

计算机控制系统的软件抗干扰技术

李会娟

(青海大学 化工机械系, 青海 西宁 810016)

摘要: 工业现场存在着各种强烈干扰, 很容易导致计算机控制系统不稳定甚至无法正常工作, 这就可能造成程序错误或运算错误, 导致设备失控和误动作。因此, 提高计算机控制系统稳定性是一个必须认真对待的问题。针对计算机控制系统的软件抗干扰措施作出了比较细致的探讨。

关键词: 计算机控制系统; 软件; 抗干扰; 可靠性

中图分类号: TP273+.5 TP802+.6

文献标识码: A

Software anti-jamming techniques for computer control system

LI Hui Juan

(Department of Chemical Engineering and Machinery, Qinghai University, Xining 810016, China)

Abstract: The computer control system working in industry scene is under bad production environment and specially intense interferences. There are kinds of disturbances in industry scene, which cause the system stable, possibly creates the program error or the operation mistake. Therefore, how to enhance the stability is must earnestly be treated. In this article, we had carefully discussed the software anti-jamming measures.

Key words: computer control system; software; anti-jamming; reliability

计算机控制系统就是利用计算机(微型机、工业PC、控制器、单片机等)来实现生产过程自动控制的系统^[1, 2]。计算机控制系统是微电子技术、计算机技术和网络技术与自动化技术的典型结合和综合实现, 近几年来, 在农业、工业、教育、国防等各个领域得到了广泛应用, 并已经成为我国现代化建设的必要环节和发展的必然趋势。在这些应用过程中, 如何防止外界对控制系统的干扰, 确保计算机控制系统安全、可靠地正常运行是一个非常重要的问题。计算机控制系统在各个环节的设计和配置具有完备功能的前提下, 一方面使系统紧凑、布局合理, 另一个重要问题就是稳定性, 否则再完善的系统和功能设计, 抗干扰性能不好, 也将前功尽弃。

计算机控制系统工作环境恶劣、干扰频繁, 如不对其加以抑制, 将会影响到控制系统的可靠性和稳定性。计算机控制系统的抗干扰措施具体可分为硬件和软件

两方面, 在硬件措施的基础上, 对于窜入系统的少数干扰, 要由软件来消除干扰引起的副作用^[2, 3]。软件抗干扰以其设计灵活、不需要增加硬件资源、成本低等优势得到广泛采用。

在计算机控制系统的输入输出通道中, 采用某种计算方法对通道的信号进行数字处理, 以削弱或滤除干扰噪声, 这就是数字滤波的方法。这种廉价而有效的软件程序滤波在计算机控制系统中被广泛采用。而对于那些可能穿过通道而进入CPU的干扰, 可采取指令冗余、软件陷阱以及程序运行监视等措施来使CPU恢复正常工作。下面分别对数字滤波方法、指令冗余技术和软件陷阱技术分别进行介绍和分析。

1 数字滤波方法

由于工业生产的现场环境非常恶劣, 各种干扰源很多, 计算机控制系统通过输入通道采集到的数据信号, 虽经硬件电路的滤波处理, 但仍会混有随机干扰噪

声。因此,为了提高系统性能,达到准确的测量与控制,一般情况下还需要进行数字滤波。

数字滤波,即计算机系统对输入信号多次采样,采用某种计算方法进行数字处理,以削弱或滤除干扰噪声造成的随机误差,从而获得一个真实信号的过程。这种滤波方法只是根据预定的滤波算法编制相应的程序,实质上是一种程序滤波。可靠性高,稳定性好,修改滤波参数也容易,而且一种滤波子程序可以被多个通道所共用,因而成本很低。另外,数字滤波可以对各种干扰信号、甚至极低频率的信号进行滤波。其不足之处是需要占用 CPU 的机时。

常用的数字滤波方法有:平均值滤波、中值滤波、限幅滤波和惯性滤波等^[4]。

1.1 平均值滤波

平均值滤波即对多个采样值进行平均算法,是消除随机误差最常用的方法。根据采样的次数以及采样数据的差异采取不同的处理方法,平均值滤波有以下几种:

(1) 算术平均滤波

在采样周期 T 内,对测量信号 y 进行 m 次采样,把 m 个采样值相加后的算术平均值作为本次的有效采样值,采样次数 m 值决定了信号的平滑度和灵敏度。提高 m 的值,可提高平滑度,但系统的灵敏度随之降低,采样次数 m 的取值随被控对象的不同而不同。一般情况下,流量信号可取 10 左右,压力信号可取 4 左右,温度、成分等缓变信号可取 2 甚至不进行算术平均。在编制算法程序时, m 一般取 2、4、8 等 2 的整数幂,以便于用移位来代替除法求得平均值。这种算法适用于对周期性干扰的信号滤波;

