

# 하드웨어 소프트웨어 통합설계

Hello SystemC



학 과: 컴퓨터정보공학부

담당교수: 이준환교수님

학 번: 2017202087

성 명: 홍 세 정

## 1. Problem statement

이번 과제는 SystemC를 다운받아 설치하고, SystemC 구현환경을 세팅하여 visual studio로 Hello\_SystemC module을 구현한다.

결과화면 'Hello SystemC World!'를 출력할 수 있다.

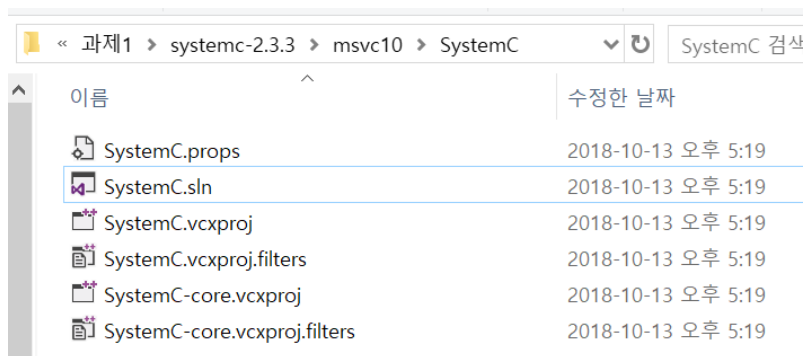
## 2. Installation SystemC library and setup implementation library

<https://www.accellera.org/downloads/standards/systemc> 에서

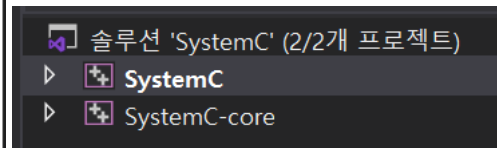
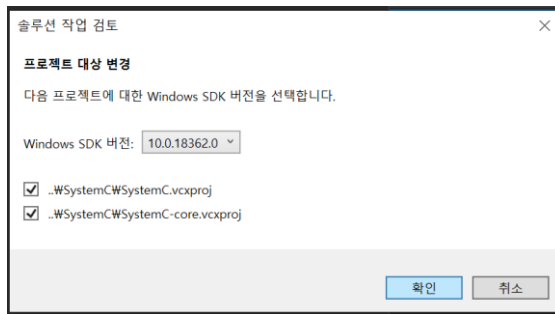
SystemC 2.3.3 core SystemC language and Examples을 다운 받는다.

### Current Releases

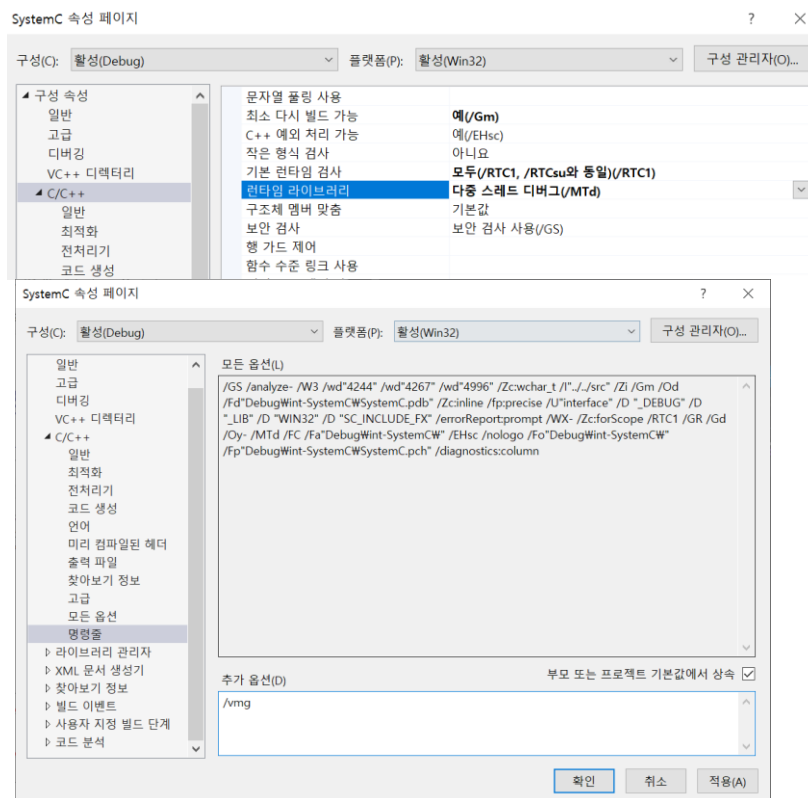
Item	Download	Date Modified
SystemC 2.3.3 (Includes TLM)	<a href="#">Core SystemC Language and Examples (tar.gz)</a>	2018-11-05
	<a href="#">Core SystemC Language and Examples (.zip)</a>	2018-11-05
	<a href="#">SystemC Regression Test Suite (tar.gz)</a>	2018-11-05
	<a href="#">SystemC Regression Test Suite (.zip)</a>	2018-11-05



