

Board Inspection System Based on Object-Oriented Technology of LabVIEW

Yugang Wang, Xingtang Zhao, Yuanlei Li, Xuhua Dong

Naval Aeronautical Engineering Academy of Qingdao Branch, Qingdao
Email: flyingmanwang@163.com

Received: Jan. 16th, 2013; revised: Jan. 31st, 2013; accepted: Feb. 7th, 2013

Abstract: This article introduces the certain board inspection system based on object-oriented technology. Through the methods of graphical object-oriented programming, to separate this system into several discrete models. Using the merits of object-oriented sufficiently to realize inspecting boards eventually.

Keywords: LabVIEW; Object-Oriented; GOOP; Inspection System

基于 LabVIEW 面向对象技术的某型电路板测试系统

王玉刚, 赵兴堂, 李元垒, 董绪华

海军航空工程学院青岛校区, 青岛
Email: flyingmanwang@163.com

收稿日期: 2013 年 1 月 16 日; 修回日期: 2013 年 1 月 31 日; 录用日期: 2013 年 2 月 7 日

摘要: 本文介绍了基于 LabVIEW 面向对象技术的某型电路板检测系统, 通过面向对象程序设计方法(Graphical Object-Oriented Programming), 将现有代码分解为一个个离散的模块, 充分利用面向对象设计与开发的优点与好处, 最终实现电路板检测任务。

关键词: LabVIEW; 面向对象; GOOP; 测试系统

1. 引言

LabVIEW 是虚拟仪器集成开发环境的总称, 它是为替代常规的 Basic 或 C 语言而设计的, LabVIEW 是编程语言而不仅仅是一个软件开发环境。作为编写应用程序的语言, 除了编程方式不同外, LabVIEW 具备语言的所有特性, 因此又称之为 G 语言^[1]。G 语言是一种适合应用于任何编程任务、具有扩展函数库的编程语言。同时 G 语言丰富的扩展函数库还为用户编程提供了极大的方便。这些扩展函数库主要面向数据采集、GPIB 和串行仪器控制以及数据分析、数据显示和数据存储。G 语言还包括常用的程序调试工具, 比如容许设置断点、单步调试、数据探针和动态显示执

行流程等功能。G 语言与传统编程语言最大的差别在于编程方式, 一般高级语言采用文本编程, 而 G 语言采用图形化编程方式。

近年来, 面向对象的思想逐渐在程序设计领域占据主流, 并且广泛的应用于系统的分析、设计、测试等领域。今天, 同样的发展也发生在图形化的 G 语言身上^[2]。LabVIEW 的 G 语言本身是一种基于数据流的图形化语言, 在过去, 它在中小型的测试系统开发中具有快速直观的优点。但随着技术的快速发展, 越来越多的测试工程师倾向于在基于数据流的 G 语言上使用面向对象的技术来增强软件的升级性、开发性、代码重用性, 我们称之为 GOOP(Graphical Object

Oriented Programming)。

2. GOOP 的特性

通过对面向对象的支持，用户能够利用面向对象的设计方法和工具来设计和分析系统，例如 UML 统一建模语言、Rational Rose 设计工具等。这使得通过 LabVIEW 开发和维护大型工程变得更加容易。图形化的面向对象编程使基于组件的开发方式更容易得到实施，因此极大地扩展了传统基于数据流的 LabVIEW 的能力。GOOP 的特性总的来说，归纳如下：

2.1. 便于升级与维护

LabVIEW 传统的编程范例是自顶而下进行功能性分解，随着程序的增大，模块数目增多，由于模块间的关系是强耦合性的，因此对其中任一个模块的改变都可能引起设计者所意想不到结果。而对象的核心是被封装的数据空间以及私有方法，它与外界的接口由一组公有方法的调用来实现，这一系列的可编程接口能够有效的遏制其间的依赖性^[3]。因此面向对象技术(OO)具有低耦合、高内聚的特性，对象之间采用松散连接，具有非常好的升级性与维护性。

2.2. 代码复用性

非面向对象技术(OO)环境下，从应用层到全局数据空间，严重的数据依赖性大大减少了潜在的代码复用。但是代码复用对于面向对象技术(OO)程序非常简便。因为对象的接口是一组已经定义好的公共方法，这使得设计者能够轻松的决定复用构件的功能性。程序设计者通过类所定义的公共方法来进行数据访问，

