DWIN Linux LCD Screen (40 series) QT Cross-compile Configuration

1 Install QtCreator 1

1.1 System Requirements 1

1.2 Download QtCreator 1

1.3 Install QtCreator 1

2 Configuring the cross-compilation environment 5

2.1 Run the QtCreator 5

2.2 Configuring the cross-compilation environment 5

3 Compile Qt project 10

3.1 Open the project 10

3.2 To add environment variables 11

3.3 Run qmake 12

3.4 Build 14

1 Install Qt Creator

1.1 System Requirements

This document is based on Ubuntu 14.04 system for verification. Other versions of Ubuntu systems should work but are not verified.

1.2 Download Qt Creator

The version of Qt Creator used in this document is 2.7.2. Please download the version that matches the operating system.

source/	02-Jul-2013 19:43	-	
qt-creator-windows-opensource-2.7.2.exe	02-Jul-2013 19:43	53M	Details
qt-creator-mac-opensource-2.7.2.dmg	02-Jul-2013 19:43	53M	Details
qt-creator-linux-x86_64-opensource-2.7.2.bin	02-Jul-2013 19:43	62M	Details
qt-creator-linux-x86-opensource-2.7.2.bin	02-Jul-2013 19:43	63M	Details
qt-creator-2.7.2-src.zip	02-Jul-2013 19:43	27M	Details
qt-creator-2.7.2-src.tar.gz	02-Jul-2013 19:43	22M	Details

1.3 Install Qt Creator

Copy the installer to your Ubuntu system and add execute permissions to the file:

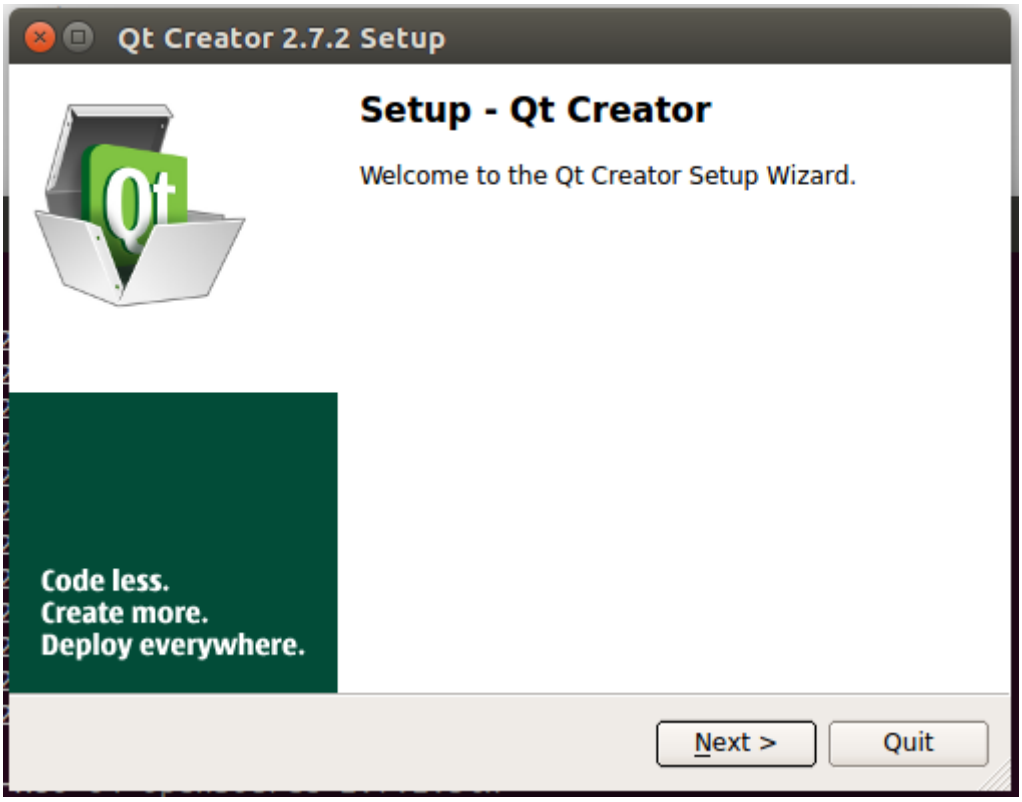
chmod +x qt-creator-linux-x86_64-opensource-2.7.2.bin

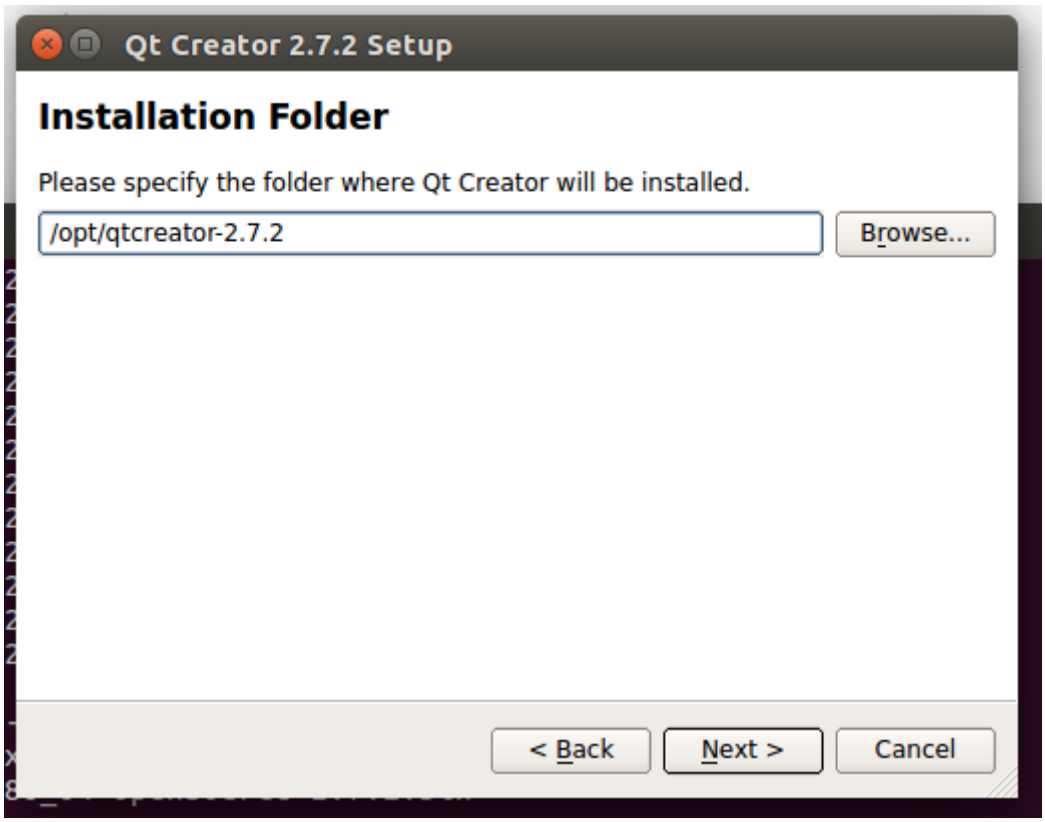
```
dwin@ubuntu:~$ cd /home/dwin
dwin@ubuntu:~$ chmod +x qt-creator-linux-x86_64-opensource-2.7.2.bin
```

