单片机实时控制系统程序失控的若干防护措施

孙正凤 山东科技大学济南校区 (250031)
孔德刚 济宁排水管理处 (272100)
张运才 将军集团九州纸业有限责任公司 (250100)
姚福强 山东科技大学济南校区 (250031)

Abstract

In single chip real-time control system, the problem of progrme out of control is often encountered. This paper introduces several shields for program out of control. Application of the methods can improve the system reliability efficiently. When its program is out of control the system has an ability of self-resume.

Keywords: single chip, program out of control, shield

摘 要

单片机实时控制系统中，程序失控是经常遇到的问题。本文介绍了几种防止程序失控的方法和措施，应用这几种措施，可有效提高系统运行的可靠性，使系统具有程序失控自恢复能力。

关键词：单片机，程序失控，防护

1 引言

抗干扰设计、容错设计（包括故障检测与诊断技术）和功能设计是单片机实时控制系统设计三位一体缺一不可的重要内容。在满足各项规定功能的前提下，为了将系统付诸使用，必须提高其可靠性；但是由于工业现场环境复杂，常会受大量的电磁设备启动、停止、电源波形畸变等因素的影响，各种干扰不可避免，若只靠避错设计很难满足要求，而且也很难确保这些硬件万无一失，所以必须为系统配备容错功能。

在工业现场，大多数情况下干扰不会造成单片机系统硬件的损坏，主要是对软件运行造成不良影响。其主要特征是：指令码或数字码受干扰，使程序的执行出现错误。最典型的错误有：1）CPU 中的程序计数器 PC 的值发生跳变，使程序跑飞（run out），转去执行一个毫无意义或错误的程序段，使系统出现混乱或失控，严重时可能会造成设备损坏，甚至危及人身安全；2）输出口被失控程序非法操作，使控制量发生波动或使系统出现“死锁”（dead lock）；3）RAM 区域受干扰，造成数据被冲毁，使系统出现运行不正常，输出出错。本文以 MCS-96 系列单片机所组成的实时控制系统为例，提出了一些有效的程序失控防护措施，通过在不同系统中应用，均获得非常满意的效果。

2 发现和拦截跑飞程序的方法

2.1 指令冗余

单片机最易受干扰的是内部程序计数器——PC 的值。在受到强干扰时，PC 的值被改变，改变后的值是随机的，为一不确定值。这可使 CPU 将程序从正确的位置跑飞到 ROM 中的任何一个地址，当 PC 值飞到用户工作程序 ROM 区内时，可采用指令冗余的方法使程序走上正轨。具体做法是：1）在一些对程序流向起决定性作用的指令，如 SJMP、LJMP、LCALL、CALL 等之前插入几条 NOP 指令；2）在某些对系统工作至关重要的指令，如中断、堆栈等之前插入几条 NOP 指令；3）在程序中每隔若干条指令，插入一个 NOP 指令；4）在多字节指令前插入一条或两条 NOP 指令。

由于单片机指令大多数为单字节指令，在一个程序中，其中断和堆栈指令使用的次数也很有限，因此，采用这种方法增加存储单元的数量不会太多。但实践表明，它可大大提高系统的可靠性。

2.2 软件陷阱

当 PC 值飞到非用户程序 EPROM 或 E2PROM

区,如 EPROM 中未被使用的空间或程序中的数据表格区时,常采用软件陷阱的方法来使程序纳入正规。

所谓软件陷阱,就是一条引导指令,利用这条指令强行将程序引向一个指定的地址,在指定的地址上有一个专门的出错处理程序。假设该段程序的入口标号为 ERROR,则软件陷阱由以下三条指令构成:

```
NOP
NOP
LJMP    ERROR
```

该软件陷阱除了安置在未使用的用户 EPROM 区外,还常常安置在未使用的中断向量区、表格区的最后和程序的断裂点后(断裂点是指象 LJMP、SJMP、RET 等类指令)。

2.3 看门狗(WATCHDOG TIMER)

