

Vehicle Solutions Group

Qt 5.12.2 User's Guide

Linux and Windows

Revision C

Table of Contents

1.	I	ntroduction	4
	1.1	Purpose	4
	1.2	Acronyms and Definitions	5
	1.3	References	5
	1.4	Revision History	5
2.	F	Requirements	6
	2.1	Hardware	6
	2.1.1 2.1.2 2.2	2 Recommended Equipment	7
	2.2.1 2.2.2 2.2.3	2 Grayhill Qt Support Files (https://www.grayhill.com/qt43d)	7
3.	I	nstallation	8
	3.1	Install the Development Kit	8
	3.2	Download and Install Qt Creator	8
	3.3	Windows Utilities	2
	3.3.1 3.4	1 Download and Install PuTTY	
	3.4.1 3.4.2	2 Windows	25
	3.4.3 3.4.4		
		4 Configure IP address	
	V	Vindows2	29
	3.5	Download and Install Support Files	2
	3.5.2 3.5.2 3.6	1 Linux	82 83
	3.6.		
		_inux3	
	V	Nindows3	8
	3.6.2	1 1 5	
	L	.inux3	9

Wir	ndows	
3.6.3	Build Project	
Appendix	A: Configuring a Manual Qt Kit for Grayhill Displays	
Appendix	B: Configuring a 3Dxx Project	61
Appendix	C: Debugging	75
Appendix	D: Setting up a 3Dxx Qt Program to Run at Boot Up	79
Appendix	E: Interfacing 3Dxx Hardware from QT Software	81
Appendix	F: Setting 3Dxx Flash File System R/W Mode	114
Appendix	G: Building Qt Library Source (optional)	115
Appendix	H: Dynamic IP Address	116
Appendix	I: Static IP Address	117

1. Introduction

1.1 Purpose

This document describes:

- Setup and usage of the Qt-based development environment for Grayhill 3Dxx display products
- Code development for a 3Dxx Display product in the Qt IDE
- Accessing various 3Dxx hardware features via this code
- Loading developed application code onto a 3Dxx Display product

The Qt cross-platform development environment runs under both Linux and Windows 10. The Linux platform is supported by a virtual machine using Oracle's VirtualBox (<u>https://www.virtualbox.org/wiki/VirtualBox</u>) software.

The virtual machine is Ubuntu 16.04 using gnome flashback for the desktop; additionally PuTTY (telnet client software - https://www.putty.org) is installed. 16.04 is a Long Term Support release, currently scheduled for end of life in April 2021. The VM also comes with Qt Creator and libraries installed.

For Virtual Machine installation, please reference "Virtual Machine Installation Using VirtualBox", which is available on the Grayhill web site (<u>https://www.grayhill.com/qt43d</u>)

This document is intended for use by software developers familiar programming in C/C++ using the Qt framework. Experience developing applications for Linux platforms is a definite plus.

Screen shots try to be as accurate as possible and are provided as reference.

N.B. Screen images are mixed between the Windows version of Qt Creator and Linux, but the steps are the same.

Note: Qt is licensed under the terms of LGPL and GPL. These are open-source licensing agreements. Please reference <u>https://www1.qt.io/qt-licensing-terms/</u> for a detailed explanation. Additional information is also located at <u>https://www.gnu.org/licenses/licenses.html</u>.

1.2 Acronyms and Definitions

3Dxx	Reference to any of the Grayhill 3D series displays (3D50, 3D70, 3D2104, 3D101)
CAN	Controller Area Network
GB	Giga Byte
RAM	Random Access Memory
USB	Universal Serial Bus
VM	Virtual Machine

1.3 References

[1] VSTN2021-01	Linux - Upgrade existing Qt 5.9.3 Libraries to Qt 5.12.2
[2] VSTN2021-02	Windows 10 - Upgrade existing Qt 5.9.3 Libraries to Qt 5.12.2
[3] VSUD2019-02	Virtual Machine Installation Using VirtualBox

1.4 Revision History

Revision	Author	Date	Description
А	K. Struss	9/6/2019	Initial Release combining the previous
			independent Linux and Windows manuals
В	K. Jalowiec	11/18/2020	Added model 3D101 display
С	K. Jalowiec	10/12/2021	Corrected libghio references

2. Requirements

2.1 Hardware

2.1.1 Supported Grayhill Display Hardware

The Qt-based development environment supports the following Grayhill 3Dxx Color Display Models:

- 3D50
- 3D70
- 3D2104
- 3D101

The table below summarizes the key features of each of these models. Note that the features of a specific product may vary depending on the purchased hardware configuration.

Model Number	3D50-x00	3D70-x00	3D2104-x00	3D101-200
Display Size (inches)	5	7	10.4	10.1
Pixel Count (w x h)	800 x 480	800 x 480	1024 x 768	1280 x 800
Touch Screen Input	Yes	Yes	Yes	Yes
Real Time Clock	Yes	Yes	Yes	Yes
CAN Ports	2	2	3	3
Camera Inputs	2	3	4	4
	1	1	1	1
USB ports	(maintenance	(maintenance	(maintenance	(maintenance
	only)	only)	only)	only)
	1	1	1	1
RS232	(maintenance	(maintenance	(maintenance	(maintenance
	only)	only)	only)	only)
Built-in Ethernet	0	1	1	1
Digital Input (dedicated)	1	4	0	0
Digital Output (dedicated)	1	4	0	0
Digital Input / Output	3	0	4	4
Analog Input	0	2	0	0
Audio Output	No	1 channel	No	No
Buzzer	No	Yes	Yes	Yes

In order to use Qt 5.12.2 on Grayhill 3Dxx display models running the Linux 3.0.35 kernel, the April 28, 2017 version or later of the 3.0.35 kernel MUST be installed.

Supported Linux OS software for the 3Dxx displays is available for download on the Grayhill web page:

https://www.grayhill.com/3d-series-displays

2.1.2 Recommended Equipment

It is strongly recommended the associated development kit be used for development.

- 3D50DEV-100 3D50 Development Kit
- 3D70DEV-100 3D70 Development Kit
- 3D2104DEV-100 3D2104 Development Kit
- 3D101DEV-200 3D101 Development Kit

PC Running Windows 10 with the following minimum configuration:

- 4 GB RAM (minimum)
- 10/40 (VM) GB available hard drive space (minimum)
- Ethernet (RJ45) port (or USB adapter)¹
- RS232 Port (or USB to serial adapter)
- Internet Access

2.2 Software

The following software packages are available on-line

2.2.1 Qt Installer (<u>https://www.qt.io/download</u>)

2.2.2 Grayhill Qt Support Files (<u>https://www.grayhill.com/qt43d</u>)

- QtGhInstall5122Linux
- QtGhInstall5122Win10.exe
- Virtual Machine Appliance (optional)

2.2.3 Windows Utilities

- Notepad++ (https://notepad-plus-plus.org/)
- PuTTY (<u>https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html</u>)

¹ An Ethernet port connected to a DHCP server connected to the 3Dxx display. This port should be on the same network as the development PC.

3. Installation

This is a brief overview of the installation steps for the Qt-based development environment for Grayhill 3Dxx displays.

- Connect the 3Dxx Development Kit hardware to the PC
- Qt Creator for Windows is downloaded and installed on the development PC
- Optional third party utilities are downloaded, installed, and configured (Windows)
- The serial and Ethernet links to the target 3Dxx display hardware are established.
- Grayhill support files are downloaded and installed
- A script is run to configure the target 3Dxx display board
- Instructions on how to open and run a Qt demonstration project on the 3Dxx display target hardware or desktop environment. This demonstration project illustrates:
- using touch screen "buttons"
- using touch screen swipes
- setting the 3Dxx backlight
- operating the 3Dxx camera input
- accessing and setting the real time clock

If a VM is going to be the development environment, it must be installed now. See [3] for complete instructions and configuration.

For VM installation, the procedure continues at 3.4 Configuring 3Dxx Display's IP Address.

3.1 Install the Development Kit

Connect the serial port and Ethernet port interfaces. The 3D50 display procedure is described in the document "3D50DEV Quick Start Guide.pdf" and the 3D70 in "3D70DEV Quick Start Guide.pdf"

3.2 Download and Install Qt Creator

N.B. This section is **mandatory** for Windows users. The Linux VM comes with Qt Creator pre-installed.

In this section, the Qt on-line installer will be downloaded and executed to download and install files from Qt. Once all the files are downloaded; Qt will be installed.

- Using your favorite web browser:
- <u>https://www.qt.io/download</u>
- scroll down and click "Go open Source"
- scroll down and click "Download"
- After the file downloads, open the downloads folder and double click on the file to execute the installer. If a "Security Warning" similar to below appears; click "Run"



Qt Setup
Welcome to the Qt online installer
This installer provides you with the option to download either an open source or commercial
version of Qt.
Commercial users: Please log in with your Qt Account credentials. Open source users: You have the option to log in using your Qt Account credentials (e.g.
your Qt Forum login). If you do not have a login yet, you have the option to create one in the next step.
Ot Account - Get access to a variety of services Packaging and pricing options
LGPL compliance & obligations Choosing the right license for your project
Settings Next Cancel

Click Next

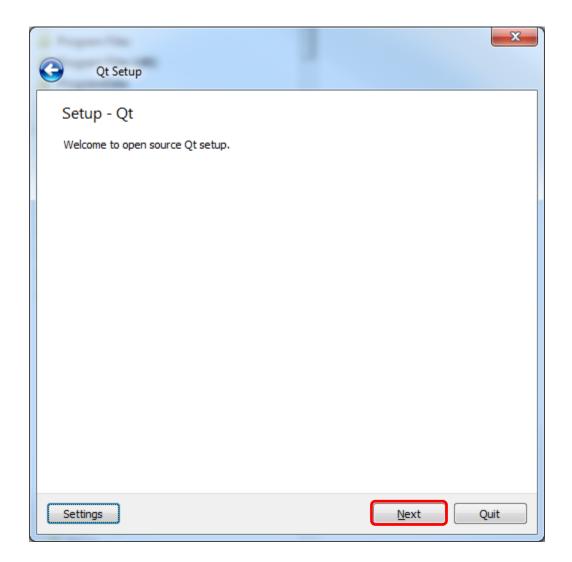
Please log in to Qt Account Login Email Password Ergot password? Need a Qt Account? Valid email address Password Confirm Password I accept the service terms.	x	×	Qt Setup) q
Login Email Password Forqot password? Need a Qt Account? Sign-up Valid email address Password Confirm Password		jin to everything Qt	ccount – Your unified log	Qt Acc
Password Forgot password? Need a Qt Account? Sign-up Valid email address Password Confirm Password			Please log in to Qt Account	
Forgot password? Need a Qt Account? Sign-up Valid email address Password			Email	Login
Need a Qt Account? Sign-up Valid email address Password			Password	
Sign-up Valid email address Password			Forgot password?	
Sign-up Valid email address Password			Need a Ot Account?	
Password Confirm Password		1		Sign-u
			-	
I accept the <u>service terms</u> .			Confirm Password	
			I accept the <u>service terms</u> .	
Settings Skip Car		Skip Cancel	-	Cattinger

• Create an account, if desired - otherwise click "Skip"

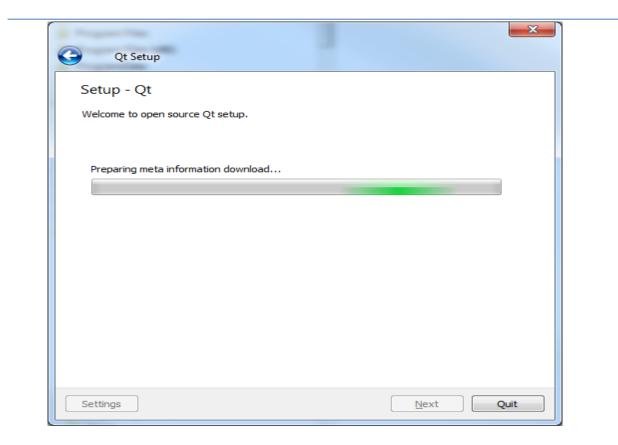
Qt	Setup
Qt Acco	ount – Your unified login to everything Qt
Login E	ease log in to Qt Account imail
	Need a Qt Account? John.Employee@company.com
	•••••••••• I accept the <u>service terms</u> .
Settings	Next Cancel

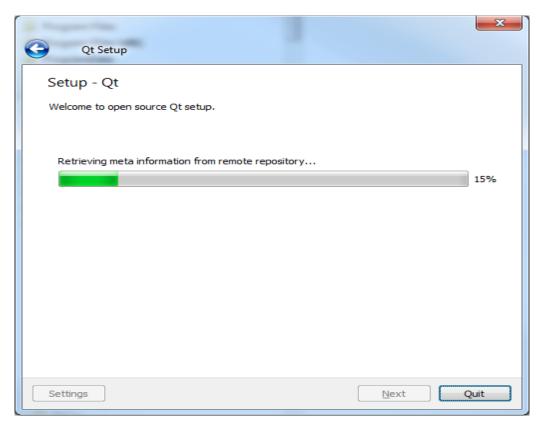
• If an account was created click "Next" - otherwise this screen will not appear

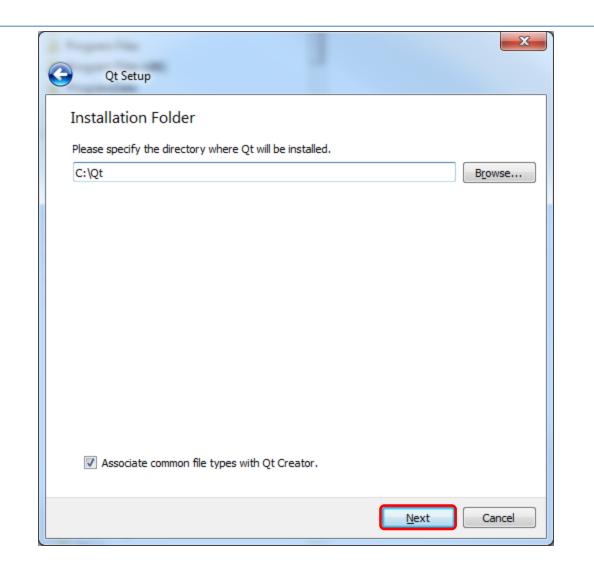
• Whether "Skip" or an account was created; installation continues here



• Click "Next"







• Click "Next"

N.B. Due to the nature of Qt and the way it stores configuration information; Qt must be installed in C:\Qt.

→ Qt 5.12.2 nGW 7.3.0 32bit"	
nGW 7.3.0 32bit"	Qt
 Android ARM64-v8a Android ARMv7 Sources Qt Charts Qt Data Visualization Qt Purchasing Qt Virtual Keyboard Qt WebEngine Qt Network Authorization Qt WebGL Streaming Plugin Qt Script (Deprecated) Qt Debug Information Files Developer and Designer Tools 	This component will occupy approximately 6.33 GB on your hard disk drive.
	Next Cancel

• Click "Next"

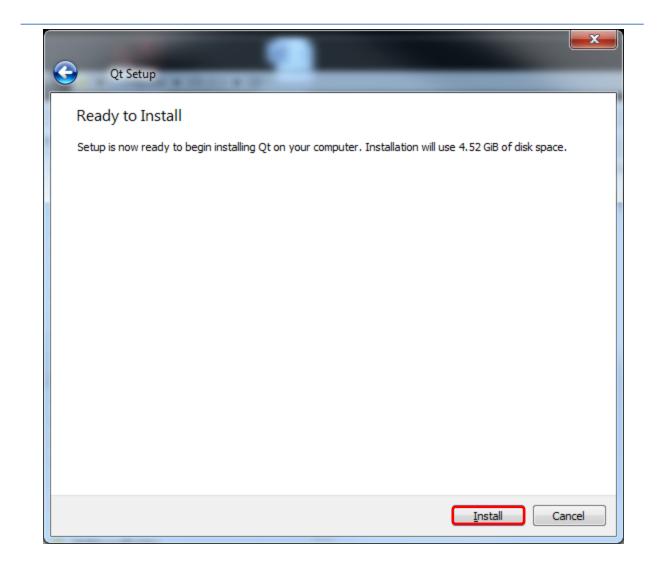
-	Qt Setup
Li	cense Agreement
	ease read the following license agreements. You must accept the terms contained in these agreements fore continuing with the installation.
C	Qt Installer LGPL License Agreement
N	ython Software Foundation License Version 2 /IICROSOFT SOFTWARE LICENSE TERMS MICROSOFT WINDOWS SOFTWARE DEVELOPMENT K
	ENERAL
-	ENERAL
o o a re Q d o	It is available under a commercial license with various pricing models and packages that meet a variety f needs. Commercial Qt license keeps your code proprietary where only you can control and monetize n your end product's development, user experience and distribution. You also get great perks like dditional functionality, productivity enhancing tools, world-class support and a close strategic elationship with The Qt Company to make sure your product and development goals are met. It has been created under the belief of open development and providing freedom and choice to evelopers. To support that, The Qt Company also licenses Qt under open source licenses, where most f the functionality is available under LGPLv3. It should be noted that the tools as well as some add-on
alic	omponents are available only under GPLv3. In order to preserve the true meaning of open development nd uphold the spirit of free software, it is imperative that the rules and regulations of open source censes are followed. If you use Qt under open-source licenses, you need to make sure that you comply <i>i</i> th all the licenses of the components you use.
	It also contains some 3rd party components that are available under different open-source licenses. lease refer to the documentation for more details on 3rd party licenses used in Qt.
G	PLv3 and LGPLv3
	GNU LESSER GENERAL PUBLIC LICENSE
	The Qt Toolkit is Copyright (C) 2017 The Qt Company Ltd. Contact: https://www.qt.io/licensing
0	You may use, distribute and copy the Qt GUI Toolkit under the terms of GNU Lesser General Public License version 3, which supplements GNU General Public License Version 3. Both of the licenses are displayed below.
) I have read and agree to the terms contained in the license agreements.
С	$\frac{1}{2}$ do not accept the terms and conditions of the above license agreements.
	Next Cancel

- If accepting of the license agreement select "I have read..."
- Click "Next"

Note: Qt is licensed under the terms of LGPL and GPL; these are opensource licensing agreements. Please reference <u>https://www1.qt.io/qt-</u><u>licensing-terms/</u> for a detailed explanation. Additional information is also located at <u>https://www.gnu.org/licenses/licenses.html</u>.

