Broadview
www.broadview.com.cn

嵌入式技术丛书

亚嵌教育 www.akaedu.org 作品系列

嵌入式GUI开发设计
——基于MiniGUI

北京亚嵌教育研究中心 组编
韩超 编著

本书特点：

◎基于处于快速发展阶段的MiniGUI系统，由权威培训机构为读者解读MiniGUI3.0新技术的应用。

◎结合了GUI理论和MiniGUI系统的实现，让读者同时从广度和深度上对嵌入式GUI乃至嵌入式系统有深入的了解。

◎提供了详尽的配套光盘，包含大量的源代码和应用，可使读者具备快速搭建系统的能力，并能快速上手，搭建自己的系统。

◎书中包含众多示例，可操作性、可扩展性强，能充分发挥读者的想象力。

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

MiniGUI®

嵌入式技术丛书

亚嵌教育 www.akaedu.org 作品系列

嵌入式GUI开发设计——基于MiniGUI

北京亚嵌教育研究中心 组编
韩超 编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

《嵌入式 GUI 开发设计——基于 MiniGUI》是一本嵌入式图形用户系统应用开发方面的教程，由作者韩超结合飞漫官方授权资料编著。本书通过介绍 MiniGUI 系统的开发，给读者一个学习嵌入式 GUI 程序设计和实现的平台。MiniGUI 系统是一个由中国人实现、目前广泛在应用的优秀嵌入式 GUI 系统，通过这个系统的学习，可以了解嵌入式 GUI 开发的思想和方法，也可以加深对嵌入式系统开发的理解。本书结合嵌入式 GUI 系统的通用知识对 MiniGUI 系统进行了充分的介绍，篇幅控制详略得当，配合光盘使用，可以达到事半功倍的效果。本书主体分成三个层次：MiniGUI 的架构、MiniGUI 的各种功能、基于 MiniGUI 构建应用程序，这是学习嵌入式 GUI 系统较为便捷和高效的方式。它既适合需要进入 MiniGUI 应用程序开发领域的人员，也适合需要对通用的 GUI 技术学习的人员。

图书在版编目（CIP）数据

嵌入式 GUI 开发设计：基于 MiniGUI / 北京亚嵌教育研究中心组编.—北京：电子工业出版社，2009.5
ISBN 978-7-121-08606-9

Ⅰ. 嵌… Ⅱ. 北… Ⅲ. 软件工具－程序设计 Ⅳ. TP311.56

中国版本图书馆 CIP 数据核字（2009）第 048632 号

责任编辑：李 冰
印 刷：北京智力达印刷有限公司
装 订：北京中新伟业印刷有限公司
出版发行：电子工业出版社
北京市海淀区万寿路 173 信箱 邮编 100036
开 本：787×1092 1/16 印张：19.5 字数：435 千字
印 次：2009 年 5 月第 1 次印刷
印 数：4000 册 定价：45.00 元（含光盘 1 张）

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

时至今日，MiniGUI 已经走过了十年的路程。目前的 MiniGUI 已经成为一套具有完整清晰定义的移植层、优秀的窗口管理机制、丰富而易用的控件并可以进行外观定制的嵌入式 GUI 系统；在上层的组件中，MiniGUI 还包含了 3D 接口、输入法、打印甚至浏览器等组件。从一款默默无闻的小型 GUI 程序，到现在成为全球嵌入式软件领域的鼎鼎大名的软件。对于中国人来说，MiniGUI 所承载的不仅仅是一款软件的成功，更体现了中国人的东方式智慧完全有能力在世界的嵌入式软件行业中占据一席之地。可以肯定地说，MiniGUI 的诞生和发展鼓舞了中国人在软件研发领域的热情，也坚定了中国人进入世界嵌入式软件开发前沿的信念。

MiniGUI 的发展历程同时也从一个侧面体现了中国嵌入式技术的发展历程。MiniGUI 的创始人魏永明先生等一批中国嵌入式技术的先驱引领了嵌入式技术在中国发展的潮流。自从 1998 年魏永明先生发布 MiniGUI 的第一个版本开始，这个项目就一直吸引并带领着来自不同专业、不同领域的国人进入了嵌入式技术的世界。作为一个交叉学科，嵌入式系统的技术需要综合性的人才，MiniGUI 的发展吸引了更多人进入这个领域。作为一个嵌入式的 GUI 系统，MiniGUI 的开发让人们更多熟悉了嵌入式系统的开发流程。在这个过程中，嵌入式技术的门槛对于大众降低了很多，以前某些只有研发企业和高校才有能力开发的项目，现在普通的爱好者也可以参与开发。更多人的参与同时也促进嵌入式技术在中国有了更好的发展。

