# 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 <u>stackoverflow</u> 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 <u>https://cygwin.com/install.html</u>. Make sure to include the following packages as given in Figure 1 at the end of the document. <u>enter image description here</u>. We installed cygwin in C:\cygwin64. directory.

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



- systemc-2.3.3: you can download the zip file from <u>https://www.accellera.org/downloads/standards/systemc</u> 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".
- (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 <u>https://www.eclipse.org/downloads/</u>
- 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.

⊜C++ Project				×
C++ Project			_	$\diamond$
Create C++ project of selected type				
Project name: SystemC Hello World				
✓ Use default location				
Location: C:\Users\Cristi\workspace_sc\SystemC Hell	o World	E	Browse	
Choose file system: default \vee				
Project type:	Toolchains:			
<ul> <li>&gt; GNU Autotools</li> <li>&gt; Executable</li> <li>&gt; Empty Project</li> <li>&gt; Hello World C++ Project</li> <li>&gt; Shared Library</li> <li>&gt; Static Library</li> <li>&gt; Makefile project</li> </ul>	Cross GCC Cygwin GCC MinGW GCC			
☑ Show project types and toolchains only if they are	supported on the platform			
A Back N	ext > Finish		Cancel	

New SystemC Empty project in Eclipse

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

C++ Project		— 🗆 X	
Select Configur	ations		
Select platforms	and configurations you wish to deploy on		
Proiect type:	Executable		
Toolchains:	Cygwin GCC		
Configurations:			
🖂 🐯 Debug		Select all	
🗠 🐯 Release		Deselect all	
		Advanced settings	
Use "Advanced se	ettings" button to edit project's properties.		
Additional config Use "Manage con	urations can be added after project creation. figurations" buttons either on toolbar or on property pages.		
?	< Back Next > Finish	Cancel	

# 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.

pe filter text	Settings		← < <>
Resource Builders C/C++ Build Build Variables	Configuration: Debug [Active]		~
Logging	Tool Settings Container S	ettings 🎤 Build Steps 🙅 Build Artifact 🗟 Binary Parsers 😣	Error Parsers
Tool Chain Editor C/C++ General Linux Tools Path Project Natures Project References Run/Debug Settings Task Tags Validation WikiText	<ul> <li>SGCC Assembler</li> <li>General</li> <li>Sygwin C++ Compiler</li> <li>Dialect</li> <li>Preprocessor</li> <li>Optimization</li> <li>Debugging</li> <li>Warnings</li> <li>Miscellaneous</li> <li>Sygwin C Compiler</li> <li>Dialect</li> <li>Preprocessor</li> <li>Includes</li> <li>Optimization</li> <li>Debugging</li> <li>Warnings</li> <li>Miscellaneous</li> <li>Miscellaneous</li> <li>Miscellaneous</li> <li>Miscellaneous</li> <li>Miscellaneous</li> </ul>	Include paths (-1)  "C:\cygwin64\sysclibs\include" Include files (-include)	<ul> <li>• 1 월 전 월</li> </ul>
	<ul> <li>Sygwin C++ Linker</li> <li>General</li> <li>Libraries</li> <li>Miscellaneous</li> <li>Shared Library Settings</li> </ul>		

- 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 -I" 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**.

### workspace\_sc - C/C++ - Eclipse

<u>File Edit Source Refactor Navigate Search Project Run Window Help</u>

📬 🗕 🖪 🕲	•	🔨 • 🔜 🛛 🖉 📸 • 🔂 • 🧯	• @ • 🕸 • 🕥 • 🤮	∎ ▼	<b>♀</b> ▼ 🗁 🗁 🖋 ▼	: R' 🗉 🔳 🖢
Project Explore	er S	3 🗖 🔁 🐌 🔻 🗖 🗖				
🗸 😂 SystemC H	مللم	Morld				
> 🖑 Binaries		New	:		😚 Project	
<ul> <li>Include:</li> </ul>	5	Go Into		C	🕈 File	
🕨 🗁 Debug		Open in New Window			📍 File from Template	
	Đ	Сору	Ctrl+C	C	🖞 Folder	
	Ē	Paste	Ctrl+V	e	🖇 Class	
	×	Delete	Delete		🕆 Header File	
	Ð	Remove from Context	Ctrl+Alt+Shift+Down		Source File	
		Source			Source Folder	
		Maria				
		Wove		C	🞽 C Project	
		Rename	F2	C	🗄 C++ Project	
	è	Import		F	🕈 Example	
	പ്പ	Export				Chill N
		Ruild Project		Ŀ	Other	Ctri+N
		Clean Project				
		Refresh	F5			
		Close Project		Co	onsole 🛛 🔲 Prope	erties
		Close Unrelated Projects		VS	stemC Hello World.ex	e [C/C++ Applicat
		Build Targets		>		- [_, , ippilea
		Indox				
		Index				

Create a new source folder

🖨 New Sourc	e Folder	,		×
Source fold (i) Exclusion	er patterns of 1 source folder(s) updated to solve nesting	g.	[	C
Project name:	SystemC Hello World		Browse	
Folder name:	src		Browse	
Update exc	usion filters in other source folders to solve nesting.			
?		Finish	Cancel	

Create a new source folder

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

😂 worksp	ace_sc	- C/C++ - Eclipse	-t Dura Minalauri Hala		
	Source	Refactor Navigate Search Proje		• Q • Q • i	9 @ _ / ▼ : R
Projec	rt Expl				
<ul> <li>✓ <sup>25</sup> Sy:</li> <li>&gt; <sup>3</sup>/<sub>8</sub></li> <li>&gt; <sup>6</sup>/<sub>1</sub></li> </ul>	stemC Binari	Hello World es les			
> <mark>/</mark>	src D	New	>	📑 Project	
_		Go Into		File	
		Open in New Window		File from Ten	nplate
	Þ	Сору	Ctrl+C	🗳 Folder	
	Ē	Paste	Ctrl+V	Class	
	×	Delete	Delete	Header File	
	_52_	Remove from Context	CtrI+Alt+Shift+Down	C Source File	r
		Move	,		
		Rename	F2	C++ Project	
	2	Import		Example	
	്പ	Export		Other	Ctrl+N
		Rafrach	F5		Properties
		Create a	new source file		
	ew Sou	ırce File		-	
Sour	ce Fil	e			
Crea	ate a n	ew source file.			<b>C</b>
Sour	ce fold	er: SystemC Hello World/src			Browse
Sour	ce file:	Τορ.αρ			
Temp	olate:	Default C++ source template		~	Configure
					<u> </u>
(?)				Finish	Cancel

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;
}</pre>
```



# 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 "#ifdef SC\_INCLUDE\_FX" and its corresponding "endif".