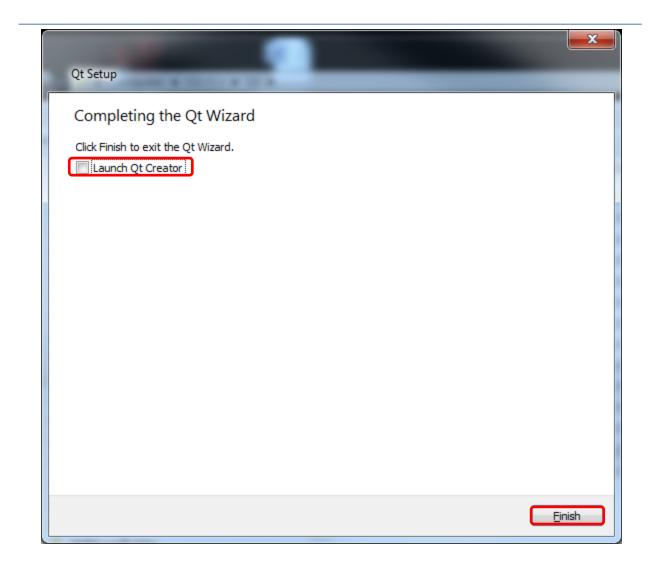
Start Menu shortcuts	
Select the Start Menu in which create a new directory.	you would like to create the program's shortcuts. You can also enter a name
Qt	
Accessories	
Administrative Tools	
JIRA	
Linaro GCC 4.8-2013.12	
Maintenance	
MinGw	
Python 3.6	
Qt	
Ruby 2.5.0-1-x64	
Startup	
TaskReporting	
Windows Virtual PC	

• Click "Next"



• Click "Install"

	\times
← Ot Setup	
← Qt Setup	
Installing Qt	
	13%
Downloading archive "4.9.2-0qtcreator.7z" for component Qt Creator 4.9.2.	
22.25 of 105.93 MB (8.05 MB/sec) - 10 second(s) remaining.	
Show Details	
	-
Install Can	:ei



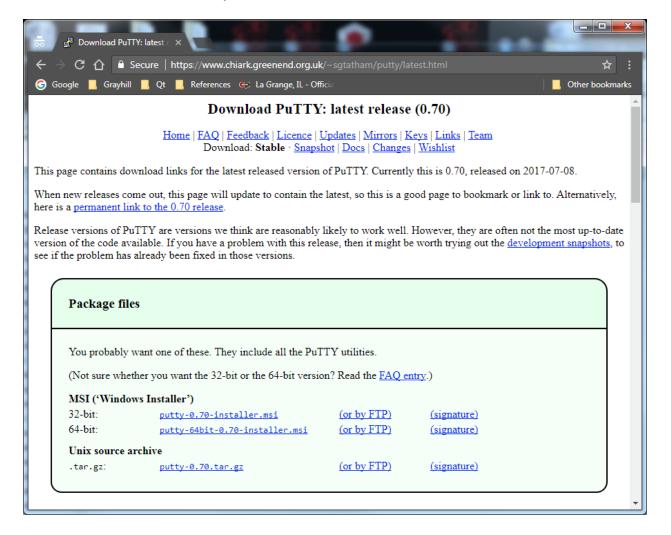
- Unselect "Launch Qt Creator"
- **N.B.** Qt Creator does **not** know the IP address of the target board at this time; the target board's IP address will be discovered and configured later. Any time the IP address of the display changes, Qt Creator must be relaunched if using the /etc/hosts file for IP address resolution.
- Click "Finish"

3.3 Windows Utilities

3.3.1 Download and Install PuTTY

The examples shown in this document reflect the use of PuTTY. Feel free to substitute a different client.

Download Putty



- Open the downloads folder and double click to execute the PuTTY installer
- Follow the installation instructions connection configuration is described later on in the document

3.4 Configuring 3Dxx Display's IP Address

In order to complete the setup of the Qt development environment for the 3Dxx Display hardware; the IP address assigned to the 3Dxx Display must be determined.

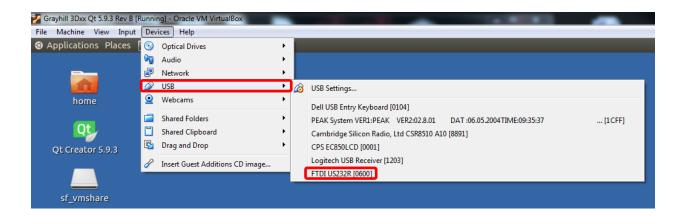
In order to perform these tasks, it is necessary to connect the 3Dxx Display to the same network as the development PC.

- Connect the 3Dxx display serial port to a serial port on the development PC
- Determine the serial port device name to use for PuTTY (serial communication between the physical PC and the target)

3.4.1 Linux

This depends on how the 3Dxx Display serial port is physically connected to the development PC. If using a built-in serial port on the development PC, the serial port device name is "/dev/ttyS0". If using a USB to serial port adapter, the serial port device name is "/dev/ttyUSB0".

Only if using the USB to serial port adapter: it must be activated at this time by clicking on the "Devices" menu option at the top of the VirtualBox screen. Select "USB" and click on the USB to serial port adapter device name in order to select it. A sample is shown here (the USB device name may be different than shown):



The Linux VM comes with PuTTY pre-installed, so minimal configuration is required. The "Serial line" value may need to be updated based on the above connecter.

• Launch PuTTY, the "PuTTY Configuration" screen appears

- Select COM1
- o Click "Load"

😣 PuTTY Config	uration	
Category: ▼ Session Logging ▼ Terminal Keyboard Bell Features ▼ Window Appearance Behaviour Translation Selection Colours Fonts ▼ Connection Data	Basic options for your PuTTY ses Specify the destination you want to connect Serial line /dev/ttyS0 Connection type: Raw Telnet Rlogin SSH Load, save or delete a stored session Saved Sessions COM1 Default Settings COM1 COM4 (USB)	
Proxy Telnet Rlogin ► SSH	Close window on exit: Always O Never O Only on cle	ean exit
About	Open	Cancel

o Click "Open"

3.4.2 Windows

- Launch PuTTY, the "PuTTY Configuration" screen appears configure as follows:
 - Select the "Serial" button
 - Set "Serial line" to appropriate COM Port
 - Change the "Speed" to 115000
 - Enter a name in "Saved Sessions" (e.g. comPort1)
 - Click "Save"

N.B. If "Open" is clicked any unsaved configuration modifications are lost!

Reputitive Configuration	2 ×
Category: Session Cogging Terminal Keyboard Bell Features Window Appearance	Basic options for your PuTTY session Specify the destination you want to connect to Serial line Speed COM1 115000 Connection type: State Raw Telnet Rlogin State Load, save or delete a stored session State State State
Behaviour Translation Translation Selection Colours Ornection Proxy Telnet Rlogin H SSH	Sav <u>e</u> d Sessions comPort1 Default Settings IP Address SerialCom1 Usb SerialCom4 comPort1 Defete
About <u>H</u> elp	Close window on e <u>xi</u> t: Always Never Only on clean exit <u>Open</u>

- Click on "Data"
- Set "Auto-login username" to "root"

Reputity Configuration		8 ×
Putty Configuration Category: □ - Session □ - Logging □ - Terminal □ - Keyboard □ - Bell □ - Features □ - Window □ - Appearance □ - Behaviour □ - Translation □ - Colours □ - Data □ - Proxy □ - Telnet □ - Rlogin □ - SSH □ - Serial	Data to se Login details Auto-login usemame When usemame is not sp Prompt OUse system Terminal details Terminal details Terminal speeds Environment variables Value	end to the server
About <u>H</u> elp		Open <u>C</u> ancel

ener baer en eesterrij aren ener eare agan	0	Click back on	"Session",	then cli	ick "Save"	again
--	---	---------------	------------	----------	------------	-------

Reputity Configuration	8 ×
Category:	
Category: Session Terminal Keyboard Bell Features Window Appearance Behaviour Translation Selection Colours Connection Data Proxy Telnet Rlogin SSH Serial	Basic options for your PuTTY session Specify the destination you want to connect to Serial line Speed COM1 115000 Connection type: Image: Speed Raw Image:
About <u>H</u> elp	<u>O</u> pen <u>C</u> ancel

• Lastly, click "Open" to establish a connection

3.4.3 Verification of Established Session

Make sure that the 3Dxx display is powered up and press the "Enter" key (Linux on left, Windows on right)

COM1 serial port to target	● 🛛 😣	B COM1 - PuTTY	
ghiimx6 login:		ghiimx6 login:	

- A "ghiimx6 login:" prompt should appear. If the 3Dxx display was just powered up; startup messages may appear as well, but when they are done, pressing the "Enter" key should produce a "ghiimx6 login:" prompt as shown.
- At the "ghiimx6 login:" prompt enter "root" (no password is required)
- Depending on the IP address type, refer to the appropriate appendix:
 - Dynamic Appendix H: Dynamic IP Address
 - Static Appendix I: Static IP Address

3.4.4 Configure IP address

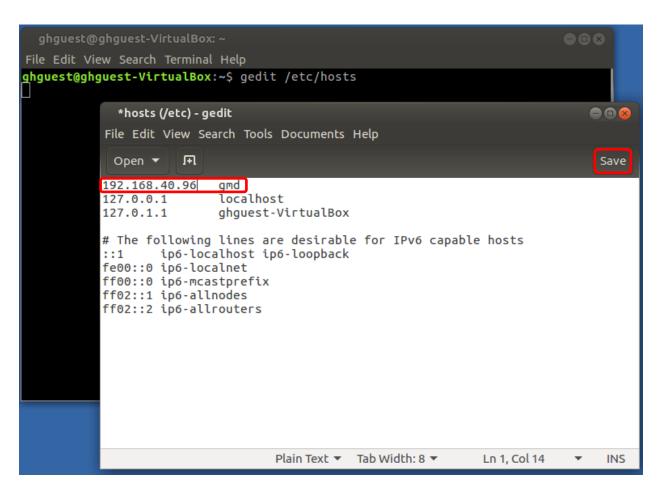
Create an alias for the display's IP address to be referenced by the host computer.

N.B. If the IP address of the display changes; *hosts* must be updated and Qt Creator re-launched.

Linux

- Launch a Terminal Command Window
- In the terminal window type the following command:
 gedit² /etc/hosts
- Update the IP address associated with "gmd"
- Click "Save"; then close the editing session

² vi is also available as a text editor if preferred



Windows

- Open Windows Explorer window (<Window>-e)
- Navigate to C: → Windows → System32 → drivers → etc and select "hosts"

	DS (C;)	▶ Windows ▶ System32	► drivers ► etc	- 49	Search etc	ـــــــــــــــــــــــــــــــــــــ
Eile Edit View Tools He	_					
Organize 👻 📄 Open 👻	Buri	n New folder				0
Resources SchCache	*	Name	Date modified	Туре	Size	
schemas		hosts	2/7/2018 2:47 PM	File	1 KB	
security		Imhosts.sam	6/10/2009 4:00 PM	SAM File	4 KB	
ServiceProfiles		networks	6/10/2009 4:00 PM	File	1 KB	
servicing		protocol	6/10/2009 4:00 PM	File	2 KB	
Setup		services	6/10/2009 4:00 PM	File	18 KB	
ShellNew						
SoftwareDistribution						
Speech						
symbols						
symbols						No preview
System32	_					available.
drivers						
ar-SA	=					
s-CZ						
a-DK						
de-DE						
el-GR						
en-US						
es-ES						
etc						
fi-FI	-					
1 item selected	Ŧ				📜 Compu	ter

- Right click to edit the file using your favorite flavor of editor (Screenshot illustrates Notepad++)
- After the editor is launched, Windows Explorer can be closed
- Add the IP address and "gmd" as illustrated below:

```
- -
                                                                                           X
C:\Windows\System32\drivers\etc\hosts - Notepad++ [Administrator]
<u>File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?</u>
                                                                                              х
                                                                                             >>
 😑 hosts 🖸 🚞 config3Dxx-Qt-5.6.2.sh 🛛 블 3Dxx-Qt-5.6.2-qmake.conf 🖸 블 README.txt 🛛 블 QtLibrarySetup3Dxx-5.9.1 🟹 블 config.sh 🔀 🚺 🔹
      # Copyright (c) 1993-2009 Microsoft Corp.
  2
  3
      # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
  4
      # This file contains the mappings of IP addresses to host names. Each
  5
  6
      # entry should be kept on an individual line. The IP address should
  7
      # be placed in the first column followed by the corresponding host name.
  8
      # The IP address and the host name should be separated by at least one
  9
      # space.
  10
      # Additionally, comments (such as these) may be inserted on individual
  11
 12
      # lines or following the machine name denoted by a '#' symbol.
 13
      # For example:
 14
 15
      #
 16
      #
             102.54.94.97
                             rhino.acme.com
                                                      # source server
 17
             38.25.63.10
                           x.acme.com
                                                      # x client host
      ±.
 18
      # localhost name resolution is handled within DNS itself.
 19
 20
          127.0.0.1
                         localhost
      #
 21
                          localhost
          ::1
     192.168.40.118 gmd
 22
Norr length: 842 lines: 22
                            Ln:22 Col:15 Sel:0|0
                                                                                         INS
                                                             Windows (CR LF)
                                                                          UTF-8
```

• Save the file

N.B. The editor may ask to restart in admin mode; allow it to continue as *hosts* is a system file

3.5 Download and Install Support Files

This section details downloading and installation of the necessary Qt support files. It also describes configuration of the host machine and 3Dxx display for operation with the Qt development environment. The scripts work for all display models 3D50, 3D70, 3D2104, and 3D101.

- Launch an internet browser
- Navigate to <u>https://www.grayhill.com/qt43d</u>

3.5.1 Linux

N.B. Firefox can be launched from Applications \rightarrow Internet

- Download QtGhInstall5122Linux
- Copy/move the downloaded file to /home/ghguest
- Open a terminal window and cd to home (cd)
- Make QtGhInstall5122Linux executable (it should already be executable)
 o chmod 755 QtGhInstall5122Linux
- Unarchive the files (self-extracting archive)
 ./QtGhInstall5122Linux
- Make installation script executable
 chmod 755 QtGhInstallLinuxInstall
- Install Qt support files on display
 ./QtGhInstallLinuxInstall

The above script without any arguments defaults to updating both the VM and the display. To update additional displays, connect the 3Dxx and update **gmd** (see previous section) with the IP address then re-run the installation script to configure the display.

• ./QtGhInstallLinuxInstall 3dxx

The script reboots the display and requires a few minutes to complete execution. If everything works correctly these are the last few lines:

```
setup3Dxx completed successfully... rebooting
Wed Apr 4 13:34:24 CDT 2018
```

If a message similar to this does not appear, the problem(s) **must** be corrected before continuing.

