

基于组态的试飞实时监控技术

贾雨¹, 王爽², 覃杨森¹, 齐禅颖¹

(1. 中国飞行试验研究院, 陕西 西安 710089;

2. 西安远方航空技术发展总公司, 陕西 西安 710089)

摘要:针对当前飞行试验强度不断加大,验证科目特别是风险科目越来越多,试飞实时监控软件设计时间紧、修改频繁的问题,提出了基于组件的试飞自主实时监控平台软件设计方法。通过采用组件式设计思想,利用组件式方法在 Visual Studio 平台下开发实现了基于组件的自主监控平台。解决了以往“一个监控科目,一套监控画面软件”的技术瓶颈问题,改变了以往实时监控软件开发代码复杂庞大、重复性开发工作量巨大、不具备根据新要求实时动态修改的能力等固有缺陷。软件实现了图形元素可重组、参数算法可定义和兼容多个实时系统平台,用户可以利用组态软件的功能,不用进行繁琐复杂的编程操作,只需要通过鼠标点击创建所需功能组件并进行相应的参数配置,即可构建一套最适合自己的实时监控软件。经过飞行试验使用验证,证明了该平台能够实现安全监控画面的动态设计,同时能够根据试飞需求对监控画面进行快速调整,有效提高了试飞监控准备效率。

关键词: 飞行试验;实时监控;监控画面设计;组态软件

中图分类号:TN06

文献标识码:A

文章编号:1673-629X(2017)09-0145-05

doi:10.3969/j.issn.1673-629X.2017.09.032

A Real-time Monitoring Technology of Flight Test Based on Configuration

JIA Yu¹, WANG Shuang², QIN Yang-sen¹, QI Chan-ying¹

(1. Chinese Flight Test Establishment, Xi'an 710089, China;

2. Xi'an Yuanfang General Aviation Technology Development Corporation, Xi'an 710089, China)

Abstract: In view of the problem that flight test strength increasing, verified subjects especially risk subjects becoming more and more currently, and real-time monitoring software in tight design time and modified frequently, a software design method for autonomous real-time monitoring platform based on components is proposed. By adopting the thought of component designing, an independent monitoring platform is developed by using the component method on the platform of Studio Visual, which has solved the technical bottleneck of "a monitoring subject, a set of monitoring screen software" and changed the inherent defects of complex and huge developing code for previous real-time monitoring software, heavy workload of repetitive development, and not having dynamic modification in accordance with new requirements. It is realized for software that the graphic elements are restructuring, the algorithm parameters are defined and real-time system platform is compatible. Users can use the functions of configuration software, and without the complex programming operation, only need to create the required functional components by mouse click and conduct the corresponding parameter configuration, which can build a real-time monitoring software most suitable for their own. After flight test, it is proved that the platform can realize the dynamic designing of the safety monitoring picture and adjust the monitor screen quickly according to the test requirements, which can effectively improve the flight test monitor preparation efficiency.

Key words: flight test; real-time monitoring; monitoring picture designing; configuration software

0 引言

飞行试验遥测监控技术一直是国内外飞行试验领域研究的关键技术之一,是实现飞行试验实时监控、确

保试飞安全、提高试飞效率、缩短试飞周期的重要手段。近年来,随着国内试飞型号任务的增加,实时监控课题也越来越多。为满足日益增长的实时监控需求、

收稿日期:2016-08-31

修回日期:2016-12-07

网络出版时间:2017-07-05

基金项目:国防基础科研项目(A0520132031)

作者简介:贾雨(1988-),男,硕士研究生,研究方向为飞行试验实时监控及数据处理。

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1450.TP.20170705.1650.046.html>

提高监控效率,如何提升“一个监控科目,一套监控画面软件”的实时监控模式,已成为飞行试验实时监控系统技术研究中重点解决的难题。另外,随着飞行试验“智能化监控技术”研究方向的提出,对地面飞行实时监控系统的开发在“多目标多科目实时自主监控”方面提出了新的目标,并需要进行更广、更深、更细的研究,有规划分阶段逐步突破所面临的技术难题。

