



AN 918: Using the Intel[®] HLS Compiler Standard Edition with an IDE

Updated for Intel[®] Quartus[®] Prime Design Suite: **19.1**



[Subscribe](#)

[Send Feedback](#)

AN-918 | 2020.05.29

Latest document on the web: [PDF](#) | [HTML](#)



Contents

1. Using the Intel® HLS Compiler Standard Edition with an IDE.....	3
2. Using Intel HLS Compiler Standard Edition with an Eclipse* IDE on Linux Operating Systems.....	4
2.1. Debugging i++ Executable Outputs in the Eclipse IDE	6
3. Using Intel HLS Compiler Standard Edition with Microsoft* Visual Studio* on Windows Operating Systems.....	9
4. Document Revision History.....	13



1. Using the Intel® HLS Compiler Standard Edition with an IDE

Integrating the Intel® HLS Compiler with an IDE lets you use a native compiler and debugger to work on the functional correctness of your testbench and HLS component before optimizing your component for FPGA performance.

This document covers using the Intel HLS Compiler Standard Edition Version 19.1 with an IDE. For instructions on using the Intel HLS Compiler Pro Edition with an IDE, see [AN 834: Using the Intel HLS Compiler Pro Edition with an IDE](#).

Only the Intel HLS Compiler `i++` command generates the High-Level Design Reports that help you to optimize your component.

On Linux operating systems, the native compiler is `g++`, while on Windows operating systems the native compiler is Microsoft Visual C++ (MSVC).

At a high level, using a native compiler (`g++` or MSVC) involves the following steps:

1. Export the path to the Intel HLS Compiler `include` files to your environment:

```
<hls_installdir>/include
```

Where `<hls_installdir>` is the path to your Intel HLS Compiler. The default installation locations for the Intel HLS Compiler are as follows:

- Linux: `/home/<username>/intelFPGA/19.1/hls`
- Windows: `C:\intelFPGA_standard\19.1\hls`

2. Export the path to the Intel HLS Compiler emulation library to the same environment:

- Linux:

```
<hls_installdir>/host/linux64/lib
```

- Windows:

```
<hls_installdir>/host/windows64/lib
```

3. In the environment set with the exported paths, run your IDE (or compiler) and compile your code with the emulation library and any other flags described in the ["Compiler Interoperability" in Intel HLS Compiler Standard Edition Version 19.1 Reference Manual](#).

You can also use your IDE to debug executables generated by the Intel HLS Compiler compiler, including your simulation testbench.

2. Using Intel HLS Compiler Standard Edition with an Eclipse* IDE on Linux Operating Systems

Before you use the Intel HLS Compiler Standard Edition Version 19.1 with an Eclipse* IDE, ensure that you have complete the following tasks:

- Install the Intel HLS Compiler Standard Edition.
For information about installing the Intel HLS Compiler, see [Intel High Level Synthesis Compiler Standard Edition Getting Started Guide](#).
- Install an Eclipse IDE and the Eclipse C/C++ Development Tooling (CDT)

To use the Intel HLS Compiler Standard Edition with an Eclipse IDE, start a terminal session and complete the following steps in that terminal session:

1. Run the following command:

```
source <hls_installdir>/init_hls.sh
```

Where `hls_installdir` is the path to your Intel HLS Compiler. The default installation location for Intel HLS Compiler Standard Edition is `/home/<username>/intelFPGA/19.1/hls`.