The following files/directories are created on the VM:

- GrayhillDisplayPlatform <dir> sysroot for cross-compiling
- GrayhillExamples <a> <a> <a> <a><

- QtGhInstallLinuxInstall <file> installation script (re-run for additional displays)
- targetFiles

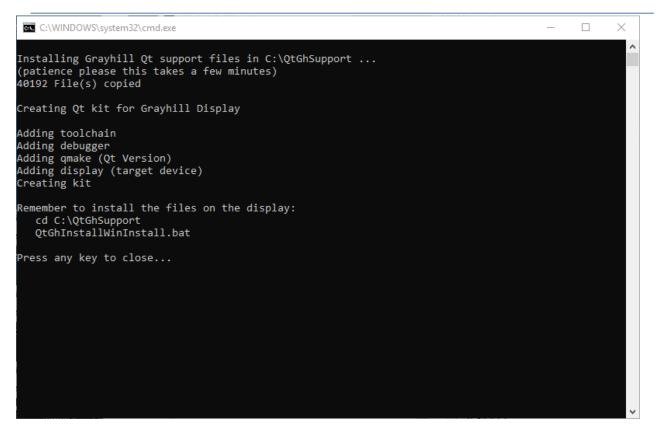
3.5.2 Windows

- Download "Qt Creator Windows Support Files" from the Grayhill website
- Open the download folder and double click on "QtGhInstall5122Win10.exe"
- A User Account Control window may pop-up
 Click "Yes" to allow the self-extracting zip file to proceed
- The following window appears

Grayhill Q	t 5.12.2 Support Files	\times
?	Do you want to install the Grayhill Qt 5.12.2 support files?	
	Yes No	

o Click "Yes"

4 11% Extracting	×
	Cancel



• Using Windows Explorer; navigate to "C: QtGhSupport" and verify the folder was installed

📕 🗹 📕 🖛 Qt0	GhSupport							_	
File Home	Share View								~ 🕐
access	Paste Copy path Paste Paste shortcut	Move to * Copy to * Organiz	elete Rename Ve	Rew item ▼ Easy access ▼ New	Properties	Select all Select none Invert selection Select			
$\leftarrow \rightarrow \cdot \uparrow$		QtGhSupport		1400	open	Select	5 V	Search QtGhSupport	م
🖈 Quick access	Name	^	Date modified 8/7/2019 1:29 PM	Type File folder	Size			Search Gensepport	
OneDrive		naro-2013	8/7/2019 1:29 PM	File folder					
💻 This PC		illDisplayPlatform illExamples	8/7/2019 1:29 PM 8/7/2019 1:30 PM	File folder File folder					
💣 Network	Perl64		8/7/2019 1:30 PM	File folder					
	Pythor		8/7/2019 1:30 PM 8/7/2019 1:30 PM	File folder File folder			Select a file	to preview.	
	🔒 targeti 💿 addKit		8/7/2019 1:30 PM 7/23/2019 2:49 PM	File folder Windows Batch Fi	ile 4 KB				
		nstallWinInstall.bat	8/6/2019 9:57 AM	Windows Batch Fi					
	<				>				
10 items									

• Double click on "QtGhInstallWinInstall.bat" to configure the display

C:\Windows\system32\cmd.exe

Ensuring display file system is writable... The authenticity of host 'gmd (192.168.5.70)' can't be established. RSA key fingerprint is SHA256:DqmL/DR2Jk4/Klvji+cGNGs2J88o//GH8LplHzRlAd8. Are you sure you want to continue connecting (yes/no)?

```
C:\WINDOWS\system32\cmd.exe
```

Ensuring display file system is writable... Transferring files to display... (patience please this takes a few minutes) Configuring the display... Configuring Grayhill 3Dxx Display for Qt-5.12.2 Wed Aug 7 13:57:02 CDT 2019 Disabling VUI Builder and Codesys applications on <u>3Dxx</u> Updating GCC Libraries on 3Dxx Display Upgrading gcclibs on root from 4.7.3 to 4.8.3... Removing obsolete file /usr/lib/libstdc++.so.6.0.17. Configuring gcclibs. Updating Grayhill I/O Libraries on 3Dxx Display Installing ghdrv-lib (1.1) to root... Installing ghdrv-lib (1.1) to root... Configuring ghdrv-lib (1.1) to root... Updating glibc Libraries on 3Dxx Display Upgrading glibc on root from 2.16.0 to 2.18.0... Removing obsolete file /lib/libnss_compat-2.16.so. Removing obsolete file /lib/libutil-2.16.so. Removing obsolete file /lib/libutil-2.16.so. Removing obsolete file /lib/libutil-2.16.so. Removing obsolete file /lib/libnsl-2.16.so. Removing obsolete file /lib/libnsl-2.16.so. Removing obsolete file /lib/libnsl-2.16.so. Removing obsolete file /lib/libnsl-2.16.so. Removing obsolete file /lib/libnss_files-2.16.so. Removing obsolete file /lib/libnss_files-2.16.so. Removing obsolete file /lib/libnss_files-2.16.so. Removing obsolete file /lib/libnss_dns-2.16.so. Removing obsolete file /lib/libres0lv-2.16.so. Removing obsolete file /lib/libres0lv-2.16.so. Removing obsolete file /lib/libres0lv-2.16.so. Removing obsolete file /lib/libres0lv-2.16.so. Removing obsolete file /lib/librt-2.16.so. Configuring glibc. Updating GPU Libraries on 3Dxx Display Installing gpu-viv2 (3.0.101+4.1.1) to root... Configuring gpu-viv2. Installing Qt5 Libraries on 3Dxx Display No packages removed. No packages removed. Installing qt5122 (5.12.2) to root... Configuring qt5122. bootargs already set to console=ttymxc0,115200 lpj=7905280 rootfstype=ext4 root=/dev/mmcblk0p1 ro rootwait board-ghi_imx 6.pn=3D70VT-100 quiet Updating /etc/profile.local script Updating /usr/lib/fonts setup3Dxx completed successfully... rebooting Wed Aug 7 13:58:47 CDT 2019 Making 3Dxx Flash File system writeable and adding writeablefs script Rebooting to make following steps run faster Press any key to close...

• Restore any custom modifications. The setup script preserves files by appending a timestamp

 \times

3.6 Build and Run 3Dxx Embedded Application

This section details how to build and run a demo application.

A Qt QML demonstration project is provided which runs (configured as necessary) on each of the 3Dxx displays as well as the host machine.

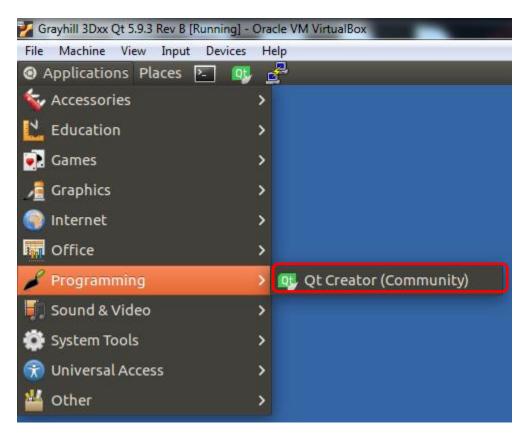
Complete configuration instructions are in the appendices.

3.6.1 Launch Qt Creator

Linux

Launch Qt Creator using one of the following methods:

• Select "Applications" (upper left-hand corner of the Linux window), then navigate through "Programming" and click on "Qt Creator ..."



• Click on the Qt icon in the panel



• Double click on the Qt icon on the desktop



Windows

- Launch Windows Explorer (<Windows>-e)
- Navigate to C: \rightarrow Qt \rightarrow Tools \rightarrow QtCreator \rightarrow bin \rightarrow qtcreator.exe

) ► Qt ► Tools ► QtCreator ► bin ►		- €9	Search bin		× ם م
<u>File Edit View Iools H</u> elp			· · · · · ·	Justanour		~
Organize ▼ 🐻 Open ▼ Bu	rn New folder					
translations	Name	Date modified	Туре	Size	A	
ist 🔒	S Qt5Svg.dll	12/3/2017 8:35 PM	Application extens	259 KB		
Docs	Qt5Test.dll	12/3/2017 8:35 PM	Application extens	156 KB		
Examples	Sector State S	12/3/2017 8:35 PM	Application extens	4,300 KB		
Licenses	S Qt5Xml.dll	12/3/2017 8:35 PM	Application extens	147 KB		
QtSupport	Qt5XmlPatterns.dll	12/3/2017 8:35 PM	Application extens	2,185 KB		
Jgcc-linaro-2013	💷 qtcdebugger.exe	12/3/2017 8:29 PM	Application	33 KB		
GrayhillExamples	😳 qtcreator.exe	12/3/2017 8:12 PM	Application	809 KB		
targetSysroot	qtcreator_ctrlc_stub.exe	12/3/2017 8:08 PM	Application	12 KB		
퉬 Tools	💷 qtcreator_process_stub.exe	12/3/2017 8:08 PM	Application	13 KB		
鷆 mingw530_32	🚳 QtcSsh4.dll	12/3/2017 8:09 PM	Application extens	1,481 KB		
🍌 bin	💷 qtdiag.exe	12/3/2017 8:35 PM	Application	52 KB		No previev available.
🍌 etc	💷 qtpromaker.exe	12/3/2017 8:29 PM	Application	37 KB		available
퉬 i686-w64-mingw3	sdktool.exe	12/3/2017 8:29 PM	Application	237 KB		
🌗 include	🚳 Sqlite4.dll	12/3/2017 8:08 PM	Application extens	759 KB		
🌗 lib	🚳 ssleay32.dll	12/3/2017 8:35 PM	Application extens	268 KB		
🌗 libexec	🚳 Timeline4.dll	12/3/2017 8:11 PM	Application extens	212 KB		
🌗 licenses	🚳 Utils4.dll	12/3/2017 8:09 PM	Application extens	1,411 KB		
퉬 opt	win32interrupt.exe	12/3/2017 8:39 PM	Application	97 KB		
🌗 share	win64interrupt.exe	12/3/2017 8:42 PM	Application	112 KB	=	
QtCreator	💷 winrtdebughelper.exe	12/3/2017 8:29 PM	Application	11 KB		
]] bin	🚳 xgejom.bat	12/3/2017 8:37 PM	Windows Batch File	1 KB		
퉬 lib 👻	🖭 xgejom.xml	12/3/2017 8:37 PM	XML Document	1 KB	-	2
em selected					💵 Computer	

• Right click to select options like

- "Send to" \rightarrow Desktop (create shortcut)
- Double click to launch Qt Creator

3.6.2 Open project

• Select "Projects"

07 Qt Creator		
<u>File E</u> dit <u>B</u> u	iild <u>D</u> ebug <u>A</u> nalyze <u>T</u> ools <u>W</u> in	dow <u>H</u> elp
Welcome	Projects	+ New Project
Edit	Examples	Sessions Recent Projects
Design	Tutorials	1 Default (last session)
الله Debug	New to Qt?	
Projects ? Help	Learn how to develop your own applications and ≡ explore Qt Creator.	
ныр	Get Started Now	
_	L Ot Account	
	Online Community	
	Blogs User Guide	
	₽, Type to locate (Ctrl+K)	1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 Debugger Console 8 Test Results 🗢 🔺 🗍 🍃

- Click on "Open Project" ("Welcome" should be automatically selected on launch)
- Navigate to the desired project

Linux

/home \rightarrow GrayhillExamples \rightarrow ghQmlDemo4115

Windows

C: \rightarrow QtGhSupport \rightarrow GrayhillExamples \rightarrow ghQmlDemo4115

🔍 Open File					×
← → • ↑ 📙 « (QtGh!	Support > GrayhillExamples > ghQmlDem	o4115 ~ 0	ン シ Search g	hQmIDemo4115
Organize 🔹 New fol	lder				•== • •
	^	Name	Date modified	Туре	Size
🗊 3D Objects		📜 qml	12/11/2020 8:58 AM	File folder	
E Desktop		🚾 ghQmlDemo4115.pro	12/8/2020 10:38 AM	Qt Project file	4 KB
Documents					
🖶 Downloads					
👌 Music					
Pictures					
📑 Videos					
🐛 OS (C:)	~				
File <u>n</u>	<u>a</u> me:	ghQmlDemo4115.pro		 All Projects (*. 	pro *.qbs *.pyproj 🗸
				<u>O</u> pen	Cancel

- Select ghQmlDemo4115.pro
- Click "Open"
- If a similar box appears, click "Yes"

🞯 Settin	gs File for "gh7indemo" from a different Environment?	×
?	No .user settings file created by this instance of Qt Creator was found. Did you work with this project on another machine or using a different settings path before?	;
	Do you still want to load the settings file "C:\QtGhSupport\GrayhillExamples \gh7indemo\gh7indemo.pro.user"?	
	Yes No	ב

• If a similar box appears, click "OK". **Refer to Appendix B: Configuring a 3Dxx Project before continuing.** The current project configuration file is not compatible with the current version of Qt Creator and the project's settings need to be re-configured.

😳 No Va	alid Settings Found	×
	No valid settings file could be found.	
	All settings files found in directory "C:\QtGhSupport\GrayhillExamples \gh7indemo" were either too new or too old to be read.	
	0	K

Qt Creator uses the kit Qt-5.9.3-3Dxx to parse the project.	
Type to filter kits by name	
✓ Select all kits	
✓ 🖵 Desktop Qt 5.12.2 GCC 64bit	Details •
☑ 및 Qt-5.12.2-3Dxx	Details •
Import Build From	Details •

3.6.3 Build Project

- Select "Projects" view
- Select "Build" under "Qt-5.12.2-3Dxx"³
- Expand qmake
- Verify qmake "Additional arguments:" is set to: "hw_present=yes target=3D70 kernel=4".

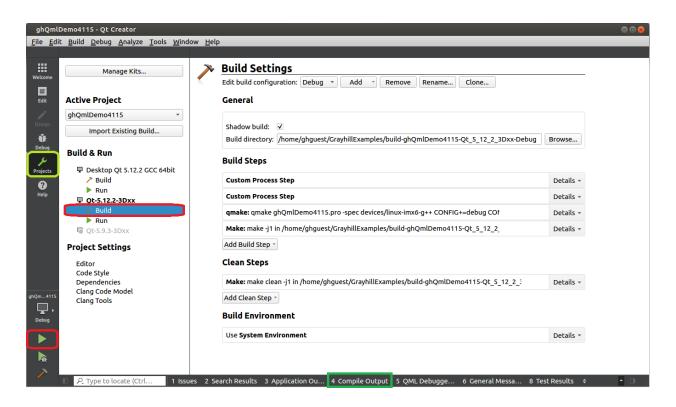
N.B. – for the **target** argument, use 3D70, 3D50, 3D2104, or 3D101 based on actual display.

N.B. – the **kernel=4** argument is used for 3Dxx display models running linux kernel 4.1.15

N.B. – the **hw_present** argument must not be present for Desktop builds

N.B. – for Windows builds of the demo, the additional argument **windowsOnly=true** needs to be set

³ To build for the desktop select "Build" under "Desktop" Certain features are not supported (e.g. Camera)



 Click on the green arrow to run (a check to see if the executable is up to date is performed; if compilation is necessary the output can be viewed by clicking on the "Compile Output" tab)

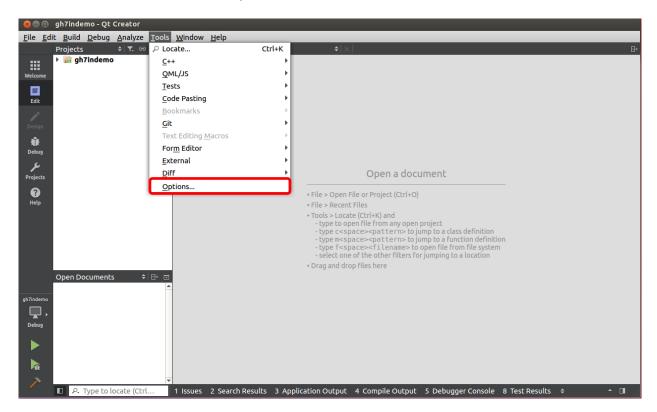
ghQml	Demo4115 - Qt	Creator										
<u>F</u> ile <u>E</u> di	t <u>B</u> uild <u>D</u> ebug	<u>A</u> nalyze	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp							
	Projects	🗧 🕈 💭 👄			<no documer<="" th=""><th>nt></th><th></th><th></th><th></th><th></th><th></th><th>8+</th></no>	nt>						8+
Welcome	🕨 📠 ghQmlDei	mo4115										
Edit							Oper	n a docume	ent			
						• File > Oper • File > Rece	File or Projec	ct (Ctrl+O)				
🐞 Debug						• Tools > Loo - type to o	ate (Ctrl+K) ar	any open proje				
پر Projects						- type m< - type f<	pace> <patt pace><file< th=""><th>ern> to jump t ename> to open</th><th>o a class definit o a function def file from file sy</th><th>finition stem</th><th></th><th></th></file<></patt 	ern> to jump t ename> to open	o a class definit o a function def file from file sy	finition stem		
? Help							ne of the othe rop files here		ping to a locatio	חכ		
ghQm4115												
Γ.	Open Documen	ts \$	8+ 📼	Applicatio	on Output		 + −					^ 🖂
Debug			^		•		Pate					~ 🖻
				Backlight	emo4115 (on 3Dx On - 100 ial output: 0x0 i							-
ka →					User requested s Timeout waiting			ish.				
	■ P. Type to	locate (Ctrl		1 Iss 🗈	12 2 Search	3 Applica	4 Compil	5 QML D	6 Genera	8 Test Re	¢	▲ 🔲

- Select the "Application Output" tabClick the red (when application is running on target) square to terminate the target session

Appendix A: Configuring a Manual Qt Kit for Grayhill Displays

N.B. This appendix is included for reference and is not a required installation step. Grayhill automatically installs the kit configuration as part of the support file installation. A kit is a collection of utilities (qmake, compilers, debugger, etc...) used to build a project.

- Launch Qt Creator
- Select Tools → Options



Alternatively, "Manage Kits" can be selected from the "Projects" view.

General

The "General" tab is where project wide customization is done. Review and select the desired configuration.