近年来,国内外基于组态^[1-2]的监控技术取得了飞速发展,应用领域也日趋拓展,吸引了国内外众多自动化公司纷纷投资开展相关技术的研究。组态监控软件具备灵活、便捷的开发环境,集成了非常多的监控功能,因而可以适应不同工业自动化领域,具有很强的通用性。其通用性特征,为其赢得了广泛的应用领域,但是同时也带来了一定的后果:组态监控软件产品集成了非常多的功能,使得系统过于庞杂,开发成本和使用成本急剧增加,而对于某一特定应用环境,提供的许多“大而全”的功能对于多数用户并不适用,造成极大的资源浪费。通用式的组态软件,由于其开发面向各行业用户,没有针对某行业进行特定的优化,导致行业开发人员在开发某一专用系统时一些专业性的 ActiveX 控件^[3-4]和特殊的 I/O 设备的驱动需要手动添加,与基于传统程序设计语言开发相比,实际上并没有减轻多少工作量。据统计,监控组态软件产品的功能只能发挥 30% 以下。遥测实时监控系統针对飞行试验地面实时监控的特定需求,其数据接口、数据传输模式^[5]、监控软件组成结构、数据表现形式都有其自身特点。针对试飞实时监控的特殊需求,目前还没有一款相关软件能够满足实时监控任务。

针对上述问题,该项目借鉴工业化组态软件设计方法^[6],采用模块化设计思想,研发基于组件的自主监控平台,实现了组件对象可重组、参数算法可定义,为数据处理工程师提供快速构建监控软件的平台。平台使用灵活的组件方式(而不是编程方式),使用户能根据监控需求任意组态,设计完成实时监控软件。项目中根据实时监控中的实际需求,编写设计一套完整的数字、曲线、仪表、柱状图、飞机平显及通用电子地图等监控组件,数据处理工程师能够按照实时/预处理任务书中监控参数的要求表现形式选择相应组件并进行相应的参数配置,即可快速完成监控软件的设计,而不要数据求处理工程师具备专业的编程知识,有效提高了监控软件准备效率,降低了监控软件设计的难度和复杂度;在监控过程中,数据处理工程师、试飞工程师可以根据任务需求实时进行监控参数的调整;此外,该平台实现了数据处理常用的基本算法,满足常规的数据处理需求,使得监控参数以正确的形式呈现给试飞工程师及地面指挥人员。

1 软件总体架构设计

自主实时监控组态软件平台以功能模块划分为设计主导思路,划分为三个功能模块:数据 I/O 模块、运行管理模块、图形界面模块。三个部分接口清晰、功能独立,总体架构如图 1 所示。

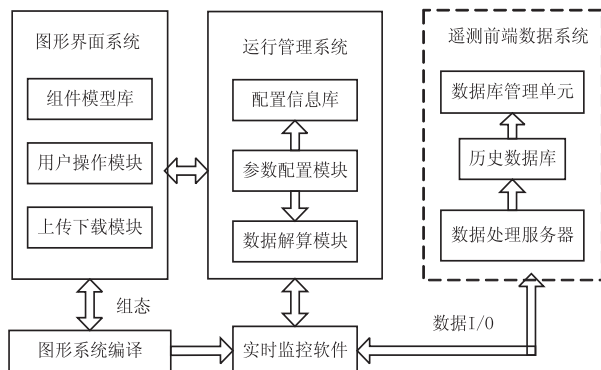


图 1 软件总体架构图

(1) I/O 驱动模块:是试飞实时监控组态软件中必不可少的部分,由于和 I/O 设备通信,互相交换数据,包括接口模块、实时通信模块等。它是实时监控软件与遥测前端数据处理系统的桥梁,是该软件的基础,为整个软件提供数据来源,向下为实时监控客户端提供监控参数名,向上提供实时遥测数据。

(2) 运行管理模块:用于管理和维护工程中的各种信息,如监控软件通信配置信息、监控页面参数配置信息等,同时还将接收到的网络数据处理为可理解和便于进一步利用的结果,并把处理结果递交给图形界面模块以驱动监控组件。此外,该模块还负责新建工程、工程备份等功能。

(3) 图形界面模块:是一个进行图形系统生成工作所依赖的开发环境。图形界面是监控系统的前端,通过建立一系列用户数据文件及操作,生成最终的图形目标应用系统。生成的目标应用程序可在图形运行环境中运行。在图形界面上还具备报警通知及确认、报表组态及打印、显示功能。各种报警、报表、趋势都是动画连接的对象。其数据源都可以通过图形界面指定,然后通过运行模块,取得各种处理好的参数,把监控对象通过仪表、曲线、告警灯等形式反映给试飞工程师。

