

μC/OS- II 在中频感应炉数据采集系统中的应用

Application of μC/OS- II in Data Acquisition System for Medium Frequency Induction Furnace

李 哲 吴玉香

(华南理工大学自动化学院,广东 广州 510641)

摘 要: 针对目前中小企业生产的中频感应炉 MFIF 缺少配套监控设备的现状,为提高中频感应炉的自动化水平,提出一种以 DSP 处理器 TMS320F2812 和实时嵌入式操作系统 μC/OS- II 为基础的中频感应炉数据采集方案。μC/OS- II 通过将独立的应用程序封装到每个任务中,并按链表的形式进行排序和优先级比较,实现数据的采集、存储和显示,异常数据的识别和处理,以及 RS-232、以太网和 CAN 总线三种通信模式。该方案不仅提高了系统的可靠性和实时性,而且有利于系统的进一步改进与升级。

关键词: μC/OS- II TMS320F2812 RS-232 CAN 总线 数据采集

中图分类号: TP274 **文献标志码:** A

Abstract: Aiming at the current status of lacking auxiliary monitoring equipment for medium frequency induction furnace (MFIF) in small and medium scale enterprises, and in order to enhance the automation level of these furnaces, the data acquisition strategy based on DSP TMS320F2812 and real-time embedded operating system μC/OS- II is proposed. By μC/OS- II, the individual application program is packaged into each task, ranked and compared the priority in the form of chain table for implementing data acquisition, storage and display, as well as identifying and processing the abnormal data, and providing three of the communication patterns including RS-232, Ethernet and CAN bus. This strategy improves the reliability and real-time capability, and also makes system upgrade and improvement easier.

Keywords: μC/OS- II TMS320F2812 RS-232 CAN bus Data acquisition

0 引言

中频感应炉 MFIF (medium frequency induction furnace) 是指频率高于 50 Hz、小于 10 kHz 的感应加热炉。它是利用电磁感应原理,使处于交变磁场中的金属材料内部感应产生电流,从而把材料加热直至熔化的一种电热设备。中频感应炉在各种金属及其合金的熔炼、透热和热处理方面都得到广泛的应用,具有加热快、铁损少、效率高、节约能耗和易于实现自动化等优点^[1]。通过生产现场的实地考察,发现许多中小企业生产的中频感应炉缺少配套的监控和保护设备。为了提高中频感应炉的自动化水平,需要对多种信号进行同时采集和监测,以便实现性能分析、过程控制和故障处理等目的。

目前,常用的数据采集装置多采用单片机实现,软件多采用单任务顺序机制,这使得系统不仅处理能力有限,而且存在稳定性差的问题^[2]。随着工业自动化的不断发

展,对软件的实时性和设计的灵活性要求越来越高,传统的前后台软件设计方法已不能适应这样的需求。因此,嵌入式操作系统的应用显得尤为迫切^[3]。本文详细介绍了 μC/OS- II 嵌入式实时操作系统在中频感应炉数据采集系统中的应用,即以 μC/OS- II 在 TMS320F2812 (以下简称 F2812) 上的移植为基础,实现了中频感应炉数据采集系统的改进。

1 系统硬件设计

系统的硬件结构如图 1 所示。

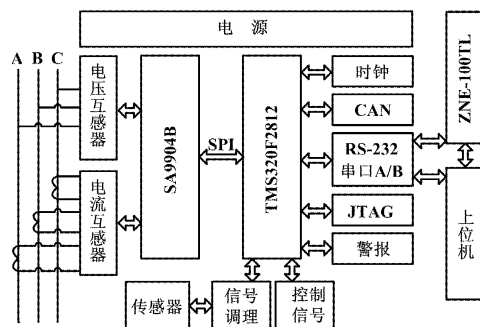


图 1 系统硬件结构

Fig. 1 Hardware structure of the system

科技型中小企业技术创新基金资助项目(编号:09C26214412155)。

修改稿收到日期:2010-07-22。

第一作者李哲,男,1986年生,现为华南理工大学控制理论与控制工程专业在读硕士研究生;主要从事数据采集与处理、嵌入式系统开发和智能控制应用方面的研究。

系统硬件部分主要由一块数据采集板和一台普通的 PC 机组成。数据采集卡包括以电能计量芯片 SA9904B 为核心的电量采集电路、传感器、信号调理电路、F2812、时钟电路、JTAG 接口电路、报警电路、串行通信电路、CAN 总线通信电路、以太网通信电路和电源电路等。其中,以太网通信电路由串口 B 和 RS-232 转以太网模块 ZNE-100TL 组成^[4]。温度传感器选用性能稳定、应用广泛的负温度系数热敏电阻;压力检测部分选用压阻式压力传感器。考虑到严格的接口保护,这里选用环氧灌封电流型电压互感器 TV16 和环氧灌封精密电流互感器 TA12-200 进行隔离采样。互感器技术参数如表 1 所示。

