微机继电保护软件可靠性探讨

所 旭[1],张 萍[2]

(1. 北京四方继保自动化有限公司,北京 100085; 2. 北京大学信息管理系,北京 100871)

摘要: 微机保护的软件的性能是否可靠,是影响微机保护运行可靠性的关键因素之一。在微机保护开发的整个过程中,应该强调软件的可靠性。简要介绍了衡量软件可靠性的几个要素,指出当前微机保护开发人员对软件可靠性认识的误区,提出一种微机保护软件开发的标准流程。文章强调,应该在开发流程上和开发制度上保证微机保护装置软件的可靠性。

关键词: 微机保护; 软件可靠性; 开发规范

中图分类号: TM77 **文献标识码**: A **文章编号**: 1003-4897(2004)12-0043-04

0 引言

众所周知,电网对继电保护的性能要求可以归结为:可靠性、选择性、快速性和灵敏性,统称为"四性"。继电保护的四性有的相辅相成,有的相互制约,需要针对不同的需求,做出不同的协调。

其中,可靠性的意思是电网所配置的继电保护只能在预先规定需要其动作的情况下才可动作,在其他一切不需要其动作的情况下都不动作[1]。而在目前微机保护原理和技术都趋于成熟、微机保护大量投运的情况下,继电保护的可靠性通过下列因素得以保证:

1) 继电保护原理保证其可靠性;

2) 继电保护与安全自动装置之间相互配合,保证可靠性;

3) 微机继电保护装置硬件的可靠性;

4) 微机继电保护软件的可靠性。

对于第1)条和第2)条,继电保护工作者在研究传统继电保护时,已经形成严密的理论体系;第3)条是微机继电保护工作者首先要考虑的问题,目前也得以很好地解决。虽然并没有正式的组织机构或会议讨论或规程规定,但是我国继电保护工作者都普遍接受的将我国微机保护划分为第一代、第二代、第三代的方法,其划分标准就是凭借此点,即:

①第一代微机保护:单CPU工作,多插件组合;

②第二代微机保护:保护功能在单个插件内实现,总线不出插件;

③第三代微机保护:保护功能集成于一个芯片,总线不出芯片。

可见随着微机保护的硬件设计思想不断进步,保证可靠性不断提高。

本文将对第4)条展开讨论。

1 软件可靠性

"软件可靠性"通常是指在某一规定时间内,软件无差错地完成其基本功能的能力。衡量软件可靠性的几个要素如下:

1) 可用度,指软件运行后在任意随机时刻需要执行规定任务或完成规定功能时,软件处于可使用状态的概率。可用度是对应用软件可靠性的综合度量。

2) 初期故障率,指软件在初期故障期内单位时间的故障数。一般以每100 h的故障数为单位。可以用它来评价交付使用的软件质量与预测什么时候软件可靠性基本稳定。

3) 偶然故障率,指软件在偶然故障期内单位时间的故障数。一般以每1000 h的故障数为单位,它反映了软件处于稳定状态下的质量。

4) 平均失效前时间(MTTF),指软件在失效前正常工作的平均统计时间。

5) 平均失效间隔时间(MTBF),指软件在相继两次失效之间正常工作的平均统计时间。在实际使用时,MTBF通常是指当 n 很大时,系统第 n 次失效与第 $n+1$ 次失效之间的平均统计时间。

6) 缺陷密度(FD),指软件单位源代码中隐藏的缺陷数量。通常以每千行无注解源代码为一个单位。一般情况下,可以根据同类软件系统的早期版本估计FD的具体值。如果没有早期版本信息,也可以按照通常的统计结果来估计。典型的统计表明,在开发阶段,平均每千行源代码有50～60个缺陷,交付后平均每千行源代码有15～18个缺陷。

7) 平均失效恢复时间(MTTR),指软件失效后恢复正常工作所需的平均统计时间。对于软件,其失效恢复时间为排除故障或系统重新启动所用的时间,而不是对软件本身进行修改的时间。

