

Configuring **SystemC 2.3.3** development environment on Windows-10 with **cygwin** and **eclipse**

COE838: System-on-Chip Design

This document is adapted mainly from a post on [stackoverflow](#) and the blog by Cristian Slav as given below.

1. <http://cfs-vision.com/2017/06/17/learning-systemc-000-learning-materials-and-initial-setup/>
2. <https://stackoverflow.com/questions/69845570/how-to-integrate-eclipse-systemc-2-3-3-and-cygwin-on-windows>

Main steps to integrate eclipse, systemc-2.3.3, and cygwin on a windows-10 operating system.

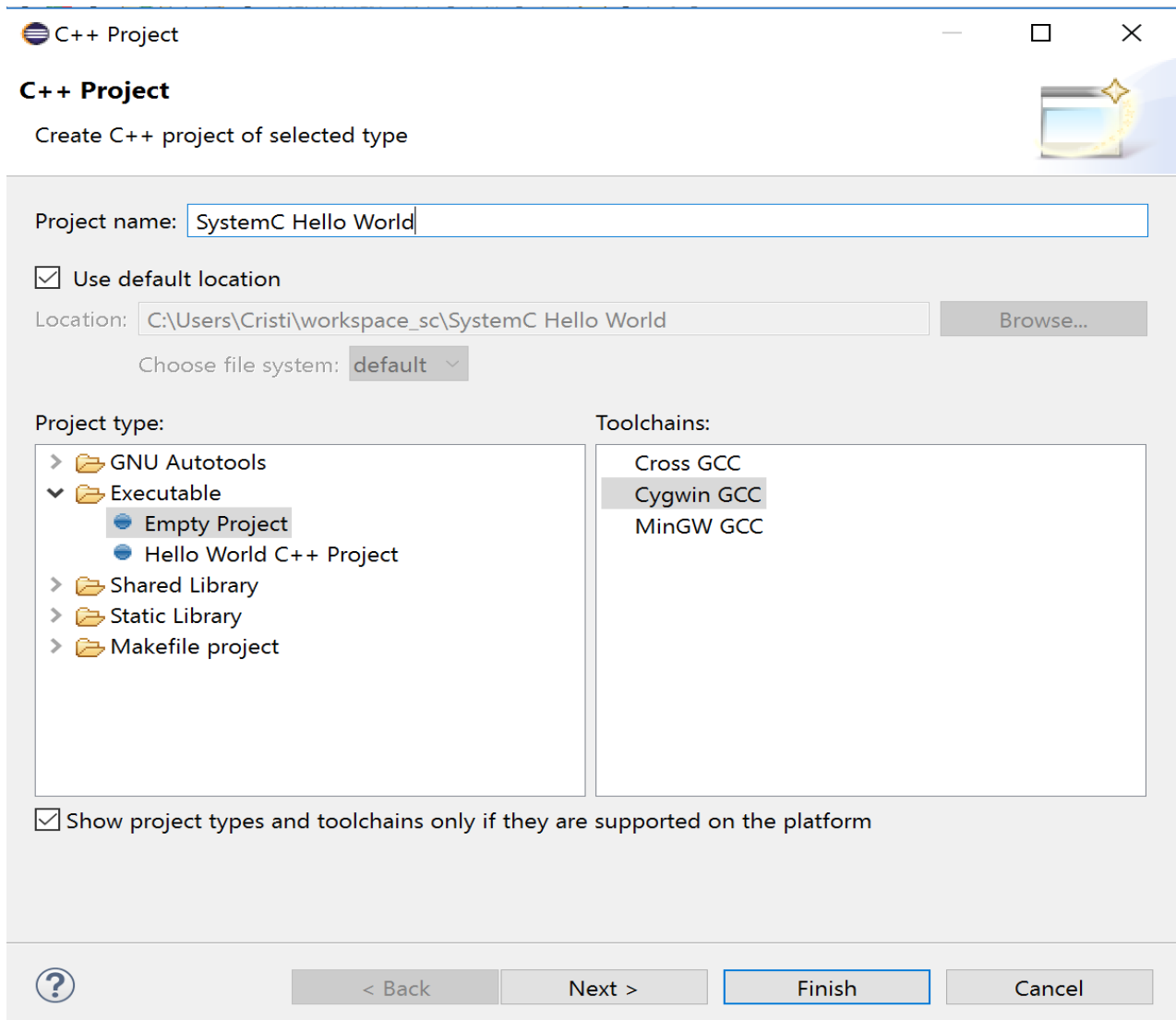
1. cygwin: you can download and install it from <https://cygwin.com/install.html>. Make sure to include the following packages as given in Figure 1 at the end of the document. [enter image description here](#). We installed cygwin in C:\cygwin64. directory.

