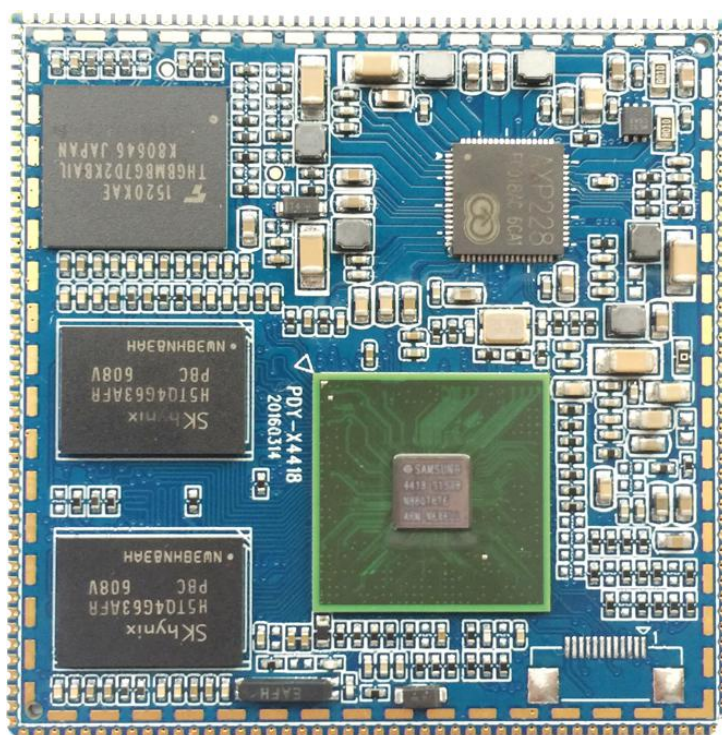


Linux QT 文件系统的搭建



深圳葡萄雨技术有限公司

www.graperain.cn

版权声明

本手册版权归属深圳市葡萄雨技术有限公司所有，并保留一切权力。非经葡萄雨技术有限公司同意(书面形式)，任何单位及个人不得擅自摘录本手册部分或全部，违者我们将追究其法律责任。

敬告：在售开发平台的手册会经常更新，请在<http://www.graperain.cn/>网站下载最新手册或与我司销售联系取得，不再另行通知。

版本说明

| 版本号 | 日期 | 作者 | 描述 |
|--------|----------|-------------|------|
| Rev.01 | 2016-4-7 | David Huang | 修订版本 |
| | | | |

技术支持

如果您对文档有所疑问，您可以拨打技术支持电话或 E-mail 联系。

网 址：<http://www.graperain.cn/>

联系电话：0755-23025312

E-mail：info@graperain.com

销售与服务网络

公司：深圳市葡萄雨技术有限公司

地址：深圳市宝安区西乡街道银田路 4 号

邮编：518101

电话：0755-23025312

网址：<http://www.graperain.cn/>

目录

版权声明..... 错误！未定义书签。

第 1 章 Linux QT 文件系统的搭建..... 5

 1.1 下载 buildroot..... 5

 1.2 配置 buildroot..... 5

 1.3 编译 buildroot..... 10

 1.4 测试 QT5.4 默认示例..... 11

 1.5 安装 QT Creator..... 15

 1.6 编译 QT Creator 默认示例..... 16

第 2 章 产品线介绍..... 29

 2.1 核心板系列..... 29

 2.2 开发板系列..... 29

 2.3 卡片电脑系列..... 29



第 1 章 Linux QT 文件系统的搭建

对于一个平台，我们如何构建 linux QT 文件系统呢？本章节将会带您一步步搭建 linux 文件系统。

1.1 下载 buildroot

在 buildroot 官网下载最新的 buildroot 包，下载地址如下：

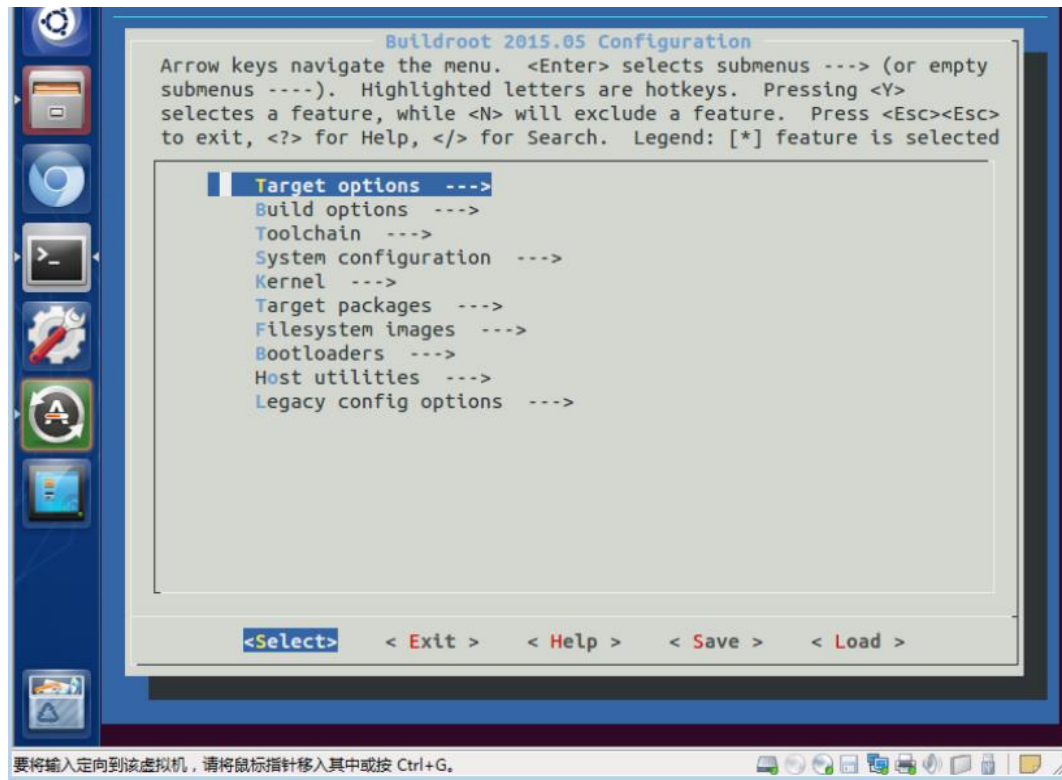
<http://buildroot.uclibc.org/download.html>