对于多数软件开发人员和项目管理人员而言,可靠性等同于正确性,开发人员热衷于寻找软件中的"缺陷(Bug)"然后改正。也就是讲,他们认为微机保护软件开发的步骤是:编码,然后改错。这是一个认识问题,客观地讲,正是由于这种态度,使得影响微机继电保护软件可靠性的因素在开发的全过程中潜入并生存下去。目前影响我国微机保护软件可靠性的因素归结下来有:

1) 需求分析定义不够准确,例如用户提出的需求不完整,对用户需求的变更未及时消化,软件开发人员和用户对需求的理解不同等等,这也是造成诸多"非标"的原因;

2) 软件结构设计失误,例如在进行软件总体结构设计的时候,缺乏对特殊情况和错误处理的考虑等;

3) 编码有误;

4) 针对某一模块,还未给出逻辑框图就着手编码,造成考虑问题不周;

5) 测试不规范,例如数据准备错误,测试用例错误,测试不充分等;

6) 文档不齐全,文档相关内容不一致,文档版本不一致,缺乏完整性等。

诚然,在前述态度指导下,靠寻找软件缺陷(Bug)的办法是保证软件质量、提高可靠性的一种方法。然而还存在更好的办法,就是把软件可靠性和软件开发的全过程联系起来,在这个过程中的每一个阶段,都做出高质量的计划、文档、编码和测试。

2 制度和规范

微机继电保护软件的可靠性相对于保护硬件可靠性而言,更难保证,如果在开发前和开发过程中对软件可靠性没有提出明确的要求,往往可能造成只注重速度、结果的正确性和用户界面的友好性等,而忽略了可靠性。在投入使用后才发现大量可靠性问题,增加了维护难度和工作量,严重时只有束之高阁,得过且过。

实际上,开发微机保护软件时,有许多可以参考的标准、规范和制度,例如 IEEE 就有许多关于软件质量的标准。这些标准囊括了软件质量保证、软件生效、软件计划管理、软件测试、测试文档、软件检查等内容。再如国外一些著名的软件生产商也有可供借鉴的完整的软件研发、可靠性保证体系。

3 继电保护软件开发

3.1 微机继电保护装置开发步骤

一般来讲,整套继电保护装置开发的全过程可以是:

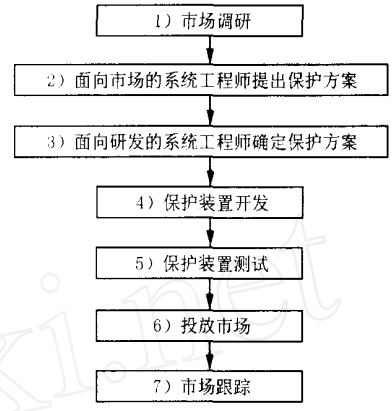

图 1 继电保护装置开发流程

Fig. 1 Developing process of the relay protection

在上述流程中,任何一个步骤都不是可以一次完成的,需要多次反复方可确定。本文关心的软件可靠性问题主要在图中第 3)、第 4)、第 5)和第 7)个步骤中保证。

3.2 微机继电保护软件开发的各个阶段

考虑到微机继电保护开发面临的实际问题,如成本问题、开发工期问题、人员配备问题等,参考已知的成熟案例[2,4],结合本文作者的开发经验,制定软件开发需要遵循的解决方案,由如下几个阶段组成。

1) 需求分析阶段

这个阶段由面向市场的设计工程师完成,例如某些公司在商务部门设置"市场系统工程师(Market System Engineer)"职位,负责收集用户需求,和用户进行全面和深入的沟通,以明确用户所需的究竟是一种什么样的装置,我们的软件应该为用户完成什么样的功能,并形成文档,并接受开发部门的反馈,修改需求分析,形成最终方案。在这个阶段,微机保护装置及软件将要完成的任务大体确定。

2) 功能定义阶段

这个阶段主要由开发部门的软件设计人员完成,保护装置的软硬件组件在这个阶段通过软件工程师和硬件工程师的共同努力初步成型。在功能定义阶段,负责软件设计的人员要预先估计未来可能

