

# 基于 C# 和 SQL Server 的信号接收机自动校准系统软件设计与实现

余磊, 聂纯, 马晖, 王毅, 刘连照, 徐宙

(中国洛阳电子信息装备试验中心, 河南 洛阳 471003)

**摘要:** 为保证信号接收数据的准确性, 提高测试校准工作的自动化水平, 文章基于 C# 与 SQL Server 对信号接收机自动校准系统软件进行了设计; 基于多线程设计系统程序, 提高了测试校准效率; 基于双缓存技术绘制测试数据波形, 解决了波形显示闪屏的问题; 采用 SQL Server 数据库存储测试数据和仪器控制指令, 实现高效数据存储和管理; 运用测试报告自动生成技术实现校准证书和原始记录的自动生成; 该自动校准系统设计对提高信号接收机校准工作效率, 增强设备计量保障能力具有较强的应用价值。

**关键词:** 信号接收机; 校准; 多线程; 双缓存

## Software Design and Implementation of Automatic Calibration System for Signal Receiver Based on C# and SQL Server

Yu Lei, Nie Chun, Ma Hui, Wang Yi, Liu Lianzhao, Xu Zhou

(Luoyang Electronic Equipment Test Center, Luoyang 471003, China)

**Abstract:** In order to ensure the accuracy of receiving data and improve the automation level of testing and calibration, this paper designs the software of automatic calibration system for signal receiver based on C# and SQL Server. Design system program based on multi-threading to improve the efficiency of test calibration. Drawing test data waveform based on double buffer technology solves the problem of waveform display flash screen. SQL Server database is used to store test data and instrument control instructions to achieve efficient data storage and management. The automatic generation of calibration certificate and original record is realized by using the automatic generation technology of test report. The design of the automatic calibration system has a strong application value for improving the calibration efficiency of signal receiver and enhancing the measurement support ability of equipment.

**Keywords:** signal receiver; calibration; multi-threading; double buffering

## 0 引言

信号接收机随着使用时间的增加, 由机械损伤以及电子元器件老化等因素引起的测试数据不准确现象经常发生。为保证信号接收机接收信号的准确度和可靠性, 需要对其开展测试校准工作。常规的手动校准不仅耗时长、效率低, 而且校准工作自动化水平低, 难以满足大工作量的测试校准工作需求。

进行以机架和叠加式的测试仪器操作存在体积大、运输繁重以及操作复杂等缺点, 开展测试校准工作时, 便携程度低, 不便于多场地现场校准。目前, 随着板卡式测试仪器应用技术越来越成熟, 以板卡仪器构建测试平台式的自动测试系统研究也越来越多。板卡仪器中, PXI 板卡仪器具有高性能、低价位和模块化的特点, 广泛应用于试验、测量与数据采集等各种系统中, 其良好的功能特性能够满足便携式校准系统的构建需求<sup>[1-3]</sup>。

本文以 PXI 仪器平台为硬件基础, 设计了一种基于 C

# 与 SQL Server 的信号接收机自动校准系统 (以下简称“自动校准系统”)。对自动校准系统软件进行了详细设计, 采用多线程技术设计系统程序, 使用双缓存技术绘制测试数据波形, 将测试数据和仪器控制指令存储在 SQL Server 数据库中, 最后采用报表自动生成技术生成了测试报告。

## 1 系统构成与功能分析

### 1.1 系统构成

自动校准系统由 PXI 校准平台和自动校准软件构成, 硬件结构如图 1 所示。系统控制由 PXI 机箱中的嵌入式控制器板卡实现, 测试仪器由矢量信号源、音频分析仪、信号分析仪、程控衰减器、程控开关、示波器等 PXI 板卡组成, 选用的仪器模块技术指标均能满足信号接收机测试校准需求。

为了有效检测信号接收机的各项性能指标, 需要控制测试仪器对信号进行测量并对测量结果进行实时显示, 以保证测试数据的准确性。为此设计了自动校准系统软件, 系统软件在 VS2010 平台上采用 C# 编程语言进行设计, 采用 Windows 界面设计风格, 具有良好的人机交互效果。

