

嵌入式系统在微机继电保护中的应用

黄伟¹, 查智²

(1. 重庆工业职业技术学院自动化系, 重庆 400050; 2. 重庆大学自动化学院, 重庆 400030)

摘要: 探讨了基于 ARM9 和 $\mu\text{C}/\text{OS-II}$ 的微机继电保护系统设计方案。系统以 AT91RM9200 芯片为核心构成高速保护装置, 利用实时嵌入式操作系统 $\mu\text{C}/\text{OS-II}$, 不仅能较好地完成复杂的继电保护, 而且可保证装置的处理速度。讨论了基于 ARM9 和 $\mu\text{C}/\text{OS-II}$ 的系统硬件结构、开发特点以及系统软件流程。初步试运行效果表明, 所设计系统运行的稳定性和健壮性是满意的。

关键词: 微机继电保护; AT91RM9200; 嵌入式系统; 实时多任务系统; 串口通信

Application of embedded system in microprocessor-based protection

HUANG Wei¹, ZHA Zhi¹

(1. Department of Automation, Chongqing Institute of Poly-tech Occupation, Chongqing 400050, China;

2. College of Automation, Chongqing University, Chongqing 400030, China)

Abstract: The paper explores how to design microprocessor-based protection device based on ARM9 and $\mu\text{C}/\text{OS-II}$. AT91RM9200 is adopted as kernel of high speed protection device and $\mu\text{C}/\text{OS-II}$ is adopted as software platform. The protection system can not only achieve complicated protection task, but also improve the reaction speed. The paper discusses the hardware frame, and developing features as well as software flow of microprocessor-based protection system. The preliminary running effect of the system shows that it is satisfactory in running stability and robustness for the microprocessor-based protection function.

Key words: microprocessor-based protection; AT91RM9200; embedded operation system; real-time multi-task system; serial-port communication

中图分类号: TM77 文献标识码: B 文章编号: 1003-4897(2007)13-0011-04

0 引言

随着电力系统的日趋复杂, 继电保护装置对软、硬件提出了更高的要求。目前, 微机保护系统的处理器一般采用高性能单片机、数字信号处理器 DSP 和可编程逻辑器件 PLD 等来提高处理器性能。但是, 随着继电保护系统向多功能、智能化、可视化及网络化方向的发展, 这些芯片已经不能完全满足需求。因此, 在设计新的微机保护装置时, 有必要选用高性能、低成本的新型处理器。近年来, ARM 处理器凭借高性价比、低功耗等特点, 在嵌入式领域获得了广泛应用。而 $\mu\text{C}/\text{OS-II}$ 作为一种免费的、性能卓越的嵌入式操作系统, 为微机保护软件提供了统一的开发平台。本文结合工程实践, 对上述二者在微机继电保护中的应用作些讨论。

1 微处理器与实时嵌入式操作系统的选取

1.1 微处理器

微处理器选择 AT91RM9200, 它是基于 ARM920T 内核、ARM/Thumb 指令集的完整片上系统, 工作在 180 MHz 频率下其运算速度可高达 200 MIPS, 集成了丰富的应用外设及标准接口。该处理器包括一个高速的片上 SRAM 工作区和一个低延迟的外部总线接口 (EBI), 其内置控制器可用于控制同步 DRAM、突发访问模式 FLASH 存储器和静态存储器; 提供与 SmartMedia、Compact Flash 和 NAND Flash 的接口, 集成了 USB2.0 接口和媒体访问控制器 (MAC), 扩展了外部器件应用的种类, 使开发出来的产品可以工作在网络层, JTAG-ICE 接口、UART 通道 (DBGU) 和内嵌实时跟踪器提供的功能使受实时性限制的应用成为可能, 适合作为继电保护装置单片机解决方案的硬件平台。

1.2 嵌入式操作系统

嵌入式操作系统选择 $\mu\text{C}/\text{OS-II}$ 。早期这个系统叫做 $\mu\text{C}/\text{OS}$, 是 1992 年编写的嵌入式多任务实时操作系统, 后经过修改, 1999 才推出了 $\mu\text{C}/\text{OS-II}$, 它