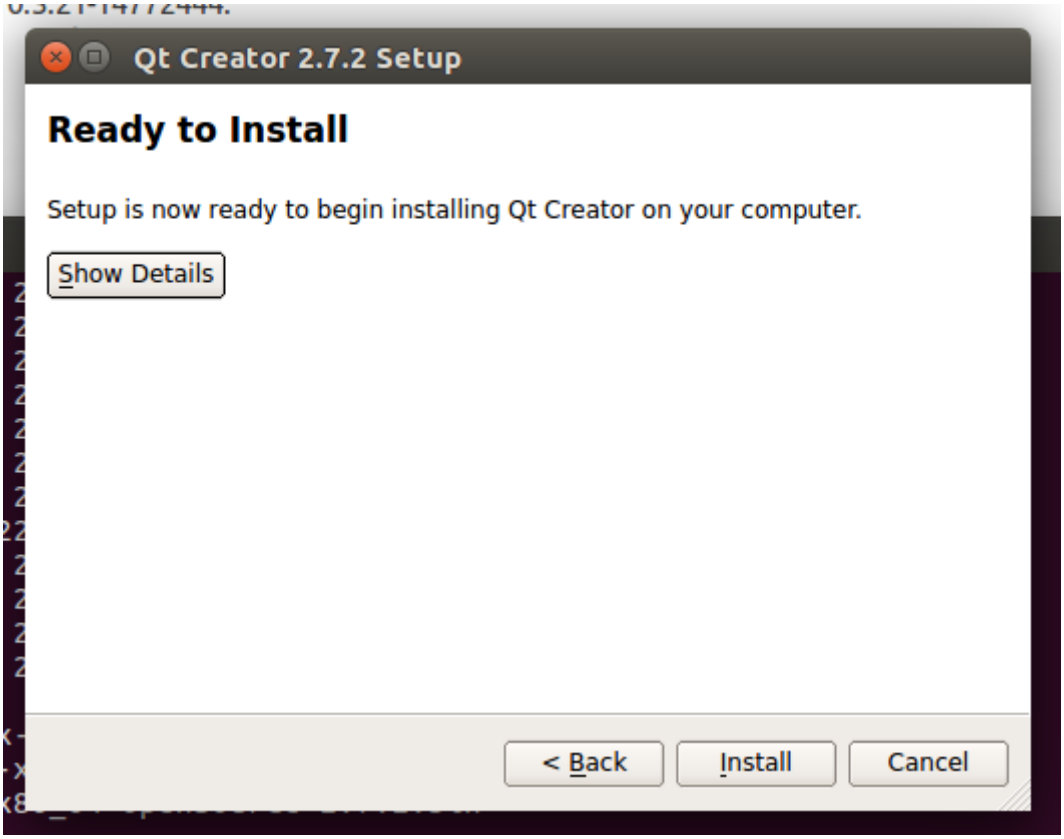
Run the installer

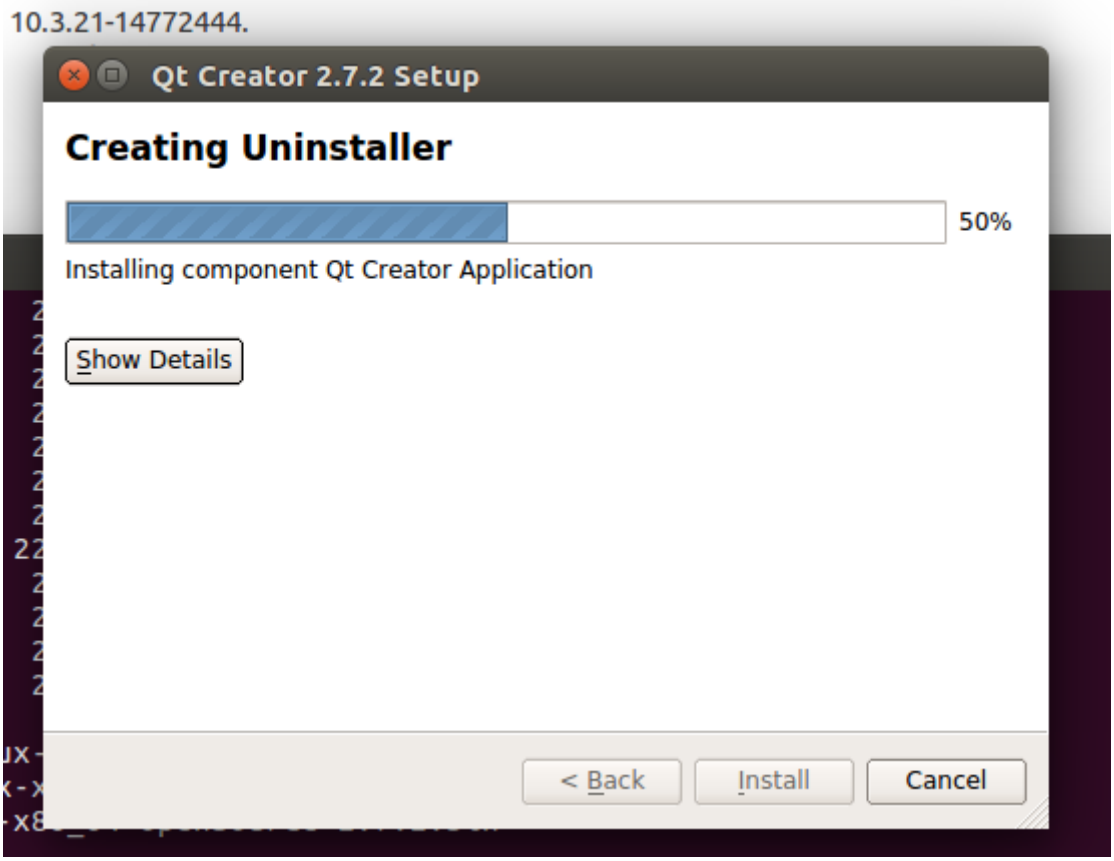
sudo ./qt-creator-linux-x86_64-opensource-2.7.2.bi

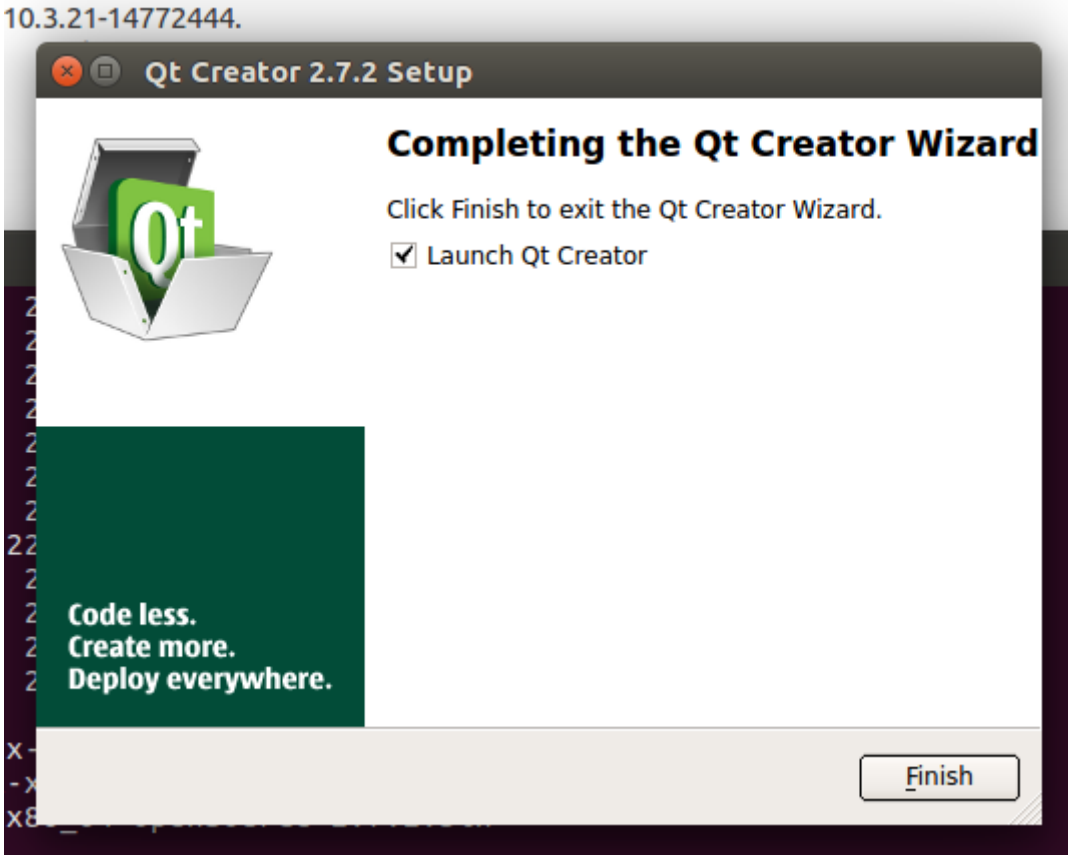
Click the “Next”:


Qt Creator 2.7.2 Setup
Setup - Qt Creator
Welcome to the Qt Creator Setup Wizard.
Code less.
Create more.
Deploy everywhere.
Next >
Quit


Qt Creator 2.7.2 Setup
Installation Folder
Please specify the folder where Qt Creator will be installed.
/opt/qtcreator-2.7.2
Browse...
< Back
Next >
Cancel


Qt Creator 2.7.2 Setup
Ready to Install
Setup is now ready to begin installing Qt Creator on your computer.
Show Details
< Back
Install
Cancel


10.3.21-14772444.
Qt Creator 2.7.2 Setup
Creating Uninstaller
50%
Installing component Qt Creator Application
Show Details
< Back
Install
Cancel

