

Compiling visacmd

From TestWiki

Contents

- 1 prerequisites
- 2 installation
 - 2.1 mingw
 - 2.2 java
 - 2.3 eclipse
 - 2.4 tekvisa
- 3 creating a project
 - 3.1 configure to use visa development files
- 4 moving forward
- 5 resources

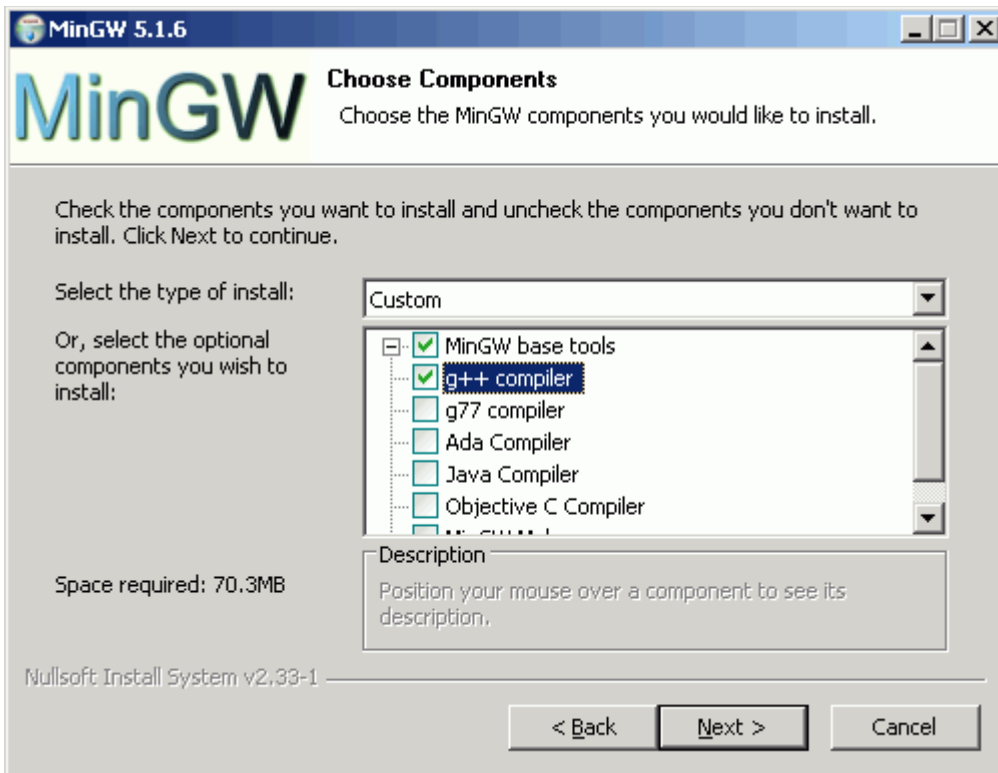
prerequisites

- mingw (v5.1.6) - <http://sourceforge.net/projects/mingw/files/>
- java (v1.6.0_13 aka version 6 update 13) - <http://java.com/en/download/manual.jsp>
- eclipse (v6.0.2) - <http://www.eclipse.org/downloads/>
- visa communications driver with development support files (TekVISA v3.3.4) - <http://www2.tek.com/search?q=tekvisa&hl=en&lr=&access=p&sort=date:D:S:d1&partialfields=document-subtype:SUT&filter=0>

installation

mingw

follow the prompts and use the default values. when prompted to choose componets, only the g++ compiler is required.



java

you may already have a java runtime environment. to check open a command prompt window and enter the following command

```
java -version
```

if your pc is currently configured to use java, you should see version information

```
C:\>java -version
java version "1.6.0_13"
Java(TM) SE Runtime Environment (build 1.6.0_13-b03)
Java HotSpot(TM) Client VM (build 11.3-b02, mixed mode, sharing)
```