### 1.2 系统功能

自动校准系统通过校准软件控制校准平台中的仪器模

收稿日期:2019-06-16; 修回日期:2019-10-28。

作者简介:余磊(1992-),男,江西抚州人,硕士,主要从事自动测试系统及时间频率计量方向的研究。

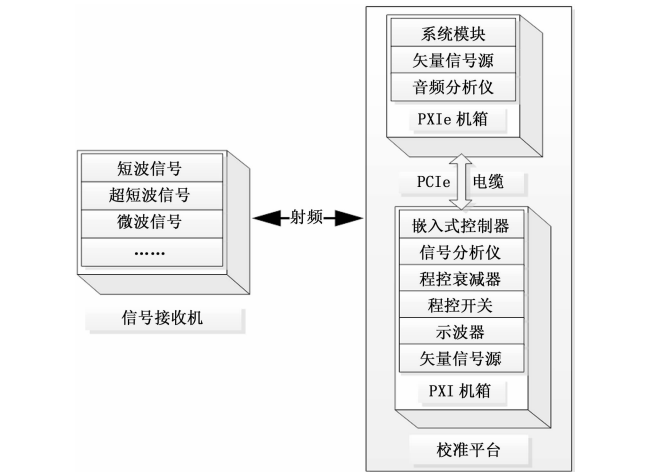


图 1 自动校准系统硬件结构图

块开展测试校准工作，测试校准过程必须满足国家检定规程的要求，具备信号接收机自动校准、测试数据显示与存储以及校准报告自动生成等基本功能。为完善自动校准系统功能，提高人机交互效率和校准系统自动化水平，增加了系统自检和报警、校准任务规划、仪器模块配置以及校准参数配置等功能，使得校准系统软件操作更加规范化。

2 自动校准系统软件总体设计

2.1 系统软件总体架构

自动校准系统软件通过测试电缆、总线等实现测量设备控制、数据分析与处理、故障分析、测量数据管理、数据动态交换和用户登陆等功能。它由用户接口层、测试管理核心层、外围服务层、测试驱动层以及仪器设备层组成，总体架构如图 2 所示。

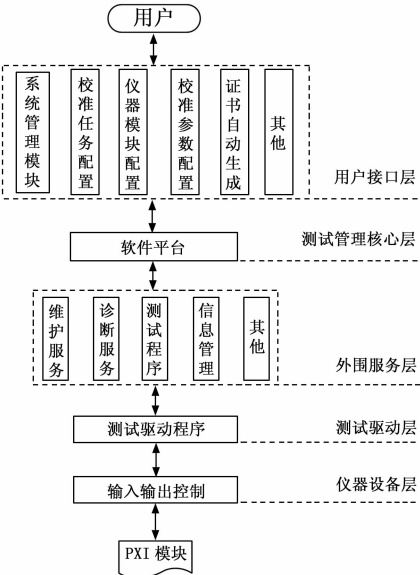


图 2 自动校准软件总体架构示意图

其中用户接口层作为人机界面完成与用户间的交互，它包含校准系统各个功能模块的操作显示；测试管理核心层提供通用测试平台软件的核心构件；外围服务层主要完

成具体应用的功能服务；测试驱动层作为测试程序与仪器设备等相关资源进行通信的纽带；仪器设备层直接面向各类测试资源。

2.2 系统软件工作流程

按照国家检定规程设计校准系统软件工作流程，具体工作流程如图 3 所示。

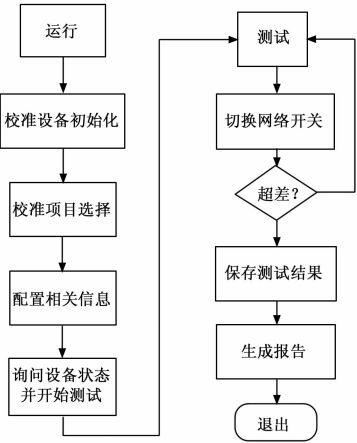


图 3 自动校准系统软件工作流程图

具体流程如下：连接好信号接收机和 PXI 校准平台，开机预热并启动校准软件，操作仪器自检模块实现测试仪器模块的初始化；选择被测信号接收机需要进行的校准项目并配置勤务信息（包括校准时间、人员、生产厂家等信息）；确认信号接收机工作状态后点击开始测试按钮启动测试校准程序，测试程序控制网络开关的切换，实现测试仪器模块的连接；执行测试校准程序后，判断测试结果是否超差并生成原始记录和校准证书，完成整个测试校准任务。