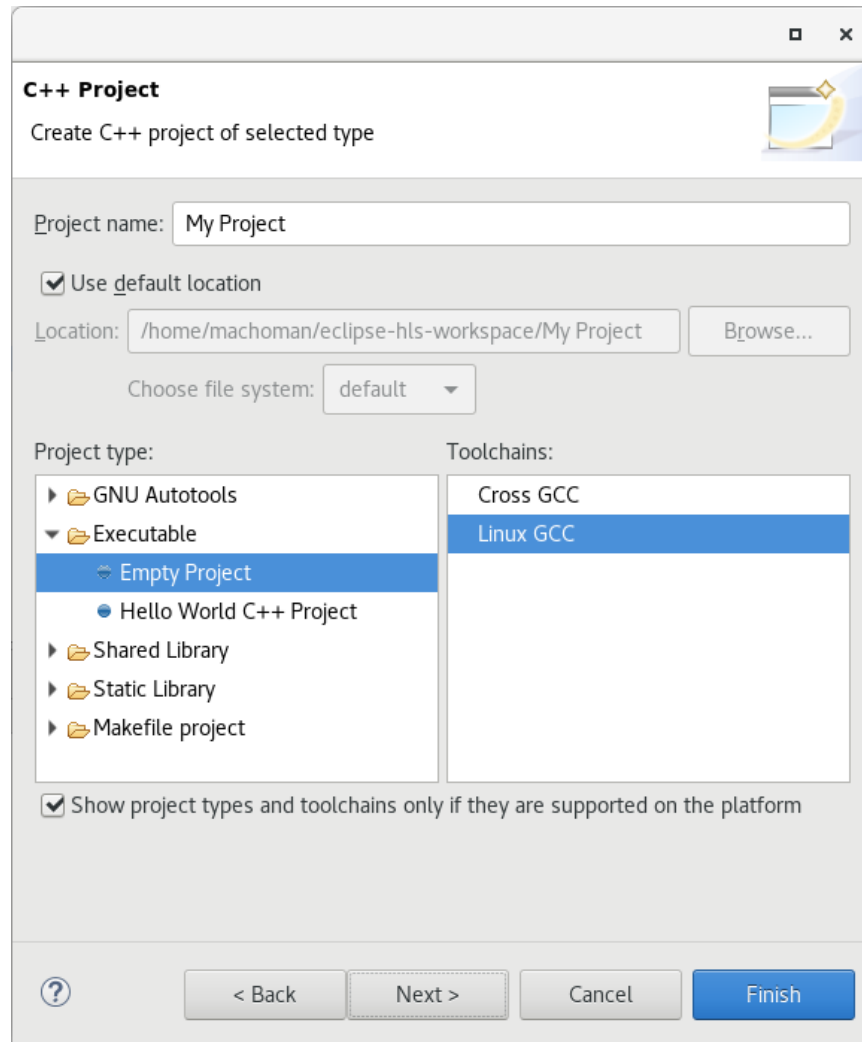
2. Set the `CPATH` and `LIBRARY_PATH` environment variables as follows:

```
export "CPLUS_INCLUDE_PATH=<hls_installdir>/include"  
export "LIBRARY_PATH=<hls_installdir>/host/linux64/lib"
```

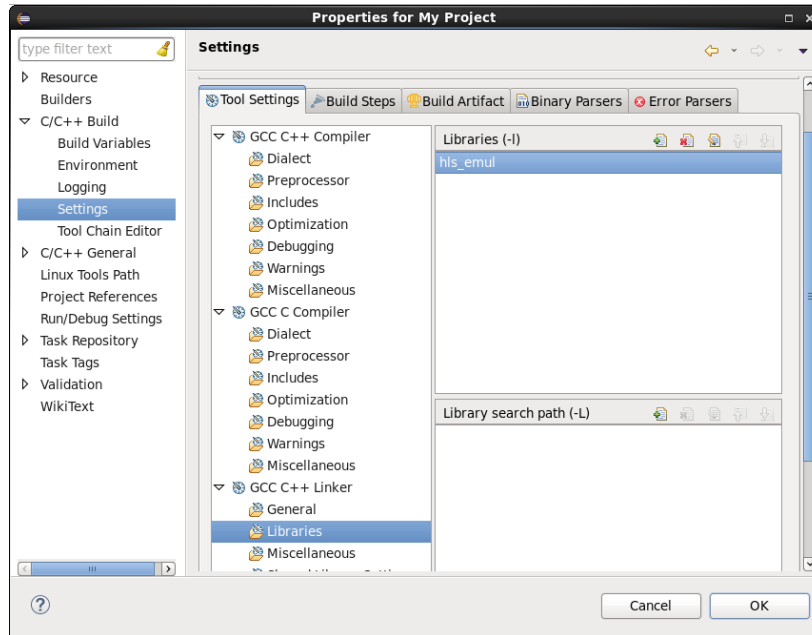
3. In the same terminal session, start your Eclipse IDE.

If you have an HLS project that you have compiled before, you are ready to run and debug your HLS component, and you can skip the next step.