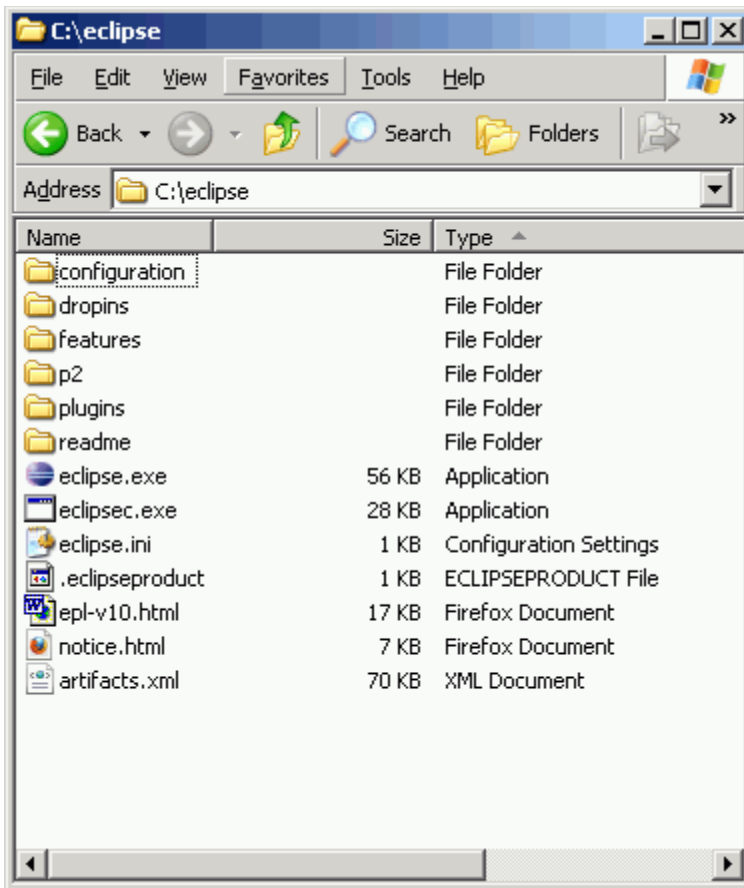
if not, you'll see something like

```
C:\>java -version
'java' is not recognized as an internal or external command,
operable program or batch file.
```

download the latest version of the Java Runtime Environment (JRE) and install it by following the prompts and using the default values. Java is required for Eclipse.

eclipse

eclipse does not install. simply extract the archive to where you would like to keep it. for this tutorial i will place the eclipse files in the root directly of drive C:.



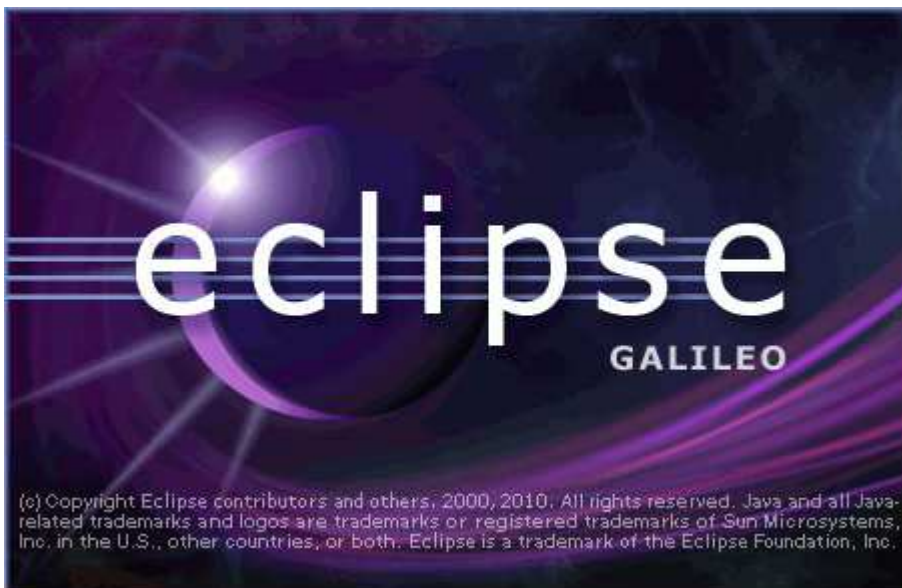
tekvisa

download and launch the tekvisa installer. follow the prompts and reboot when prompted.

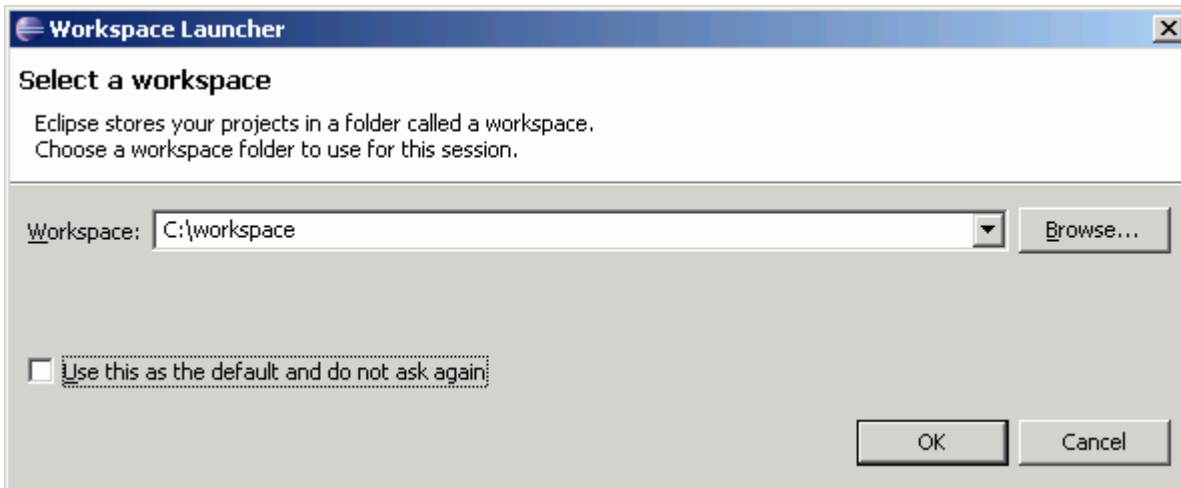
option test connectivity with the instrument to verify operation.

creating a project

launch eclipse.



