制造型企业小型产品质量管理系统

——统计过程控制管理模块的设计与实现

摘 要

通过绪论、需求分析、概要设计、详细设计与实现、测试和总结等各章对统计过程管理控制管理模块进行了详细的分析与设计。针对各个部分的需要，绘制了相关的图示和表格，用以达到最优的项目设计文档说明效果。

本系统是针对制造型企业对产品质量检验的管理分析而设计。主要包括生产过程检验管理和统计过程控制管理两大模块。同时该系统还可以作为生产制造执行系统中的一个有机组成部分，协助生产制造执行系统建立集成的生产现场控制与品质管理平台和完善的生产过程数据库。

产品质量管理工作对提升[企业竞争力](http://baike.baidu.com/view/540265.htm" \t "_blank)有重要意义。过程检验可以防止出现大批不合格品，避免不合格品流入下道工序去继续进行加工。实施统计过程控制则可以帮助企业在质量控制上真正作到"事前"预防和控制。

本系统采用的C#进行编码，同时采用微软公司的Visual studio 2008和微软公司的Microsoft SQL Server 2008 开发环境进行开发。系统的结构组成主要有客户端、web服务和数据库访问层以及数据库持久化层。是系统层次明确，逻辑清晰，便于编码和后期维护。

**关键词**：统计过程 控制 质量 C#

Small Product Quality Management System For Manufacturing Enterprise

——Design and Implementation of Statistical Process Control Management Module

**ABSTRACT**

This paper through the introduction, demand analysis, outline design, detailed design and implementation, testing and summary of the statistical process management module for the statistical process of the detailed analysis and design. For each part of the need to draw the relevant diagrams and tables, in order to achieve the best results of the project design documentation.

This system is designed for the management of manufacturing enterprises to the product quality inspection. Mainly including the production process inspection management and statistical process control and management of the two modules. At the same time the system can also as a production manufacturing execution is an integral part of the system, assist the manufacturing execution system to establish the integration of production control and quality management platform and improve the production process database.

Product quality management is of great significance to enhance the competitiveness of enterprises. Process inspection can prevent the emergence of a large number of substandard products, to avoid the flow of substandard goods into the next process to continue processing. The implementation of statistical process control can help enterprises in the quality control of truly "advance" prevention and control.

This system uses the C# to carry on the code, simultaneously uses the Microsoft Corp studio Visual 2008 and the Microsoft Corp SQL Server Microsoft 2008 development environment to carry on the development. The structure of the system consists of client, web service, database access layer and database persistence layer. System level is clear, clear logic, easy to encode and post maintenance.

**Key words:** Statistical Process Control Quality C#

目 录

[第一章 绪论 1](#_Toc16644)

[1.1 项目背景与现状 1](#_Toc13737)

[1.1.1 项目背景简介 1](#_Toc11469)

[1.1.2 当前国内外现状 1](#_Toc20185)

[1.2 选题意义与系统主要工作 2](#_Toc32311)

[1.2.1 选题目的及意义 2](#_Toc27595)

[1.2.2 本系统主要工作 3](#_Toc15227)

[1.3 统计过程控制原理（SPC） 3](#_Toc8408)

[1.3.1 控制图简要介绍 3](#_Toc23145)

[1.3.2 控制图的应用方法 4](#_Toc31723)

[1.4 开发工具简介 5](#_Toc19254)

[1.4.1 Microsoft Visual Studio 2008 5](#_Toc9872)

[1.4.2 Microsoft SQL Server 2008 5](#_Toc19135)

[1.5 本章小结 5](#_Toc17836)

[第二章 需求分析 6](#_Toc16361)

[2.1 背景简介 6](#_Toc17671)

[2.2 可行性分析 6](#_Toc23476)

[2.2.1 社会可行性 6](#_Toc17775)

[2.2.2 经济可行性 6](#_Toc428)

[2.2.3 技术可行性 7](#_Toc6300)

[2.3 功能需求分析 7](#_Toc2306)

[2.3.1 功能划分 7](#_Toc3994)

[2.3.2 系统使用角色分析 8](#_Toc20002)

[2.5 本章小结 9](#_Toc20106)

[第三章 概要设计 10](#_Toc16514)

[3.1 系统结构设计 10](#_Toc23419)

[3.1.1 系统整体结构 10](#_Toc6304)

[3.1.2 质量管理系统结构 11](#_Toc21747)

[3.1.3 电芯检验过程数据检查系统结构 11](#_Toc10975)

[3.2 系统实体设计 12](#_Toc5597)

[3.2.1 系统实体 12](#_Toc9805)

[3.2.2 系统整体实体联系 14](#_Toc31154)

[3.3 数据库表结构设计 15](#_Toc18073)

[3.3.1 数据库表结构设计 15](#_Toc19219)

[3.3.2 数据库表间关系 19](#_Toc31429)

[3.4 数据流需求分析 20](#_Toc30781)

[3.4.1 数据流 20](#_Toc20055)

[3.4.2 数据流分层分析 20](#_Toc17396)

[3.4.3 数据字典 21](#_Toc4369)

[3.5 本章小结 24](#_Toc7888)

[第四章 详细设计及实现 25](#_Toc22910)

[4.1 背景 25](#_Toc29731)

[4.2 设计及实现 25](#_Toc8406)

[4.2.1 控制图数据查询导出至EXCEL设计与实现 25](#_Toc27394)

[4.2.2 预控图数据查询与导出到EXCEL设计与实现 30](#_Toc7572)

[4.2.3 控制项目添加设计与实现 33](#_Toc17356)

[4.2.4 控制项目与BOM号绑定设计与实现 37](#_Toc23629)

[4.2.5 控制图数据录入的设计与实现 40](#_Toc25213)

[4.2.6 预控图数据录入的设计与实现 42](#_Toc19787)

[4.3 本章小结 44](#_Toc3954)

[第五章 结论 45](#_Toc4403)

[5.1项目归纳与总结 45](#_Toc894)

[5.1.1 项目归纳 45](#_Toc5479)

[5.1.2 项目结论 45](#_Toc30465)

[5.2 新发展趋势 45](#_Toc22306)

[5.3 课题尚待改进的建议 46](#_Toc1827)

[5.3.1 课题尚存在的问题 46](#_Toc10336)

[5.3.2 课题改进的建议 46](#_Toc2462)

[5.4 本章小结 46](#_Toc19701)

[参考文献](#_Toc27626)

[致 谢](#_Toc14733)

# 绪论

## 1.1 项目背景与现状

### 1.1.1 项目背景简介

本题目来源于天津市电子计算机研究所项目一部，是从电池行业生产制造执行系统中提炼出来的一个小型系统，项目专门针对制造型企业对产品质量检验的管理分析而设计。主要包括生产过程检验管理和统计过程控制管理两大模块。同时该系统还可以作为生产制造执行系统中的一个有机组成部分，协助生产制造执行系统建立集成的生产现场控制与品质管理平台和完善的生产过程数据库。

做好产品质量管理工作对提升[企业竞争力](http://baike.baidu.com/view/540265.htm" \t "_blank)有重要意义。过程检验可以防止出现大批不合格品，避免不合格品流入下道工序去继续进行加工。实施统计过程控制则可以帮助企业在质量控制上真正作到"事前"预防和控制[1]。

### 1.1.2 当前国内外现状

自从80年代（上个世纪）开始，统计过程控制软件就开始萌芽，并且在许多具有大量工业的发达国家开始复兴，当今世界，为满足生产制造的需要，许许多多的大公司在现有技术的支持先，也逐渐的积极推广并且应用统计过程控制软件在自己的公司中以及工厂生产制造中[2]。

统计过程控制软件的作用很广泛，它开始于质量监控的前提下的产品制造，但经过半个多世纪以及多年来的实践和发展，特别是和计算机技术的相关技术融合后，统计过程的原理以及使用的方法，都早已大面积使用到了产品设计和统购包销以及服务支持加上管理等过程。

在我国实际运用统计过程控制软件进行质量管理的公司并不多，大多数企业为大型外资企业（尤其是欧州资本主义国家 、美利坚合众国、加拿大、墨西哥以及亚洲的日本的外资企业）及其供应。部分中国国内的大中型企业（如力神电池股份有限公司）及民企等也都使用的是统计过程控制软件来进行质量管理与检测，我国国内主要行业为汽车及相关零部件制造、电子、电器、机械加工等。而绝大多数的中小型企业和小型企业对统计过程控制软件的关注比较短缺[3]。

