# Obtaining the Source

Microsoft HLS SDK & Player Framework Plugins

Build Instructions

The source for the Microsoft HLS SDK and the related player framework plugins is available at <http://github.com/MicrosoftDX/MicrosoftHLSSDK>. Before you build you will need to clone the source to your development machine using either Visual Studio or your favorite Git client.

Once you finish cloning, you will notice three folders under the MicrosoftHLSSDK root folder – namely SDK, PFPlugins and Samples. The SDK folder contains the source code for the HLS SDK, the PFPlugins folder contains the same for the Player Framework plugins for HLS and the Samples folder contain the source code for the HLS Stream Monitor reference sample.

If you do not want to build the source, you can also use the pre-built installers found in the Installers folder.

# Building the SDK

### Windows 8.1

Using Visual Studio 2013, open the solution named Microsoft.HLSClient found under SDK/Windows8.1. Select Batch Build, select all and then click Clean. Once all projects are cleaned, deselect all and check the following projects. The click build.

|  |  |
| --- | --- |
| Project Name | Configuration |
| Microsoft.HLSClient.Windows | Debug | ARM |
| Microsoft.HLSClient.Windows | Debug | x86 |
| Microsoft.HLSClient.Windows | Debug | x64 |
| Microsoft.HLSClient.Windows | Release | ARM |
| Microsoft.HLSClient.Windows | Release | x86 |
| Microsoft.HLSClient.Windows | Release | x64 |
| Microsoft.HLSClient.WindowsPhone | Debug | ARM |
| Microsoft.HLSClient.WindowsPhone | Debug | x86 |
| Microsoft.HLSClient.WindowsPhone | Release | ARM |
| Microsoft.HLSClient.WindowsPhone | Release | x86 |

Once all 10 projects build successfully, switch your Visual Studio build configuration to Release | Any CPU, right click on the Microsoft.HLSClient.Setup project under the Setup solution folder and click build.

Once this is completed, you will find the SDK installer named Microsoft.HLSClient.Setup.vsix under the SDK/Windows8.1/Microsoft.HLSClient.Setup/Bin/Release folder.

### Windows 10

Using Visual Studio 2015, open the solution named Microsoft.HLSClient found under SDK/Windows 10. Select Batch Build, select all and then click Clean. Once all projects are cleaned, deselect all and check the following projects. The click build.

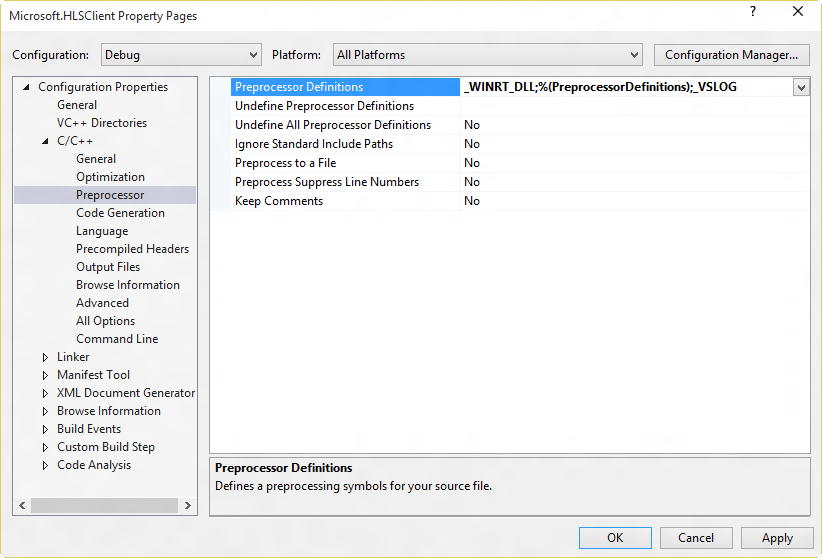
|  |  |
| --- | --- |
| Project Name | Configuration |
| Microsoft.HLSClient.Windows | Debug | ARM |
| Microsoft.HLSClient.Windows | Debug | x86 |
| Microsoft.HLSClient.Windows | Debug | x64 |
| Microsoft.HLSClient.Windows | Release | ARM |
| Microsoft.HLSClient.Windows | Release | x86 |
| Microsoft.HLSClient.Windows | Release | x64 |

Once all 10 projects build successfully, switch your Visual Studio build configuration to Release | Any CPU, right click on the Microsoft.HLSClient.Setup project and click build.

Once this is completed, you will find the SDK installer named Microsoft.HLSClient.Setup.vsix under the SDK/Windows 10/Microsoft.HLSClient.Setup/Bin/Release folder.

Enabling Logging

The HLS runtime can supply rich log information to the Visual Studio output window when your app is run in debug mode from within Visual Studio. To enable this, add the \_VSLOG preprocessor directive to all debug configurations of the SDK project before building.



If you are debugging remotely on a device or debugging the SDK for the phone on a USB connected phone device it is advisable to not use this setting as it might slow your debugging experience significantly.