表 1 技术参数

Tab.1 Technique parameters

名称	电压互感器	电流互感器
型号	TV16	TA12-200
变比	1 000:1 000	2 000:1
输入电流	2.5 mA	0~5 A
输出电流		0~2.5 mA
最大不失真输出电压	2 V	
取样电阻	400 Ω	800 Ω
取样电压		2 V
相移	≤30'	≤5'
工作频率	20 Hz~20 kHz	20 Hz~20 kHz
抗电强度	2 kV AC/1 min	6 kV AC/1 min

试验用中频感应炉为三相 220 V 交流供电,功率为 540 kW。三相电压直接通过接线端接入;三相电流通过一级电流互感器降为 0~5 A,再接入数据采集卡上的二级电流互感器。

2 μC/OS-II 在 F2812 上的移植

要实现嵌入式实时操作系统在中频感应炉数据采集系统中的应用,首先要实现 μC/OS-II 在 F2812 上的移植。μC/OS-II 移植的主要工作是根据 F2812 的特点编写与处理器相关的代码和配置文件^[5],包括:os_cpu.h、os_cpu_a.asm、os_cpu_c.c、dsp_ram.cmd 等。uC/OS-II 移植的要点包括:定义函数 OS-ENTER-CRITICAL 和 OS-EXIT-CRITICAL;定义函数 OS-TASK-SW,执行任务切换;定义 OSCtxSW,实现用户级上、下文切换;定义 OSIntCtxSW,实现中断级任务切换;定义 OSTickISR,时钟节拍 ISR;定义 OSTaskStkInit,初始化任务的堆栈。

选用 TI 公司的 DSP 芯片必须要编写或者修改链接命令文件(CMD 文件),这是移植过程的重点之一。由于 F2812 没有专门的操作系统来定位执行代码,且

每个用户设计的 DSP 系统的硬件配置也不尽相同,因此,需要设计人员自己编写 CMD 文件,并定义代码的存放和加载位置^[6],一般只需要在 TI 提供的 CMD 框架上进行扩展即可。以下是针对 μC/OS-II 部分的 CMD 文件配置。

```

RESET; //复位中断向量
origin = 0x3FFFC0, length = 0x000002
MULSTACK; //多任务堆栈段
origin = 0x000100, length = 0x000200
MAINSTACK; //主堆栈段
origin = 0x000300, length = 0x000060
IDLESTACK; //空闲任务堆栈段
origin = 0x000360, length = 0x000040
.reset : > RESET, PAGE = 0, TYPE = DSECT
CpuTimer1RegsFile : > CPU_TIMER1, PAGE = 1
CpuTimer2RegsFile : > CPU_TIMER2, PAGE = 1
MulStackFile : > MULSTACK, PAGE = 1
MainStackFile : > MAINSTACK, PAGE = 1
IdleStackFile : > IDLESTACK, PAGE = 1
    
```

移植代码编写和存储器配置完成后,进行多任务调度和内核通信调试,调试成功后即可添加用户任务。

3 用户任务创建与调度

3.1 用户任务创建

μC/OS-II 的移植是为了实现对多任务的管理。μC/OS-II 可以管理 64 个任务,其中包括两个已经被系统所占用的任务,即空闲任务和统计任务,其余 62 个任务可根据需要来创建。根据数据采集系统的功能需求,创建的用户任务包括:串行外设接口通信处理任务(SPITask1)、A/D 转换处理任务(ADCTask2)、异常处理任务(TroubleManageTask3)、警报任务、串口通信任务(SCIATask5)、CAN 总线通信任务(CanTask6)和以太网通信任务(SCIBTask7)。

串行外设接口通信处理任务是指 DSP 通过 SPI 定时接收 SA9904B 上传的四项电参数,计算出实际的测量值,然后将测量值与预设值进行比较,判断电路是否出现异常状况;A/D 转换处理任务是指 DSP 通过采样中断方式处理 A/D 转换通道采集的温度和压力信号,并将测量值与预设值进行比较,判断电路是否出现异常状况;异常处理任务是指在系统工作异常时,执行相应的保护措施;警报任务是指当系统工作异常时发出声光报警提醒;串口通信任务、CAN 总线通信任务和以太网通信任务是指通过三种不同的方式将数据发送给上位机。

3.2 用户任务调度

系统运行时,程序首先进入 main() 主函数,完成 DSP 和 μC/OS-II 初始化工作后,调用 OSTaskCreate() 函数,创建一个优先级为 0 的任务(TaskStart),并在内部任务就绪表内标记该任务为就绪状态(我们按照任务的重要性和实时性来分配优先级,优先级数值越低,任务的优先级越高);然后进入 TaskStart 任务,创建 7 个用户任务;最后调用 OSStart() 函数,启动多任务环境,把 CPU 控制权交给操作系统,开始多任务调度。