大量统计数据也显示了这一问题。 我国的长三角地区，有些数据通过对106家制造型企业进行统计过程控制软件应用及其有效性的调研或调查，从调查中产生的数据发现，在我国长三角地区，使用统计过程控制软件的企业有40.5%左右。而认为实施统计过程控制软件比较有用的的生产制造企业中仅仅11.5%左右。从以上数据来看，我国国内的大部分企业中，生产制造比较广泛的企业应用SPC并不是特别的理想[4]。

现今，在国外统计过程控制软件的应用是相当普遍的。世界500强公司都无一例外的通过统计过程控制软件来管理来控制其产品的质量品质。例如美国通用汽车公司、标准石油公司和美国钢铁公司（前身是成立于1864年的卡内基钢铁公司）等等制造型企业都在使用统计过程控制软件软件进行生成制造管理。

由此对比，我国的统计过程控制软件的普及远远比不上国外。在当前信息时代的背景下，让产业智能化将成为我国生产制造的迫切解决的问题。

## 1.2 选题意义与系统主要工作

### 1.2.1 选题目的及意义

统计过程控制，简称SPC。它是生产制造企业为了调高生产效率和生产质量，将统计技术应用到过程中，并且对过程中的各个阶段的相关指标进行有效的评估、监控，同时建立保持机制，对生产制造过程的相关指标使之处于可接受的状态，达到稳定的水准，以此来保证产品和服务的相关数据合乎规定需要的一种质量控制手段[5]。

过程控制检查是过程控制的一部分[6]。从内容上来看，主要包含了两个方面的内容：首先是通过对控制图录入、导出到报表中，进而分析过程中这个状态的稳定性，达到对过程的存在的异常因素进行预定的警报；其次是计算过程的能力指数的分析的稳定的过程能力，以此合乎技术的要求的程度，对过程质量进行评估与判定。

产品制造与加工的过程中，企业的产品的尺寸、重量、密度等规格，会由于许许多多的原因，使之发生一定的理性或非理性的波动，这种波动的幅度与频率或对产品的质量产生比较大的影响，但是，生产制造企业完全可以通过采取有效的措施，将这些措施应用到生产制造过程中，进而来避免或者除去这种波动所造成的影响，这种措施就是过程控制。它是一种预防性方法；强调全员参与；强调整个过程，主要的内容就是在于其过程性[7]。

其应用范围有：加工过程、设计过程、管理过程、服务过程等。主要作用是：

（1）保证生产制造过程的稳定的持续和可预测。

（2）提高企业的生产的产品的质量和生产能力以及降低企业生产制造过程中的成本。

（3）为制造过程的分析提供有效的合理的确切的依据。

（4）区分变差的特殊原因和普通原因，作为采取局部措施或对系统采取措施的指南。

正确的SPC软件都可帮助企业实现以下4点目标[8]：

（1）科学并且有效的提高企业的生产力效率。

（2）极大的降低产品的报废率，提高产品的工艺精度。

（3）及时发现生产流程中的问题的隐患。

（4）有效压缩生产的成本。

（5）在制造现场随时做出准确的判断和有效的决策。

（6）对生产流程中任何变化迅速地做出有效的且及时的反应。

### 1.2.2 本系统主要工作

做好产品质量管理工作对提升[企业竞争力](http://baike.baidu.com/view/540265.htm" \t "_blank)有重要意义。过程检验可以防止出现大批不合格品，避免不合格品流入下道工序去继续进行加工。实施统计过程控制则可以帮助企业在质量控制上真正作到"事前"预防和控制。

（1）控制项目管理子模块，实现创建控制项目并设检测样本数和频率的功能。

（2）BOM号与控制项目子模块，实现BOM号与控制项目绑定，并设定检验指标功能。

（3）预控图录入子模块，实现预控图检验数据增删改查的功能。

（4）控制图录入，实现控制图检验数据增删改查的功能。

（5）控制图导出，实现控制图检验数据以图表格式输出到excel文件的功能。

## 1.3 统计过程控制原理（SPC）

### 1.3.1 控制图简要介绍

统计过程控制所产生的核心的控制图，它是对过程质量加以测定、记录从而进行控制管理的一种用科学方法设计的重要的依据[9]。如图1.1所示：

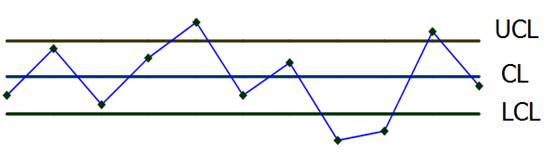


图1.1 控制图

Fig .1.1 Control chart

控制图的主要组成包含了四部分因素：   
 中心线：此条线就是样本值的平均预期的数据。如果样本的波动越是集中到这条线的附近，证明样本的指标的稳定[10]。  
 样本上限：它是样本中预期的所能承受的最大的波动值，如果超过这条线，就说明，有因素影响了样本的稳定性，进而对生产制造过程进行合理的分析与调整。   
 样本下限：它同样本上限类似，只不过是样本中的预期的最低值不能超过样本下限，如果超过，就说明生产制作过程中某个环节出现了问题。需要及时的调整，来避免产品的质量的不合格。   
 样本波动曲线：它是将所要检查的项目的样本值，连线成一条折线图，以此来分析生产制造过程的稳定性。

### 1.3.2 控制图的应用方法

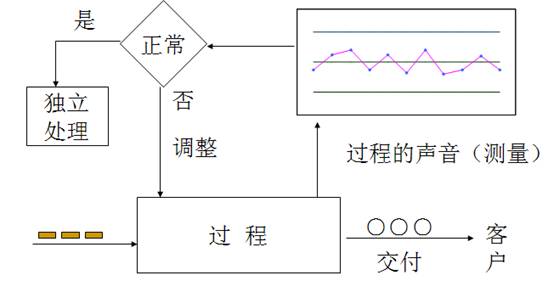


图1.2 控制原理图

Fig .1.2 Control principle diagram

如图1.2所示，控制图是生产过程中通过采样进而产生的折线图。通过控制图来判定生产制造是不是正常，如果正常，则独立处理，如果不正常，则需要调整生产制造的过程的某些因素，然后重新生产控制图，来分析生产制造过程中的稳定性。最后将正常的产品交付给客户[11]。

1.3.3 统计过程控制的控制图建立步骤

统计过程控制的建立可以分为六个步骤：

（1）择取质量特征数据   
 （2）明确管制图形式   
 （3）规划样本的区间,以及抽样频率和抽样的形式   
 （4）收录具体的样本值    
 （5）计算控制中各部分的参数，如上下限等   
 （6）持续收集数据,利用管制图监视制程

## 1.4 开发工具简介

### 1.4.1 Microsoft Visual Studio 2008

本系统在程序设计时采用的开发工具是Visual studio 2008。VS是一个基本完整的开发工具集，它包括了整个生命周期中所需要的大部分工具。VS具有代码提示功能，有利于提高程序开发的效率。VS包含.NET Framework的框架，支持C#语言程序的开发。C#也是一门建立在C++基础上的高级语言，是用来编写.NET框架的语言[12]。

### 1.4.2 Microsoft SQL Server 2008

本系统所采用的数据库是SQL Server 2008。现阶段关系型数据库管理软件基本承担了大多数数据持久化的管理任务。同时，SQL Server 就是一个比较优秀的关系数据库管理系统[13]。SQL Server 2008的扩展它的安全性很高，主要是由于以下几点内容：

1. 存储数据的密文化

微软公司的Microsoft SQL Server 2008的安全系数在同类数据库管理软件中是比较高的。首先它可以对整个数据库的框架、数据文件的形式以及日志文件进行加密处理，使用者没有必要改动相应的应用程序。对数据进行加密使使用者基本上可以满足遵守的规范和及其关注数据隐私的要求。 对数据加密的优势包括使用任何的范围或模糊的查询或者搜索加密的数据以及加强数据安全性从而防止没有被分配权限的非法访问者的访问和数据的加密。这些可以在不改变已有的应用程序的情况下进行[14]。

2. 对主外健的合理控制

Microsoft SQL Server 2008为加密和密钥提供了一个全面的解决方案。为了满足不断发展的对数据中心的信息的更强安全性的需求，公司投资给供应商来管理公司内的安全密钥。Microsoft SQL Server 2008通过支持第三方密钥管理和硬件安全模块（HSM）产品为这个需求提供了很好的支持[15]。