平台主要实现的功能包括:

- (1) 实现监控画面软件的动态编辑设计;
- (2) 使用灵活的组件方式(而非编程方式),实现组件对象可重组、参数算法可定义,为数据处理工程师提供快速构建监控画面的平台;
- (3) 在监控过程中,数据处理工程师、试飞工程师可以根据任务需求实时进行监控参数的动态调整;
- (4) 实现试飞数据二次处理常用算法动态配置功

能,满足常规的数据处理需求。

2 软件功能模块设计

2.1 图形界面的设计及实现

图形界面模块为试飞人员提供了与平台交互的接口,是整个软件的核心部分,用户可以方便地通过鼠标拖拽组件的方式而不是编程方式进行控件的创建和配置编辑工作。监控画面的制作过程完全是一个可视化操作,利用鼠标等输入设备,动态创建所需要的监控组件并对其进行任意缩放和拖动位置以及进行参数和运算配置,从而实现用户对监控画面的自由组态。编辑好的画面进入运行状态后,各组件接收过程逻辑层解算好的参数实现实时状态刷新。该模块的实现主要包括监控组件模型库的开发^[3]以及组件的动态调用。

2.1.1 组件模型库开发

在试飞遥测实时监控过程中,需要通过数字、仪表、曲线、柱状图等多种数据表现形式对机上测试参数进行地面复现,自主实时监控组态软件需要搭建完备的组件模型库以供用户选择。因此,根据试飞监控中的实际需求,开发了一组基于 MFC 的监控组件,如仪表、曲线、柱状图等,以及飞机平显、地图等专用组件。监控组件的开发分为静态和动态两部分,静态部分没有参数驱动,一般为控件的基本属性部分(如背景、参数名称等)。动态部分的任务是以直观的动画来模拟现场,反映被控对象的变化,其通过实时接收的数据流来驱动。

监控过程中,通常需要创建很多监控组件,并对其进行移动、缩放等操作,这种情况下,会产生严重的闪烁现象,给用户带来很大的不便。产生闪烁的原因是:窗口需要重绘时,总是先用背景色将显示区域清除,然后才调用 OnPaint 进行绘制,由于背景色与绘图内容差别较大,就会产生闪烁现象。该系统采用双缓冲机制^[7]绘制图形,在绘图区域显示想要的画面之前,先在内存中创建一个与绘图区域一致的对象,将图形绘制到这个内存对象上,再一次性将这个对象上的图形拷贝到屏幕上,加快绘图的速度,从而避免闪烁现象。

中国飞行试验研究院遥测实时监控在几十年的发展过程中,开发积累了相当多的界面友好、功能强大的通用平台监控组件。软件平台中对常用的部分监控组件进行提取和封装优化,满足了基本的监控软件设计监控组件元素的需求。平台综合加载了自定义组件、专用组件以及通用平台组件,并对外提供统一接口形式进行管理,以实现组件的无差别动态生成、参数配置、运算配置与解析以及组件配置信息的存储与解析。同时,具有良好的扩展性,新开发组件按照接口定义规则可以方便地加入到平台中。丰富的组件选择为画面的

动态生成提供了极大的灵活性。

2.1.2 组件的动态调用

为了实现监控组件的动态生成,位置移动和缩放,软件在实现过程中为每一个监控组件添加一个运行时包装类。通过这种方法,可以实现将自定义组件如仪表曲线以及封装好的通用平台组件以统一的表现形式呈现给开发人员,实现二者的统一灵活加载。软件运行时,用户点击一个组件图标并拖放到指定位置,实际上就是为该组件的包装类创建一个对象并进行显示,同时可以用线框选中该组件,进行组件的移动、缩放,实现良好的人机交互。组件创建好之后,为该组件添加双击相应事件,在弹出的对话框中设置该组件的基本属性如组件名称及背景颜色等,配置要监控的参数以及参数的运算信息。每一个组件的创建与运行都是独立的,单个组件的修改与增删不会对系统的运行产生任何影响。图2是一幅监控画面的一般设计过程。