从 MiniGUI 的开源版本中，我们深刻地感受到了这个中国人自己创作的一套完整嵌入式 GUI 系统的魅力。随着对 MiniGUI 的学习，很多人在数年之前进入了嵌入式系统领域，他们当中很多人成为了现今中国大陆地区嵌入式技术的中坚。在对 MiniGUI 的关注中，每一个从事嵌入式研发相关的人员都可以从 API 的制定、为移植构建的抽象层、代码组织、软件组织架构乃至整个嵌入式系统中间件的架构得到丰富的收获。

飞漫公司的成立已经有 6 个年头了，MiniGUI 在飞漫公司的运作中得到了更大的发展。在这个过程中，我们欣喜地看到 MiniGUI 并没有像很多纯粹的技术产品一样虎头蛇尾，而是在开源版本和商业版本双线上成功地发展。MiniGUI 同时具有了开源软件开放性的特点和商业软件高品质的特点。

作为一款软件产品，MiniGUI 的商业版本使得 MiniGUI 得到了长足的进展，让 MiniGUI 稳定地运行到了更多的设备上。目前，MiniGUI 在小型嵌入式系统上面的移植具有无可比拟的优势。对比同类的嵌入式 GUI 系统，MiniGUI 不仅性能开销较低，而且明显具有更强的可移植性和可配置性。MiniGUI 可以灵活提供各种功能定制：小到简单的图形绘制库，大到整个系统的解决方案。由于 MiniGUI 系统生根发芽的土壤在中国，它相比其他的 GUI 系统更适合由中国人来开发。目前 MiniGUI 在中高端手机、PDA 类产品、机顶盒、智能家居以及工控、仪表领域都有了广泛的应用。

尤其值得指出的是，虽然飞漫是一家研发软件产品的公司，但是魏永明先生带领飞漫

团队对中国技术界本着非常负责任的态度，依然在进行着一定程度的技术传播和交流，这也很大程度上促进了嵌入式软件技术的提高。飞漫的成功对中国软件业发展也是一个很好的启迪。

MiniGUI 在稳定发展了若干年后，目前最新的版本为 MiniGUI3.0。此时的 MiniGUI 在接口上已经基本稳定，但也具有了很强的扩展性。MiniGUI 正向着可以提供整套的解决方案、友好的开发环境、并且可以接受第三方的应用的方向发展。飞漫的定制模式也让基于 MiniGUI 的软件开发的分工更加明确，这样可以使基于 MiniGUI 的产品具有更快更好的开发效率。MiniGUI 的发展历程不仅是技术上的发展，也是对行业产业链和发展模式的思考。

本书主要内容

本书通过介绍 MiniGUI 系统的开发，给读者展示一个学习 GUI 系统程序的平台。本书的内容可以体现两个层次的内容：其一是对 MiniGUI 的应用程序开发的学习，读者可以通过对照光盘中的程序学习本书的内容，在学习之后可以单独进行 MiniGUI 应用程序的开发；其二是对 GUI 系统程序设计思想的学习，虽然 MiniGUI 系统和其他 GUI 系统存在差异，但是 GUI 系统整体设计的思想是通用的，读者根据本书的内容学习并进行思考后，在开发其他的 GUI 系统的时候也可以快速地上手。

对于一个 GUI 系统的学习，可以分成三个层次：第一个层次是了解 GUI 系统层次结构和程序框架；第二个层次是熟悉这个 GUI 系统所提供的功能；第三个层次是如何使用 GUI 系统构建完整的应用程序。

本书的第 3 章、第 4 章主要介绍了 MiniGUI 系统在第一个层次的内容。通过这两章的学习，读者可以熟悉 MiniGUI 系统的软件各个层次的关系，也可以学会如何使用 MiniGUI 构建一个简单的应用程序。

