

Закодируй Greenфон

Для разработки вашего первого мобильного приложения не надо ни быть крутым хакером, ни даже иметь сам Greenphone, пишет Грэм Моррисон.



Тrolltech Greenphone – открытая платформа, и инструменты для разработки под нее доступны по лицензии GPL. А значит, еще никогда не было так просто создать собственное приложение для мобильного телефона – до того просто, что нам хватит трех страниц, чтобы показать это. Наша программа будет всего-навсего пересчитывать сантиметры в дюймы, но ее так же легко можно употребить как конвертер валют или расширить любыми способами. Знакомство с C++ вам поможет, но если вы таковым не располагаете – вот отличный шанс научиться.

Greenphone использует особую версию Trolltech Qt API, у которого много общего с Qt API, используемом для создания KDE-приложений. Community SDK от сообщества Greenphone (www.qtopia.net/modules/mydownloads/singlefile.php?lid=22), необходимый для разработки GPL-приложений для Greenphone, представляет собой образ CD. Он построен на основе VMware для использования вместе со свободно распространяемым VMware Player. Это предустановленный дистрибутив Linux, включающий набор инструментов разработчика GNU, а также собственные библиотеки Trolltech, заголовочные файлы и приложения с графическим пользовательским интерфейсом. Так как это образ VMware, то вам не нужен Linux – VMware Player также доступен и под Windows. Но, что более важно, вам не нужен даже Greenphone: Trolltech предоставляет собственный «Эмулятор Greenphone», и вы можете играть с виртуальным телефоном, не боясь повредить оборудование.

Создание среды разработки

Первый шаг – установка Community SDK со скачанного образа CD. После этого у вас появится каталог с файлами VMware, которые необходимо загрузить в виртуальную машину. После ее запуска вы увидите зеленые тона экрана рабочего стола Trolltech KDE. На нем разбросано несколько иконок, и две из них используются для эмуляции среды Greenphone. Нажмите на иконке Runqvf, затем на иконке Runqre.



› Комплект разработчика Greenphone использует образ VMware. И бесплатный VMware Player..

Первая иконка запустит виртуальный фрейм-буфер для Greenphone, встроенный в графическое представление телефона. Вторая иконка запустит ОС Qtopia внутри эмулируемой среды [строго говоря, Qtopia не является самостоятельной ОС – она базируется на Linux, – прим. ред.]. В результате вы увидите виртуальное представление Greenphone на рабочем столе, которое ведет себя как настоящее, когда вы нажимаете на кнопки. (Любители повозиться с оборудованием могут подробнее узнать о Greenphone на сайте www.trolltech.com/products/Qtopia/greenphone или в обзоре LXF91.)

Виртуальная машина включает каркас для простого приложения

Greenphone. Вы можете обнаружить его, нажав на иконку Konsole для открытия командной строки и набрав `cd projects/application` для перехода в каталог, содержащий исходный код. Просмотрев содержимое каталога, вы увидите исходный код (`example.h` и `example.cpp`), пользовательский интерфейс (`examplebase.h`) и документацию в каталоге **Help**. Только об этих файлах и надо позаботиться при создании собственного приложения.

В командной строке наберите `source /opt/Qtopia/SDK/scripts/develop86.sh`, чтобы установить переменные окружения для нормальной компиляции примера. Затем потребуется команда `qttopiamake`, она подготовит исходные файлы примера для механизма сборки Qtopia. После этого пробежитесь по знакомым командам `make clean`, `make` и `make install`, чтобы собрать и установить пример проекта. После этого необходимо попросить Greenphone обновить свой список приложений, так что наберите `gph -rescan`.

