

多任务机制在继电保护装置中的应用

陈杰¹, 刘亚平¹, 张旭光²

(1. 许昌继电器研究所, 河南 许昌 461000; 2. 哈尔滨电业局, 黑龙江 哈尔滨 150001)

摘要: 提出了一种简单实用的基于多任务机制的继电保护装置的单片机系统程序结构。介绍了具体的实现方法, 并给出了一个在变压器保护装置中的应用实例。

关键词: 多任务机制; 单片机系统; 继电保护装置

中图分类号: TM774

文献标识码: A

文章编号: 1003-4897(2001)08-0049-03

1 引言

近年来,采用单片机系统构成的继电保护装置在电力系统中得到了广泛的应用。这种传统的单片机系统的程序结构通常是基于单任务机制构成的。这种机制具有简单直观易于控制的优点。然而由于程序缺乏灵活性,而且维护性和可移植性较差,也难以实现复杂的功能。随着电力系统自动化程度的提高,对继电保护装置的要求也越来越高。例如在综合自动化系统中往往要求装置集保护、控制、测量、遥信等自动化功能于一身。为了更好更快地设计和完善继电保护装置,必须对传统的单任务机制进行改进。本文介绍一种基于单片机系统的简单、实用的多任务机制。

2 多任务机制的原理与实现

多任务机制是现代操作系统才具有的突出优点。例如:Windows就是一种基于多任务机制的操作系统。在继电保护装置的设计中引入多任务机制,可以有效地改善程序结构,实现复杂的功能,满足复杂系统的要求。而且采用多任务机制的程序结构后,各任务之间相关性少,耦合较松(基本没有耦合),便于模块化处理。可读性和可移植性也好。同时各个任务相互独立、目标明确、结构简单,便于程序员分工协作,模块化编程和调试。对程序的扩充也更方便。不用更改原有结构,只需增加要扩充的部分即可。

目前,在继电保护装置中也开始应用了一些实时多任务系统。这些实时多任务系统大多是购买的国外公司的商业软件。一般应用在嵌入式处理器系统和DSP数字信号处理器系统中。这类实时多任务系统性能优越,功能强大。彻底掌握后应用非常方便。但也有一些缺点。例如价格较昂贵,大多在

数万元至数十万元之间。设计开发的成本太高。而且这类软件规模较大,要详细了解深入掌握,并达到运用自如的难度较大,时间也较长。因此这类软件一般被一些经济实力和实力较强的厂家应用在中高压系统的继电保护装置中。而在电力系统中应用面最广、数量最多的仍然是中低压系统的继电保护装置。这些装置基本上都是由各种各样的单片机系统构成的。而单片机运算处理能力的有限要求应用的多任务调度算法必须简洁。因此,在单片机系统中引入简单、实用的多任务机制才是解决问题的最有效的途径。

在多任务机制下,处理器要完成的工作被划分为功能不同的子任务。各个子任务相对独立,按照不同优先级在任务调度程序的安排下轮流切换交替工作。而处理器的运行时间则被划分为许多小的时间片,由任务调度程序按优先级的不同分配给各个子任务。各个子任务分别在自己的时间片内占用处理器,从而造成微观上分时轮流工作,宏观上并列运行的多任务效果。

在单片机系统中实现多任务机制的方法是比较多的。方法之一可以利用单片机的时钟中断方式来实现“时间片”。首先将单片机要完成的系统功能分解为若干个相对独立的子任务。然后设置一个定时器时钟中断来提供系统时钟,将CPU的运行时间划分为若干个时间片。再构造一个任务调度表和一个系统任务指针。而系统程序则由一个主循环程序和一个时钟中断程序构成。各个子任务由系统调度程序启动,在时钟中断程序中执行。执行完毕后返回主程序中。各个子任务的执行频率则不同。执行频率表现为它们的优先级。优先级越高的子任务在单位时间内执行的次数越多。而那些对执行频率无要求的任务模块则放在主循环程序中运行。这样各子任务在自己的时间片内执行。通过合理地设计时间片