本书的第 5 章到第 12 章主要介绍了 MiniGUI 系统在第二个层次的内容。主要内容为 MiniGUI 主窗口、对话框、控件的使用、GDI 编程以及其他一些方面的内容。其中，主窗口、对话框和控件的使用三者构成了完成 MiniGUI 应用程序的主体；GDI 名为图形设备接口，这是一组较低层次的 API，GDI 编程可以让读者深入控制系统，进行更灵活的程序设计；其他方面的内容包括菜单、键盘和鼠标处理、图标、光标、定时器等内容。第 12 章单独介绍了 MiniGUI 扩展库的内容，包括扩展的对话框、控件和皮肤等功能。

本书的第 13 章到第 16 章主要介绍了 MiniGUI 系统在第三个层次的内容。这些章节通过展示 MiniGUI 演示库的例子，让读者学习使用 MiniGUI 提供的接口构建完成的应用程序。这些简单的应用程序可以体现 GUI 系统设计的一些思路。

本书读者对象

本书既适合需要进入 MiniGUI 应用程序开发领域的人员，也适合需要学习通用的 GUI 技术的人员。MiniGUI 系统是一个目前广泛应用的嵌入式 GUI 系统，可以通过学习这个系统进入 GUI 程序开发的领域。本书要求读者具有 C 语言编程的一定基础。本书的开发环境可以使用 Ubuntu、Fedora 或 OpenSuse 等 Linux 发行版，需要安装了 GCC、make 等基

本的工具，然后可以根据本书提供的光盘建立开发环境。本书光盘中也提供了部分Windows 中的开发环境，可以在 Windows 的 Visual C++6.0 环境中使用。

在本书的学习过程中，为了取得事半功倍的效果，读者可以使用边学边练的方式，对于书中的示例程序，应对照光盘中的程序或自己练习编写程序进行验证和扩展。由于各个控件的使用相似性较多，本书只选取了具有代表性的进行介绍，并在附录中列出了控件的特性。读者可以通过参考 MiniGUI 的 API 手册和头文件中的描述，自己编写程序进行更多的练习。对于完整 GUI 程序的组织，读者应根据自身需要完成软件的需求，结合一般软件工程的思想和 MiniGUI 接口的特点进行设计和实现。此外，MiniGUI 的 API 大都是通过深思熟虑发布的，具有简洁、含义明确和便于扩展的特点，读者在这方面进行思考，可以提高自身软件工程的思想。

本书由韩超规划和编著，北京飞漫软件技术有限公司为本书提供了丰富的资料。本书的配套光盘由韩超组织完成，其中部分内容依据北京飞漫软件技术有限公司授权加入。在本书及其配套光盘的编著过程中，亚嵌嵌入式教育研究中心为本书提供了支持，郎铁山、王旭光参与了审校工作。本书的出版也是飞漫多年技术积累的成果。本书作者和参与编著人员长期立志于推广嵌入式技术，努力促进公众的技术共同进步。由于时间仓促，本书可能依然存在一些错误和问题，请读者见谅。MiniGUI 的发展需要广大的技术人员共同参与，欢迎读者讨论和指点。

目　录

第 1 章

嵌入式 GUI 系统概述

本章是对嵌入式 GUI 系统的概述部分。在内容上，主要从发展历程和系统架构的方面介绍嵌入式 GUI 系统。

在学习本章的过程中，读者应重点关注以下的内容：

- 嵌入式 GUI 系统和 PC 的 GUI 系统的差异
- 嵌入式 GUI 系统的设计理念
- 嵌入式系统软件架构及嵌入式 GUI 系统的位置
- 嵌入式 GUI 系统本身的层次结构
- QTE、MicroWindows 和 MiniGUI 三个系统的结构

1.1 嵌入式 GUI 系统

GUI 系统的含义为图形用户接口（Graphic User Interface）系统，它是计算机系统和用户的接口。在嵌入式领域，随着用户需求的增加和技术的发展。GUI 系统已经有了越来越广泛的应用。

1.1.1 嵌入式 GUI 系统的作用

在 20 世纪 90 年代，作为人机交互的接口，GUI 系统在桌面计算机系统中就已经有了很广泛的应用，其中具有代表性的是：微软的 Windows 系列（包括 Windows95、Windows98、WindowsMe 等）和 WindowsNT 系列（包括 Windows NT 4.0、Windows 2000、Windows XP 等）系统，桌面 Linux 的 Gnome 系统和 KDE 系统。相比传统的命令行接口（Command Line Interface）方式。GUI 系统为用户提供了更友好的界面，将用户从枯燥界面中解脱出来。同时，GUI 系统也让计算机系统的使用更加简单。