当跑飞的程序既没有落入软件陷阱,又没有遇到冗余指令,而是在用户程序之间或用户根本未使用的地址空间内跳来跳去,自动形成一个死循环,解决这一问题的办法是利用软件启动单片机的监视定时器,俗称"看门狗"。当出现上述情况时,利用它来使系统复位。这种方法简单、直观,只需不超过 64K 状态周期(16ms)的时间(用 12M 晶振时),计算机就可恢复正常。但一定不要忘了"喂狗",即通过软件每隔一定时间如(15ms)使 WDT 复位一次。

3 无扰动重恢复技术

上述各项措施,只解决了如何发现系统被干扰和如何捕捉住失控的程序,这对于诸如巡回检测、显示之类的普通单片机应用系统已足够了。但是,在一些关键的工业控制系统中,由于工作过程和生产工艺的逻辑性和顺序性,当程序失控后,希望引导系统恢复执行刚才失控发生时的那个程序模块,不希望,甚至不允许程序从入口处重新执行。更重要的是,失控程序往往会乱涂乱写,不仅会破坏一些重要的信息,而且会对输出口进行非法操作。在此情况下,前述方法就显得太不完整了。因此,如何恢复系统的重要信息,尽量无扰动地重新进入正常工作状态,是一个必须解决的问题,同时也是一个比较难解决的问题。

3.1 利用软件选择启动方式的方法

复位有两种方式:即初始复位和再次复位。前者习惯上称为"冷启动",后者称为"热启动"。"冷启动"时,系统的状态全部无效,需进行彻底的初始化操作。而"热启动"仅对系统的当前状态进行修复和有选择的初始化,从而使系统尽可能快的恢复正常。系统初次上电投入运行时,必须是"冷启动"。运行过程中,由于抗干扰措施引起的复位,一般均为"热启动"。为了使系统能正确地决定采用何种启动方式,往往由软件用"上电标志"来区分。系统入口程序设计策略如图 1 所示。

为使"热启动"顺利进行,首先要关中断,重新设置堆栈,将所有的 I/O 口设置为安全状态,封锁 I/O 操作,以免事态扩大,然后进行信息的恢复和状态的重入工作。

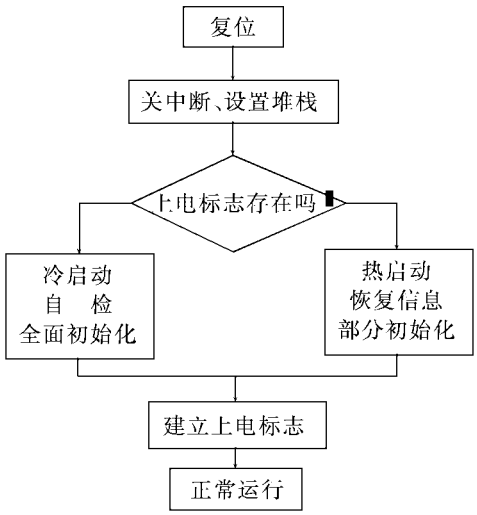

图 1 系统复位处理流程图

3.2 使受扰程序快速重入正常运行状态的方法

系统软件是由完成各种功能的程序组成的,因此可分成若干个功能模块。为了能使程序快速重入系统的正常运行状态,首先要将系统软件编制成模块化结构,并尽可能地将其细分为若干功能模块,每个功能模块在运行中需具有写入和记录功能,即设置 RAM 区的有效标志;记录编号和首地址;记录关键的又不可重新获得的数据;模块还应具有给运行监视系统发脉冲的功能等。为判断程序是否跑飞,要求在每个功能模块的结尾处将指定单元中保存的标志与本功能模块预先设置的标志进行对比。若不同,则程序跑飞,然后把它恢复到指定单元中保存的标志所对应的功能模块去重新执行;若相同,则运行正常。对于功能模块中的程序跑飞,可根据具体情况对结果的合理性进行分析和判断。若不合理,则卷回重新执行;若合理,则进入下一个功能模块。具有上述功能的程序流程图如图 2 所示。