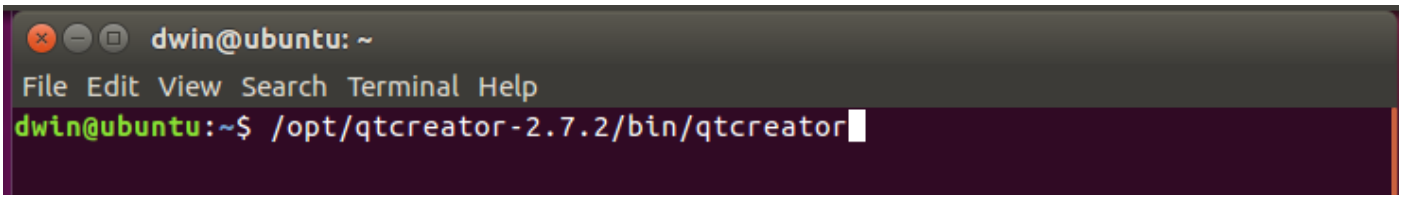

10.3.21-14772444.
Qt Creator 2.7.2 Setup
Completing the Qt Creator Wizard
Click Finish to exit the Qt Creator Wizard.
Launch Qt Creator
Code less.
Create more.
Deploy everywhere.
Finish

2 Set up the cross-compilation environment

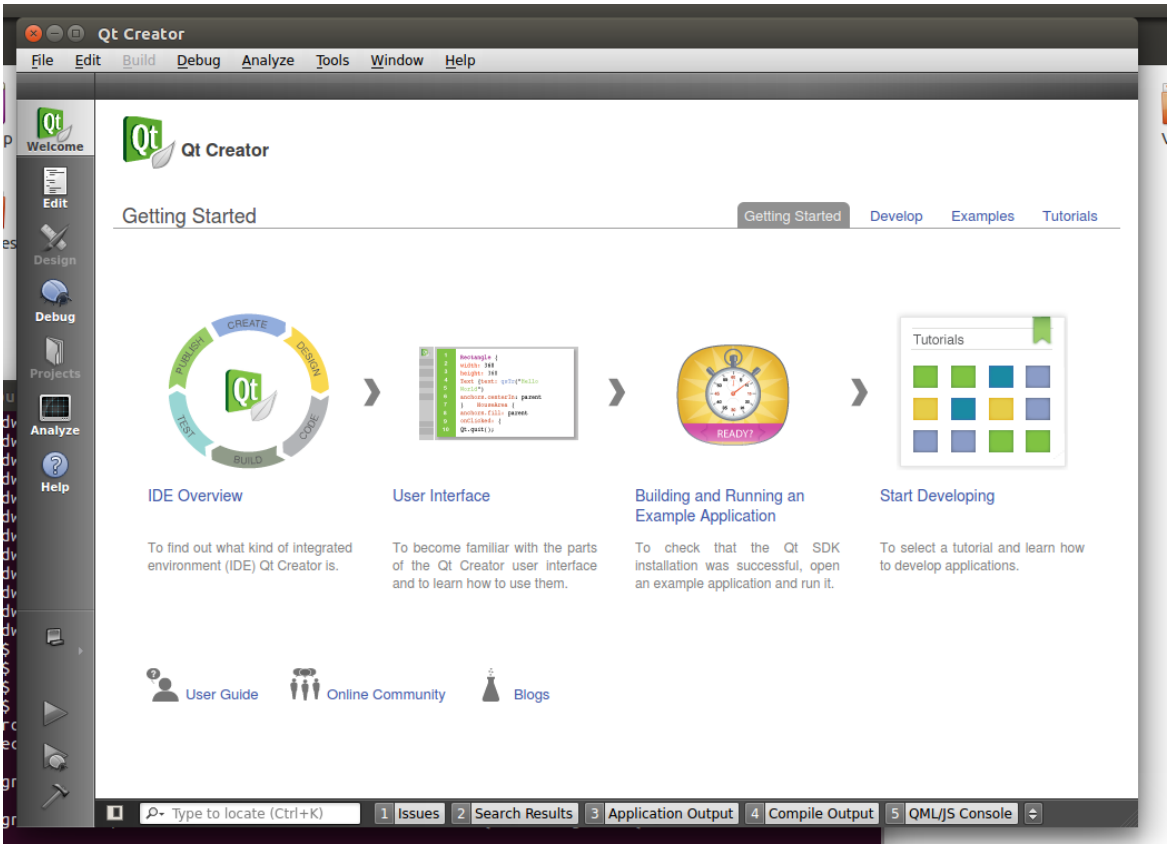
2.1 Run the Qt Creator

The Qt Creator executable file is in the bin directory of the installation directory.