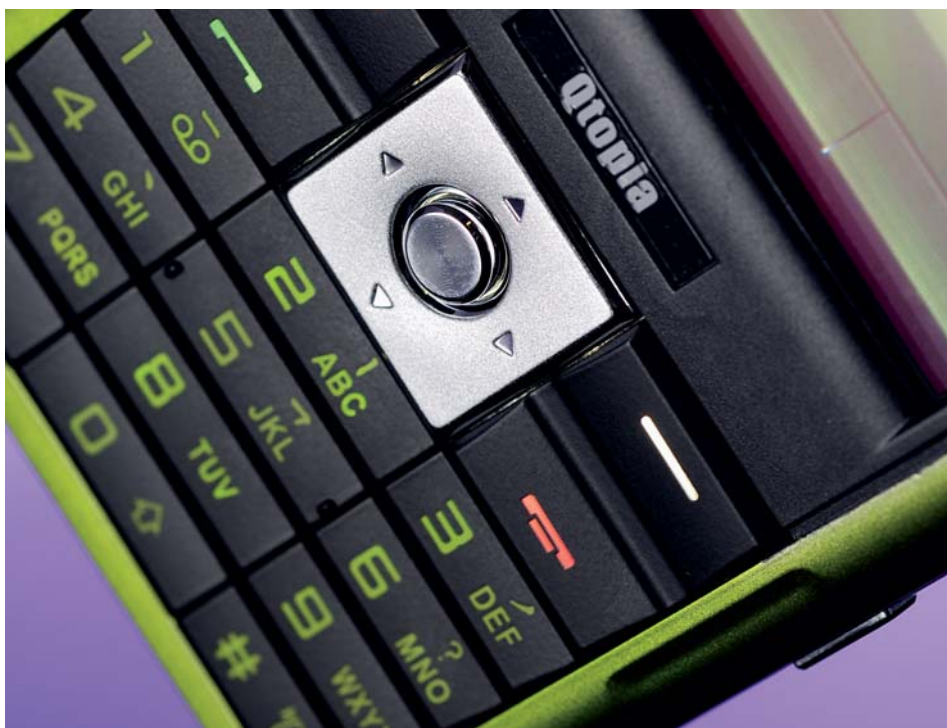
Теперь вы готовы запустить полученное приложение, как с помощью виртуального Greenphone, так и настоящего оборудования, если вы его подключили. Просто перейдите в папку **Приложения (Applications)** в телефоне и выберите иконку **Example** (она выглядит как красный знак вопроса на белом фоне). Запустится простое приложение, которое Trolltech включило в состав SDK – и вы сразу же обнаружите, что оно не делает ничего. Абсолютно ничего. Все, что можно сделать, это нажать на **Выход (Quit)**. Наша задача – заставить программу выполнять полезную работу.

Проектируем пользовательский интерфейс

Прежде чем добавлять какую-либо функциональность, необходимо спроектировать пользовательский интерфейс. Как и для других KDE-и Qt-приложений, для этого есть инструмент *Designer*, и вы должны видеть его иконку на рабочем столе – она похожа на пару штангенциркулей. При первом запуске программа спросит, какой вид формы вы хотите спроектировать. Так как мы собираемся загрузить пользовательский интерфейс из каталога примеров Greenphone, то нажмем **Заккрыть (Close)**.

Если вы никогда не видели *Designer*, то, возможно, его интерфейс вас слегка отпугнет. Панели разбросаны по всему экрану – он выглядит как *Gimp* на стероидах. Каждая панель заполнена опциями: небольшими окошками и выпадающими меню. Вам надо отыскать опцию **Open Form (Открыть форму)** в меню **File (Файл)**. Далее выберите **Projects (Проекты) > Application (Приложение)**, и откройте затаившийся там файл `examplebase.ui`. Теперь вы должны узреть интерфейс нашего пустого приложения для Greenphone.

Сейчас нам необходимо создать интерфейс пользователя для программы-конвертера. Нажмите на текст внутри окна, затем щелкните



правой кнопкой и выберите **Break Layout (Разбить компоновку)** из меню **Form (Форма)**. Таким образом мы разблокируем компоненты внутри окна и сможем передвигать их и добавлять новые. Нам понадобятся одно поле ввода, которое будет принимать входную величину, и два выпадающих меню для выбора входного и выходного формата. Также потребуется поле вывода для отображения конечного результата и кнопка для осуществления преобразования.

Чтобы добавить эти компоненты, перетащите их из панели компонентов, расположенной слева на экране. Из раздела **Display Widgets** нам понадобятся четыре метки и один **LCD Number** для вывода результата. Из раздела **Input Widgets** выберите два **Combo Box** и один компонент **Line Edit** для ввода значения. Наконец, нам нужна кнопка **Push Button** из списка **Buttons**.

Выделите исходный текст примера и нажмем **Delete**, чтобы удалить его с формы – мы оставим только кнопку **Quit**. Можете расположить компоненты по своему вкусу. Текст на трех метках изменим на **Amount**, **From** и **To** соответственно (щелкните правой кнопкой мыши и выберите **Change text (Сменить текст)**). Если вы хотите выровнять текст по центру, то выберите свой компонент и свойство **alignment** из **Property Editor (Редактор Свойств)** и измените горизонтальное выравнивание на **AlignHCenter**.

