

Proteus 软件在自动控制系统仿真中的应用

陈朝元, 鲁五一

(中南大学信息科学与工程学院, 湖南 长沙 410075)



摘要: 基于 Proteus 强大的微处理器仿真功能和丰富的元件仿真模型, 提出了新的用于自动控制系统的仿真方法。例程中使用常用的 AT89C52 芯片作为微处理器, 再加上外围电路设计了 PID 控制电路仿真原理图, 用 C 语言编程后进行了详尽的系统仿真试验, 可以直观地观测到系统仿真的控制效果。这种基于 Proteus 软件的仿真方法在自动控制系统的教学演示和实际设计等方面中具有很大的辅助作用。

关键词: Proteus; 自动控制系统; PID; 仿真

中图分类号: TN46; TP391.9 **文献标识码:** A **文章编号:** 1004-731X (2008) 02-318-03

Application of Proteus in Simulation on Automatic Control System

CHEN Chao-yuan, LU Wu-yi

(The School of information science and engineering, Central South University, Changsha 410075, China)

Abstract: Based on the powerful function of simulation on the MCU and series of simulation models for devices in Protues, a new way was provided to simulate the automatic control system. In the example, the schematics were figured out by using one commonly-used chip named AT89C52 as MCU, together with some other devices. After programming with C language, detailed tests were given on the simulation on the system, resulting in a direct and clear outcome. This way is very helpful to demonstrate the automatic control system in classroom and practical design of circuit.

Key words: Proteus; automatic control system; PID; simulation

引言

在自动控制系统设计期间, 系统仿真扮演着非常重要的角色, 通过对系统的仿真, 可以大大提高设计效率。在常规的自动控制系统仿真中, 使用最多的仿真软件应当是 MATLAB, MATLAB 仿真功能强大, 可以应用于各个学科和领域。但是我们在实际设计中, 往往是应用微处理器作为控制核心, 编写相应的控制算法进行控制, 编写程序使用的大多都不是 MATLAB 语言, 而是基于微处理器的象 C 语言之类的编程语言。这就涉及到 MATLAB 编程与微处理器编程的不一致性, 仿真时也不能直观的反映出微处理器在系统中的作用。Proteus 软件是来自英国 Labcenter electronics 公司的 EDA 工具软件, 它除了有和其它 EDA 工具一样的原理图编辑、PCB 自动或人工布线及电路仿真的功能外, 其革命性的功能是, 他的电路仿真是互动的, 针对微处理器的应用, 还可以直接在基于原理图的虚拟原型上编程, 并实现软件源码级的实时调试。这样, 利用 Proteus 软件, 就可以进行微处理器编程而实现各种控制算法的仿真, 直观的观测到对被控制对象的控制效果。

1 Proteus 软件简介

EDA 工具的应用非常广泛, 按主要功能或主要应用场

合, 分为电路设计与仿真工具、PCB 设计软件、IC 设计软件、PLD 设计工具及其它 EDA 软件。

Proteus 软件是来自英国 Labcenter electronics 公司的 EDA 工具软件, 具有其它 EDA 工具如 PROTELL 99 等软件的常规功能: 原理图编辑、PCB 自动或人工布线及电路仿真的功能外, 此外它最大的特点是可以对众多 MCU 进行多种编程语言源码级的仿真^[1], 包括 PIC, 8051, AVR 等类型微处理器, 有大量的与 MCU 接口的仿真模型和 Laplace 变换仿真模型, 可以直观的实现交互式仿真调试, 如液晶显示器, 键盘等。

利用 Proteus 中 Laplace 变换仿真模型, 即可对自动控制系统中的被控制对象进行仿真, 再利用 MCU 仿真模型以及相关的外部接口模型, 即可通过对 MCU 编程而实现控制算法的调试, 最终实现整个自动控制系统的软硬件仿真和调试。并且可以在此基础上直接进行电路 PCB 板的设计, 大大提高了系统设计的效率。