# Building the Player Framework Plugins

### Windows 8.1

Using Visual Studio 2013, open the solution named MSHLS.PF found under PFPlugins/Windows8.1 folder. Select Batch Build, select all and then click Clean. Once all projects are cleaned, deselect all and check the following projects. The click build.

|  |  |
| --- | --- |
| Project Name | Configuration |
| HLSClientExtensions.Windows | Debug | ARM |
| HLSClientExtensions.Windows | Debug | Win32 |
| HLSClientExtensions.Windows | Debug | x64 |
| HLSClientExtensions.Windows | Release | ARM |
| HLSClientExtensions.Windows | Release | Win32 |
| HLSClientExtensions.Windows | Release | x64 |
| HLSClientExtensions.WindowsPhone | Debug | ARM |
| HLSClientExtensions.WindowsPhone | Debug | Win32 |
| HLSClientExtensions.WindowsPhone | Release | ARM |
| HLSClientExtensions.WindowsPhone | Release | Win32 |
| Microsoft.CC608 | Debug | ARM |
| Microsoft.CC608 | Debug | Win32 |
| Microsoft.CC608 | Debug | x64 |
| Microsoft.CC608 | Release | ARM |
| Microsoft.CC608 | Release | Win32 |
| Microsoft.CC608 | Release | x64 |
| Microsoft.CC608.WindowsPhone | Debug | ARM |
| Microsoft.CC608.WindowsPhone | Debug | Win32 |
| Microsoft.CC608.WindowsPhone | Release | ARM |
| Microsoft.CC608.WindowsPhone | Release | Win32 |
| MSHLS.PF.JS.CC608.Windows | Debug | ARM |
| MSHLS.PF.JS.CC608.Windows | Debug | x86 |
| MSHLS.PF.JS.CC608.Windows | Debug | x64 |
| MSHLS.PF.JS.CC608.Windows | Release | ARM |
| MSHLS.PF.JS.CC608.Windows | Release | x86 |
| MSHLS.PF.JS.CC608.Windows | Release | x64 |
| MSHLS.PF.JS.CC608.WindowsPhone | Debug | ARM |
| MSHLS.PF.JS.CC608.WindowsPhone | Debug | x86 |
| MSHLS.PF.JS.CC608.WindowsPhone | Release | ARM |
| MSHLS.PF.JS.CC608.WindowsPhone | Release | x86 |
| MSHLS.PF.JS.Windows | Debug | ARM |
| MSHLS.PF.JS.Windows | Debug | x86 |
| MSHLS.PF.JS.Windows | Debug | x64 |
| MSHLS.PF.JS.Windows | Release | ARM |
| MSHLS.PF.JS.Windows | Release | x86 |
| MSHLS.PF.JS.Windows | Release | x64 |
| MSHLS.PF.JS.WindowsPhone | Debug | ARM |
| MSHLS.PF.JS.WindowsPhone | Debug | x86 |
| MSHLS.PF.JS.WindowsPhone | Release | ARM |
| MSHLS.PF.JS.WindowsPhone | Release | x86 |
| MSHLS.PF.XAML.CC608.Windows | Debug | ARM |
| MSHLS.PF.XAML.CC608.Windows | Debug | x86 |
| MSHLS.PF.XAML.CC608.Windows | Debug | x64 |
| MSHLS.PF.XAML.CC608.Windows | Release | ARM |
| MSHLS.PF.XAML.CC608.Windows | Release | x86 |
| MSHLS.PF.XAML.CC608.Windows | Release | x64 |
| MSHLS.PF.XAML.CC608.WindowsPhone | Debug | ARM |
| MSHLS.PF.XAML.CC608.WindowsPhone | Debug | x86 |
| MSHLS.PF.XAML.CC608.WindowsPhone | Release | ARM |
| MSHLS.PF.XAML.CC608.WindowsPhone | Release | x86 |
| MSHLS.PF.XAML.Windows | Debug | ARM |
| MSHLS.PF.XAML.Windows | Debug | x86 |
| MSHLS.PF.XAML.Windows | Debug | x64 |
| MSHLS.PF.XAML.Windows | Release | ARM |
| MSHLS.PF.XAML.Windows | Release | x86 |
| MSHLS.PF.XAML.Windows | Release | x64 |
| MSHLS.PF.XAML.WindowsPhone | Debug | ARM |
| MSHLS.PF.XAML.WindowsPhone | Debug | x86 |
| MSHLS.PF.XAML.WindowsPhone | Release | ARM |
| MSHLS.PF.XAML.WindowsPhone | Release | x86 |

Once all of the projects build successfully, switch your Visual Studio build configuration to Release | Any CPU, right click on the MSHLS.PF.Setup project, and click build.

Once this is completed, you will find the SDK installer named MSHLS.PF.XAML.Setup.vsix under the PFPlugins/Windows8.1/Setup/MSHLS.PF.Setup/Bin/Release folder.

### Windows 10

Using Visual Studio 2015, open the solution named MSHLS.PF found under PFPlugins/Windows10 folder. Select Batch Build, select all and then click Clean. Once all projects are cleaned, deselect all and check the following projects. The click build.

|  |  |
| --- | --- |
| Project Name | Configuration |
| Microsoft.CC608 | Debug | ARM |
| Microsoft.CC608 | Debug | Win32 |
| Microsoft.CC608 | Debug | x64 |
| Microsoft.CC608 | Release | ARM |
| Microsoft.CC608 | Release | Win32 |
| Microsoft.CC608 | Release | x64 |
| MSHLS.PF.XAML | Debug | ARM |
| MSHLS.PF.XAML | Debug | x86 |
| MSHLS.PF.XAML | Debug | x64 |
| MSHLS.PF.XAML | Release | ARM |
| MSHLS.PF.XAML | Release | x86 |
| MSHLS.PF.XAML | Release | x64 |

Once all of the projects build successfully, switch your Visual Studio build configuration to Release | Any CPU, right click on the MSHLS.PF.XAML.Setup project and click build.

Once this is completed, you will find the SDK installer named MSHLS.PF.XAML.Setup.vsix under the PFPlugins/Windows10/Setup/MSHLS.PF.XAML.Setup/Bin/Release folder.