

# 智能型变电站自动化系统软件的实用化设计

李洪波<sup>1</sup>,张吉赞<sup>1</sup>,崔建国<sup>2</sup>

(1.烟台师范学院数学与信息学院,山东 烟台 264025;

2.烟台东方电子集团保护及变电站自动化事业部,山东 烟台 264001)

**摘要:**分析了国产变电站自动化系统软件在适应性、稳定性与通用性方面存在的问题。运用界面对象和模板文件增强了界面的适应性。设计了将用户和设备源程序隔离的方案,该方案能根据用户的特殊需求而定义运算规则,提高了系统软件功能级的适应性。分析了影响变电站自动化系统软件稳定性的因素,提出了相应的解决措施。

**关键词:**适应性; 稳定性; 通用性; 界面对象; 模板文件; 参数文件

**中图分类号:** TM76      **文献标识码:** A      **文章编号:** 1003-4897(2005)17-0058-04

## 0 引言

变电站自动化系统(Substation Automation Systems)的智能电子装置(Intelligent Electronic Device IED)是用传感器直接采集并处理变电站现场信号的智能电子设备,它是变电站综合自动化系统的关键设备。从对象的功能上来分,变电站自动化系统的 IED可分为保护对象、遥测对象、遥信对象、电镀对象以及遥控对象、小电流接地选线对象、无功综合控制对象和防止误操作对象等实体。这些对象是构成变电站综合自动化系统的基本单元,基本单元系统软件的灵活性、稳定性与通用性直接影响到产品的质量和生产厂家的信誉,直接关系到变电站持续在线并安全可靠地运行。在硬件资源具备的前提下,好的软件设计能直接增强适应性、保证稳定性与提高通用性。

目前,变电站自动化系统的 IED的 CPU大多是性能较高的 MC68360等高档 CPU,内存资源相对充裕,显示器分辨率一般在 64 ×128以上;系统软件大多数是基于嵌入式实时多任务操作系统,编程语言大多数为多范式 C++程序设计语言。但在产品设计时,变电站自动化系统软件还没有把硬件的性能全部开发出来,还没有充分利用面向对象的三大要素,使得工程安装、维护与使用还有许多不便之处。

## 1 变电站自动化系统软件存在的问题

目前,国产 IED的系统软件在通用性、稳定性、灵活性和适应性方面普遍存在问题,导致用户使用不方便,甚至影响用户正常工作。

### 1.1 界面级灵活性和适应性的问题

提示信息意文不明或为原始的英文提示,不能

根据实际的线路名称而改变界面信息,不能直观说明被测对象。

对重要的复杂的操作无联机帮助,产品使用说明书不详细。

### 1.2 功能级灵活性和适应性的问题

产品灵活性差,不能适应用户的特殊需求。运行程序本身不能根据现场的实际情况对采集到的信号进行算术运算规则、关系运算规则和逻辑运算规则的定义。程序要完全适应现场的具体情况,必须由开发人员按用户的实际需求现场修改程序完成。现场仓促修改程序,可能会留下意想不到的隐患,影响设备安全持续在线运行。

### 1.3 系统稳定性方面存在的问题

出现无规律的死机现象;不能重现的错误;对一般性错误的屏蔽能力较差;对输入数据没有进行充分并且有效的有效性检查,结果造成不合要求的数据进入数据库。

## 2 智能型变电站自动化系统软件的实用化设计

鉴于变电站 IED存在的问题,用软件工程的思想指导智能适应型变电站 IED软件系统的设计,从而有效解决上述问题。

### 2.1 界面适应性的需求与设计

#### 2.1.1 适应性的需求

智能型 IED要赢得客户的信赖,赢得市场的信誉,不只是把设备开发出来客户就可以使用,而是必须做到客户用起来很顺手,不觉得很混乱,这属于适应性,非常重要。影响与决定软件适应性的因素有:用户的特点、不同的用户和不同的使用者。厂商开

发出来的智能型 IED 不仅是给一些受过高等教育的专门设计人员用的,设计时应考虑到不同层次的使用者。否则会直接影响智能型 IED 最后的质量。如果产品能做到傻瓜式的应用,会减少厂家维护的次数和时间,降低安装与维护的费用。