2 Proteus 仿真设计

下面所设计的例程中电路、编程和调试都非常简单直观, 旨在能快速体会到 Proteus 软件在自动控制系统仿真中的直观性和高效性。

2.1 Proteus 仿真模块的介绍

Proteus 软件的 ISIS 工具是用于编辑电路原理图和电路仿真, 它除了包括大量常用的分立元件和集成块元件外, 还包括许多类型的微控制器及其外围接口器件的仿真元件, 此

收稿日期: 2006-10-31

修回日期: 2006-12-12

作者简介: 陈朝元(1974-), 男, 湖南道县人, 硕士生, 研究方向为 EDA 技术, 自动控制和系统工程; 鲁五一(1957-), 男, 湖南长沙人, 教授, 研究方向为嵌入式技术, 自动控制和测量技术。

外还包含全面的 Laplace 转换模块。

在这里的例程中主要使用到 MCS-51 系列的 AT89C52 芯片, AD 和 DA 转换芯片, 逻辑门电路芯片, 运算放大器芯片, Laplace 转换模块以及一些分立元件。

2.2 仿真结构框图

自动控制系统中最为常见的是 PID 控制, 本文就以数字 PID 控制为例来探讨, 其控制系统框图如图 1 所示。

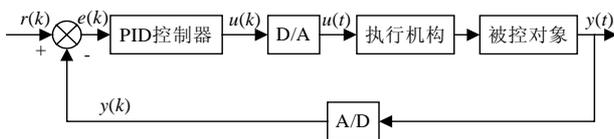


图 1 数字 PID 控制框图

在 Proteus 仿真原理图中, $r(k)$ 设定值可在程序中给定, PID 控制算法由 AT89C52 微处理器的程序实现, D/A 转换器选用 DAC0808, 执行机构由运算放大器仿真 (实际应用中一般还需要功率放大), 被控对象则由 Laplace 转换模型中的一阶系统仿真, A/D 转换器选用 ADC0808。此外为了直观的象 MATLAB/SIMULINK 仿真那样分析阶跃响应, 还增加了“ANALOGUE ANALYSIS”仿真图形显示, 可以将被控对象的输出曲线显示出来, 便于观测和分析。具体的用 ISIS 设计的仿真原理图如图 2 所示。

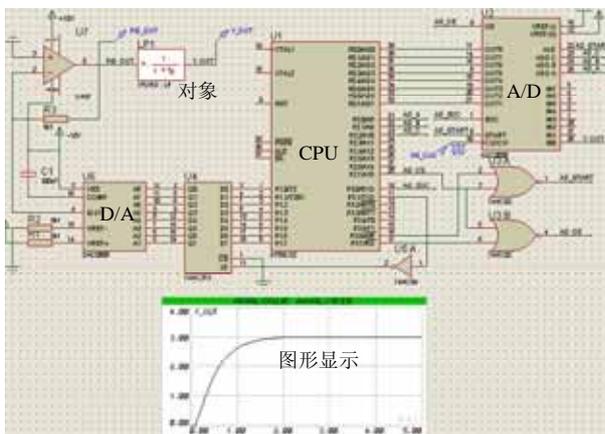


图 2 ISIS 软件编辑的仿真原理图

2.3 微控制器程序设计

此仿真系统的微控制器程序使用 Keil C51 工具编写和编译^[2], 程序中主要包括定时器初始化函数 Init_Timer(), PID 初始化函数 PIDInit(), PID 计算函数 PIDCalc(), 定时中断处理函数 timer() 以及 AD 转换函数 Read_AD() 等。

定时器中断处理程序中先完成反馈信号采集和数字化, 再对误差进行 PID 计算, 再将输出的数字信号通过 DA 转换输出。这部分程序是整个程序设计的主要部分, 而其中的核心所在即是 PID 输出计算的编程, 下面对其原理和编程进行详细说明。