4. Create and configure an Eclipse project for your HLS component:
 - a. Create a new C++ project in Eclipse (**File > New > C++ Project**).
 - b. Select a **C++ Managed Build** template for the project and select the following properties:
 - **Project type: Executable > Empty Project**
 - **Toolchains: Linux GCC**



- c. Click **Finish** to finish creating your project.
- d. Import or create your HLS source files.
- e. If there are problems with the header files, rebuild the index.
To rebuild the index, right-click your project in the **Project Explorer** view and select **Index > Rebuild**.
- f. Set the libraries call properties for the project:
 - i. In the **Project Explorer** view, right-click your project and select **Properties**.
 - ii. In the **Properties** window for you project, go to **C/C++ Build > Settings**.
 - iii. On the **Tool Settings** tab, go to **GCC C++ Linker > Libraries** and add the `hls_emul` library to the list of libraries.



You can now develop, run, and debug your HLS component using your Eclipse IDE.

After you have confirmed the functional correctness of your component, you can start to optimize your HLS component by using the information in the Intel HLS Compiler High-Level Design Reports. These reports are generated when you compile your component with the `i++` command.

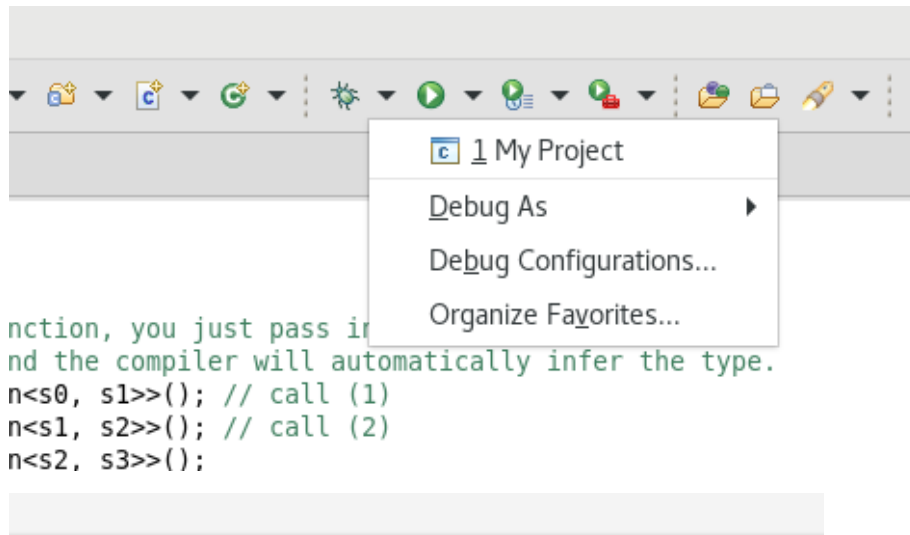
You can also use your Eclipse IDE to debug executable files that are produced by compiling your component with the `i++` command

2.1. Debugging i++ Executable Outputs in the Eclipse IDE

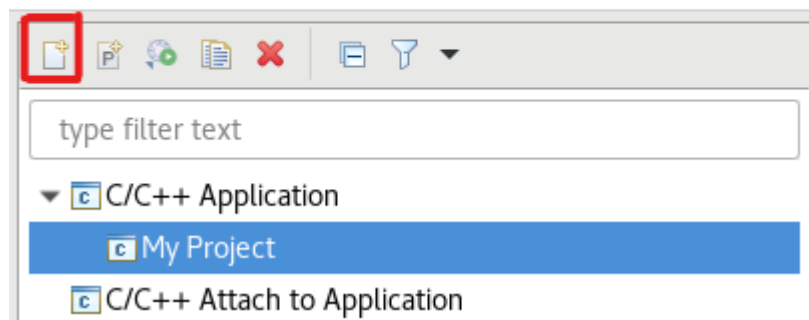
You can use your Eclipse IDE to debug executable files that are produced by compiling your component with the `i++` command by creating and using a debug configuration for the executable.