通常选择最新版本的下載即可。

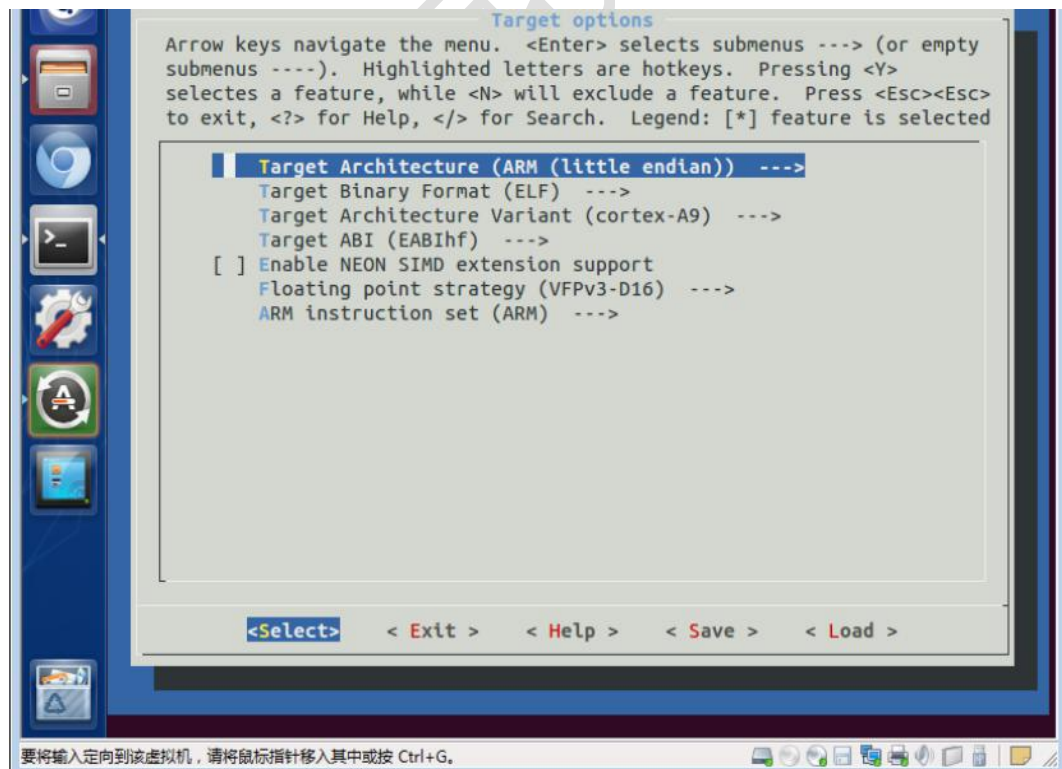
1.2 配置 buildroot

将下载的 buildroot 包拷贝到 ubuntu 系统用户目录并解压，得到 buildroot 目录，通过命

令终端进入 buildroot 目录，执行 make menuconfig，进入配置界面：



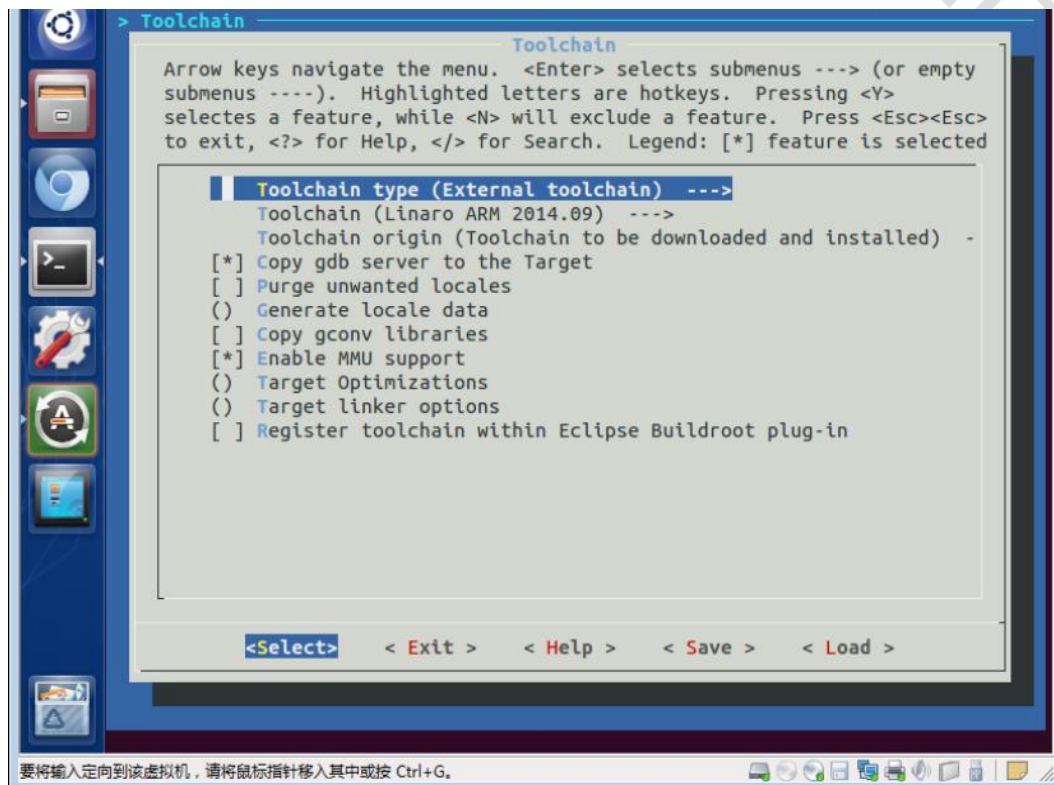
进入 Target options 菜单，



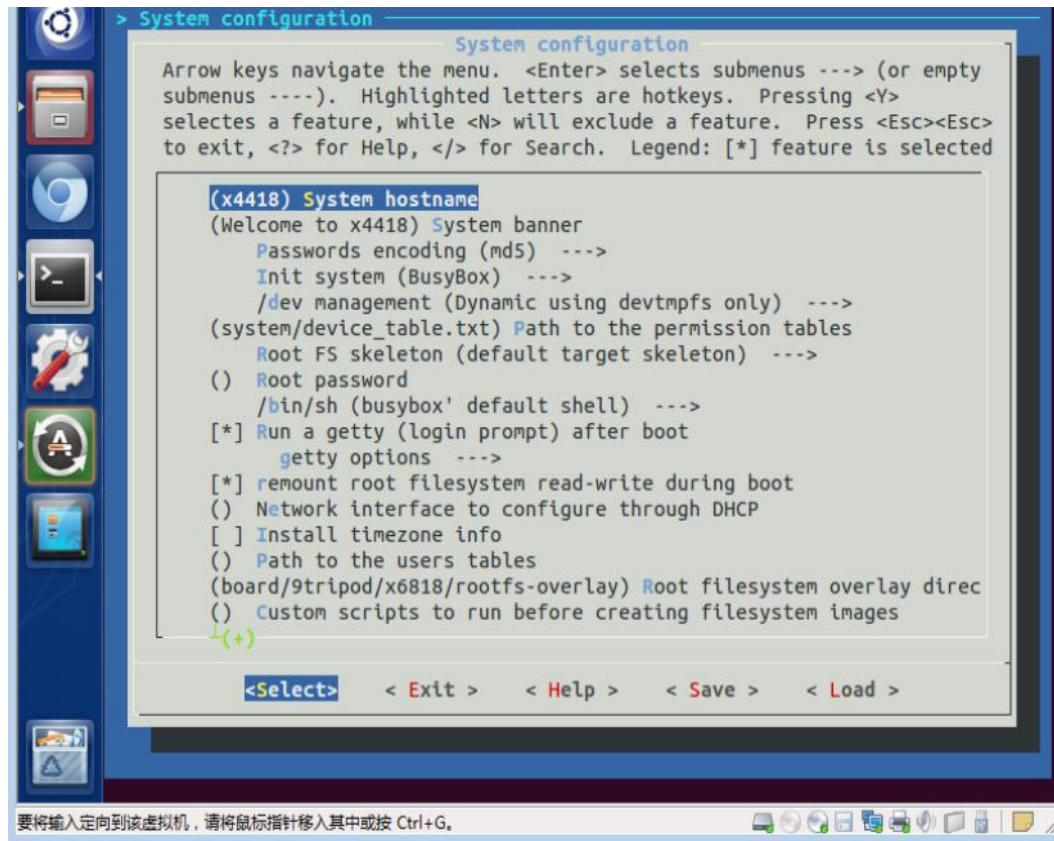
在 Target Architecture 中选择 ARM(little endian) , 在 Target Binary Format 中选择 ELF ,
Target Architecture Variant 中选择 cortex-A9 , Target ABI 选择 EABIhf , Floating point
strategy

选择 VFPv3-D16 , ARM instruction set 选择 ARM。

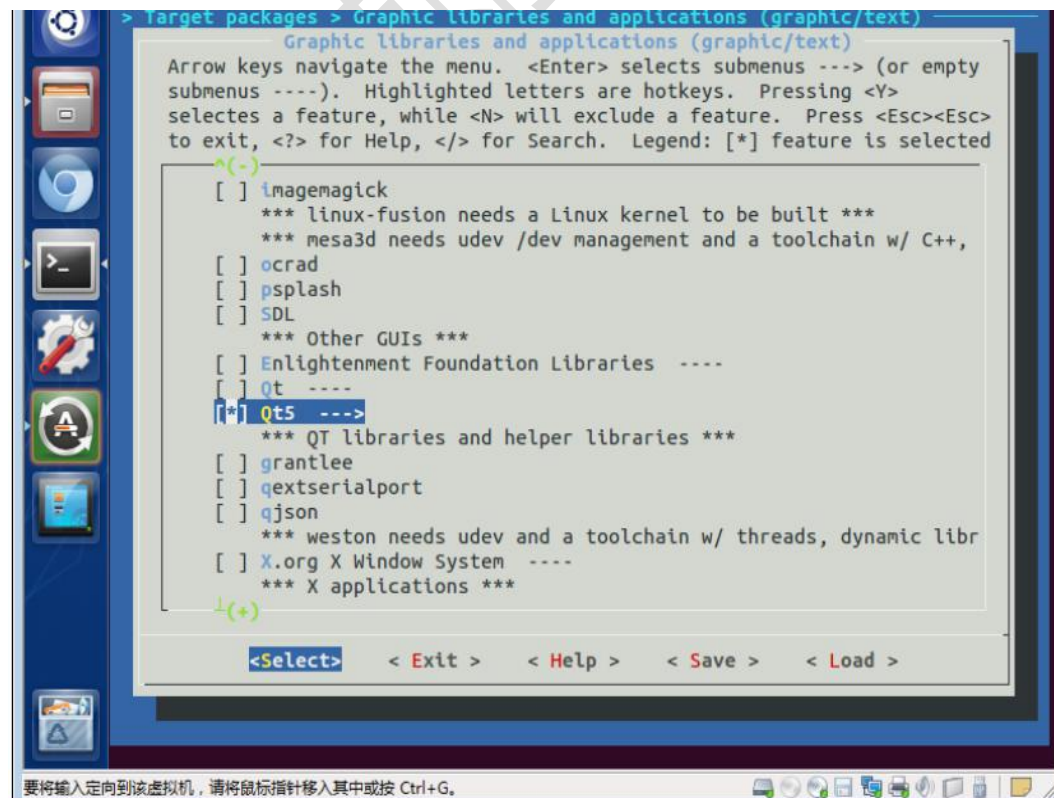
退回上一级，进入 Toolchain 目录，按下图配置：



退回上一级，进入 System configuration 目录，作如下配置：



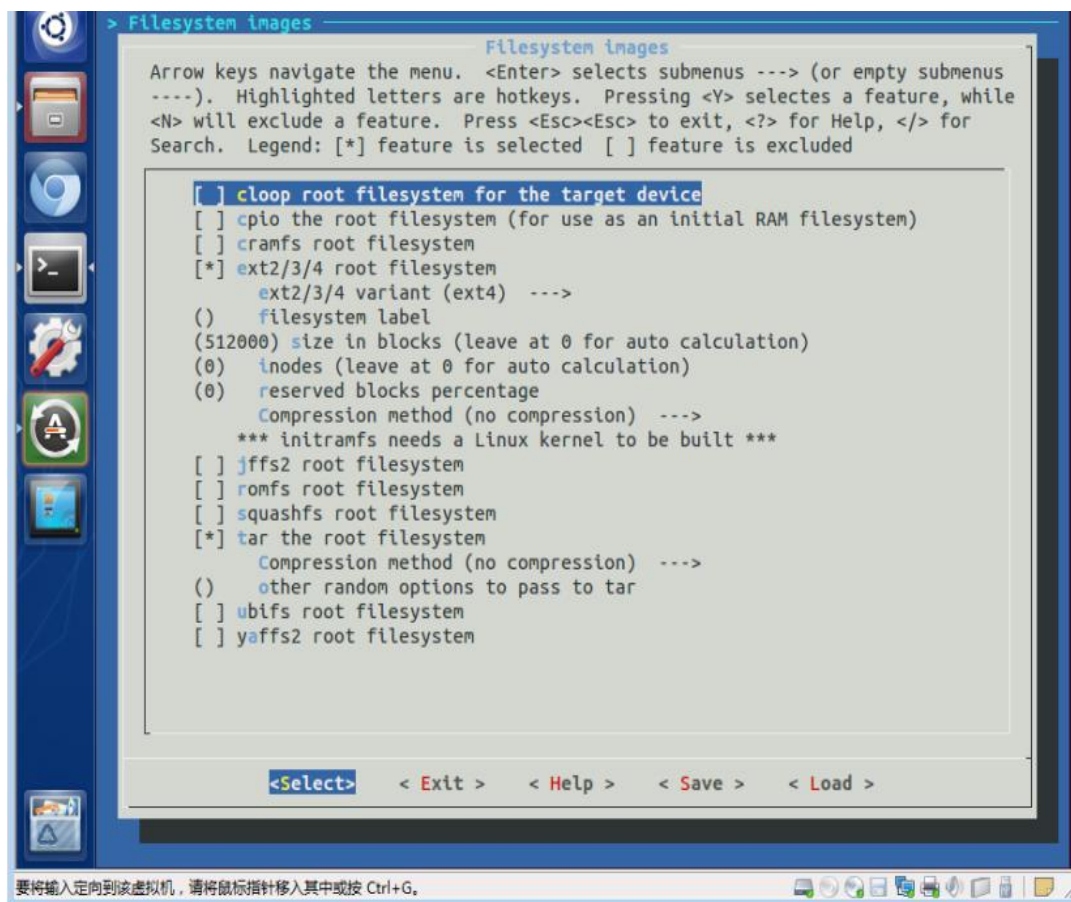
退回上一级，进入 Target packages 目录，再进入 Graphic libraries and applications (graphic/text)目录，选择 Qt5，如下图所示：