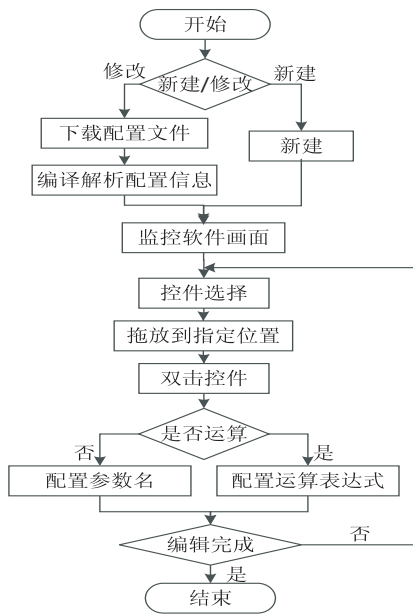


图2 画面生成过程

2.2 运行管理模块设计

2.2.1 参数运算动态配置及解析实现

参数解算模块接收到的 PCM 参数通常是由服务器软件已经解算好的物理量,对于该类参数,解算模块不做任何操作直接输出到该组件的数据驱动接口。但是,同时也存在着部分服务器无法完成解算或解算过于复杂的参数,如一个 PCM^[8-9]字中包含了多个开关量。对于这部分参数,则需在监控画面中进行计算以得到正确的物理量。遥测数据处理常用算法包括多 PCM 字合并、取位以及表达式运算,图3是一个组件完整的参数提取解算过程。

表达式动态解析^[10-11]是该模块实现的重点与难点。表达式解析是通过词法分析和堆栈操作对表达式进行重新组合,形成一棵表达式树,表达式计算是自上

而下地深度遍历表达式树,计算每一个树节点的值,根节点的值就是表达式最后的值。

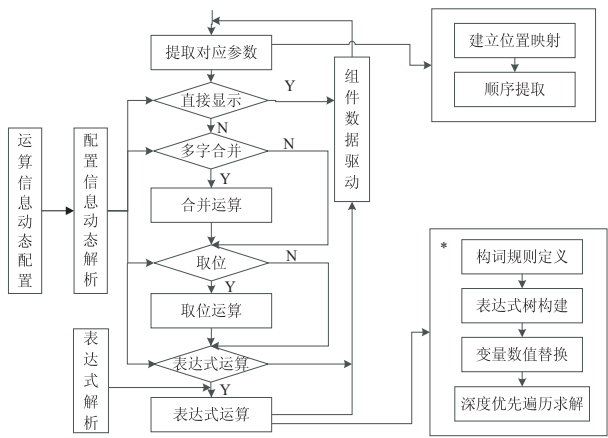


图 3 参数提取过程

2.2.2 配置信息的管理

在一个监控软件参数配置文件中,所有组件呈树状关系,如图 4 所示。在画面存储时希望也能够保持这种层次关系。该软件采用 XML(eXtensible Markup Language)文档来表示这种树状关系。XML^[12-14]定义了利用简单、易懂的标签对数据进行标记的一般语法,提供了计算机文档的一种标准格式。利用 XML 文档存储编辑好的监控画面,不但结构清晰,而且完全以文本方式存储,可以通过任何一种编辑工具进行查看和修改,使用户编辑十分方便。同样,当需要打开一幅画面时,就可以根据 XML 文件中的节点及其属性生成正确的图形组件。