### 2.1.2 适应性的设计

适应性设计主要包括界面设计和联机帮助系统设计两个方面。

#### 1) 面向对象的界面模板设计

界面能够让用户轻松操作 IED,能够把 IED 的所有功能展现出来。界面设计应体现出设备的全部功能、符合行业特点,顺应用户的习惯,适应现场对物理信号进行千差万别的逻辑定义。变电站自动化系统软件的逻辑特征点有:遥测、遥信、遥控、电操、保护事件和 SOE 事件 6 种。实现方法是设计一个通用的界面模板,使其根据设备的具体功能和现场的实际接线情况,将现场的全部逻辑特征点与实际的物理信号相对应,如图 1 所示。

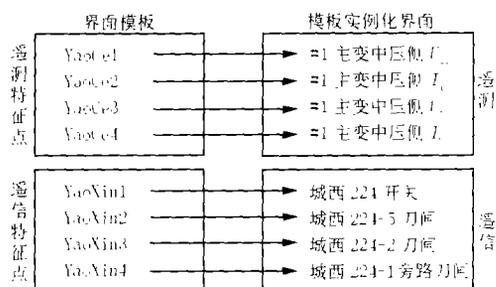


图 1 界面模板与模板实例化界面示意图

Fig 1 Interface template diagram and its instance

从图 1 不难看出有两个问题: 如何设计界面模板, 界面模板如何实例化。运用面向对象的封装技术和一个模板文件 Display.txt 加以解决。将显示界面封装为一个对象。模板文件 Display.txt 的实质就是界面模板,将现场实例化内容按模板的格式填入模板文件中。界面对象通过自己指定的接口函数将实例化模板文件读入自己创建的内存界面堆对象,并将自己的指针指向该内存界面堆对象,通过自己的显示成员函数将内存界面堆对象的实际内容从显示器上输出。

显示对象的设计:显示对象的标识如图 2 所示。

界面对象的数据成员含义说明如下:

count: IED 所需的不同显示界面的总数。

column: 显示器的最多显示行数。

char \* p[count][column]: 当  $0 \leq i < \text{count}$  且  $0$

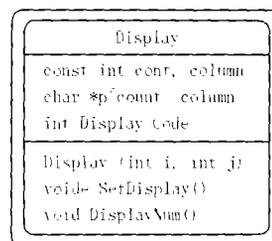


图 2 界面对象标识图

Fig 2 Schema for interface object

$j < \text{column}$  时,  $p[i][j]$  指向第  $i$  号显示界面中第  $j$  行字符串的首址。二维指针数组  $p$  管理内存界面堆对象。

DisplayCode: 指明当前正在显示的界面号。

界面对象的接口成员函数的说明如下:

构造函数 Display(int i, int j):  $\text{count} = i$ ,  $\text{column} = j$ , 根据模板文件 Display.txt 的实际信息创建并填写内存堆对象  $p$ 。

SetDisplay() 函数的功能是将实例化模板文件 Display.txt 信息重新读入自己创建的内存堆对象  $p$ 。

DisplayNum() 函数的功能是显示输出 DisplayCode 号界面的信息。

#### 模板文件的设计

假定显示器的行数为 4, 界面数为 2, 则图 1 所示的默认模板文件和实例化模板文件的格式及内容见图 3 所示。

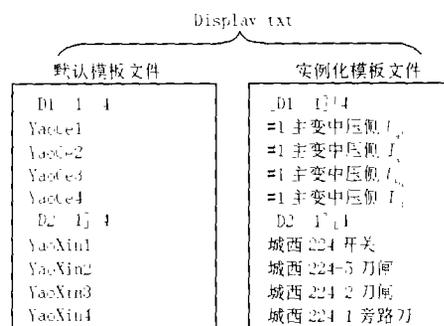


图 3 模板文件格式和内容示意图

Fig 3 Format and content of template file

在图 3 中,  $[D1][1][4]$  行的含义是显示界面号为 1, 起始显示行为 1, 结束显示行为 4, 也就是界面 1 总行数为 4。界面 1 的第 1 行的默认显示内容为“YaoCe1:”, 实例化后显示内容为“#1 主变中压侧  $U_{a1}$ ”; 第 2 行的默认显示内容为“YaoCe2:”, 实例化后显示内容为“#1 主变中压侧  $I_{a1}$ ”, ..., 以此类推。