退回 menuconfig 的开始界面，进入 Filesystem images 菜单，作如下配置：



到此， buildroot 配置完成。默认配置保存在 buildroot 根目录的.config 中，我们可以备份该配置文件，以防后续配置出错。执行如下指令备份配置文件：

cp .config g4418_config

cp .config g4418_config

1.3 编译 buildroot

配置完成后，执行 make 指令即可编译 buildroot 了。编译 buildroot 会会依赖一些第三方插件和库，在 QT5.4 移植章节的编译文件系统小节中，已经给出了需要安装的包，在编译之前需要提前安装，否则会报错。编译完成后，文件系统映像 rootfs.ext2 会生成到



output/images 目录。

1.4 测试 QT5.4 默认示例

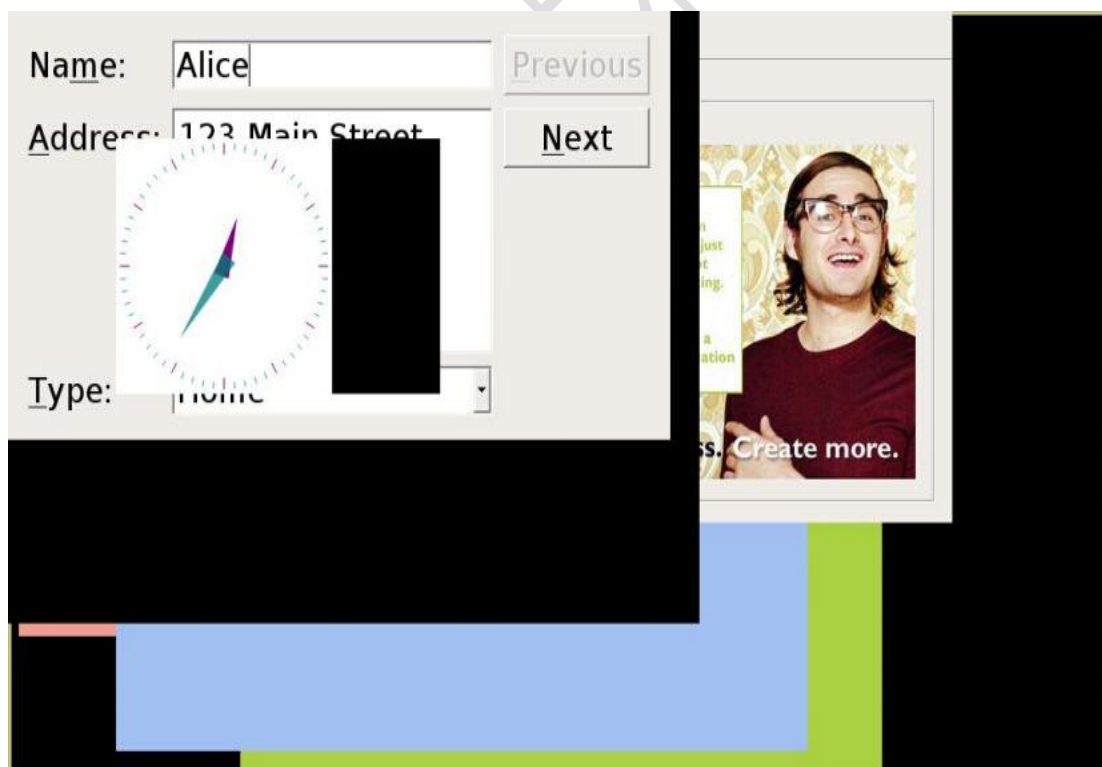
将 uboot，内核，文件系统烧写进开发板，进入 linux 文件系统后，可以进入

/usr/lib/qt/examples 目录测试 QT 示例。

进入 gui/analogclock 目录，执行 analogclock 文件，指令如下：

```
./analogclock &
```

这时，在开发板上可以看到有一个时钟图案被绘制出来，如下图所示：

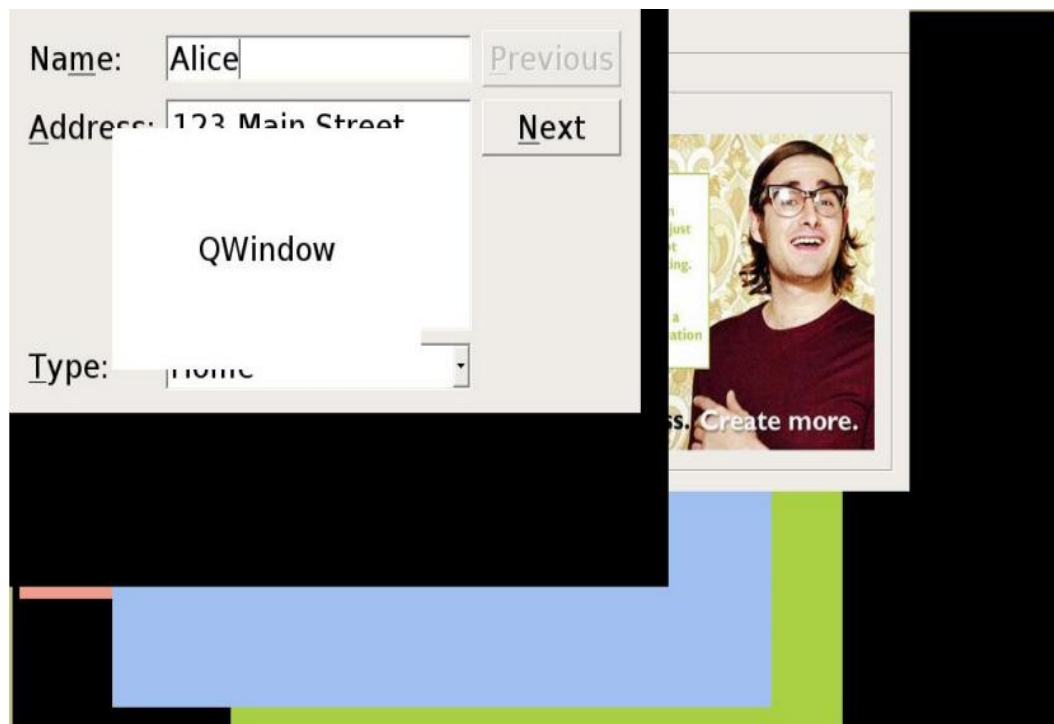


进入 gui/rasterwindow 目录，执行 rasterwindow 文件，指令如下：



`./ rasterwindow &`

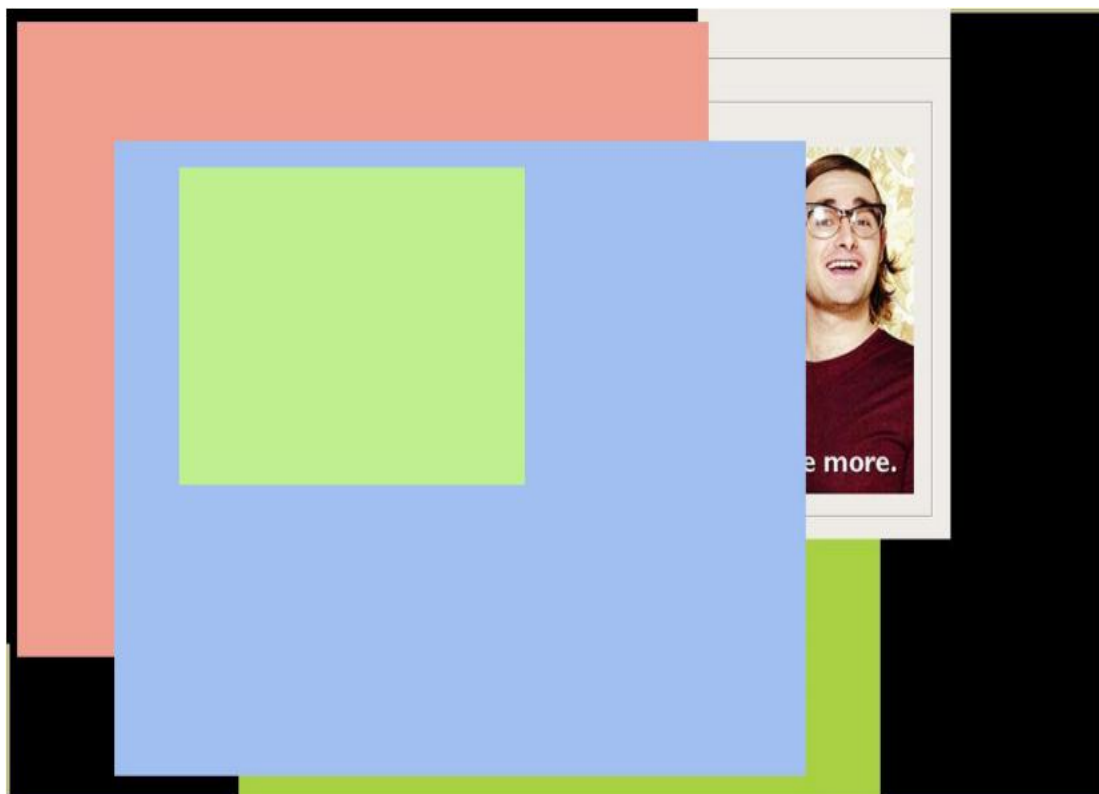
这时，在开发板上可以看到一个标注有 QWindow 的图案被绘制出来，如下图所示：



