uC/OS-II 在 80C51 下的移植

The Migration of uC/OS-II to 80C51

秦绍华* 陈 涤

QIN Shao-hua CHEN Di

摘 要 本文讨论了嵌入式操作系统 uC/OS-II 在 80C51 系列单片机上的移植,并针对在 KEIL C51 编译器环境下移植的一些特殊问题进行了阐述,最后简要分析了 uC/OS-II 嵌入式操作系统的资源和响应时间问题。

关键词 嵌入式实时操作系统 uC/OS-II

Abstract This paper researches the method of porting open source RTOS uC/OS-II to 80C51, point out the difference about the porting to the compiler of keil c51, and then analyse the resource and the response time of the system.

Keywords Real time operating system uC/OS-II

随着 PC 机和因特网的发展,我们已经进入后 PC 时代,尼葛洛庞帝曾经预言嵌入式智能电脑将是 PC 和因特网后的最伟大的发明。正如我们所看到的,嵌入式系统正在通信、工控、汽车、信息家电、国防等领域得到广泛的应用。嵌入式实时多任务操作系统(RTOS)可以管理系统中的软硬件资源,是嵌入式软件的运行平台,在嵌入式系统中应用操作系统可以使工程师更专注于应用程序的设计,有利于复杂程序的设计。而操作系统是一个通用的程序,要在自己的嵌入式系统中应用操作系统,必须根据所用 CPU 的不同来进行移植。本文将具体论述嵌入式实时多任务操作系统 μC/OS-II 在 80C51 下的移植,并就在 Keil C51 编译器环境下的一些特殊性作了说明。而且进一步讨论了由于引入操作系统而引起的嵌入式系统资源的增加和实时性的变化。

1 背景介绍:

μC/OS 是美国人 Jean J. Labrosse 于 1992 年编写的一个公开源代码的实时多任务操作系统。1998 年发布 μC/OS-II,目前的版本为 μC/OS-II V2.7。μC/OS-II虽然是一个公开源代码免费的 RTOS,其性能和安全性却可以与商业产品竞争。而且已经通过了联邦航空局(FAA)商用航行器认证。可以用于航空器等与人性命攸关的领域。μC/OS—II具有源码公开,多任务,占先式内核,实时性强,可裁减,便于移植等特点。

80C51 系列单片机具有结构简单,应用灵活等特点,现在微处理器的性能飞速发展,32 位机的应用逐渐增多,但是 80C51 作为 8 位机,市场占用率依然很高,现在是 8 位机,16 位机,32 位机三分天下,特别是在嵌入式 Internet 快速发展的今天,利用廉价的 80C51 实现众多传感器、控制器等电子设备的网络互联,更是具有先天优势,因此在 80C51 上移植 μC/OS-II 是很有意义的。Keil C51 是 Keil 公司的一款针对 51 系列单片机的编译器,其 uVision2 集成开发环境将项目管理、源代码编辑和程序调试等组合在一起,功能强大,使用方便。在 Keil C51 环境下移植 μC/OS-II 可以直接进行软件仿真,不必下载到硬件上运行。

2 μC/OS-II 的移植:

移植 μC/OS-II 必须满足以下条件:

a. 处理器的 C 编译器能产生可重入代码;

b. 用 C 语言就可以打开和关闭中断(在 C 语言中能嵌入行汇编也可以);

c. 处理器支持中断,并且能产生定时中断;

d. 处理器支持能够容纳一定量数据的硬件堆栈;

e. 处理器有将堆栈指针和其他 CPU 寄存器读出和存储到堆栈或内存中的指令。

对于 80C51 及其编译器来说,上面这些条件都是可以满足的。

μC/OS-II 内核分为与处理器无关部分和与处理器相关部分,其中与处理器相关的文件有三个:C 语言文件 OS_CPU.H,OS_CPU_C.C 和汇编文件 OS_CPU_A.ASM。主要移植工作就是针对这三个文件做一些改写,使之与处理器相适合。

OS_CPU.H 中定义了与处理器相关的常量、宏和类型定义及其他一些与处理器有关的函数。在 μC/OS-II 中进出代

* 山东大学信息与工程学院 250100

码的临界区时要开关中断,在 C51 中用 EA = 0 关中断;EA = 1 开中断来实现。这样定义既减少了程序行数,又避免了退出临界区后关中断造成的死机;C51 堆栈从下往上增长,所以 OS_STK_GROWTH 定义为 0;μC/OS-II 中进行任务切换时通过执行 OS_TASK_SW()来产生中断,要求处理器提供软中断或是陷阱指令来完成这个功能,因为 C51 没有软中断指令,所以用程序调用代替,# define OS_TASK_SW() OSCtxSw(),两者的堆栈格式相同,而用 RETI 指令复位中断系统;