## 1.5 本章小结

本章对本课题的选题意义和国内外发展情况进行了调查，明确了项目的背景信息，然后对本系统所涉及的核心的SPC技术进行说明。SPC是生产制造过程中控制质量的重要手段。本章节也对该项目的开发环境进行了简要的设定，项目主要使用的Visual studio 2008和微软公司的Microsoft SQL Server 2008。

# 需求分析

## 2.1 背景简介

本题目来源于天津市电子计算机研究所项目一部，是从电池行业生产制造执行系统中提炼出来的一个小型系统。统计过程控制是为了生产制造过程对生产制造中的产品质量进行检测，以此来合理的调整生产制造。

## 2.2 可行性分析

### 2.2.1 社会可行性

项目专门针对制造型企业对产品质量检验的管理分析而设计。主要包括生产过程检验管理和统计过程控制管理两大模块。同时该系统还可以作为生产制造执行系统中的一个有机组成部分，协助生产制造执行系统建立集成的生产现场控制与品质管理平台和完善的生产过程数据库。

做好产品质量管理工作对提升[企业竞争力](http://baike.baidu.com/view/540265.htm" \t "_blank)有重要意义。过程检验可以防止出现大批不合格品，避免不合格品流入下道工序去继续进行加工。实施统计过程控制则可以帮助企业在质量控制上真正作到"事前"预防和控制。

法律因素方面。开发中所有商业软件都采用购买的方式使用正版软件。所有源代码为自主开发，自有知识产权。涉及到第三方控件和支付接口，也通过正规渠道进行商务购买。开发中使用的开源技术不存在法律纠纷。在用户使用可行性方面，用户对于电子采购交易平台的理解和接受需要一定时间周期。具体操作和维护人员需要进行专项培训。

由于本选题是源于天津市电子计算机研究所项目一部的力神迈尔斯方型电池控制系统，本系统已经应用于天津力神动力电池制造公司的生产制造过程中。所以，本系统没有违背法律和道德，所以在社会上是可行的。

### 2.2.2 经济可行性

本项目专门针对制造型企业对产品质量检验的管理分析而设计。主要包括生产过程检验管理和统计过程控制管理两大模块。同时该系统还可以作为生产制造执行系统中的一个有机组成部分，协助生产制造执行系统建立集成的生产现场控制与品质管理平台和完善的生产过程数据库。做好产品质量管理工作对提升[企业竞争力](http://baike.baidu.com/view/540265.htm" \t "_blank)有重要意义。过程检验可以防止出现大批不合格品，避免不合格品流入下道工序去继续进行加工。实施统计过程控制则可以帮助企业在质量控制上真正作到"事前"预防和控制。

支出方面，搭建该系统需要组建开发团队，需要架构师，项目经理，分析员，测试员，程序员若干位，预计需要6个月，需要支出组建团队的费用，硬件设备支出方面，需要内网数据库服务器2台， 存储系统，备份系统，应用服务器2台，Web服务器2台，网络设备，还有安全设备，软件支出方面，需要服务器操作系统（应用内网和外网服务器操作系统Unix、Linux 、Windows2003），SQL server数据库，备份软件，虚拟化软件，Web 中间件，杀毒软件等等，其他支出费用还有材料差旅费用，调研费用，还有最后的系统运行维护费用等等。

### 2.2.3 技术可行性

首先在资金、软硬件设备和人员都到位的情况下，功能目标能够如期达到。按对项目影响程序由大小到依次排序为：资金、人员配置、软硬件设备[3]。其次使用ASP技术，现有的功能目标完成可以达到。市场上已经较多成功案例和模式可以参考。初期要求开发人员较少，以了解需求和设计为主。要求有多年项目经验开发人员和项目经理参与，项目经理对于项目的成败起重要作用。随着设计逐步完成，需要的开发人员较多，且应有一至两年的使用ASP.NET开发网站的经验为宜[3]。在开发架构方面，项目团队的人员熟练掌握以B/S为基础的三层架构的软件项目开发，在访问数据库方面，项目团队人员熟练使用ADO.NET还有LINQ等数据库访问技术，所以技术上是完全可行的。在安全保密性方面，采用SSL，SET等国内外安全技术，所以这个也是完全可以的。

## 2.3 功能需求分析

### 2.3.1 功能划分

1. 质量管理子系统管理

(1)（电芯）控制项目管理：在此功能模块中，管理员可以为电芯控制项目新增控制项目，同时也可以进行相应的数据维护（修改、删除）。在添加新项目时，需要指定项目类型，类型包含X-Bar图预控图。同时也需要指定工序和样本容量和频率，用于后期的统计。

(2)（电芯）BOM号与控制项目管理：在此功能模块中，主要是进行BOM号与控制项目的绑定。在进行添加时，需要指定统计图类型和工序，选定控制项目和BOM号绑定，同时需要设定控制图中心线（CL）、统计图波动上限（UCL）和波动下限(LCL)以及工程规范上限和下限，设定预控图的中心线（CL）。

(3)（电芯）预控图数据录入：此功能模块中，主要是维护电芯的预控图数据。包括对预控图的数据的添加修改和删除操作。

(4)（电芯）控制图数据录入：此功能模块中，主要是维护电芯的控制图数据。包括对控制图的数据的添加修改和删除操作。

2.电芯过程检验数据查询

(1)（电芯）X-Bar图：此功能模块中，主要是查询电芯特定控制项目的的X-Bar图的相关数据，同时也可以产生电芯特定控制项目的X-Bar图并导出该图至excel文件中。

(2)（电芯）预控图：此功能模块中，主要是查询电芯特定控制项目的预控图的相关数据，同时也可以产生电芯特定控制项目的预控图并导出该图至excel文件中。

### 2.3.2 系统使用角色分析

1.管理员

(1)质量管理。

A.（电芯）控制项目管理。包含增加、修改和删除。

B.（电芯）BOM号与控制项目管理。包含增加、修改和删除。

C.（电芯）预控图数据录入。包含增加、修改和删除。

D.（电芯）控制图数据录入。包含增加、修改和删除。

(2)电芯过程检验数据查询。

A. 查看电芯X-Bar图，同时可以导出X-Bar图至excel文件中。

B. 查看电芯预控图，同时可以导出预控图至excel文件中。

如图2.1所示，该图刻画的是管理员的用例图。

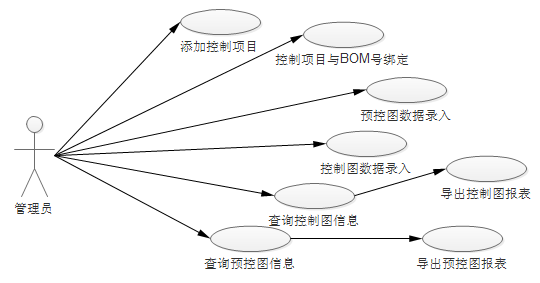


图2.1 管理员的用例图

Fig .2.1 Administrator's rights management use case diagram

## 2.5 本章小结

本章对统计过程管理进行了需求分析。在功能需求分析中。首先对该项目的可行性进行了分析，包括社会可行性、经济可行性和技术可行性。在明确该项目的可行性后，该部分对系统的功能模块进行了分析，明确了本系统的功能用例，绘制了系统的功能用例图。