3 自动校准系统软件实现关键技术

3.1 基于多线程的系统程序设计

自动校准系统软件设计时，考虑到测试校准程序运行、测试数据波形实时显示以及数据文件自动生成等进程需要同时进行，采用多线程技术对系统程序进行设计。设计校准系统界面时，在窗体中使用 PictureBox 控件显示测试数据波形。由于 C# 设计窗体程序时，操作 UI 控件（界面显示控件）程序默认为系统主线程，如果将测试校准程序和测试数据波形绘制程序同时写入系统主程序，根据程序运行的时间顺序，会出现波形绘制不连续，不能实现实时显示的问题。如果将测试校准程序写入主程序进行测试操作，测试数据波形绘制程序作为子线程，由于波形绘制时涉及到 PictureBox 控件的操作，与 UI 线程产生冲突，会导致线程阻塞，产生系统假死现象。

为解决测试数据波形实时显示以及 UI 控件程序控制相冲突的问题，设计多线程系统程序时，将波形绘制程序写入主线程中，实现波形显示和操作界面控制，并将测试校准程序作为次线程运行。在进行项目校准时，点击仪器配置按钮，系统成功识别仪器模块并确认链路连通后，提示“仪器模块配置成功”，启动 Timer 定时器（刷新时间设置

为 50 ms) 不间断获取仪器模块测试数据, 实现测试数据的实时采集, 并将测试数据传送至波形绘制程序中, 实现了数据波形实时刷新。点击校准开始按钮启动测试校准程序, 系统给主、次线程随机分配 CPU 时间片, 运行测试校准程序时, 改变仪器模块参数设置以获得不同的测试数据, 在运行主程序时对测试数据进行显示, 可以保证测试数据的准确性。

多线程测试程序设计实现如下:

```
# region 单击仪器模块配置控件, 配置成功后进行波形显示
private void btn_BZ_Set_Click(object sender, EventArgs e)
{
    .....
    ucPictureBox.DrawBackGround();
    lbx_state.Items.Add(DateTime.Now + " 标准设备配置完成!!!");
    MessageBox.Show("请按照连接图要求连接设备!!!");
    timer2.Start();
    ucPictureBox.TimerDraw.Start();
    .....
}
# endregion

# region 启动定时器 timer2, 将测试数据传入波形绘制程序中
private void timer2_Tick(object sender, EventArgs e)
{
    List<string> data = GenerateTestData();
    ucPictureBox.Data = data;
}
# endregion

# region 单击校准开始控件, 启动校准程序次线程
private void btn_Test_Click(object sender, EventArgs e)
{
    Thread t = new Thread(new ThreadStart(BeginTast));
    t.Start();
    tsb_TestBegin.Enabled = false;
    btn_Test.Enabled = false;
}
# endregion

# region 开始校准测试, 遍历校准项目并开始相应校准项目测试
public void BeginTast()
{
    DataRowView drv = lbx_canjian.SelectedItem as DataRowView;
    DataRow dr = drv.Row;
    itemname = dr[lbx_canjian.ValueMember].ToString().Trim();
    .....
    switch (itemname)
```

```
{
    case "01 工作频率":
        WorkingFrequency();
        break;
    .....
}
}
# endregion
```

### 3.2 基于双缓存的测试数据波形绘制程序设计

自动校准软件设计了数据波形显示模块, 用于显示测试仪器模块采集到的测试数据波形。设计测试数据波形绘制程序时, 使用 Graphics.DrawLine() 和 Graphics.DrawString() 等方法将测试数据绘制在 PictureBox 上, 并将绘制程序包装在函数中, 通过函数调用的形式实现波形绘制。采用这种波形绘制方法在进行波形数据刷新时, 需要对已有数据波形进行清除, 重新绘制背景色后, 然后再绘制新采集的测试数据。这种方法在绘制波形时发现了一些问题: 一是重复的绘制与清除波形数据, 导致波形刷新缓慢, 出现闪屏现象; 二是波形数据量大的情况下, 占用系统内存大; 三是在一定的屏幕刷新速率下, 波形绘制与清除占用时间长, 容易产生波形不完整现象。