OS_CPU_C.C 文件中需要用户定义 6 个 C 语言函数:OSTaskStkInit(),OSSTaskCreateHook(),OS2TaskDelHook(),OSTaskSwHook(),OSTaskStatHook(),OSTimeTickHook()。实际必须定义的只有 OSTaskStkInit()函数,其它 5 个函数需要声明,但不一定要有实际内容,它们是由系统函数调用的钩挂函数,让用户能在操作系统中加入自己需要的一些功能,如果不需要使用也可以不定义,但要将 OS_CFG.H 中的 OS_CPU_HOOKS_EN 设为 0。OSTaskStkInit()函数由任务创建函数 OSTaskCreate() 调用,功能是初始化任务堆栈。初始状态的堆栈模拟发生一次中断后的堆栈结构,按照中断后的进栈次序预留各个寄存器的存储空间。它需要的参数是:任务代码起始地址(task)、参数指针(pdata)、任务堆栈顶端的地址(ptos)。堆栈初始化工作结束后,OSTaskStkInit()返回新的堆栈指针,OSTaskCreate() 将指针保存在任务的 OS_TCB 中。在 80C51 中要保存的内容有任务地址、PSW、ACC、B、DPL、DPH、R0-R7,还有由于函数重入引入的仿真栈指针?C_XBP。

OS_CPU_A.ASM 中要改写四个汇编语言函数:OSStarHighRdy(),OSCtxSw(),OSIntCtxSw(),OSTickISR()。OSStartHighRdy()主要功能是获取当前就绪的最高优先级任务的堆栈指针,然后将其寄存器内容恢复,并用 RETI 返回;OSCtxSw()主要功能是进行任务切换,保存相应寄存器内容;OSIntCtxSw()主要功能是进行中断任务切换,与 OSCtxSw()不同的是在中断任务切换中要根据不同处理器进行堆栈指针调整,以忽略由于中断而压入的多余内容,在 80C51 中 SP = SP-4;OSTickISR()主要功能是为系统提供一个时钟资源,在时钟中断到来时调用系统函数 OSTimeTick()为用户提供延时服务。

3 Keil C51 编译器环境下的一些问题:

本来原则上我们不用修改与处理器无关的代码,但是由于在 Keil C51 编译器环境下移植,还要根据编译器的特点进行一些改写。因为 Keil C51 缺省情况下编译的代码不可重入,而多任务系统要求并发操作导致重入,所以要在每个 C 函数及其声明后标注 reentrant 关键字。pdata、data 在 uC/OS-II 中用做一些函数的形参,但它同时又是 Keil C51 的关键字,会导致编译错误,可以把 pdata 改成 ppdata,data 改成 ddata 解决此问题。OSTCBCur、OSTCBHighRdy、OSRunning、OSPrioCur、OSPrioHighRdy 这几个变量在汇编程序中用到了,为了使用 Ri 访问而不用 DPTR,应该用 Keil C51 扩展关键字 IDATA 将它们定义在内部 RAM 中。另外,在 Keil C51 中尽量使用指定存储类型的指针(memory-specific pointer)而不使用一般指针(generic pointer),外部变量都申明为 xdata 类型,指向外部变量的指针改为 xdata 数据类型的指针,指向函数的指针改为 code 数据类型的指针,这样效率高。

4 资源和响应时间问题:

在嵌入式系统中引入操作系统会增加一些额外的代码空间,同时操作系统的任务调度和任务间的通信等也会占用一些内存。这些都会增加系统 ROM 和 RAM 容量,但随着半导体技术的发展,CPU 的速度和存储器容量不断增长,由于引入操作系统而增加的消耗,与其带来的便利性和稳定性相比是可以接受的。uC/OS-II 在 80C51 中移植后,如果完全应用,则占用 ROM 为 8K 左右,占用 RAM 为 5K 多,如果压缩使用,即将任务数由 63 个减为 16 个,不使用统计任务,则占用 ROM 为 5K 右,占用 RAM 为 2K 多。另外由于 uC/OS-II 中使用时钟节拍,在 80C51 中由定时器 T0 提供,因此 T0 不能为用户使用。