# 概要设计

## 3.1 系统结构设计

### 3.1.1 系统整体结构

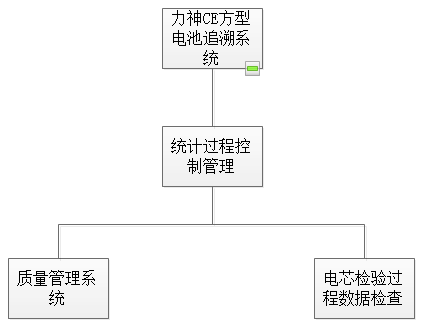


图3.1 SPC系统总体结构图

Fig .3.1 Overall structure of SPC system

如图3.1所示：

力神CE方型电池追溯系统是制造型小型企业生产制造质量管理系统。主要用于生产制造过程中质量管理。

统计过程控制管理属于力神CE方型电池追溯系统的一个二级子系统。它包含了两个三级子系统：质量管理系统和电芯检验过程数据检查。

三级子系统中的质量管理系统主要是用于质量管理的基本信息录入。三级子系统中的电芯检验过程数据检查主要是用于质量管理的信息查询导出。

### 3.1.2 质量管理系统结构

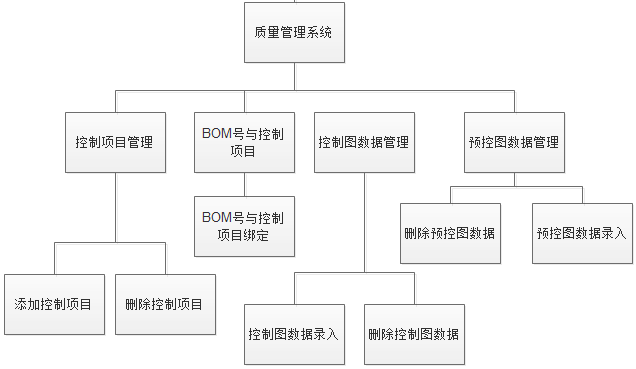


图3.2 SPC质量管理子系统总体结构图

Fig .3.2 Overall structure of SPC quality management subsystem

如图3.2所示：

质量管理系统是统计过程控制管理的子系统。

质量管理系统的二级子系统包含了控制项目管理、BOM号与控制项目、控制图数据管理和预控图数据管理。

二级控制项目管理子系统中包含了添加控制项目和删除控制项目两个功能模块，分别用于控制项目的添加和删除。二级BOM号与控制项目中包含了BOM号与控制项目绑定一个功能模块，用于BOM号与控制项目的绑定。二级控制图数据管理中包含了控制图数据录入和删除控制图数据功能模块。二级预控图数据管理中包含了删除预控图数据和预控图数据录入两个功能模块。

### 3.1.3 电芯检验过程数据检查系统结构

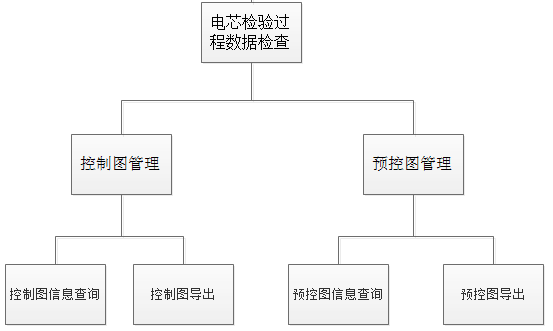


图3.3 SPC电芯检验过程数据检查子系统结构图

Fig .3.3 Structure diagram of data check subsystem of SPC core testing process

如图3.3所示：

电芯检验过程数据检查是统计过程控制管理的子系统。

电芯检验过程数据检查的二级子系统包含了控制图管理和预控图管理。

二级控制图管理中包含了控制图信息查询和控制图导出两个功能模块，分别用于控制图的信息查询与导出。二级预控图管理中包含了预控图信息查询和预控图导出，分别用于预控图的信息查询和导出。