8 Options						
Filter	Build & Run					
Environment	General Kits Qt Versions Compilers Debuggers Qbs CMake					
Text Editor	Projects Directory					
FakeVim	O Current directory					
Help	Directory /home/ghguest Browse					
{} c++	Build and Run					
🗸 Qt Quick	☐ Save all files before build					
🚯 Build & Run	✓ Always build project before deploying it					
🔎 Debugger	✓ Always deploy project before running it ✓ Word-wrap application output					
💥 Designer	Open Compile Output pane when building					
Malyzer	✓ Open Application Output pane on output when running					
Version Control	Open Application Output pane on output when debugging					
Devices	Always ask before stopping applications					
🛐 Code Pasting	Stop applications before building: None 👻					
QA Test Settings	Default build directory: tProject:Name}-%{CurrentKit:FileSystemName}-%{CurrentBuild:Name}")}					
✓ Apply X Cancel ✓ QK						

- Select "Build & Run"
- Select "General" tab
- Click "Apply" to continue and select other tabs, "OK" if finished

Device

The section describes how to establish an Ethernet based connection to the display.

- Select "Devices"
- Click "Add..."

.

or Options	The second se	×
Filter	Devices	
Environment	Devices Android QNX	
Text Editor	Device: Local PC (default for Desktop)	<u>A</u> dd
FakeVim	General	Remove
Help	Name: Local PC Type: Desktop	Set As Default
() C++	Auto-detected: Yes (id is "Desktop Device")	Show Running Processes
Qt Quick	Current state: Unknown	
🕕 Build & Run	Type Specific	
Debugger		
🔀 Designer		
Analyzer		
Version Control		
Devices		
Code Pasting		
QA Testing		
	ОК	Cancel Apply

📴 Device Configuration Wizard Selection
Available device types:
Generic Linux Device
QNX Device
Start Wizard Cancel

- Select "Generic Linux Device"
- Click "Start Wizard"

😣 New Generic Lin	ux Device Configuration Setup		
Connection			
Connection	The name to identify this configuration:	3Dxx Target	
Summary	The device's host name or IP address:	gmd	
	The username to log into the device:	root	
	The authentication type:	ullet Password $igodot$ Key $igodot$ Agent	
	The user's password:		
	The file containing the user's private key:	/ghguest/.ssh/id_rsa Browse	
		<u>N</u> ext > Cancel	

- Populate the fields as illustrated above
- **N.B.** The IP address associated with **gmd** is located in */etc/hosts* (Linux) and *C:\Windows\System32\drivers\etc* (Windows)
- Click "Next"

😵 New Generic Li	8 New Generic Linux Device Configuration Setup				
Summary					
Connection	The new device configuration will now be created. In addition, device connectivity will be tested.				
> Summary					
	Rack Fisikh Concel				
	< <u>B</u> ack <u>Finish</u> Cancel				

- Verify the 3Dxx Display is still powered up
- Click "Finish" The Ethernet link to the 3Dxx Display will be tested and if successful the following result screen appears

😣 Device Test
Connecting to host Checking kernel version Linux 3.0.35 armv7l
Checking if specified ports are available All specified ports are available.
Device test finished successfully.
X Close

Click "Close"

8 Options	
Filter	Devices
Environment	Devices Android QNX
Text Editor	Device: 3Dxx Target (default for Generic Linux)
🚮 FakeVim	General <u>R</u> emove
Help	Name: 3Dxx Target Set As Default
{} c++	Type: Generic Linux Test
🚄 Qt Quick	Auto-detected: No Show Running Processes
🚯 Build & Run	Current state: Unknown Deploy Public Key
🔍 Debugger	Type Specific
💥 Designer	Machine type: Physical Device
Analyzer	Authentication type: Password Key Key via ssh-agent
Version Control	Host name: gmd SSH port: 22 🗘 Check host key
Devices	Free ports: 10000-10100 Timeout: 20s
Code Pasting	Username: root
QA Test Settings	Password: Show password Private key file: Browse
	✓ Apply X Cancel

• Click the upper arrow on the right side of the "Timeout:" box to increase timeout value to "20s"

Devices Summary

8 Options	
Filter	Devices
Environment	Devices Android QNX
Text Editor	Device: 3Dxx Target (default for Generic Linux)
🕌 FakeVim	General <u>R</u> emove
Help	Name: 3Dxx Target Set As Default
{} c++	Type: Generic Linux Test
📣 Qt Quick	Auto-detected: No Show Running Processes
🕕 Build & Run	Current state: Unknown Deploy Public Key
🔍 Debugger	Type Specific
💢 Designer	Machine type: Physical Device
🛄 Analyzer	Authentication type: • Password · Key · Key via ssh-agent
Version Control	Host name: gmd SSH port: 22 🗘 🖸 Check host key
Devices	Free ports: 10000-10100 Timeout: 20s
Code Pasting	Username: root
QA Test Settings	Password:
	Private key file: Browse Create New
	✓Apply X Cancel ✓QK

- Name name of the device
- Host name **gmd** alias -- specified in **hosts**
- Timeout 20s
- Username root

N.B. Remember verify connectivity using "Test"

Compiler

Select "Build & Run"

Select "Compilers" tab

Click "Add"; then select GCC \rightarrow C

Options		×
Iter	Build & Run	
Environment	General Kits Qt Versions Compilers Debuggers Qbs CMake	
Text Editor	Name Туре	Add 🔻
	V Auto-detected	MinGW 🕨
FakeVim	MinGW 5.3.0 32bit for C MinGW	GCC 🔸
Help	✓ C++	Clang 🔸
[]} C++	MinGW 5.3.0 32bit for C++ MinGW	Custom 🔸
	V Manual C	QCC
Qt Quick	C++	
🕕 Build & Run		
Debugger		
🗶 Designer		
Analyzer		
Version Control		
Devices		
Code Pasting		
	OK Cancel	Apply

Populate the fields as illustrated

- "Name:" ARM-GCC
- "Compiler path:" Click "Browse..." and navigate to the desired file
 - /opt/OSELAS.Toolchain-2013.12.3/arm-cortexa9-linux-gnueabi/gcc-4.8.3-glibc-2.18-binutils-2.24-kernel-3.12-sanitized/bin/arm-cortexa9linux-gnueabi-gcc

Recent	📢 🖸 opt OSELAS.Toolchain-2013.12.3 arm-cortexa9-linux-gnueabi gcc-4.8.3-glibc-2.18-binutils-2.24-kernel-3.12	-sanitized	oin 🕨
Home	Name	Size	Modified
Desktop	🗇 arm-cortexa9-linux-gnueabi-addr2line	4.2 MB	4 Dec
Documents	🗇 arm-cortexa9-linux-gnueabi-ar	4.3 MB	4 Dec
-		6.3 MB	4 Dec
 Downloads 		2.8 MB	4 Dec
J Music		4.1 MB	4 Dec
Pictures	🗇 arm-cortexa9-linux-gnueabi-cpp	2.8 MB	4 Dec
Pictures		108.3 kB	
Videos		2.8 MB	4 Dec
sf vmshare	arm-cortexa9-linux-gnueabi-gcc	2.8 MB	4 Dec
	gcc-4.8.3	2.8 MB	4 Dec
Other Locations	♦ arm-cortexa9-linux-gnueabi-gcc-ar	151.2 kB	
		150.9 kB	
		150.9 kB	4 Dec
			All Files 🔻

• C:\QtGhSupport\gcc-linaro-2013\bin\ arm-linux-gnueabi-gcc.exe

÷ → × ↑ 📴 > '	This PC > OS (C:) > QtGhSupport > gcc-lina	ro-2013 > bin		ע ט Searc	h bin	Q
Organize 🔻 New fo	lder					?
	Name	Date modified	Туре	Size		
📌 Quick access	📧 arm-linux-gnueabi-dwp.exe	1/18/2018 7:26 PM	Application	3,837 KB		
🝊 OneDrive	📧 arm-linux-gnueabi-elfedit.exe	1/18/2018 7:26 PM	Application	965 KB		
	📧 arm-linux-gnueabi-g++.exe	1/18/2018 7:26 PM	Application	619 KB		
💻 This PC	📧 arm-linux-gnueabi-gcc.exe	1/18/2018 7:26 PM	Application	617 KB		
鹶 Network	📧 arm-linux-gnueabi-gcc-4.8.3.exe	1/18/2018 7:26 PM	Application	617 KB		
-	📧 arm-linux-gnueabi-gcc-ar.exe	1/18/2018 7:26 PM	Application	42 KB		
File	name: arm-linux-gnueabi-gcc.exe			~ All F	iles (*)	~

Click "Open"

"ABI:" Select "arm-linux-generic-elf-32bit"

The configuration portion of the screen should look similar to:

😳 Options		×
Filter Build & Run		
Environment General Kits	Qt Versions Compilers Debuggers Qbs CMake	
Text Editor	Type	^ Add ▼
FakeVim V C		Clone
P Help AF	nGW 5.3.0 32bit for C MinGW M GCC (Linaro 2013) GCC	Remove
	nGW 5.3.0 32bit for C++ MinGW	
AF Qt Quick	M G++ (Linaro 2013) GCC	
Build & Run G	C GCC	~
Debugger		
Mame:	ARM-GCC	
Compiler path:	C:\QtGhSupport\gcc-linaro-2013\bin\arm-linux-gnueabi-gcc.exe	. .
Platform code	en flags:	
Version Control Platform linker	flags:	
Devices <u>A</u> BI:	arm-linux-generic-elf-32bit 🔻 arm 🔻 - linux 🔻 - generic 🔻 - elf 🔻 - 32bit	~
Code Pasting		
	OK Ca	ancel Apply

Repeat the above steps for GCC \rightarrow C++

📴 Options	×
Filter Build & Run	
Environment General Kits Qt Versions Compilers Debuggers Qbs CMake	
Text Editor ARM GCC (Linaro 2013) GCC	^ Add ▼
FakeVim C++ MinGW 5.3.0 32bit for C++ MinGW	Clone
PHelp ARM G++ (Linaro 2013) GCC	Remove
Qt Quick GCC GCC	
GCC GCC	~
Debugger	
Designer Name: ARM-G++	
C:\QtGhSupport\gcc-linaro-2013\bin\arm-linux-gnueabi-c+	++.exe Browse
Version Control Platform linker flags:	
Devices ABI: arm-linux-generic-elf-32bit 🔻 arm 🍸 - linux	🔻 - generic 🔻 - elf 🖤 - 32bit 🔻
Code Pasting	
	OK Cancel Apply

Click "Apply"

Debugger

Select the "Debuggers" tab

Click "Add"

Populate the fields as illustrated

- "Name:" 3Dxx Target Debugger
- "Path:" Click "Browse..." and navigate to the desired file (should be previous directory)
 - /opt/OSELAS.Toolchain-2013.12.3/arm-cortexa9-linux-gnueabi/gcc-4.8.3-glibc-2.18-binutils-2.24-kernel-3.12-sanitized/bin/arm-cortexa9linux-gnueabi-gcc

8	Choose Executa	ble		
0	Recent	• 🕑 opt OSELAS.Toolchain-2013.12.3 arm-cortexa9-linux-gnueabi gcc-4.8.3-glibc-2.18-binutils-2.24-kernel-3.12-s	anitized b	oin ►
ŵ	Home	Name	Size	Modified
	Desktop	◊ arm-cortexa9-linux-gnueabi-gcc	2.8 MB	4 Dec
р	Documents		2.8 MB	4 Dec
Ц	Documents	🚸 arm-cortexa9-linux-gnueabi-gcc-ar	151.2 kB	
÷	Downloads	🚸 arm-cortexa9-linux-gnueabi-gcc-nm	150.9 kB	4 Dec
99	Music	🚸 arm-cortexa9-linux-gnueabi-gcc-ranlib	150.9 kB	4 Dec
		🗇 arm-cortexa9-linux-gnueabi-gcov	1.6 MB	4 Dec
Ō	Pictures	🔹 arm-cortexa9-linux-gnueabi-gdb	25.8 MB	4 Dec
H	Videos	🚸 arm-cortexa9-linux-gnueabi-gprof	4.7 MB	4 Dec
_	cf. ymchara	♦ arm-cortexa9-linux-gnueabi-ld	5.8 MB	4 Dec
-	sf_vmshare		5.8 MB	4 Dec
+	Other Locations	arm-cortexa9-linux-gnueabi-nm	4.2 MB	4 Dec
	Other Eocacions	arm-cortexa9-linux-gnueabi-objcopy	5.2 MB	4 Dec
		🚸 arm-cortexa9-linux-gnueabi-objdump	6.0 MB	4 Dec
		arm-cortexa9-linux-onueahi-ranlih	4 3 MR	4 Dec
			F	All Files 🔻
		C	ancel	Open

• C:\QtGhSupport\debugger\arm-linux-gnueabi-gcc.exe

Choose Executable								>
← → · ↑ → This PC → OS (C:) → QtGhSupport → debugger → · ♥ ♂ Search of								م
Organize 🔻 🛛 New f	older				ļ	== -		?
1 Ouish serves	^ Name	Date modified	Туре	Size				
📌 Quick access	lib	3/13/2018 1:13 PM	File folder					
📤 OneDrive	🔳 gdb-arm-none-linux-gnue	abi.exe 4/21/2012 10:27 AM	Application	3,618 KB				
This PC	libexpat-1.dll	4/21/2012 10:27 AM	Application extens	429 KB				
= 11131 C	libiconv-2.dll	4/21/2012 10:27 AM	Application extens	1,228 KB				
A Network	× 🗟	A/01/0010 10.07 AKA	A	חש רדר ר				
Fi	le <u>n</u> ame: gdb-arm-none-linux-gnuea	abi.exe		~	All Files (*)			\sim
					Open	Cr	incel	

• Click "Open"; the configuration portion of the screen should look similar to

options				-		X
Filter	Build & Run					
Environment	General Kits Q	t Versions Compilers	Debuggers Qbs CMake			
Text Editor	Name		Location	Туре		Add
FakeVim			C:\Qt\Tools\mingw530_32\bin\gdb.ex			Clone
P Help	Manual	at c:\MinGW\bin\gdb.exe	-	GDB		Remove
{} c++	3Dxx Targe	t Debugger	C:\Qt\QtSupport\debugger\gdb-arm	i-none-linux-gnueabi.exe GDB		
🗸 Qt Quick						
🕕 Build & Run						
🔎 Debugger						
💓 Designer	Name:	3Dxx Target Debugger				
Analyzer	Path:		gdb-arm-none-linux-gnueabi.exe		Browse	
Version Control	Type:	GDB	gub-ann-hone-nnux-gnueabl.exe		browse	
Devices	ABIs:	arm-linux-generic-elf-32bit				
Code Pasting	Version:	7.4.0				
QA Testing	Working directory:				Browse	
				ОК	Cancel	Apply

Click "Apply"

qmake

Select the "Qt Versions" tab

Click "Add" (Select a qmake Executable dialog box appears, still referencing the last path)

Navigate to the qmake version associated with the library

o /usr/local/Qt-5.12.2-3Dxx/bin/qmake

8	Select a qmake	Executable		
\odot	Recent	◀ 🕄 usr local Qt-5.9.3-3Dxx bin ▶		
	Home	Name 🔺	Size	Modified
	Desktop	🛷 qmake	35.3 MB	10:49
D	Documents			
÷	Downloads			
99	Music			
o	Pictures			
	Videos			
	sf_vmshare 📤			
+	Other Locations			
				qmake 🔻
		Can	ncel	Open

C:\QtGhSupport\qmakeInsatll\bin\qmake.exe

💷 Select a qmake Exec	utable								×
\leftrightarrow \rightarrow \checkmark \uparrow	This	PC > OS (C:) > Qt	GhSupport → qmakelnstal	l > bin		~ ⊘	Search bin		٩
Organize 🔻 New	folder								?
📌 Quick access	^	Name	^	Date modified	Туре	Size			
A Galekaccess		📧 qmake.exe		3/8/2018 2:46 PM	Application	3,845 KB			
ineDrive 🍊 🍊									
💻 This PC									
A Network	~								
1	File <u>n</u> an	ne: qmake.exe				~	qmake (qmake*)		\sim
							<u>O</u> pen	Cancel	

• Click "Open"

• Update "Version name:" to "Qt-5.12.2-3Dxx"

🞯 Options		×
Filter	Build & Run	
Environment	General Kits Qt Versions Compilers Debuggers Qbs CMake	
Text Editor	Name qmake Location	Add
FakeVim	Qt 5.9.3 MinGW 32bit C:\Qt\5.9.3\mingw53_32\bin\qmake.exe	Remove
PHelp	Qt-5.9.3-3Dxx C:\QtGhSupport\qmakeInstall\bin\qmake.exe	Clean Up
{} C++		
Qt Quick		
🕕 Build & Run		
Debugger		
📡 Designer		
Analyzer	Version name: Ot-5.9.3-3Dxx 48	
Version Control	gmake location: C:\QtGhSupport\gmakeInstall\bin\gmake.exe Browse	
Devices		
Code Pasting	Qt version 5.9.3 for Embedded Linux Details ▼	
	OK Cancel	Apply

Kit

Select the "Kits" tab

Click "Add"

