

基于 Qt 的继电保护逻辑组态软件开发

陆玉军, 李澄, 陈颢

(江苏方天电力技术有限公司, 江苏南京 211102)

摘要:根据继电保护装置现场灵活应用的需要,利用 Qt 强大的图形界面开发功能,设计开发了可支持逻辑组态的继电保护组态工具软件。介绍了该工具软件的开发架构和方案,并给出了软件的图元编辑组态设计、组态逻辑合法性检查排序、生成源码和工程编译、模块开发的详细实现过程,利用该工具已实现了继电保护装置的逻辑组态。

关键词:组态软件;Qt;继电保护;逻辑组态

中图分类号:TP311.52

文献标志码:A

文章编号:1009-0665(2015)05-0043-05

当前继电保护装置的软件开发模式仍以传统的开发模式为主,即根据一次设备保护实现功能要求而设计开发,由于不同现场的差异,同一类设备的保护功能要求往往难以完全一样。因此,需要根据实施现场不断完善,这不仅导致开发的工作量大、代码重用率低、装置测试过程繁琐,而且由于开发人员的后续修改补充,很容易在软件中植入新的缺陷而难以发现。

组态软件利用计算机软件的各种资源,按特定需要进行配置,能适应各种现场需要,具有极大的灵活性,代码重用率高,实施成本低,在集散控制系统(DCS)中作为监控组态软件得到了大量应用,如 Wonderware 公司的 InTouch,其地位日益重要,已成为自动化领域的主流软件^[1]。目前组态软件在国外的使用非常普遍,除了电厂自动化领域外,在工控领域、科学研究、系统仿真计算领域、电子 IP 内核设计、及各跨国大公司的设备中都大量使用,电力系统继电保护领域国外厂家也推出可针对自身产品组态的工具软件。国内组态软件主要集中在工控领域、电厂自动化领域,比较著名的工控软件有:昆仑动态的 MCGS、亚控科技的组态王。国内继电保护设备厂家则还没有开发出成熟的组态软件,在这方面虽然很多单位做了一些具体工作,但至今未见投入实际使用的工具软件^[2-5]。继电保护逻辑组态软件具备功能模块图形化显示,便于现场人员学习掌握设备原理;现场可灵活组态易于满足现场需求的变化;可实现运行时内部状态监视具有维护方便的特点,有着良好的应用前景。

1 利用 Qt 组态开发的优势

Qt 是由 Trolltech 公司开发的跨平台 C++ 的图形用户界面应用程序开发框架,提供给应用程序开发者建立艺术级的图形用户界面所需的所有功能。它是面向对象的框架,易于扩展,且允许组件编程,除优良的跨平台特性、丰富的应用程序接口(API)和大量的开

发文档、XML 的支持等特点外,在图形组态程序开发方面还有如下的明显优势^[1,6]:

(1) 项视图体系。基于视图(QGraphicsView)、场景(QGraphicsScene)、项(QGraphicsItem),Qt 引入了全新的视图体系,提供了一个基于项的高级接口,非常易于实现对项的操作和绘制、独立变换、管理,这种视图体系完全适用于组态软件的图形组态。

(2) 独特的事件响应机制。信号槽(signal/slot)驱动机制是其一项关键技术,该方法取代了传统图形用户界面(GUI)的回调函数,实现了对对象间的通信,该响应机制在图元对象操作中使用很方便。

(3) 支持快速图形界面程序开发。基于 Qt 设计工具 Qt Designer 生成图形用户界面,可利用图形界面生成工具快速生成应用程序代码,是一种快速开发应用程序的方法。将代码开发编译所需工具以插件的形式集成在开发者熟悉的 VS2008 及以上版本的开发环境中,非常易于项目开发。

2 组态软件架构和开发方案

典型 DCS 控制组态软件架构如图 1 所示,这类组态软件一般分组态开发环境和运行环境,开发环境实现监控画面组态、控制逻辑组态、控制设备组态及数据库组态,运行环境实现组态工程运行和监视控制,实时数据信息显示。