## 3.2 系统实体设计

### 3.2.1 系统实体

1.预控图实体

它是用来预控图数据录入与导出的实体模型。如图3.4所示：

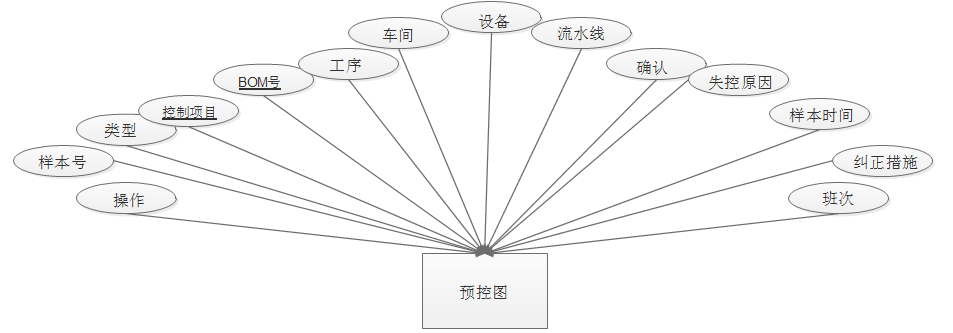


图3.4 预控图实体数据模型

Fig. 3.4 Mapping the entity data model

2.物料清单（BOM）实体

它是统计过程管理控制质量检测的的内容实体模型。如图3.5所示：

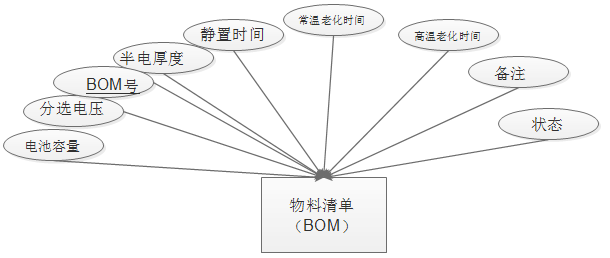


图3.5 物料实体数据模型

Fig .3.5 Material entity data model

3.控制制图实体

它是控制图数据录入与导出控制图的实体模型。如图3.6所示：

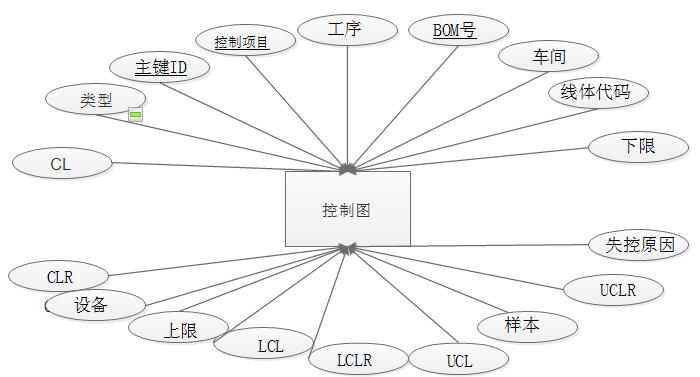


图3.6 控制图实体数据模型

Fig .3.6 Entity data model of control chart of material entity data model

4.控制项目实体

它是统计过程管理控制所要检测的项目的实体模型。如图3.7所示：

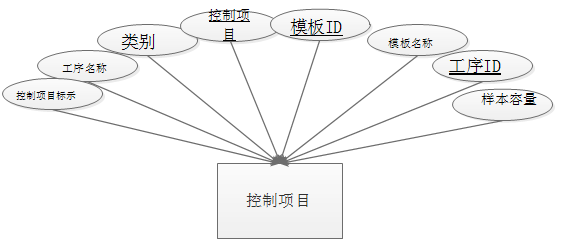


图3.7 控制项目实体数据模型

Fig. 3.7 Control project entity data model

5.批次表实体

它是统计过程管理控制质量检测的批次的实体模型。如图3.8所示：

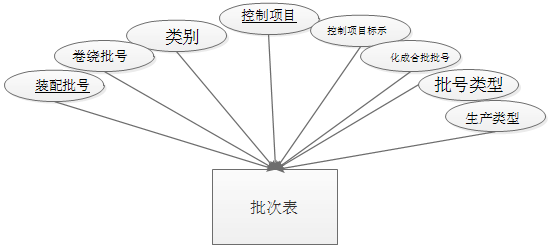


图3.8 批次实体数据模型

Fig. 3.8 Batch entity data model

### 3.2.2 系统整体实体联系图

如图3.9所示：

控制项目与预控图是绑定关系，一个控制项目和对应多个预控图实体模型。

物料清单（BOM）与控制项目是绑定关系，一个控制项目对应一个BOM号实体。

控制图实体和控制项目是绑定关系，一个控制项目可对应多个控制图。

控制项目与批次是绑定管理，一个控制项目对应多个批次。

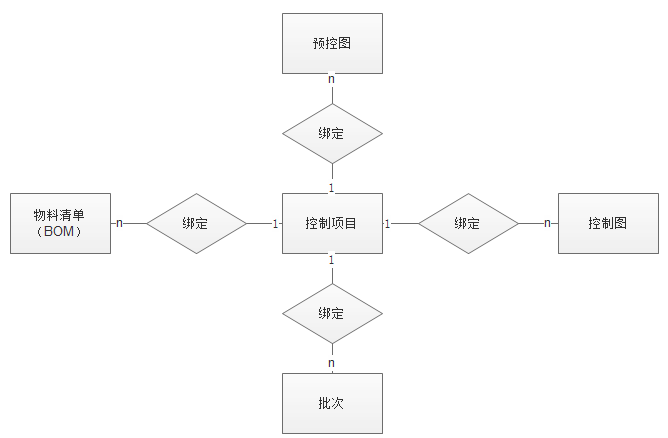


图3.9 全局实体数据模型

Fig .3.9 Global entity data model

## 3.3 数据库表结构设计

### 3.3.1 数据库表结构设计

1.BOM号表

用于存放物料清单相关信息。如表3.1所示：

表3.1 BOM号表（AK\_BOMNO）

Table 3.1 BOM number table (AK\_BOMNO)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否为NULL | 含义 |
| PACKBOMNO | [varchar](255) | NOT NULL | BOM号 |
| BATTERYCAPACITY | [numeric](9, 3) | NOT NULL | 电池容量 |
| VOLTAGE | [numeric](9, 3) | NOT NULL | 分选电压 |
| HALFETHICKNESS | [numeric](9, 3) | NOT NULL | 半电厚度 |
| STAYTIMELEN | [numeric](5, 1) | NOT NULL | 静置时间: 单位小时 |
| AGINGTIMELEN | [numeric](5, 1) | NOT NULL | 常温老化时间：小时 |
| HIGHAGINGTIMELEN | [numeric](5, 1) | NOT NULL | 高温老化时间：小时 |
| REMARKS | [nvarchar](100) | NOT NULL | 备注 |
| USESTATE | [varchar](10) | NOT NULL | 状态 |
| CREATER | [varchar](20) | NOT NULL | 创建人 |
| CREATETIME | [datetime] | NOT NULL | 创建时间 |
| UPDATER | [varchar](20) | NOT NULL | 更新人 |
| UPDATETIME | [datetime] | NOT NULL | 更细时间 |

2.控制项目与BOM号绑定表

用于存放控制项目与BOM号绑定数据。如表3.2所示：

表3.2 控制项目与BOM号绑定表（T\_BOM\_CONTROL\_PROJECT）

Table 3.2 Control project and BOM number binding table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否为NULL | 含义 |
| BOMNO | [varchar](255) | NOT NULL | BOM号 |
| CONSEQ | [numeric](18, 0) | NOT NULL | 控制项目标示 |
| CONTROL\_TYPE | [varchar](20) | NOT NULL | 类别 |
| PREDICT\_LIMIT | [numeric](9, 4) | NOT NULL | 预控限 |
| NORM\_LIMIT | [numeric](9, 4) | NOT NULL | 规格限 |
| PREDICT\_LIMIT\_2 | [numeric](9, 4) | NOT NULL | 预控限 |
| NORM\_LIMIT\_2 | [numeric](9, 4) | NOT NULL | 规格限 |
| UCL1 | [numeric](9, 4) | NOT NULL | UCL1 |
| CL1 | [numeric](9, 4) | NOT NULL | CL1 |
| LCL1 | [numeric](9, 4) | NOT NULL | LCL1 |
| UCL2 | [numeric](9, 4) | NOT NULL | UCL2 |
| CL2 | [numeric](9, 4) | NOT NULL | CL2 |
| LCL2 | [numeric](9, 4) | NOT NULL | LCL2 |
| STA\_UPPER | [numeric](9, 4) | NOT NULL | 工程规范上限 |
| STA\_FLOOR | [numeric](9, 4) | NOT NULL | 工程规范下限 |
| CREATER | [varchar](20) | NOT NULL | 创建人 |
| CREATTIME | [datetime] | NOT NULL | 创建时间 |
| UPDATER | [varchar](20) | NOT NULL | 更新人 |
| UPDATETIME | [datetime] | NOT NULL | 更新时间 |
| INSPECTLOCK | [nvarchar](10) | NOT NULL | 插入锁定 |
| CL | [numeric](9, 4) | NOT NULL | CL |
| CPTYPE | [int] | NOT NULL | CP类型 |

3.控制项目表

用于存放控制项目相关信息。如表3.3所示：

表3.3 控制项目表（T\_CONTROL\_PROJECT）

Table 3.3 Control project table (T\_CONTROL\_PROJECT)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否为NULL | 含义 |
| CONSEQ | [numeric](18, 0) | NOT NULL | 控制项目标示 |
| CONTROL\_TYPE | [varchar](20) | NOT NULL | 类别 |
| CONTROL\_NAME | [nvarchar](50) | NOT NULL | 控制项目 |
| TEMPLATEID | [numeric](18, 0) | NULL | 模板ID |
| TEMPLATENAME | [varchar](50) | NULL | 模板名称 |
| PROCESSESID | [numeric](18, 0) | NOT NULL | 工序ID |
| PROCESSNAME | [varchar](50) | NOT NULL | 工序名称 |
| SAMPLE\_CAPACITY | [nvarchar](20) | NOT NULL | 样本容量 |
| FREQUENCY | [nvarchar](20) | NOT NULL | 频率 |
| STANDARD | [nvarchar](20) | NULL | 工程规范 |
| CREATER | [varchar](20) | NOT NULL | 创建人 |
| CREATTIME | datetime | NOT NULL | 创建时间 |
| UPDATER | [varchar](20)] | NOT NULL | 更新人 |
| UPDATETIME | [datetime] | NOT NULL | 更新时间 |
| PIPELINE | [nvarchar](20) | NULL | 管道 |
| PROCESS\_TYPE | [nvarchar](20) | NULL | 类型 |
| CPTYPE | [int] | NULL | 0-电芯, 1-PACK |

4.BOM号表

用于存放物料清单相关信息。如表3.4所示：

表3.4 批次表（T\_REAL\_FLOW）

Table 3.4 Batch list (T\_REAL\_FLOW)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否为NULL | 含义 |
| INSTANCEID | [numeric](18, 0) | NOT NULL | 控制项目标示 |
| BOMNO | [varchar](255) | NOT NULL | 类别 |
| SIZENO | [varchar](255) | NOT NULL | 控制项目 |
| SIZENO\_TYPE | [int] | NOT NULL | 批号类型 |
| PRODUCE\_TYPE | [varchar](20) | NULL | 生产类型 |
| C\_SIZENO | [varchar](255) | NULL | 卷绕批号 |
| A\_SIZENO | [varchar](255) | NULL | 装配批号 |
| F\_SIZENO | [varchar](255) | NULL | 化成合批批号 |
| OLD\_INSTANCEID | [numeric](18, 0) | NULL | 插入编号 |
| PIPELINE | [nvarchar](20) | NULL | 管道 |
| PROCESSDEFINITIONID | [numeric](18, 0) | NOT NULL | 工序 |
| FLOW\_NAME | [varchar](100) | NOT NULL | 线体 |
| BOMNO\_SIZENO | ([BOMNO]+[SIZENO]) | NOT NULL | BOM号和批次号 |
| NEXT\_NODEID | [numeric](18, 0) | NOT NULL | 下次标注编号 |
| NEXT\_NODENAME | [varchar](50) | NOT NULL | 下次标注名 |
| FRONT\_NODEID | [numeric](18, 0) | NOT NULL | 前置标注编号 |
| FRONT\_NODENAME | [varchar](50) | NOT NULL | 前置标注名称 |
| NOW\_NODEID | [numeric](18, 0) | NOT NULL | 当前标注编号 |
| NOW\_NODENAME | [varchar](50) | NOT NULL | 当前标注名 |
| FIRST\_NODEID | [numeric](18, 0) | NULL | 第一次标注编号 |
| FIRST\_NODENAME | [varchar](50) | NULL | 第一次标注名称 |
| LAST\_NODEID | [numeric](18, 0) | NULL | 最后标注编号 |
| LAST\_NODENAME | [varchar](50) | NULL | 最后标注时间 |
| STATE | [varchar](10) | NULL | 状态 |
| LOCKFLAG | [varchar](5) | NULL | 是否锁定 |
| CREATER | [varchar](20) | NOT NULL | 创建者 |
| CREATTIME | datetime | NOT NULL | 创建时间 |
| UPDATER | [varchar](20) | NULL | 更新人 |
| UPDATETIME | datetime | NOT NULL | 更新时间 |
| ISCLOSED | int | NULL | 是否封闭 |