的大小和各任务的优先级,就可以充分利用“时间片”的机制来提高程序运行效率。这种利用“时钟中断”来实现多任务机制的关键在于合理地将系统功能分解为各个子任务。而划分任务的原则在于各个子任务的优先级。

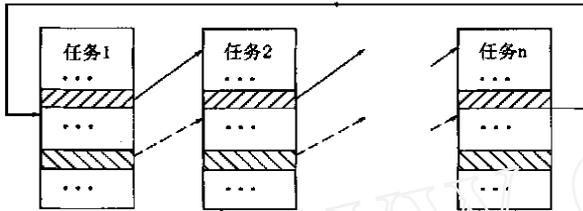


图1 任务切换

方法之二是利用单片机系统的“堆栈”来实现多任务机制。同第一种方法一样,也要将单片机完成的功能分解为若干个相对独立的子任务。但区别在于划分任务的原则不是优先级,而是完全按照要实现的功能的不同来划分子任务。各个子任务的地位是完全一样的。另外需要在单片机的RAM中开辟一段空间作为“堆栈”。每个任务都有一段堆栈区,用来保存任务间切换时“跳走”和“返回”的地址指针和每个子任务内部程序执行时需要保存的堆栈数据。然后再构造一个“任务切换”调度程序。CPU上电/按钮复位后,进行必要的硬件初始化和软件初始化。然后程序就从安排好的任务列表中的第1个子任务开始执行。执行一段程序后,调用“任务切换”调度程序。在“任务切换”调度程序的控制下主动释放控制权。进入第2个任务中执行,执行一段程序后,由第2个任务调用“任务切换”调度程序,再进入第3个任务中执行.....依次类推进行到最后一个任务中,执行一段程序后,调用“任务切换”调度程序。由调度程序控制,再返回到第1个任务中上次“跳走”的位置接着继续执行。再执行一段程序后,再进入第二个任务中,从第二个任务上一次“跳走”的位置接着执行.....依次类推,循环执行、轮流切换,如图1所示。可见,这种多任务机制的核心可以归纳为“循环切换、轮流工作、分时处理”,也即“微观分时、宏观并行”。这种机制实现起来也非常简单。整个任务切换都是利用“堆栈”由“任务切换”调度程序自动控制的。各个子任务完全独立,只需在隔一段调用一次“任务切换”调度程序即可。占用的资源也非常少。只需用到几个寄存器、几百个字节长的RAM空间(堆栈区)即可。而“任务切换”调度程序也只有短短的十几条语句。在任何一种单片机系统上都可以轻松实现。而且相对于第一种方法而

言,这种方法更易于实现,编程设计更方便。

3 应用实例

这里用一个采用第二种“任务切换”的方法构成多任务机制的变压器保护装置为实例来详细说明。该变压器保护装置是综合自动化变电站中的间隔层IED设备。装置集保护、控制、遥测、遥信功能于一身。该装置采用一片Intel公司的80C196KC作为处理器。

该装置的任务划分为六个子任务。分别为(1)保护计算任务:完成采样数据滤波、计算、保护逻辑判别功能;(2)串口/网络通讯任务:完成与后台监控之间的通讯功能;(3)菜单管理任务:完成人机接口菜单操作的功能;(4)液晶显示任务:完成液晶显示处理功能;(5)动作报告处理任务:完成保护动作报告处理功能;(6)键盘处理任务:完成键盘处理功能。另外有一个每20/12ms一次的定时器中断程序来完成A/D采样中断服务功能。