Populate the fields as illustrated

- "Name:" Qt-5.12.2-3Dxx
- "Device type:" Select "Generic Linux Device" from the pick list N.B. Automatically updates Device
- "Sysroot": Click "Browse..." and navigate to desired path
 - /home/ghguest/GrayhillDisplayPlatform/sysroot-target

8	Choose Director	гу						
Ø	Recent	•	₽ghguest	GrayhillDisplayPlatform	sysroot-target	►		C 7
企	Home	Nai	me				Size	Modified
	Desktop		include					8 Dec
ם	Documents		kernel-head lib	ers				8 Dec 8 Dec
⇒	Downloads		usr					8 Dec
99	Music							
ø	Pictures							
H	Videos							
	sf_vmshare 📤							
+	Other Locations							
]							All Files 🔻
						(Cancel	Open

Click "Open" C:\QtGhSupport\GrayhillDisplayPlatform\sysroot-target

😰 Choose Directory							>	×
← → · ↑	> This	PC > OS (C:) > QtGhSuppo	ort		√ Č	Search QtGhSupport	م]
Organize 🔻 Ne	w folder						== • ?	,
	^	Name	Date modifie	d Type	Size			^
📌 Quick access		Perl64	3/13/2018 1:	13 PM File folder				
a OneDrive		Python27	3/13/2018 1:1	13 PM File folder				
		🔤 qmakeInstall	3/13/2018 1:	13 PM File folder				
💻 This PC		targetRootFiles	3/13/2018 1:	13 PM File folder				
A Network	~	📙 targetSysroot	3/13/2018 1:	13 PM File folder				¥
	Folder:	targetSysroot						
					C	Select Folder	Cancel	

ck "Select	Folder	19
npiler:	C:"	Select "ARM-GCC" from the pick list
npiler:	C++:"	Select "ARM-G++" from the pick list
ougger:"		Select "3Dxx Target Debugger" from the pick list
version:"		Select " <mark>Qt-5.12.2-3Dxx</mark> " from the pick list
	npiler:	bugger:"

N.B. The selected names must match those used when creating the various kit sub-components

Summary

🧐 Options			×
Filter	Build & Run General Kits Qt Version	s Compilers Debuggers Qbs CMake	
Text Editor FakeVim	Name Auto-detected Desktop Qt 5.9.3. Manual Qt-5.9.3-3Dxx 	MinGW 32bit (default)	Add Clone Remove Make Default
C++	Name:	Qt-5.9.3-3Dxx	Q
Build & Run	File system name: Device type:	Generic Linux Device	
Debugger	Device:	3Dxx Display (default for Generic Linux)	Manage
💥 Designer	Sysroot:	C:\QtGhSupport\targetSysroot C: ARM-GCC	Browse
Malyzer	Compiler:	C: ARM-GCC C++: ARM-G++	Manage
Devices	Environment:	No changes to apply.	Change
Code Pasting	Debugger: Ot version:	GNU gdb 7.10.1 for MinGW 5.3.0 32bit	Manage
QA Testing	Qt mkspec:		Manage
	CMake Tool:	×	Manage
	CMake generator:	<pre><none> - <none>, Platform: <none>, Toolset: <none></none></none></none></none></pre>	Change
	CMake Configuration Additional Qbs Profile Settings	CMAKE_CXX_COMPILER:STRING=%{Compiler:Executable:Cxx}; CMAKE_C_COMPILER:STRING=	Change
		OK Cancel	Apply

- Verify contents are correct
- Click "OK"

Now that a Qt kit is configured; it is possible to develop, build, test, debug, run and enjoy Qt applications.

Appendix B: Configuring a 3Dxx Project

The Grayhill support package contains 2 sample demo projects:

- GrayhillExamples/ghQmlDemo (linux kernel 3.0.35)
- GrayhillExamples/ghQmlDemo4115 (linux kernel 4.1.15)

Demo project ghQmlDemo is intended for the -100 Series of 3D50, 3D70, and 3D2104 displays running linux kernel 3.0.35.

Demo project ghQmlDemo4115 is intended for the -200 Series of 3D50, 3D70, 3D2104, and 3D101 displays running linux kernel 4.1.15.

This section details how to setup and configure the **ghQmIDemo4115** example project for the target (3Dxx).

N.B. This appendix is included for reference and is not a required installation step; Grayhill automatically configures the project as part of the support file installation.

If not already running, launch Qt Creator. (See

Build and Run 3Dxx Embedded Application)

Open a project from "Qt Creator" main window click on "Open Project" button.

N.B. If present, a previous project can be opened by clicking on the project name listed below "Recent Projects".

👳 gh7inder	mo - Qt Creator			
<u>F</u> ile <u>E</u> dit	<u>B</u> uild <u>D</u> ebug <u>A</u> nalyze <u>T</u> ools <u>W</u> ine	dow <u>H</u> elp		
Welcome	Projects	+ New Project	🗁 Open Project	
Edit	Examples	Sessions	Recent Projects	
Design	Tutorials	1 D default (current session)		
û Debug				
لكر	New to Qt?			
Projects	Learn how to develop your own applications and			
? Help	explore Qt Creator.			
	Get Started Now			
gh7indemo				
Debug	L Qt Account			
	Online Community			
	Blogs			
	Oser Guide			
~	P, Type to locate (Ctrl+K)	1 Iss 2 2 Search 3 Applic	4 Compil 5 Debug 6 Gener 8 Test	:R 🗢 🔺 🔳 🎢

- An "Open File" dialog window will appear
- Navigate to the 3Dxx Demo project's ".pro" file for either Linux or Windows as illustrated below
 - /home/GrayhillExamples/ghQmlDemo4115

Open File				8
⊘ Recent	GrayhillExamples ghQmlDemo4115			
🟠 Home	Name		Size	Modified
🖿 Desktop	📄 qml			13:41
Documents	📄 ghQmlDemo4115.pro		3.4 kB	8 Dec
🕹 Downloads				
JJ Music				
Pictures				
🛏 Videos				
🔳 sf_vmshare 📤				
	All Pro	jects		•
		C	ancel	Open

C:\QtGhSupport\GrayhillExamples\ghQmlDemo4115

🕫 Open File				×
← → × ↑ 💄 « QtG	hSupport > GrayhillExamples > ghQmlDemo	04115 > ∽	Ů	hQmIDemo4115
Organize 🝷 New folder				·= • • •
🗢 This PC	Name	Date modified	Туре	Size
🗊 3D Objects	📕 qml	12/17/2020 3:52 PM	File folder	
E Desktop	nghQmlDemo4115.pro	12/17/2020 3:50 PM	Qt Project file	4 KB
Documents				
Downloads				
Music				
Pictures				
Videos				
🐛 OS (C:)				
File <u>n</u> ame	ghQmlDemo4115.pro		 ✓ All Projects (C Open 	CMakeLists.txt;*.cre ~

• Click "Open"

If the "*project*.pro.user" file is missing, which is normal if the project has never been opened before, a "Configure Project" dialog appears. If this dialog doesn't appear, proceed to where the "Projects" icon is selected.

If the "Configure Project" dialog appears (remember screen shot illustrations are for reference purposes and may not reflect current observations)

ghQmlD	emo4115 - Qt Creator								•••
<u>F</u> ile <u>E</u> dit	<u>B</u> uild <u>D</u> ebug <u>A</u> nalyze <u>T</u> ools <u>W</u> ind	ow <u>H</u> elp		_	_	_	_		
Welcome	Manage Kits		Configure Project						-
Edit	Active Project	A results	The following kits can be used The project ghQmlDemo4115 Qt Creator uses the kit Qt-5.9	is not yet configure	ed.				
1	ghQmlDemo4115 -		Type to filter kits by name						
Design	Import Existing Build		Select all kits						
Debug	Build & Run		☑ 🖵 Desktop Qt 5.12.2 GC	64bit				Details 👻	
Projects	Desktop Qt 5.12.2 GCC 64bit		🗌 🖵 Qt-5.12.2-3Dxx					Details 👻	
? Help	 Qt-5.12.2-3Dxx Qt-5.9.3-3Dxx 		Import Build From					Details 👻	
ghQm4115	Project Settings						Configu	re Project	1
Unconfigur ed	Editor Code Style Dependencies						connge		I
► Aî	Clang Code Model Clang Tools								
~	■ P. Type to locate (Ctrl 1 Issu	es 2 Se	arch Results 3 Application	4 Compile Out	5 QML Debugg	6 General Mes	8 Test Resul	ts ¢	▲ □ //

"Desktop Qt 5.12.2 GCC 64bit"

For Linux:

- Expand by clicking on "Details"
 Unselect "Release"
 - Unselect "Profile"

For Windows:

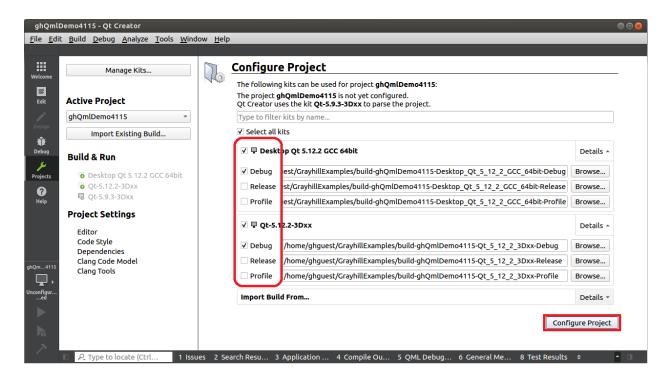
- Expand by clicking on "Details"
 - Unselect "Debug"
 - o Unselect "Profile"

N.B. It is recommended that for Windows Desktop builds, 'Debug' configuration not be used. An issue has been observed when running the Debug configuration in which a "Cannot load library" error for qtquickcontrolsplugind.dll has occurred. The demo project has been configured with the Debug build removed.

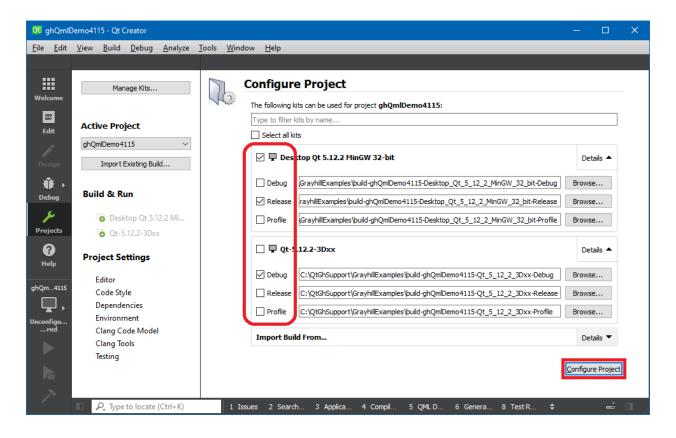
"<mark>Qt-5.12.2-3Dxx</mark>"

- Expand by clicking on "Details"
- Select "Qt-5.12.2-3Dxx" (this selection will select the three boxes below)
 Unselect "Release"
 - Unselect "Profile"

Linux Reference

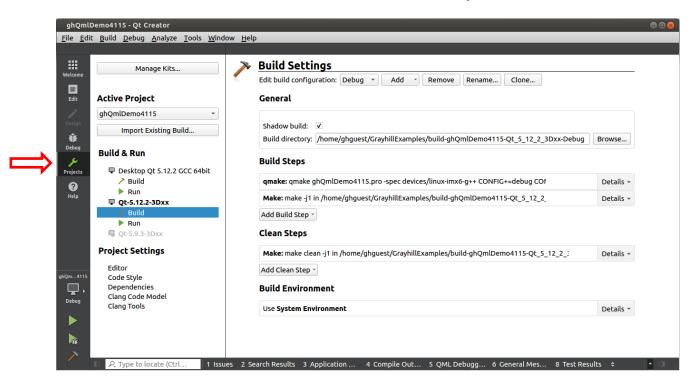


Windows Reference



• Click "Configure Project"

• On the main "Qt Creator" window select "Projects"



If the desired kit is not shown see

Appendix A: Configuring a Manual Qt Kit for Grayhill Displays

N.B. Clicking "Manage Kits" is the same as selecting "Tools → Options"

"Active Project" is a drop down pick list with the active project shown.

"Build & Run" lists the available kits.

N.B. The selected kit is emphasized in **bold**. A kit (set of utilities) is how the project will be built, e.g. the main kit difference between desktop and target is the compiler as the Qt-5.12.2-3Dxx kit uses a cross compiler for the display.

N.B. Clicking on an actual kit name selects either Build or Run (depending on which one was previously selected)

Build

This section describes how to configure the ghQmlDemo4115 example project for the target (3Dxx).

- Select "Build"
- Expand the Details tab associated with qmake (under Build Steps)
- "Additional arguments"
 - Enter "hw_present=yes target=3D70 kernel=4"

N.B. This is a **case sensitive** field.

N.B. Parameters are automatically added to the "effective qmake call" command syntax. This field is configured based on the actual target hardware display size. The processing of these arguments is in the .pro file for the project.

ghQmlE	emo4115 - Qt Creator			•
le <u>E</u> dit	<u>B</u> uild <u>D</u> ebug <u>A</u> nalyze <u>T</u> ools <u>W</u> ind	dow <u>H</u> elp		
	Manage Kits	>	Build Settings	
elcome			Edit build configuration: Debug 🔹 Add 🔹 Remove Rename Clone	
Edit	Active Project		General	
	-		Uchelat	
esign	ghQmlDemo4115 •		Shadow build:	
ŵ	Import Existing Build		Build directory: //home/ghguest/GravhillExamples/build-ghOmlDemo4115-Ot 5 12 2 3Dxx-Debt &	Browse
ebug	Build & Run			
1			Build Steps	
ojects	Desktop Qt 5.12.2 GCC 64bit Puild		Custom Process Step	Details 🔺
? Help	 Run 		Commonda — Mile Assurt	-
пецр	🖵 Ot-5.12.2-3Dxx		Command: /bin/touch	Browse
	 Build Run 		Arguments: /home/ghguest/GrayhillExamples/ghQmlDemo4115/buildinfo.c	
	© Qt-5.9.3-3Dxx		Working directory: %{buildDir} &ga	Browse
	Project Settings		Custom Process Step	Details 🔺
	Editor		Command: /home/ghguest/GrayhillExamples/ghQmlDemo4115/cfgTarget	Browse
	Code Style		Arguments: single	₿.
	Dependencies Clang Code Model		Working directory: %{buildDir}	Browse
	Clang Tools			
			qmakes qmake ghQmlDemo4115.pro -spec devices/linux-imx6-g++ CONFIG+=debug COI $\oslash \land \lor >$	Cetails 🔺
			qmake build configuration: Debug	
m4115			Additional arguments: hw_present=yes target=3D70 kernel=4	
_ ,			Generate separate debug info:	
Debug			Enable QML debugging and prover A Might make your application vulnerable. Only use in a safe environment	onment.
			Enable Qt Quick Compiler:	
			Effective qmake call: //usr/local/Qt-5.12.2-3Dxx/bin/qmake /home/ghguest/GrayhillExample ghQmlDemo4115/ghQmlDemo4115.pro -spec devices/linux-imx6-g++	
A			CONFIG+=debug CONFIG+=qml_debug hw_present=yes target=3D70 && /usr/bin/make gmake_all	
<u>/</u>	■ P. Type to locate (Ctrl 1 iss		arch Results 3 Application 4 Compile Out 5 QML Debugg 6 General Mes 8 Test Res	ults 🗢 🔺

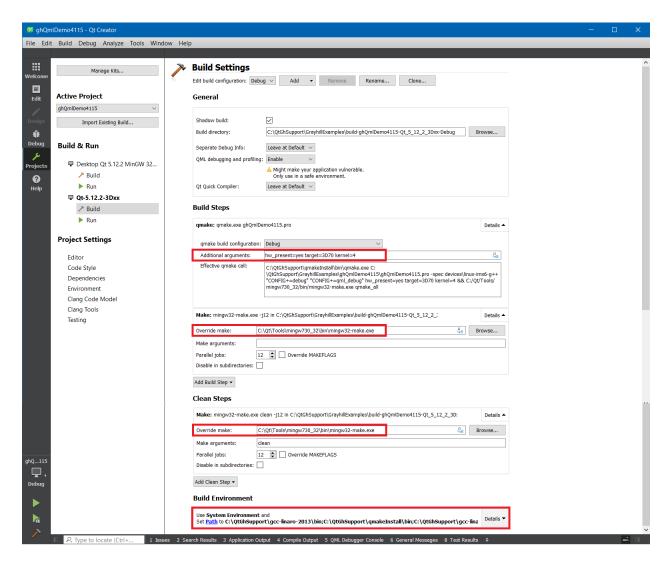
The above image also shows two custom steps, the first used for versioning and the second for camera configuration.

Kernel 4.1.15 supports multiple camera views on the 3D70, 3D2104, and 3D101.