#### 模板文件与用户接口

用户可以用维护软件从维护端口将模板文件下

载到设备中。

综上所述,通用适应性面向对象的界面模板运行机制如图 4所示。客户用维护软件通过维护端口向设备下载实例化模板文件,界面对象根据实例化模板文件生成具体的界面。

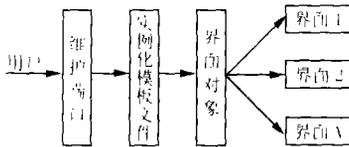


图 4 面向对象的界面模板实现机制

Fig 4 Realization of object-oriented interface template

## 2) 联机帮助系统设计

联机帮助系统是在脱离产品使用说明书的情况下帮助用户对设备进行正确的操作。联机帮助系统采用动态界面设计技术。动态界面是将帮助界面作为界面模板设计的一部分来看待,能够灵活修改。

总之,借助于上述的动态界面设计技术,能够以不变的程序和硬件结构,适应现场万变的界面定义,有效地提高了 IED 界面的灵活性与适应性。

### 2.2 功能灵活性和适应性设计

IED 系统软件的功能适应性弱,不能适应用户的特殊需求。系统软件本身不能根据现场的实际情况而改变自身的算术运算规则、逻辑运算规则或关系运算规则。目前,也没有相应的维护工具支持用户或服务人员现场妥善地修改程序,以完全适应现场的实际要求,必须由开发人员按用户的实际需求现场修改程序完成。

为避免手工修改源程序,设计一个智能化的维护工具,该工具将服务人员或用户与源程序隔离开,用户或服务人员不能直接修改源程序,只能按用户要求输入合法的表达式,通过智能化维护工具去修改源程序。智能化维护工具验证表达式合法后,经用户确认后便可自动修改源程序、编译链接源程序,最后更新设备内的可执行代码。

问题是智能化维护工具修改源程序的哪个文件,如何将新的规则表达式传送到该文件的正确的位置。解决方法是可以将设备源程序分解成两部分,一部分为不可变的部分,另一部分为可变的部分。可变的部分是智能化维护工具能够修改的部分,不可变的部分即使是智能化维护工具也不能修改。可变的部分是为适应用户的特殊需求配置的软接口,可以利用 C++ 的宏定义来接受用户的新规则。这样,可变的部分实质就是一个变电站自动化系统的一个参数文件,用户借助智能化维护工具去

修改它。

这里,引入虚遥信、虚遥测以及虚保护信号的概念。“虚”是指并不对应真实的某路物理信号,而是两路以上物理信号经过运算规则转换后的量。用虚遥信、虚遥测以及虚保护信号来接受用户定义的特殊规则表达式。虚遥信、虚遥测以及虚保护信号的数量和宏名由系统软件预先定义好,统一存入一个指定的可变接口文件。智能维护工具软件可以访问可变接口文件,能够修改常量串,只能引用宏名,不能修改宏名。工作原理如图 5所示。

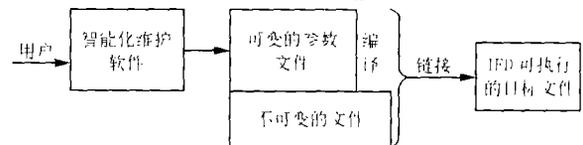


图 5 功能适应性设计原理

Fig 5 Principle for improving adaptability of IED's function

例如,某设备软件系统参数文件名为 Visual h,系统共定义 2 个虚遥信、2 个虚遥测、4 个实遥信、4 个实遥测。文件 Visual h 内容如图 6所示。

```
#include MVARITAL_2003
#define MVARITAL_2003
extern int YX1
extern int YX2
extern int YX3
extern int YX4
extern float Yc1
extern float Yc2
extern float Yc3
extern float Yc4
#define YX5 YX1&& YX2 - YX3
#define YX6 YX1 - YX2
#define Yc5 Yc3 - Yc4
#define Yc6 Yc3&& Yc4
#endif
```

图 6 参数文件实例

Fig 6 Example of parameter file

在图 6中,虚遥信的宏名分别为 YX5 和 YX6,虚遥测的宏名分别为 Yc5 和 Yc6,不能改变,它被不可变的程序部分所引用。4 个实遥信的变量名分别为 Yx1、Yx2、Yx3 和 Yx4,4 个实遥测的变量名分别为 Yc1、Yc2、Yc3 和 Yc4,这些也是与不可变的程序交换信息的变量,变量名称不能改变。宏名代表的字符串的合法符号为实遥测名、实遥信名以及算术运算符、关系运算符、逻辑运算符或条件运算符,该串是由可变文件中的变量名或宏名以及上述三种表达式构成的合法 C++ 表达式。