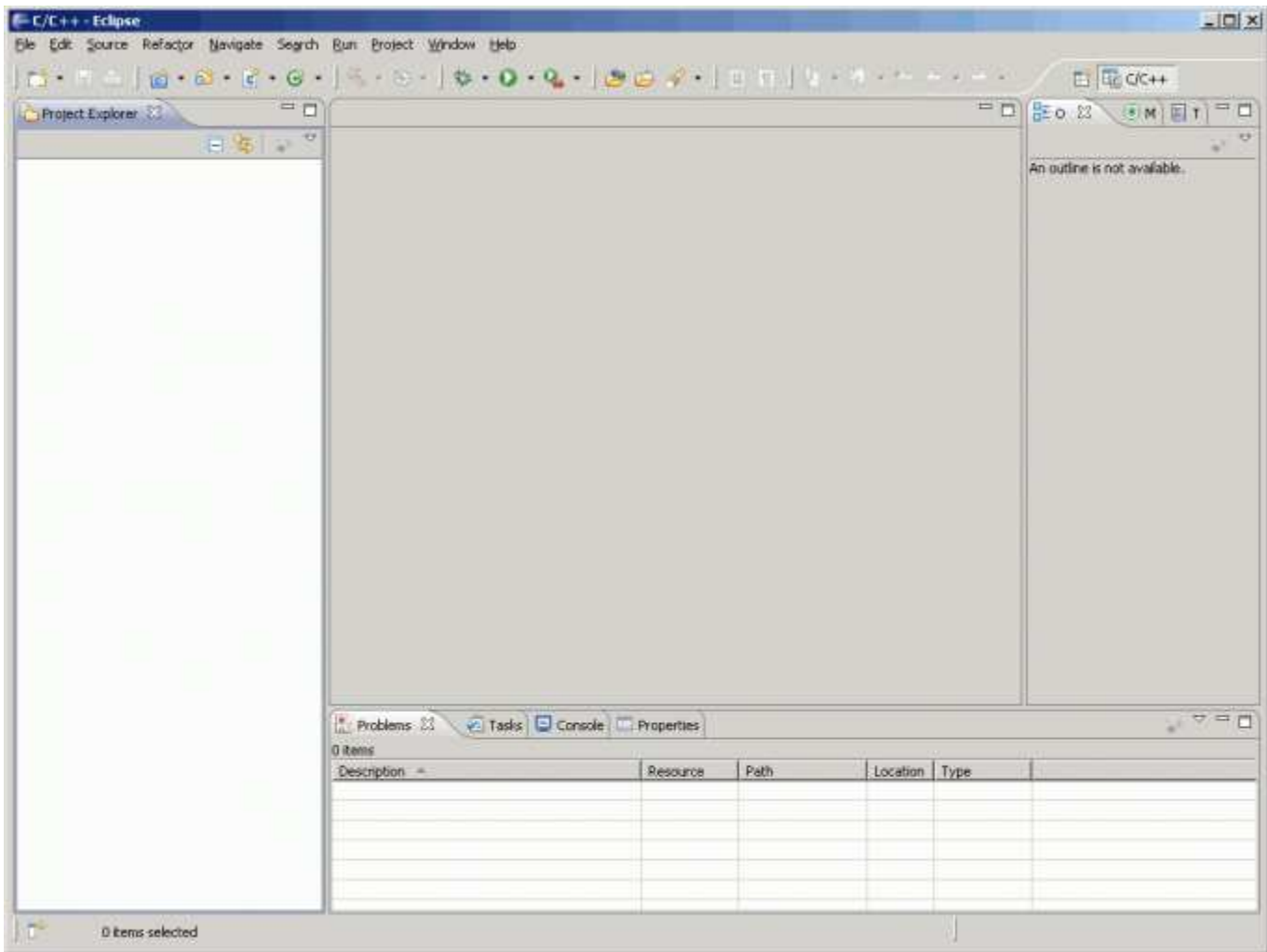
you may be prompted for a workspace directory (location to keep all your programming projects). for this example i will use **C:\workspace**.