To debug an executable file produced by the `i++` command:

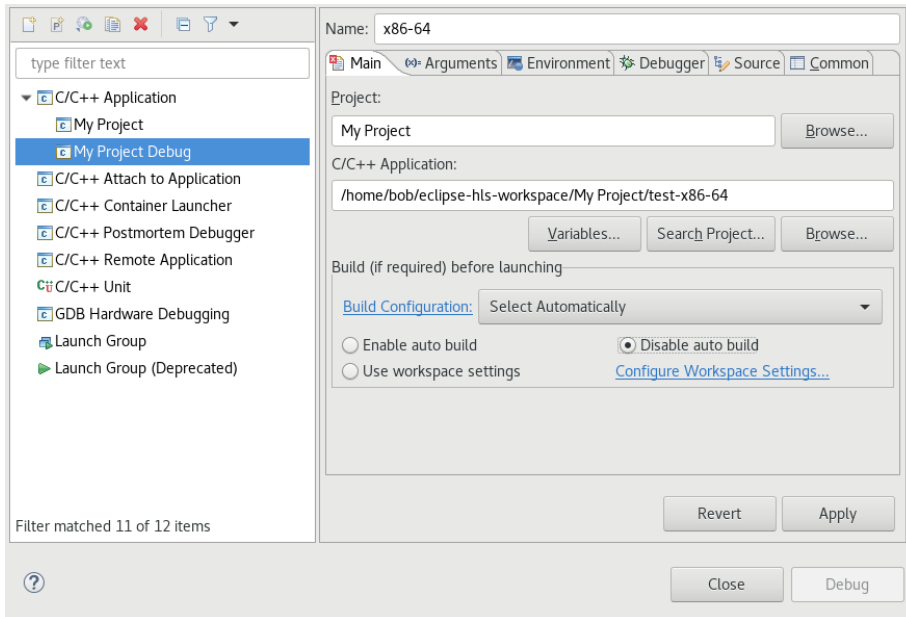
1. Open **Debug Configurations** and create a new debug configuration for the executable:



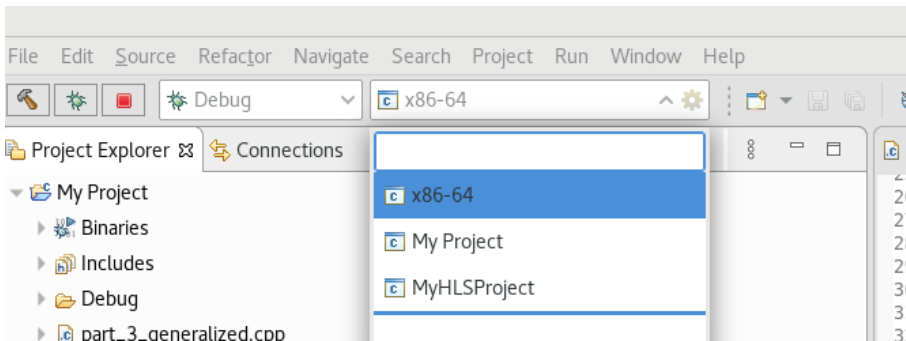
Create, manage, and run configurations



2. Load the executable file produced by the `i++` command and disable auto build:



3. Debug your executable by selecting your new debug configuration and clicking the debug button:



3. Using Intel HLS Compiler Standard Edition with Microsoft* Visual Studio* on Windows Operating Systems

Before you use the Intel HLS Compiler Standard Edition Version 19.1 with Microsoft* Visual Studio*, ensure that you have complete the following tasks:

- Install the Intel HLS Compiler Standard Edition.
For information about installing the Intel HLS Compiler, see [Intel High Level Synthesis Compiler Standard Edition Getting Started Guide](#).
- Install a version of Microsoft Visual Studio* that is supported by the Intel HLS Compiler Standard Edition.
For supported versions of Microsoft Visual Studio*, see "Intel High Level Synthesis Compiler Standard Edition Prerequisites" in [Intel High Level Synthesis Compiler Standard Edition Getting Started Guide](#).

To use the Intel HLS Compiler Standard Edition with a Visual Studio IDE:

1. Start a Visual Studio x64 Native Tools Command Prompt session.

For example, C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC\bin\amd64\vcvars64.bat.

2. In your command prompt session, run the following command:

```
<hls_installdir>\init_hls.bat
```

Where *<hls_installdir>* is the path to your Intel HLS Compiler installation. For example, C:\intelFPGA_standard\19.1\hls.