为继电保护装置保护逻辑组态而开发的设计工具,同样具有开发环境和运行环境,但开发环境在 PC 机上,运行环境则在保护装置内,是一种交叉环境的开发方式。装置的组态逻辑由 PC 机开发编译完成,然后再将目标程序下装至装置内运行,组态软件可支持装置运行时的监控;为简化设计和提高装置运行效率,组态软件也可不实现监控装置运行。文中将模块开发和组态开发进行了分离设计,使组态软件具有较好的扩展性和人机交互特点,结构如图 2 所示。组态软件整体上分模块开发和组态开发两大部分。

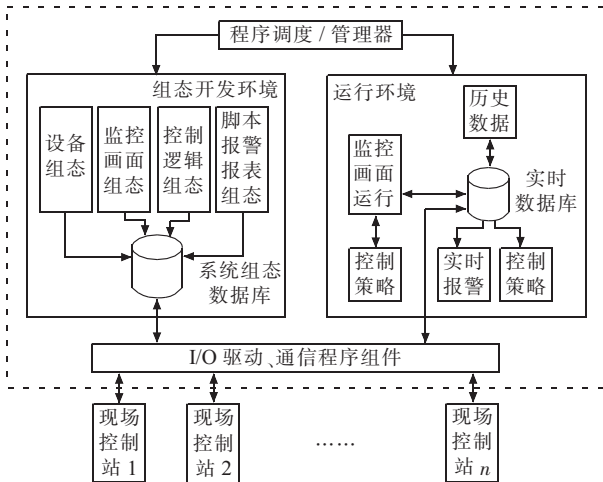


图1 组态软件架构

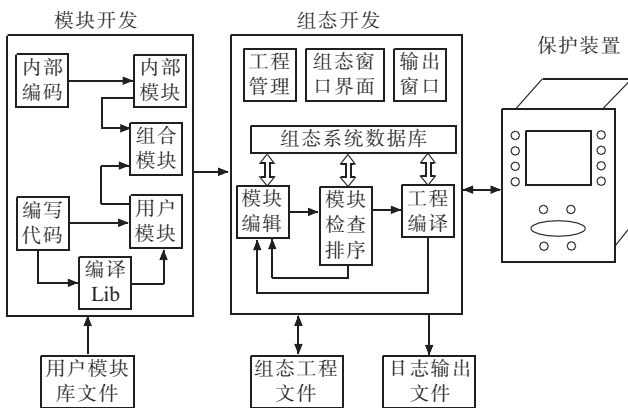


图2 继电保护逻辑组态软件

(1) 模块开发。模块是继电保护设备逻辑功能分解的结果,设备实现则是将相应的模块重新组合。根据设备功能,模块主要分为3种类型:内部模块(基本模块、接口模块、信号或数据输出模块)、用户模块(各种类型保护逻辑、控制功能模块)、组合模块(前两者的任意组合,可简化模块组态界面)类型。模块由用户编写符合接口规范的C++源码文件或经编译后的源码lib库提供,模块库中的模块载入调用则是由与库关联的接口文件实现。

(2) 组态开发。在上图的组态开发环境中,组态工具提供人机交互窗口时,主要是实现3大功能。① 模块编辑:根据要实现的目标装置逻辑功能,导入用户模块到模块管理区,从模块库管理区中拖放模块到组态窗口中,用导线连接模块接口间的连线,及通过完成相应的编辑操作实现逻辑组态。② 检查与排序:检查模块间的连接状态、接口数据类型的匹配、然后根据合法性检查结果进行模块计算的先后次序排序,并输出检查失败、类型匹配错误、产生信号环路的错误告警信息及图形提示。③ 工程编译:将检查通过的组态工程根据模块顺序生成目标源码文件,并完成文件编译生成可下装到装置的目标代码。工程编译时给出相关的编译错误告警信息与提示。