在嵌入式系统发展的初级阶段，GUI 系统的应用相对较少。例如，在相对简单的、以单片机为核心的简单工控系统中，人机交互通常以 LED（发光二极管）和按键相结合的方式，随后才有了简单的屏幕。这些只是简单的输入、输出系统，不能被称为“GUI 系统”。

GUI 系统在早期的嵌入式系统中应用较少的主要原因有两个：一个方面，功能角度考

虑，很多传统的嵌入式系统逻辑相对简单，人需要从系统获取的信息较少，人对系统的操作也较少；另一个方面，从技术角度，嵌入式处理器的计算能力较低、嵌入式系统内存较少等现实和缺少适用于嵌入式系统的 GUI 系统，使得嵌入式系统中实现图形化的人机交互界面有一定的技术难度。

随着嵌入式系统的发展和普及，GUI 在嵌入式系统中的作用越来越突出。当智能手机、PDA 等系统出现后，嵌入式系统已经不是仅仅给个别专业操作人员所使用的，而需要适用于很广泛的群众。虽然在传统的嵌入式控制领域，人机交互的内容并不是很复杂，但是使用者同样需要更友好的界面。因此，GUI 在嵌入式系统中的普及是大势所趋。

GUI 系统需要最终向用户提供输出和输入两个方面。在输出方面，GUI 系统向用户提供一个图形化的界面，在输入方面，GUI 系统需要接受用户的操作，从而达到通过界面控制系统的目的。

1.1.2 嵌入式 GUI 系统的层级结构

嵌入式系统的设计一般秉承精简、高效的原则，其软件的层次结构相对简单。自下而上，一般可以分为硬件层、操作系统层、中间件、应用层等几个层次，如图 1-1 所示。

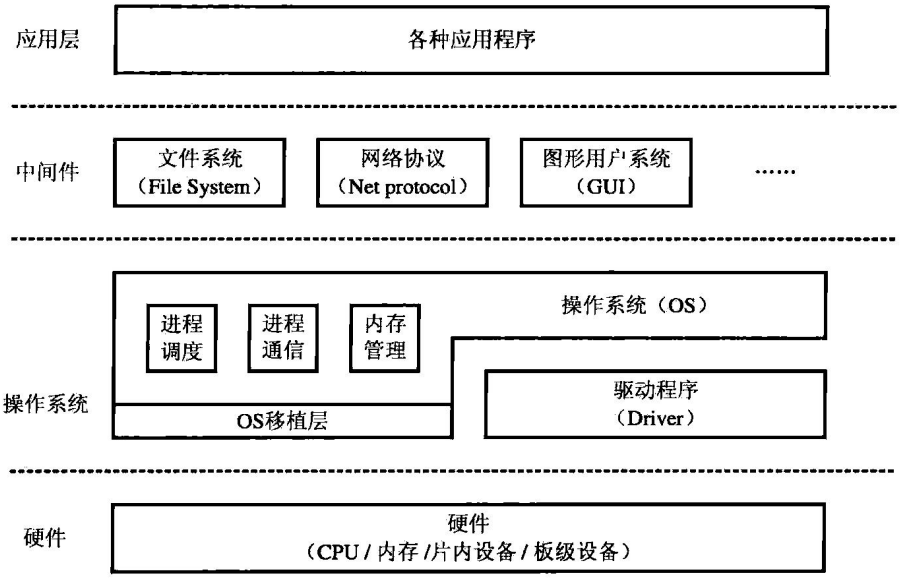

图 1-1 嵌入式系统的软件架构

嵌入式系统的硬件分为处理器（通常包含 CPU 和片内设备）、内存、板级硬件几个部分。CPU 涉及了特定体系结构的运算和控制单元，例如 ARM、MIPS 等，在整个系统的软件开发中，一般都要基于该体系结构的编译工作。片内设备是处理器内部的硬件模块，比较重要的包括内存管理器、中断控制器、定时器、GPIO 等。嵌入式系统的内存的 RAM 主要通过 SRAM 和 SDRAM 实现，可固化的存储器主要应用 Nor Flash 和 Nand Flash。板级的硬件包含了嵌入式系统中需要，但是处理器片内没有集成的部分，通常包括片内部件的外围硬件、总线扩展、GPIO 扩展几种形式。