The camera configuration for demo project ghQmlDemo4115 is given in the cfgTarget custom step by specifying one of the following for Arguments:

- single (only 1 camera can be selected for display)
- multi (2 cameras can be displayed simultaneously)

Windows Reference



Build Steps

- Additional arguments (see Linux screen capture above)
- Override make → C:\Qt\Tools\mingw730_32\bin\mingw32-make.exe

Clean Steps

Override make → C:\Qt\Tools\mingw730_32\bin\mingw32-make.exe

Build Environment

• Path Append ;C:\Qt\5.12.2\mingw730_32\bin

N.B. An issue has been observed when using Qt Creator for Windows where the compile will fail with an error similar to:

```
mingw32-make: *** [Makefile:403: qrc_qml.cpp] Error -1073741701
mingw32-make: *** Waiting for unfinished jobs....
mingw32-make: *** [Makefile:750: moc_disp3dxx.cpp] Error -1073741701
```

This issue has been observed with Qt Creator running on Windows; it has not been observed with Qt Creator running on the Linux VM.

At the time of this document writing, a solution has not been found, as the issue appears to be with tools not controlled by Grayhill.

However, Grayhill has found the following potential (albeit temporary) workarounds that will usually eliminate the error and get the compile working again:

- Reboot the development PC
 -OR-
- Uninstall and re-install Qt

Run/Deployment

This section describes how to compile and deploy the example project to the target (3Dxx).

- Select "Run"
- Deployment
 - Method: Deploy to Remote Linux Host (default)
 - Files to deploy:

Local File Path	location on host	(auto-populated)
Remote Directory	location on target	(auto-populated)

N.B. File information may not populate until after a build is done.

- Expand "Details" for "Upload files via SFTP"
- Make sure neither box is selected

N.B. On rare occasions, Qt Creator thinks the files have already been deployed and will not re-send the files to the target; disabling this functionality avoids the situation.

ghQml	emo4115 - Qt Creator		B B 8						
<u>F</u> ile <u>E</u> dit <u>B</u> uild <u>D</u> ebug <u>A</u> nalyze <u>I</u> ools <u>W</u> indow <u>H</u> elp									
	Manage Kits	Run Settings	-						
Welcome									
		Deployment							
Edit	Active Project	Method: Deploy to Rem 👻 Add 👻 Remove Rename							
Design	ghQmlDemo4115 *	Files to deploy:							
-	Import Existing Build		ote D						
û Debug		/home/ghguest/GrayhillExamples/ghQmlDemo4115/qml /opt	/ghQ						
ير ا	Build & Run	/home/ghguest/GrayhillExamples/build-ghQmlDemo4115-Qt_5_12_2_3Dxx-Debug/ghQmlDemo4115_/opt,	/ghQ						
Projects	🖵 Desktop Qt 5.12.2 GCC 64bit	4	Þ						
•	Build	Check for free disk space Detai	ils 👻						
Help	Run	· · ·							
	-	Qt-5.12.2-3Dxx Kill current application instance Build							
	🕨 Run	Upload files via SFTP Detai	ls ≜						
	🐻 Qt-5.9.3-3Dxx	Incremental deployment							
	Project Settings	Ignore missing files							
	Editor	Add Deploy Step *							
	Code Style								
	Dependencies	Run							
	Clang Code Model Clang Tools								
	5	Run configuration: ghQmlDemo41 * Add * Remove Rename Clone							
ghQm4115		Executable on device: /opt/ghQmlDemo4115/bin/ghQmlDemo4115							
		Alternate executable on device:							
Debug		Executable on host: /home/ghguest/GrayhillExamples/build-ghQmlDemo4115-Qt_5_12_2_3Dxx-t)ebi						
		Command line arguments:							
		Working directory: Browse							
		🗌 Run in terminal							
		:0 Groward to local disp	play						
	P. Type to locate (Ctrl 1 Issue)	sues 2 Search Results 3 Application 4 Compile Out 5 QML Debugg 6 General Mes 8 Test Results 💠	A 🗆 /						

• Save! File → Save All

ghQmlDemo411	15 - Qt C	reator							
<u>F</u> ile <u>E</u> dit <u>B</u> uild	<u>D</u> ebug	<u>A</u> nalyze	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp				
🛯 🔒 <u>N</u> ew File or Pro	Ctrl+N								
朣 <u>O</u> pen File or P	Ctrl+O								
Open File <u>W</u> ith									
Recent <u>F</u> iles					•				
Recent P <u>r</u> oject	s				•				
S <u>e</u> ssions					•				
Close Project '	Close Project "ghQmlDemo4115"								
Close All Files	Close All Files in Project "ghQmlDemo4115"								
Close All Proje	cts and E	ditors							
💹 Save					Ctrl+S				
🔯 Save <u>A</u> s									
Save A <u>l</u> l					Ctrl+Shift+S				
🗟 Revert to Save	d								
Close					Ctrl+W				
Close All					Ctrl+Shift+W				
Close Others									
Close All Excep	ot Visible								
🗎 <u>P</u> rint					Ctrl+P				
🖲 E <u>x</u> it					Ctrl+Q				

• Build options

ghQmlDo	emo4115 - Qt Creator					
<u>F</u> ile <u>E</u> dit	<u>Build D</u> ebug <u>A</u> nalyze <u>T</u> ools <u>W</u> indow <u>H</u> elp					
	➢ Build All	Ctrl+Shift+B				
	Build Project "ghQmlDemo4115"	Ctrl+B				
Welcome	Generate Compilation Database for "ghQmlDemo4115"					
	Run qmake					
Edit	Deploy All					
<u>_</u>	Deploy Project "ghQmlDemo4115"					
Design	🛠 Rebuild All					
Ú	Rebuild Project "ghQmlDemo4115"					
Debug	🛓 Clean All					
يو ا	Clean Project "ghQmlDemo4115"					
Projects	Cancel Build	Alt+Backspace				
?	🕨 Run	Ctrl+R				
Help	Run Without Deployment					
	Open Build and Run Kit Selector					
	QML Preview					

- Build Let Qt Creator decide what is out of date
- Rebuild Force Qt creator to re-compile everything
- **Clean** Remove existing artifacts generated by previous builds
- **Run** Deploy the executable to the target and execute the image

• Build and Run the image for the target by clicking the green triangle



The bottom ribbon of Qt Creator has various panes (views) that can be examined. "Application Output" is shown; this pane is also where qDebug messages will be output.

Click the paintbrush icon to clear the contents.

Click the red square to terminate the target session.

N.B. Errors and issues are summarized in the "Issues" tab.

Appendix C: Debugging

Let's face it; code never initially does what it is *supposed* to do; but rather what it was **told** to do! Luckily, Qt Creator has a built-in debugger!

N.B. In order to debug QML, then file(s) must be list in the QML folder. If they are not, then check to make sure qtquickcomplier is not set in the .pro file. Also verify that "Enable Qt Quick Compiler" is not checked in Build for qmake step.

Additional debugging information can be found by Googling "qt debugging" which includes the following link.

https://doc.qt.io/qtcreator/creator-debugging.html

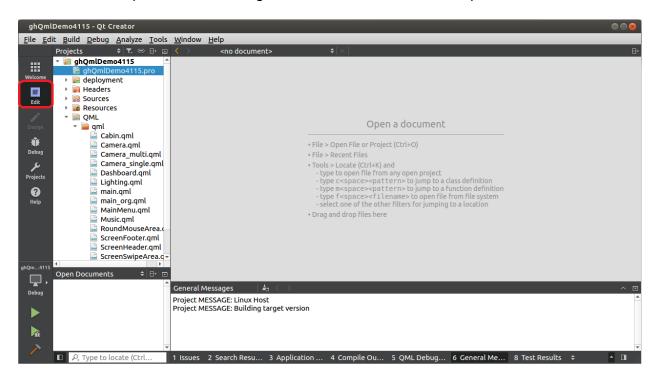
Debugger stepping option icons (Mouse over the icons for a description)

Debugger	🗧 🖒 🗄 🖬 💭 🖬 🕼 E
----------	-----------------

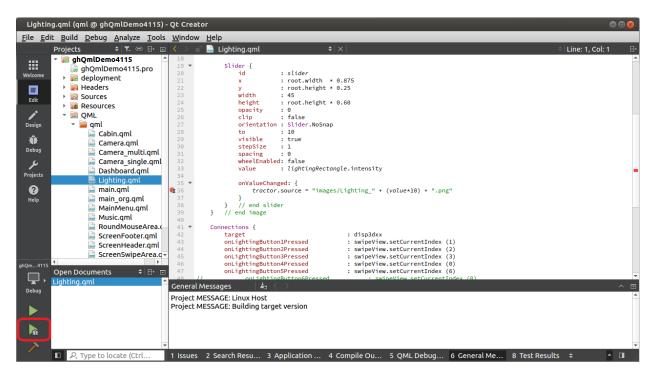
- Continue
- Stop
- Step Over <F10>
- Step In <F11>
- Step Out <Shift>+<F11>

The debugger toolbar icon's functions are also available under the Debug drop down menu.

- Load ghQmlDemo4115
- Select the "Edit" view
- Expand contents of ghQmlDemo4115 \rightarrow QML \rightarrow qml



- Select "LightingForm.qml"
- Add a breakpoint by left clicking in the gutter at line 36 (tractor.source = ...)

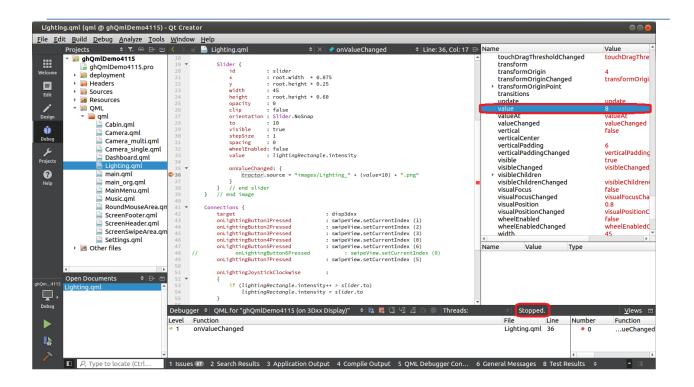


- Verify building for target and configuration parameters are set
- Click Green triangle with cute little bug!

The code begins to execute once compiled (the project may re-compile) and then hits the breakpoint. This happens during the initial loading of the form.

Press <F5> to Continue (or click on continue icon in the Debugger bar).

- Select the lighting screen
- Click on the 8th lighting level



- Execution will stop (the display will not update, as the breakpoint is at the point where the lighting level image is loaded)
- Expand "this" and scroll down to value to see the new value

Appendix D: Setting up a 3Dxx Qt Program to Run at Boot Up

This section describes how to configure a program to automatically execute at boot up.

- Open a terminal window on the 3Dxx dsisplay (Error! Reference source not found. describes how to launch "PuTTY")
- Create⁴ a launch script for the desired application

Explanation

88

cd /etc/init.dchange into proper directoryecho "#! /bin/sh –ltreat as login (runs profile)/opt/ghQmlDemo/bin/ghQmlDemo &" > launchQtAppcat launchQtAppcat launchQtAppchmod 755 launchQtAppmake script executable

COM1 serial port to target

root@ghiimx6:" cd /etc/init.d root@ghiimx6:/etc/init.d echo "#! /bin/sh -1 >cd /home/demo7in >/home/demo7in/gh7indemo &" > launchQtApp root@ghiimx6:/etc/init.d cat launchQtApp #! /bin/sh -1 cd /home/demo7in /home/demo7in/gh7indemo & root@ghiimx6:/etc/init.d chmod 755 launchQtApp root@ghiimx6:/etc/init.d

⁴ vi (text editor) can also be used for those familiar with vi, instead of the command line

• Create a link to the launch script created above

Explanation

cd /etc/rc.d set into proper directory In -s /etc/init.d/launchQtApp S12qtApp Is -I S12qtApp verify link creation

COM1 serial port to target	•••
root@ghiimx6:~ cd /etc/init.d root@ghiimx6:/etc/init.d echo "#! /bin/sh -l	
>cd /home/demo7in	
>/home/demo7in/gh7indemo &" > launchQtApp root@ghiimx6:/etc/init.d cat launchQtApp	
#! /bin/sh -l	
cd /home/demo7in	
/home/demo7in/gh7indemo & reat@ahiimx6*/etc/init_d_chmod_755_launch0tApp	
root@ghiimx6:/etc/init.d cd /etc/rc.d	
root@ghiimx6:/etc/rc.d ln -s /etc/init.d/launchQtApp S12qtApp root@ghiimx6:/etc/rc.d ls -l S12qtApp	
lrwxrwxrwx 1 rootroot 23 Jan 18 13:02 S12qtApp -> /etc/init.d/launchQtApp	
root@ghiimx6:/etc/rc.d	

N.B. Do not try to launch multiple Qt applications at boot up or try to launch the ghvehicleapp application along with a Qt application as they conflict with one another.

N.B. When switching from running one application to another, even between Qt applications, it is a good idea to do a reboot of the 3Dxx Display in between to make sure that the hardware is properly reset. This can be done by entering the "reboot" command on the 3Dxx Display Linux console.

Appendix E: Interfacing 3Dxx Hardware from QT Software

This section explains how to access the functionality of these components. The programming interfaces and provided API functions are covered, with the syntax and parameters defined. Sample code is also provided where appropriate.

The 3Dxx Display contains the following custom component interfaces:

- Analog Input driver (Model 3D70 only)
- Audio Output (Model 3D70 only)
- Buzzer (Models 3D70, 3D2104, 3D101)
- Camera driver
- CAN driver
- Digital I/O driver
- LCD
- LCD Backlight

Analog Inputs (Model 3D70 only)

The Model 3D70 Display has two analog inputs. Analog Input 1 is connected to Pin 4 on Connector B and Analog Input 2 is connected to Pin 5 on Connector B. The Analog Inputs can be used to read resistance, voltage, or current with respect to the analog return pin (pin 7 on Connector B).

The following Grayhill files are required:

- ghiolib.h (header)
- libghio.so (linker)

Interface

A Qt application may configure or read an analog input pin by calling the appropriate C library function as described below.

```
#define GHIOLIB CH1
                        (0x01)
#define GHIOLIB CH2
                      (0x02)
#define GHIOLIB MAX ANALOG IN
                               (2)
#define GHIOLIB ANALOG 5V
                               (0)
#define GHIOLIB ANALOG 15000HM (1)
#define GHIOLIB ANALOG 10V
                               (2)
#define GHIOLIB ANALOG 50000HM (3)
#define GHIOLIB ANALOG 20MA
                               (4)
#define GHIOLIB RET OK
                                 0
#define GHIOLIB RET ERROR
                                 1
#define GHIOLIB RET NOTSUPPORTED 2
typedef struct ADCVALUES
{
     uint16 t adcch;
     uint16 t adcvref;
     uint16 t adcstatus;
     uint16 t adcconfig;
} ADCVALUES, *PADCVALUES;
```

ghiolib_setADCcfg

This function configures an analog input for one of five different reading modes.

Syntax
int ghiolib_setADCcfg(int ch, uint8_t config);

Туре	Name	in/out	Description	Values
int	ch	in	Channel to	GHIOLIB_CH1
			configure	GHIOLIB_CH2
uint8_t	config	in	Channel	GHIOLIB_ANALOG_5V
			configuration	GHIOLIB_ANALOG_10V
			-	GHIOLIB_ANALOG_1500OHM
				GHIOLIB_ANALOG_5000OHM
				GHIOLIB_ANALOG_20MA
int			Return value	GHIOLIB_RET_OK
				GHIOLIB_RET_ERROR
				GHIOLIB_RET_NOTSUPPORTED

ghiolib_getADCin

This function reads a value from an analog input pin.

Syntax

int ghiolib_getADCin(int ch, PADCVALUES p);

Туре	Name	in/out	Description	Values
int	ch	in	Channel to read	GHIOLIB_CH1
				GHIOLIB_CH2
PADCVALUES	Р	out	Read value is retur	ned in member "adcch" of this structure.
			Other items in this	structure can be ignored.
int			Return value	GHIOLIB_RET_OK
				GHIOLIB_RET_ERROR
				GHIOLIB_RET_NOTSUPPORTED

```
#include <QDebug>
// For access to ghiolib
typedef u int16 t uint16 t;
typedef u int8 t uint8 t;
#ifdef cplusplus
extern "C" {
#endif
#include "ghiolib.h"
#ifdef cplusplus
}
#endif
int
        channel = 0;
ADCVALUES analogData;
int gpioStatus;
// Set analog input 1 to read 0 to 10 volts
gpioStatus = ghiolib setADCcfg(channel + 1, GHIOLIB ANALOG 10V);
if (GHIOLIB RET OK != gpioStatus)
{
    qDebug("ERROR (%d) doing ghiolib setADCcfg on channel: %d\n",
           gpioStatus, channel + 1);
}
// Get current reading
gpioStatus = ghiolib getADCin(channel + 1, &analogData);
if (GHIOLIB RET_OK != gpioStatus)
{
    qDebug("ERROR (%d) doing ghiolib getDigOut on channel: %dn",
           gpioStatus, channel + 1);
}
qDebug("Reading from channel %d is %d millivolts\n", channel + 1,
analogData.adcch);
```

Audio Output (Model 3D70 only)

The Model 3D70 Display has the ability to play an mp3 audio file and send the audio output to a monaural line out (pins 1, AUDIO OUT, and 2, AUDIO RET, on the B connector).