파일을 압축해제하면 다음과 같은 파일이 있는 것을 확인할 수 있다.



두개의 프로젝트를 세팅해줄 수 있다.



속성으로 들어가서 런타임 라이브러리는 다중 스레드 디버그(MTd)로 변경해주고 명령줄에 /vmg를 추가해준다.

/vmg는 클래스 멤버에 대한 포인터를 선언한다.

SystemC-core 프로젝트도 SystemC 프로젝트와 같이 변경해주었다.

```
2>SystemC.vcxproj -> C:\Users\wsejung\Desktop\3학년 2학기\하드웨어소프트웨어통합설계\과제_HW,S\기말\과제1\SystemC-2.3.3\msvc10\SystemC\Debug\SystemC.lib
2>"SystemC.vcxproj" 프로젝트를 빌드했습니다.
===== 빌드: 성공 2, 실패 0, 최신 0, 생략 0 =====
```

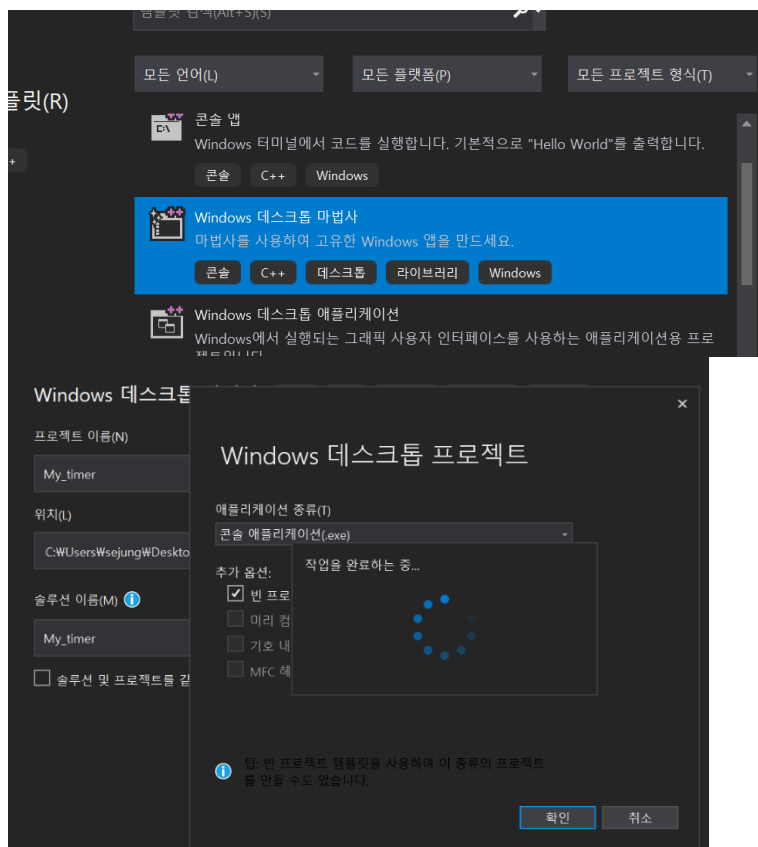
빌드를 성공하였다.