5.预控图与控制图数据表

用来存储预控图或控制图相关信息。如表3.5所示：

表3.5 预控图与控制图数据表（T\_XR\_DATA）

Table 3.5 Re control chart and the control chart data table (T\_XR\_DATA）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否为NULL | 含义 |
| vseq | [numeric](18, 0) | NOT NULL | 主键ID |
| TYPE | [varchar](50) | NOT NULL | 类型 |
| PROJECT | [varchar](50) | NOT NULL | 控制项目 |
| PROCESS | [varchar](50) | NOT NULL | 工序 |
| BOM | [varchar](50) | NOT NULL | BOM号 |
| LINE | [varchar](50) | NOT NULL | 流水线 |
| WORKSHOP | [varchar](50) | NOT NULL | 车间 |
| LINECODE | [varchar](50) | NOT NULL | 线体代码 |
| DEVICE | [varchar](50) | NULL | 设备 |
| UCL | [numeric](9, 4) |  | UCL |
| CL | [numeric](9, 4) | NULL | CL |
| LCL | [numeric](9, 4) | NULL | LCL |
| UCLR | [numeric](9, 4) | NULL | UCLR |
| CLR | [numeric](9, 4) | NULL | CLR |
| LCLR | [numeric](9, 4) | NULL | LCLR |
| UPPER\_LIMIT | [numeric](9, 4) | NULL | 上限 |
| LOWER\_LIMIT | [numeric](9, 4) | NULL | 下限 |
| TEST\_DATE | datetime | NULL | 测试日期 |
| WORKNAMEID | [nchar](20) | NULL | 工作编号 |
| CLASSNAME | [varchar](50) | NOT NULL | 班次 |
| WORKNAME | [varchar](50) | NOT NULL | 操作人 |
| CONFIRMNAME | [varchar](50) | NOT NULL | 确认人 |
| SIZENO | [varchar](50) | NOT NULL | 批号 |
| SAMPLETIME | [datetime] | NOT NULL | 样本时间 |
| SAMPLEGROUPNUMBER | [varchar](50) | NOT NULL | 样本组号 |
| SAMPLENUMBER | [varchar](50) | NOT NULL | 样本号 |
| SAMPLEVALUE | [numeric](9, 4) | NOT NULL | 样本值 |
| PROGRAM\_TYPE | [varchar](50) | NOT NULL | 类型 |
| NOCOLREASON | [varchar](255) | NULL | 失控原因 |
| RESOLVENT | [varchar](255) | NULL | 纠正措施 |

### 3.3.2 数据库表间关系

控制项目数据表与其他数据表间是主外键的关系，控制项目表是主键表，其他表中有相应的外键与其产生主外键约束关系。同时物料清单表中的BOM号作为主键也与其他表有主外键约束关系。如图3.10所示：

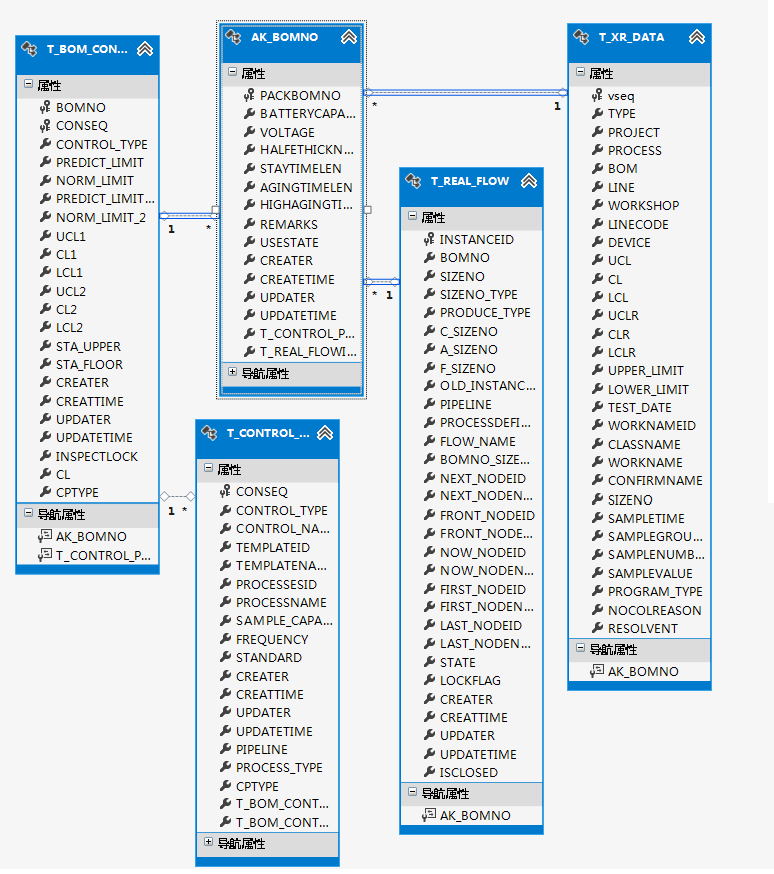


图3.10 数据库表间关系

Fig .3.10 Relationship between database tables

## 3.4 数据流需求分析

### 3.4.1 数据流

数据流图(亦称为DFD图）：设计人员可以利用此工具从数据传递和处理方面，通过图形方式来阐述系统的功能（如逻辑功能）以及数据在系统结构内部的流动方向和逻辑改变过程，是系统分析的主要工具和用于表示模型的一种方法。

### 3.4.2 数据流分层分析

1. 顶层数据流图

在该系统统计过程控制管理系统中，统计过程控制管理模块包括控制项目管理、BOM号与控制项目、预控图录入、控制图录入和控制图、预控图导出五部分。可实现多种控制项管理的动态灵活配置。如图3.11所示：

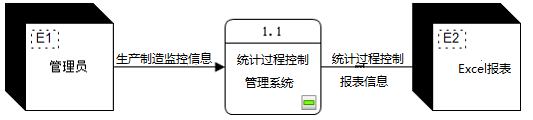


图3.11 统计过程管理顶层数据流图

Fig .3.11 Statistical process management top-level data flow diagram

2. 第二层数据流图

管理员发起管理事务，首先在质量管理子系统中，添加控制项目相关信息，之后再电芯检验过程数据查询子模块中查询并产生统计过程的相关的控制图，并导出至excel文件中。如图3.12所示：

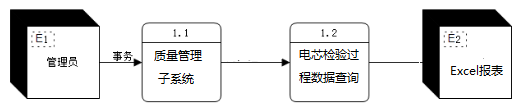


图3.12 统计过程管理第二层数据流图

Fig .3.12 Statistical process management second layer data flow diagram

3. 第三层数据流图

如图3.13所示：

管理员发起管理事务，首先添加控制项目，控制项目就是该质量管理系统所需要进行控制的项目的内容信息，其添加后存在在控制项目表中。

紧接着，绑定BOM号与控制项目。在此过程中，需要BOM号信息和控制项目信息，将其绑定后存入BOM号与控制项目绑定信息表中。

信息绑定完成后，管理员可以对控制图和预控图进行信息数据录入，分别存入控制图信息表和预控图信息表中。

预控图和控制图信息录入完成后，管理员可以在电芯检查过程数据查询，然后到处数据到报表中，分别形成预控图和控制图。

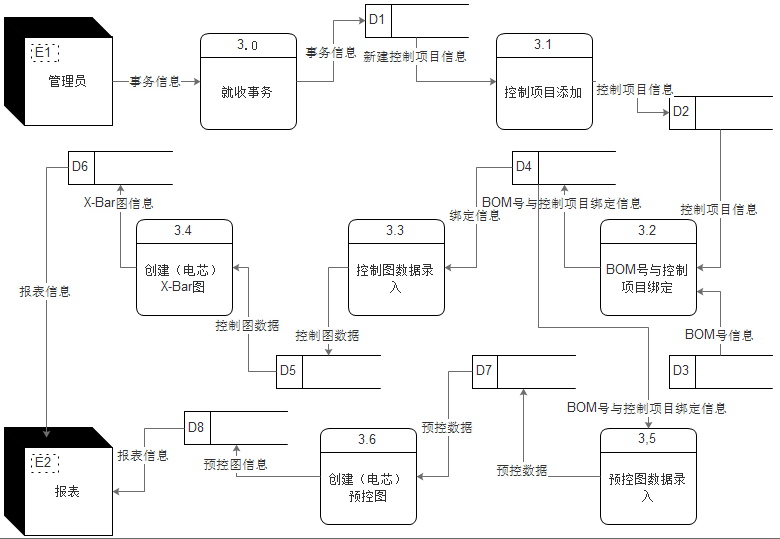


图3.13 统计过程管理第三层数据流图

Fig .3.13 Statistical process management third layer data flow diagram

### 3.4.3 数据字典

数据字典（英文简介DD）是关于本系统相关的数据的信息集合，它的作用是把数据流图中的相关元素进行定义，放到相应的集合中。数据自定通过对软件提供分析并且在设计过程中对有关的数据信息进行描述。本系统中相关的数据字典如表3.6至表3.13所示：