嵌入式的操作系统建立在硬件之上。操作系统通常具有进程（任务）调度、进程间通信、内存管理等方面。嵌入式系统的操作系统一般都具有一定的可移植性，可以建立在不同的硬件平台上。操作系统的移植层通常包括对某种体系结构 CPU 的支持，需要涉及定时器、中断控制器、系统内存等硬件。驱动程序是操作系统和硬件的接口，大量的硬件需要通过操作系统框架内的驱动程序，向上层提供控制硬件的接口。

提示： 关于操作系统和上层软件的关系，某些操作系统被称为内核空间（Kernel Space），而操作系统以上的部分被称为用户空间（User Space）。按照功能，操作系统以上的部分可以分为中间件和上层应用两个部分。

中间件一般提供了一些相对底层的软件层次的功能。它的实现一般不包括应用程序的逻辑，而是向上层软件提供了各种方便的应用程序接口（API）。中间件需要通过对操作系统的调用来建立，常常需要控制硬件。在嵌入式系统中，常用的中间件包含文件系统（File System）、网络协议（Net protocol）、图形用户系统（GUI）等几种，它们一般都需要控制特定的硬件来实现。此外，数据库（Database）等不需要控制硬件的下层软件，通常也作为中间件的形式出现。

应用层包含了应用程序的逻辑，它通过调用中间件和操作系统来实现。应用层的软件程序也可以由上下若干层和不同的模块组成。

注意： 图 1-1 是一个嵌入式系统软件相对通用的结构图，并没有依赖特定的硬件平台和操作系统。

uC/OS 是一种简单的操作系统，操作系统内核的功能比较简单，只有任务调度、任务通信和简单的内存管理功能，其内核空间和用户空间的界限也不是很明显。文件系统、网络协议、图形系统均通过的中间件的形式实现。

在 Linux 操作系统中，文件系统和网络协议集成在操作系统内核中。Linux 的文件系统包含了上层通用的虚拟文件系统（Virtual File System）和下层的操作系统实现（如 ROMFS、JIFFS），建立在下层的 RAM 等 Flash 硬件上。Linux 的网络模块也是操作系统组成部分，提供了对 TCP/IP 栈的支持，网络模块建立的硬件基础是网络设备的驱动程序。然而，Linux 操作系统没有包括对 GUI 的支持，因此 Linux 的 GUI 系统通常以中间件的方式实现。

在 Linux 中，GUI 系统的实现如图 1-2 所示。

在 Linux 中，GUI 系统具有中间件典型的特点：核心实现提供了界面、控件等功能，API 层是为上层应用提供的接口、移植层适应不同的硬件的需求。GUI 系统核心通常需要建立在 C 库和移植层的基础上。GUI 核心通常依赖操作系统提供的一些机制，如多线程支持等。GUI 系统的移植实现基础包含输出设备和输入设备两个部分。输出设备在 Linux 中通常需要通过对帧缓冲区驱动（例如：/dev/fb0）的调用来实现，输出设备的移植通常是 GUI 系统的重点。与之相比，输入设备是多种多样的，包括鼠标、键盘、触摸屏、按键等设备，对不同输入设备的支持是 GUI 系统的难点和重要特性。

