

基于嵌入式思想的PID数字控制系统设计

沈兆军

(盐城工学院 信息学院, 江苏 盐城 224051)

摘要:介绍了用8051单片机实现PID数字控制系统的结构,简要描述了PID控制原理,着重介绍了系统的硬件结构和嵌入式(sEOS)任务软件设计思想,并给出了相应的原理框图和软件原型。该系统具有实时性好、控制速度快、稳定性强等优点。

关键词:PID控制;单片机;sEOS系统

中图分类号:TP368.1

文献标识码:A

文章编号:1671-5322(2009)01-0053-04

PID控制器是控制系统中应用最广泛的一种控制器,可实现对温度、压力、流量等变量的实时调节控制。传统的PID控制器多为模拟调节器,主要通过电路硬件来实现。随着计算机的发展,特别是现代嵌入式微处理器的大量应用,将原来主要由硬件实现的功能通过软件来实现,即用计算机软件实现PID控制算法。这样具有更大的灵活性、可靠性和更好的控制效果,因此把它称作数字PID控制器^[1-2]。本文主要介绍用简单多任务嵌入式操作系统sEOS(The simple operating embedded system),结合8051单片机实现的PID数字控制系统的一种通用结构。

1 PID控制规律

1.1 模拟PID调节器的控制规律

$$u(t) = k_p[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt}]$$

采用PID调节器的控制系统如图1所示。其中, $r(t)$ 是设定值, $c(t)$ 是实际输出值,控制偏差 $e(t) = r(t) - c(t)$, $u(t)$ 是输出控制量。 K_p 为比例系数, T_i 为积分系数, T_d 为微分系数。比例、积分和微分控制作用是关联的关系,参数可以分别调节,也可以只采用其中一种或两种控制规律。

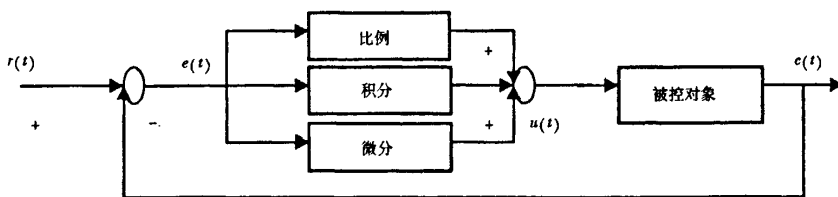


图1 PID控制原理框图

Fig. 1 Frame of PID control principle

1.2 数字PID控制器^[3]

将模拟PID控制规律进行适当的变换后,用计算机软件来实现PID控制和校正,就是数字PID控制。由于计算机控制实际是一种采样控制,它根据采样时刻的偏差值计算控制量。所以原式中积分项和微分项不能直接使用,必须进行

离散处理。方法是以 T 作为采样周期, r 作为采样序号,则离散采样时间 rT 对应着连续时间 t (即令 $t = rT$),用求和的形式代替积分,用增量的形式代替微分,可得到离散的PID表达式:

收稿日期:2008-11-23

作者简介:沈兆军(1972-),男,江苏盐城人,讲师,硕士,主要研究方向为现代通信与嵌入式系统。

$$u_k = k_p [e_k + \frac{T}{T_i} \sum_{j=0}^n e_j + \frac{T_D}{T} (e_k - e_{k-1})]$$

$$\text{或: } u_k = k_p e_k + K_I \sum_{j=0}^k e_j + K_D (e_k - e_{k-1})$$

其中, e_k 为第 k 次采样时刻输入的偏差值; K_p 为比例常数; K_I 为积分常数; K_D 为微分常数。如果采样周期取得足够小, 上两式的近似计算可获得足够精确的结果, 离散控制过程与连续控制过程十分接近。上面的控制算法给出了全部控制量的大小, 因此称为全量式或位置式 PID 控制算法。下面给出用 C 语言实现的该 PID 算法模型, 其中函数 PIDCalc () 的返回值即为输出控制量 U_k , 供主控制程序调用。

```
typedef struct PID
{ double SetPoint; // 设定目标 Desired Value
  double Proportion; // 比例常数 Proportional
} PID;

Const
double Integral; // 积分常数 Integral Const
double Derivative; // 微分常数 Derivative

Const
double LastError; // Error[k-1]
double PrevError; // Error[k-2]
double SumError; // 累积偏差值 Sums of Errors

PID; /* PID 计算部分 */
double PIDCalc( PID *pp, double NextPoint )
{ double dError, Error;
  Error = pp->SetPoint - NextPoint; // 偏差
  pp->SumError += Error; // 积分
  dError = pp->LastError - pp->PrevError; // 当前微分
  pp->PrevError = pp->LastError;
  pp->LastError = Error;
  return (pp->Proportion * Error // 比例项
    + pp->Integral * pp->SumError // 积分项
    + pp->Derivative * dError); // 微分项
}
```