3. Set additional environment variables and settings required:

```
set "INCLUDE=%INSTALLROOT%include;%INCLUDE%"
set "LIB=%INSTALLROOT%host\windows64\lib;%LIB%"
set "CL=/MDd -D_ITERATOR_DEBUG_LEVEL=0"
set "_LINK=hls_emul.lib"
set "_IsNativeEnvironment=true"
```

4. Start your Visual Studio IDE:

```
devenv /useenv
```

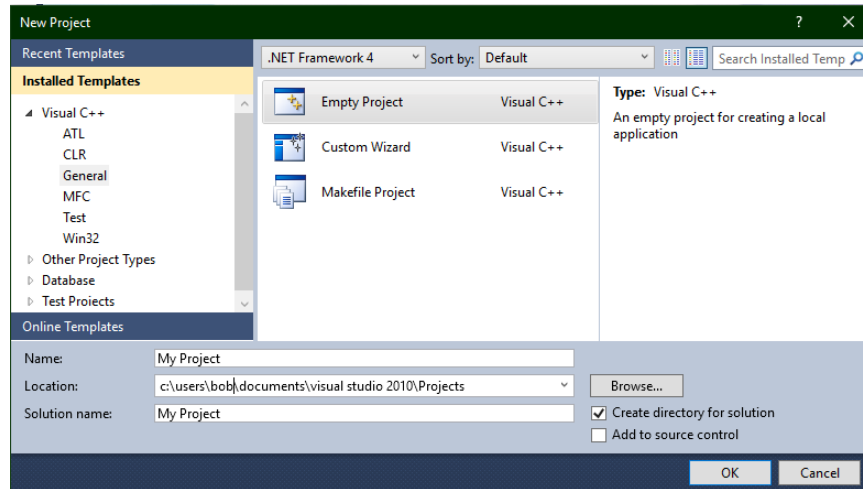
If you have an HLS project that you have compiled before, you are ready to run and debug your HLS component, and you can skip the next step.

5. Create and configure a Visual Studio project for your HLS component:
 - a. Create a new project:

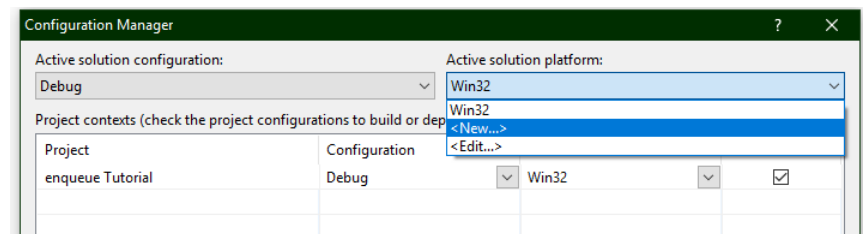
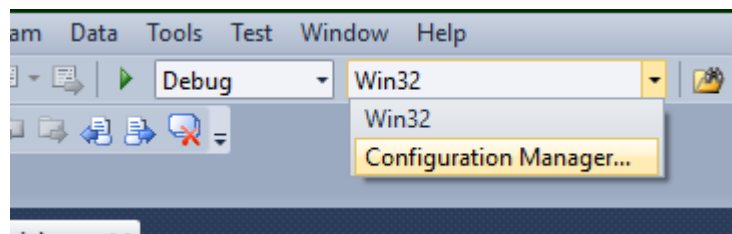