测试波形绘制程序设计时, 采用 Timer 定时器以 50 ms 的时间间隔获取测试数据, 也就意味着在 50 ms 时间内, 必须完成 PictureBox 上已有波形的清除与重新绘制, 否则将会出现波形不完整和闪屏的现象。原因在于波形的清除与重新绘制时间过长, 导致 50 ms 内波形未绘制完成又开始获取新的波形数据并继续对旧波形进行清除, 始终无法完成一个完整的波形绘制, 导致波形不完整。另外由于绘制波形时, 背景色和面板原始颜色存在明显色差, 在清除完背景色和旧波形后, 由于重新绘制背景色和波形需要一定的时间, 导致 PictureBox 从背景色到面板原始色再到背景色存在一定的时间差, 从人眼直观的感觉就是产生了画图闪屏的现象。双缓存技术可以有效解决画图时波形不完整和闪屏问题, 它避开了直接在 PictureBox 控件上绘制波形, 采用虚拟画布 (内存中的一块区域) 的方式实现, 所有的绘制操作均在虚拟画布上完成, 绘制完成后将画布内容粘贴到 PictureBox 控件上<sup>[4-5]</sup>。基于双缓存绘图方法在单次绘图时只涉及一次 PictureBox 控件的操作, 进行波形刷新时, 也只需刷新一次 PictureBox 控件, 不需要对背景色和旧波形进行清除处理, 大大减少了波形绘制时间, 提高了波形绘制效率, 也解决了画图波形不完整和闪屏问题。

基于双缓存的测试数据波形绘制程序设计如下:

```
private System.Windows.Forms.PictureBox picShow; // 定义 PictureBox
.....
# region 波形绘制
public void DrawWave()
{
```

```
Bitmap tmp =(Bitmap) screen.Clone(); //构建虚拟画布,重画时背景无需重新绘制
Graphics ppy_graph = Graphics.FromImage(tmp);//
.....
for (int i = 0; i < n - 1; i++)//画出一组功率值
{
    ppy_graph.DrawLine(myPen3, x[i], y[i], x[i + 1], y[i + 1]);
}
.....
for (int i = 0; i <= 10; i++)
{
    ppy_graph.DrawString(reF - 10 * i + "dBm", new Font("Times New Roman", 8), new SolidBrush(Color. White), 25, 5 + 36 * i);
}

picShow.Image = tmp;//将虚拟画布粘贴到 PictureBox 上,实现波形绘制
}
```

**3.3 基于数据库的测试数据和程控指令存储设计**

测试数据是指使用自动校准系统开展信号接收机测试校准工作过程中涉及到的数据统称,它包括被校接收机基本信息、校准系统信息、勤务数据、信号通道误差数据、校准数据以及证书数据等。在测试数据存储设计时,采用 SQL Server 数据库实现数据存储,基于数据库的存储技术具备完善的数据索引和修改机制,能够实现数据高效的存储,它采用严格的身份验证和加密控制方式,可以有效保证数据安全,另外数据库存储技术能够实现完整的事务和数据一致性控制,确保数据有效,维护数据完整性<sup>[6-8]</sup>。基于数据库存储技术的自动校准系统数据库数据架构如图 4 所示。

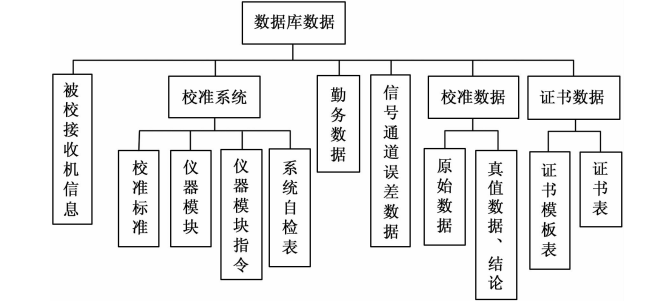


图 4 自动校准系统数据库数据架构

程控指令库是测量仪器的指令代码、测量参数和检定结果评判方法的集合。指令库的存储方式有两种:一种是基于本地计算机的文件调用技术,如 XML、INI 等文件形式存储;一种是基于数据库访问技术,包括 Access、SQL Server、Oracle 等数据库存储方式。利用数据库存储技术的优点,检定系统采用 SQL Server 数据库存储指令库,实现仪器程控指令和仪器控制过程分离,即编制自动测试程序时,将控制过程编码置于校准系统程序中,而用于控制仪