GUI 系统的核心库通常作为嵌入式系统的中间件，而使用 GUI 核心库的 GUI 应用程

序属于应用程序层的程序。

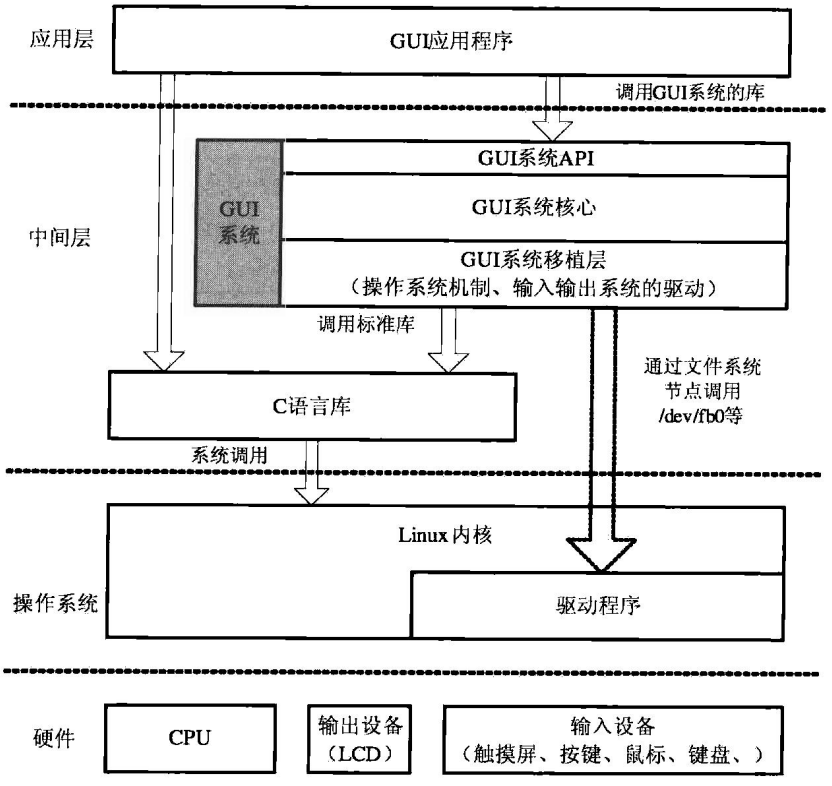

图 1-2　Linux 中 GUI 系统的软件结构

1.2　嵌入式 GUI 系统的设计原则

嵌入式 GUI 系统，要提供给上层的应用程序绘制图形界面以及接收用户输入的能力。从实现的方式上，它既可以是一套库，也可以是和应用程序一起编译的源代码。

在嵌入式系统中，GUI 系统是实现图形化界面的核心。由于嵌入式系统的特殊性。在设计原则方面，嵌入式的 GUI 系统应该具有以下几个特点：

- 可移植性

相比通用计算机系统统一的软硬件结构，各个不同的嵌入式系统之间相差较大。作为一款成功的嵌入式 GUI 系统，应该能在不同的嵌入式平台中运行，这就需要嵌入式 GUI 系统具有较强的可移植性。

所谓可移植性，体现在操作系统和硬件结构两个方面。在操作系统方面，由于嵌入式 GUI 系统需要建立在操作系统提供的一定机制上面（例如多任务支持、任务间通信等），而不同的操作系统提供的机制完全不同。为了能在不同的操作系统中运行，嵌入式 GUI 就需要具有一定的可移植层来支持不同的操作系统。硬件方面又包括了 CPU 体系结构、输出设备、输入设备等方面：适应不同 CPU 体系结构，需要嵌入式 GUI 系统的代码可以

在不同的编译器上编译，一般来说C语言实现的系统可以满足这个要求，但使用特定体系结构汇编的实现就不能适应这种需求；输出设备（显示屏）和输入设备（键盘、鼠标、触摸屏、按键）在不同的嵌入式系统中相差也很大，这也要求嵌入式GUI系统可以支持不同系统的硬件接口。

- 较高的稳定性和可靠性

嵌入式系统对稳定性和可靠性有很严格的要求。个人PC的崩溃可以通过重新启动等方式弥补，但是嵌入式系统的崩溃就可能导致无法挽回的严重后果。因此嵌入式GUI系统，需要具有更强的稳定性和可靠性。例如：在一些控制、通信系统中，GUI系统属于提供的是辅助的人机交互功能，如果由于GUI系统的问题，导致系统核心功能崩溃，这绝对是得不偿失的。

事实上，在一些嵌入式系统中，缺少了不同任务的保护机制，整个系统运行在一个内存空间内，因此，由于一个子系统的问题导致整个系统崩溃的概率更高。对于嵌入式GUI系统，一方面需要有较高的稳定性和可靠性，减少崩溃的概率；另一方面，在GUI系统已经崩溃的情况下，也需要确保尽量较少影响其他子系统的工作，将问题控制在一定范围内。