autoconf	13-1	Keep	▼
automake	11-1	Keep	▼
cmake	3.20.0-1	Keep	▼
cygwin32-gettext	0.19.5.1-1	Keep	▼
gcc-core	11.2.0-1	Keep	▼
gcc-g++	11.2.0-1	Keep	▼
gcc-objc++	11.2.0-1	Keep	▼
make	4.3-1	Keep	▼
mingw64-x86_64-gcc-g++	11.2.0-1	Keep	▼
mingw64-x86_64-gcc-objc	11.2.0-1	Keep	▼
mingw64-x86_64-gettext	0.21-1	Keep	▼
mingw64-x86_64-libs-g++2.0	2.10.0-1	Keep	▼

Figure 1

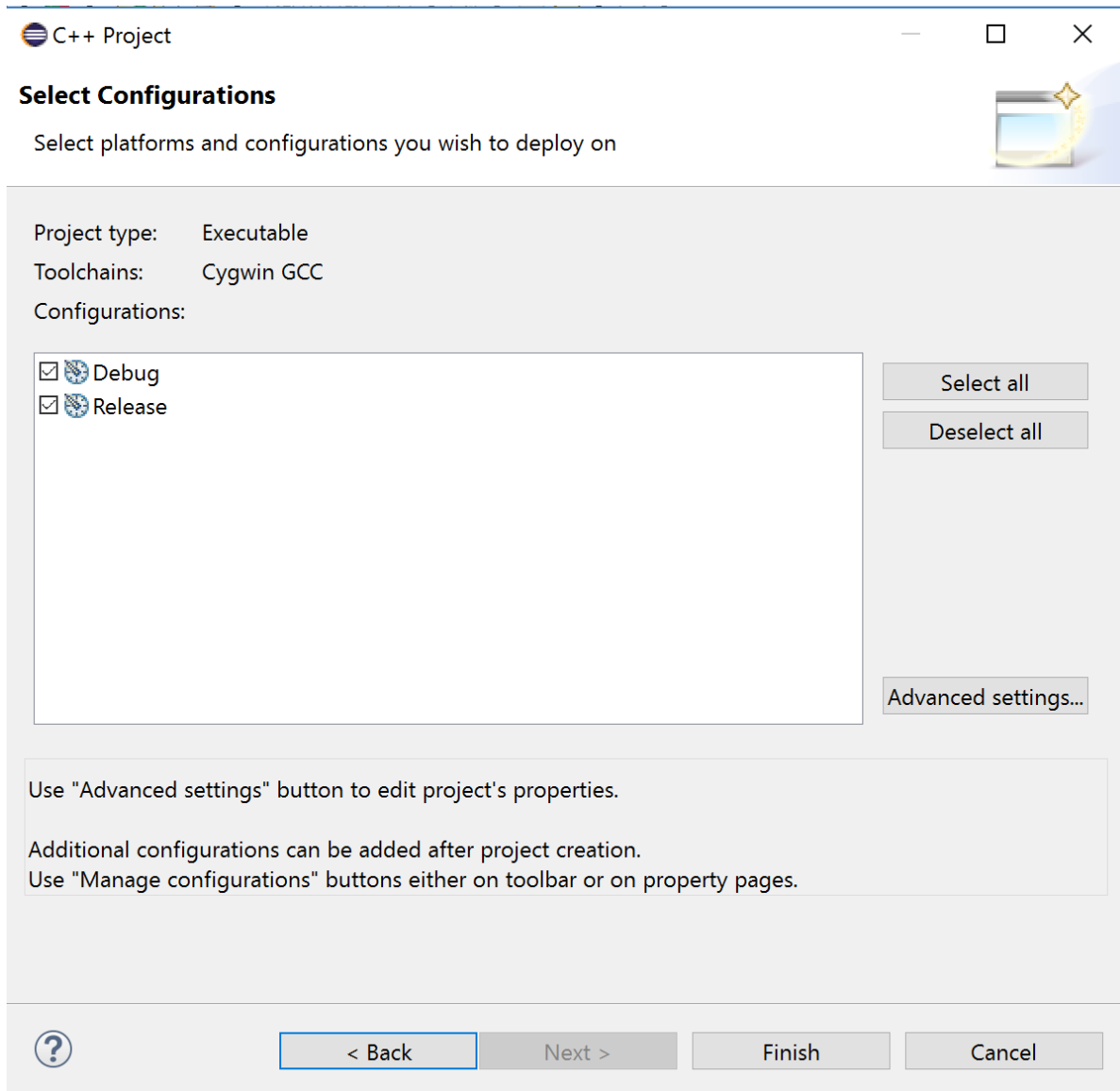
2. systemc-2.3.3: you can download the zip file from <https://www.accellera.org/downloads/standards/systemc> and extract the folder in your local disk (e.g., c:\systemc2.3.3).
3. Please read the content of the INSTALL file in the systemc2.3.3 extracted folder.
4. Open CygWin64 terminal.
5. Navigate to the folder you have extracted the systemc zip file.
6. Create temporary directory "objdir" via "mkdir objdir" as explained in the INSTALL file.
7. Change to the temporary directory via "cd objdir" as explained in the INSTALL file.
8. Choose your compiler export CXX="g++ -std=c++14".
9. (Optional Step avoid it for your first-time setup) If you would like to include fixed-point library you need to add DSC_INCLUDE_FX flag to the compiler definition, i.e., export CXX="g++ -std=c++14 -DSC_INCLUDE_FX".
10. run configure file as "../configure --prefix="location of the libs", e.g., "/sysclibs"
11. After MakeFile is successfully created run it via "make"

