

解决微机保护系统程序出轨的若干措施

丁书文¹, 杨雪萍¹, 张承学²

(1. 郑州电力高等专科学校, 河南 郑州 450004; 2. 武汉水利电力大学电力系, 湖北 武汉 430072)

摘要: 以 8031 单片机应用系统为例, 针对微机保护装置由于干扰可能出现的程序出轨现象, 提出了若干简单、有效、实用的对策, 并通过在实际产品中的应用, 验证了这些措施在抗干扰方面具有很强的程序出轨自恢复能力和较高的运行可靠性。

关键词: 微机保护; 程序出轨; 软件措施

中图分类号: TM77 **文献标识码:** A **文章编号:** 1003-4897(2000)05-0035-03

1 概述

由于微机保护装置在强电磁环境中长期连续工作, 所受到的干扰比较严重。干扰对微机保护程序运行造成不良影响, 其主要特征可能表现为: 数据码或指令码的个别字受干扰而发生跳变, 使程序出轨。其中, 最常见的错误情况是随机干扰因素破坏了程序计数器 PC 值, 使 CPU 从 EPROM 中读取指令时出错, 改变了程序正常运行的顺序, 其最终结果不是碰到一条 CPU 不认识的指令操作码而停止工作, 就是进入一种非预期的死循环或程序跑飞, 使整个微机系统失控, 严重时引起保护装置误动或拒动, 甚至危及整个电力系统安全稳定运行。虽然微机保护系统在硬件设计时采取了多种常见的措施, 如各种接地处理、屏蔽、隔离、滤波、退耦、旁路等^[1], 抑制或消除了一定干扰, 但这些措施并非万能, 因微机应用系统本身的整体配置、元器件(特别是新型器件)的正确使用及系统硬件、软件设计也会对可靠性产生重要影响。所以, 从软件方面进行防护和抑制干扰的研究, 软、硬件设计中进行综合考虑的抗干扰研究, 正逐渐受到人们重视。本文根据运用 8031 单片机的实际设计经验为例, 提出几种有效、实用的程序出轨对策, 能使微机保护装置在抗干扰方面具有更高的运行可靠性和程序出轨的自恢复能力。

2 程序出轨拦截对策

2.1 一种简单、可靠软件陷阱的设置方法

当 PC 值失控造成程序出轨后, CPU 离开原程序轨道而不断进入非程序区。在这种情况下, 可在非程序区设置拦截措施。这就是使程序进入陷阱, 然后强迫程序进入初始入口状态。对 MCS-51 系列单片机的微机应用系统, 理论上可用指令 LJMP # 0000H, 即在非程序区, 程序存储器中(EPROM)写入

指令码 020000H, 实际上的情况要比这种情形复杂一些。因为 LJMP 的指令码是 02, 而 JB 的指令码为 20, NOP 的指令码为 00, 故可以连续使用这些指令的组合。LJMP # 0000H, NOP, JB 0, # 000H, NOP, LJMP # 0000H ... 的指令码之组合为 0200000020000000020000 ..., 用这种码填满非程序区, 不论 PC 失控后指向这串码中的哪个字节, 最后都能导致程序执行 020000 指令码, 返回到地址 0000H 处自动复位。

2.2 设置定时中断监视主程序运行状态

使用定时中断来监视主程序运行状态时, 定时器的定时时间要稍大于主程序正常运行一个循环的时间, 而且要在主程序执行过程中执行一次对定时器时间常数的刷新操作, 这就保证定时器在程序运行正常时不会出现中断, 而当程序失去控制时, 会产生中断, 再利用中断服务程序使系统自动复位。在微机保护应用 8031 构成的硬件系统中, 作为拦截程序出轨的一个实例, 具体做法为: (1) 利用接口芯片 8155 中的一个定时器来作为定时中断使用, 其产生的中断信号作为 8031 的外部中断源 INT1 的输入信号, 用 555 定时器作为 8155 定时器的外部时钟输入。(2) 通过软件设计使 8155 定时器的定时值稍大于主程序正常循环时间, 并且在主程序循环期间重置 8155 定时器的定时时间常数。(3) 若中断源只供抗干扰使用, 只编制相应的中断服务程序即可; 若与硬件中断合用, 则对是硬件复位还是定时中断产生的自动复位进行判断。利用 8155 的好处是 8155 还含有三个扩充口和 256 个存储单元。当然, 单片机 8031 中有定时器, 如果系统不用, 其内部定时器可供其它场合使用。