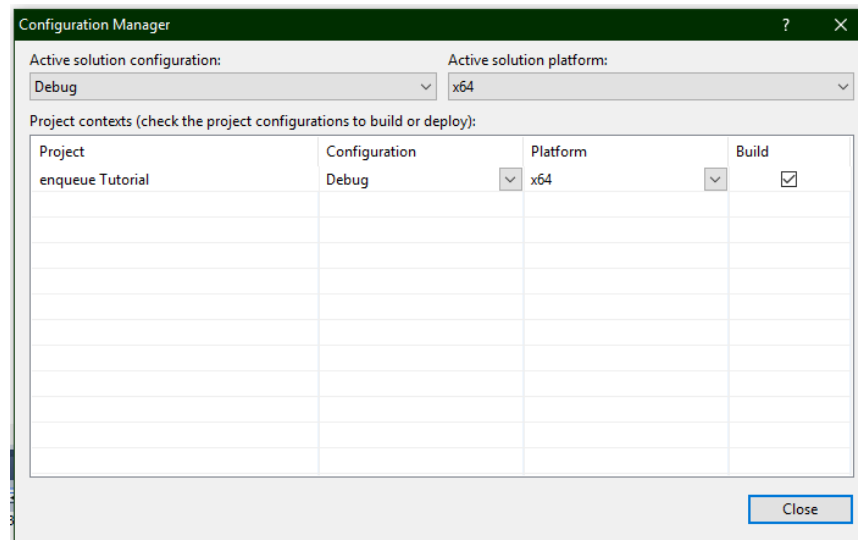
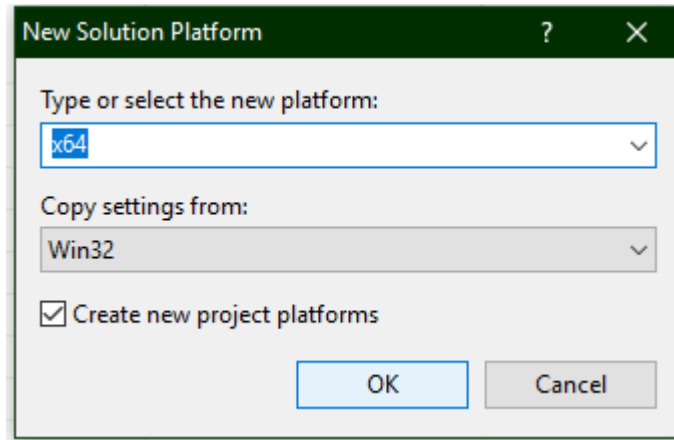
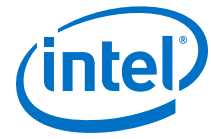
3. Using Intel HLS Compiler Standard Edition with Microsoft* Visual Studio* on Windows Operating Systems

AN-918 | 2020.05.29



b. Add a 64-bit configuration to your project:



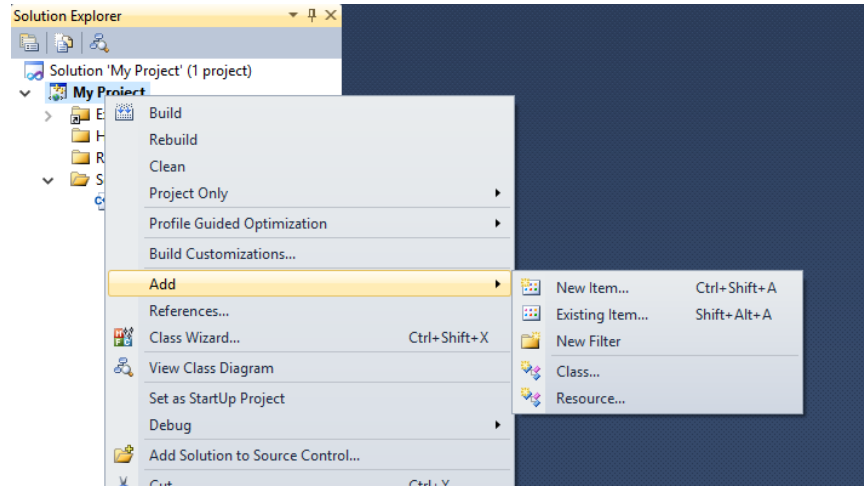


c. Add your sources:



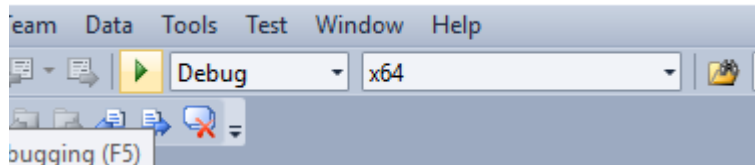
3. Using Intel HLS Compiler Standard Edition with Microsoft* Visual Studio* on Windows Operating Systems

AN-918 | 2020.05.29



You can now develop, run, and debug your HLS component using your Microsoft Visual Studio* IDE.

To run your component in Visual Studio, choose the 64-bit target:



After you have confirmed the functional correctness of your component, you can start to optimize your HLS component by using the information in the Intel HLS Compiler High-Level Design Reports. These reports are generated when you compile your component with the `i++` command.

With Intel HLS Compiler Standard Edition, you cannot use Microsoft Visual Studio to debug executables created the `i++` command.



4. Document Revision History

Table 1. Document Revision History for AN 918: Using the Intel HLS Compiler Standard Edition with an IDE

Date	Version	Changes
May 2020	2020.05.29	• Initial release