进入 `qpa/windows` 目录，执行 `windows` 文件，指令如下：

`./windows &`

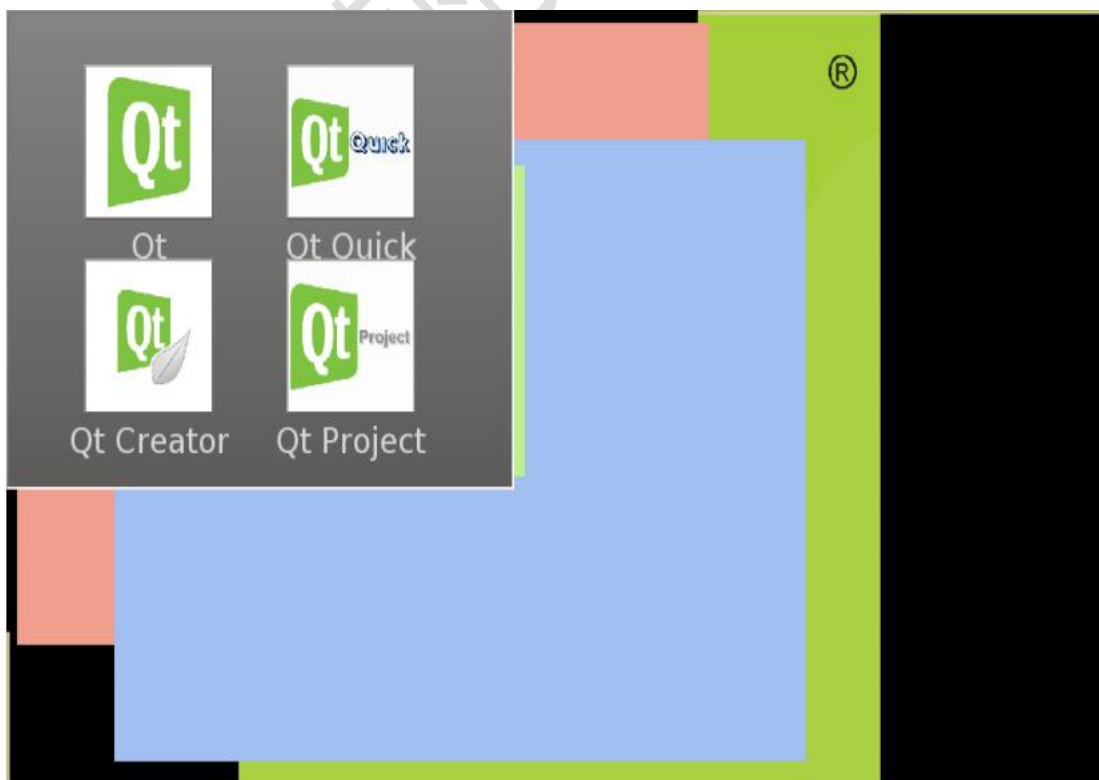
这时，在开发板上可以看到有三幅图案被绘制出来，如下图所示：



进入 sql/drilldown 目录，执行如下指令：

```
./drilldown &
```

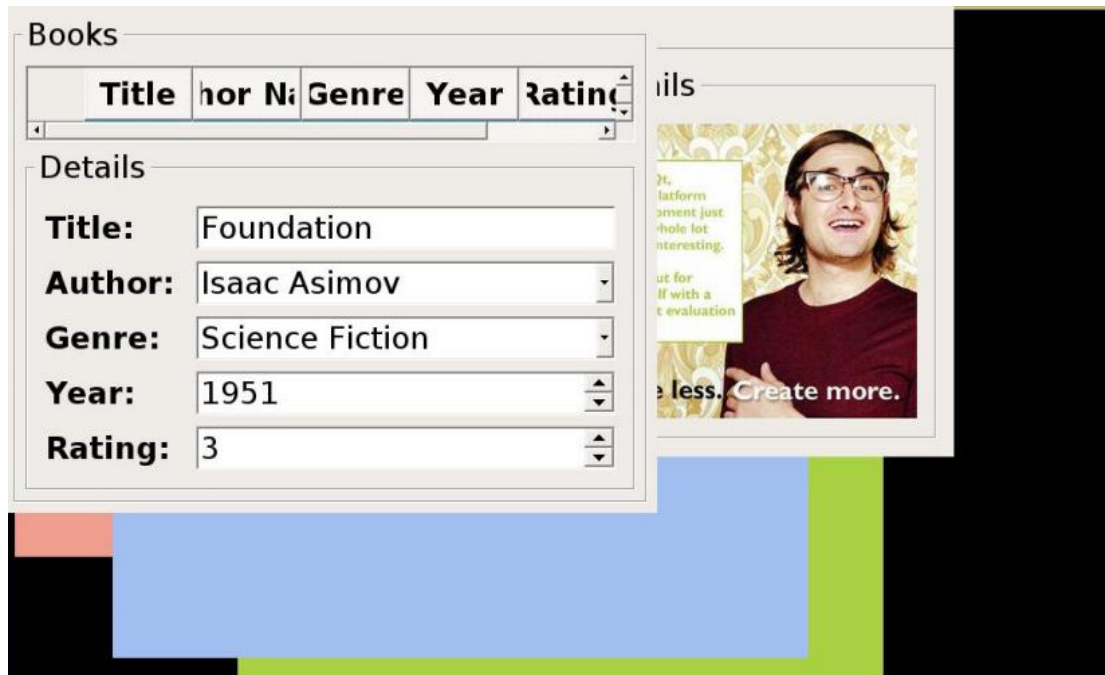
这时，在开发板上可以看到有四个 QT 画面被绘制出来，如下图所示：



进入 sql/books 目录，执行如下指令：

```
./books &
```

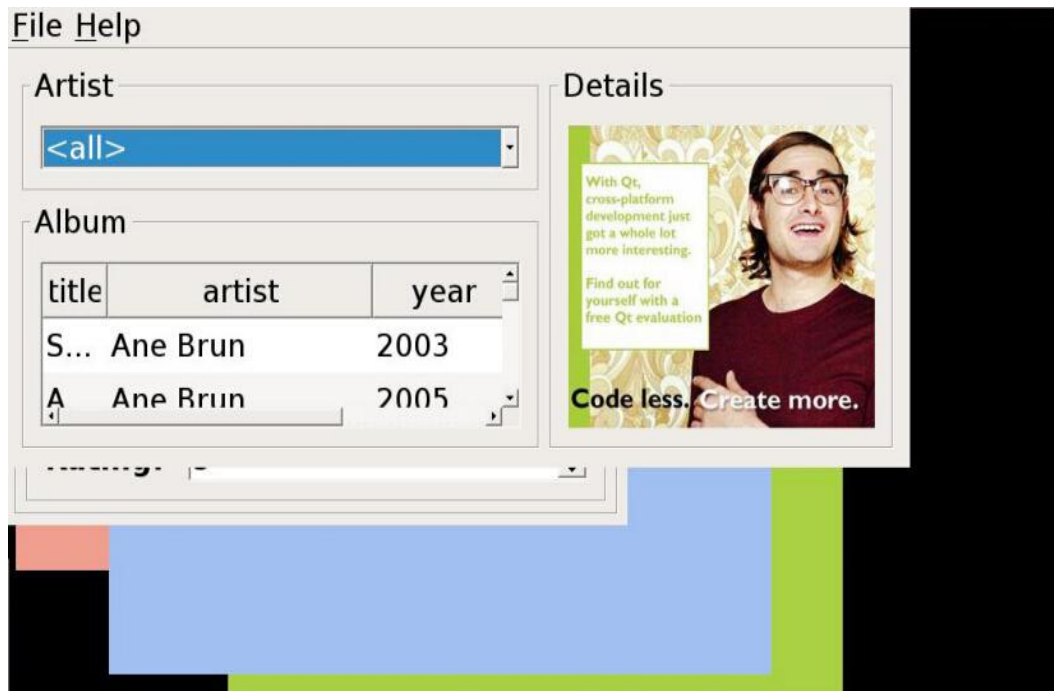
这时，在开发板上可以看到有一个对话框被绘制出来，如下图所示：



进入 sql/masterdetail 目录，执行如下指令：

```
./masterdetail &
```

这时，在开发板上可以看到有一个对话框绘制出来，如下图所示：



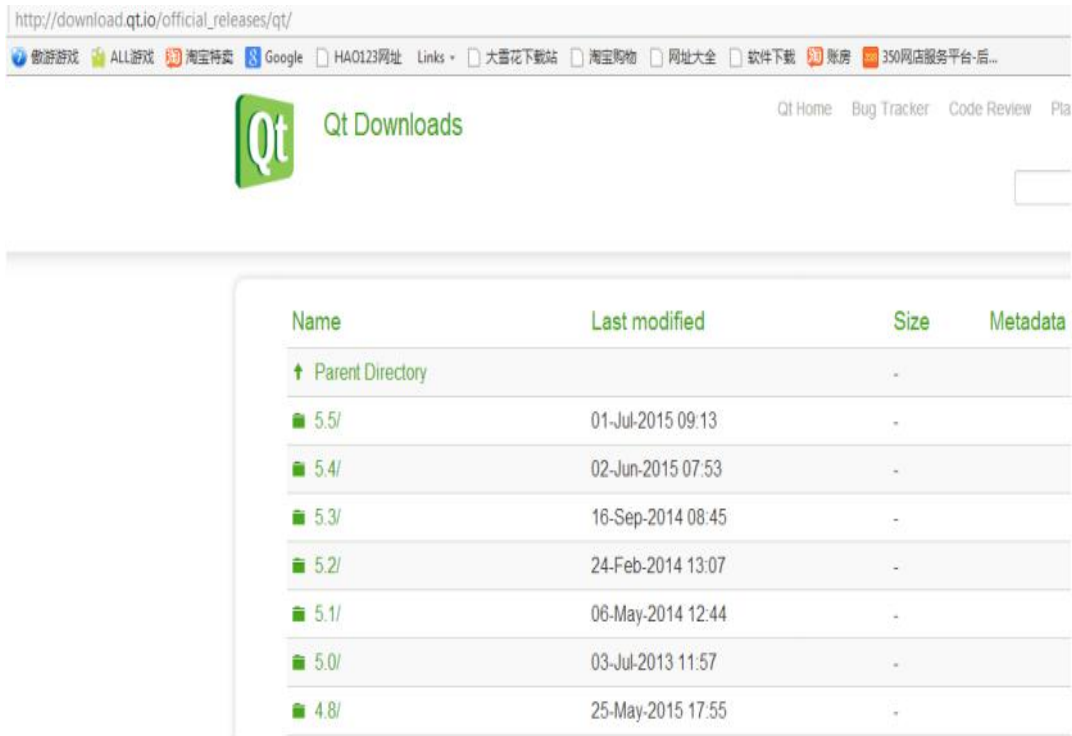
默认 buildroot 编译出了很多示例，这里不带——列举，用户可以自行尝试。

