

doi: 10.7690/bgzd.2013.08.027

基于嵌入式 Linux 系统下的 Qt 测试软件开发

廖熹, 易克非

(中国电子科技集团公司第三十研究所装备工程部, 成都 610041)

摘要: 针对某型通信模块测试装备的嵌入式测试软件图形界面设计, 阐述基于嵌入式 Linux 系统下 Qt 测试类程序图形界面的开发过程。详细介绍嵌入式 Linux 下 Qt 开发环境的搭建, 使用 Qt 视图体系的 QGraphicsScene 场景及 QGraphicsItem 子类项完成该测试软件图形界面的设计和方法。结果证明: 该图形界面具有开发周期较短、交互界面友好、界面反应速度快等优点, 对于类似的嵌入式系统图形界面开发具有一定的参考价值。

关键词: Qt; Linux; 嵌入式系统; 图形界面设计

中图分类号: TP311.52 **文献标志码:** A

Qt Test Software Development Based on Embedded Linux System

Liao Xi, Yi Kefei

(Department of Equipment Engineering, No. 30 Research Institute of CETC, Chengdu 610041, China)

Abstract: For the embedded test software graphical interface design, under a certain type of communication module test equipment, the GUI development process of Qt test program is elaborated based on embedded Linux. It introduces the build-up of the Qt embedded Linux development environment in details, using Qt QGraphicsScene and QGraphicsItem subclass item to complete the test software graphical interface design and methods. The results prove that the graphical interface has several advantages such as a shorter development cycle, interactive user-friendly, interface reaction speed, and etc. It provides certain reference value to a similar graphical interface in the area of the embedded system development.

Key words: Qt; Linux; embedded system; graphical interface design

0 引言

当前, 嵌入式技术发展迅猛, 应用极为广泛, 在计算机、消费电子、通信、汽车电子、工业控制和军事国防这 6 大主要应用领域, 嵌入式 Linux 占据了很大的市场规模, 但是由于在 Linux 嵌入式操作系统本身上进行开发难度较高, 所以需要开发者掌握 Linux 系统及其具体开发工具^[1]。嵌入式软件的图形界面将给用户提供最先的感受, 决定用户的青睐程度, 因而嵌入式软件图形界面设计的简洁性、易操作性, 对嵌入式系统来说非常重要。笔者通过针对某型通信模块的嵌入式测试软件设计, 基于嵌入式 Linux 系统下 Qt 测试类应用程序的开发过程, 提出了图形界面的一种实现方法及其移植过程。

1 嵌入式系统架构介绍

嵌入式系统是一个软、硬件结合的综合系统, 一个嵌入式系统的开发包含了硬件开发和软件开发^[2]。ARM 嵌入式芯片是一种高性能、低功耗的 RISC 芯片^[3]。该测试设备硬件方面, 选用华天正科技推出的开发套件 Real6410, 该套件由核心板和底板组成, 核心板上集成三星 S3C6410 处理器, 底板提供了各种外设接口, 通过底板上的串口与接口适配器连接, 接口适配器再通过连接线缆与被测通信

模块连接; 软件方面, 选用了 u-boot1.1.6 作为 bootloader, Linux2.6.28 作为操作系统内核, Qt/E 作为应用程序的开发工具。

2 软件开发环境建立

2.1 软件平台搭建

首先需要构建 arm-linux 交叉编译环境, 在开发用计算机上安装虚拟机软件 vmware, 在 vmware 软件中安装 Linux 操作系统 Fedora9, 作为宿主机。在 Fedora9 中安装交叉工具链工具, 搭建 TFTP 服务器和 NFS 服务器, TFTP 服务器用于通过 u-boot 工具将交叉编译完成的 Linux 内核和文件系统下载到核心板的 nandflash 中, NFS 服务器用于开发阶段核心板 mount 宿主机上的 NFS 文件系统, 减少在核心板 flash 上的烧写次数。

2.2 Qtcreator 和 QT Embedded 介绍

QtCreator 是 Qt 被 Nokia 收购后推出的轻量级集成开发环境(integrated development environment, IDE)。该 IDE 能够跨平台运行, 支持的系统包括 Linux(32 位及 64 位)、Mac OS 以及 Windows。QT Embedded 是 Qt 用于嵌入式开发的版本。