器的程控指令代码、测量参数和校准结果评判方法存储在程序之外。

校准系统对指令库的访问贯穿于整个校准过程的始终。指令库访问流程图 5 所示。

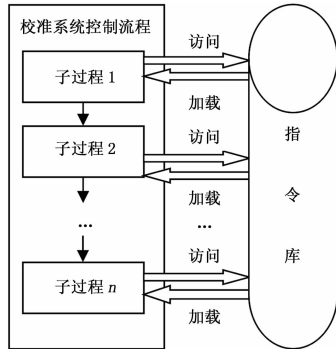


图 5 指令库访问流程图

**3.4 测试报告自动生成程序设计**

自动校准系统中,测试报告包括检定证书和原始记录报告,依据检定规程要求,生成测试报告是校准工作的最后一步。进行测试报告生成时,采用自动生成方法,点击报告生成按钮自动将 SQL Server 数据库中测试数据自动加载到报告模板中,实现报告生成。

报告自动生成程序实现主要分为报告模板设计和报告程序生成设计两部分。进行报告模板设计时,使用 Word 文档绘制报告表格模板,在需要更新报告基本数据(包括校准时间、校准人员、温湿度等信息)的表格中使用文本域控件 TextField 进行数据表示,在需要插入校准项目测试数据的位置设置域,便于将数据库中查询到的数据插入报告中。

进行报告自动生成程序设计时,采用功能强大的 Spire.Doc 组件实现。Spire.Doc 组件是一个专业的 Word.NET 库,能够在 C# 平台上实现 Word 文档的创建、读取、编辑、转换和打印等操作,它支持绝大多数 Word 元素,包括页面、小节、页眉、页脚、脚注、列表、文本、字段、书签、图片以及背景等,而且它在创建多页面 Word 文档效果非常好,可以说是生成多页面测试报告的完美选择。使用 Spire.Doc 组件进行程序实现时,先使用 Document() 方法创建生成测试报告模板文档,遍历文档中 TextField 文本域控件并赋值(值从数据库中查询勤务信息获得),查询域获取插入测试数据的位置,使用 Table.Addtable() 方法在该位置插入表格,查询数据库校准项目测试数据并依次写入表格中,测试数据写入完成后,使用 Document.SaveToFile() 方法将测试报告保存到指定位置。

**4 实验结果与分析**

在搭建信号接收机自动校准硬件平台的基础上,综合运用校准软件对某型信号接收机开展测试校准工作,以检验自动校准系统的性能,测试效果如图 6 所示。

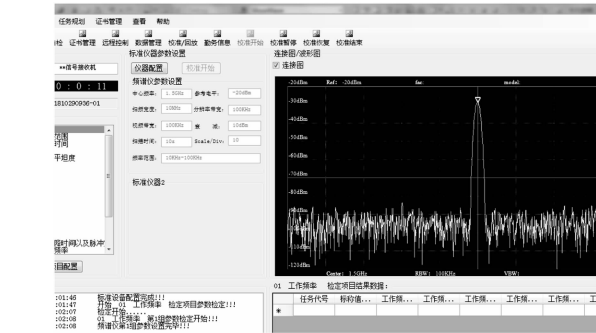


图 6 信号接收机自动校准效果图

性能检验采用对比实验的方式进行，同一技术人员分别采用自动校准系统自动操作和台式仪器叠加手动操作的形式开展测试校准工作。由于测试数据的准确性取决于测试方法、仪器性能指标和操作人员技术水平决定，在进行对比实验时，操作人员严格按照校准规程进行仪器操作，同时校准规程规定的测试方法也被写入校准程序中，降低了测试方法和操作人员技术水平对测试数据准确度的影响。并且 PXI 仪器模块和台式仪器性能指标均满足信号接收机的测试校准需求，对比实验时不考虑将测试数据准确度纳入对比范围，主要考查两者在时间成本上的差异。

采用自动校准系统软件开展测试校准工作，主要包括校准项目配置、勤务信息配置、仪器参数配置、校准以及报告生成共 5 个步骤，而采用台式仪器叠加的手动测试校准工作主要包括测试校准、数据收集、数据评定和报表生成四个步骤，两者对比实验效果如表 1 所示。