2 单片机控制部分

2.1 硬件电路设计

硬件电路从功能模块上来分主要有: 主机电路、数据采集电路、键盘显示电路、控制执行电路

以及掉电保护电路等。

硬件结构框图如图 2 所示。

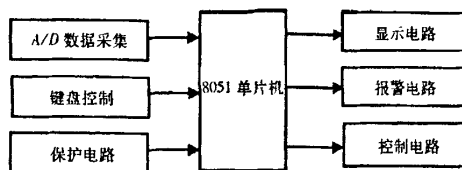


图 2 系统硬件结构框图

Fig. 2 Frame of system hardware structure

2.1.1 主机电路的设计

主机可选用 PHILIP 公司的 P89C51RD2 系列单片机来实现, 该器件每个机器周期为 6 个时钟周期标准组成, 因此运行速度可达传统 8051 的 2 倍, 运算速度快, 控制功能完善。该系列单片机是 80C51 微控制器的派生器件, 是采用先进 CMOS 工艺制造的 8 位微控制器, 指令系统与 80C51 完全相同。

2.1.2 I/O 通道的硬件电路的设计

(1) 数据采集电路 就本系统来说, 需要实时采集被控系统数据, 然后经过 A/D 转换为数字信号, 送入单片机中的特定单元以待处理。

(2) 显示电路 实时显示采集到的被控系统数据, 便于了解系统控制情况。可采用 LED 或 LCD 显示。

(3) 键盘电路 可利用键盘实现对控制参数的设定或修改, 以改善控制效果。

(4) 控制电路 通过 PID 算法得到的控制量要经由单片机输出去控制被控对象。控制电路主体部分通常可用双向可控硅触发电路组成, 其触发信号由单片机通过 PID 算法输出, 从而实现闭环自动控制。

(5) 报警和掉电保护电路 为了安全运行, 该系统对被控量进行上限和下限报警处理。另外, 为防止系统因为意外掉电而丢失数据, 可设计掉电保护电路。

2.2 软件设计——简单嵌入式操作系统 sEOS

在嵌入式实时控制领域, 过去一般把整个软件设计成一个前后台系统。应用程序是一个无限循环, 循环中调用相应的函数完成相应的操作, 这部分是后台行为; 中断服务程序处理异步事件, 这部分是前台行为。这对于单任务的控制是比较适合的。但是在控制任务较多的时候, 这种传统的软件架构就很难在精确的时间间隔执行某个任务

X()。这就需要有一个适当的操作系统来进行多任务的合理调度,同时降低系统开发难度,使得实时应用程序的设计和扩展变的容易,不需要大的改动就能增加新的功能。而 sEOS 就是一个用于开发中小型项目比较合适的简单嵌入式操作系统。

sEOS (The simple operating embedded system), 可视为一种简单嵌入式操作系统,象其它许多嵌入式系统一样,可以有效地提供周期性执行单任务的功能,同时亦可建立多任务状态系统。在多状态系统中,每个状态可能会调用一个或多个函数,不同状态之间存在转换的规则,并且系统在状态之间转换时,可能会调用一个或多个函数。状态之间转换时,采用 C 语言的 switch - case 语句实现,简单明白,易于控制。根据控制状态之间转换规则的特性不同,多状态任务系统常见的有两种模式:

(1)多状态时间驱动式(timed) 这种模式状态之间的转换只取决于时间的推移。例如,系统可能有状态 A 开始,重复执行 FunctionA() 10 s;然后可能转入状态 B 且保持 5 s,重复执行 FunctionB();随后可能又返回状态 A,如此循环。常见的交通灯控制系统就是采用这种模式。