- 系统开销少

相对PC系统，嵌入式系统的资源都是相对有限的。系统的资源包括，包括处理器的频率、Flash的空间和RAM空间等几个部分。在嵌入式系统中，不但资源有限，而且通常还运行着一些比GUI系统更重要的程序。因此，嵌入式GUI系统必须具有开销小的特点，不能抢占系统过多的资源。从编译的角度，GUI子系统代码规模要有限制，避免占用太多的Flash；从运行的角度，GUI子系统的处理器开销和内存开销也是需要严格限制的。如果占用系统资源太多，不但GUI系统将无法正常运行，甚至造成整个系统无法工作。

- 较高可配置性

在嵌入式应用中，由于不同的系统相差较大，因此嵌入式GUI系统最好具有一定的可配置型，从而适应不同系统的需求。成功的嵌入式GUI系统需要适应不同嵌入式应用的需求。

可配置性通常包括可裁减性、界面特性配置、皮肤和主题配置等方面。在剪裁性方面，GUI系统可以提供很多的功能，但是在所有的功能并不是都要在某种特定系统上使用，可以去掉不相关的功能来节省系统的开销。对于界面的特性配置，需要适应不同的解决方案需求，例如，一个智能手机的界面和一个工控仪表的界面相差很大，用户输入方式的差别更大。此外，在界面基本相同的情况下，皮肤和主题的更改和配置也可提供系统的灵活性。

可配置性的实现通常有两种手段，一种是通过条件编译来实现，一种是通过配置文件实现运行时（Run Time）的配置。

提示：在学习一个嵌入式GUI特性和功能的时候，需要关注可移植性、稳定性和可靠性、系统开销、可配置性等几个方面的内容。

一般来说，嵌入式GUI系统的设计并不依赖于某个特定的系统。实现嵌入式系统的GUI应用程序的开发，只需要对GUI系统移植和使用。在嵌入式系统的开发中，GUI

系统一般属于中间层，不需要改动。对于某些开源的 GUI 系统，也可以针对特定的应用改动或者优化。对于嵌入式 GUI 系统，关注的重点在上层的接口应用和下层的移植两个方面。

1.3 嵌入式 GUI 系统发展状况

目前在常用的嵌入式 GUI 领域，广泛使用系统由嵌入式 QT、Microwindows 和 MiniGUI 三种。

1.3.1 QT/Embedded

Qt/Embedded 是著名的 Qt 库开发商 TrollTech 公司（http://www.trolltech.com/）发布的面向嵌入式系统的 Qt 版本。QT 是桌面 Linux 系统普遍使用的图形库（KDE 桌面系统基于 QT）。与桌面版本不同，Qt/Embedded 已经直接取代掉 X Server 及 X Library 等多层次，直接使用 Frame buffer，所有的功能全部整合在一起。Qt/Embedded 为用户提供了与桌面 QT 相似的应用接口，这可以让桌面的应用很容易移植到嵌入式系统上。QT 和 QTE 系统结构如图 1-3 所示。

QT 是一个功能非常强大的 GUI 系统。实际上，QT 的功能已经超越了传统图形库的范畴。在 QT 中不但包括了 GUI 系统的窗口、控件等内容，也包括画布、网络甚至数据库模块。实际上 QT 提供给应用程序的是一个平台。

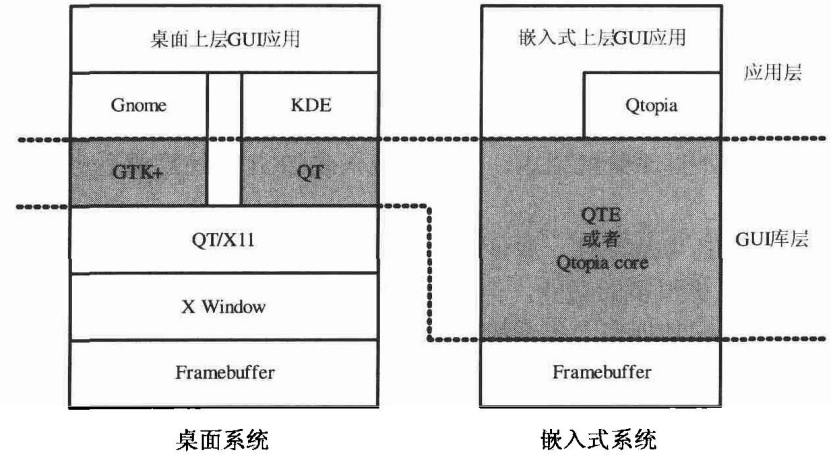

图 1-3 QT 和 QTE 系统结构

QT 编程使用 C++面向对象的所有机制，并且使用 QT 自身一些基于 C++附加的功能、信号和槽以及相应的宏编译（moc）机制。QTE 的强大开发功能，为快速建立嵌入式 GUI 程序提供了很大的方便。