(2) 去极值平均滤波

对连续采样的 m 个数据进行比较,去掉其中的最大值与最小值,然后计算余下的 $m-2$ 个数据的算术平均值。在编制算法程序时,为便于用移位来代替除法求得平均值, $m-2$ 应取 2、4、8 等,故 m 取 4、6、10 等。这种算法适用于工业场合经常遇到的尖脉冲干扰的信号滤波;

(3) 加权平均滤波

对每次采样值不以相同的权系数而以增加新鲜采样值的权重相加。这种算法能协调系统的平滑度和灵敏度的矛盾,提高灵敏度,更适用于纯滞后较大的对象;

(4) 滑动平均滤波

在每个采样周期只采样一次,将这一次采样值和过去的若干次采样值一起求平均,所得结果即为有效采样值。具体作法可由循环队列结构方式来实现数据的存放,比如取 m 个采样值求滑动平均,只要在 RAM

中开辟 m 个数据暂存区,每次新采集一个数据便存入暂存区的队尾,同时冲掉队首的一个数据,这样在存储队列中始终保持有 m 个最新的数据。而滑动平均滤波算法的最大优势就是实时性好,提高了系统的响应速度。

1.2 中值滤波

中值滤波是将信号的连续 m 次采样值按大小进行排序,取其中间值作为本次的有效采样值。本算法为取中值,故采样次数 m 应为奇数,一般 3~5 次即可。编制中值滤波的算法程序,首先把 m 个采样值从小到大(或从大到小)进行排队(可采用几种常规的排序算法如冒泡算法),然后再取中间值。中值滤波对缓变过程中的偶然因素引起的波动或采样器不稳定造成的误差所引起的脉动干扰比较有效,而对快速变化过程(如流量)的信号采样则不适用。

1.3 限幅滤波

限幅滤波是把 2 次相邻的采样值相减,求得本次采样值增量的绝对值,再与 2 次采样所允许的最大差值进行比较,如果小于或等于所允许的最大差值,表示本次采样值是真实的,则取本次采样值为有效采样值;反之,本次采样值是不真实的,则取上次采样值作为本次有效采样值。限幅滤波对随机干扰或采样器不稳定引起的失真有良好的滤波效果。

1.4 惯性滤波

惯性滤波是模拟硬件 RC 低通滤波器的数字实现。常用的 RC 滤波器的传递函数可简化为一阶惯性环节,其中时间常数 $T_f=RC$ 的大小直接关系到滤波效果。一般说来, T_f 越大,滤波器的截止频率(滤出的干扰频率)越低,滤出的电压纹波较小,但输出滞后较大。由于大的时间常数及高精度的 RC 电路不易制作,所以硬件 RC 滤波器不可能对极低频率的信号进行滤波。为此可以模仿式硬件 RC 滤波器的特性参数,用软件做成低通数字滤波器,从而实现一阶惯性的数字滤波。一般采样周期 T 远小于 T_f ,即滤波系数表明本次有效采样值(滤波输出值)主要取决于上次有效采样值(滤波输出值),而本次采样值仅起到一点修正作用。这对于变化缓慢的采样信号,其滤波效果良好。

惯性滤波器的程序编制设计时,应根据采样周期与截止频率适当选取滤波系数值,使得滤波器的输出既无明显纹波,又不太滞后。显然,该算法比较简单,比平均值滤波法速度快,且能很好地消除周期性干扰和较宽频率的随机干扰信号。

2 指令冗余技术

CPU 对指令的执行过程是:先取操作码,再取操作数^[5]。当计算机系统受到外界干扰,CPU 正常的工作时

序被破坏,可能造成程序计数器 PC 的值发生改变,跳转到随机的程序存储区。当程序跑飞到某一单字节指令上,程序便自动纳入正轨;当程序跑飞到某一双字节指令上,有可能落到其操作数上,则 CPU 会误将操作数当操作码执行;当程序跑飞到三字节指令上,因它有 2 个操作数,出错的机率会更大。