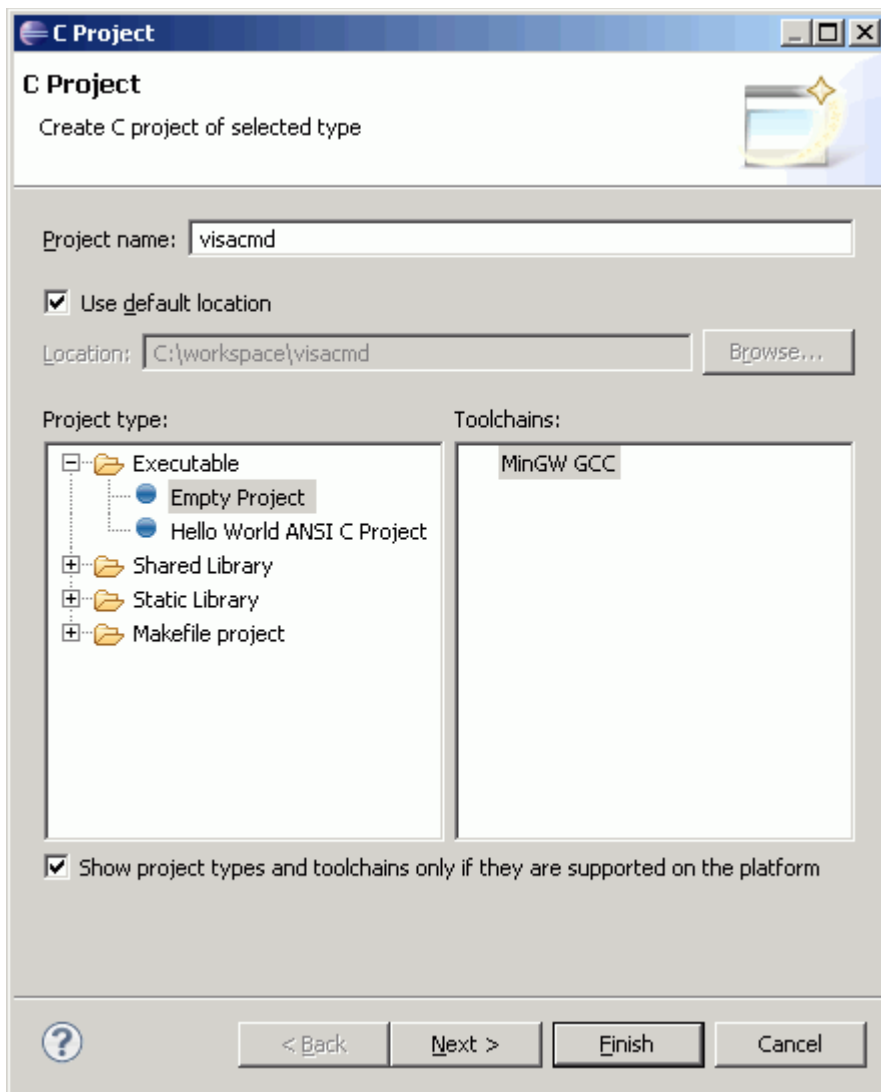
after entering in a workspace directory, you will be greeted with a welcome screen.



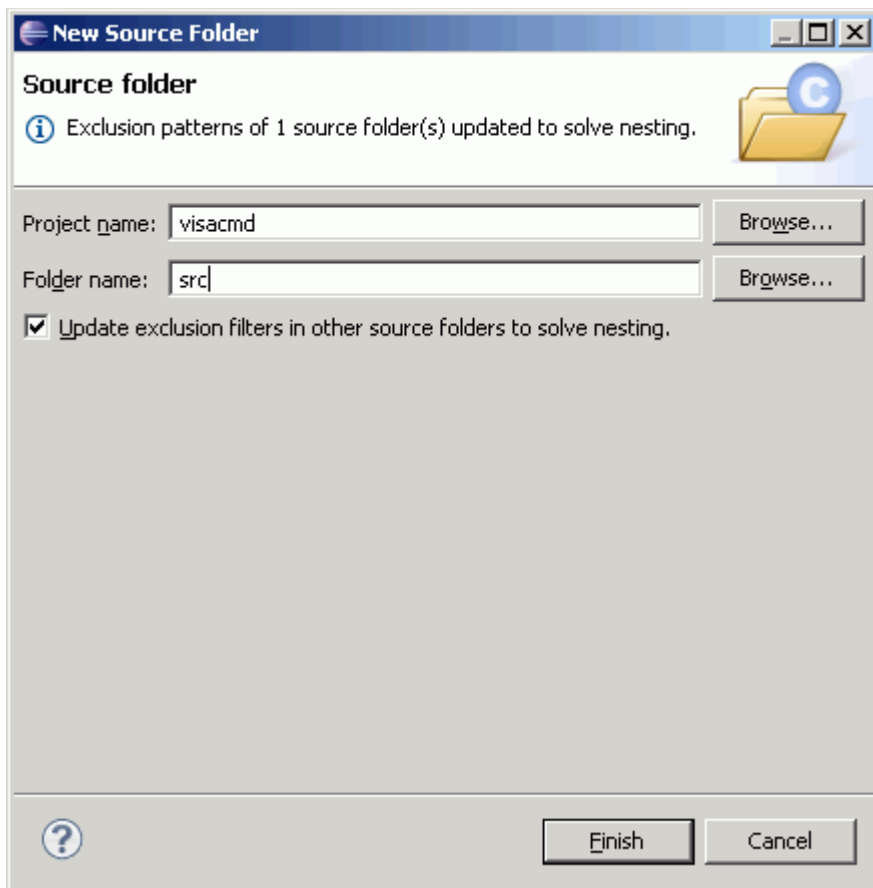
close the welcome tab. you will be dropped right into the main screen of eclipse.