智能维护软件的作用是根据用户的特殊要求生

成 Visual h 黑体字的常量串,并保证这些常量串的合法性。满足合法性后,经用户确认无误,便可编译链接,生成适合现场特殊要求的可执行目标文件并下载到设备的 EPROM 中。

通过上述设计,有效地提供了智能化隔离工具。借助该工具,既能够完成用户特殊的规则定义,又能够有效、安全地修改源程序,提高了 IED 系统软件的灵活性与适应性,避免了现场仓促修改程序可能带来的隐患。这样,程序一旦开发完毕,就被封装起来,服务人员或用户在现场灵活地配置,符合软件工程的要求。

### 2.3 软件稳定性问题分析与解决措施

1) 程序运行过程中不断申请资源,但没有完全释放资源,造成系统性能越来越低,并出现无规律的死机现象。

从堆上分配,亦称动态内存分配。程序在运行的时候用 malloc 或 new 申请任意多少的内存,程序员自己负责在何时用 free 或 delete 释放内存。动态内存的生存期由程序员决定,使用非常灵活,但问题也最多。

内存错误一般很难发现。编译器不能自动发现这些错误,通常是在程序运行时才能捕捉到。而这些错误大多没有明显的症状,时隐时现,增加了改错的难度。

常见的内存错误及其对策如下:

内存分配未成功,却使用了它。

常用解决办法是,在使用内存之前检查指针是否为 NULL。如果指针 p 是函数的参数,那么在函数的入口处用 `assert(p != NULL)` 进行检查。如果是用 malloc 或 new 来申请内存,应该用 `if(p == NULL)` 或 `if(p != NULL)` 进行防错处理。

内存分配虽然成功,但是尚未初始化就引用它。

犯这种错误主要有两个起因:一是没有初始化的观念;二是误以为内存的缺省初值全为零,导致引用初值错误(例如数组)。

内存的缺省初值究竟是什么并没有统一的标准。因此无论用何种方式创建数组,都必须首先赋初值(即便是赋零值也不可省略,不要嫌麻烦)。

内存分配成功并且已经初始化,但操作越过了内存的边界。

忘记了释放内存,造成内存泄露。

含有这种错误的函数每被调用一次就丢失一块内存。刚开始时系统的内存充足,看不到错误。终

有一次程序突然死掉,系统出现提示:内存耗尽。

动态内存的申请与释放必须配对,程序中 malloc 与 free 的使用次数一定要相同,否则肯定有错误(new/delete 同理)。

释放了内存却继续使用它。有三种情况:

a 程序中的对象调用关系过于复杂,实在难以搞清楚某个对象究竟是否已经释放了内存,此时应该重新设计数据结构,从根本上解决对象管理的混乱局面。

b 函数的 return 语句写错了,注意不要返回指向“栈内存”的“指针”或者“引用”,因为该内存存在函数体结束时被自动销毁。

c 使用 free 或 delete 释放了内存后,没有将指针设置为 NULL。导致产生“野指针”。

2) 不能重现的错误,有些与代码中的未初始化变量有关,有些与系统不检查异常情况有关。加强对一般性错误的屏蔽功能。对输入数据进行充分并且有效的有效性检查,不合要求的数据不能进入数据库。

总之,内存的有效管理,指针、引用与变量正确地使用与控制,才能够设计健壮、安全、稳定的程序,有效地保证系统的稳定性,提高变电站自动化系统产品的质量。

## 3 结论

硬件资源是 IED 系统软件的载体,IED 系统软件是硬件的灵魂。在硬件条件具备的前提下,变电站自动化系统软件设计能够把硬件资源的全部潜能挖掘出来,能够使得变电站自动化系统通用、稳定以及灵活,能够满足用户的特殊需求,提高产品的质量和厂家的声誉,提高工程安装与维护的效率。

收稿日期: 2004-12-31; 修回日期: 2005-01-25

作者简介:

李洪波(1969-),男,硕士,讲师,主要从事信息管理系统、网络规划与管理、嵌入式系统开发、面向对象的分析与设计等方面的研究;E-mail: fast-run-man@126.com