/opt/qtcreator-2.7.2/bin/qtcreator

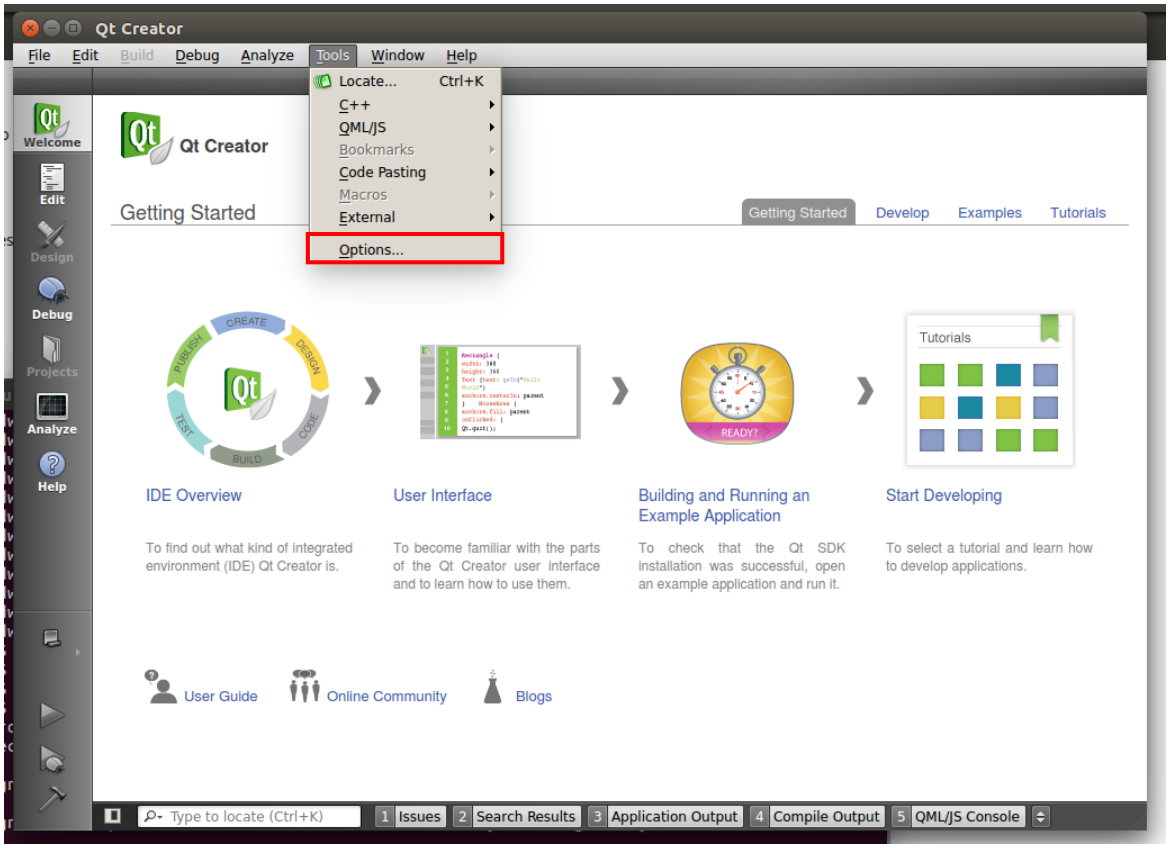


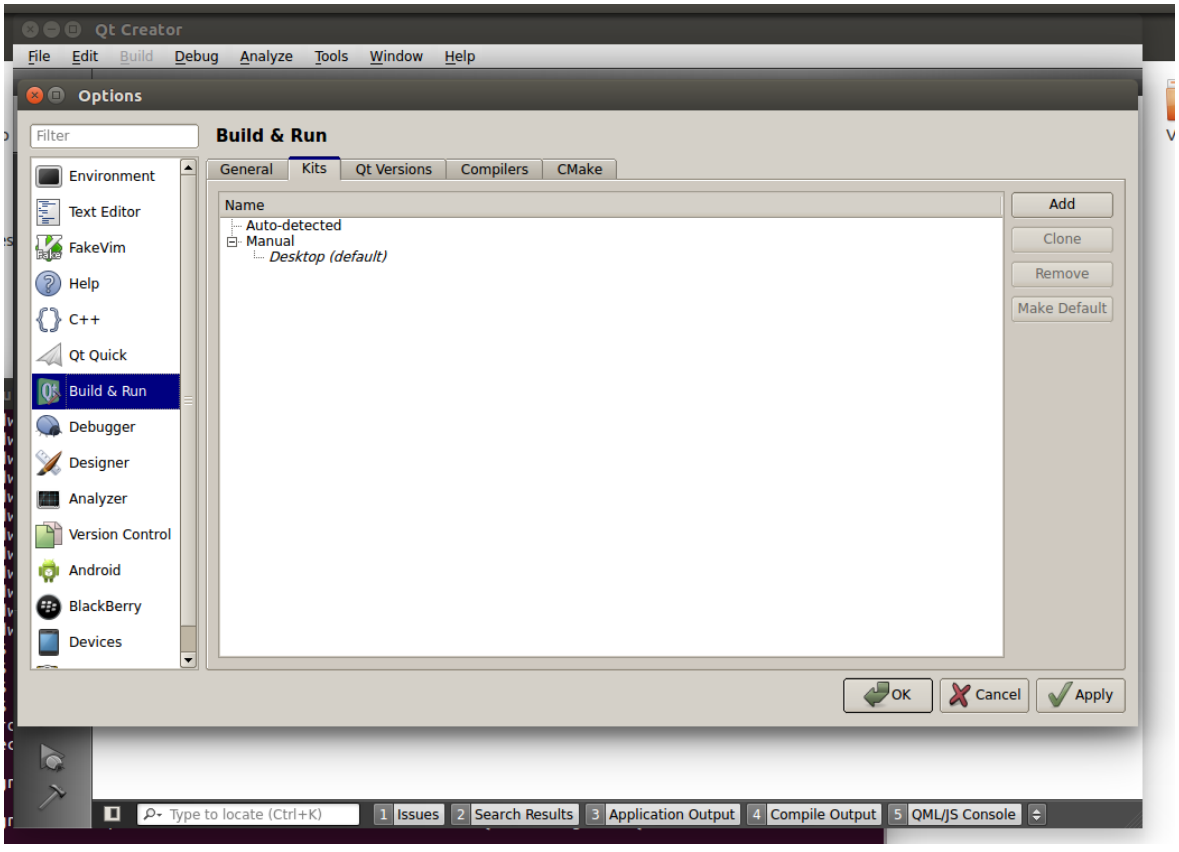
The interface of software is as below.



2.2 Set up the cross-compilation environment

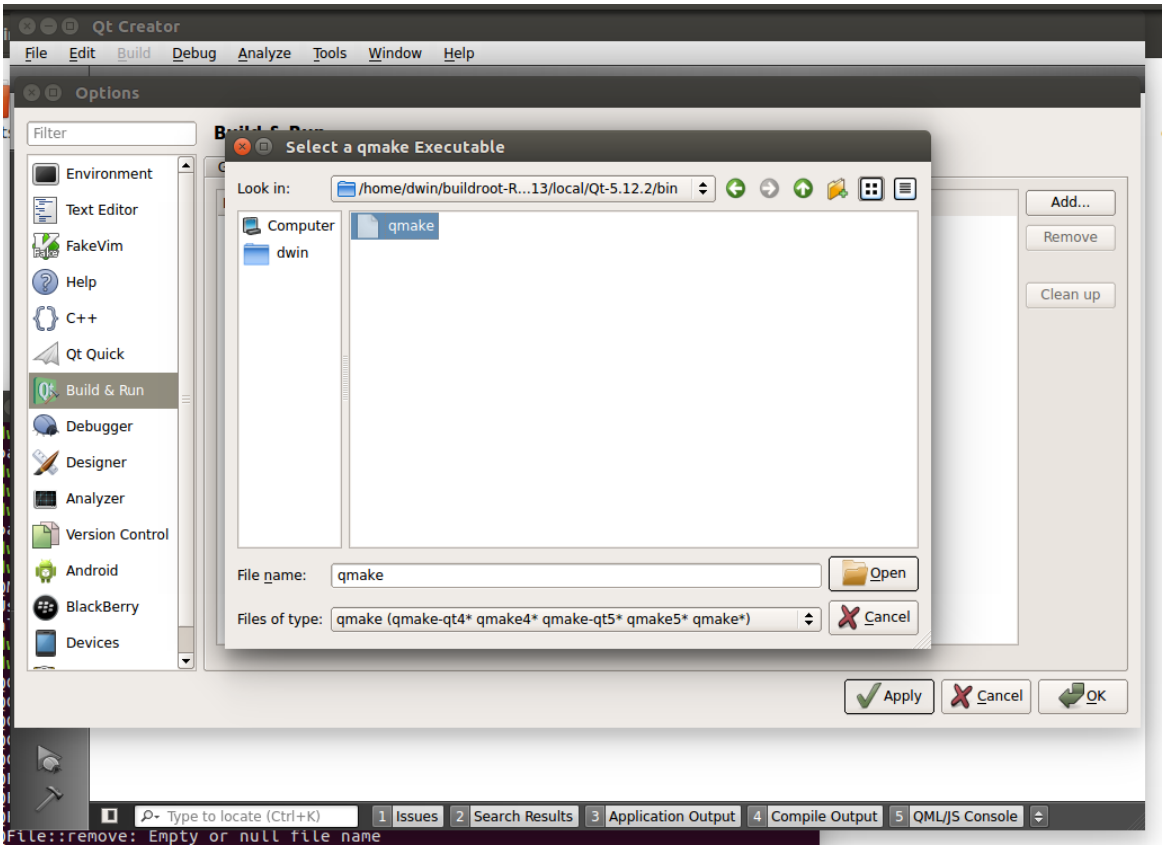
Choose [tool] – [options] as below.