12. After the run is successful execute "make install". At this point you have created the system c libraries in "cygwin64/sysclibs".
13. Make sure to set PATH "C:\cygwin\bin" in the windows system environment variable.
You can use command prompt admin and setx /m PATH "c:\cygwin64\bin"
14. Install eclipse for C/C++ development <https://www.eclipse.org/downloads/>
15. Then move to **Creating SystemC Applications in Eclipse.**
16. Now open eclipse and create a new C++ project (do not create a C/C++ project). Start with a new project and choose C++ project. Then select "Cygwin GCC" in the Toolchains as shown below.



New SystemC Empty project in Eclipse

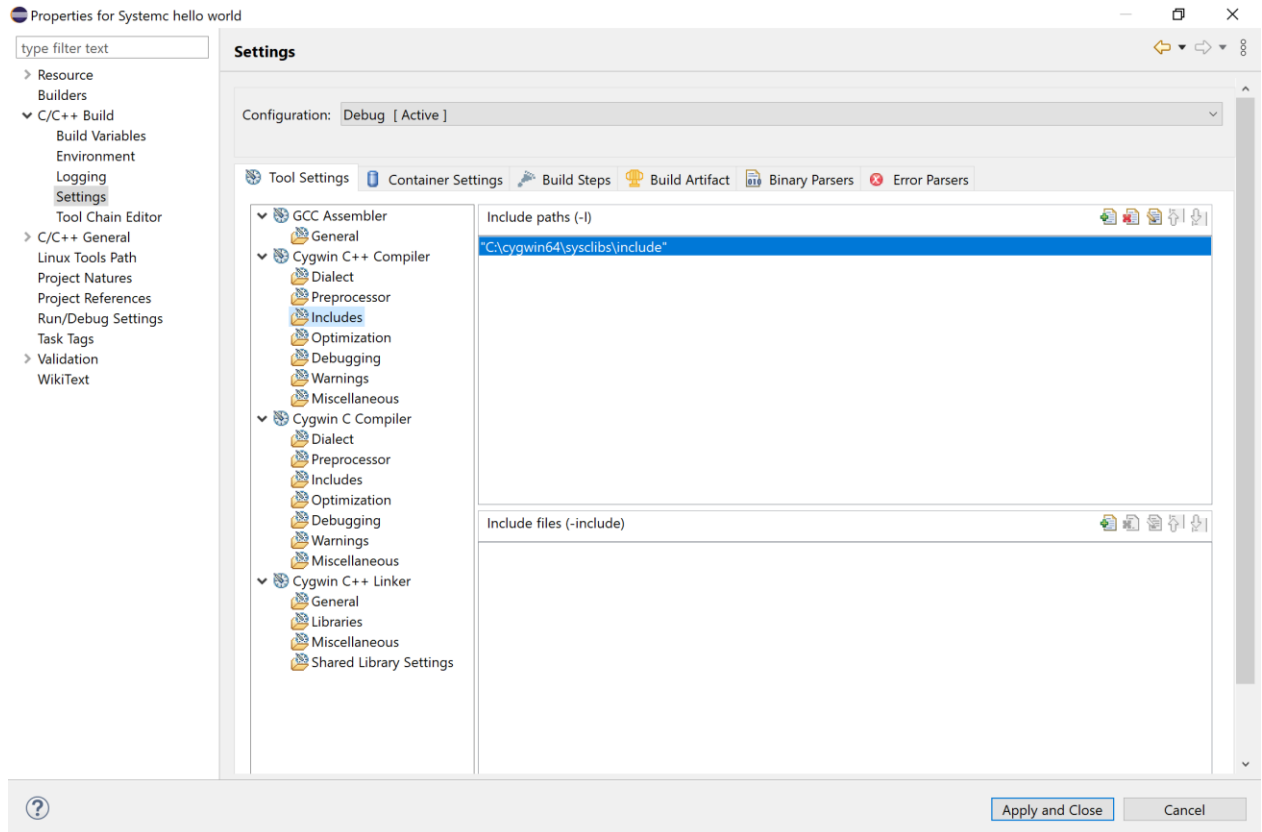
17. Click: **Next >** and then select the Advanced Settings shown in the Figure of next page.