表3.6 事务数据流存储

Table 3.6 Transaction data stream storage

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D1 |
| 名称 | 事务清单 | | |
| 组织形式：按照编号进行组织排序 | | | |
| 关键字：事务名称，事务内容 | | | |
| 记录组成：{事务信息} | | | |
| 备注：记录的是事务信息 | | | |

表3.7 控制项目管理数据流存储

Table 3.7 Control project management data stream storage

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D2 |
| 名称 | 控制项目表 | | |
| 组织形式：按控制项目编号进行组织排序 | | | |
| 关键字：控制项目名称、编号 | | | |
| 记录组成：{控制项目信息} | | | |
| 备注：记录的是控制项目的分组信息 | | | |

表3.8 BOM号数据流存储

Table 3.8 BOM data stream storage

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D3 |
| 名称 | BOM号存储表 | | |
| 组织形式：按照BOM号组织排序 | | | |
| 关键字：BOM号、编号 | | | |
| 记录组成：{BOM号信息} | | | |
| 备注：记录的是BOM号信息 | | | |

表3.9 控制项目与BOM号绑定数据流存储

Table 3.9 Control item and BOM number binding data stream storage

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D4 |
| 名称 | 控制项目与BOM号绑定表 | | |
| 组织形式：按照编号进行组织排序 | | | |
| 关键字：控制项目编号、BOM号编号 | | | |
| 记录组成：{控制项目与BOM号绑定基本信息} | | | |
| 备注： | | | |

表3.10 控制图数据流存储

Table 3.10 Control chart data stream storage

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D5 |
| 名称 | 控制图数据表 | | |
| 组织形式：按照角编号进行组织排序 | | | |
| 关键字：控制图名称、控制图数据 | | | |
| 记录组成：{控制图数据信息} | | | |
| 备注： | | | |

表3.11 X-Bar图数据流存储

Table 3.11 X-Bar graph data stream storage

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D6 |
| 名称 | X-Bar图报表 | | |
| 组织形式：按照编号进行组织排序 | | | |
| 关键字：控制项目、控制图名称 | | | |
| 记录组成： | | | |
| 备注： | | | |

表3.12 预控图数据流存储

Table 3.12 Mapping data stream

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D7 |
| 名称 | 预控图数据表 | | |
| 组织形式：按照编号进行组织排序 | | | |
| 关键字：预控图名称、预控图数据 | | | |
| 记录组成： | | | |
| 备注： | | | |

表3.13 预控图数据流存储

Table 3.13 Mapping data stream

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 统计过程管理 | 编号 | D8 |
| 名称 | 预控图报表 | | |
| 组织形式：按照编号进行组织排序 | | | |
| 关键字：控制项目名称、预控图名称 | | | |
| 记录组成：{预控信息} | | | |
| 备注：记录的是预控图报表信息 | | | |

## 

## 3.5 本章小结

本章对统计过程管理进行了概要设计。本部分分析并绘制了该项目的系统结构图，明确了该系统的整体框架。然后，对该系统中所涉及的实体进行了分析，通过对项目中所涉及的实体进行分析，绘制了各实体的图例，之后绘制了实体之间的联系图（即全局实体联系图）和数据库表结构。而后通过对系统结构和实体联系的分析，抽象出了系统的数据库表关系图。紧接着，本章对系统的数据流向进行了分析，明确了该系统的数据流向后，绘制了数据流图，包括顶层数据流图、二层数据流图和三层数据流图。

# 详细设计及实现

## 4.1 背景

本系统详细设计是将先前的对系统的分析具体的进行物理分析与设计具体的实现，包括界面的设定，程序的编排。在本章，我们将对系统的物理的实际实现进行设计，在此过程中，包含了该系体的权限管理的业务流程图和部分实现代码等。

## 4.2 设计及实现

### 4.2.1 控制图数据查询导出至EXCEL设计与实现

1. 功能简介

（电芯）X-Bar图数据查询与导出至EXCEL文件中。

2. 功能数据基本要求

控制图数据查询与导出模块中。管理员需提供控制项目、工序、流水线、BOM号和车间，以及查询的时间范围。同时在导出的时候，需要提供设备名称。除了设备名称字段需要管理员手动输入外，其他数据都可通过下拉列表进行选定。当管理员查询到控制图数据后，发起导出事务，程序会通过已经查出的数据产生X-Bar图进行导出。

3.程序内部功能实现流程

在程序内部，管理员录入好数据后，点击添加按钮，客户端程序会调用web服务层的创建控制项目的方法，同时将录入好的数据封装到模型中，一并传递到服务层添加函数中，服务层会调用数据库访问层的添加方法，将数据插入到数据库中，如果添加成功，添加函数会返回新添加的控制项目的基本信息，经由服务层传递到客户端进行结果显示。当点击导出按钮后，程序调用导出程序，将前台的数据进行封装传递至导出程序中，导出程序加载X-Bar模板构建控制图，然后导出至excel文件中。

4.功能模块时序图构建

在时序图中，需要五个对象，分别是

操作者。承担的是事务的发起，和数据的录入以及程序的开启任务。

客户端。承担的事务的接收和程序的入口

Web服务。对客户端提供相应的操作服务。

DAO。主要给服务层提供数据库操作入口，在接受数据库操作事务后，对数据库进行增加、删除、修改和查询等操作。

DATABASE。用于数据持久化，对数据库访问层提供支持。

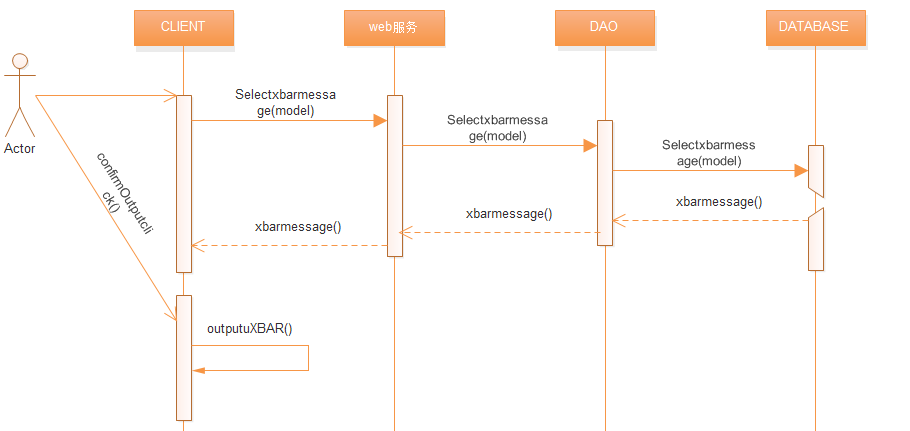


图4.1 X-Bar 图查询与导出时序图

Fig .4.1 X-Bar figure query and export timing diagram

如图4.1所示：在时序图中，管理员是事务的发起者，当管理员发起查询事务的时候，需要从客户端（即CLIENT）提供的用户接口发起事务操作信息。

当客户端接收到管理员员发起的查询事务信息后，客户端会创建WEB服务对象，而WEB服务对象提供了查询事务的查询服务接口，用于对客户端提供服务支持。当查询事务过来后，服务层的服务程序会调用数据库访问层的数据库操作对象，数据库操作对象会对数据库进行查询SQL语句的创建。

在创建数据库查询SQL语句完成后，数据库操作对象会以事务的形式提交查询SQL语句到数据库中进行查询。在数据以数据集的形式返回查询数据后，数据库操作对象会将查询出来的数据集进行解析，并封装到相应的对象模型中，最后将解析后的数据信息传递至WEB服务对象中，最终交付给客户端程序。客户端调用显示程序进行显示。

查询数据完成后，当管理员发起导出控制图事务后，客户端不再创建和调用WEB服务对象了，这是因为，导出程序已经嵌入到了客户端程序中。客户端程序只需调用导出程序，在导出程序中解析前台界面的信息后封装到导出模型中，直接将数据导出至EXCEL文件中去。

