# 基于 RT-Linux 和 QT 的嵌入式注塑机控制系统设计

姜 喆 李 平 浙江大学工业控制技术研究所(310027)

**Abstract**

This article analyze the fault of Linux as a real-time operating system,and how the RT-Linux improve its real-time performance.Then an efficient embedded GUI:QT-Embedded and the design of injection modeling controller based them is illustrated.

**Keywords**:Embedded Linux,RT-Linux,QT-Embedded,injection modeling

**摘 要**

文章介绍了嵌入式 Linux 在实时性方面的不足之处、RT-Linux 对 Linux 实时性的改进、嵌入式图形系统 QT-Embedded,以及基于它们开发嵌入式注塑机控制系统的过程。

**关键词**:嵌入式 Linux,RT-Linux, QT-Embedded,注塑机

早期开发的注塑机控制系统由 2 个单片机系统构成,分别称为上、下位机。如图 1 所示,上位机主要完成人—机接口、远程通讯、控制算法、系统管理等功能;下位机直接与传感器和执行器连接,主要负责数据采集、开关量 I/0、数据预处理、D/A 转换等任务。上、下位机之间通过 RS 232 串行通信方式交换数据。上位机选用的是 WB77e58 高档单片机, 基于 C 语言开发图形界面和控制软件。这个系统存在着用户界面不够美观,开发过程繁琐等不足,且系统可扩展性差。因此, 笔者对原有系统的软硬件平台进行了改造, 选用 PC-104 板卡 AR-B1422, 在上面成功的运行了嵌入式 RT-Linux 操作系统, 并基于嵌入式图形工具包 QT-Embedded 开发图形用户界面。

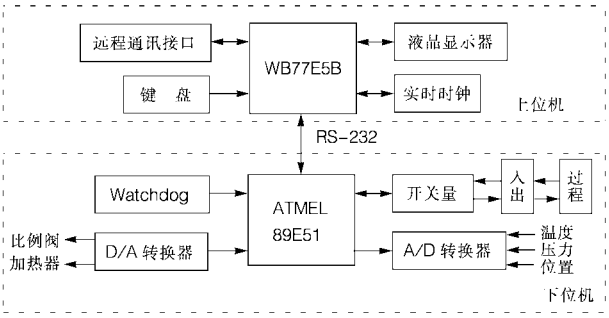


图 1 注塑机控制系统结构图

## 1 嵌入式 Linux 操作系统

如上所示,构建一个实时系统的时候,传统的开发方法一般是采用一个前台/后台循环系统,一个应用软件由一个无穷的循环构成,该循环调用一些模块(也就是函数)来执行希望进行的操作(后台)。而中断服务例程(ISR)用来处理异步事件(前台)。按照这种方式构造的系统有如下几个主要缺点:

1)系统的响应时间难以确定,它的响应时间依赖于后台循环所花费的时间, 而这个执行时间并不是一个常数;

2)系统灵活性差,不易维护,如果想向其中添加新的功能,则必须重新安排整个系统;

3)当系统任务较多时,要考虑的各种可能也多,各种资源如调度不当就会发生死锁,降低软件可靠性,程序编写任务量成指数增加。

基于此, 现在越来越多的开发者开始选用嵌入式 RTOS(实时操作系统)作为自己的开发平台。目前市场上有各种各样的嵌入式操作系统, 如 pSos、vxWorks、QNX、Embedded Linux 等, 它们在是否需付版权费、系统调用、支持的处理器类型以及提供的工具方面存在很大差别。与其它操作系统相比,Linux 具有诸多优点, 笔者通过裁减普通 Linux 得到嵌入式 Linux,然后在它的基础上开发自己的应用系统。不过,由于 Linux 是一个分时系统, 它在实时性方面存在如下不足:

1)其内核是不可切换的,当一个进程运行于核心态时,它将不被换出,直到退出核心态。这样,一个低优先级的进程有可能因为处于核心态而阻碍实时进程的运行;

2)Linux 在进入“临界区”的时候常常关中断,导致实时任务不能被及时调度执行;

3)Linux 的时钟粒度太粗, 它的时钟中断间隔为 10mS,无法满足实时响应的需要;

4)虚拟内存机制导致了系统的不确定性。

由于注塑机控制系统对实时性要求很高, 基于普通 Linux 构建的系统无法满足要求, 选用 Linux 的实时版本 RT-Linux。

为了适应于实时系统的开发,RT-Linux 对 Linux 的内核进行改造, 它通过修改 linux/arch/i386/下与体系结构有关的部分,在 Linux 和硬件之间,加入了一个小巧的,可预测的 RT-Linux 内核。根据评测数据,在一台 386 机器上,RT-Linux 从处理器检测到中断

到中断处理程序开始工作不会超过 15 微秒；对一个周期性的任务，在 35 微秒内一定会执行。如果是普通的 Linux，一般是在 600 微秒内开始一个中断服务程序，对周期性的任务很可能会超过 20 毫秒(20,000 微秒)。因此基于 RT-Linux 开发系统，既可以继续使用原 Linux 的所有功能，如强大的图形和网络环境，又可以保证系统的实时性。