3.3 利用数据冗余技术实现 RAM 内容自救的方法

为了保证系统实现无扰动重入正常运行状态,必须保证重要数据的正确性。实现这一目的的方法是采

用数据冗余技术。

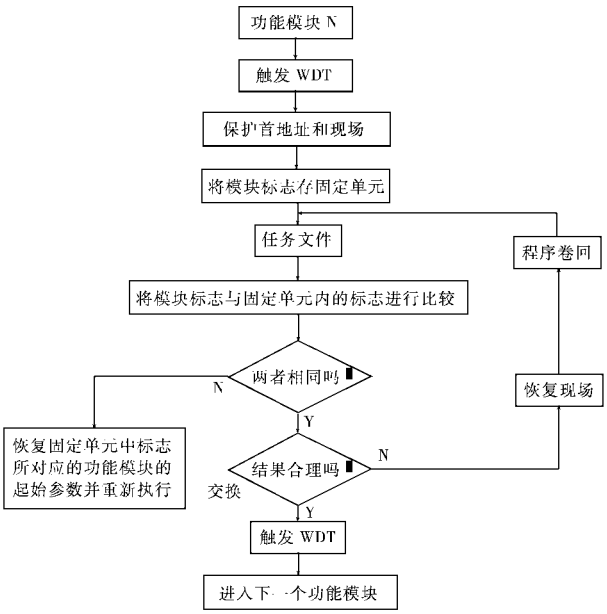

图 2　功能模块程序流程图

在实时控制过程中，干扰造成比较严重的另一危害就是冲毁 RAM 中的数据，由于 RAM 中保存的是各种原始数据、标志、变量等，如果被破坏，会造成系统出错或无法运行。数据被冲毁的情形，一般有如下三类：(1)整个 RAM 区数据被冲毁；(2)RAM 中某片数据被冲毁；(3)个别数据被冲毁。令人欣慰地是，对几乎所有的单片机实时控制系统而言，RAM 中的大部分内容是为了进行分析、计算、比较而临时寄存的，不允许丢失的数据也只占 RAM 内容的极少部分。在这种情况下，除了那些不允许丢失的数据外，其余大部分内容允许短时被破坏，最多不过引起系统一个很短时间的波动，但很快就能恢复正常。因此，在实时软件中，只要注意对少数不允许丢失的数据进行保护即可。常用的方法有“校验法”和“设标法”。这两种方法各有千秋，校验法比较繁琐，但查错的置信度高；设标法简单，但对数据表中个别数据被冲毁的情况无能为力。在编程中应综合使用。具体做法是：(1)将 RAM 工作区重要区域的始端和尾端各设置一个标志码“0”或“1”；(2)对 RAM 中固定不变的数据表格设置校验字。

在程序执行过程中，每隔一定时间通过事先设计的查错程序来校验各标志码是否正常，如果不正常，则利用数据冗余技术通过抗干扰处理程序来进行修正。冗余设计的一般原则是：在 RAM 区中相隔尽可能远且远离堆栈区的不同区域将数据备份 3 份，当读取数据时，把 3 份数据备份相比较，采用 3 取 2 的表决原则，确保数据的正确性。

3.4 锁定输出口的方法

为了防止失控程序对输出口发生非正常操作，使控制量产生波动和破坏系统的安全性，必须对输出口的操作进行严格的审查。解决的办法是硬件上采用锁定控制器，软件上采用功能块标志和口令字。

锁定控制器由两个 D 触发器来实现，如图 3 所示。平时两个锁定控制器的输出端 Q1、Q2 均为低电平，而且 Q1、Q2 只要有一个信号是低电平，输出通道就处于被封锁状态。只有 Q1、Q2 同时为高电平时，该通道才被打开。为了防止程序对输出通道的非法写入，平时程序通过端口控制信号和置 Q1、Q2 为低电平来关闭输出通道。而仅当需要输出时，程序通过端口控制信号和置 Q1、Q2 为高电平打开输出通道。程序输出时，需先给出口令字。输出模块程序流程图如图 4 所示。