张吉赞(1973-),男,硕士,讲师,主要从事信息安全、软件工程、算法分析与设计等方面的研究;

崔建国(1974-),男,硕士,软件工程师,主要从事变电站 RTU 的开发与维护、变电站监控系统的开发与维护等方面的研究。

(下转第 69 页 continued on page 69)

## 参考文献:

- [1] 张建功, 杨子强, 王建彬, 等. 配电自动化实用模式探讨 [J]. 电网技术, 2003, 27(1): 80-83, 87.  
ZHANG Jian-gong, YANG Zi-qiang, WANG Jian-bin, et al Research on Several Practical Modes of Distribution Automation [J]. Power System Technology, 2003, 27(1): 80-83, 87.
- [2] 林芳, 邱振敏. 浅析环网柜在城网改造中的应用 [J]. 电气开关, 2002, (1): 31-33.  
LN Fang, QIU Zhen-min Elementary Analysis for Application of Looped Network Cabinet in City Network Transform [J]. Electric Switchgear, 2002, (1): 31-33.
- [3] 黄宇红, 孙少陵. 通用分组无线业务 [J]. 电信科学, 2002, (5): 7-10.  
HUANG Yu-hong, SUN Shao-ling General Packet Radio Service [J]. Wireless Communication Technology, 2002, (5): 7-10.
- [4] 邓志成. GPRS技术与市场 [J]. 通信世界, 2000, (5): 27.  
DENG Zhi-cheng GPRS-Technology and Market [J]. Communication World, 2000, (5): 27.
- [5] 钟志伟, 冯文胜. 浅谈 10 kV 环网柜在城市配电网中的应用 [J]. 广东电力, 2001, 14(4): 45-46, 49.  
ZHONG Zhiwei, FENG Wen-sheng Application of 10 kV Switchgear Cubicle for Loop Network to Urban Power Distribution Network [J]. Guangdong Electric Power, 2001, 14(4): 45-46, 49.
- [6] DSP56F801/803/805/807 16 - Bit Digital Signal Processor User's Manual MOTOROLA [Z]. 2003.
- [7] Poly Phase Multifunction Energy Metering IC with per Phase Information Rev 0 ANALOG DEVICES [Z]. 2003.

收稿日期: 2004-12-08; 修回日期: 2005-03-16

## 作者简介:

刘翔宇 (1982 - ), 男, 硕士研究生, 从事电力系统自动化、电能质量分析、谐波治理等方面的研究; E-mail: l\_xiuxy@163.com

王文成 (1968 - ), 男, 助理研究员, 从事智能控制、变电站自动化、微机自动化方面的教学与科研工作。

### Research on the advanced switchgear cubicle for loop network based on GPRS network

LU Xiang-yu, WANG Wen-cheng  
(China Agricultural University, Beijing 100083, China)

**Abstract:** The distribution of the urban power distribution network lead to incredible amount of investment and poor credibility of the communication by means of coaxial cable and optical cable. To solve this problem, a new type of advanced switchgear cubicle based on GPRS network is developed. The central part of the terminal is DSP. The voltage, current, active power, reactive power and the fault of the power cable of three phase can be monitored. It is the equipment that comprise switchgear cubicle, power meter and SVC. It takes the GPRS network as its communication channel with the advantage of real-time, high reliability and low cost.

**Key words:** switchgear cubicle; GPRS; FTU

(上接第 61 页 continued from page 61)

### Practical design of intelligent substation automation systems

LI Hong-bo<sup>1</sup>, ZHANG Ji-zan<sup>1</sup>, CUI Jian-guo<sup>2</sup>  
(1. College of Mathematics and Information Science, Yantai Normal University, Yantai 264025, China;  
2. Dongfang Electronics Group, Yantai 264001, China)

**Abstract:** The problems with IED in transformer substation automation systems (SAS) on adaptability, stability and general use are analyzed. Adaptability of software in SAS with interface object and template file are enhanced. A more reasonable scheme for separating user from source code of IED, which can define new computing rules according to user's special requirement and improve the adaptability of IED's function, is proposed. The factors that affect the stability of software in transformer substation automation systems (SAS) and the corresponding solutions are discussed.

**Key words:** adaptability; stability; general use; interface object; template file; parameter file