嵌入式系统中实时性是一个重要问题,中断响应时间是实时性的一个重要指标。因为任何系统在进入临界区代码之前都要关中断,执行完临界区代码之后再开中断,所以

中断延迟 = 关中断的最长时间 + 硬件调用中断服务程序的时间。对于前后台系统,

中断响应时间 = 中断延迟 + 保存 CPU 内部寄存器的时间。

对于占先式内核,

中断响应时间 = 中断延迟 + 保存 CPU 内部寄存器的时间 + 内核的进入中断服务函数的执行时间。

对于 uC/OS-II 来说,关中断的最长时间发生在消息序列查询函数 OSQQuery()中,内核的中断服务函数为 OSIntEnter(),以 24MHZ 的 AT89C51 在 Keil C51 环境下进行编译执行为例,函数 OSQQuery()中关中断的时间为 697us,硬件调用中断服务程序为 3 条指令,6 个指令周期,执行时间为 3us,保存 CPU 内部寄存器为 21 条指令,34 个指令周期,执行时间为 17us,内核的中断服务函数 OSIntEnter()的执行时间为 5us。则中断响应时间为 722us。可见应用操作系统增加的一部分中断响应时间是确定的。而在前后台系统中,中断响应时间是不确定的,它与后台程序的不同编写有关,(下转第 44 页)

```
    zgh: = inputbox('请输入职工号!','职工号','');
    clientdataset1. SetKey; //职工号在前面已经被设置为主键
    if not clientdataset1. FindKey([zgh])
    then showmessage('查无此人');
end;
```

保存,运行程序。

运行程序结果如图 6 所示。

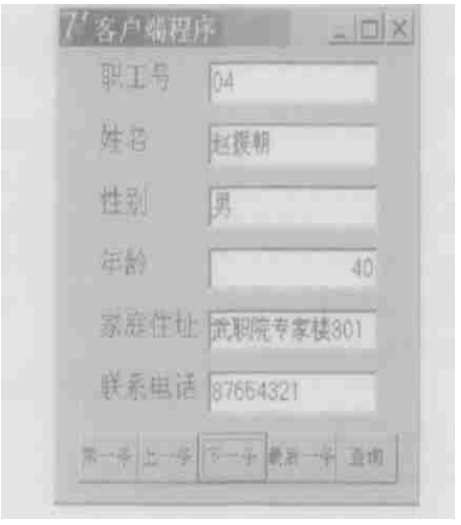

图 6 客户端程序运行结果

本程序经调试运行正常,至此,三层 C/S 客户-服务器数据库应用系统成功实现。

4 小结

三层数据库应用系统开发已经成为目前企业信息开发的主要趋势。在 Delphi 中主要通过 DataSnap 等技术来实现,这是 Delphi 的核心技术之一。本文以通俗易懂的方式讲述三层数据库应用系统开发技术,具有较强的操作性,对于想要开发此类数据库应用系统的读者一定会有所帮助,作者只是在这里抛砖引玉,相信读者一定会有更多启示。

参考文献:

[1] 张春林,马成勇,刘均. Delphi7 数据库系统设计与开发[M]. 北京:清华大学出版社

[2] 陈瑞,叶核亚. Delphi 程序设计实用教程[M]. 北京:电子工业出版社

[作者简介] 何定华,男,1972 年 9 月出生,1995 年毕业于中国地质大学计算机系,1995 年至今在武汉职业技术学院执教,现为武汉职业技术学院讲师。1997 年至 1999 年就读于华中科技大学研究生班,专业为计算机应用。出版教材 3 本。从事第一线教学与科研,主要方向,程序设计和中小型软件开发,熟悉《Visual Basic. NET》、《Visual C ++ 》、《ASP. NET》、《Delphi 程序设计》、《JAVA》等开发工具。

(收稿日期:2004-12-02)

(上接第 40 页)

如果系统复杂,为保证多个任务的实时性,前后台系统使用多个中断,中断嵌套的增多不仅增加了程序的复杂程度,而且带来了中断时间的不确定性。

所以对于简单的应用,使用前后台系统具有简单,占用空间小,实时性好等优点,但对于复杂系统,多任务占先式操作系统的实时性比前后台系统好,中断时间确定,而且将应用程序分为多个任务后,使应用程序层次化,简化程序的编写和维护。

5 应用

通过以上改写,在 Keil C51 上编译后,uC/OS-II 即可下载到 80C51 系列微处理器上运行,也可以在 Keil C51 上进行仿真运行。uC/OS-II 移植成功后,即可在此平台上搭建自己的应用程序,我们已在 Keil C51 环境下仿真运行 uC/OS-II 成功,并在其上建立多个应用程序进行测试,程序运行稳定。

6 总结

总之,在单片机上应用嵌入式操作系统后,由嵌入式操作系统来管理硬件和软件资源,大大方便了应用程序的设计和扩展,也使单片机应用于复杂的场合变得可能。

参考文献:

[1] JEAN J. LABROSSE 著,邵贝贝译. 嵌入式实时操作系统 uC/OS-II. 北京航空航天大学出版社,2003 年

[2] 马忠梅编著. 单片机的 C 语言应用程序设计. 北京航空航天出版社,2003 年

[作者简介] 秦绍华:山东大学信息与工程学院,研究生,主要研究方向为嵌入式系统。

陈涤:山东大学信息与工程学院,教授,主要研究方向为嵌入式系统,计算机自动测试与监控系统,现代通信系统等

(收稿日期:2004-09-29)