3 组态图形编辑开发

3.1 基本图元类型

在组态视图中,构成组态的基本图元对象分为3种:导线(用连线表示,用于实现模块间的连接),功能模块(由各种形状的图元表示)、文本(标识、解释或提供数据显示),上述模块分别由项视图体系中的QGraphicsLineItem、QGraphicsItem、QGraphicsTextItem类派生实现。每个图元的正确显示或响应事件的操作则需重新实现基类中的绘制函数paint()、形状函数shape()、碰撞检测boundingRect()。每个图元均有多个属性以支持显示、操作(如支持翻转操作等)及动画响应的函数。在保护组态图中,模块相当于一个保护装置的某个具体功能模块,如低压线路保护装置的复压方向过流保护可用一个模块表示,导线相当于功能模块对应输入输出间的信号传导关系,文本为便于显示或理解而添加的文字信息。

3.2 编辑操作

编辑操作主要是实现模块图元拖放、导线绘制、文本放置等基本操作要求。除上述操作外,还支持图元添加、删除、剪切、拷贝、移动、修改属性及撤销/恢复等操作。

图元移动,有带橡皮线(rubberline)移动或不带橡皮线移动操作,带橡皮线功能避免了部分图元移动导致的连线中断,实现了连线自动跟随模块移动变化。

3.3 事件响应设计

事件响应设计在图形编辑中有鼠标、按键、菜单响应等操作。图元选中在查找元件、故障定位图元时显得尤为重要,可帮助快速定位目标。选中复制的图元被粘贴前,可随鼠标移动过程中概要显示将要粘贴的图元虚拟图形,有助于粘贴时精确定位。hover事件可使鼠标移入或移出图元区域时发生相应变化,为捕获目标点位提供清晰提示。

3.4 智能化操作

视图智能显示需更多的智能操作。视图操作:为使图元清晰可辨或显示整体布局,对显示的视图区要求能够自动实现缩放,且缩放操作支持多样化。视图背景:良好的视图背景有助于减轻组态编辑时的压力并减少编辑错误的概率,实现视图背景切换也有必要。框选图元:鼠标划定一个矩形框,在框内的图元均被置为选中状态。边界识别:图元在拖放、绘线、移动、粘贴等操作中能准确识别图形边界,使得操作总能控制在边界内而不越界。

3.5 组态工程管理

分页显示:由于视图/场景显示区域易受窗口大小限制,在单个视图窗口中往往难以容纳一个保护测

控装置的组态逻辑中所需的数量众多的图元,此时需要扩展多个视图,采用分页模式显示,在保护逻辑设计中可增减分页。分页的另一好处是可将相关性大的模块集中在一个页面中,以使保护逻辑组态更清晰。

工程管理:在组态工具软件的组态环境中只容纳一个装置的组态工程将使得进行操作或编辑查看变得非常繁琐或无法实现,需要可对多个组态工程进行树状式管理,在继电保护组态中,一个工程往往表示一种类型装置,如一个工程表示低压线路保护、另一个表示电容器保护。从方便设计角度看,采用一个相对独立的 MDI 子窗体管理工程较为合适,多个工程通过用不同的子窗体分别管理,对子窗体的操作实际是对不同工程的操作。工程管理的结构如图 3 所示。

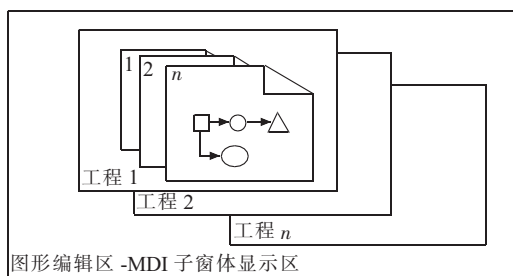


图 3 工程与视图管理

4 检查排序和工程编译设计

图形编辑完成后,需要对组态图形进行检查校验,最终以正确的模块连接关系、确定模块计算序列才能正确生成目标源码,完成编译输出可执行目标代码。

4.1 检查排序

检查的规则应能准确查找出组态编辑中的疏漏或错误,是保证正确的一种手段。该软件主要根据保护功能模块的特点而设计规则:所有模块输入必须连接、与模块输入相关的驱动必须唯一、模块输入数据类型必须与驱动信号一致等。