« msvc10 > SystemC > Debug >			Debug 검색
이름	수정한 날짜	유형	
int-SystemC	2020-11-01 오전 12:24	파일 폴더	
int-SystemC-core	2020-11-01 오전 12:38	파일 폴더	
SystemC.lib	2020-11-01 오전 12:38	Object File L	
SystemC.pdb	2020-11-01 오전 12:24	Program Dek	
SystemC-2.3.3.lib	2020-11-01 오전 12:38	Object File L	

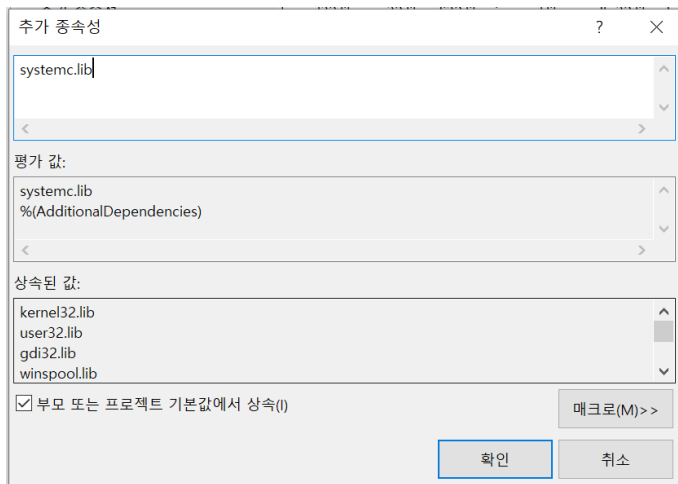
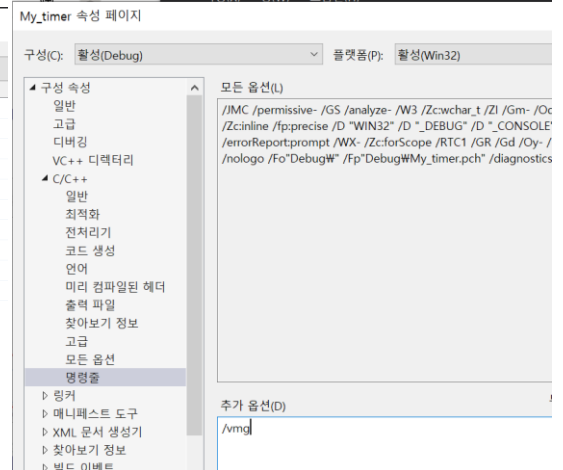
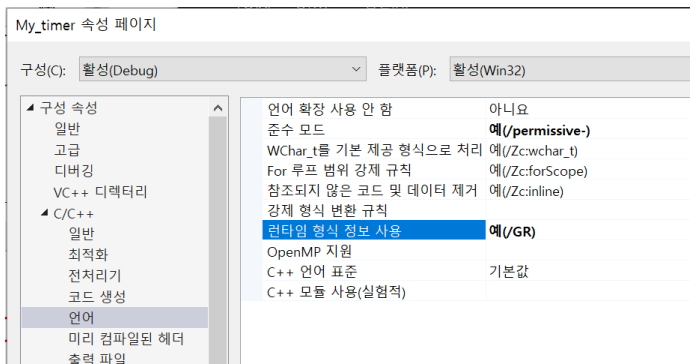
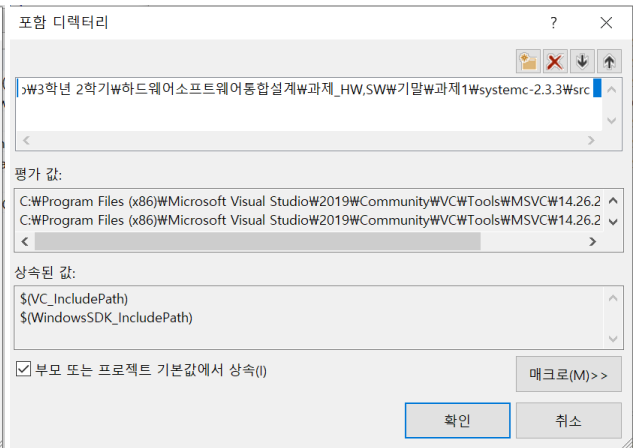
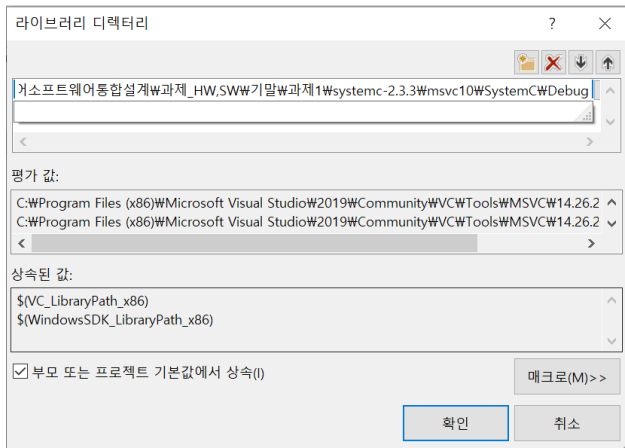
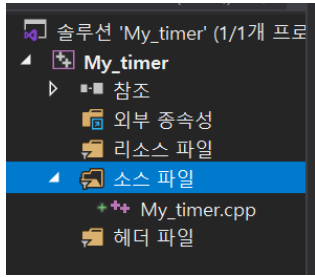
다음과 같이 Debug파일이 생성된 것을 확인할 수 있다.