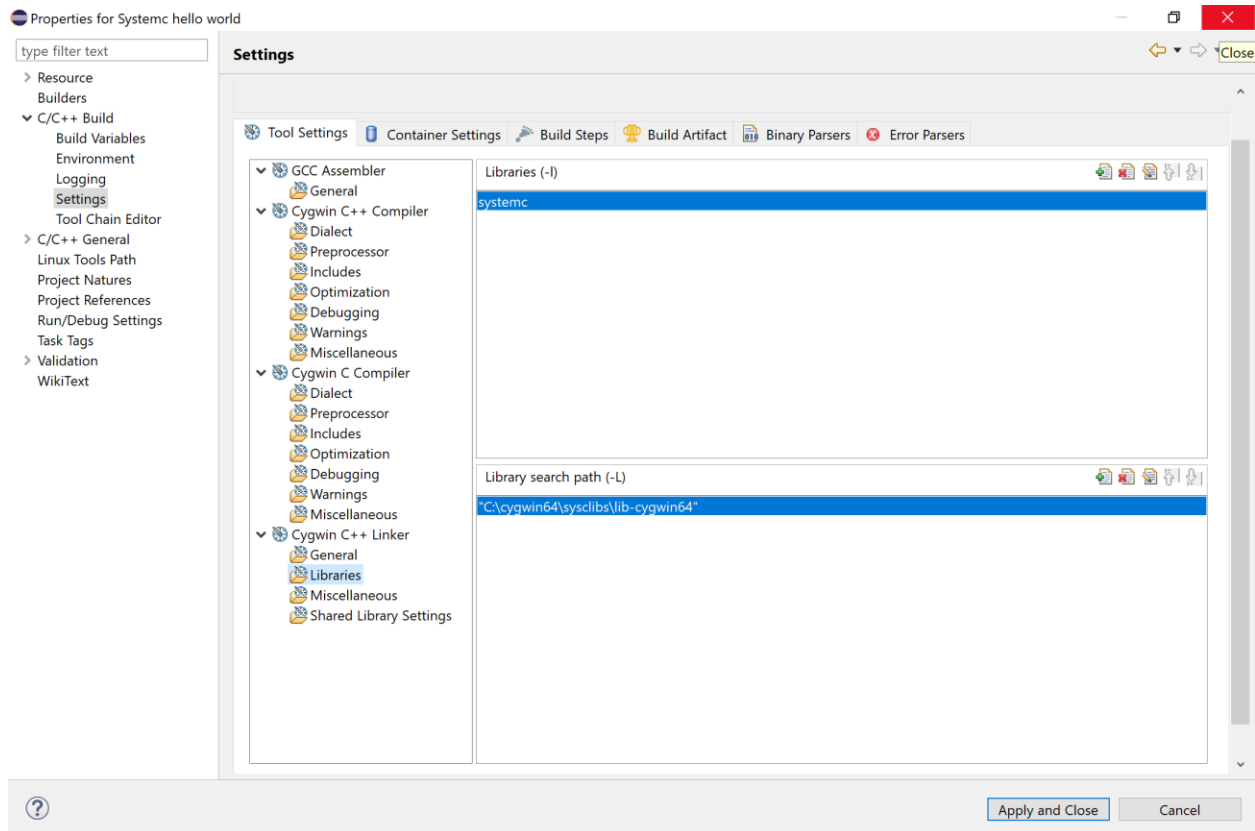
Select Advanced Settings

18. Select Advanced Settings in the above window. **Go to "project properties"** as shown on the next page. Under "**C/C++ Build**" menu select "Tool Settings" and first select the "Cygwin C++ Compiler". In other words, **C/C++ Build > Settings > Tool Settings > Cygwin C++ Compiler > Includes i.e. Include Paths –I**