这里使用常用的位置式 PID 控制算法^[3], 对模拟 PID 控

制器离散化后即可得到表达式^[4]

$$u(k) = K_p \{ e(k) + \frac{T}{T_i} \sum_{j=0}^k e(j) + \frac{T_d}{T} [e(k) - e(k-1)] \} \text{ 或}$$

$$u(k) = K_p e(k) + K_i \sum_{j=0}^k e(j) + K_d [e(k) - e(k-1)]$$

式中的 K_p 为比例系数, T 为采样周期, T_i 为积分时间常数, T_d 为微分时间常数, $K_i = K_p T / T_i$, $K_d = K_p T_d / T$ 。 $e(k)$ 为第 k 次采样时刻系统误差值。根据第二个公式可得到位置式 PID 控制算法的程序流程图如 3 所示^[5]。

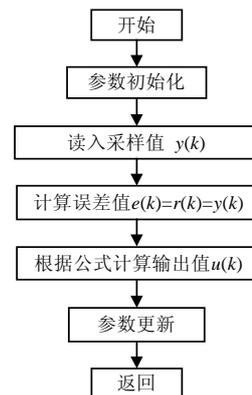


图 3 位置式 PID 控制算法程序流程图

下面列出 PID 计算函数 PIDCalc() 的源程序^[6]:

```
//PID 输出计算函数
int PIDCalc(struct PID *pp, int NextPoint)
{
    int 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 / 10 // 比例项, 因为 PID 参数扩大了 10 倍, 此处要缩小还原
            + pp->Integral * pp->SumError / 10 // 积分项, 因为 PID 参数扩大了 10 倍, 此处要缩小还原
            + pp->Derivative * dError / 10); // 微分项, 因为 PID 参数扩大了 10 倍, 此处要缩小还原
}
```

在仿真过程中, 还可以根据实际要求, 对 PID 输出值进行限幅处理。

2.4 元件属性设置

原理图中各个元件都有其具体的属性设置, 大多按缺省设置即可, 这里影响仿真的设置主要是 CPU 属性设置和 Laplace 模块属性设置。

在此仿真过程中, 选择了在 CPU 属性中直接指定 HEX 程序文件的仿真方法, 而没有采用设置较复杂的 Proteus 与 Keil C51 联调的模式, 主要是为了让大家能尽快了解这种新型的自动控制系统仿真方法。在原理图编辑中, 用鼠标右键单击 AT89C52 将其选中, 再在其上面用鼠标左键单击, 就弹出“Edit Component”编辑窗口, 将“Program File”属性

中填入编译后生成的PID.HEX文件名即可。

原理图中使用的是常用的一阶低通Laplace模型,可以通过修改其静态放大系数和时间常数来观测不同的仿真结果。

3 仿真与分析

完成上面仿真原理图和程序设计后,用鼠标单击“ANALOGUE ANALYSIS”仿真图形上面的标题栏,弹出放大的仿真图形窗口,通过单击窗口左下角的图标,在弹出窗口设定“Stop time”为合适的停止时间(此例中设定为5秒),再通过单击窗口左下角的图标,在弹出窗口中的“Probe P1:”下拉菜单选择Y_OUT,最后通过单击窗口左下角的图标,软件将进行仿真运算并显示输出曲线。通过修改程序中的PID参数,即可观测系统的阶跃响应曲线图。下面是调整PID参数而测得的不同阶跃响应曲线图。

当程序中控制设定值为3.0V,采样周期为100mS, PID参数设定为 $K_p=2.0$, $K_i=0$, $K_d=0$ 时的阶跃响应曲线图如图4所示。

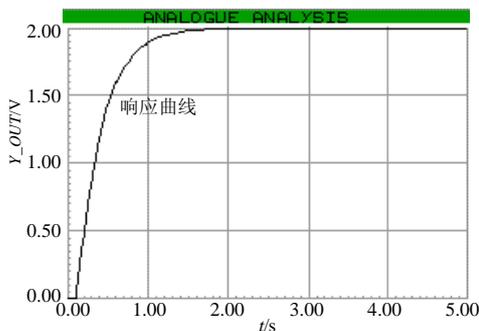


图4 仿真输出图之一

当程序中控制设定值为3.0V,采样周期为100mS, PID参数设定为 $K_p=2.0$, $K_i=0.2$, $K_d=0$ 时的阶跃响应曲线图如图5所示。