(2)多状态输入/时间驱动式(input/timed)

这种模式很常见,状态之间的转换将取决于时间的推移及系统的输入。例如,只有当某个系统产生输出后 Ns 内收到某个特定的输入,系统才会在状态 A 和状态 B 之间转换。我们熟悉的全自动洗衣机的控制系统就遵循这种模式。

根据本系统的设计要求,采用多状态输入/时间驱动式的 sEOS,主要可分配以下几个任务:

- ①键盘输入任务;(state:KEY)
- ②A/D 数据采集任务;(state:AD)
- ③实时显示当前数据任务;(state:LED)
- ④PID 算法处理任务;(state:PID)
- ⑤输出控制信号任务;(state:OUT)
- ⑥异常报警任务;(state:ALARM)

当然,在具体的控制系统任务中,可根据使用要求对任务进行增减,只需增减相应的任务函数,使用非常方便。

sEOS 实时操作系统应用任务管理函数原型如下:

```
void main(void)
{ X_Init(); //初始化任务 X()
  sEOS_Init_Timer2(n); //设定时间间隔,每
```

隔 Ts 调用中断(ISR)刷新函数

```
while(1) //任务在超级循环中由中断 ISR 调用
{
  sEOS_Go_To_Sleep(); //系统运行在节电模式,提高 CPU 工作效率
}
```

该系统实现任务管理的原理,是采用定时器 2 的溢出所产生的中断调用中断服务例程(ISR),这里设为 X()。在中断服务例程(ISR)中,实现所需的任务调度及任务间的转移(或多状态间的转移),ISR 将轮流地被定时器溢出所调用。定时器 2 具有 16 位自动重装载功能,能产生精确的时间间隔 Ts,时间间隔可根据需要设定,一般可设定几十 ms。

中断服务例程(ISR)函数 X()原型:

```
void X(void) interrupt 5
{ TF2 = 0; //定时器 2 溢出标志位,需手动置位
  X_Update(); //任务调度函数,以执行具体任务,是系统软件设计的重点
}
```

根据本系统的具体任务分配要求,任务调度函数 X_Update()原型设计如下:

```
void X_Update(void)
{ //每隔 nms 刷新调用
  switch(state)
  { case KEY:
    { KEY(); //调用键盘子程序,进行系统参数设定或修改
      state = AD; break; //系统设定好后转入状态 AD,进行 A/D 数据采集
    }
    case AD:
    { AD(); //调用 A/D 转换子程序
      state = LED; break; //转入状态 LED,实时显示当前数据
    }
    case LED:
    { LED(); //调用显示子程序
      if(data >= H || data < L) state = AL-
RAM; //数据超限,转状态 ALRAM 报警
      else state = PID; break; //转入状态 PID,进行
```

PID 运算

```

}
case PID:
{ PID(); //调用 PID 算法子程序
state = OUT; break; //转入状态 OUT, 输出控制信号
}
case OUT:
{ OUT(); //输出控制信号
state = AD; break; //转入状态 AD, 进行新一轮 A/D 数据采集
}
case ALARM:
{ ALARM(); //数据超限报警信号
AD();
if( data > = H || data < = L) state = ALARM;
else state = AD; break; //状态正常时退出
}
}
}

```

在系统中嵌入 sEOS 可以把整个程序分成许多任务, 每个任务相对独立, 任务间可相互转移。在单片机系统中嵌入 sEOS, 提高了系统的可靠性, 并使调试程序变得简单, 同时也增强了系统的可扩展性和产品开发的可持续性, 很适用基于 8051 的嵌入式系统的需要。

3 实例分析及结束语

实例分析: 该控制系统用于某系统温度(如炉温, 水温等)控制过程可描述如下(硬件模型可参考图 2, 可在 A/D 数据采集电路前加温度传感器及放大电路; 程序主要模块设计可参考本文软件设计说明)。