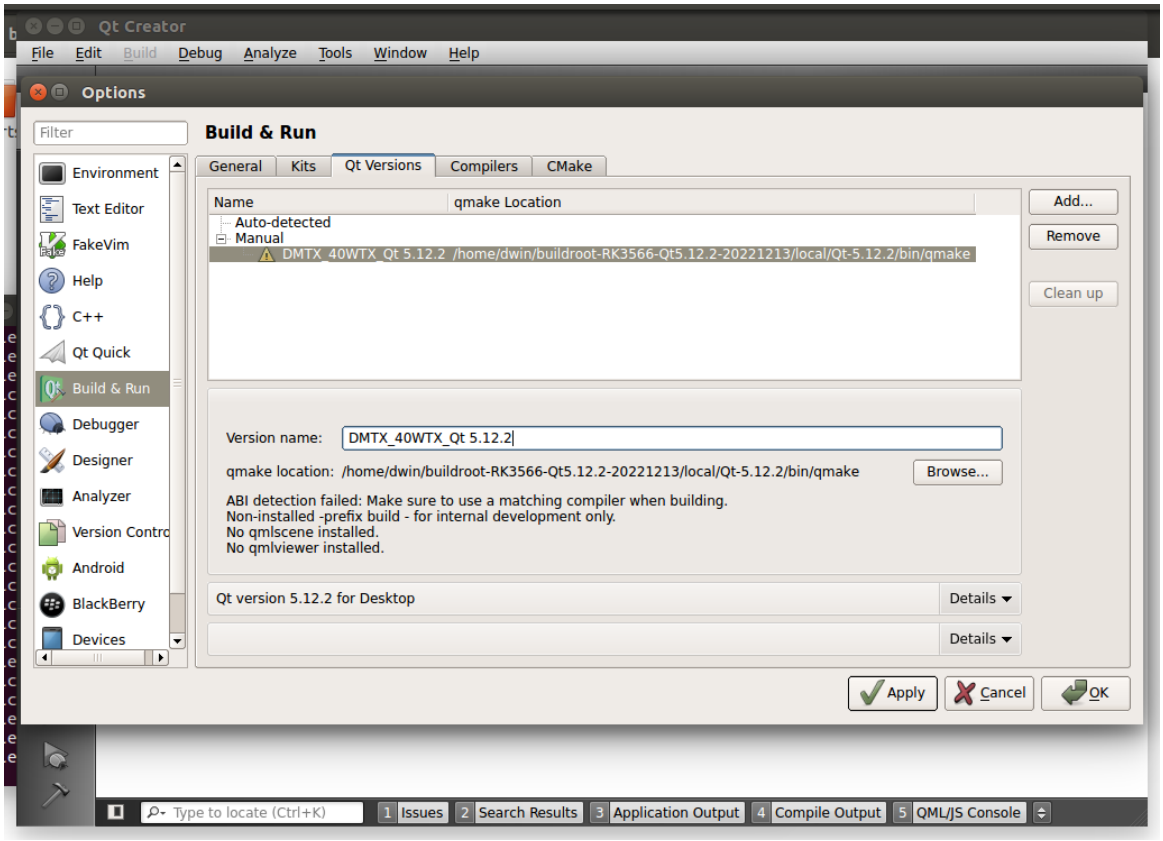




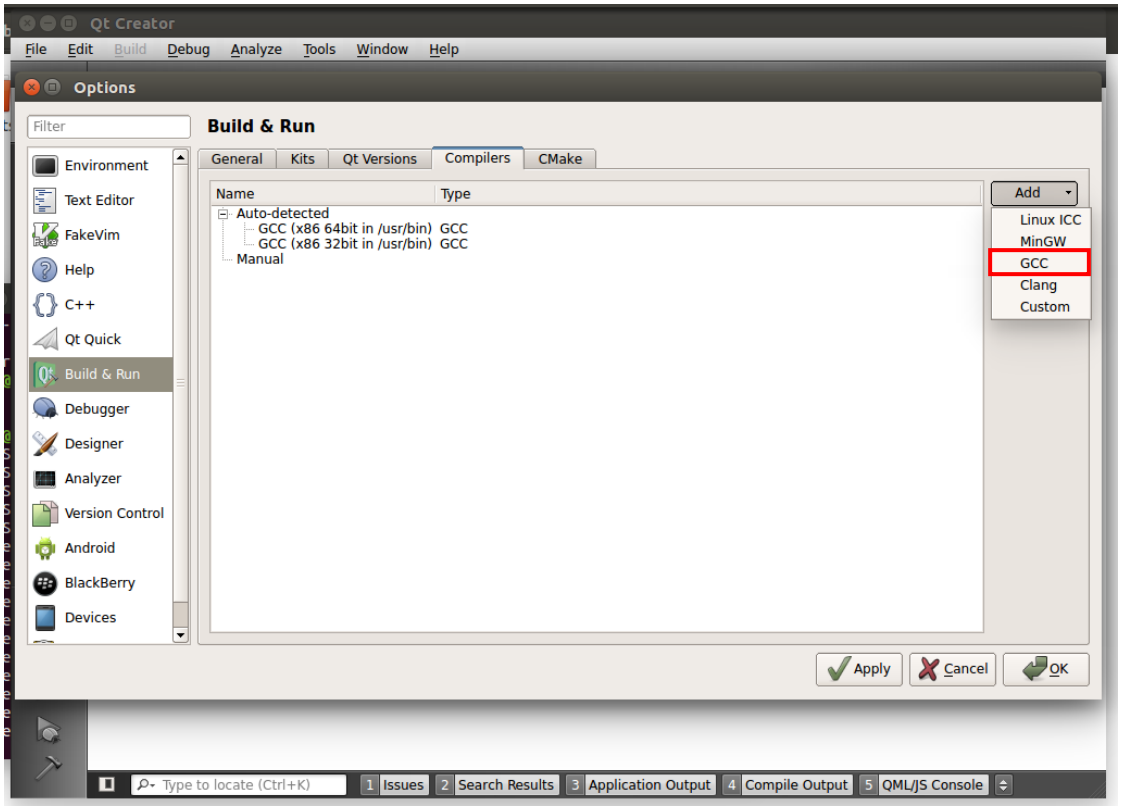
Set qmake: choose [Build & Run] – [Qt Version] – [Add],

"qmake" is in the 'local/Qt-5.12.2/bin/' directory of buildroot-RK3566-Qt5.12.2-20221213.tar.gz

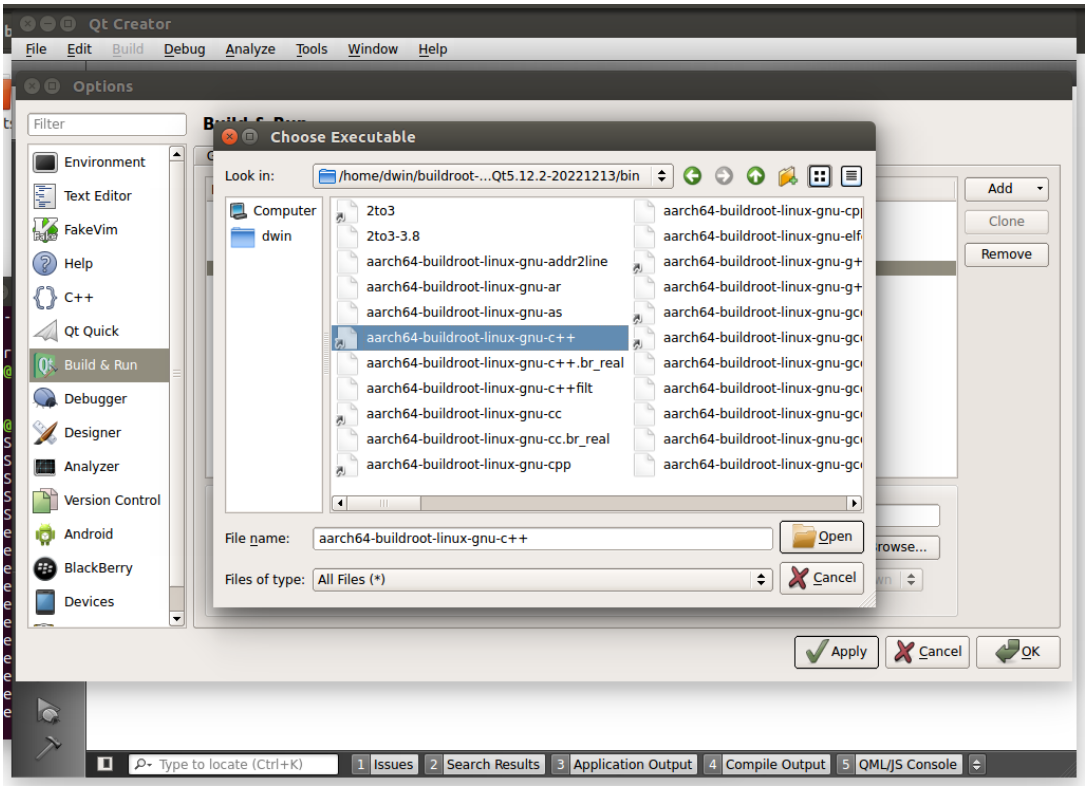


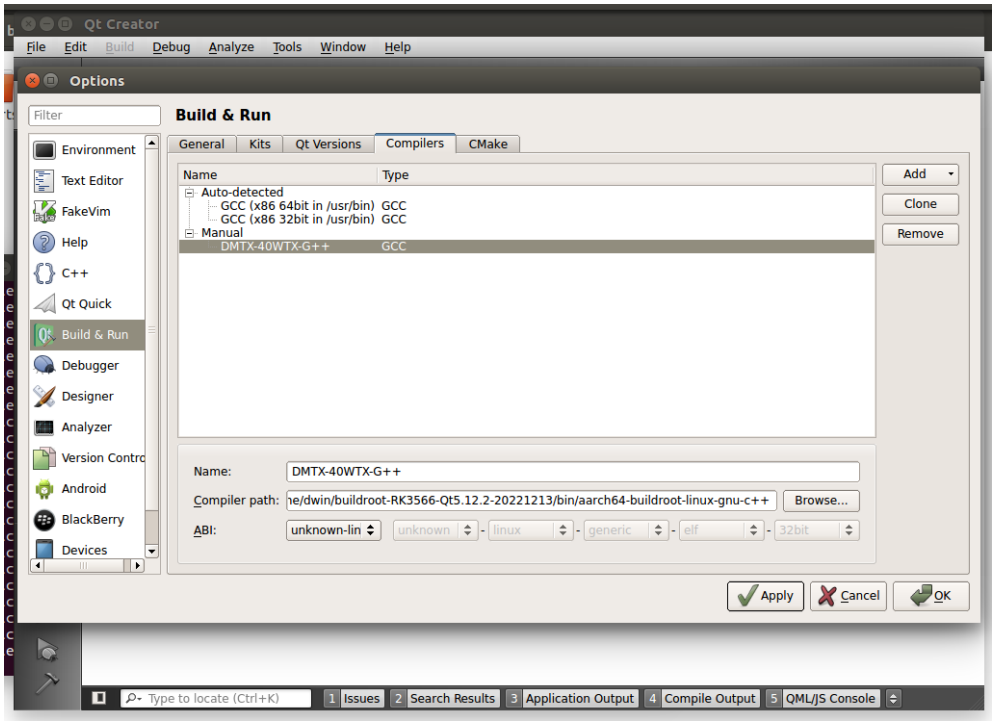


Set compilation toolchain: choose [Build & Run] – [Compilers] – [Add] – [GCC]:

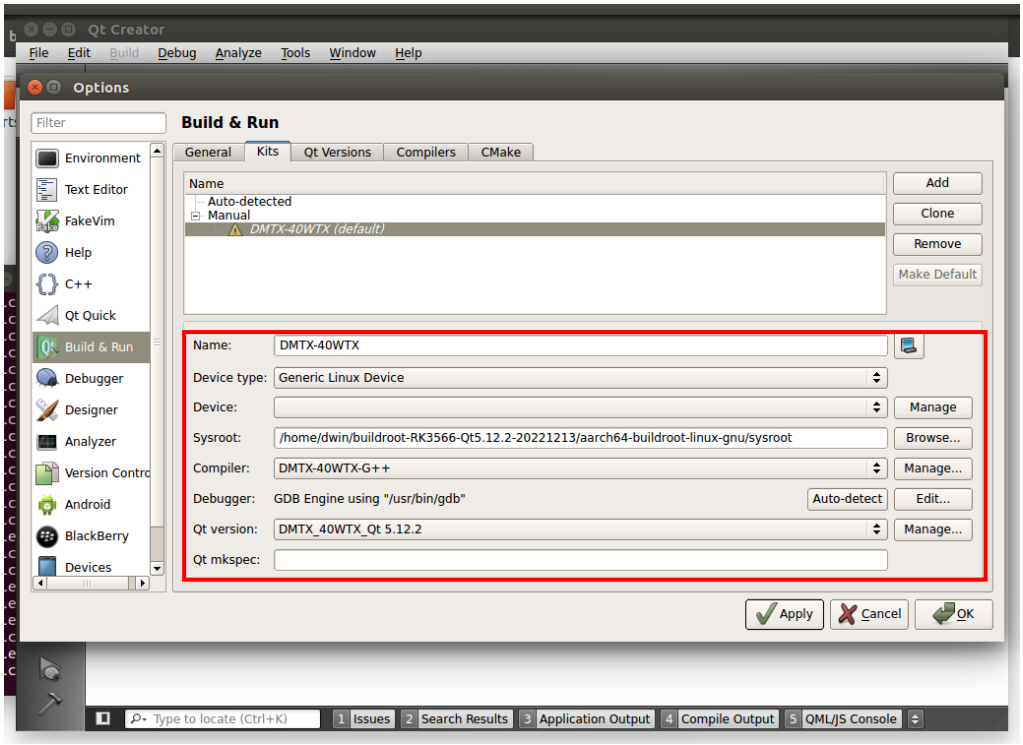


The compiler is located in the 'bin' directory of the 'buildroot-RK3566-Qt5.12.2-20221213.tar.gz' pac kage.





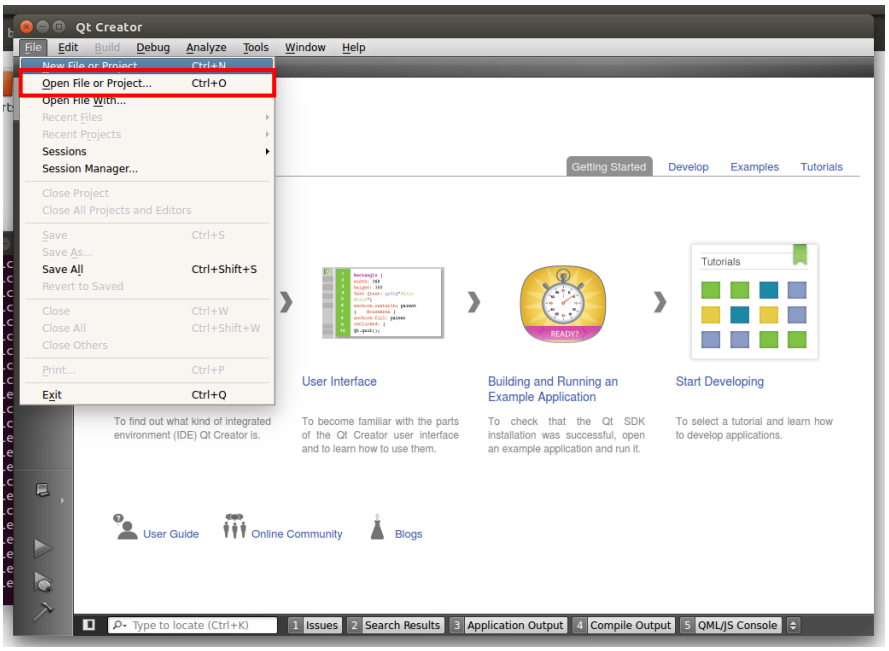
Set up the build kit: choose [Build & Run] – [Kits]:



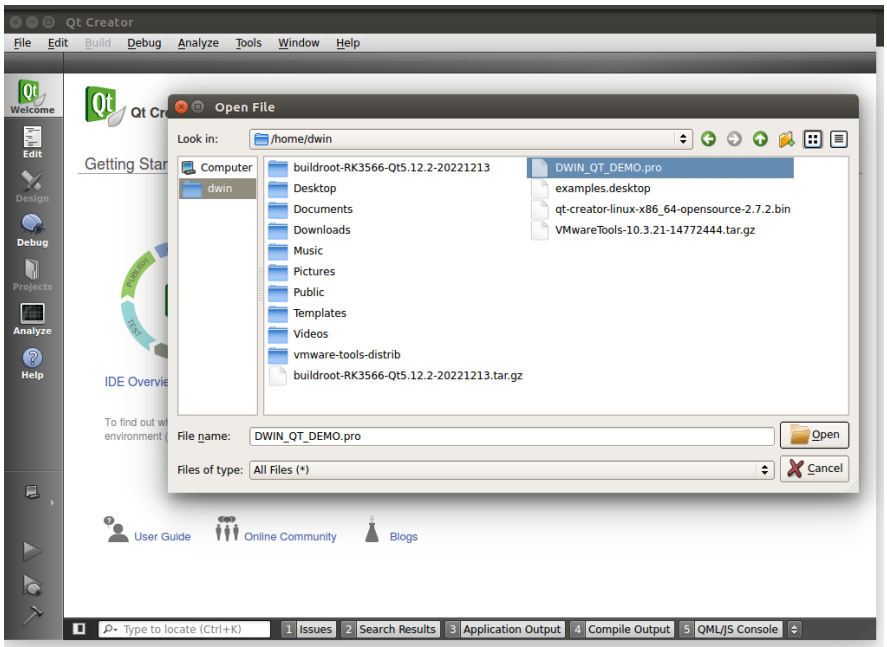
3 Compile Qt project

3.1 Open the project

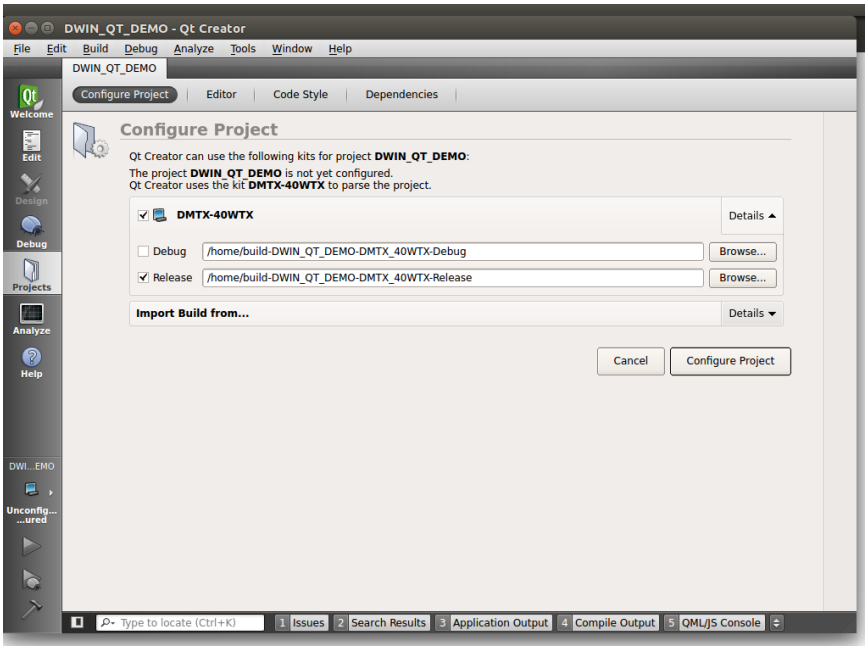
[File] – [open file or project]:



Choose the Qt project:



Configure the project:



3.2 **Add environment variables**

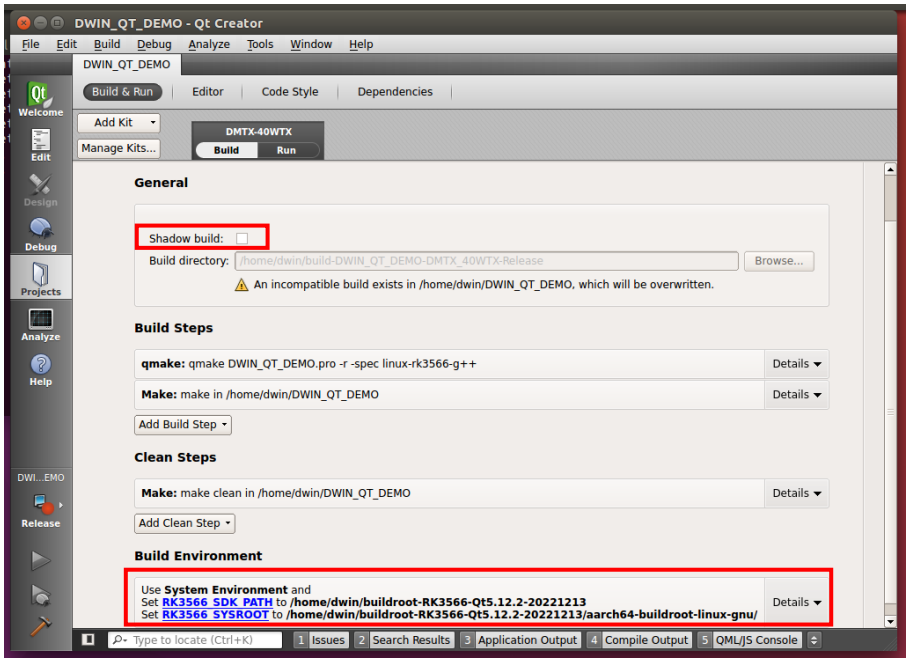
Go to [Projects] – [Build & Run[- [Build Environment], and add a variable:

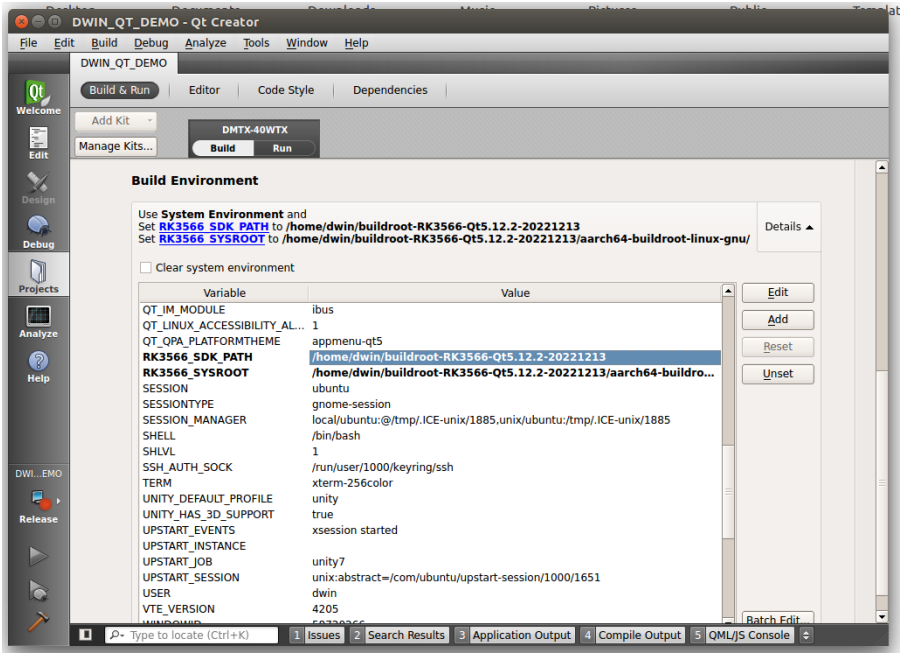
Variable Name 1: RK3566_SDK_PATH

Value 1: Root directory of the 'buildroot-RK3566-Qt5.12.2-20221213.tar.gz' package

Variable Name 2: RK3566_SYSROOT

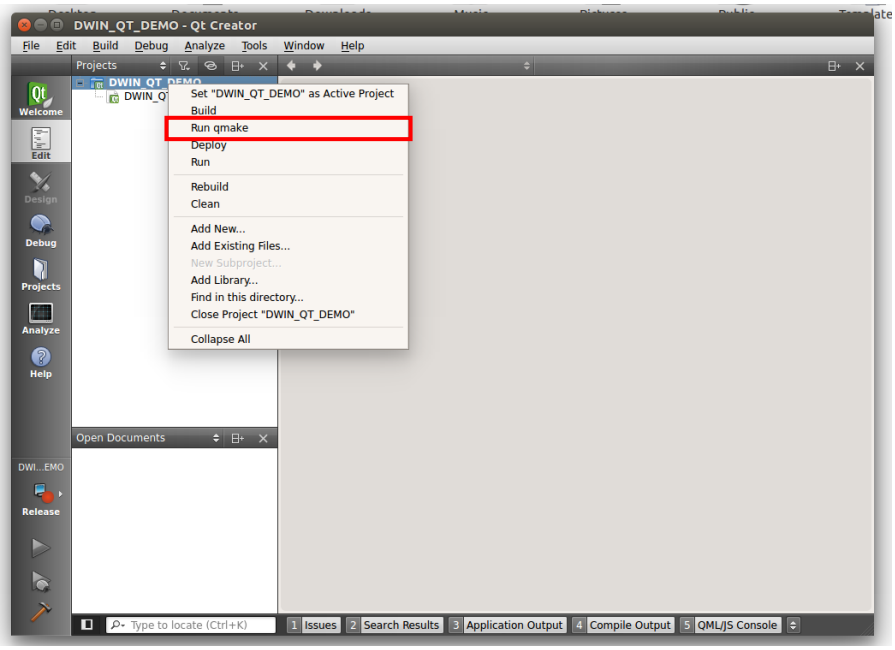
Value 2: 'sysroot' directory in the 'aarch64-buildroot-linux-gnu' directory of the 'buildroot-A40i-QT-sdk-20220623.tar.gz' package.