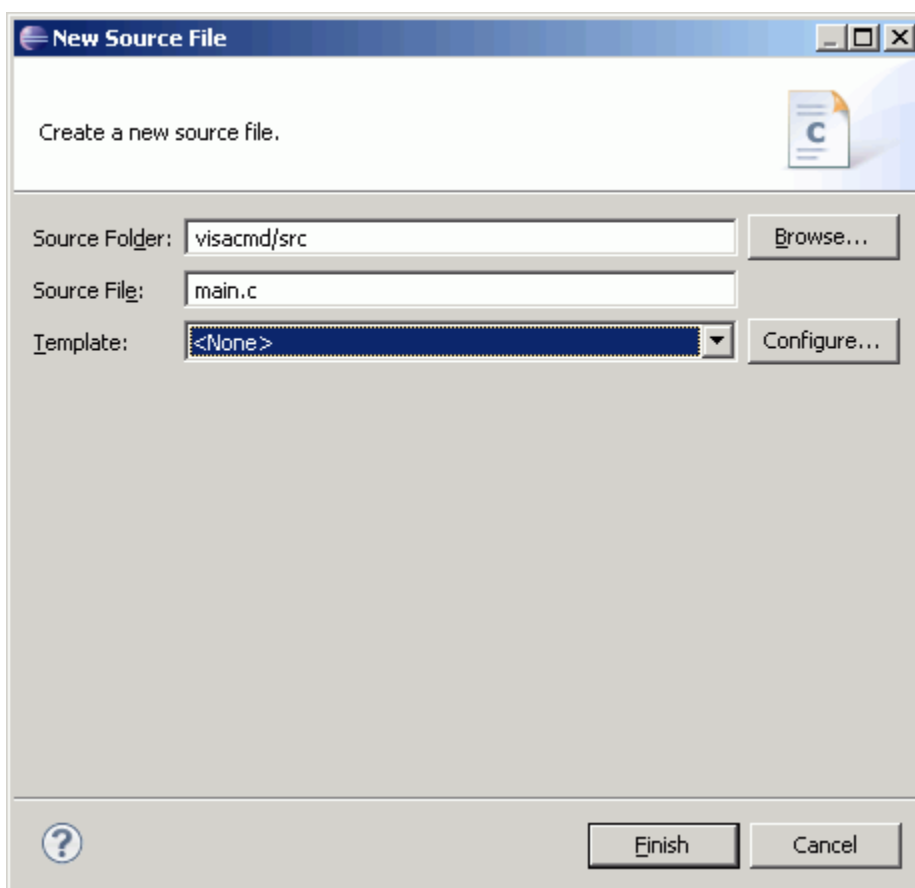
From here you will create a new project. In the menu bar at the top, click on File, New, C Project. The **C Project** window will appear. In this window under the **Project name:** field, enter **visacmd**. From the **Project type:** list, select **Empty Project** from the **Executable** sub-section. If MinGW was installed properly, Eclipse should automatically detect it and list it within the **Toolchains:** list. If more then one element is available, select **MinGW GCC**. Click **Finish**.



Now create a source folder for your new project. In the menu bar at the top, click on File, New, Source Folder. Enter **src** as the **Folder name:** and click **Finish**.

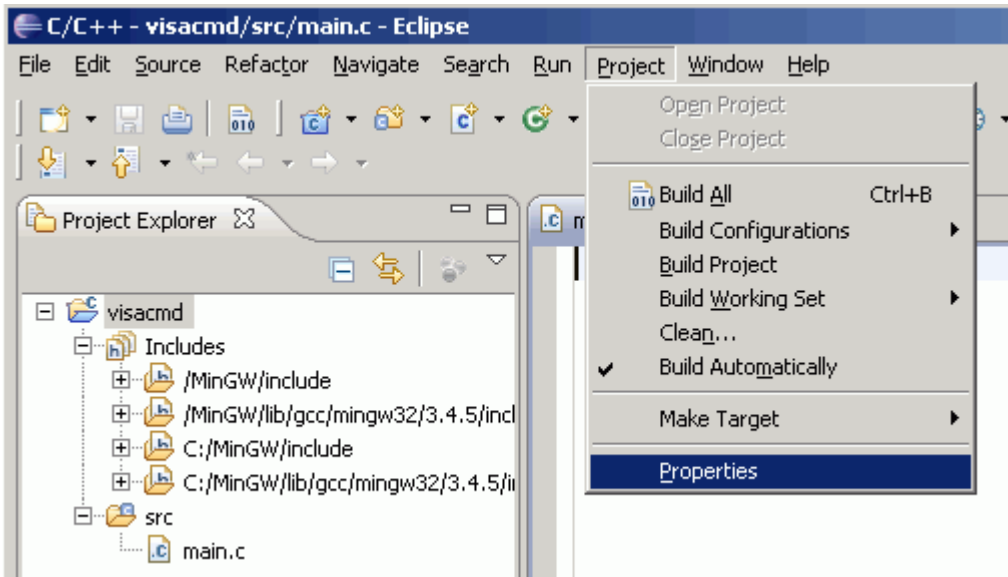


Now create a new source file. In the menu bar at the top, click on File, New, Source File. Enter **main.c** as the **Source File:**. The **Source Folder:** should be the folder you created in the previous step.