符合 RTCA DO-178B 标准的认证,有足够的稳定性和安全性^[1]。μC/OS-II 是用 C 语言和汇编语言编写的。其中绝大部分代码都是用 C 语言编写的,只有极少部分与处理器相关的代码是用汇编语言编写的,只要稍加修改就可以把它移植到各类的嵌入式处理器上,因此可选择作为继电保护装置的软件开发平台。

2 保护装置的硬件系统设计

微机保护装置将待保护系统送来的电流、电压信号,经电流、电压互感器变换后产生低电压信号送入主模板^[2]。主模板内的 AT91RM9200 运行片内的保护软件,进行信号采样,完成各种数值运算、分析及处理,从而确定待保护系统的运行状态。如有故障,则由处理器经模板发出跳闸动作信号,以保护系统。同时通过各种通信方式将保护动作信息送入管理模板和上位机,记录和保存数据并报警。硬件系统采用模块化设计,可分为电流互感器/电压互感器模块、A/D 转换单元、开入/开出单元、人机交互单元、存储器单元、对外通信单元及电源模块等,硬件主体结构如图 1 所示。

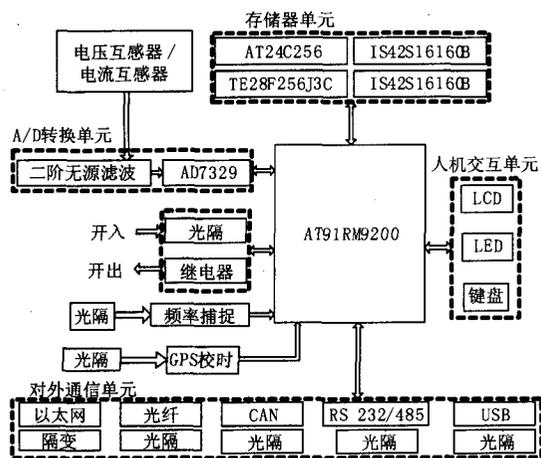


图 1 硬件主体结构图

Fig.1 Main architecture of hardware system

2.1 A/D 转换单元

A/D 转换是微机保护的关键功能,采样芯片选用 AD7329。该芯片是一款真正双极性、8 通道、低功耗、带符号位的 12 bit、1 MSPS 转换速率 ADC,输入电压范围达 ± 10 V,因此可以将输入噪声的影响降到最低,同时提供很高的直流和交流阻抗。

2.2 存储器单元

由于移植实时操作系统的需要,本系统采用两片容量较大的 SDRAM (IS42S16160B),每片 32 MB,构成 32 位的高速数据总线,用于存放程序代码和各

种数据。采用一片 32 MB 的 NOR FLASH (TE28F256J3C),用于固化操作系统、应用程序代码、操作过程中的事件信息和录波数据。另外,采用一片 32 KB 的 E²PROM 用于存放需要经常读写的保护定值。

2.3 通信单元

考虑到电力系统中大数据量和实时数据传输的需要,根据 AT91RM9200 微处理器的特点,本系统设计了 CAN、以太网、RS232/485、光纤、USB。CAN 控制器采用完全支持 CAN 总线 V2.0A 和 V2.0B 技术规范,通信速率为 1 Mb/s、SPI 接口的 MCP2510。考虑到光纤传输距离远、频带宽、发射天线小、保密性好及抗电磁干扰等优点,将光纤和以太网通信结合起来,充分发挥二者的优越性,可大大提高系统的应用范围和可靠性^[3]。系统采用 IP113A 作为以太网至光纤收发器,IP113A 是二端口(包括 TP 端口和 FX 端口) 10/100Mbps 以太网集成交换机,由一个二端口控制器和一个以太网快速收发器组成;遵守 IEEE802.3x 规则。