19. Add the path **C:\cygwin64\sysclibs\include** as shown in the Figure on next page.

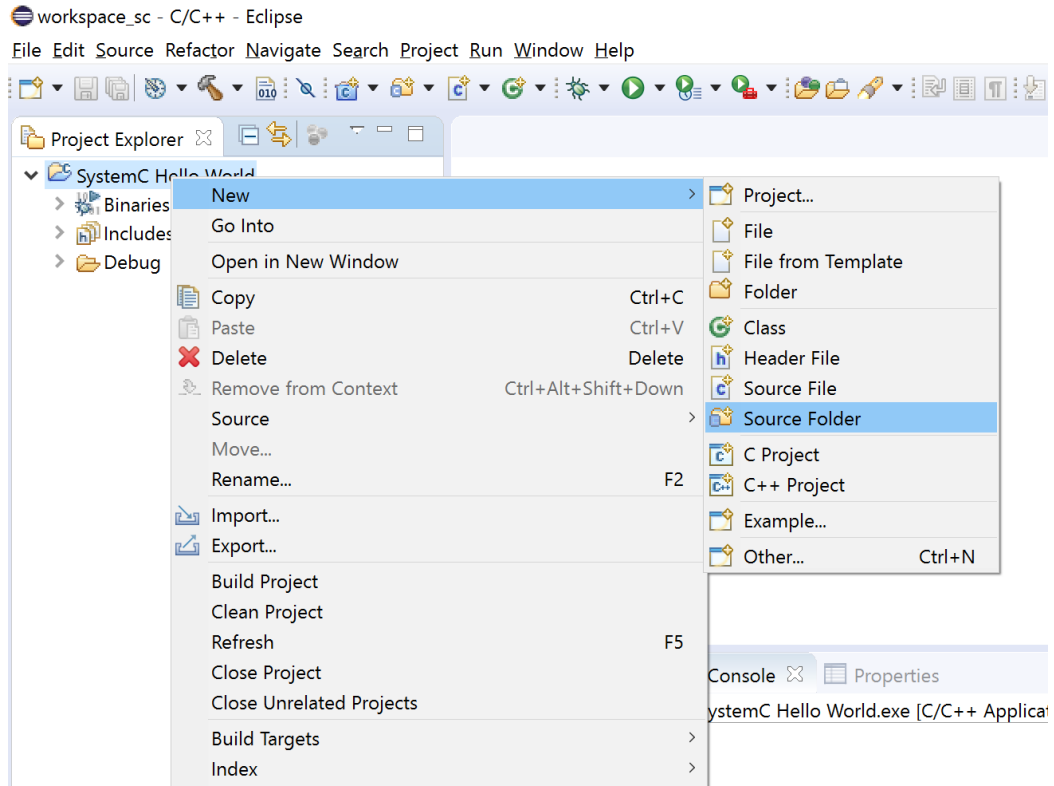


20. Do similar for "Cygwin C++ Linker". (If you have included the fixed-point library, add -SC_INCLUDE_FX to the Miscellaneous of the C++ compiler.)
21. Go to: **C/C++ Build > Settings > Tool Settings > Cygwin C++ Linker > Libraries** and Configure **"Libraries -l"** and **"Library search path -L"** as described next.
22. Under "Library Paths" add the path to the compile systemc libraries, in the form of **"cygwin64/sysclibs/lib-cygwin64"** and as shown in the Figure on the next page.
23. Under "Libraries" tab, add "systemc". Just press add and type "systemc". As shown in the Figure on next page.