多任务启动流程如图 2 所示。

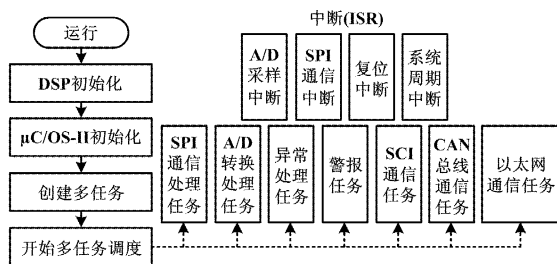


图 2 多任务启动流程图

Fig.2 Flowchart of multitask start-up

main() 函数代码如下。

```
void main (void)
{
    DINT;
    EALLOW; // 保护寄存器
    InitSysCtrl(); // 系统初始化
    InitPieVectTable(); // 初始化中断向量表
    InitPieCtrl();
    EDIS;
    OSInit(); // 初始化 μC/OS-II
    OSTaskCreate ( TaskStart, ( void * ) 0, ( void * )
        &TaskStartStk[0], 0);
    OSStart(); // 启动多任务环境
}
```

TaskStart() 在创建用户任务的同时,将任务挂起(除了优先级为 10 的任务),此时只有 SPITask1 处于就绪态,则 OSStart() 函数根据就绪表求出优先级最高的就绪任务为 SPITask1,然后调用 OSStartHighRdy() 运行该任务。需要特别说明的是,在操作系统中的中断向量选择 RTOSINT。该向量为 F2812 专门为实时操作系统提供的中断向量^[8]。

TaskStart() 函数代码如下。

```
void TaskStart (void * data)
{
    data = data; // 防止编译出现警告
```

```
EALLOW;
IER |= 0x8000;
// 使能实时操作系统中断向量 RTOSINT
PieVectTable. RTOSINT = &OSCtxSw;
// 中断向量指向任务调度函数
EDIS;
OSTaskCreate ( SPITask1, ( void * ) 0, ( void * )
    &TaskStk[0][0],10); // 任务 1 创建
OSTaskCreate ( ADCTask2, ( void * ) 0, ( void * )
    &TaskStk[0][0],12); // 任务 2 创建
OSTaskCreate( TroubleManageTask3, ( void * ) 0,
    ( void * ) &TaskStk[0][0],14); // 任务 3 创建
OSTaskCreate ( AlarmTask4, ( void * ) 0, ( void * )
    &TaskStk[0][0],16); // 任务 4 创建
OSTaskCreate ( SCITask5, ( void * ) 0, ( void * )
    &TaskStk[0][0],18); // 任务 5 创建
OSTaskCreate ( CanTask6, ( void * ) 0, ( void * )
    &TaskStk[0][0],20); // 任务 6 创建
OSTaskCreate ( SCIBTask7, ( void * ) 0, ( void * )
    &TaskStk[0][0],22); // 任务 7 创建
OSTaskSuspend( 12);
OSTaskSuspend( 14);
OSTaskSuspend( 16);
OSTaskSuspend( 18);
OSTaskSuspend( 20);
OSTaskSuspend( 22);
for(;;) {}
// 任务需写成无限循环的形式,以免破坏系统堆栈空间
```

进入 SPITask1() 任务后,将优先级高的任务(SPI-Task1)挂起,恢复优先级为 12 的任务(ADCTask2),此时优先级 12 为最高优先级,所以下一步执行 ADC-Task2()。以此类推,建立链表顺序,使得任务调度清晰、简单。因为优先级 22 为最低优先级,所以循环最后只需将优先级为 10 的任务恢复,而不需要将优先级为 22 的任务挂起,这样就可以使高优先级的任务抢占 CPU。

4 结束语

μC/OS-II 嵌入式实时操作系统是开放源码且得到实际验证的软件平台,而 DSP 处理器具有强大的 32 位 RISC 性能。试验表明,基于 μC/OS-II 以及 DSP 控制器的中频感应炉数据采集系统方案能将测量、控制、通信和显示等功能集于一体,有效提高系统的可靠性和实时性,并充分发挥 DSP 高效能的优点。

(下转第 86 页)

开始一帧图像的截取,是非阻塞的。而图像截取是否完毕由 VIDIOCSYNC 判断。

④ `int v4l_grab_sync (v4l_device *), ioctl (vd->fd, VIDIOCSYNC, &(vd->frame_current)` 完成截取图像的同步工作,它在截取一帧图像后调用,返回表明一帧图像截取结束。