2.4 其它相关单元

开入回路和开出回路均由快速光电隔离芯片和逻辑编码电路组成,增加了电路的抗干扰性能。

用于人机交互的 LCD,采用 128×64 的点阵液晶显示屏,直接使用 PIO 口进行控制,键盘采用 3×3 的键盘电路,通过键盘查看系统参数和修改各种定值参数。系统各装置的保护、监控、事件顺序记录(SOE)、故障录波等功能对时间精度和同步性有较高要求,GPS 可以提供一个精确的时间坐标^[4],本系统采用 GPS 的 B 码授时,GPS 系统接受卫星时间信号,输出 IRIG-B 时间码系列,设备通过总线对时间进行同步。

3 软件系统设计

3.1 传统的前后台系统和实时多任务系统的比较

3.1.1 前后台系统

在前后台系统中,应用程序由后台运行,一般是一个无限循环,循环中调用相应的函数(子程序)完成相应的操作(称为后台行为或任务级);用中断来处理随机事件(称为前台行为或中断级)^[4],如图 2 所示。

这种程序规模较小、功能单一、智能化程度低,而且多是采用汇编语言编写的线性程序。它虽然具有代码精炼,某些关键操作执行效率高的优点,但是延长了编程时间,难以保证系统所有任务对实时性的要求。某个任务的响应时间取决于后台循环执行的时间。循环过程中,程序需要根据不同的状态和方式(中断服务子程序或用户的设置可能改变这

些状态和运行方式), 决定程序的走向, 所以每次循环的执行时间不同。某一任务的真正响应要等到循环顺序执行完前面所有的任务之后, 既不及时, 时间也不确定。同时, 程序的可读性很差, 调试困难, 维护也比较困难。

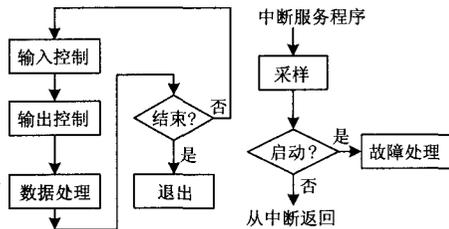


图2 传统微机保护程序机制

Fig.2 Traditional mechanism of microprocessor-based protection program

3.1.2 实时多任务系统

对于一个复杂的嵌入式实时系统而言, 当采用中断处理程序和一个后台主程序软件结构难以实时、准确、可靠地完成任任务时, 或存在一些互不相关的过程需要在一个计算机中同时处理时, 就需要采用实时多任务系统。随着应用的复杂化, 一个嵌入式控制系统可能要同时控制或监视很多外设, 有严格的实时响应要求, 需要处理任务比较多, 各个任务之间有多种信息需要实时传递, 如果仍采用原来的程序设计方法将存在两个问题: 一是中断可能得不到及时响应, 处理时间过长, 这对实时控制场合是不允许的, 对于网络通信而言则会降低系统整体的信息流量; 二是系统任务多要考虑各种可能出现的情况(尤其在任务使用共享资源时, 如果任务调度不当就可能导致系统死锁, 从而降低软件可靠性, 导致程序编写任务量成倍增加)。为降低系统的复杂性, 保证系统的实时性, 可维护性是必不可少的。

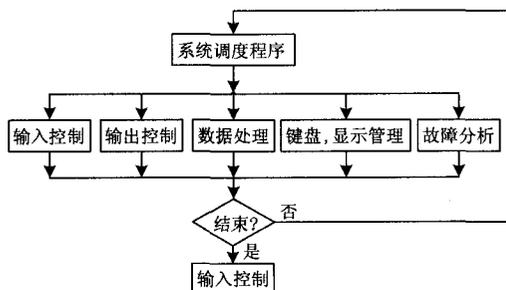


图3 使用RTOS的微机保护程序机制

Fig.3 Mechanism of microprocessor-based protection program using RTOS

实时多任务系统的实现必须有实时多任务操作系统的支持。操作系统主要完成任务切换、任务调

度、任务间的通信、同步、互斥、实时时钟管理以及中断管理。实时多任务系统实际上是由多个任务、多个中断处理过程和实时操作系统组成的有机整体。每个任务是顺序执行的, 并以并行性的方式通过操作系统完成, 任务间的相互通信和同步需要操作系统的支持。使用实时多任务系统的微机保护程序机制如图3所示。