Hello\_SystemC World!를 출력하기 위해 프로젝트를 생성한다.

평소하던 프로젝트와 다르게 Windows 데스크톱 마법사 템플릿을 사용한다.



My\_timer 프로젝트를 생성하였다.



포함 디렉터리: C:\Users\sejung\Desktop\3학년 2학기\하드웨어소프트웨어통합설계\과제\_HW,SW\기말\과제1\systemc-2.3.3\src

라이브러리 디렉터리: C:\Users\sejung\Desktop\3학년 2학기\하드웨어소프트웨어통합설계\과제\_HW,SW\기말\과제1\systemc-2.3.3\msvc10\SystemC\Debug

다음 위치를 추가하였다.

런타임 라이브러리는 다중 스레드 디버그(/MTd)로 변경해주었다.

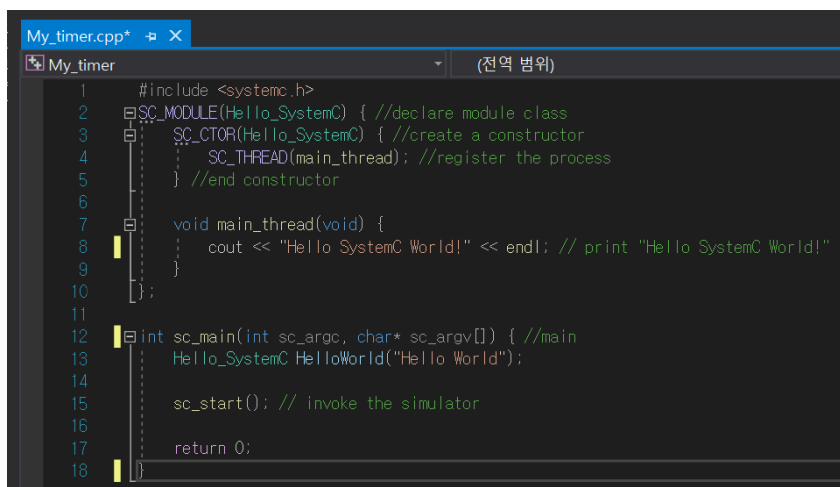
런타임 형식 정보 사용은 예로 바꿔주었다.

명령줄에 /vmg를 추가해주었다.

추가 종속성에 systemc.lib를 추가해주었다.

이렇게 해서 SystemC 구현환경을 세팅하고 module을 구현할 환경을 만들어 주었다.

### 3. Implement Hello\_SystemC module



```
1 #include <systemc.h>
2 SC_MODULE(Hello_SystemC) { //declare module class
3     SC_CTOR(Hello_SystemC) { //create a constructor
4         SC_THREAD(main_thread); //register the process
5     } //end constructor
6
7     void main_thread(void) {
8         cout << "Hello SystemC World!" << endl; // print "Hello SystemC World!"
9     }
10 };
11
12 int sc_main(int sc_argc, char* sc_argv[]) { //main
13     Hello_SystemC HelloWorld("Hello World");
14
15     sc_start(); // invoke the simulator
16
17     return 0;
18 }
```

다음과 같이 Hello SystemC World!를 출력하는 module을 구현하였다.



```
C:\WINDOWS\system32\cmd.exe
SystemC 2.3.3-Accellera --- Nov  1 2020 00:37:14
Copyright (c) 1996-2018 by all Contributors,
ALL RIGHTS RESERVED

Warning: (W506) illegal characters: Hello World substituted by Hello_World
In file: C:\Users\sejung\Desktop\3학년 2학기\하드웨어소프트웨어통합설계\과제_HW,SW\기말\과제1\systemc-2.3.3\src\msvc\kernel\sc_object.cpp:247
Hello_SystemC World!
계속하려면 아무 키나 누르십시오 . . .
```

Hello SystemC World!가 출력되었다.