

Technical Note

Grayhill Vehicle Solutions

Date: May 7, 2021
Revision: B
Subject: Windows 10 - Upgrade existing Qt 5.9.3 Libraries to Qt 5.12.2
Reference: VSTN2021-02

Intended Audience

This technical note is intended for customers who are currently using Qt 5.9.3 libraries on a computer running Windows 10. Previous Windows versions while still operational are **not** supported.

Please note that the name of the kit is 5.9.3 and was named as such as to indicate that this kit was released to support Qt 5.9.3. The upgrade updates/replaces files in the necessary folders (e.g. qmakeinstall). So, while the name of kit remains the same, its actual contents have been updated to utilize the Qt 5.12.2 libraries.

Introduction

This technical note provides the necessary instructions for updating a Windows 10 computer with Qt 5.9.3 libraries to Qt 5.12.2 libraries.

Prerequisites

The following conditions must be true:

- Computer running Windows 10 with Qt 5.9.3 installed
- Existing 3Dxx display also configured for Qt 5.9.3
- Ethernet connection to the 3Dxx display
- RS232 (or serial) connection to the 3Dxx display

Procedure

While these instructions make every attempt to be complete, certain discrepancies may occur based on the actual development environment being used.

[PC] Download Qt 5.12.2 files

- Launch your favorite internet browser
- Navigate to www.grayhill.com/qt43d and download
 - QtGhUpgrade5122Win10.exe

[3Dxx] Determine the IP address of the 3Dxx display

Using the serial (RS232) connection to the 3Dxx display

- `ifconfig eth0`

[PC] Update “gmd” reference to the display’s IP address

Using Notepad++ (or another editor with admin right capabilities) edit `C:\Windows\System32\drivers\etc\hosts` and add/update the following line

```
net.net.net.host gmd
```

where “net.net.net.host” is the IP address discovered above

[PC] Unarchive the files

- Open the folder with **QtGhUpgrade5122Win10.exe** (most likely Downloads folder)
- Double click to run the program (self-extracting archive)

Note: If a “Program Compatibility Assistant” window appears, click “Close”

[PC] Install Qt 5.12.2 libraries on the 3Dxx display remotely

- Navigate to the `C:\QtGhSupport` folder
- Double click on **QtGhUpgradeWin10Install.bat**

To update additional displays, connect the 3Dxx and update **gmd** (see above) with the IP address then re-run the script

[PC] Verify Qt Creator is configured for “gmd” vs. hard coded IP

- Launch Qt Creator (C:\Qt\Tools\QtCreator\bin\qtcreator.exe)
- Verify “Host name:” is set to gmd (Tools→ Options → Devices (left hand column) → Devices (tab))

[PC] Verify sample project builds and runs on target

- Launch Qt Creator (if not running)
- Open Project
 - For 3Dxx displays running linux 3.0.35 kernel (-100 series), use project GrayhillExamples/ghQmlDemo/ghQmlDemo.pro
 - For 3Dxx displays running linux 4.1.15 kernel (-200 series), use project GrayhillExamples/ghQmlDemo4115/ghQmlDemo4115.pro

- Select Projects (wrench icon)
- Select Build under Qt-5.9.3-3Dxx
- Expand qmake
- Verify qmake “Additional arguments:” is set correctly

- For the ghQmlDemo (linux 3.0.35, -100 series) project, set to

“hw_present=yes target=3D70”

Note: for **target** setting, use 3D70, 3D50, or 3D2104 based on actual display.

- For the ghQmlDemo4115 (linux 4.1.15, -200 series) project, set to

“hw_present=yes target=3D70 kernel=4”

Note: for **target** setting, use 3D70, 3D50, 3D2104, or 3D101 based on actual display.

- Build, deploy and run the project by clicking on the green triangle

[PC] (optional) Verify sample project builds and runs on desktop

- Terminate previously running application (Application Output tab → red square)
- Select Projects(wrench icon)

- Select Build under Desktop Qt 5.9.3 MinGW 32bit
- Verify qmake “Additional arguments:” is set correctly
 - For the ghQmlDemo (linux 3.0.35, -100 series) project, set to

“target=3D70 windowsOnly=true”

Note: for **target** setting, use 3D70, 3D50, or 3D2104 based on desired display simulation

- For the ghQmlDemo4115 (linux 4.1.15, -200 series) project, set to

“target=3D70 kernel=4 windowsOnly=true”

Note: for **target** setting, use 3D70, 3D50, 3D2104, or 3D101 based on desired display simulation

- Build, deploy and run the project by clicking on the green triangle