一般开发模式是先用桌面版本进行程序的开发, 然后用 QT Embedded 进行交叉编译, 硬件平台

收稿日期: 2013-03-09; 修回日期: 2013-04-27

作者简介: 廖熹(1979—), 男, 四川人, 工程硕士, 工程师, 从事装备工程、测试技术和软件工程研究。

在 NFS 模式下挂载文件系统进行调试。所以要在虚拟机内的 linux 系统中安装 Qtcreator 和 Qt Embedded 这 2 个软件。

2.3 安装 Qtcreator 软件

从 qt.nokia.com/download 下载 Qtcreator 源代码文件，在 linux 命令行中直接执行这个文件，按照安装提示，完成 Qt Creator 的安装。在首次运行 Qt Creator 时，需对 QtCreator 的 Qt4 库进行配置。

2.4 Qt Embedded 的安装

从 qt.nokia.com/downloads 下载 Qt/E 源代码文件，在 linux 中开启终端，使用以下命令解压 Qt/E 源代码文件：

```
tar jxvf qt-embedded-linux-opensource-src-4.5.2.tar.bz2
```

进入该文件夹：

```
cd qt-embedded-linux-opensource-src-4.5.2
```

对 Qt/E 进行编译配置，在配置完成后，执行 make 进行编译。

设置运行时的环境变量，在 /nfsroot/nfs_root /usr/local/qte4.5.2-arm 中添加脚本文件 setenv-arm.sh，添加如下内容：

```
export QTDIR=/usr/local/ qte4.5.2-arm
export QPEDIR=/usr/local/ qte4.5.2-arm
export PATH=$QTDIR/bin:$PATH
export
LD_LIBRARY_PATH=$QTDIR/lib:/usr/local/lib:$LD_LIBRARY_PATH
export TSLIB_TSDEVICE=/dev/input/event1
export TSLIB_CONFFILE=/usr/local/etc/ts.conf
export TSLIB_PLUGINDIR=/usr/local/lib/ts
export TSLIB_CALIBFILE=/etc/pointercal
#指定触摸屏的工作参数
export
QWS_MOUSE_PROTO="TSLIB:/dev/input/event1"
export QWS_SIZE='800x480'
export QWS_KEYBOARD=TTY:/dev/tty1
至此，完成了 QEmbedded 的运行环境设置。
```

3 应用程序开发

3.1 软件任务需求

本次设计嵌入式测试软件的主要功能是实现对某通信模块的各项功能和性能指标进行测试，支持对单个或多个通信模块的各项功能进行单项测试及组合测试，同时具有测试结果管理功能。

3.2 软件模块划分

该测试软件主要包括人机交互界面、协议解析封装、串口通信、日志管理、用户管理和 IETM 手册等模块，其组成原理框图如图 1 所示。

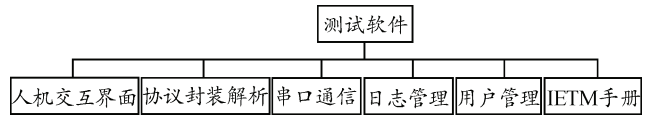


图 1 测试软件组成框图

人机交互界面模块负责图形界面的设计，响应用户操作及反馈相应信息；协议封装解析模块负责将发送的数据按自定义协议进行封帧后交给串口发送，将接收的数据进行解析；串口通信模块负责通过串口将协议数据进行发送或者接收；日志管理模块负责测试结果的维护管理；用户管理负责用户的增加、删除、修改、权限设置等；IETM 手册负责提供电子式交互手册，为用户提供诊断技术和专家知识解析技术。笔者以人机交互界面模块的设计为例，介绍 Qt 应用程序图形界面的开发。

3.3 人机交互界面模块设计

人机交互界面模块主要包括主界面、测试功能项选择、测试功能项设置和日志管理等界面。因该软件人机交互是通过在 7 寸触摸屏上进行触控操作完成，故要求界面简洁、操作简易，具有相对统一的界面风格，合理的布局结构，并合理地通过不同的动画、声音及图形效果，醒目的字体，突出界面的特色，增强用户的体验感。