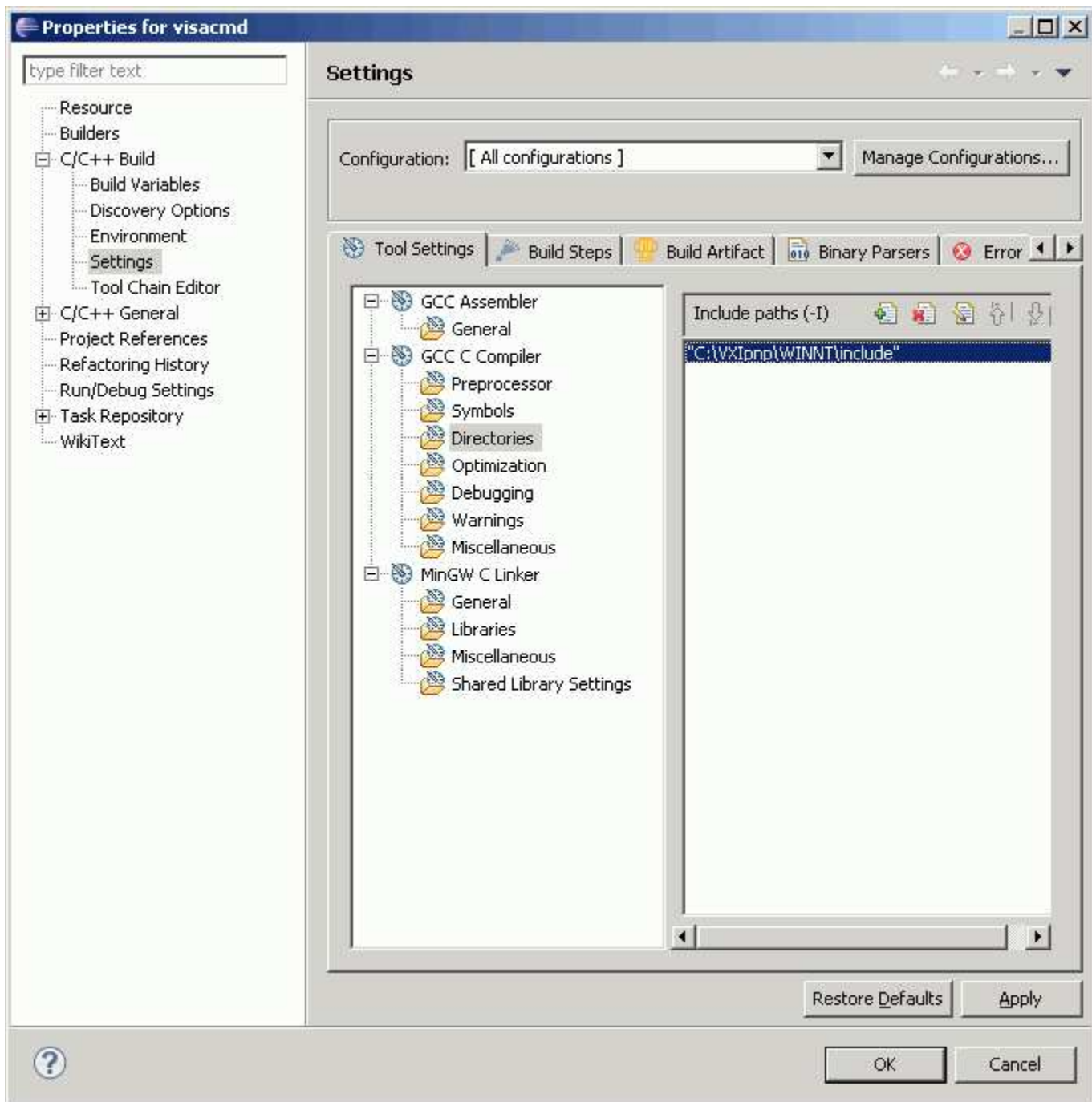


configure to use visa development files

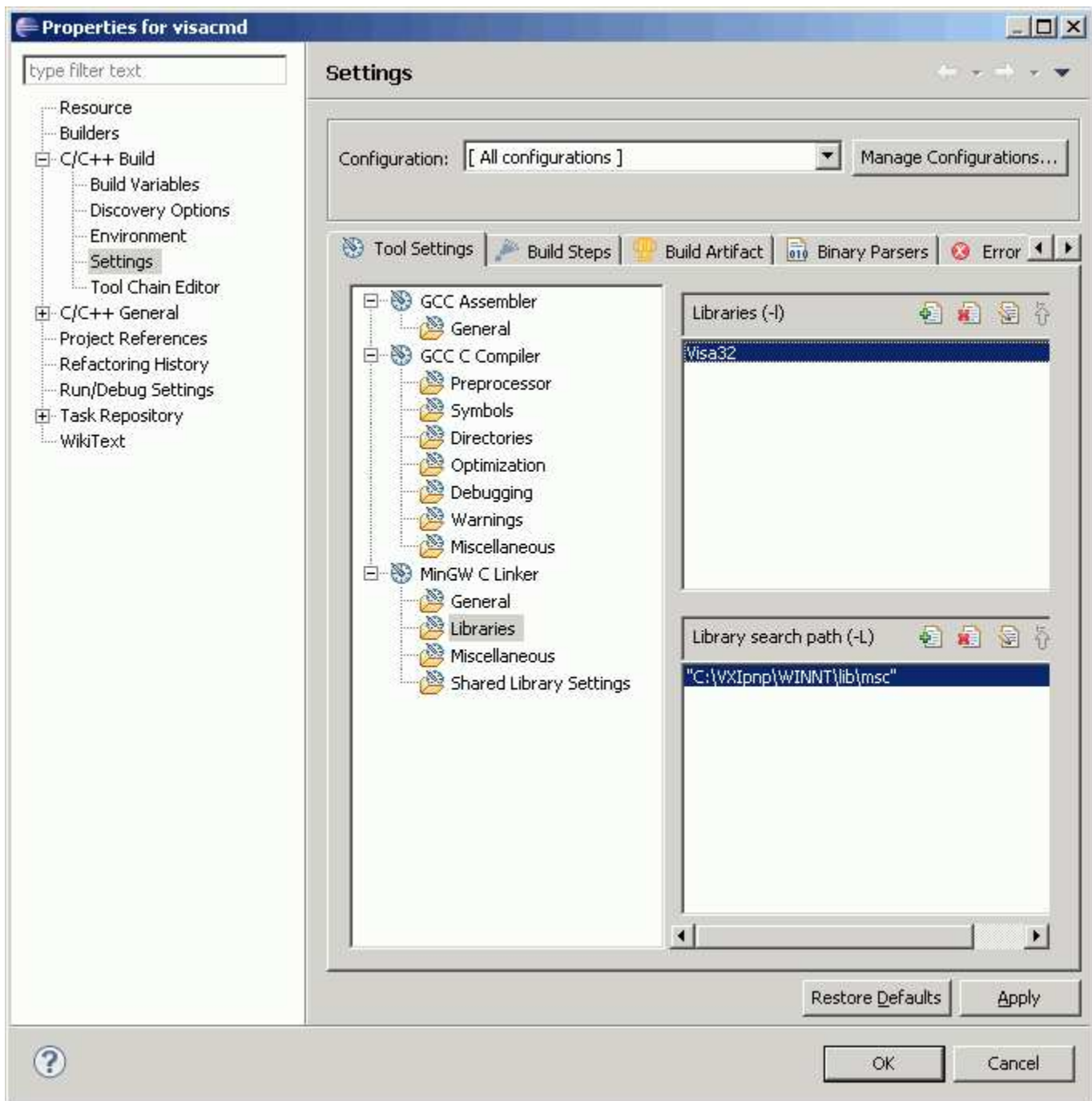
With the source folder and source file created, you can now configure the build settings with the location of the header files and import libraries. In the menu bar at the top, click on Project, Properties.



Change the **Configuration:** pull-down menu to [**All configurations**]. Expand the **C/C++ Build** menu on the left and select **Settings**. Under the **Tool Settings** tab, expand the **GCC C Compiler** menu and select **Directories**. Add the path **C:\VXIpn\WINNT\include** to the **Include paths** list.