There are no required header or linker files, but the mpg123 application must be installed on the display and its location be in the search PATH.

Interface

A Qt application can start playing an mp3 audio file and can stop the playing of the audio file using a Linux utility called mpg123.

```
//
// To play mp3 file "sounds.mp3"
//
// Note that by placing mp3 file in "images" folder, Qt will
// automatically download the mp3 file to the target with the
// other image files being used.
//
// Command shown to play mp3 file will first stop playing any mp3 //
file that may already be playing.
//
system("test `pidof mpg123` && kill `pidof mpg123` ;"
    "mpg123 -q images/sounds.mp3 &");
// To stop playing mp3 file (if any)
system("test `pidof mpg123` && kill `pidof mpg123`");
```

Buzzer (Models 3D70, 3D2104, 3D101)

The Model 3D70, 3D2104, and 3D101 Displays have an internal buzzer that can be sounded on command.

There are no additional required files.

Interface

A Qt application can turn the internal buzzer on or off by sending the proper number to the buzzer control file.

```
#include <QString>
#include <QDebug>
QFile
          buzzerFile;
bool
           buzzerFileOpen;
buzzerFile.setFileName("/sys/class/backlight/pwm-
backlight.3/brightness");
buzzerFileOpen = buzzerFile.open(QIODevice::WriteOnly |
QIODevice::Text);
if (false == buzzerFileOpen)
{
    qDebug("Error opening buzzer file\n");
}
// To turn buzzer ON
if (true == buzzerFileOpen)
{
    QTextStream buzzerOut(&buzzerFile);
    buzzerOut << 10;</pre>
}
// . . .
// To turn buzzer OFF
if (true == buzzerFileOpen)
{
    QTextStream buzzerOut(&buzzerFile);
    buzzerOut << 0;</pre>
}
```

Camera Driver Interface (Single camera)

The following describes the camera driver interface that is used with the 3.0.35 kernel. This interface allows only one camera input to be active at a time.

This interface can be used with the 4.1.15 kernel; however, for new designs it is recommended that the multi-camera interface described later is used with display models running the 4.1.15 kernel.

The Grayhill 3Dxx Display device can contain multiple camera inputs. NTSC and PAL format video inputs are supported by modifying the camera input sensor parameters. The camera output can be displayed on the LCD. The following camera display parameters can be modified:

- Window parameters window size and window position
- Color parameters brightness, contrast, saturation and hue
- Rotation
- Input sensor parameters provides support for NTSC and PAL formats
- Camera output to LCD foreground or background with color key

Camera output is 30 frames per second (fps).

The following Grayhill files are required:

• libghdrv.so (linker)

N.B. Only one camera input can be active at a time with kernel 3.0.35.

Kernel 4.1.15 will support multiple camera views on the 3D2104, 3D70, and 3D101.

Data Types

```
typedef struct SENSORPARAMS // Must be set according to camera input
type
                           // NTSC
                                      PAL
{
   unsigned int top;
                           // 4
                                      5
                           // 0
   unsigned int left;
                                      4
                           // 480
                                      567
   unsigned int height;
   unsigned int width;
                         // 640
                                      640
} SENSORPARAMS, *PSENSORPARAMS;
#define FOREGROUND
                   (1)
#define BACKGROUND
                   (0)
#define FB DEV 0
                   (0) // GRAPHICS being sent to /dev/fb0
#define FB DEV 1
                   (1) // GRAPHICS being sent to /dev/fb1
11
// These are the only allowed values for VIDEO COLOR KEY xxx
11
#define VIDEO COLOR KEY BLACK (0x0000000)
#define VIDEO COLOR KEY RED (0x00FF0000)
#define VIDEO COLOR KEY GREEN (0x0000FF00)
#define VIDEO COLOR KEY BLUE (0x00000FF)
#define VIDEO COLOR KEY YELLOW (0x00FFFF00)
#define VIDEO_COLOR_KEY_CYAN (0x0000FFFF)
#define VIDEO COLOR KEY MAGENTA (0x00FF00FF)
#define VIDEO COLOR KEY WHITE (0x00FFFFFF)
typedef struct DISPLAYPARAMS
                       // top left window y-coordinate
   unsigned int top;
   unsigned int left;
                        // top left window x-coordinate
                        // (must be divisible by 4)
   unsigned int height; // window vertical size
   unsigned int width; // window horizontal size
                        // NOTE: top + height must not exceed
                        11
                               height of display
                                and left + width must not exceed
                        11
                                display width
                        11
   unsigned int rotate; // 0-7, see below
                        // FOREGROUND or BACKGROUND +
   unsigned int fg;
                        // VIDEO COLOR KEY XXX
} DISPLAYPARAMS, *PDISPLAYPARAMS;
```

The camera output always operates in native landscape mode. Use the following rotation values to support other display and camera orientations:

Value	Rotation			
0	No rotation			
1	Vertical flip			
2	Horizontal flip			
3	180			
4	90 right			
5	90 right with vertical flip			
6	90 right with horizontal flip			
7	90 left			

Interface

The Qt application interfaces with the Camera driver using the Camera class.

Camera::Camera

Camera class constructor

Syntax

```
Camera:: Camera (int camnum, int fbdev = FB DEV 0);
```

Туре	Name	in/out	Description	Values
int	camnum	in	Camera Number	3D50 : 1-2
				3D70 : 1-3
				3D2104: 1-4
				3D101: 1-4
int	fbdev	in	Frame buffer	See below
			device	

The "fbdev" value indicates whether the GRAPHICS are being sent to fb0 or fb1.

When GRAPHICS are being sent to fb0, then video will be sent to fb1 and only foreground mode is allowed. This is the default assumed if "fbdev" is missing.

If GRAPHICS are being sent to fb1, then video will be sent to fb0 and both foreground and background modes are supported. In order to send GRAPHICS to fb1, add this parameter to the command line that launches Qt: -display LinuxFb:/dev/fb1

Camera::setdisplayparams

Sets the following display window parameters

- origin
- window size
- rotation
- foreground or background with color key (When using background mode the camera video only shows through where the graphics data is set to the color that matches the specified color key. Graphics of any other color will appear on top of the camera video image.)

Syntax

int Camera::setdisplayparams(PDISPLAYPARAMS p);

Туре	Name	in/out	Description	Values
PDISPLAYPARAMS	р	in	refer to DISPLAYP	ARAMS structure
int			Return value	0 – success -1 - failure

Camera::setcolorparams

Sets the following camera color parameters

- Brightness
- Saturation
- Contrast
- Hue

Syntax

int Camera::setcolorparams(PCOLORPARAMS p);

Туре	Name	in/out	Description	Values
PCOLORPARAMS	р	in	refer to COLORPA	RAMS structure
int			Return value	0 – success -1 - failure

Camera::setsensorparams

Sets the camera sensor parameters

Syntax

int Camera::setsensorparams(PSENSORPARAMS psensor);

Туре	Name	in/out	Description	Values
PSENSORPARAMS	psensor	in	refer to SENSORP.	ARAMS structure
int			Return value	0

Camera::show

Enables or disables the camera

Syntax

int Camera::show(int enable);

Туре	Name	in/out	Description	Values
int	enable	in	Turn on/off camera	0 – disable 1 - enable
int			Return value	0 – success -1 - failure

Sample Code

```
#include ``camera.h"
COLORPARAMS color;
DISPLAYPARAMS disp;
int cameranum = 1;
                       // camera input 1
Camera cam(cameranum);
disp.top
           = 0;
disp.left = 80;
disp.height = 480;
disp.width = 640;
disp.rotate = 4; // rotate 90 degree right
disp.fg
        = FOREGROUND;
// configure display parameters
cam.setdisplayparams(&disp);
// start camera
cam.show(1);
// change color parameters
color.brightness = 50;
color.saturation = 128;
color.contrast = 128;
color.hue = 0;
// configure color parameters
cam.setcolorparams(&color);
. . . .
// stop l+camera
```

cam.show(0);

Camera Driver Interface (Multi-camera)

The following describes the camera driver interface that is used with the 4.1.15 kernel. This interface allows multiple cameras to be displayed simultaneously.

- Up to 2 simultaneous cameras may be displayed for models 3D50 and 3D70. Camera 1 and camera 3 cannot be displayed simultaneously.
- Up to 3 simultaneous cameras may be displayed for models 3D2104 and 3D101. Camera 1 and camera 4 cannot be displayed simultaneously.

Data Types

```
#define FOREGROUND (1)
#define BACKGROUND (0)
//
// These are the only allowed values for VIDEO_COLOR_KEY_xxx
//
#define VIDEO_COLOR_KEY_BLACK (0x0000000)
#define VIDEO_COLOR_KEY_RED (0x00FF0000)
#define VIDEO_COLOR_KEY_GREEN (0x0000FF00)
#define VIDEO_COLOR_KEY_BLUE (0x00000FF)
#define VIDEO_COLOR_KEY_YELLOW (0x00FFFF00)
#define VIDEO_COLOR_KEY_YELLOW (0x00FFFF0)
#define VIDEO_COLOR_KEY_MAGENTA (0x00FF0FF)
#define VIDEO_COLOR_KEY_MAGENTA (0x00FFFFF)
#define VIDEO_COLOR_KEY_WHITE (0x00FFFFFF)
```

The camera output always operates in native landscape mode. Use the following rotation values to support other display and camera orientations:

Value	Rotation
0	No rotation
1	Vertical flip
2	Horizontal flip
3	180
4	90 right
5	90 right with vertical flip
6	90 right with horizontal flip
7	90 left

```
typedef struct _DISPLAYPARAMS
{
    unsigned int top;
```

```
unsigned int left;
        unsigned int height;
        unsigned int width;
         unsigned int rotate;
                                // 0-7
         unsigned int fg; // FOREGROUND or BACKGROUND + VIDEO COLOR KEY xxx
   } DISPLAYPARAMS, *PDISPLAYPARAMS;
  typedef struct COLORPARAMS
                                  // 0-255
        unsigned int brightness;
                                     // 0-255
        unsigned int saturation;
                                      // 0x7F, 0 , 0x80
        unsigned int hue;
                                     // 0-255
         unsigned int contrast;
   } COLORPARAMS, *PCOLORPARAMS;
   typedef struct SENSORPARAMS
   {
        unsigned int top;
        unsigned int left;
        unsigned int height;
         unsigned int width;
   } SENSORPARAMS, *PSENSORPARAMS;
  enum camera system mode {
        MODEO GR FBO CAM FBO,
        MODE1 GR FB1 CAM FB0 LALPHA,
        MODE2 GR FB1 CAM FB0 COLORKEY,
        MODE3 GR FB0 CAM FB1 LALPHA,
        MODE4 GR FB0 CAM FB1 COLORKEY
   };
   typedef struct CAMERAMODE
   {
         enum camera system mode mode;
        unsigned int alpha;
        unsigned int local alpha;
         unsigned int color key;
         unsigned int options;
   } CAMERAMODE, *PCAMERAMODE;
API calls
```

```
int Camera_system_init(CAMERAMODE *, int verbose);
int Camera_system_uninit(void);
int Camera_show(int cam_num, PDISPLAYPARAMS pdisplay, PSENSORPARAMS
psensor);
int Camera_hide(int cam_num);
int Camera_freeze(int cam_num);
int Camera_setcolorparams(int cam_num, PCOLORPARAMS p);
```

```
int Camera_getcolorparams(int cam_num, PCOLORPARAMS p);
int Camera_save_image_to_buffer(int cam_num, void *buffer, unsigned
buf_size);
int Camera get image storage size(int cam num);
```

Camera_system_init

This function configures the mode used by the camera system driver for displaying graphics and camera video on the frame buffers.

Syntax

```
int Camera system init (CAMERAMODE *pmode, int verbose);
```

Parameters

Туре	Param	Description	Values
CAMERAMODE *	pmode	refer to CAMERAMODE structure	
int	verbose	Print some info messages	

The mode is set according to the following options:

mode setting (camera_system_mode)	Frame Buffer 0 (fb0)	Frame Buffer 1 (fb1)	Mixing
MODE0_GR_FB0_CAM_FB0	Graphics, Cameras		
MODE1_GR_FB1_CAM_FB0_LALPHA	Cameras	Graphics	Alpha Blending
MODE2_GR_FB1_CAM_FB0_COLORKEY	Cameras	Graphics	Color Key
MODE3_GR_FB0_CAM_FB1_LALPHA	Graphics	Cameras	Alpha Blending
MODE4_GR_FB0_CAM_FB1_COLORKEY	Graphics	Cameras	Color Key

For modes 1 and 3, the 3Dxx display provides for alpha blending between the 2 frame buffers (fb0, and fb1). Frame buffer fb1 is 'on top of' fb0. This allows, for example, text in fb1 to overlay graphics in fb0.

The alpha setting is a value between 0x00 and 0xFF, and is interpreted as follows:

Alpha	fb0	fb1
0x00	100%	0%
0x80	50%	50%
0xFF	0%	100%

By default, the alpha setting is 0x80, which is 50% blending between fb0 and fb1

Return Value

int 0 = success, -1 = failure

Required Files

Header File: camera_eng.h Link Library: libghdrv.so

Camera_system_uninit

This function unconfigures the camera system driver mode.

Syntax

int Camera system uninit (void);

Parameters

None

Return Value int 0 =success, -1 =failure

Required Files

Header File: camera_eng.h Link Library: libghdrv.so

Camera_show

This function makes the display window visible for the specified camera.

Syntax

Parameters

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1
			3D70 : 0-2
			3D2104 : 0-3
			3D101 : 0-3
PDISPLAYPARAMS	pdisplay	refer to DISPLAYPARAMS structure	
PSENSORPARAMS	psensor	refer to SENSORPARAMS structure	

Return Value

int 0 = success, -1 = failure

Required Files

Header File: camera_eng.h

Link Library: libghdrv.so

Camera_hide

This function makes the display window invisible for the specified camera.

Syntax

int Camera hide (int cam num);

Parameters

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1 3D70 : 0-2 3D2104 : 0-3 3D101 : 0-3

Return Value

int 0 = success, -1 = failure

Required Files Header File: camera_eng.h Link Library: libghdrv.so

Camera_freeze

This function freezes the video window for the specified camera.

Syntax

int Camera_freeze (int cam_num);

Parameters

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1 3D70 : 0-2 3D2104 : 0-3 3D101 : 0-3

Return Value

int 0 =success, -1 =failure

Required Files

Header File: camera_eng.h Link Library: libghdrv.so

Camera_setcolorparams

This function sets the following color parameters of the display window for the specified camera:

- Brightness
- Saturation
- Contrast
- Hue

Syntax

```
int Camera setcolorparams (int cam num, PCOLORPARAMS pcolor);
```

Parameters

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1
			3D70 : 0-2
			3D2104 : 0-3
			3D101 : 0-3
PCOLORPARAMS	pcolor	refer to COLORPARAMS structure	

Return Value

int 0 = success, -1 = failure

Required Files

Header File: camera_eng.h Link Library: libghdrv.so

Camera_getcolorparams

This function gets the following color parameters of the display window for the specified camera:

- Brightness
- Saturation
- Contrast
- Hue

Syntax

int Camera getcolorparams (int cam num, PCOLORPARAMS pcolor);

Parameters

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1
			3D70 : 0-2
			3D2104 : 0-3
			3D101 : 0-3
PCOLORPARAMS	pcolor	refer to COLORPARAMS structure	

Return Value

int 0 = success, -1 = failure

Required Files

Header File: camera_eng.h Link Library: libghdrv.so

Camera_save_image_to_buffer

This function saves a snapshot of the video display window for the specified camera.

Syntax

Parameter

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1
			3D70 : 0-2
			3D2104 : 0-3
			3D101 : 0-3
void *	buffer	Pointer to buffer to receive image	
		data	
unsigned	buf_size	size of receive buffer	

Return Value

int 0 = success, -1 = failure

Required Files

Header File: camera_eng.h Link Library: libghdrv.so

Camera_get_image_storage_size

This function returns the size of the image data for the specified camera.