表 1 对比实验效果表

项目		手动校准	自动校准
校准准备	校准项目选择	无	人工 人工
	勤务信息录入		
	仪器参数配置		
项目校准	20 项校准项目	人工	自动
数据处理	数据收集	人工	自动
	数据评定	人工	自动
报表生成	原始记录/证书 耗时	人工 约 200min	自动 约 30min

可以看出两者在时间成本上存在很大差异，采用自动校准方法耗时主要集中在校准工作准备阶段，包括手动录入

勤务信息和仪器参数配置信息，但针对同一型号信号接收机，一次仪器参数配置可以供后续测试校准使用，加上人工录入信息的时间，校准信号接收机全部项目总共消耗大约 30 分钟时间。而手动校准方法在校准项目测试和报表生成上耗时较多，同一校准项目涉及多个测试频点，技术人员在重复操作仪器参数设置花费时间较大，同时人工记录数据并根据数据生成原始记录和校准证书也需要花费一定的时间，计算下来大约花费 200 分钟，基本上校准一台信号接收机要花费一上午时间。相比之下，采用自动校准的方法可以节省 85% 的时间成本，大大提高了测试校准效率，降低了由人工操作带来的失误，同时采用基于 PXI 的硬件结构也大大增强了校准系统的便携性。

5 结束语

针对信号接收机手动校准存在耗时长、效率低等问题，采用 PXI 仪器模块构建了自动校准系统硬件平台，并使用 C# 编程语言对自动校准系统软件进行设计。软件设计时，采用多线程技术设计测试校准程序，提高了测试校准效率；基于双缓存技术绘制测试数据波形，解决了波形显示闪屏的问题；采用 SQL Server 数据库存储测试数据和仪器控制指令，实现高效数据存储和管理；运用测试报告自动生成技术实现校准证书和原始记录的自动生成。该自动校准系统设计对提高信号接收机校准工作自动化水平，增强设备计量保障能力具有较强的应用价值。

参考文献：

[1] 孙吉, 张丕状. 基于 PXI 总线的信号调理卡的设计 [J]. 山西电子技术, 2016 (1): 43-45.

[2] 栾宝宽, 田华明, 王传刚. 基于 PXI 的模块电路检测诊断系统软件设计 [J]. 微机与应用, 2015 (10): 8-10.

[3] 夏彦超. 基于 PXI 总线的检测系统研制 [D]. 哈尔滨: 哈尔滨工业大学, 2010.

[4] 刘婷婷, 潘彤, 陈彬茹. 利用双缓存技术提高绘图速率 [J]. 电脑编程技巧与维护, 2012 (18): 22-23.

[5] 张仁忠, 常明志, 许德新. 利用 MFC 实现双缓存机制改善图形的显示效果 [J]. 应用科技, 2005 (32): 47-49.

[6] 龙守湛. 数据库技术及应用 [M]. 北京: 科学出版社, 1995.

[7] 王海霞, 姚新宇, 韩守鹏. 分布实时仿真系统数据库设计与性能优化 [J]. 计算机仿真, 2007 (7): 142-145.

[8] 孙海荣. 网络数据库安全性研究 [J]. 电脑知识与技术, 2008 (3): 1115-1116.

[12] 彭国金. 基于类 PCM 结构过程的试飞多科目 iNET 处理技术 [J]. 计算机测量与控制, 2015, 23; 205 (10): 306-308.

[13] 风河公司著, 王金刚, 宫霄霖. VxWorks 网络程序员指南 [M]. 北京: 清华大学出版社, 2003.

[14] 张晓华, 李智涛, 徐钊. VxWorks 网络协议栈的 MUX 接口 [J]. 单片机与嵌入式系统应用, 2002 (5): 23-25.

[15] 彭国金, 王建军, 刘嫚婷. 基于预留缓存的飞行试验 iNET 数据处理技术 [J]. 现代电子技术, 2016, 39 (12): 48-50.

(上接第 177 页)

[9] 房瑾, 霍朝晖, 聂睿. 机载网络数据实时处理软件的设计与实现 [J]. 电子设计工程, 2015 (12): 60-62.

[10] 寇云林, 陈怀民, 段晓军. VxWorks END 网络驱动软件的开发与实现 [J]. 计算机测量与控制, 2009 (1): 224-226+239.

[11] 宗思瑶. 机载实时数据处理系统设计与实现 [D]. 西安: 西安电子科技大学, 2013.