1.5 安装 QT Creator

通常我们使用 QT Creator 创建基于 QT 的工程。在 QT 官网下载最新的 QT 安装包，下载地址如下：

http://download.qt.io/official_releases/qt/

打开链接页面如下：



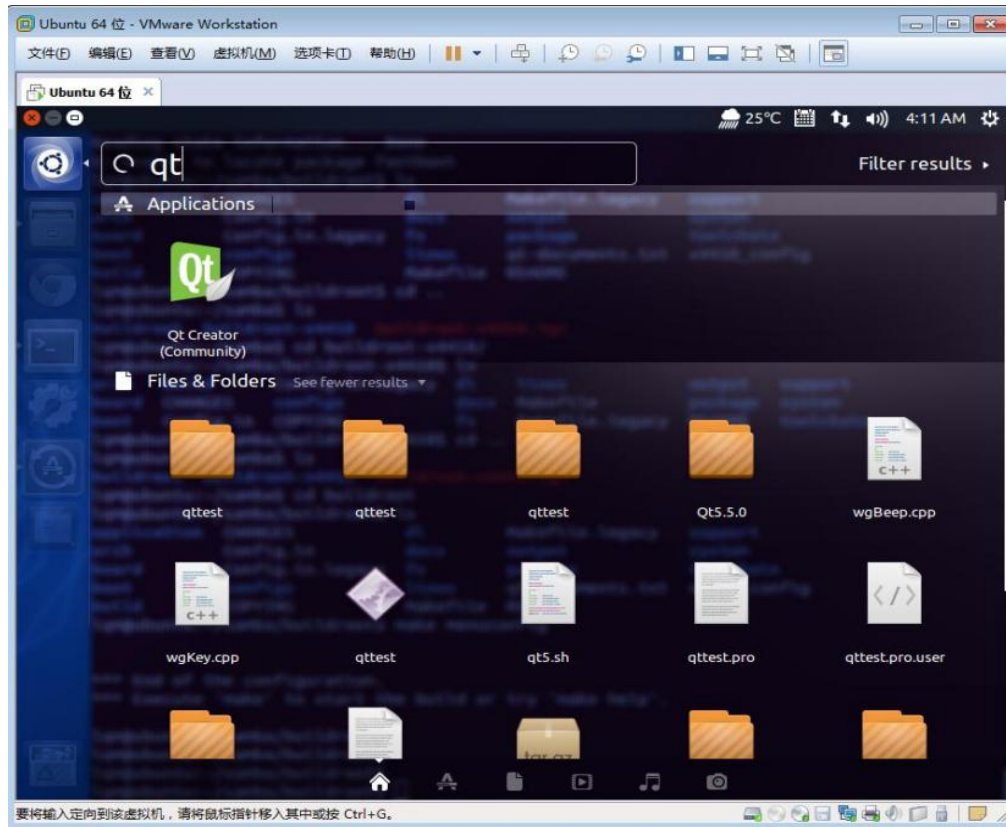
点击 5.5，下载最新的安装包，得到名为 qt-opensource-linux-x64-5.5.0-2.run 的文件，将它拷贝到 ubuntu 的用户目录，使用如下指令安装：

```
./qt-opensource-linux-x64-5.5.0-2.run
```

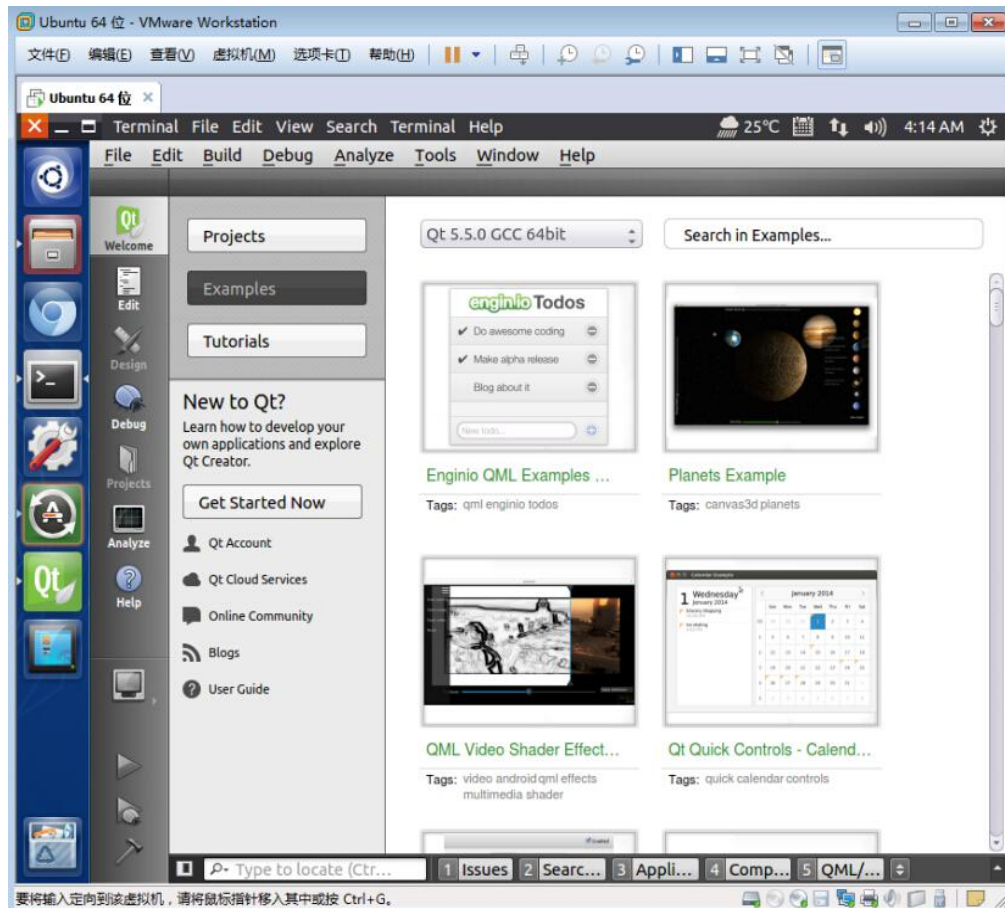
安装完成后，QT Creator 也就安装完成了。

1.6 编译 QT Creator 默认示例

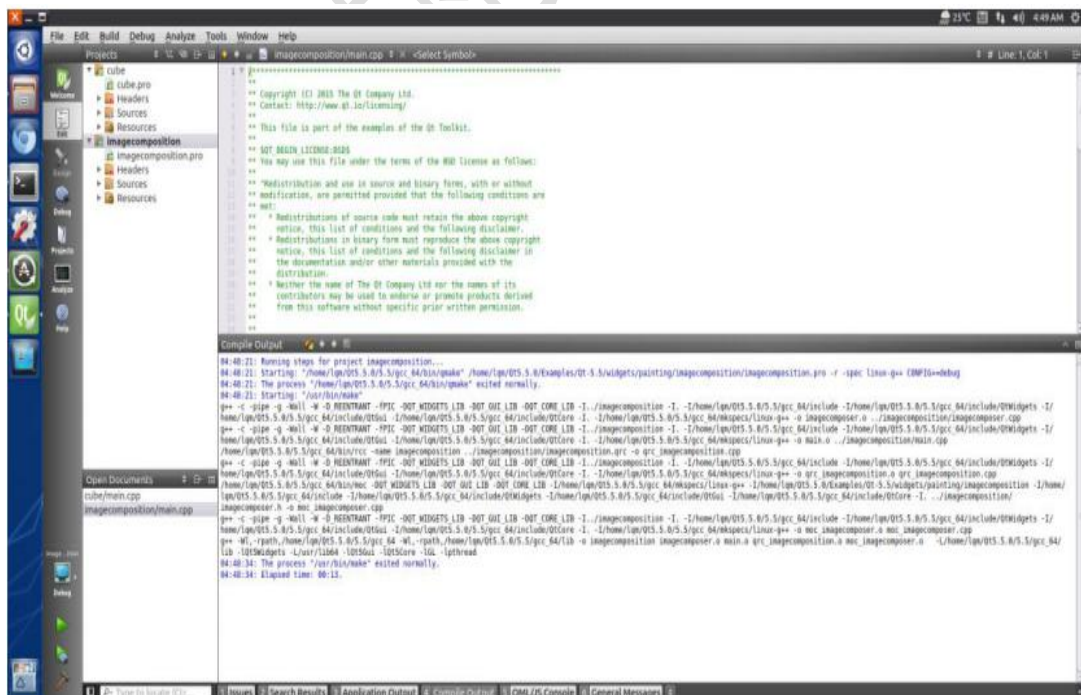
点击 ubuntu 图标，输入 qt，将会查找含有 qt 的文件，同时，安装好的 QT Creator 也会被列出来，如下图所示：