通过对网络表的搜索处理,可获得模块连接关系,从而为模块输入检查、数据类型校验、模块排序、信号环路校验提供了支持。

在校验过程中执行每一步均输出校验信息,如前一步校验失败,则停止后续步检查校验,并给出当前步骤全部错误信息且提供错误的模块或连线快速定位选中。校验结束,给出排序结果和信号环路提示,提醒组态人员及时处理环路计算问题。在编辑组态中在模块间非常容易出现计算环路,而实际计算逻辑时均按顺序执行,这是组态图所特有的现象,需进行解环处理。

4.2 工程编译

工程编译是将组态逻辑生成的源代码文件编译为目标代码的过程。

生成源码:根据排序的结果顺序提取模块的各自

源代码(或调用库)函数,并组织生成组态逻辑的源代码文件。

编译源码:根据生成的源码,通过调用装置 CPU 的专用编译工具,生成装置的目标代码文件。

生成代码文件和编译源码文件时每一步均给出错误提示或编译信息,提醒组态时注意编译错误或异常。

4.3 编译后的工作

编译后的目标代码,可以通过调试工具下载到装置内运行,或组态工具软件集成目标文件下载功能,运行目标程序。

实际上,编译后的代码可能存在模块前期调试中未发现的错误、或由于代码过大或计算过于复杂,导致装置无法运行,此时应将组态后的代码进行充分调试,并给出装置运行代码的编译限制条件,确保组态工具能生成正确的可执行目标代码。

文中组态工具调用的保护模块是利用前期开发的模块代码并已经过细致的调试,未经调试的代码模块,应在模块开发阶段完成调试。否则将导致组态逻辑不能正确运行。

5 基于 Qt 的组态模块设计

在组态环境中模块不仅具有图元操作的所有属性,还具有可最终在装置内执行的代码和操作访问的数据,实际上是图元、代码、数据的综合体。图 4 为一低压线路保护的复压方向过流保护模块的图元、代码及数据属性。

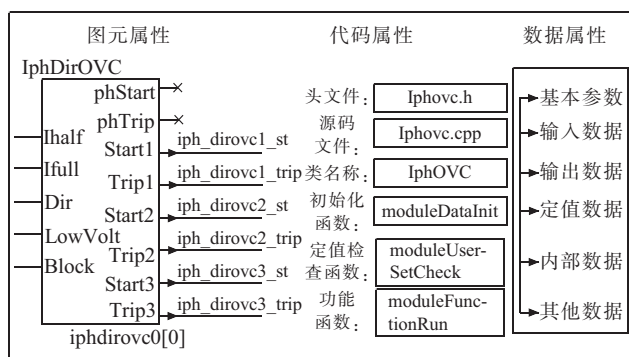


图 4 模块属性

在编辑时,模块表现是具有相应接口的图元,输入输出分别对应其执行函数的输入条件和输出结果;在编译时即相当于对其输入数据初始化关联,并将其输出作为其他模块的关联输入。而模块定值等需与用户交互的数据则对应为装置内部数据库数据的一部分。

所有类型的继电保护装置都由相关的计算、保护、测量、控制、录波等功能组成,即按功能可分解为多种不同模块与多个同样模块的组合,如在装置内实现特定保护功能则需为其设计一个与其功能相当的模块。

根据前述分析,模块开发要点为以下方面。

(1) 功能模块划分。以低压线路保护为例,可分解为:实现装置接口的模块(输入输出接口模块)、采样计算的模块(FFT 或 DFT 等算法模块);针对实现继电保护功能的模块(三段式相间过流保护、过流反时限保护、三段式零序过流保护、零流反时限保护、加速保护、三相一次重合闸等模块);实现测量的模块(频率测量、功率计算、有效值、相角、谐波计算等模块);实现控制的模块(保护出口控制、开关遥控、软压板控制、直控等模块);实现事件管理的模块(告警事件报告、跳闸事件报告、变位事件报告等模块);其他模块(基本逻辑、基本运算、数据或信号输出、故障录波、告警信息显示等模块)。