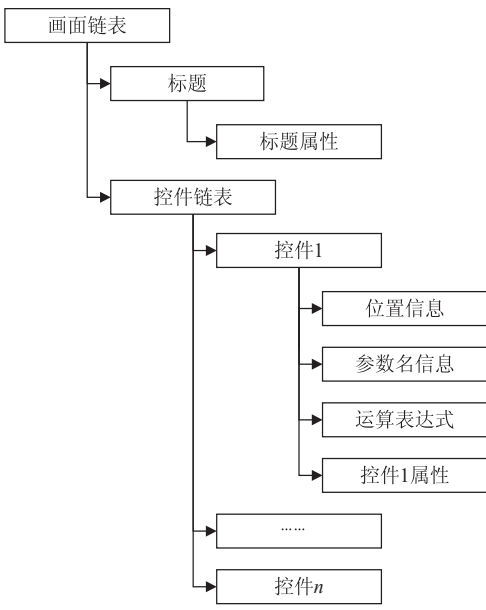


图 4 画面存储结构

对于不同型号不同科目监控软件的配置信息,采用结构化层次划分进行管理,通过型号及科目划分建立分层的配置信息库结构,从而实现所有监控软件配置信息的结构化存储管理。

2.3 I/O 驱动模块设计

系统通过建立 Socket 套接字的方式与遥测实时数据处理前端进行数据交换。当前,常用的网络数据传输协议包括 TCP(Transmission Control Protocol,传输控制协议)和 UDP(User Datagram Protocol,用户数据报协议)两种。TCP 是一种面向连接的协议,允许从一台计算机发出的字节流无差错地发往网络上的其他计算机。两台 TCP 主机间进行数据交换之前,必须通过三次握手机制先相互建立会话,通过确认和按顺序传递数据来确保数据的可靠传输,其在发生丢包时会进行数据包重传。TCP 一般应用在要求数据传输具有较高可靠性的场合。UDP 提供尽量传递的无连接数据服务,它不能确保或确认数据传递或数据顺序,由使用 UDP 的程序负责提供传输数据所需的可靠性。对同一个局域网来说,使用 UDP 能够获得良好的性能,它能够满足数据传输量不太大、数据实时性较高的应用场合,在这种情况下,UDP 的低开销和多播能力比 TCP 更合适。

试飞实时监控网络属于专用网络,网络规模较小,数据一般不需要在交换设备间进行多级跳转,因此,数据包丢失或出现错序的几率较低。同时,试飞实时监控对数据实时性具有很高的要求,因此,选用 UDP 传输协议具有其特定的优势,PCM 数据处理前端服务器与该自主实时监控组态软件建立 UDP 链接,从而将遥测数据“推送”到监控软件画面,确保数据的实时性。图 5 是监控软件数据传输流程图。

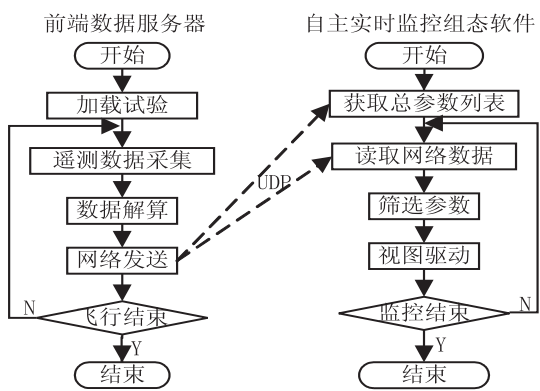


图 5 监控软件数据传输流程

3 应用验证及分析

将软件安装到实际飞行试验中的监控客户端主机上,为试飞工程师及飞行指挥员提供快速构建的监控画面及实时监控平台,并通过配置信息库对其进行管理,如图 6 所示。

在试飞过程中,试飞工程师根据任务需求通过鼠标“拖拽”方式创建仪表、曲线等某一特定监控组件,然后绑定监控参数数据源并对其运算信息进行动态配

置,自主实时监控组态软件自动接入遥测实时监控系
统,便可完成试飞课题所关心机载测试参数的实时监
控,极大简化了实时监控任务准备流程,有效降低了任
务准备压力和复杂度。目前,该软件已成功应用于运
输机、ARJ21 支线客机等型号的多架飞机试飞实时监
控当中,在确保试飞安全和保障试飞任务高质量完成
中发挥着重要作用。

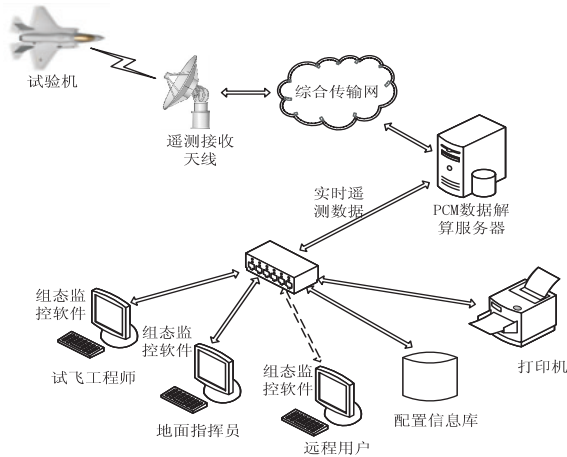


图6 自主实时监控组态软件应用场景图

4 结束语

为解决试飞实时监控软件设计时间紧、修改频繁
的问题,提出了基于组件的试飞自主实时监控平台软
件设计方法,并构建了试飞自主实时监控平台。该自
主实时监控组态软件平台在 Visual Studio 平台下,采
用组态软件设计思想和模块化方法开发完成。通过该
平台,数据处理工程师根据监控参数的表现形式选择
相应的监控组件进行参数配置和运算配置,即可完成
监控画面的编辑任务;试飞工程师在监控过程中可以
灵活调整监控参数和画面布局;软件根据配置参数接
收网络数据并根据定义的运算进行相应解算,驱动整
个监控画面图形元素的运行,为指挥人员和地面试飞