Syntax

int Camera get image storage size (int cam num);

Parameters

Туре	Param	Description	Values
int	cam_num	Camera number	3D50 : 0-1 3D70 : 0-2 3D2104 : 0-3 3D101 : 0-3

Return Value

int size of image

Required Files Header File: camera_eng.h Link Library: libghdrv.so

Sample Code (1 camera displayed)

```
#include <camera.h>
#include ``3Dxx-4115/camera eng.h"
// camera index
#define CAM1 (0)
DISPLAYPARAMS
              camDisplayParms;
             camSensorParams;
SENSORPARAMS
CAMERAMODE *cm = NULL;
// configure camera initialization mode
11
cm
              = new CAMERAMODE;
cm->mode = MODE1_GR_FB1_CAM_FB0_LALPHA;
cm->options = 0;
                        // 50% blending
cm->local alpha = 0x80;
cm->alpha = 0xff;
cm->color key = 0;
// initialize camera system
11
Camera system init (cm, false);
// set camera display parameters
//
camDisplayParms.fg = FOREGROUND;
camDisplayParms.rotate = 0; // No rotation
camDisplayParms.top = 0;
camDisplayParms.left = 0;
                             // Must be divisible by 4
camDisplayParms.height = 480;
camDisplayParms.width = 640;
// set camera sensor parameters for NTSC
11
camSensorParams.top = 4;
camSensorParams.left = 0;
camSensorParams.width = 640;
camSensorParams.height = 480;
// turn camera on
11
Camera show (CAM1, &camDisplayParms, &camSensorParams);
. . . .
// turn camera off
11
Camera hide (CAM1);
```

```
Sample Code (2 cameras displayed)
```

#include <camera.h>

```
#include ``3Dxx-4115/camera eng.h"
#define MAX CAMERAS (3)
// camera indices
#define CAM1 (0)
#define CAM2 (1)
#define CAM3 (2)
DISPLAYPARAMS camDisplayParms[MAX CAMERAS];
SENSORPARAMS camSensorParams;
CAMERAMODE *cm = NULL;
// configure camera initialization mode
11
              = new CAMERAMODE;
CM
cm->mode
             = MODE1 GR FB1 CAM FB0 LALPHA;
cm->options = 0;
                            // 50% blending
cm->local alpha = 0x80;
          = 0xff;
cm->alpha
cm->color key = 0;
// initialize camera system
11
Camera system init (cm, false);
11
//\ {\rm camera}\ 1 on left half of screen
11
// set camera #1 display parameters
11
camDisplayParms[CAM1].fg = FOREGROUND;
camDisplayParms[CAM1].top = 0;
                         = 0;
                                   // Must be divisible by 4
camDisplayParms[CAM1].left
camDisplayParms[CAM1].height = 480;
camDisplayParms[CAM1].width = 320;
// set camera sensor parameters for NTSC
11
camSensorParams.top
                   =
                         4;
camSensorParams.left =
                         0;
camSensorParams.width = 640;
camSensorParams.height = 480;
11
// camera 2 on right half of screen
11
// set camera #2 display parameters
11
camDisplayParms[CAM2].fg
                          = FOREGROUND;
camDisplayParms[CAM2].rotate = 0;  // No rotation
```

```
camDisplayParms[CAM2].top
                           = 0;
camDisplayParms[CAM2].left = 320;
                                       // Must be divisible by 4
camDisplayParms[CAM2].height = 480;
camDisplayParms[CAM2].width = 320;
11
//\ same sensor parameters as camera 1
11
// turn camera 1 on
//
Camera_show (CAM1, &camDisplayParms[CAM1], &camSensorParams);
// turn camera 2 on
11
Camera show (CAM2, &camDisplayParms[CAM2], &camSensorParams);
. . . .
// turn cameras off
11
Camera hide (CAM1); // cam 1
Camera hide (CAM2); // cam 2
```

CAN Driver Interface

The 3D50 and 3D70 displays includes two CAN controller modules. Available CAN ports are CAN1 and CAN2. The 3D2104 and 3D101 displays include three CAN controller modules. Available CAN ports are CAN1, CAN2, and CAN3. The CAN controller supports both standard and extended frames.

The following Grayhill files are required:

• libghdrv.so (linker)

Data Types

```
/*
 * special flag bits for the CAN ID
 */
#define CAN EFF FLAG 0x80000000 /* EFF flag (add to ID to
                                                               */
                                  activate 29-bit ID)
#define CAN RTR FLAG 0x40000000 /* remote transmission request */
#define CAN ERR FLAG 0x20000000 /* error frame
                                                               */
struct CANMSG
{
unsigned int ID;
unsigned int Length; // Data Length Code of the Msg (0..8)
unsigned char Data[8];
};
typedef struct CANMSG CANMSG, *PCANMSG;
```

Interface

The Qt application interfaces with the CAN bus driver using the CAN class.

CAN::CAN

CAN class constructor

Syntax

CAN::CAN(int num);

Туре	Name	in/out	Description	Values
int	num	in	CAN port number	3D50 : 1-2
				3D70 : 1-2
				3D2104: 1-4
				3D101 : 1-4

CAN::OpenPort

Opens the CAN socket

Syntax

int CAN::OpenPort(void);

Туре	Name	in/out	Description	Values
int			Return value	0 – success
				-1 - failure

CAN::WritePort

Writes a single CAN frame to the CAN port.

Syntax

int CAN::WritePort(PCANMSG TxMsg);

Туре	Name	in/out	Description	Values
PCANMSG	TxMsg	in	CAN frame to be written	
int			Return value	0 – success -1 - failure

CAN::ReadPort

Attempts to read a single CAN frame from the CAN port. Note that the CAN socket is configured to be non-blocking, so calls to ReadPort will return even if there is no data.

Syntax

int CAN::ReadPort(PCANMSG RxMsg);

Туре	Name	in/out	Description	Values
PCANMSG	RxMsg	in	CAN frame	
			received	
int			Return value	Bytes read or (-1) for failure

CAN::ClosePort

Closes the CAN socket

Syntax

void CAN::ClosePort(void);

Sample Code

```
#include ``can.h"
CANMSG TxMsg;
CANMSG RxMsg;
int bytesread = 0;
int cannum = 1; // CAN1
/* Init TX and RX message */
TxMsg.ID = 0x23;
TxMsq.Length = 8;
for (int i=0; i<8; i++)
      TxMsg.Data[i] = (0x11 * (i+1)); // fill random data
memset((void *)&RxMsg, 0, sizeof(CANMSG));
// CAN1
CAN can(cannum);
can.OpenPort();
can.WritePort(&TxMsg);
do
{
      bytesread = can.ReadPort(&RxMsg);
      // add delay
} while (bytesread != sizeof(CANMSG));
can.ClosePort();
```

Digital I/O Driver Interface

The Model 3D50 Display, Model 3D70 Display, Model 3D2104 Display, and Model 3D101 Display each have four digital inputs and four digital outputs, but they are configured differently and these differences will be explained. Each device uses the same library calls to read the digital inputs and set the digital outputs.

On the 3D50 Five Inch Display Pin 4 on its connector is a dedicated input only pin. Pin 5 is a dedicated output only pin. Pins 6, 7, and 8 are shared I/O pins that can be used to output a signal or input a signal.

On the Model 3D70 Seven Inch Display each of the four inputs are dedicated and so operate independently of any output pins.

On the Model 3D2104 10.4 Inch Display all digital output pins are shared I/O pins that can be used to output a signal or input a signal.

On the Model 3D101 10.1 Inch Display all digital output pins are shared I/O pins that can be used to output a signal or input a signal.

For a shared I/O pin to function as an input, the corresponding output must be set low.

Model 3D50 Pins	Model 3D70 Pins	Model 3D2104 and 3D101 Pins	
Input 1 (Pin 4)	Input 1 (Pin 4 Connector A)	Input 1 or Output 1 (Pin 10)	
Input 2 or Output 2 (Pin 6)	Input 2 (Pin 8 Connector B)	Input 2 or Output 2 (Pin 21)	
Input 3 or Output 3 (Pin 7)	Input 3 (Pin 9 Connector B)	Input 3 or Output 3 (Pin 32)	
Input 4 or Output 4 (Pin 8)	Input 4 (Pin 10 Connector B)	Input 4 or Output 4 (Pin 9)	
Output 1 (Pin 5)	Output 1 (Pin11 Connector B)		
	Output 2 (Pin12 Connector B)		
	Output 3 (Pin13 Connector B)		
	Output 4 (Pin14 Connector B)		

The following table summarizes all of the digital I/O pins for each model:

The following Grayhill files are required:

- ghiolib.h (header)
- libghio.so (linker)

Interface

A Qt application may set or get the digital I/O pin states by calling the appropriate C library function as described below.

#define GHIOLIB_CH1 (0x01)
#define GHIOLIB_CH2 (0x02)
#define GHIOLIB_CH3 (0x03)
#define GHIOLIB_CH4 (0x04)
#define GHIOLIB_DIG_IN_FLOAT (0)
#define GHIOLIB_DIG_IN_PULL_DN (1)
#define GHIOLIB_DIG_IN_PULL_UP (2)
#define GHIOLIB_RET_OK 0
#define GHIOLIB_RET_ERROR 1
#define GHIOLIB_RET_NOTSUPPORTED 2

ghiolib_setDigIncfg (Model 3D70 only)

Sets input pin pull-up/pull-down configuration.

Syntax

int ghiolib_setDigIncfg(int ch, uint8_t config);

Туре	Name	in/out	Description	Values
int	ch	in	Input pin to configure	GHIOLIB_CH1
				GHIOLIB_CH2
				GHIOLIB_CH3
				GHIOLIB_CH4
uint8_t	config	in	pin configuration	GHIOLIB_DIG_IN_FLOAT
	_			GHIOLIB_DIG_IN_PULL_DN
				GHIOLIB_DIG_IN_PULL_UP
int			Return value	GHIOLIB_RET_OK
				GHIOLIB_RET_ERROR
				GHIOLIB_RET_NOTSUPPORTED

ghiolib_getDigIn

This function reads the state of an input pin.

Syntax

int ghiolib_getDigIn(int ch, uint8_t *value);

Туре	Name	in/out	Description	Values
int	ch	in	Input pin to read	GHIOLIB_CH1
				GHIOLIB_CH2
				GHIOLIB_CH3
				GHIOLIB_CH4
uint8_t	value	out	pin status	0 – input low
				1 – input not low
int			Return value	GHIOLIB_RET_OK
				GHIOLIB_RET_ERROR
				GHIOLIB_RET_NOTSUPPORTED

ghiolib_getDigOut

Reads the current state of an output pin.

Syntax

int ghiolib_getDigOut(int ch, uint8_t *value);

Туре	Name	in/out	Description	Values
int	ch	in	Output pin to read	GHIOLIB_CH1
				GHIOLIB_CH2
				GHIOLIB_CH3
				GHIOLIB_CH4
uint8_t	value	out	pin status	0 – output low
				1 – output not low
int			Return value	GHIOLIB_RET_OK
				GHIOLIB_RET_ERROR
				GHIOLIB_RET_NOTSUPPORTED

ghiolib_setDigOut

This function sets the current state of an output pin.

Syntax

int ghiolib_setDigOut(int ch, uint8_t value);

Туре	Name	in/out	Description	Values
int	ch	in	Output pin to set	GHIOLIB_CH1
				GHIOLIB_CH2
				GHIOLIB_CH3
				GHIOLIB_CH4
uint8_t	value	in	pin status	0 – set pin output low
				10 – set pin output high
int			Return value	GHIOLIB_RET_OK
				GHIOLIB_RET_ERROR
				GHIOLIB_RET_NOTSUPPORTED

```
#include <QDebug>
// For access to ghiolib
typedef u_int16_t uint16_t;
typedef u int8 t uint8 t;
#ifdef cplusplus
extern "C" {
#endif
#include "ghiolib.h"
#ifdef cplusplus
}
#endif
int
      channel;
uint8_t digValue;
int gpioOutput;
int
      gpioInput;
int
       gpioStatus;
// Set inputs to pull down mode and read current inputs and outputs
// for each channel
gpioOutput = 0;
gpioInput = 0;
for (channel = 0; channel < GHIOLIB MAX DIGITAL IO; channel++)</pre>
{
    // Set input to pull down mode
    gpioStatus = ghiolib setDigIncfg(channel + 1,
GHIOLIB_DIG_IN_PULL_DN);
    if ((GHIOLIB_RET_OK != gpioStatus) && (GHIOLIB_RET_NOTSUPPORTED !=
gpioStatus))
    {
        qDebug("ERROR (%d) doing ghiolib setDigIncfg on channel: %d\n",
```

```
gpioStatus, channel + 1);
    }
    // Read current output setting
    digValue = 0;
    gpioStatus = ghiolib getDigOut(channel + 1, &digValue);
    if (GHIOLIB RET OK != gpioStatus)
    {
        qDebug("ERROR (%d) doing ghiolib getDigOut on channel: %d\n",
               gpioStatus, channel + 1);
    }
    else
    {
        if (1 == digValue)
        {
            gpioOutput |= (1 << channel);</pre>
        }
    }
    // Read current input
    digValue = 0;
    gpioStatus = ghiolib getDigIn(channel + 1, &digValue);
    if (GHIOLIB RET OK != gpioStatus)
    {
        qDebug("ERROR (%d) doing ghiolib getDigIn on channel: %d\n",
               gpioStatus, channel + 1);
    }
    else
    {
        if (1 == digValue)
        {
            gpioInput |= (1 << channel);</pre>
        }
    }
qDebug("GPIO initial output: 0x%x input: 0x%x\n", gpioOutput,
gpioInput);
```

}

LCD

The Grayhill 3Dxx Series Display uses a 16 bit per pixel LCD screen. The pixel dimensions of various 3Dxx Display products are shown in the section Error! Reference source not found.. The default orientation of the frame buffer is landscape mode (wider pixel dimension is in horizontal direction).

LCD Backlight

The LCD Backlight setting is a value between 0 (minimum) and 100 (maximum) inclusive. The brightness value can be set in the file /sys/class/backlight/pwm-backlight.0/brightness

Sample Code

```
int value = 80;
QFile file("/sys/class/backlight/pwm-backlight.0/brightness");
if (file.open(QIODevice::WriteOnly | QIODevice::Text))
{
    QTextStream out(&file);
    out << value;
    file.close();
}
```

Appendix F: Setting 3Dxx Flash File System R/W Mode

To immediately configure the 3Dxx Display file system for read-write mode enter the following command in a terminal window on the display:

• mount –o remount,rw /

The above command only remains in effect until the next reboot.

The 3Dxx installation script utilizes the following technique to configure the 3Dxx Display file system to be in read-write mode to make Qt development more convenient.

The above command is placed in a script file which is invoked during display initialization. This file /home/writeablefs is called via the following link:

• In -s /home/writeablefs /etc/rc.d/S03writeablefs

To leave the display in read-only mode:

• mv /home/writeablefs /home/writeablefs.bak

Alternatively, to set the 3Dxx display file system to read-write mode on boot-up, edit the file /etc/init.d/rc-once and add the above command to the end of this file just before the final "exit" command like this:

To leave the 3Dxx Display file system set to read-only mode on boot-up, edit the file /etc/init.d/rc-once and remove the "mount –o remout,rw /" line near the end of the file (or comment it out by putting a "#" in column one of that line).

Appendix G: Building Qt Library Source (optional)

N.B. This appendix is included for reference and is not a required step.

N.B. Library building is supported only under Linux.

This section describes the procedure to download and build the Qt library source code.

Please reference <u>https://doc.qt.io/qt-5/windows-requirements.html</u> for additional information.

This procedure relies on Qt Creator and the Grayhill support files being installed.

Grayhill provides an archive script with the following files:

- mkLibs script used to build the libraries
- o qmake.conf Grayhill modified configuration file
- o qt_configure.prf Grayhill modified rule file
- qtLibSrcInstall Qt library installation script
- o updlnc fix incorrectly generated makefile
- Download the Qt library installation archive from Grayhill
 qtLibSrc5122.tgz
- Copy/move the downloaded file to /home/ghguest
- Un archive

 tar xf qtLibSrc5122.tgz
- Run installer (this fetches the library source from Qt)

 ./qtLibSrcInstall
- Build Libs
 - Follow instructions provided by installation script; mkLibs provides the root password as needed

N.B. To reduce build time, increase the VM's processor count. This needs to be done in a VM powered off state.

Appendix H: Dynamic IP Address

To find the 3Dxx display's Ethernet IP address, issue the following command in a terminal window on the display

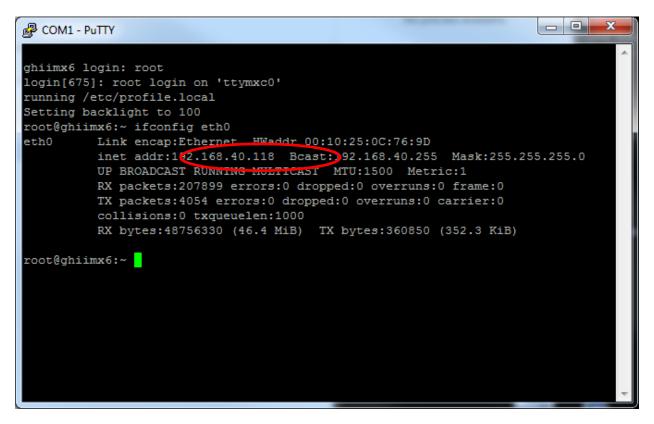
• ifconfig eth0

The IP address of the 3Dxx Display is displayed after the tag "inet addr:" and is circled in red in the example output shown below.

If the tag "inet addr:" is not present; enter these commands and try the "ifconfig eth0" command again

- ifdown eth0
- ifup eth0

Please make a note of the IP address, in this example the IP address is 192.168.40.118



Appendix I: Static IP Address

If using a **static** IP address for the display, once the address is determined:

- cp /etc/network/interfaces /etc/network/interfaces.bak
- vi /etc/network/interfaces
- replace lines

```
iface eth0 inet dhcp
udhcpc_opts -t 5 -T 3 -A 20 -S &
```

with

```
iface eth0 inet static
address 192.168.40.118
netmask 255.255.255.0
```

Google "linux interface file" for additional information.