发生变化,遵循信息隐藏的原则以提高软件的可靠性和可维护性。

我国的微机保护生产商在此方面付出了努力,文献[3]提出了保护软件分层的概念并付诸实现,取得良好的效果。微机保护的软件可以分为驱动层、系统层和应用层。不同的层次分别设计,完成不同的功能,并可以自检;驱动层还需要承担对硬件行为进行检查的任务,以便及时发现错误,提高可靠性。

3) 软件设计阶段

良好的软件设计与所采用的软件设计方法、设计工具和设计准则有关。软件设计方法主要有面向数据流的设计、面向对象的设计和面向数据的设计方法等。微机继电保护软件需要对保护装置和电力系统一次设备进行实时控制,因此使用面向数据流的设计方法是比较合适的。

设计的准则是遵循数据交换的流程,自上而下,层层细化,复杂问题简单化。

在这个阶段,微机保护各个层次被细化,每个层被分为多个软件模块,模块与模块之间需要交换的数据,模块本身实现的细节问题在此阶段设计。

软件设计必须详尽、细致,输出的文档不但要有模块功能定义、变量定义,最终还应该形成"伪代码",只有达到这个程度,才有可能把所有应该在编码之前考虑的问题全部涉及。

4) 软件代码编写和测试阶段

有了详尽的设计,软件的编码实现就相对简单了,当然,在完成每个模块和把不同模块集成的时候都要进行测试,软件的测试应该贯穿这个过程。

5) 整机测试阶段

微机继电保护的整机测试包括静态模拟实验、动态模拟实验和试运行。

3.3 软件测试

充分的测试是交付用户优质可靠的微机继电保护装置的保证。

1) 软件测试目的

①测试的目的是尽可能地发现缺陷;

②衡量一个测试用例优劣的标准是看其是否可能发现软件中隐藏的缺陷;

③一次成功的测试实例在于发现了软件中隐藏的缺陷。

可见,测试的目的是证明软件缺陷的存在,而不是证明缺陷不存在。

2) 测试方法

测试的方法有两种,分别为黑盒测试和白盒测试。

黑盒测试把系统看成一个黑盒子,不考虑程序的内在逻辑,只根据需求规格说明书的要求来检查程序的功能是否符合它的功能说明。静态模拟实验和动态模拟实验可以归为黑盒测试。

白盒测试允许测试人员对程序内部逻辑结构及有关信息来设计和选择测试用例,对程序的逻辑路径进行测试。以 C 语言为例,要求对代码进行逐条跟踪,不放过任何一个程序分支,在某些情况下也许还需要在汇编语言级对代码进行逐条跟踪,尽管不必经常这样做。我国微机保护的开发人员还是更倾向于相信汇编代码。

3) 人员配备

我国稍具规模的微机保护开发商都有自己的测试实验室,进行整机测试(黑盒测试的一个实例),并且在某个型号保护投运前还要到第三方进行动态模拟实验。

在保护开发过程中的代码编制阶段一般不配备专职测试人员,实际上可以在此阶段请代码编制人员互相测试对方的程序,也就是建立一种代码检查认证机制(Code Inspection)。代码检查必须有时间保证,代码作者和检查者必须有对事不对人的正确心态。因为任何开发人员都倾向于欣赏自己程序的成功之处,而不愿看到失败之处。

4 保证软件可靠性的管理工具

单个微机保护装置的软件规模并不是很大,但针对不同的用户需求对软件源代码的修改所形成的版本众多,而且不能保证每一次修改都可以接受。有如下所列的非常现实的困难摆在保护开发项目管理人员面前:

①开发项目的整体管理如何进行;

②项目组的各个子项目如何统一调度;

③项目组的成员之间如何进行有效协调;

④修改轨迹如何保留,以便撤消不可接受的修改;

⑤研发过程中形成的软件的各个版本如何进行标识;

