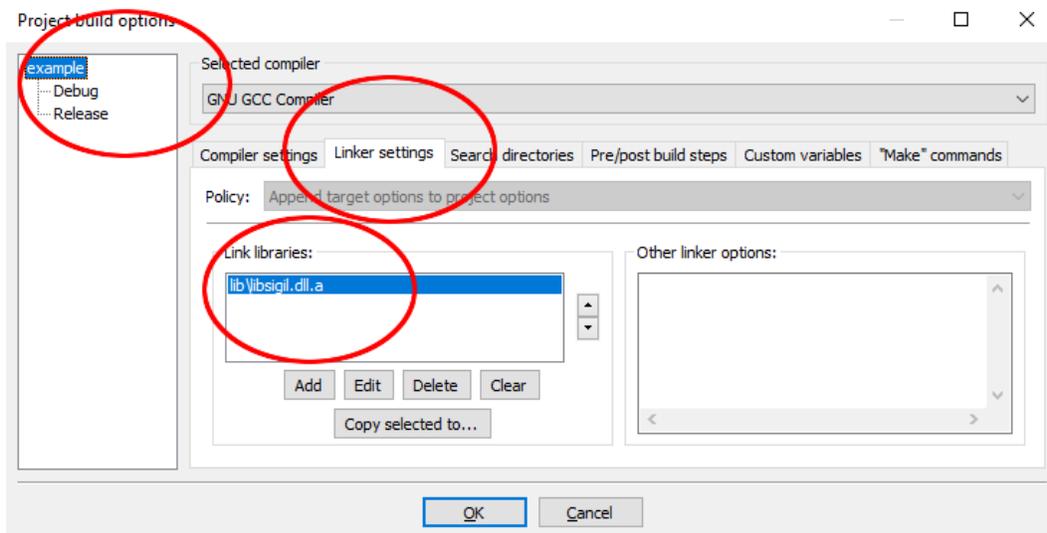


Figure 4. Explicitly tell CodeBlocks to link your project against SIGIL for all build configurations.

Lastly, you need to tell your CodeBlocks project file where to find the SIGIL header file. Select the **Search directories** tab and click **Add**. Enter **include** into the new dialog box and click **OK**. This will tell your project, when looking for **sl.h**, to look in the **include** folder we just created. The **Project build options** window should now look like this.

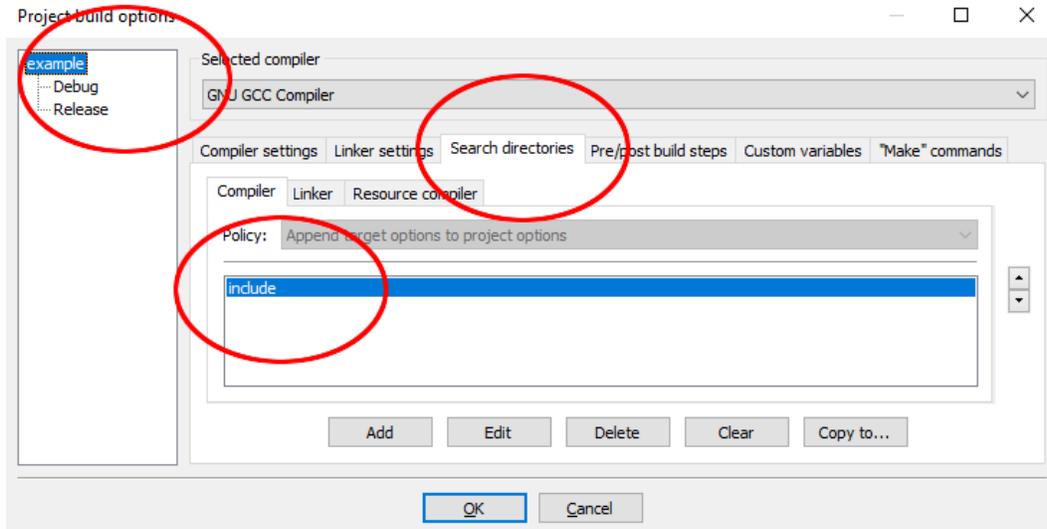


Figure 5. Add the local **include** folder to your project so CodeBlocks can find it when compiling your project.

Now, click **OK**. Save your project with **File -> Save project**. You're all set, and can start developing your SIGIL project.