点击 QT 图标，QT Creator 将会运行，如下图所示：

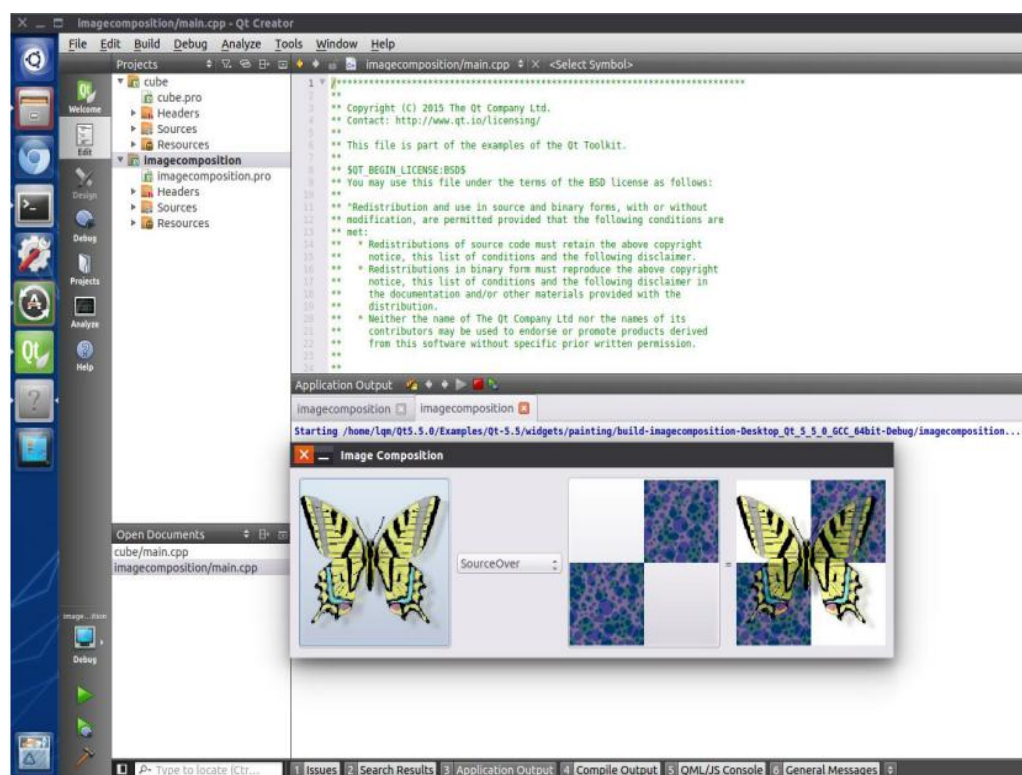


默认 QT 的配置是针对 X86 架构的，这时编译出来的示例只能在 PC 机上运行。我们打开 Image Composition Example 示例，它是一个图片叠加显示的示例，找到该示例，单击即可。打开后的界面如下：



编译完成后，编译出来的映像会自动运行，一个叠加的蝴蝶图像界面显示出来了， 界面

如下：



下面我们将该示例编译到开发板上运行。 使用 QT Creator 打开上面的示例工程

选择 Projects 一栏，可以看到最顶端的框图 1，显示 Desktop Qt 5.5.0 GCC 64bit，表明它通过 64 位的 GCC 编译，框图 2 表明在 debug 模式下编译，相对 release 模式， debug 模式

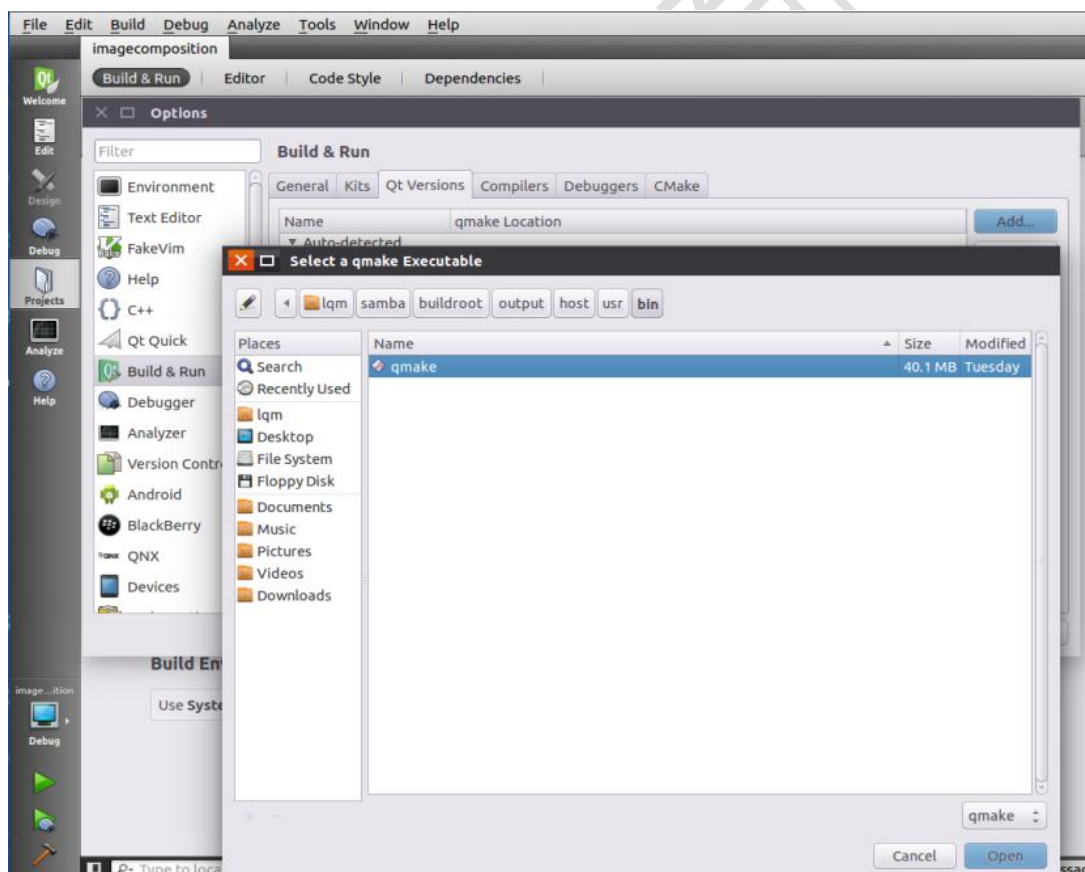
下含有大量调试信息，编译出来的映像会比较大。通常发布映像时，我们选择 release 模式。

框图 3 指定了编译的路径。框图 4 指定了 qmake 和交叉编译工具。框图 5 为调试按键，第一个用于选择编译模式为 debug 或 release，第二个为运行按钮，第三个为单步调试按钮，第四个为编译按钮。 框图 6 为一些输出信息，如 Compile Output，会给出整个编译的信息。

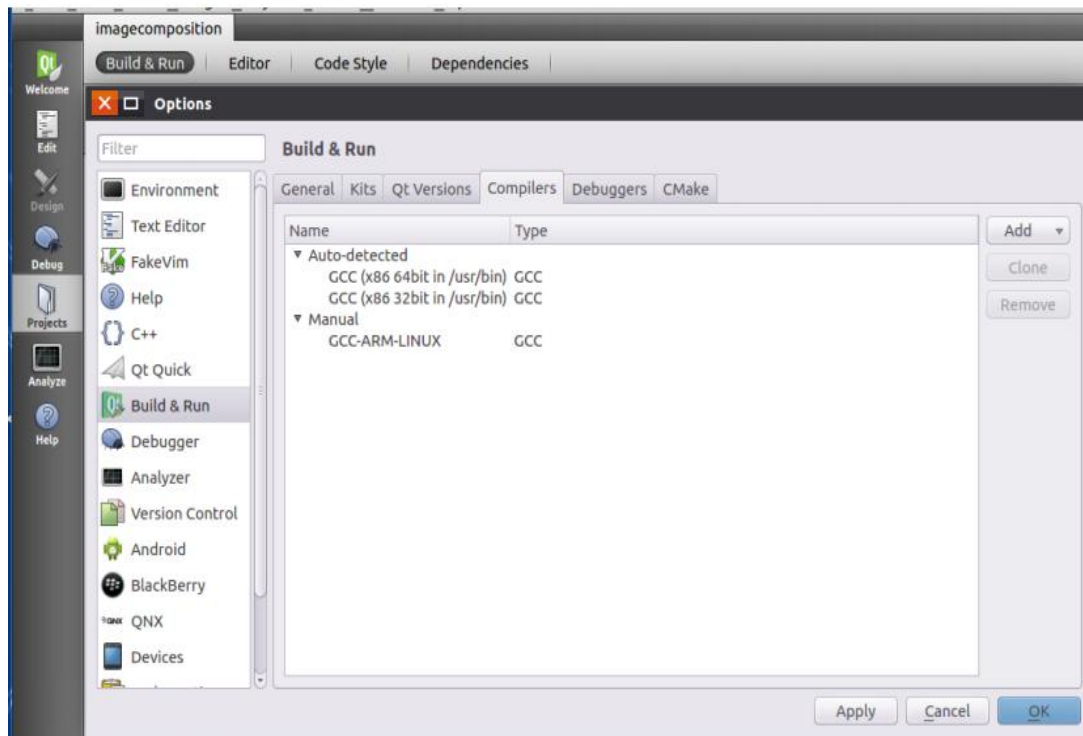
在框图 1 中， 点击 Tools->Options，如下图：

在左侧对话框中选择 Build & Run，在后边选择 Qt Versions，默认 Manual 为空，Auto-detected 选择的 QT5.5 默认的 qmake，它将运行在 X86 平台的 linux 系统上，因此我们要手动添加在 ARM 平台上运行的 qmake。