## 2 嵌入式图形系统 QT-Embedded

图形用户界面是系统和用户的对话接口，使用好图形开发工具包来开发图形界面，能提高开发效率。QT 是挪威的 Trolltech 公司推出的一种优秀的跨平台的 C++图形用户开发库，具有如下的优点：

1)优良的跨平台特性，支持 Linux，Windows，UNIX，Mac 等等。使用 QT 类编写的程序可以做到“一次编码，到处编译”，移植到一个新的平台时，往往只需要用面向该平台的编译器重新编译一遍即可；

2)运行速度快，Qt 是一个 GUI 仿真工具包，它使用各自平台上的低级绘图函数，而别的图形工具包往往使用 API 或 API 层仿真；

3)面向对象，Qt 具有良好的封装机制，其模块化程度非常高，可重用性好；

4)GUI 界面实现，支持大量标准的窗口部件；

5)大量的开发文档和开发辅助工具，譬如 Qt Designer 可以帮助用户进行快速界面设计。

Qt-Embedded 是 Qt 面向嵌入式应用的版本，保存有 Qt 的全部优点，相比之下更加小巧，且其大小可根据应用需求灵活定制。笔者选用了 Qt-Embedded，利用其构建系统的图形界面和用户接口。

## 3 系统开发简介

1)系统硬件平台。系统选用的 AR-B1422 是 PC/104 总线结构的。在硬件与软件上，PC/104 与台式标准 PC(PC/AT)体系结构完全兼容，只不过在形态上 PC/104 是十分坚固紧凑的自栈式、模块化结构。AR-B1422 支持 EDO RAM，内部集成了 VGA 显卡，100M/10M 网卡，支持 PC/AT 兼容的键盘和 PS/2 鼠标，提供 RS-232C/RJ-485 串行接口。其尺寸却只为 90.2mm×95.9mm，适用与嵌入式应用。

2)系统结构图如图 2 所示，首先，打造出合适大小的 RT-Linux 内核。选用普通 linux2.2.18 版本的内核，使用 RT-Linux 的 kernel-patch-2.2 对其进行修补，然后，在修改后的内核代码的基础上进行内核配置，去掉不必要的内核支持选项，从而得到一个精巧够用的实时 Linux 内核，在目标平台上成功的运行这个内核，是开发工作的基础。

3) 非实时部分的试运行，主要是测试基于 QT-Embedded 编写的界面。利用 QT-Embedded 提供的窗口类，派生出自己的窗口子类，编写图形用户界面，包括时间设定窗口、压力设定窗口、流量设定窗口、动作设定窗口、温度设定窗口、位置设定窗口、模具资料设定窗口。它负责对用户的操作进行提示，接受用户输入的运行参数，传递给实时模块。并根据实时模块传送过来的参数，向用户显示系统当前的工作状态，譬如处于开模状态，锁模状态，报警状态等等。由于 QT 支持 GIF，所以系统状态的中文显示是直接利用 GIF 的图片。

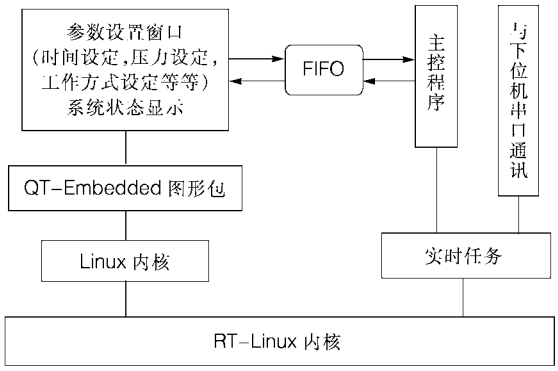


图 2 系统结构图

4)实时模块的运行。这一部分主要负责与下位机通过串口进行通讯和实现控制逻辑，它接收下位机传送过来的温度、压力等数据，行程开关、阀门等开关量；然后根据当前的动作指令和机器状态计算出下一步的动作，把控制指令发送到下位机。其实时性是利用 RT-Linux 提供的高精度定时器。实际上实时部分的程序是作为模块插入的，此时要注意模块之间的相互依赖性。在插入自己的实时模块之前，必须要先插入 RT-Linux 的调度模块 rtl_sched，以及一些相应的支撑模块，如 rtl_time,rtl_posixio 等等。在测试阶段，可以通过手工加入，以后可以在系统的启动脚本里面自行插入模块。

5)实时部分和非实时部分的集成。这一步主要是测试两者之间通过 FIFO 的通讯情况。非实时部分负责把用户输入的参数通过 FIFO 传送到实时模块，实时模块负责把系统的状态参数通过 FIFO 传递给非实时部分，由它显示给用户。

## 4 结束语

结果表明，基于 RT-Linux 和 QT-Embedded 进行系统开发，大大的减轻了开发强度，所构建的系统稳定，实时性强图形界面是标准的 Windows 风格，更加美观。更重要的是系统具有很好的可扩展性，在以后的开发中如果要实现各系统的联网数据交换，利用 Linux 的网络功能，可以容易的加入网络支撑模块。

## 参考文献

1 Yodaiken V.The RTLinux Manifesto.Internet,http://www.rtlinux.org www.trolltech.com

2 邹思轶.嵌入式 Linux 设计与应用.北京：清华大学出版社，2002

[收稿日期：2002.12.27]