为了解决这一问题,可在程序中人为地插入一些空操作指令 NOP 或重复书写有效的单字节指令,即指令冗余技术。由于空操作指令为单字节指令,且对计算机的工作状态无任何影响,因而失控的程序在遇到该指令后,能够调整其 PC 值至正确的轨道,使后续的指令得以正确地执行。

但是,在程序中不能加入太多的冗余指令,以免降低程序正常运行的效率。一般在对程序流向起决定作用的指令之前以及影响系统工作状态的重要指令之前都应插入 2~3 条 NOP 指令,还可以每隔一定数目的指令插入 NOP 指令,以保证跑飞的程序迅速纳入正确轨道。

指令冗余技术可以减少程序出现错误跳转的次数,但不能保证在失控期间不干坏事,更不能保证程序纳入正常轨道后就平安无事。解决这个问题还必须采用软件容错技术,使系统的误动作减少,并消灭重大误动作。

3 软件陷阱技术

指令冗余使跑飞的程序安定下来是有条件的,首先跑飞的程序必须落到程序区,其次必须执行到冗余指令。当跑飞的程序落到非程序区(如 EPROM 中未使用的空间、程序中的数据表格区等)时,对此情况采取的措施就是设立软件陷阱。

软件陷阱,就是在非程序区设置拦截措施,使程序进入陷阱,即通过一条引导指令,强行将跑飞的程序引向一个指定的地址,在那里有一段专门对程序出错进行处理的程序。如果将这段程序的入口标号称为 ERROR 的话,软件陷阱即为一条 JMP ERROR 指令。为加

强其捕捉效果,一般还在它前面加上 2 条 NOP 指令,因此真正的软件陷阱是由 3 条指令构成。

软件陷阱安排在以下 4 种地方:未使用的大片 ROM 空间、未使用的中断向量区、程序中的数据表格区以及程序区中一些指令串中间的断裂点处。通常在未使用的程序存储器单元空间填充软件陷阱,当跑飞程序落到此处,即自动步入正轨。可以采用手动编辑烧写文件的方式实现填充。除了未使用程序空间外,软件陷阱还常安放于未使用的中断向量区(可纠正干扰导致的错误中断)、常数表格区的最后和程序的代码段之间的空隙^[5]。

由于软件陷阱安排在正常程序执行不到的地方,故不影响程序的执行效率,在当前 EPROM 容量不成问题的条件下,还应多多安插软件陷阱指令。

计算机控制系统中的干扰是一个十分复杂的问题,在工程实践中,通常都是多种抗干扰方法并用,且硬件与软件抗干扰技术相结合,互相补充,完善系统监控程序,才能取得较好的抗干扰效果。针对不同的工业控制环境,系统地研究分析产生干扰的原因和传输途径,采用合理的抗干扰措施,是设计计算机控制系统时必须预先考虑的重要内容。

参考文献

- [1] 王建华,黄河清.计算机控制技术[M].北京:高等教育出版社,2003:1-9.
- [2] 丁健.计算机控制系统的可靠性技术研究[J].计算机工程与设计,2007,28(4):985-987.
- [3] DUGAN J B. Fault trees and imperfect coverage[J]. IEEE Transaction on Reliability, 1989,38(2):177-185.
- [4] HERMANN K, GUUTER G. TTP—a protocol for fault-tolerant real-time systems[J]. IEEE Computer, 1994, 27 (1): 14-23.
- [5] 覃毅,汤荣江.单片机应用系统的软件抗干扰措施[J].微计算机信息,2007,23(2-2):101-103.
- [6] 林敏,丁金华.计算机控制技术及工程应用[M].北京:国防工业出版社,2008:103-138.

(收稿日期:2009-03-30)

(上接第 17 页)

助用户检测病毒进程并给出预警。本软件设计的关键是将进程保存在数据库中,利用计算机能快速、精确比较字符串的特点,帮助管理员判断是病毒进程还是正常进程,对于新进程的判断还需管理员熟悉常见的系统进程及病毒进程隐藏的方法。本程序在 VB6.0+ACCESS 2003,WINDOWS 2003 及 WINDOWS XP 操作系统下运行通过。

参考文献

- [1] 张桂勇,陈芳琼.API FOR WINDOWS 2000/XP 详解.北京:清华大学出版社,2003.
- [2] 刘圣才,李春葆.visual basic 6 程序设计导学.北京:清华大学出版社,2002.
- [3] 李方人华,尉宏波,李静.Windows API 编程范例入门与提高.北京:清华大学出版社,2004.

(收稿日期 2009-03-18)