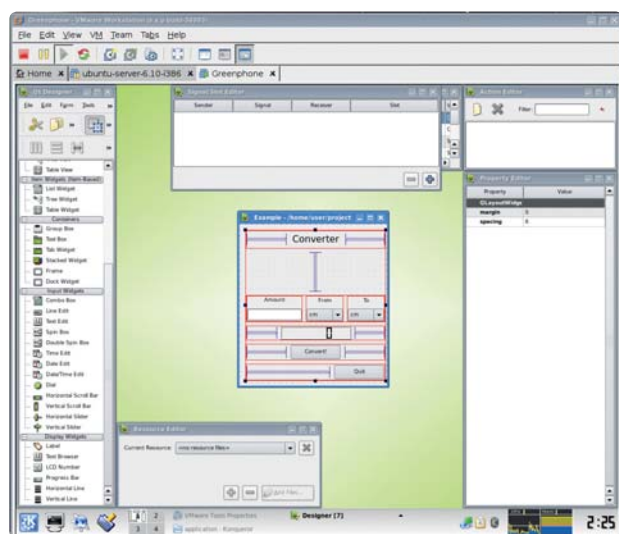
Последнюю метку можно использовать как заголовок для приложения. Размер шрифта для этого компонента может быть выбран из списка шрифтов в редакторе свойств. Вам также потребуется изменить текст на **Push Button**, и добавить пункты преобразования в **Combo Box**'ы (щелкните правой кнопкой и выберите **Edit items (Редактировать элементы)**). Для начала добавьте **inches** и **cm**.

Добавив и отредактировав все необходимые компоненты, пробежитесь по руководству расположения компонентов на следующей странице, затем сохраните результаты, переписав оригинальные файлы. Если у вас не хватает прав это сделать, то откройте *Konsole* и наберите `sudo chmod -R 777 projects`.

Пишем код

Пользовательский интерфейс спроектирован; время добавлять функции, чтобы конвертор обрел законченную форму. Мы можем это сделать, добавив одну функцию в наш пример. Откройте файл `example.h` и добавьте `'void convert()'` сразу под методом `'void goodbye()'` в разделе `'private slots'`. Именно эта функция будет выполняться, когда вы нажмете на кнопку **Convert (Преобразовать)** в приложении.

» Чтобы ввести числа в Qtopia, полсекунды подержите клавишу нажатой.



» Экспериментируя с пользовательским интерфейсом в программе *Trolltech Designer*, найдите наилучший вид для вашего приложения.





» Наша законченная программа на телефоне Greenphone!

» Далее, откройте файл **example.cpp** и добавьте следующий код сразу за строкой, начинающейся с 'connect':

```
connect(pushButton,SIGNAL(clicked()),
        this,SLOT(convert()));
```

Это знаменитый механизм сигналов и слотов от Trolltech, и мы здесь сообщаем приложению, чтобы при нажатии на кнопку **pushButton** выполнялась функция **convert**: **clicked** – это сигнал, а **convert** – это слот.

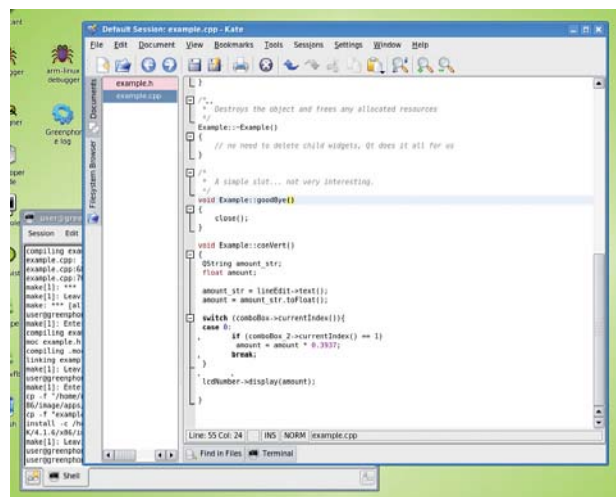
Необходимо добавить код для слота, который будет выполнять преобразование, а для этого нужно подключить заголовочные файлы для каждого используемого компонента. Добавьте в начало файла следующий код:

```
#include <qstring.h>
#include <qlineedit.h>
#include <qlcdnumber.h>
#include <qcombobox.h>
```