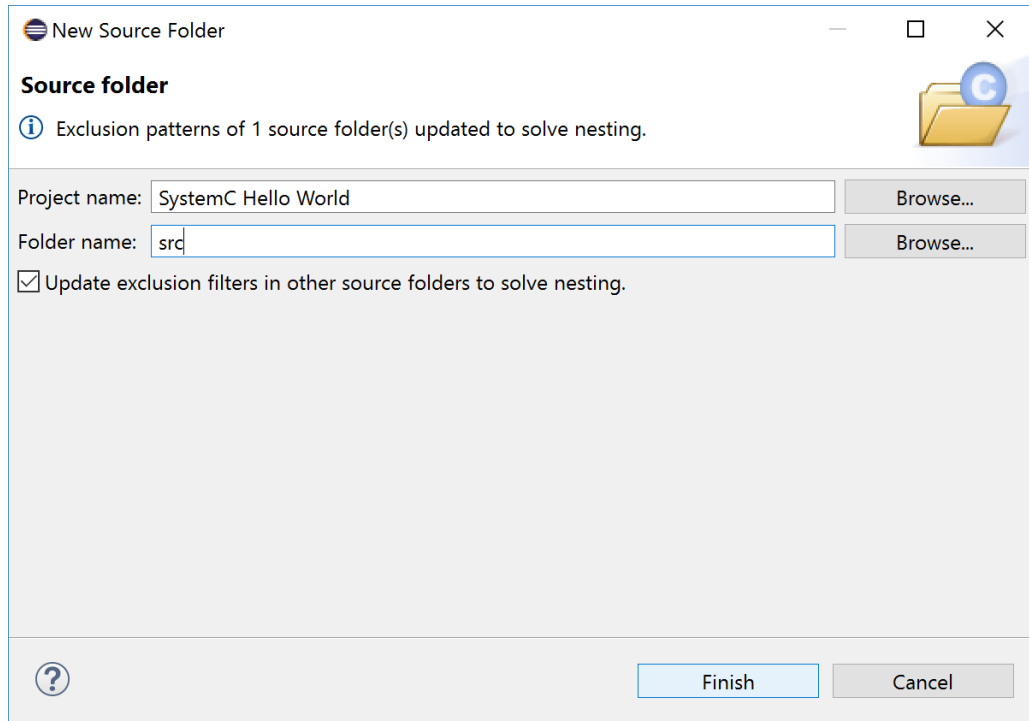


Add the SystemC library for Cygwin

24. Click Apply and Close
25. Then Finish in the next window.
26. Create a new source folder called **src**.

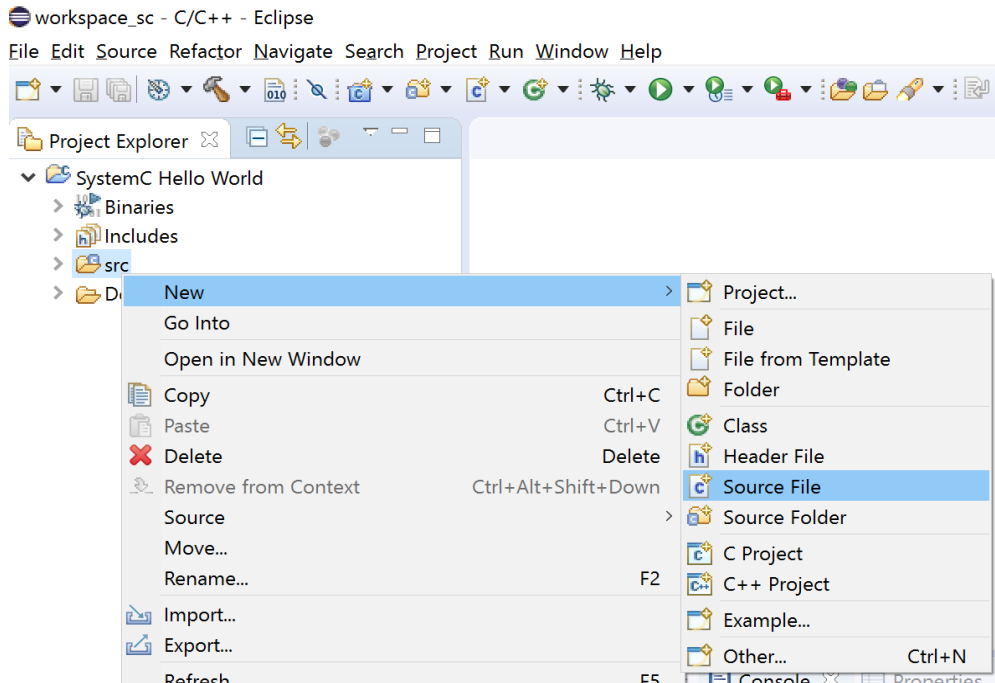


Create a new source folder

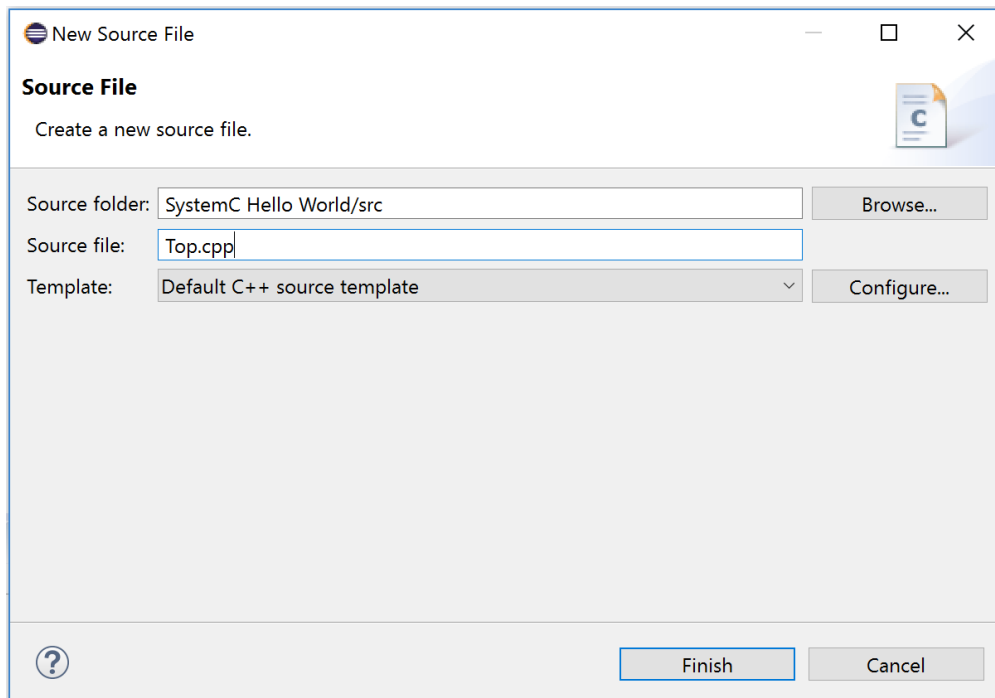


Create a new source folder

Next create a new source file called **Top.cpp**



Create a new source file

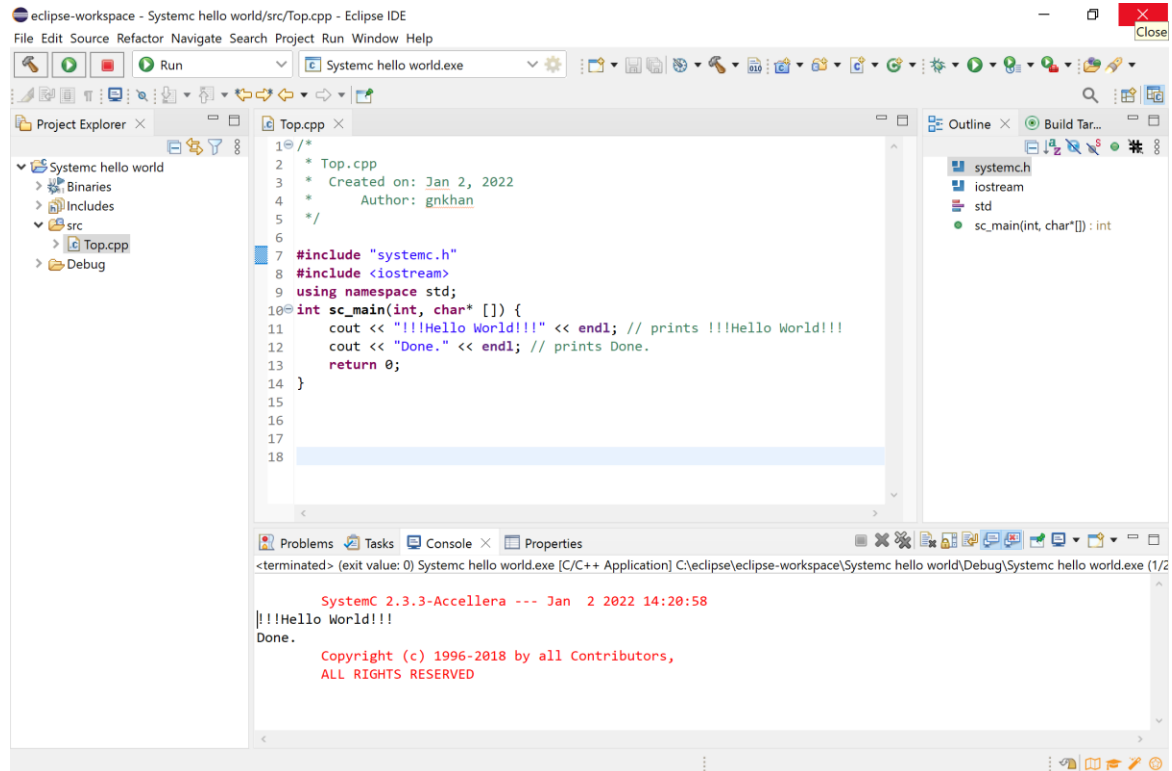


Create a new source file

Next just add the following code in **Top.cpp**

Compile the following code, and if there is no error, you have set up SystemC properly.

```
#include "systemc.h"
#include <iostream>
using namespace std;
int sc_main(int, char* []) {
    cout << "!!!Hello World!!!" << endl; // prints !!!Hello World!!!
    cout << "Done." << endl; // prints Done.
    return 0;
}
```



And that's it!

You have Eclipse configured for SystemC and Cygwin running. Execute the flipflop tutorial from Lab1 that will also generate "trace_file.vcd" file for the flip-flop test.

View and analyze the flipflop "trace_file.vcd" file using gtkwave package for windows system available here at D2L.

Appendix:

If you have included the fixed-point library, and Eclipse cannot resolve the defined fixed-point types, do the following.

- Open "/include/systemc"
- Remove "#ifndef SC_INCLUDE_FX" and its corresponding "endif".