继电保护对实时性要求较高, 任务较多, 故本文提出基于ARM9和 $\mu\text{C}/\text{OS-II}$ 的微机保护核心平台。
3.2 $\mu\text{C}/\text{OS-II}$ 性能分析及其在AT91RM9200中的移植

3.2.1 $\mu\text{C}/\text{OS-II}$ 性能分析

实时系统主要通过三个性能指标来衡量系统的实时性, 即响应时间(Response Time)、生存时间(Survival Time)和吞吐量(Throughput):

响应时间: 是实时系统从识别出一个外部事件到做出响应的的时间;

生存时间: 是数据的有效等待时间, 数据只有在这段时间内才是有效的;

吞吐量: 是在给定的时间内系统能够处理的事件总数, 吞吐量通常比平均响应时间的倒数要小一点。

其中响应时间是系统实时性最直观、最重要的指标。而系统响应时间与任务切换时间、中断延迟和调度延迟都有关系。 $\mu\text{C}/\text{OS-II}$ 的中断处理程序中不需要关中断, 它的关中断主要发生在一些原子操作和代码临界区保护的时候, 并且都非常短, 因此 $\mu\text{C}/\text{OS-II}$ 的中断延迟很短。

$\mu\text{C}/\text{OS-II}$ 是基于优先级的“可剥夺”式内核, 而且内核的调度算法非常简单, 因此 $\mu\text{C}/\text{OS-II}$ 调度延迟比较短且可以预测适应实时应用的要求。

关于上下文切换时间, 由于 $\mu\text{C}/\text{OS-II}$ 的任务都有单独的堆栈, 因而任务的切换操作非常简单, 由10多条CPU指令就可完成, 因此 $\mu\text{C}/\text{OS-II}$ 任务切换产生的延迟很小且是可以预测的。

3.2.2 $\mu\text{C}/\text{OS-II}$ 在AT91RM9200中的移植实现

$\mu\text{C}/\text{OS-II}$ 的移植条件是: 该处理器有堆栈, 有CPU内部寄存器入栈、出栈指令; 使用的C编译器支持内嵌汇编(inline assembly)或者该C语言可扩展, 可连接汇编模块, 使得关中断、开中断能在C语言程序中实现。

AT91RM9200符合其移植条件。 $\mu\text{C}/\text{OS-II}$ 的移植集中在3个文件, 头文件os_cpu.h, 汇编文件os_cup_a.s, C代码文件os_cpu_c.c。其中os_cpu.h主要包含编译器相关的数据类型的定义、堆栈类型的定义以及几个宏定义和函数说明。而os_cpu_c.c

中则包含与移植有关的 C 函数, 包括堆栈的初始化函数和一些钩子 (hook) 函数的实现。os_cpu_a.s 中则包含与移植有关的汇编语言函数, 包括开/关中断、上下文切换、时钟中断服务程序等^[5]。移植中关键的功能模块实现如以下所述。

移植工作包括以下几个内容: ①用#define 设置一个常量的值(OS_CPU.H); ②声明数据类型(OS_CPU.H); ③用#define 声明三个宏(OS_CPU.H); ④用 C 语言编写六个简单的函数(OS_CPU_C.C); ⑤编写四个汇编语言函数(OS_CPU_ASM)。