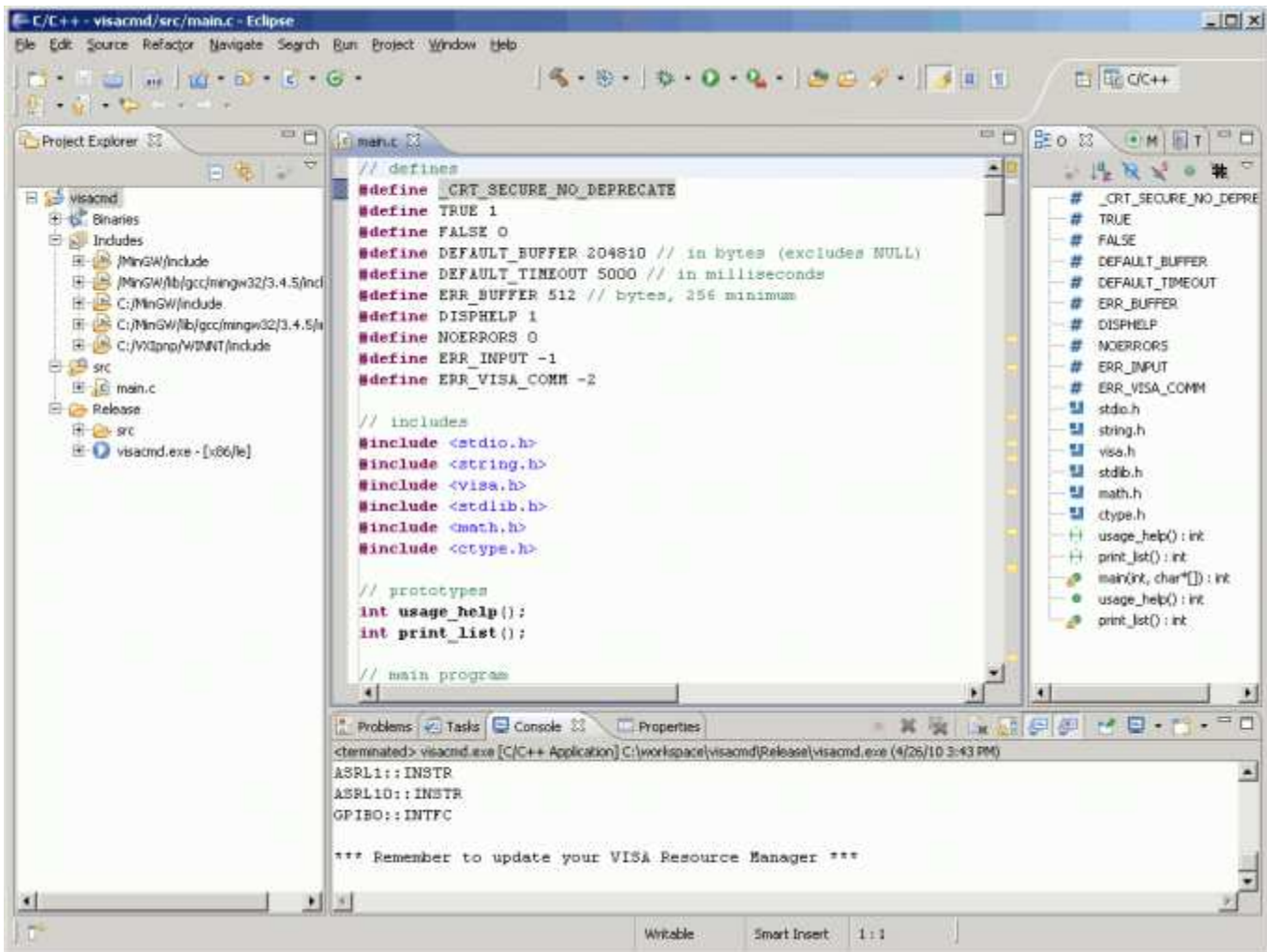


Remaining on the **Tool Settings** tab, expand the **MinGW C Linker** menu and select **Libraries**. Add **Visa32** to the **Libraries** list and add the path **C:\VXIppn\WINNT\lib\msc** to the **Library search path** list. Click **OK** to commit these changes.



Eclipse is now properly configured to create compiled C programs that will interface with VISA (and any instrument that supports VISA). You can now paste in example code into **main.c** and compile it. Here I pasted the source code for visacmd into main.c. * To build (aka compile) the project: In the menu bar at the top, click on Project, Build Project. Results of the build are printed to the **Console** tab at the bottom of the Eclipse main window.

- To launch the resulting executable within Eclipse: In the menu bar at the top, click on File, New, Source Folder. Output from the program is printed to the **Console** tab at the bottom of of the Eclipse main window.



moving forward

Configuring an IDE properly so that your project will use the VISA development support files included with TekVISA is often the biggest barrier in learning how to create remote instrument programs. Be sure to check the documentation for your IDE if you aren't using Eclipse or the same version of Eclipse. Similar steps can be applied to Microsoft Visual Studio as well or other vendors' VISA development support files.

resources

- TekVISA Programmer Manual. Information on all of the VISA functions and how they work.
- Model Specific Programmer Manual. Information on all of the commands available for your instrument. Be sure to read the waveform data format chapter and the appendix on synchronization.
- C/C++ documentation available on the web (<http://www.google.com/search?q=c++programming>) or in print (<http://oreilly.com/pub/topic/cprog>).
- software documentation: MinGW and Eclipse have documentation available freely on the web.
 - <http://www.mingw.org/wiki/MinGWWiki>
 - <http://www.eclipse.org/resources/>

Retrieved from "http://localhost/mediawiki/index.php/Compiling_visacmd"

Category: Aeware

- This page was last modified on 6 May 2010, at 21:09.