这样也大大减少了在代码复用时的误用风险。

2.3. 易开发性

传统的功能性模块分解方法虽然解决了软件开发人员的合作问题，但仍然存在着大量的协作问题。而对于面向对象的设计，系统可以以自然的方式分成几个类，这些类通过属性和方法调用来与其他类联系起来，这样开发者就能够各自独立的开发驱动类、用户接口类、逻辑运算类，而不用像功能性分解那样关心子程序之间的连接问题并使得调试工作更加容易。

3. 基于面向对象技术的某型电路板测试系统

某公司生产如下三种电路板。A：基本数据采集电路板；B：精密数据采集电路板。

公司主要生产两种类型的电路板，且每块电路板均有电阻、电容和芯片 3 种元件组成，需对 3 条流水线上每一块板卡进行检测来确定电路板上各个元件位置是否摆放正确。因此该系统在每条流水线上放置了一个摄像头，它们将电路板的图片发送到同一台计算机上，该计算机通过检测程序来判断电路板是否安装正确。系统的构成如图 1 所示。

3.1. LabVIEW 的层次结构

针对该检测任务，该系统显示了一个基于面向对象的解决方式^[4]。在 LabVIEW 中新建的类有个默认的父亲：LabVIEW Object。LabVIEW Object 是 LabVIEW 中所有类的“终极”祖先，所有的类都是它的子类。对于该型电路板测试系统，可以分为电路板设计类和

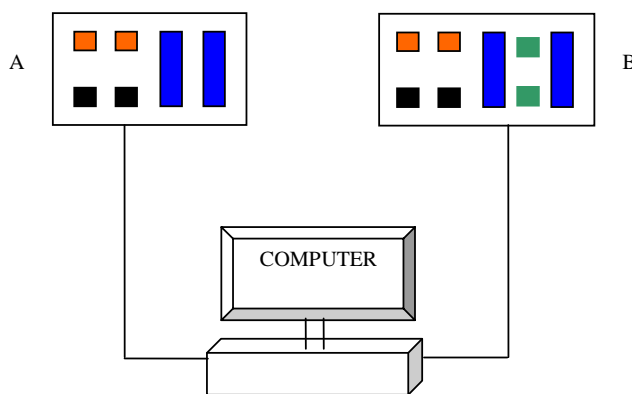


Figure 1. System configuration
图 1. 系统结构图

电路板部件类。电路板设计类中包含数据采集电路板设计类,而该公司生产的具体的 A 电路板和 B 电路板属于其中的子类。具体的类层次结构图如图 2 所示。

3.2. 电路板检测系统的实现

在面向对象的解决方案中,将该电路板检测系统分为相对独立的很多模块,面向对象的方法采用了更多的子 VI^[5]。LabVIEW 中的子 VI(SubVI)类似于文本编程语言中的函数,通过子 VI,可以把程序分割为一个个小的模块来实现。其中每一个子 VI 都对应一个非常具体的任务,因此子 VI 的程序框图更小,而子 VI 的数目更多。该系统具体的实现框图如图 3 所示:在该电路板检测系统中,包含产生测试的图片子 VI、获取测试的图片子 VI、选择类子 VI、检测输入的图片子 VI 等等。

其中,选择类用于根据输入的电路板的类型产生相应的对象^[6]。而检测输入的图片用于检测输入的对象。该电路板检测系统通过调用电路板对象的动态方法 Get Components.vi 获得相应电路板对象的元件数组,再通过调用每一个元件对象的动态方法 Self Test.vi 进行自检测。最后将电路板上每一个元件的自检测结果求“与”就得到了电路板的测试结果。

当该系统需要进一步改进时,采用面向对象的方法则会显示出更大的优势来。采用面向对象的解决方案只需要修改非常少的地方,这就使得程序更改更加容易,还极大的减少了由于修改可能带来的潜在错误。