图4只有比例作用,可见系统无法消除控制误差,图5加入了积分作用,很好的消除了控制误差,我们还可以加入微分作用,可以观看到改进效果不是很明显,表明对于此类对象一般仅需PI控制即可达到较好的控制效果。

在此例程基础上,可以更换一阶Laplace模型为二阶或



(上接第303页)

(3) 本文假设产品故障发生服从指数分布,其余风险因素的发生服从均匀分布,而工程实际情况往往更为复杂,因此在针对模型的数据分析和处理方法上还需作进一步研究。

参考文献:

[1] GJB4355-2002. 备件供应规划要求 [S]. 中国人民解放军总装备部批准, 2003.

[2] 王其藩. 系统动力学 [M]. 北京: 清华大学出版社, 1994.

[3] Janne Huiskonen. Maintenance Spare Parts Logistics: Special

高阶模型,通过调节 K_p 、 K_i 和 K_d 三个参数,观测仿真结果。

此外还可以通过修改程序来研究象模糊PID等先进PID算法,可以更进一步的体会到Proteus软件在自动控制系统仿真中的优势。

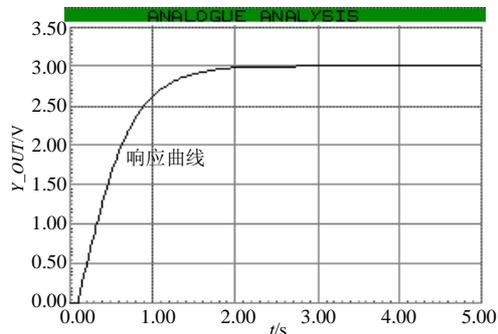


图5 仿真输出图之二

4 结论

Proteus软件强大的微处理器仿真功能和丰富的元件仿真模型为自动控制系统提供了新的仿真平台,不再是传统方框图式的仿真,而是直接与微处理器编程及控制电路仿真结合起来,既可以完成控制电路的软硬件仿真,又完成了系统的控制效果仿真。使用这种仿真方法可以大大提高设计各种数字控制器的效率,对于自动控制系统的教学演示和实际设计都具有很大的辅助作用。

参考文献:

[1] 倪升跃. Proteus——一种集单片机模拟和SPICE分析于一身的软件 [J]. 电子世界, 2004, (12): 38-39.

[2] 黄凤绚. Proteus与Ultra Edit、Keil的联合使用 [J]. 无线电, 2005, (7): 36-37.

[3] 班立权, 鲁五一. PID参数自整定算法及仿真 [J]. 计算技术与自动化, 2005, 24(3): 12-14.

[4] 刘金琨. 先进PID控制MATLAB仿真 [M]. 第二版. 北京: 电子工业出版社, 2004: 10-22.

[5] 黄文梅, 杨勇, 熊桂林, 等. 系统仿真分析与设计——MATLAB语言工程应用 [M]. 长沙: 国防科技大学出版社, 2001: 226-228.

[6] 马忠梅, 马岩, 张凯, 等. 单片机的C语言应用程序设计 [M]. 北京: 北京航空航天大学出版社, 1997: 161-164.

Characteristics and Strategic Choices [J]. Int. J. Production Economics (S0925-5273), 2001, 71(1): 125-133.

[4] Anis Chelbi, Daoud Ait-Kadi. Spare provisioning strategy for preventively replaced systems subjected to random failure [J]. Int. J. Production Economics (S0925-5273), 2001, 74: 183-189.

[5] A A Ghobbar, C H Friend. Sources of intermittent demand for aircraft spare parts within airline operations [J]. Journal of Air Transport Management (S0969-6997), 2002, 8: 221-231.

[6] Dekker Rommert, Moritz Fleischmann, Karl Inderfurth, et al. Reverse Logistics: Quantitative Models for Closed-Loop Supply Chains [M]. Berlin: Springer, 2004.