利用 V4L 编写图像采集程序与 Makefile 文件,在宿主机上交叉编译图像采集程序后查看编译好的可执行文件如下。

```
[hh@localhost image] $ file image
image: ELF 32-bit LSB executable, ARM, version 1, dynamically linked (uses shared libs), for GNU/Linux 2.6.27, not stripped
```

从上面的文件属性可以看出,得到的可执行文件是针对本文的嵌入式操作系统,将该可执行文件复制到开发板上运行,可以看到如下信息。

```
[root@linpo /]# ./image
open success!
set_norm success!
init success!
size = 475136
memory map success!
zc3xx: probe 2wr ov vga 0x0000
get image success!
syn success!
img address 0x402d8000
```

运行图像采集程序的可执行文件,则在开发板的指定目录下得到了像素为 320×240 的图像。通过 Linux 下程序运行时间算法计算可知,整个程序运行时间约为 0.07 s,图像采集部分运行时间约为 0.04 s,系统具有较好的实时性。

3 结束语

本文在 LPC3250 嵌入式 Linux 平台上利用 V4L 编程实现了基于中星微 ZC301B 芯片的 USB 摄像头的图像采集。实验证明,本文采用的图像采集方案具有较好的实时性。在实际应用中,用户也可以根据实际需求更改应用程序,通过 FrameBuffer 设备在 LCD 上实时显示图像;或将采集到的图像数据传入图像处理算法中,并借助 OpenCV 计算机视觉库对采集到的图像进行进一步的处理。

参考文献

- [1] 常慕,洪健,李钟慎. 嵌入式机器视觉系统的图像采集及显示技术[J]. 自动化仪表,2010,31(3):16-18.
- [2] 汪庆年,张广栋,李桂勇. 嵌入式 Linux 视频采集系统的设计与实现[J]. 微计算机信息,2009,25(11):39-40.
- [3] 吴晴,周健. 嵌入式图像采集系统的设计与实现[J]. 电子测量技术,2007,30(6):89-92.
- [4] 郭剑,赵建. 嵌入式 Linux 的图像采集与显示[J]. 现代电子技术,2006,29(7):129-131.
- [5] 桂电-丰宝联合实验室. 基于 LPC3250 的嵌入式 Linux 系统开发[M]. 北京:电子工业出版社,2010:8-10.
- [6] 电脑编程技巧与维护杂志社. Linux 编程典型实例解析[M]. 北京:清华大学出版社,2009:316-320.
- [7] 陈俊宏. Embedded Linux 嵌入式系统原理与实务[M]. 北京:中国铁道出版社,2004:184-224.
- [8] Li Quanxi, Liu Peiqian, Li Changyou. Research on embedded video monitoring system based on Linux[C] // Proceedings of the 2009 International Conference on Computer Engineering and Technology, Singapore, 2009:478-481.
- [9] Tian Huan, Li Hui, An Huiyao. Special effect drivers for camera on an open source platform[C] // Proceedings of the 2009 Second International Conference on Information and Computing Science, Manchester, England, UK, 2009:339-334.

(上接第 82 页)

在系统设计中引入 $\mu\text{C}/\text{OS}-\text{II}$ 是 DSP 应用复杂化后编程方式的一种改变趋势,可处理多任务的复杂系统,大大缩短了系统开发周期,便于程序的编写和调试。 $\mu\text{C}/\text{OS}-\text{II}$ 较高的可移植性也有利于系统的进一步改进和升级。目前,系统正在测试当中,任务调度正常,运行效果良好。

参考文献

- [1] 姜士林,赵长汉. 感应加热原理与应用[M]. 天津:天津科技翻译出版公司,1993:10-11.
- [2] 赵刘强,林永君,马良玉,等. 基于 $\mu\text{C}/\text{OS}-\text{II}$ 嵌入式系统的数据采集系统[J]. 微计算机信息,2010(5):62-66.

- [3] 潘超军,陈刚. $\mu\text{C}/\text{OS}-\text{II}$ 在 TMS320F2812 上的移植[J]. 江苏电机工程,2005,24(4):33-34.
- [4] 高阳,潘宏侠,吴升,等. 基于 DSP 的电力设备远程监测分析系统[J]. 电力自动化设备,2010,30(1):127-131.
- [5] 左丽霞,卢山. 基于 DSP 和 $\mu\text{C}/\text{OS}-\text{II}$ 的馈线终端装置设计方案[J]. 华东交通大学学报:自然科学版,2009,26(5):64-69.
- [6] 韩丰田. TMS320F281xDSP 原理及应用技术[M]. 北京:清华大学出版社,2009:279-280.
- [7] 王忠凯,赵磊. $\mu\text{C}/\text{OS}-\text{II}$ 任务调度研究[J]. 山东理工大学学报:自然科学版,2009,23(2):30-35.
- [8] 孙丽明. TMS320F2812 原理及其 C 语言程序开发[M]. 北京:清华大学出版社,2008:462-465.