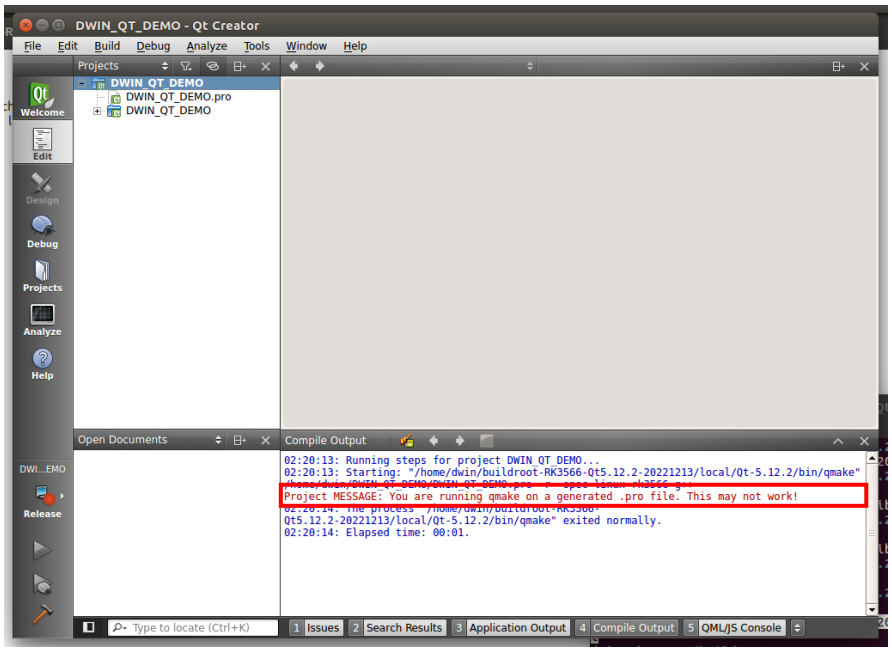


3.3 **Run qmake**

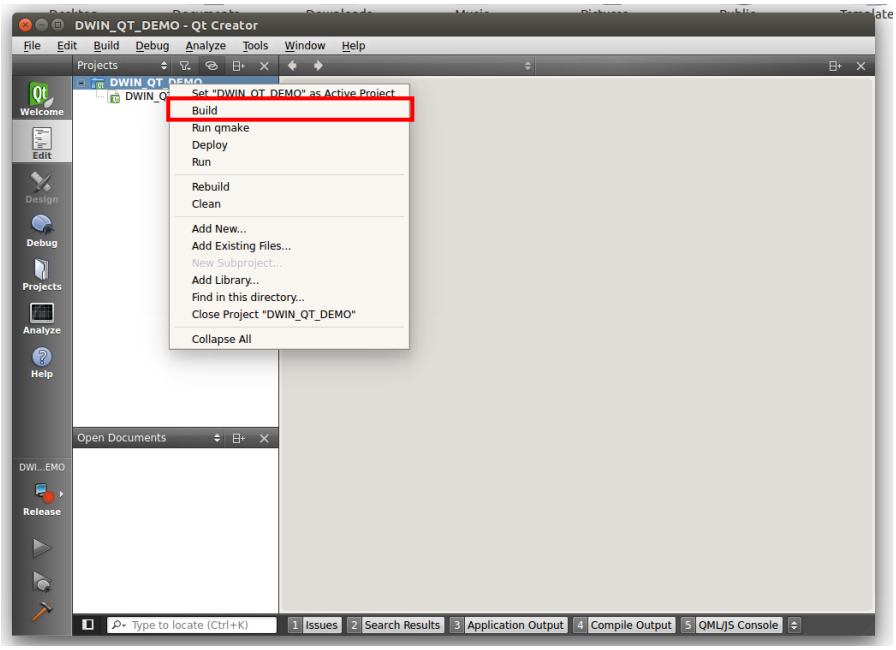
Choose project, ‘right key’ – run qmake

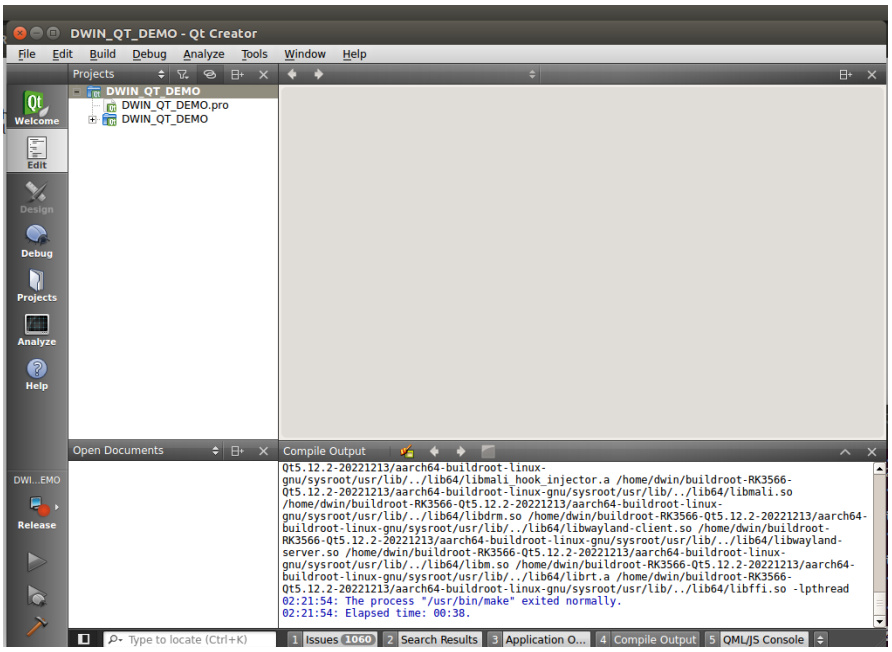


When qmake is successful, it looks like the image below (the red part is the printout of DWIN_QT_DEMO.pri, which does not affect).

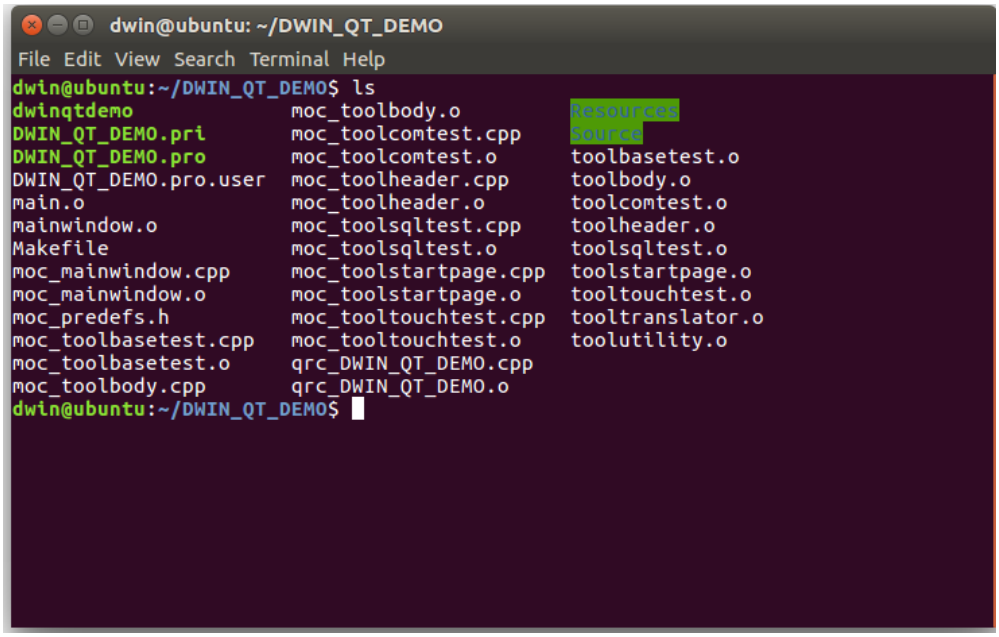


3.4 **Build**





At this point, the target files have been generated in the project directory and can be copied to the screen for execution.


dwin@ubuntu: ~/DWIN_QT_DEMO
File Edit View Search Terminal Help
dwin@ubuntu:~/DWIN_QT_DEMO$ ls
dwinqtdemo moc_toolbody.o Resources
DWIN_QT_DEMO.pri moc_toolcomtest.cpp Source
DWIN_QT_DEMO.pro moc_toolcomtest.o toolbasetest.o
DWIN_QT_DEMO.pro.user moc_toolheader.cpp toolbody.o
main.o moc_toolheader.o toolcomtest.o
mainwindow.o moc_toolsqltest.cpp toolheader.o
Makefile moc_toolsqltest.o toolsqltest.o
moc_mainwindow.cpp moc_toolstartpage.cpp toolstartpage.o
moc_mainwindow.o moc_toolstartpage.o tooltouchtest.o
moc_predefs.h moc_tooltouchtest.cpp tooltranslator.o
moc_toolbasetest.cpp moc_tooltouchtest.o toolutility.o
moc_toolbasetest.o qrc_DWIN_QT_DEMO.cpp
moc_toolbody.cpp qrc_DWIN_QT_DEMO.o
dwin@ubuntu:~/DWIN_QT_DEMO$

Revision Records

Rev	Revise Date	Content	Editor
00	2023-06-09	First Edition	Zheng Jianxin

Please contact us if you have any questions about the use of this document or our products, or if you would like to know the latest information about our products:

Customer service Tel: +86-400-018-9008

Customer service email: dwinhmi@dwin.com.cn

Website: www.dwin-global.com

DWIN Developer Forum: https://forums.dwin-global.com/index.php/forums

Thank you all for continuous support of DWIN, and your approval is the driving force of our progress!

Important Disclaimer

DWIN reserves the right to make any changes to product designs without prior notice.

Customers should ensure strictly adhering to all the relevant standards and requirements during the product application process, including but not limited to functional safety, information security, and regulatory provisions.

DWIN shall not bear any joint and several liability for any consequences that may arise from customers' adoption of DWIN products. In particular, for risks that may lead to significant property losses, environmental hazards, personal injury, or even death, especially in high-risk application areas such as military applications, flammable and explosive places, and life-saving medical equipment, customers should independently assess the risks and take corresponding preventive and protective measures. DWIN shall not bear any relevant responsibility.