3.3 软件结构设计

在软件设计时, 尽量使软件和硬件脱离, 改变传统的嵌入式软件过多依赖硬件的模式。针对保护装置的实际情况, 把整个系统分成保护、A/D 采样、自检、显示、通信等几个任务, 划分原则是在考虑系统实时性和软件效率的前提下尽量减少任务间的耦合, 使功能清晰。任务确定以后, 赋予每个任务唯一的 ID 号, 并按照实时性要求对各个任务指定其优先级, 本设计中把保护任务设置成较高优先级。进入运行状态后, 在主任务中启动其他任务, 例如显示、A/D 采样、通信等。在这些任务中又可启动另外的任务。下面给出了主程序的任务创建和资源分配的部分代码。

```
OSTaskCreate (TaskStart, (void *)0, (void *)
&TaskStartStk [TASK_STK_SIZE-1],0); //创建主任务
OS_STK Main_Stack[STACKSIZE*8]={0,}; //Main_Task
堆栈
void Main_Task(void *Id); //Main_Task
#define Main_Task_Prio 12
OS_STK Key_Scan_Stack[STACKSIZE]={0,};
//Key_Scan_Task 堆栈
void Key_Scan_Task(void *Id); //Key_Scan_Task
#define Key_Scan_Task_Prio 56
OS_STK Lcd_Fresh_Stack[STACKSIZE]={0,}; //LCD 刷
新任务堆栈
void Lcd_Fresh_Task(void *Id); //LCD 刷新任务
```

设计中的任务调度机制采用优先级调度模式: CPU 被分配给最高优先级任务, 如果几个任务优先级相同, CPU 就被分配给最先进入就绪队列的任务。如果一个任务在执行过程中, 另外一个优先级更高的任务进入就绪状态, 则原来正在进行的任务进入就绪状态, 新任务进入运行状态。

μC/OS-II 最多可支持 56 个任务。实现各种功能的保护程序按照功能划分成了一个一个的任务, 每一种任务实现一种功能, 任务之间相互独立, 只通

过实时操作系统 RTOS(Real-Time Operating System) 机制交换信息。这从根本上保证了软件的可靠性和实时性。

4 结束语

随着 ARM 处理器技术的不断发展和应用的不断推广, ARM 处理器在电力系统中必将得到更广泛的应用。而嵌入式实时操作系统 RTOS 在微机保护的应用, 为实现复杂的保护功能提供了可能。文中讨论的基于嵌入式 AT91RM9200 微处理器的微机保护装置已完成现场调试, 初步试运行效果表明, 该设计是成功的, 受到用户方的认可, 有望得到大力推广。

参考文献

- [1] 任哲. 嵌入式操作系统 μC/OS-II[M]. 北京: 北京航空航天大学出版社, 2005.
REN Zhe. Embedded Operation System μC/OS-II[M]. Beijing: Beijing: University of Aeronautics and Astronautics Press, 2005.
- [2] 王金, 陈明军. 基于 DSP 的输电线路距离保护装置设计[J]. 浙江电力, 2006, (2): 9-11.
WANG Jin, CHEN Ming-jun. Design of Transmission-line Distance Protection Device Based on DSP[J]. Zhejiang Electric Power, 2006, (2): 9-11.
- [3] 黄海波. 基于以太网的光无线通信系统的设计与实现[J]. 电子技术应用, 2004, (2): 40-42.
HUANG Hai-bo. Design and Realization of Light-wireless Communication System Based on Ethernet[J]. Application of Electronic Technique, 2004, (2): 40-42.
- [4] 谢培元. 基于 μC/OS-II 和 DSP 的微机保护方案研究[D]. 武汉: 华中科技大学, 2004.
XIE Pei-yuan. The Research of Microprocessor-based Protection Based on μC/OS-II and DSP[D]. Wuhan: Huazhong University of Science and Technology, 2004.
- [5] 夏彬彬, 耿光仁. 实时嵌入式操作系统 μC/OS-II 在 ARM9 上的移植应用[J]. 桂林电子工业学院, 2006, (3): 30.
XIA Bin-bin, GEN Guang-ren. Application of Porting RTOS μC/OS-II to ARM9[J]. Guilin Institute of Electronic Technology, 2006, (3): 30.

收稿日期: 2007-02-09; 修回日期: 2007-03-28

作者简介:

黄伟 (1964-), 女, 讲师, 研究方向为嵌入式系统在电力系统自动化系统控制中的应用;

E-mail: huangwei051001@126.com

查智 (1981-), 男, 硕士研究生, 主要研究方向为微机继电保护, 嵌入式操作系统。