在 buildroot 编译文件系统时，我们选中 QT5 后，将会在 buildroot 的 output/host/usr/bin 目录生成支持 ARM 平台的 qmake，点击 Add，指向该路径的 qmake：

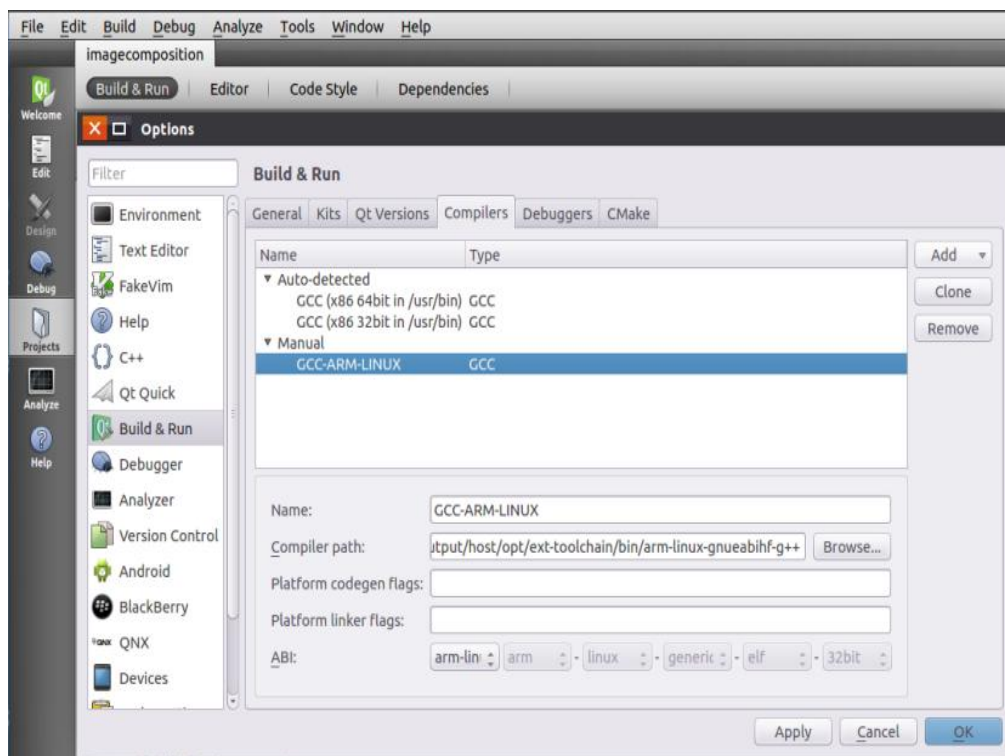


点击 Open，即在 Manual 中添加了支持 ARM 平台的 qmake。再选择 Compilers 一栏，如下图所示：



默认 Manual 为空，Auto-detected 为支持 X86 32 位和 64 位的 GCC，这将直接导致编译

出来只能在 PC 机上运行。点击 Add，选择 GCC，如下图所示：



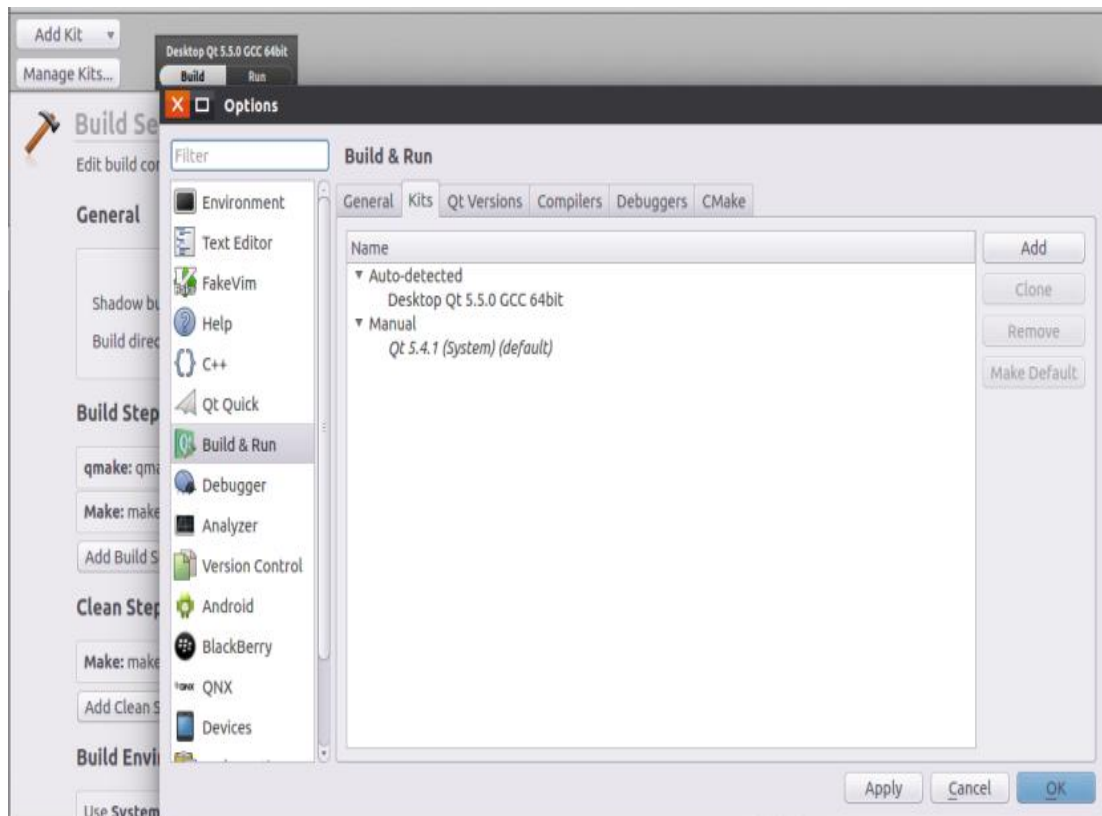
在 Name 中重命名，以区别 ARM 和 PC 平台，如我们命名为 GCC-ARM-LINUX，它将会直接显示在前面 QT 工程界面的框图 1 中，到时我们编译 QT 工程时，能够一目了然，编译出来的到底是 PC 平台还是 ARM 平台。在 Compiler path 中指定交叉编译工具，在 buildroot

中默认已经自动下载并安装了交叉编译工具，我们指定到如下路径即可：

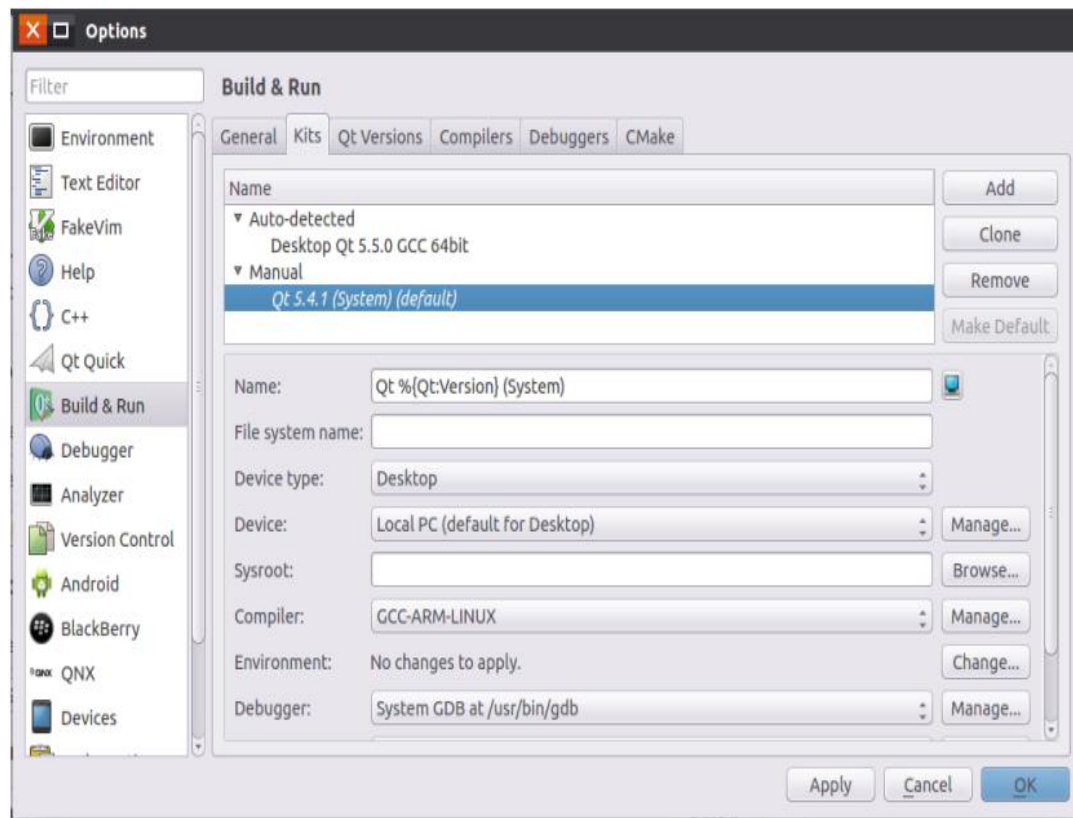
buildroot/output/host/opt/ext-toolchain/bin/arm-linux-gnueabi-g++