程序中专用寄存器使用3个,其余为通用寄存器。SP为堆栈指针。堆栈空间占用为6×64个字节。开放的中断有两个:串口中断和定时器采样中断。子程序StartTask的作用为启动多任务机制,初始化任务列表。子程序TaskSwitch为“任务切换”调度程序。汇编语言程序实例如下:

```

.....
DSEG      AT      Ram           ;RAM堆栈定义
MaxTask   equ     6             ;任务总数
lTaskList equ     64            ;单位长度
TaskIndex : dsb    2 *MaxTask + 2 ;任务列表
TaskList  : dsb    1 *MaxTask   ;堆栈区

.....

RSEG      AT      20h           ;寄存器定义
TaskAx    dsl     1             ;暂存寄存器
task_total : dsb    1           ;任务总数寄存器
task_current : dsb  1           ;当前任务寄存器

.....

CSEG      AT      2080H
START:

.....
ld        ax, # MainTaskList
lcall    StartTask

MainTaskList :
dcw      ProtTask      ;保护计算任务
dcw      CommuTask     ;网络通信任务
dcw      ManageTask    ;菜单管理任务
dcw      LcdTask       ;液晶显示任务

```

```

dcw   OPdataTask   ;动作报告处理任务
dcw   KEYscanTask  ;键盘处理任务
dcw   0
      .....
StartTask :          ;初始化任务列表
      ld   cx , # TaskIndex
      ld   dx , # TaskList + 1TaskList
      clrb task . total

StartTask1 :
      ld   sp , dx
      push [ ax ] +
      st   sp , [ cx ] +
      add  dx , # 1TaskList
      incb task . total
      cmpb task . total , # MaxTask
      jc   StartTask2
      cmp  zero , [ ax ]
      jne  StartTask1

StartTask2 :
      clrb task . current
      addb wsr , task . current , # 048h
      ld   sp , TaskIndex
      ret

TaskSwitch :        ;任务切换调度
      ldbz TaskAx , task . current
      add  TaskAx , TaskAx
      st   sp , TaskIndex [ TaskAx ]
      incb task . current
      cmpb task . current , task . total
      jnc  TaskSwitch1

```

```

      clrb task . current
TaskSwitch1 :
      addb wsr , task . current , # 48h
      ldbz TaskAx , task . current
      add  TaskAx , TaskAx
      ld   sp , TaskIndex [ TaskAx ]
      ret
      .....
ProtTask :
      .....
      lcall TaskSwitch
      .....
CommuTask :
      .....
      lcall TaskSwitch
      .....

```

4 结束语

本文将现代操作系统的多任务思想引入继电保护装置的设计中,提出了一种全新的程序结构。实践证明:这种结构模块清晰,调度灵活,简单实用。在系统规模日益扩大的情况下,具有较高的应用价值。

收稿日期: 2001-03-12

作者简介: 陈杰(1967-),男,工程师,主要从事继电保护装置研究与开发; 刘亚平(1966-),男,助工,主要从事继电保护装置研究与开发; 张旭光(1968-),男,工程师,主要从事继电保护及自动化的应用与运行。

Application of multi-task mechanism in relaying protections

CHEN Jie¹, LIU Ya-ping¹, ZHANG Xu-guang²

(1. Xuchang Relay Research Institute, Xuchang 461000, China; 2. Harbin Power Supply Bureau, Harbin 150001, China)

Abstract: It proposes a simple and practical program configuration of a single-chip processor in a relaying protection based on a multi-task mechanism in this paper. The detailed implementation approach is introduced and a practical case in a transformer protection is presented.

Keywords: multi-task mechanism; single-chip processor system; relaying protection

(上接第53页)

Setting calculation for differential protection in SIEMENS 7UT51 digital differential protection relay

XIE Wen-tao

(No. 2 Power Plant of Yangzhou, Yangzhou 225131, China)

Abstract: The character of the differential protection in SIEMENS 7UT51 digital differential protection relay is introduced in this paper. The setting calculation principle of the protection is analyzed in detail and it is used in the setting calculation of the service power system in No. 2 Power Plant of Yangzhou.

Keywords: digital differential protection relay; differential protection; setting calculation