3 程序运行重恢复措施

对于微机保护系统, 系统正常时, 其程序一直在

循环运行。而当系统出现故障后,保护装置程序将通过故障检测、选相、故障计算、判别是否跳闸等一系列程序,若在这个过程中程序出轨,总希望引导系统尽快恢复执行刚才失控发生时的那个程序模块,而不希望这时的失控程序复位,这样更有利于保护装置的快速恢复正常工作和系统运行的稳定性。因此如何使出轨程序尽快、更好地重新进入程序正常运行状态是一个非常棘手的问题。这里介绍一种在程序模块设计中解决这个问题的一种方法。

对于微机保护系统的软件,是由完成各种功能模块的程序组成的。为了能使出轨程序尽快恢复到程序的正常运行状态,在进行软件编制时,就应考虑该问题,即将装置软件编制成模块化结构,这不仅有利于程序的调试而且可以使程序出轨恢复时,发现程序出轨更早、程序恢复速度更快。具体做法是,将细分的各个功能模块进行编号,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;记录功能模块编号和首地址;具有给运行监视系统发脉冲的功能等。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特有的标志进行对比来判断运行的程序是否出轨。通过标志字的对比,若相同则标明没有发生功能模块间乱窜现象,程序继续按原来程序顺序进行;若不同则表示程序出轨,这时就让程序转到指定单元中保有的标志所对应的功能模块去重新执行。

对于功能模块内的出轨问题,可根据本功能模块的具体情况,来对结果的合理性进行分析和判断(其判断条件会因模块不同而不同)。若判断结果认为合理,则进入下一功能模块继续程序运行;若判断结果不合理,则通过调转指令返回本功能模块重新执行。

4 双重设置软件“看门狗”

当干扰使程序运行已出轨,正好进入一个重构的错误循环程序。若装置中设计有这方面的硬件电路,而该程序正好包含了“看门狗”电路的访问,且访问时间间隔小于“看门狗”的溢出时间,则上述“看门狗”电路和软件陷阱作用将失效^[2]。

对于上述问题,可在程序的另一处设计一段软件程序,使之具有“看门狗”的功能,即在 RAM 中找一个内存单元对程序循环次数进行记录,当记录数为 N 时,8031 单片机必须访问定时器以重新启动它,同时对该内存单元清零定时器的定时时间略大于 N 次循环时间 T ,如果 T 时间内 CPU 未访问定

时器,定时器将溢出引起 CPU 复位。

5 一种复位的软、硬件综合设计

一般的微型机应用系统中,设计者通常要安排硬件自复位电路,就是所说的“看门狗”电路。这样,在程序出轨后大多能实现硬件自复位功能。但是,对于微机保护系统,由于保护装置的启动通常要使用当时采集数据与过去记录数据进行比较判别。在程序出轨造成死机较长后复位,有可能会使保护装置启动误动作,这可能造成严重后果。为尽快判别程序出轨和恢复程序正常运行,在硬件自复位前设置软件条件复位就具有十分重要的意义。

“看门狗”电路本身就可看作是一个可被清除的定时脉冲发生器,它本身可以由单稳态触发器或计数器等构成。如果没有清除脉冲的话,它将产生一个固定频率的输出脉冲,这个脉冲可以引到系统的复位端。当主程序正常工作时,不断地发出清除脉冲,使脉冲发生器无输出。一旦程序出轨,清除脉冲消失,“看门狗”工作,将发出一个复位脉冲,使系统回到正确的工作状态。对上述一般的“看门狗”工作性质,初看起来很完美,但实际运行时仍有死机现象发生,其原因有:程序出轨后,“看门狗”发出复位脉冲,但程序刚刚正常尚不及发出一个脉冲时,程序再次被干扰而出轨,此时“看门狗”正处于稳态,将不再发出复位脉冲。因此,对于一个较理想的硬件复位电路应具有在检测出主程序出轨时发出的复位脉冲不应为单次脉冲,而应在主程序复位前期间内每过一定的时间间隔都要发出复位脉冲,直到主程序恢复正常工作为止。

为此,对程序出轨的复位,一种软、硬件综合设计为:在系统正常运行时,由 CPU 定时输出一个脉冲信号,其可用单触发器与计数器组成的电路来进行检测与判断,利用计数器的输出 Q_i 提供软件条件复位信号, Q_j 作为硬件自复位信号 ($j > i$)。从程序出轨而死机到硬件复位开始的这段时间内,在硬件复位之前,硬件复位电路给出一个预复位信号,该预复位信号产生一个外部中断,在该中断服务主程序中分别判断系统当时的状态,根据不同状态条件,引导程序进入合适的入口地址。所以,这个中断服务程序在硬件自复位之前实现软件复位功能,其条件可能有多个,复位入口地址也各不相同。若进入该中断服务程序后,马上给出计数器的一个消除脉冲。