工程师提供及时、准确的飞机状态信息。与传统的基
于 LabVIEW 和 C++ Builder 监控画面相比,能够有效
提高画面准备效率,提高画面编辑的灵活性。

参考文献:

[1] 王亚民,陈青,刘畅生,等. 组态软件设计与开发[M]. 西安:西安电子科技大学出版社,2003.

[2] 赵民正. 面向监控的组态软件—技术分析和设计[D]. 杭州:浙江大学,2002.

[3] 余志文,申辉军. 基于 ActiveX 的 WebGIS 实现技术[J]. 测绘通报,2003(2):53-56.

[4] 查卫翔,谭南林. ActiveX 控件在基于 B/S 结构的远程监控中的应用[J]. 北方交通大学学报,2002,26(1):58-62.

[5] 谢东亮,侯朝桢,杨国胜,等. 无线局域网中异步数据传输模式分析及其应用[J]. 计算机工程与应用,2002,38(11):149-151.

[6] 韩立,尹爱军. 基于组态技术的测控软件开发及测试[J]. 中国测试,2010,36(4):66-69.

[7] Baldwin D R. System and method using double-buffer preview mode;US,US5329630[P]. 1994.

[8] 宋政斌,张国旺. 基于 IRIG106 固态记录器记录标准的 PCM 数据处理技术研究[J]. 计算机测量与控制,2014,22(2):531-533.

[9] 卫卫国,蔡伟,郝志浪,等. 航电 ARINC429 总线监测系统的设计与实现[J]. 电子设计工程,2014,22(3):37-41.

[10] 郑致力. 算术表达式解析引擎的设计及实现[D]. 北京:北京邮电大学,2012.

[11] 邓绪斌. 基于最优树联配的正则表达式学习算法[J]. 复旦学报:自然科学版,2011,50(6):797-802.

[12] 莫正波,宋玲,吕强,等. XML 文档语义检索方法研究[J]. 计算机工程与应用,2013,49(11):121-125.

[13] 董欣,陈晓鸥. XML 文件的显示与浏览[J]. 计算机应用,2000,20(8):29-32.

[14] 王茹,宋瀚涛. XML 文档结构定义规范—XML Schema[J]. 计算机应用研究,2002,19(1):127-129.

(上接第 144 页)

[7] 马锐. 神经网络原理[M]. 北京:机械工业出版社,2010:1-13.

[8] 胡桃红. 心率变异性[J]. 临床心电学杂志,1995,4(1):23-28.

[9] Akselrod S, Gordon D, Ubel F A, et al. Power spectrum analysis of heart rate fluctuation: a quantitative probe of beat-to-beat cardiovascular control[J]. Science, 1981, 213(4504):220-222.

[10] Rohmert W, Laurig W, Philipp U, et al. Heart rate variability and work-load measurement[J]. Ergonomics, 1973, 16(1):33-44.

[11] Kleiger R J, Bigger J P, Bigger J T, et al. Decreased heart rate

variability and its association with increased mortality after a-cute myocardial infarction[J]. American Journal of Cardiology, 1987, 59(4):256-262.

[12] 闻新,李新,张兴旺,等. 应用 MATLAB 实现神经网络[M]. 北京:国防工业出版社,2015.

[13] 史峰,王小川,郁磊,等. MATLAB 神经网络 30 个案例分析[M]. 北京:北京航空航天大学出版社,2014:1-10.

[14] 柳小桐. BP 神经网络输入层数据归一化研究[J]. 机械工程与自动化,2010(3):122-123.

[15] Gupta R, Joshi P, Mohan V, et al. Epidemiology and causation of coronary heart disease and stroke in India[J]. Heart, 2008, 94(1):16-26.