Наконец, добавим функцию преобразования – припишем следующий код в конец файла **example.cpp**:

```
void Example::convert()
{
    QString amount_str = lineEdit->text();
    float amount = amount_str.toFloat();
    switch (comboBox->currentIndex())
    {
        case 0:
            if (comboBox_2->currentIndex() == 1)
                amount = amount * 0.3937;
            break;
    }
    lcdNumber->display(amount);
}
```

Наберите **make**, **make install** и **gph -rescan**, чтобы установить вашу новую программу, и запустите ее из списка приложений (она все еще называется **Example**). Вкратце, мы делаем вот что: считываем значение



» В текстовом редакторе **Kate** удобно редактировать исходный код: подсветка синтаксиса облегчает работу.

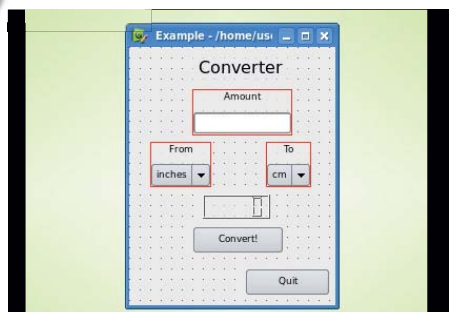
из поля ввода и преобразуем его в число, а потом проверяем значения в обоих выпадающих списках. Мы предполагаем, что первый вход будет **cm**, а второй – **inches**. Если **cm** выбрано в качестве первой записи, то выражение **if** проверит значение во втором Combo box. Если это '1', то это должно быть, дюймы, и мы производим преобразование величины, полученной из поля ввода. Наконец, мы посылаем это значение на LCD-дисплей приложения.

Очевидно, надо еще немало потрудиться, прежде чем программа станет действительно полезной, но она функционирует. И к тому же занимает меньше десяти строк! Вот краткий список возможных нововведений. Добавьте другие возможности конвертации – дюймы в сантиметры, например, или метры в ярды. Потом проверьте корректность ввода и отобразите ошибку, если введенное значение не есть число. Вы даже можете добавить возможность сохранять наиболее популярные преобразования.

Если вы закончите этот проект, мы будем рады увидеть результаты. Пошлите нам ваш исходный код (letters@linuxformat.ru), и мы положим его на наш DVD. LXF

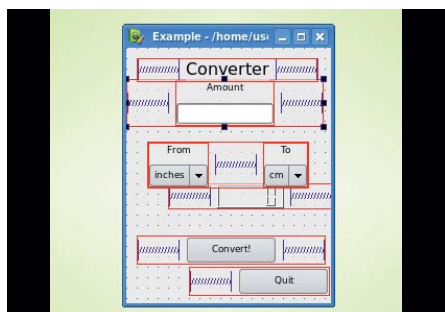


Шаг за шагом: Располагаем компоненты в Designer



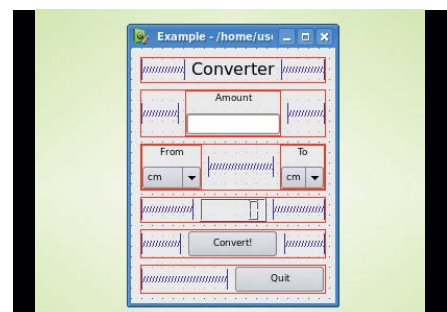
1 Группируем компоненты

Выбрав все компоненты для вашего приложения в **Designer**, расположите их, как вам нравится. Далее нажмите **Shift** и выберите каждый компонент с меткой, потом выберите **Layout Vertically** из меню **Form**. В результате создастся группа компонентов с менеджером компоновки, который будет видеть их как один компонент.



2 Выравниваем

Designer использует набор взвешенных 'разделителей' для форматирования раскладки. Перетащите горизонтальный разделитель из палитры инструментов в пространство между каждым компонентом, располагающимся в каждой строке. Далее выберите все разделители и компоненты в каждой строке и выберите **Layout Horizontally** (Расположить горизонтально) из меню **Form**.



3 Делаем сетку

Собрав все компоненты в группы и расположив их горизонтально по строкам, выберите **Layout In A Grid** из меню **Form**. Так вы добавите ваши группы компонентов в матрицу, которая будет масштабироваться в зависимости от размера и разрешения используемого экрана. Если вы захотите что-то поменять, отмените компоновку – выберите **Break Layout**.