4. 结论

显然,面向对象的解决方案容易被升级以适应新

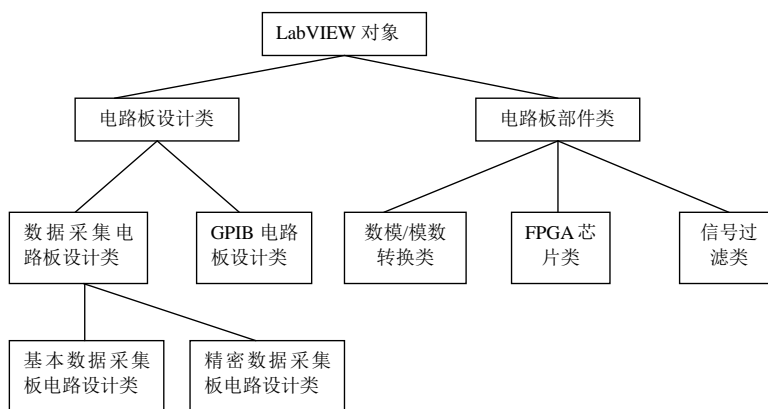


Figure 2. Class hierarchy diagram of LabVIEW
图 2. LabVIEW 的类层次结构图

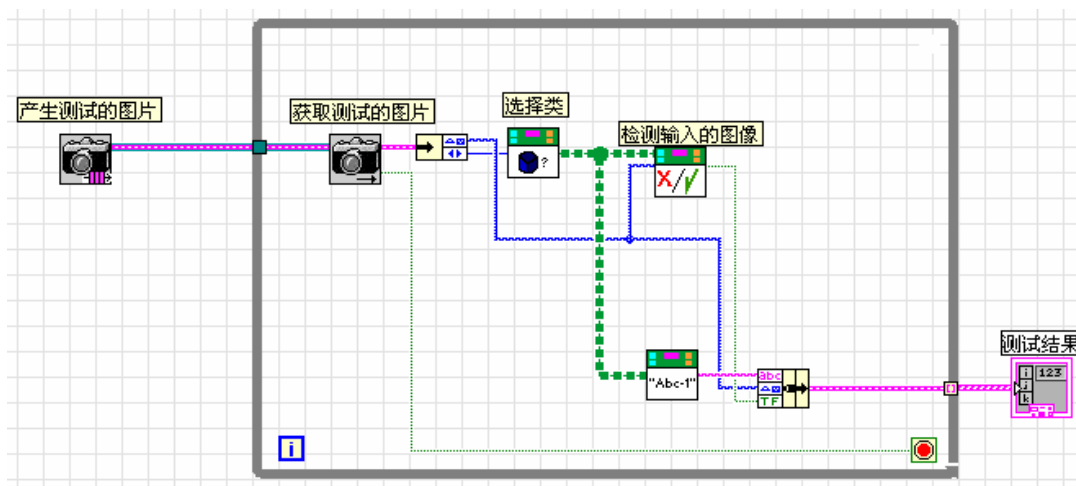


Figure 3. Bard inspection system
图 3. 电路板检测系统

的变化。一般来说,面向对象的优势在程序的第一版本中并不能得到太多体现,但是在以后的更多修改版本中,面向对象带来的代码易修改性以及代码的健壮性将会得到淋漓尽致的展现。GOOP 在 LabVIEW 中的应用,显著的提高了测试系统的开发性。随着技术的不断成熟,LabVIEW 平台下的中大型测试系统开发,终将转向 GOOP。

- [2] 徐洪安,费仁元,王民. 用 ADO 构建 LabVIEW 中的数据库访问接口[J]. 北京工业大学学报, 2003, 29(2): 138-140.
- [3] 雷振山. LabVIEW 8.2 基础教程[M]. 北京: 中国铁道出版社, 2008.
- [4] National Instruments Corporation. IMAQ vision concepts manual, 2000.
- [5] 李建文,刘笃喜,朱名詮. 基于 ADO 技术的 LabVIEW 访问数据库的方法[J]. 自动化仪表, 2003, 24(9): 15-17.
- [6] 贾小丽,郭钟宁等. 基于 LabVIEW 和 IMAQ 的微型电感器的焊点质量检测系统[J]. 机电工程技术, 2007, 36(10): 44-47.

参考文献 (References)

- [1] 杨乐平,李海涛,赵勇等. LabVIEW 高级程序设计[M]. 北京: 清华大学出版社, 2003: 470-485.