(2) 代码与数据结构设计。模块数据结构方面,接口数据非常重要,应涵盖所有需要计算所用的输入数据,但也不宜简单对待(如三相一次重合闸的闭锁条件很多,将每个闭锁条件均作为一个独立输入不合适),输出接口数据应以模块需要输出与其他模块交互数据为基准,否则模块接口显得非常庞大,内部定值数据、需要监控的模块信息数据等结构的设计,同时应满足组态编辑时对数据的操作要求。

为使设计的模块能正确实现模块功能,通常含有初始化函数、运行函数、定值检查函数。

初始化函数:在编译和运行时均需对其数据进行初始化,编译时为输入输出数据建立正确的存储地址、初次运行时以便将模块进行正确的初始条件设置。

运行函数:在每次执行时,根据输入条件执行相应的运算得到模块输出,将模块输出传至下一模块,或做装置的输出数据。

定值函数:执行与定值相关的功能,如定值修改、定值范围检查等。

(3) 图元开发。为便于理解图元的实现功能,要求图元的界面显示应设计得与功能相符。如图元模块名和对象实例名应便于理解、图元引脚接线应加以名称描述、图元引脚状态与连接与否关联、图元形状轮廓与功能相关等方面均需要详细设计。

其他图元属性:如支持变换方向、调整比例、修改名称、计算序号显示,还有响应选中、查找、快速定位等

辅助功能。

6 结束语

相比较采用传统方法开发的微机保护装置存在程序升级不便、内部逻辑不可见、监视内部状态困难等不足,组态软件则以图形模块清晰展现逻辑关系、通过重新组合模块即可实现装置升级工程应用灵活、在设备运行时又便于实现内部状态监测显示,优势显而易见。采用组态方式开发使得二次设备的开发设计变成了模块开发,开发时仅需保证各模块功能正确,即可达到整装置测试没有缺陷的可能,因此,测试成本显著降低、测试试验更容易。另外,多数现场升级的要求可通过已有模块组合实现,省去了大量反复开发测试和庞大软件工程管理的麻烦。组态软件更加具有可视性,使得差错或缺陷更加容易发现、处理缺陷的成本更低。采用组态软件设计实现保护测控设备,可以达到“一次开发,任意使用”效果,今后更多的二次设备厂家将会趋向于采用这一模式实现二次设备开发。

参考文献:

- [1] 李冬梅. 基于 Qt 的组态软件的设计与开发[D]. 大连:大连理工大学,2012.
- [2] 魏 巍,余群兵,陈 刚. 图形化控制算法组态在微机保护中的应用[J]. 江苏电机工程,2005,24(3):28-30.
- [3] 李 孝. 继电保护通用逻辑开发平台设计[D]. 南京:南京理工大学,2007.
- [4] 李 健,王云杰,梅兴虎,等. 组态式开发平台在微机保护开发中的应用[J]. 浙江电力,2009,(1):25-27.
- [5] 胡红兵,李丽君,韩民畴. 继电保护装置自动化测试系统的设计与实现[J]. 江苏电机工程,2013,32(2):25-27.
- [6] 闫锋欣等译. C++ GUI Qt4 编程 [M]. 2 版. 北京:电子工业出版社,2013:138-166.

作者简介:

- 陆玉军(1972),男,江苏句容人,高级工程师,从事变电站综合自动化系统的开发及工程应用工作;
李 澄(1970),男,江苏无锡人,研究员级高工,从事继电保护、变电综合自动化系统等研发工作;
陈 颢(1987),男,湖北钟祥人,工程师,从事变电站综合自动化系统的开发及工程应用工作。

Development of Logic Configuration Software for Relay Devices Based on Qt

LU Yujun, LI Cheng, CHEN Hao

(Jiangsu Frontier Electrical Power Technology Co. Ltd., Nanjing 211102, China)

Abstract: In order to adapt to the flexible application requirements of relay devices in fields, a novel graphic configuration program is developed based on the Qt software. The design scheme and developing frame is introduced and the realization of editable graphic components, configuration validation check, project compiling process, and modules design are explained. Finally, this developed tool is applied in relay devices development.

Key words: configuration software; Qt; relay protection; logic design