Qt 应用程序的图形设计功能正好适应这个需求，在本测试软件中，由于涉及到多个图形项的单击，故没有采用通常的 QWidget 进行设计，而是选用基于项的图形视图进行开发，Qt 的视图体系包括一个由 QGraphicsScene 充当的场景及一些由 QGraphicsItem 的子类作为场景中的项。QGraphicsView 是一个窗口部件，使用这个窗口部件可以显示场景，同时能支持缩放和旋转，帮助浏览场景^[4]。QGraphicsView 类继承自 QWidget，GraphicsView 与 QWidget 略有不同。GraphicsView 系统主要由 QGraphicsView，QGraphicsScene，QGraphicsItem 等框架类组成，它们间的关系如图 2。

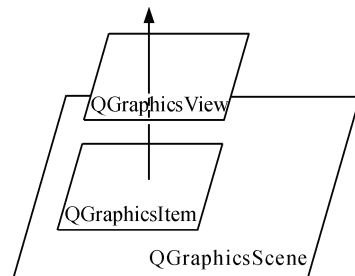


图 2 QGraphicsView 系统组成关系

通过 QGraphicsView 创建显示的矩形区域，QGraphicsItem 对象是所有的可视界面元素，由 QGraphicsScene 对 QGraphicsItem 对象进行管理^[5]。

测试软件各个界面的布局设计大致相同，故不逐一赘述，下面选取测试功能项选择界面进行说明。

3.4 测试功能项选择界面设计

通过创建自定义的 viewfunsetting 类，实现该界面的设计。viewfunsetting 类继承自 QGraphicsView，其定义如下：

```
class viewfunsetting : public QGraphicsView
{
    Q_OBJECT
public:
    viewfunsetting(const QString &images,
QWidget *parent = 0);
protected:
    void mouseReleaseEvent(QMouseEvent
*event);
private:
    QGraphicsScene *scene;
    void addItem();
    ImageItem *image;
};
```

界面初始化主要完成场景的创建，其主要代码如下：

```
scene = new QGraphicsScene(this); //创建场景实例
scene->setSceneRect(0, 0, 795, 476);
QPixmap pim(initpath + "/images/bj.jpg");
scene->setBackgroundBrush(pim.scaled(795,476,
Qt::IgnoreAspectRatio,Qt::SmoothTransformation));
setScene(scene); //设置场景
```

场景设置好后，进行图形对象的添加，其主要代码如下：

```
image = new ImageItem(1,QPixmap(initpath +
"/images/voice.png" )); //添加语音测试键图标对象
image->setData(0, i);
image->setPos(x, y);
scene->addItem(image);
```

相应图标对象全部布局好后，需添加触摸图标响应的事件，其主要代码如下：

```
if (QGraphicsItem *item = itemAt(event->pos()))
{
    if (image = qgraphicsitem_cast<ImageItem
*>(item))
    { //响应语音测试按钮触控事件
        if(1 == image->id())
        {
            flag_test_status = test_voice;
            viewfunrun1 = new viewfunrun("image");
```

```
viewfunrun1->show();
return;
}
.....
```

至此完成了测试功能选择项图形界面设计。通过交叉编译工具对其编译后，将其在文件系统内更新，下载到核心板 flash 中，程序能够正常运行，其中测试功能项选择的界面如图 3 所示，组合测试项参数设置界面如图 4 所示。



图 3 测试功能项选择界面



图 4 组合测试项参数设置界面

4 结束语

笔者选择 Qt 作为图形界面开发的工具，详细介绍了其开发环境的搭建，结合某通信模块嵌入式测试软件图形界面设计，阐述了 Qt 环境下图形界面的设计和实现方法，该测试设备已运用于某通信模块的功能测试，具有开发周期较短、交互界面简洁友好、界面反应速度快等优点，可为类似的嵌入式系统图形界面开发工作提供参考。

参考文献：

- [1] 苗忠良, 宛斌. Qtopia 编程之道[M]. 北京: 清华大学出版社, 2009: 序.
- [2] 杨宗德. 嵌入式 ARM 系统原理与实例开发[M]. 北京: 北京大学出版社, 2007: 106.
- [3] 罗致, 王仲东. ARM Linux 在 AT91RM9200 平台上的移植[J]. 兵工自动化, 2006, 25(1): 85.
- [4] Jasmin Blanchette, Mark Summerfield. C++ GUI Qt4 编程[M]. 闫锋欣, 曾泉人, 等译. 北京: 电子工业出版社, 2009: 151.
- [5] 成洁, 卢紫毅. Linux 窗口程序设计——Qt4 精彩实例分析[M]. 北京: 清华大学出版社, 2008: 185.