⑥各个版本的源代码和文档如何进行关联等。

必须引入一种管理机制,在这种机制下,不仅是源代码,而且整个开发项目也可以得到有效管理。借助项目管理工具可以实现这样的目的,目前市场上可以选择的此类工具很多,例如 Visual SourceSafe(简称 VSS)具有面向项目的特性,能有效地管理应

用程序开发工作中的日常任务。

VSS 将所有的项目源文件(源代码文件、文档等)以特有的方式存入数据库。开发组的成员不能对该数据库中的文件进行直接的修改,而是签出(Check out)到自己的工作目录下进行修改和调试,然后将修改后的项目文件签入(Check in)给 VSS,由它进行综合更新。

5 结论

本文讨论了微机继电保护装置软件开发的全过程,重点论述了软件可靠性问题和保证软件可靠性的方法。

参考文献:

[1] 杨奇逊(YANG Qi-xun). 微型机继电保护基础(Digital Relay Fundamentals)[M]. 北京:中国电力出版社(Beijing:China Electric Power Press),1988.

[2] Heising C R, Peterson R C. Digital Relay Software Quality[Z]. GE Publication GER-3660.

[3] 李铁群,吴国 ,张涛(LI Yi-qun, WU Guo-yang, ZHANG Tao). 基于模块的可编程保护装置软件设计新概念(The New Software Design Method of Module-based Programmable Digital Relay)[J]. 电力系统自动化(Automation of Electric Power Systems),2002,26(15):66-69.

[4] Maguire S. 编程精粹——Microsoft 编写优质无错 C 程序秘诀(Writing Clean Code—Microsoft Techniques for Developing Bug-free C Programs)[M]. 姜静波,佟金荣,译(JIANG Jing-bo, TONG Jin-rong, Trans). 北京:电子工业出版社(Beijing: Publishing House of Electronics Industry),1993.

收稿日期: 2003-09-25; **修回日期**: 2003-10-10

作者简介:

所 旭(1974 -),男,硕士,从事微机继电保护装置的研究与开发工作;

张 萍(1978 -),女,硕士研究生,主要研究方向为软件可靠性等。

Research on the reliability of digital protective relay software

SUO Xu[1], ZHANG Ping[2]

(1. Beijing Sifang Automation Co., Ltd, Beijing 100085, China; 2. Department of Information Management, Peking University, Beijing 100871, China)

Abstract: The reliability of digital protective relay software is emphasized in this paper. The stuffs which affect the software reliability of protective relays are introduced, the misconceives of the software reliability among developers are pointed out. In the last, this paper brings forward an approach to ensure the software reliability in the whole process of the software development.

Key words: digital protective relay; software reliability; developing guideline

(上接第 42 页 continued from page 42)

收稿日期: 2003-09-28; **修回日期**: 2003-11-20

作者简介:

许 珉(1956 -),男,副教授,从事电力系统监视与控制方面的研究;

刘 伟(1976 -),女,硕士研究生,研究方向为电力系统监视与控制;

杨宛辉(1943 -),女,教授,从事电力系统监视与控制方面的研究。

Fault simulation and training system of 110 kV city power network

XU Min, LIU Wei, YANG Wan-hui

(School of Electrical Engineering, Zhengzhou University, Zhengzhou 450002, China)

Abstract: Applied the object-oriented programming, the 110 kV city fault simulation and training system for power network is developed. Topological graph of network is integrated with the background database. All functions of the proposed system can be achieved by operating the graphical interface. By setting a fault point and fault type in the graphical interface, it can simulate the state of switch and the action of automechanism as well as the relay protection after fault occurring in terms of the relay collocation and the fixed value in database. In the fault simulation system, a method of imitating real-time data acquisition, which provided by power flow calculation and short circuit calculation, and relay protection has been adopted. After setting fault by instructor, the result of short circuit calculation will be acquired at once and compared with the setting of protection. In this way, it simulates the action of fault and relay protection.

Key words: short circuit calculation; fault simulation; student training