Qtopia 起源于 QPE（全称 Qt Palmtop Environment，QT 掌上电脑环境），是构建于 Qt/Embedded 之上的一系列应用程序。从版本 4 的 QT 开始，Trolltech 将 Qt/Embedded 并入了 Qtopia，并推出了新的 Qtopia4。在 Qtopia4 中，以前的 Qt/Embedded（基础库部分）

被称为 Qtopia Core，作为嵌入式版本的核心，既可以与 Qtopia 配合，也可以独立使用；以前的 Qtopia 也采用分层的结构：底层为核心的应用框架和插件系统，被称为 Qtopia Platform；上层为应用程序，按照不同的类型分成不同的包。

> **提示：** QT 中使用的 C++以及各种附加的机制，为编程提供了很大的方便，同时也造成了较高的系统开销。在资源非常有限的嵌入式系统中，过高的开销从某种程度上限制了 QTE 的使用。

1.3.2 Microwindows

MicroWindows 是一个开放源代码的项目，采用 MPL 条款发布（类似 LGPL 条款），目前由美国 CenturySoftware 公司主持开发。MicroWindows 的目的是把图形视窗环境引入到运行 Linux 的小型设备和平台上。作为 X Window 的替代品，Microwindows 可以使用更少的 RAM 和文件存储空间（100K-600K）提供与 X Window 相似的功能。

MicroWindows 是一个基于典型客户/服务器体系结构的 GUI 系统，基本分为三层。最底层是面向图形输出和键盘、鼠标或触摸屏的驱动程序；中间层提供底层硬件的抽象接口，并进行窗口管理；最高层分别提供兼容于 X Window 和 Windows CE（Win32 子集）的 API。该项目的主要特色在于提供了类似 X 的客户/服务器体系结构，并提供了相对完善的图形功能，包括一些高级的功能，比如 Alpha 混合、三维支持、TrueType 字体支持等。由于基本上用 C 语言实现，Microwindows 的可移植性非常好，只有某些关键代码使用了汇编以提高速度。由于缺乏核心的维护人员，Microwindows 也存在一些问题，代码质量不是很好，存在一些 bug。

MicroWindowGUI 系统结构如图 1-4 所示。

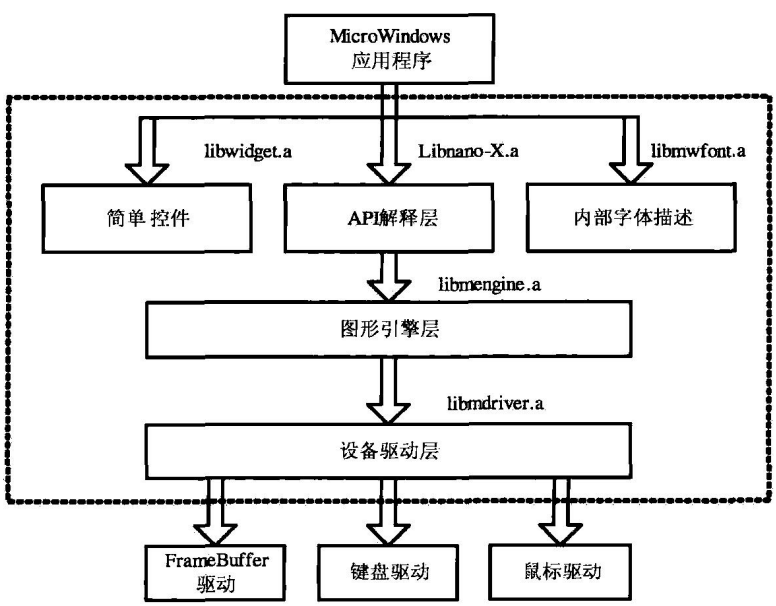

图 1-4 MicroWindows GUI 系统的结构