原理说明: 当信号采样周期到达时, 由温度传感器检测系统实际温度, 经过放大电路将温度值转换为 0~4V 的电压信号, 再由 A/D 转换芯片将电压值转换成数字量, 经单片机采集后与设定温度进行比较, 根据偏差信号进行 PID 控制运算, 得出相应控制输出量, 将控制输出量输出到执行机构, 从而控制系统温度, 使系统的实际温度向着给定温度变化并最终达到给定温度。如需要改变系统设定温度值, 可通过键盘控制设定。系统时刻检测实际温度值, 出现异常时启动蜂鸣器报警。此系统能够实现现场温度的显示和按键控

制, 如需实现远程的温度显示与控制, 单片机亦可将测得的温度值通过串口传递给上位机, 使用户能够随时对温度进行监控。

本文针对 PID 数字控制系统模型, 提出了一种基于 8051 单片机的设计方案。本设计应用嵌入式微处理器, 用 sEOS 设计思想来实现 PID 控制规律。利用 sEOS 提供的系统特征, 占用很低的 CPU 负载, 但是却既灵活又有用。可以简化多任务程序设计, 满足多个任务的时间要求, 大大降低了开发难度, 轻松地完成前后台编程方法难以完成的任务, 同时程序的可读性和可维护性也提高了。该设计方案非常适用于智能家电系统控制, 诸如冰箱、空调器温度调节, 也可用于一般工业系统中进行温度、压力、流量、转速等参数控制, 具有广泛的实用性。

主程序流程如图 3 所示, sEOS 任务调度流程如图 4 所示。

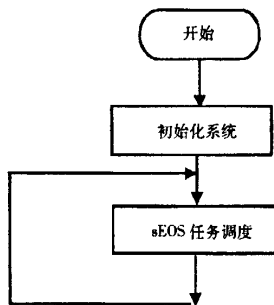


图 3 主程序流程图

Fig. 3 Flow chart of main program

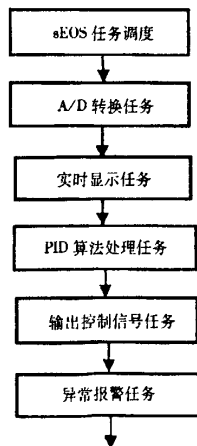


图 4 sEOS 任务调度流程图

Fig. 4 Flow chart of sEOS task

(下转第 65 页)

Based on Proteus in Single Chip Computer System Design

ZHANG Hai-liang

(College of Mechanical and Electrical Engineering, East China Jiaotong University, Jiangxi Nanchang 330013, China)

Abstract: Hardware design and software design are two parts in single chip computer system design. The traditional way is that software can be tested after hardware design is completed. If hardware circuit can not meet the demands, it must be modified. After that, software will be tested on the modified hardware again. Proteus is the simulation software for Single Chip Computer System Design. Hardware circuit design and software design can be easily modified to meet the goal if hardware circuit cannot meet the demands. The design described in the paper of the electric count system's correction and utility is tested by using proteus software to design and simulate.

Keywords: Proteus; Single Chip Computer; Hardware circuit; simulation

(责任编辑:张英健;校对:沈建新)

(上接第 56 页)

参考文献:

- [1] 刘金琨. 先进 PID 控制[M]. 北京:电子工业出版社,2004.
- [2] 吴国垣,李东海. 多变量系统分散 PID 控制器设计[J]. 清华大学学报:自然科学版,2004,44(11):1567-1570.
- [3] 龚运新. 单片机 C 语言开发技术[M]. 北京:清华大学出版社,2006.
- [4] 王田苗. 嵌入式系统设计与实例开发[M]. 北京:清华大学出版社,2003.
- [5] Wolf W. Hardware-Software Co-Design of Embedded Systems Proceedings[J]. the IEEE,2004,82(7):123-126.

The PID Digital Control System Design Based on Embedded Thought

SHEN Zhao-jun

(School of Information Engineering Yancheng Institute of Technology, Jiangsu Yancheng 224051, China)

Abstract: This paper has introduced the structure of PID digital control system realized with 8051, described PID control principle briefly, presented the hardware structure and embedded sEOS task design thought of this system, and given corresponding principle block diagram and software prototype. This system has the control speed with good real time to be quick, the advantages such as stability winner.

Keywords: PID control; Single chip computer; sEOS system

(责任编辑:张英健;校对:沈建新)