设置完成后，点击 OK，完成设置。

再回到 QT 的工程界面，我们发现框图 1 中的配置仍然针对 PC 机，点击框图 1 中的 Manage Kits，如下图所示：

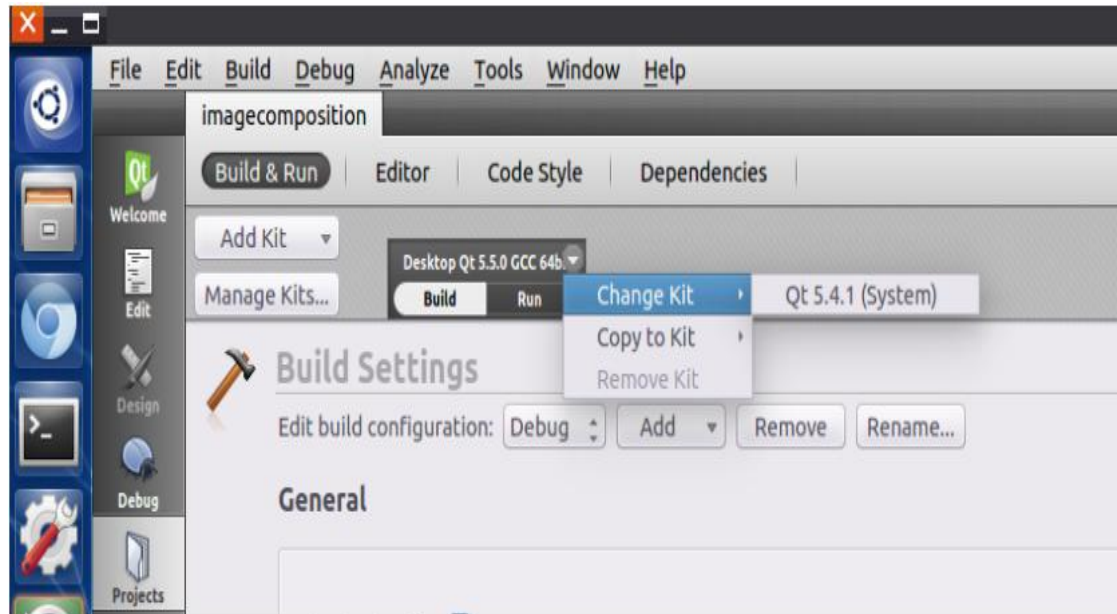


注意，这时默认 Manual 仍然为空，点击 Add，会弹出一个有很多选项的对话框，我们按如下方式配置：

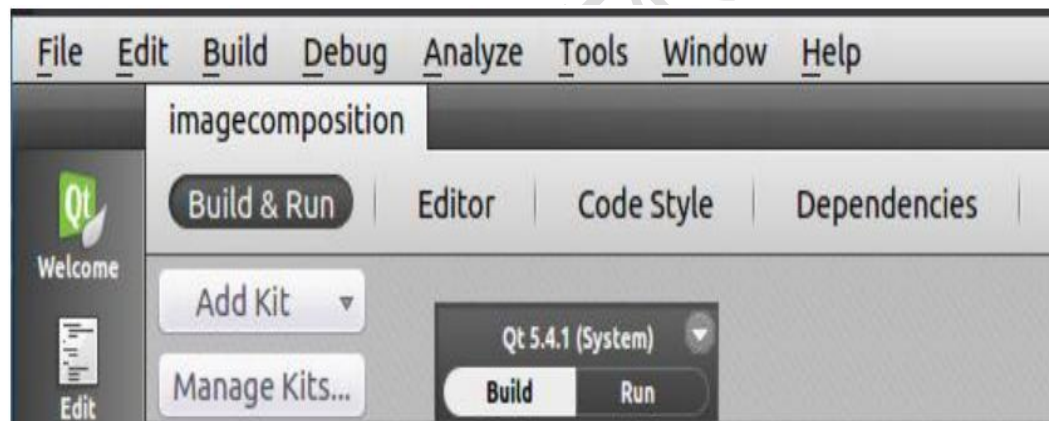


这里的 Name，我们可以点击 Qt Versions，选择 Manual 中的 qmake，下面就会有名称出来，将它拷贝过来即可

在框图一中，点击设置编译环境的下拉箭头，如下图所示：



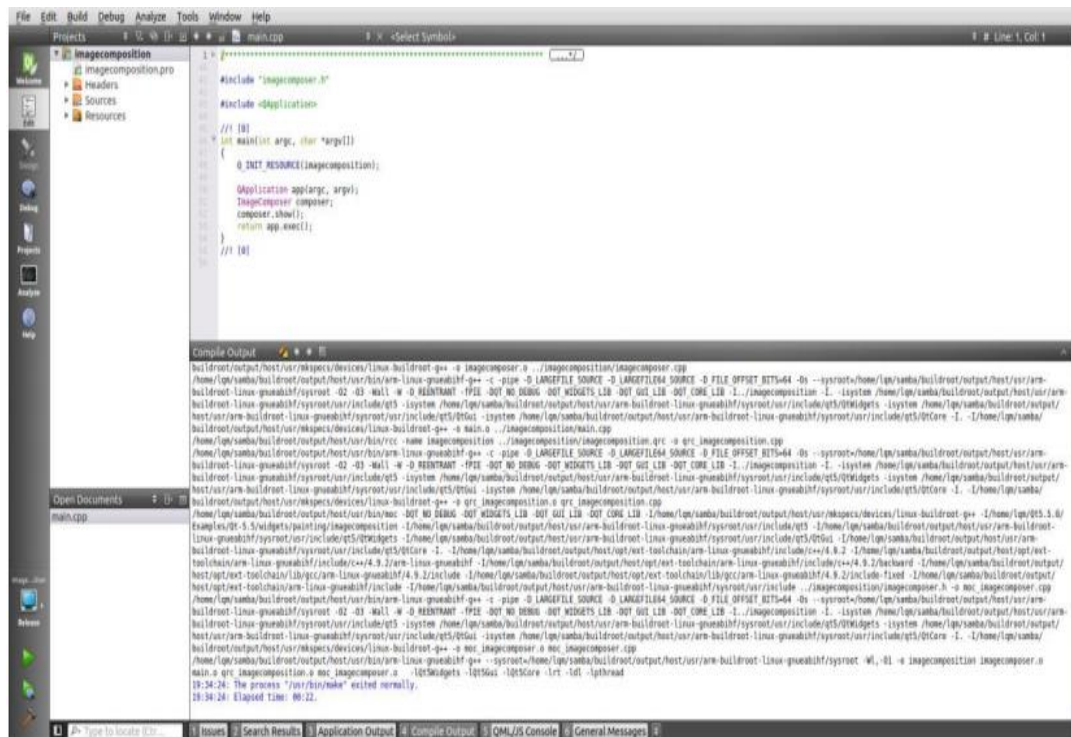
选择 Change Kit->Qt5.4.1(System) ， 更改后的界面如下：



在框图 2 中选择 release

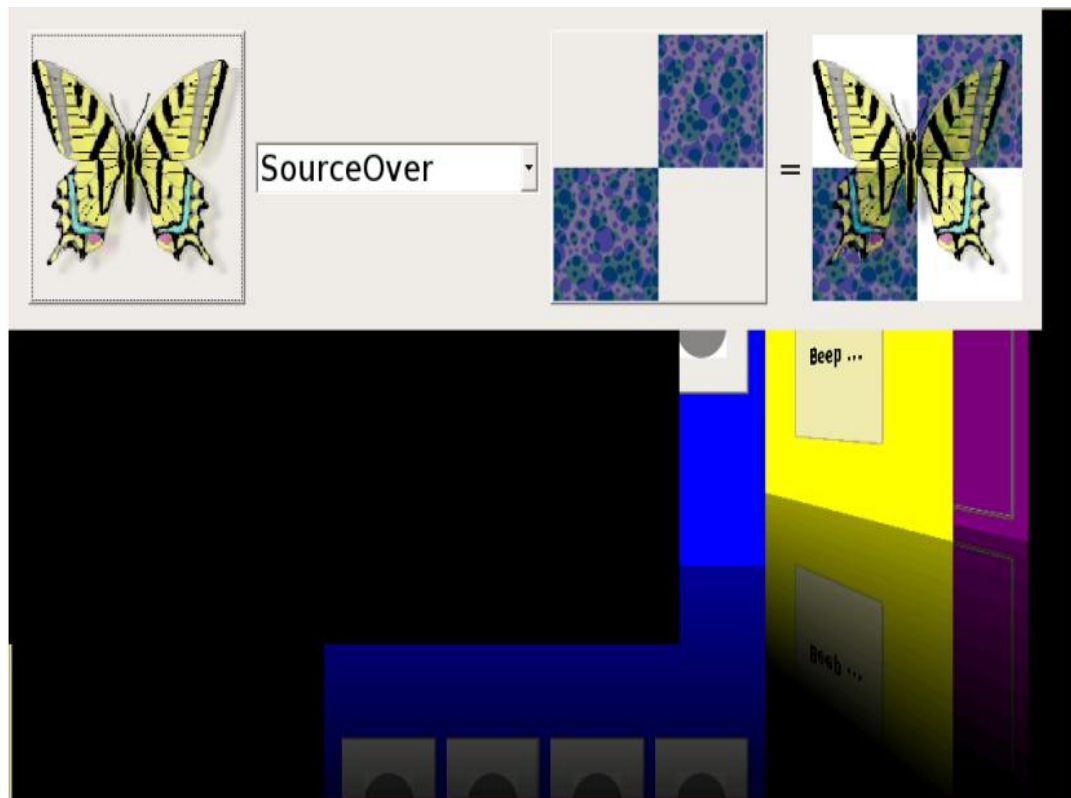
到此，配置完毕，点击框图 5 中的编译按钮，在 Compile Output 中可以看到编译信息如

下：



这时，在框图 2 中指定的目录中已经生成了能够在 ARM 平台运行的映像了；

将该文件拷贝到 x4418 开发板上运行，可以看到美丽的蝴蝶图案显示出来了。





第 2 章 产品线介绍

2.1 核心板系列

G4418 (主控为三星 4418)

G6818 (主控为三星 6818)

G210 (主控为三星 210)

M9 核心板 (主控为高通 8916)

2.2 开发板系列

G4418 开发板 (主控为三星 4418)

G6818 开发板 (主控为三星 6818)

G210 开发板 (主控为三星 210)

M9 开发板 (主控为高通 8916)

2.3 卡片电脑系列

G4418 卡片电脑 (主控为三星 4418)

G6818 卡片电脑 (主控为三星 6818)

G3188 卡片电脑 (主控为瑞芯微 3188)

说明：产品详细规格，以及更多其他产品请关注葡萄雨技术官方网站或与我们联系。