(1) 硬件自复位电路的设计

图 1 为硬件自复位电路原理接线图,利用 8031

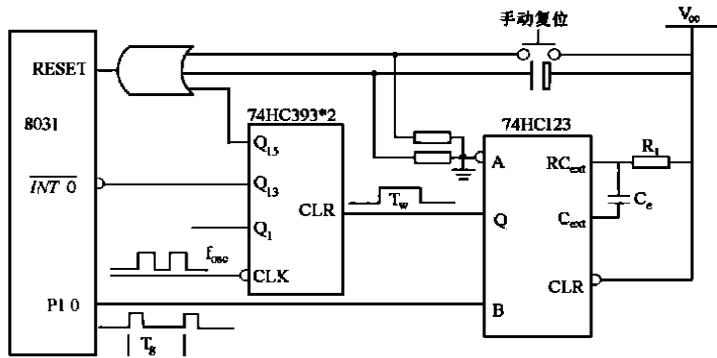


图1 硬件自复位电路原理图

单片机的口线(如用 P1.0)作为硬件复位电路的输入清除信号,用 74HC123 可再触发单稳触发器形成脉冲输出,用两片 74HC393 组成 16 位进制计数器。当程序出轨后,8031 口线 P1.0 脉冲 T_g 清除消失,单稳态触发器形成一个脉冲信号 T_w 作用于计数器的复位端,使计数器的所有输出端 $Q = 0$ 。 T_w 通过电阻 R_T 和电容 C_C 大小决定,计数器的计数时钟直接利用 CPU 的时钟发生器。CPU 的复位输入用计数器 Q_{15} 的硬件自复位, Q_{13} 作为在硬件复位前的一个预复位中断信号接到 CPU 的 $\overline{INT0}$ 上,提供软件复位的起始信号。若 CPU 不能响应 $\overline{INT0}$ 中断,则等到硬件复位时间将由硬件完成复位。若选取 CPU 时钟振荡器频率为 12MHz,通过计算,在软件条件复位不能实现的最不利情况下,从死机到完成硬件自复位的时间为 2.73ms,对于微机保护系统将不会产生太大的影响。

(2) 软件条件复位的实现

软件条件复位是通过 CPU 接受外部中断信号 $\overline{INT0}$ 后,来执行的一段中断子程序。这时 CPU 进入中断后 关闭所有中断,以便进行故障的判别和处理; 堆栈初始化以保证执行完成 $\overline{INT0}$ 中断服务子程序后栈底为初始状态; 对系统的有关状态与控制量进行比较、判断来决定程序的重新入口地址,这

些地址是主程序中基本功能模块的程序首选地址。通过这种软件条件复位,可以使系统在无扰动或小扰动下,尽快进入正常运行状态^[3],尽可能地减少对微机保护系统的冲击,尽量减少干扰对微机保护系统的影响。

6 结束语

电力系统中的各种电气设备都存在强大的电磁干扰^[4],而电力系统的安全、稳定运行又要求微机保护系统具有特别高的可靠性,干扰对微机保护系统的程序出轨问题一直是微机保护软件设计中最棘手而又必须妥善解决的问题。本文是笔者在对微机保护软件的实际编制过程中,通过反复探索,就如何微机保护系统防范程序出轨的一些设计体会。本文介绍的防范程序出轨的综合措施简单、有效、可靠,这一点已在电力系统输电线路互感参数带电测试装置、电力系统异地相角测量装置、电力系统故障精确定位装置等实际设计中得到应用和验证。其措施也可推广到各种常受干扰的微型机应用系统。

参考文献:

- [1] 陈德树. 计算机继电保护原理与技术. 水利电力出版社, 1992.
- [2] 肖冬荣. 微型计算机实时控制的抗干扰. 湖北科学技术出版社, 1993.
- [3] 张承学等. 单片机应用系统硬件自复位与软件条件复位的综合设计. 微计算机应用, 1997, 5(21).
- [4] 杨奇逊. 微型机继电保护基础. 中国电力出版社, 1998.

收稿日期: 1999-07-07 改回日期: 1999-08-20

作者简介: 丁书文(1967-),男,硕士,讲师,主要从事电力系统微机保护方面的教学与研究工作; 杨雪萍(1969-),女,大学本科,讲师,研究方向为微机在电力系统的应用; 张承学(1952-),男,系主任,教授,研究方向为电力系统控制与保护。

Some measures on programs running off the track for microprocessor based protection system

DING Shu-wen¹, YANG Xue-ping¹, ZHANG Cheng-xue²

(1. Zhengzhou Electric Power College, Zhengzhou 450004, China;

2. Wuhan University of Hydraulic and Electric Engineering, Wuhan 430072, China)

Abstract: Some examples of the use of 8031 microprocessor based protection programs running off the track as a result of interference are taken in this paper. Bring forward several methods that have been tested in some products that higher reliability and regain - self ability of the methods of microprocessor protection programs running off the track in anti-disturbance.

Keywords: program run off the track; microprocessor based protection; software measure