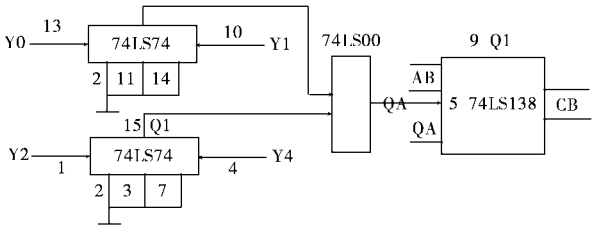

图 3　具有输出口锁定的系统原理图

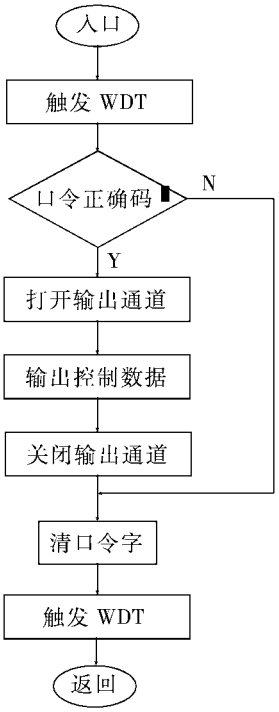

图 4　输出模块程序流程图　(下转第 45 页)

支持 MIBⅡ标准,拥有强大的 RMON(Remote Monitor)功能,使我们方便地实现标准的分布式管理。而 Transcend 系统的高层是基于图形界面的,易于使用一系列应用软件,从而能方便地与低层网管功能紧密的结合在一起。

根据炼油厂的实际状况和网络管理要求,我们选择了 Transcend for windows 平台,利用其开放的,遵循标准产品的体系结构,我们在本系统方便地实现如下功能:

1)网络图形管理 (以 Transcend for windows 为例)

Transcend 网络图形应用可以使管理员做如下工作:

· 可以做网络逻辑连接的拓扑结构图,使网络管理员对网络有一个整体印象

· 可以拷贝、存贮和删除任一个拓扑图形

· 网络拓扑图可分层设计

· 可以修改所有子系统的图标

· 可以实时地创建、移动所有的网络设备图标

· 实时地建立、移动子网及传输介质

· 可获取线路状态信息

· 可在任何一个口打印

· 建立、修改和删除任何一个网络设备的图标形状

同时,NETWORK MAP 提供以下扩展:

· 标准的地理图形背景图

· 具有可选的声音报警功能

2)配置管理功能

· 可读取所有设备的配置参数

· 可设置所有设备的配置参数

· 具有各种参数数值的说明

· 对错误输入参数值有检查功能

· 正确性检测

· 具有帮助信息(在操作时可随时使用)。

3)容错管理功能

· 详细记录了所有错误事件

· 用色彩标识各设备状态,绿色为正常工作状态,灰色为连接中断,蓝色为未知状态

· 具有五种用户自定义的颜色来标识错误信息(例如红色标识严重出错信息等)

· 当错误消除时,能自动清除错误信息,具有隔离故障功能

· 具有视、听报警功能

· 可查阅当前或以前的错误信息。有错误过滤功能,用户可按时间、子网名、标识类型来自定义过滤内容

· 定义报警上限值功能

4)网络性能管理

· 用户自定义要查询的网络统计信息

· 具有把原始数据转换成易被用户理解的图形标识信息功能

· 任何网络性能参数都被存入数据库中

· 查询信息汇总报告

5)实时统计功能

· 可对每个网络设备、网卡甚至每个通信端口选择统计信息;

· 以图形方式显示统计信息;

· 可显示实时修改的状态信息

6)安全性管理功能

具有各级的 PASSWORD 控制系统

7)虚拟组管理功能

可方便地根据带宽及管理的需求将网络或若干设备定义成虚拟组进行管理

8)管理员的管理功能

此功能主要是为网络管理员管理时所使用,即把各类信息文件或数据进行存贮、转换、删除。

5 结束语

实践说明,本系统在炼油厂的应用中圆满地完成了设计指标,真正做到了保证炼油厂信息管理系统这一大型网络系统安全、可靠、高效地运行。

(上接第 64 页)

4 结束语

上述介绍的单片机实时控制系统程序失控的若干防护措施是作者工作实践的体会。采用这些措施可以有效地提高系统运行的可靠性,并且通过简单改变就可用于其它类型微机应用系统当中,具有实用性和通用性的优点。

参考文献

1. 周航慈,单片机应用系统程序设计技术,北京航空航天大学出版社,1991

2. 孙涵芳,MCS-51 系列单片机原理及应用,北京航空航天大学出版社,1988

3. 任光龙,贺肖,计算机应用系统的干扰及抗干扰问题,电脑开发与应用,1992.3