这样的设计，即省去了程序的运行时间，同时，客户端可以对导出的信息进行自定义调整。

5. 功能模块界面实现

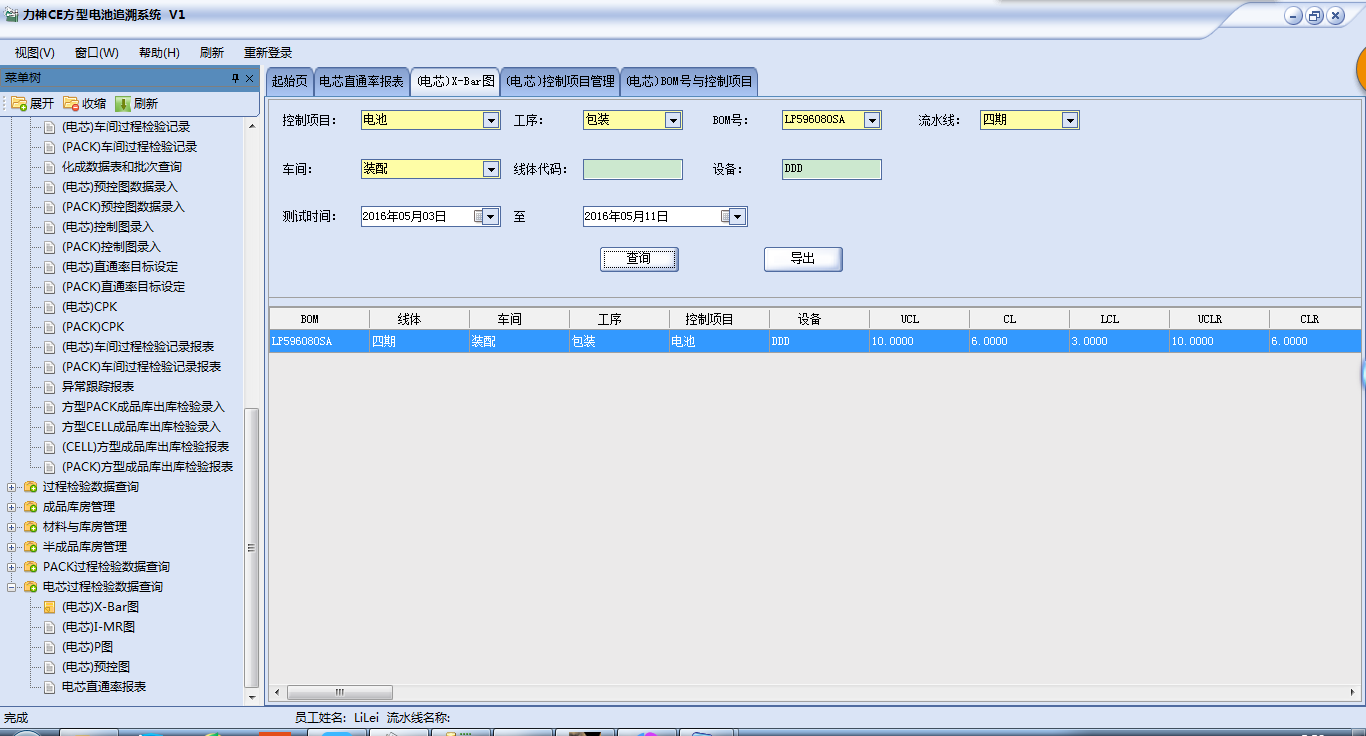


图4.2 X-Bar图数据查询与导出界面的实现结果图

Fig. 4.2 X-Bar figure data query and export interface design

如图4.2所示：界面使用的WINFORM来设计与实现。界面中包含了控制项目下拉列表，工序列表，BOM号列表，流水线列表，车间列表，线体代码文本框，设备文本框，车间下拉列表和时间区间，以及查询按钮，导出按钮和数据展示表格。

之所以这样设计是因为，控制项目等以下拉列表框实现的内容均是已经设置好的数据，用户不能自己手动输入。如果用户手动输入这些信息，会导致在查询控制图的时候，控制图的数据项不能被有效的过滤出来，还有可能会导致程序由于录入非法信息而出现未知的异常。

线体代码和设备两项设计成手动输入内容，这是因为，这两个项目在控制图数据过滤的时候会被自动忽略，不影响控制图信息的过滤。

而时间区间设计出时间格式的下拉框被用户选择，这是因为在后台过滤信息的时候，数据库中的这两个字段是时间的格式，如果出现非法的格式（即用户如果以文本的形式手动输入），则会导致在过滤信息的时候，数据库中出现日期比较异常，从而导致程序在数据库中查询信息的时候出现终止，有可能这些非法信息会导致数据库程序奔溃，或查询出的信息不符合要求。

将查询按钮和导出按钮并排设计在窗口中，这是因为，管理员在操作的过程中，在查询出信息后，可以选择导出或不导出。如果没有导出按钮，则程序在实现导出操作时，出现没有入口的情况。

6. 代码实现与说明

控制图的代码实现过程中，应运到了模型数据封装与参数数组化已经数据集以及文件操作的相关技术。

DataGridView是C#程序语言提供的数据展示控件，此控制有一个比较特别的性质是，它既可以将数据展示到前台中去，也可以将前台的数据进行添加或修改后传递至后台程序中去，即担任了“消费者”，也担任了生产者的角色。所以该模块中使用过的是该控件来控制数据的形式和内容。

paramsstring[] filename：它是可变长度的数组。由于程序在导出过程中，我们将传递参数信息设置成可变，即传递至后台的参数可以自定义内容，既可以单值传递，也可以多值传递。这样保证了程序的可复用性。大大降低了程序的代码的冗余性。

DataSet : 数据集的使用方便了程序中数据的整体传输操作。它规范了数据的形式，避免了程序在使用过程中出现数据信息的格式不正确性。其中最大的优势在于，它的内部可以包含多个表信息，方便了数据的多元化操作。同时它其中的DataRow属性可以让我们通过遍历的形式，以索引的形式来获取或存放数据信息。

SaveFileDialog：它是C#语言提供的文件操作程序集，它的内部是从C，C++这些稍低级的语言中的文件操作优化而来的。它的使用可以即保证数据信息安全的导出至目标位置，也可以是操作简单化，不像其他语言中对文件操作的繁琐和逻辑性较为复杂。

publicstaticvoid ComOutCSV(ParentForm form, DataGridView dgv, paramsstring[] filename)

{

//赋工作表(sheet)名，只方法适导出一个工作表的情况

//默认导出文件名为工作表名加日期时间

DataSet ds = newDataSet();

DataTable dtlm = newDataTable(dtname);

//对list全部记录进行遍历

for (int row = 0; row < dgv.Rows.Count; row++)

{

DataRow drlm = dtlm.NewRow();

int i = 0;

for (int col = 0; col < dgv.Columns.Count; col++)

{

if (!(dgv.Columns[col].Visible == false || dgv.Columns[col].HeaderText == "选择" || dgv.Columns[col].HeaderText == "操作"))

{

string value = dgv.Rows[row].Cells[col].FormattedValue.ToString().Trim().Replace(",", "-");

if (isNumeric(value))

{

drlm[i] = "　" + value;

}

else

{

drlm[i] = value;

}

if (dgv.Columns[col].HeaderText == "批次号"

|| dgv.Columns[col].HeaderText == "电池分档日期"

|| dgv.Columns[col].HeaderText == "LOT"

|| dgv.Columns[col].HeaderText == "托盘序号")

{

drlm[i] = "\t" + dgv.Rows[row].Cells[col].FormattedValue.ToString().Trim().Replace(",", "-");

}

else

{

drlm[i] = dgv.Rows[row].Cells[col].FormattedValue.ToString().Trim().Replace(",", "-");

}

i++;

}

}

dtlm.Rows.Add(drlm);

}

ds = SplitTable(dtlm, 60000);

//赋工作表(sheet)名，只方法适导出一个工作表的情况

//默认导出文件名为工作表名加日期时间

string dtname = form.formName;

if (filename.Length > 0)

{

dtname = filename[0];

}

DataSet ds = newDataSet();

DataTable dtlm = newDataTable(dtname);

SaveFileDialog saveDlg = newSaveFileDialog();

saveDlg.Filter = "CSV文件(\*.csv)|\*.csv";

saveDlg.FileName = ds.Tables[0].TableName + DateTime.Now.ToString("yyyyMMddHHMMss");

if (saveDlg.ShowDialog() == DialogResult.OK)

{

int size = 1024;

int sizeCnt = (int)Math.Ceiling((double)ds.Tables[0].Rows.Count / (double)2000);

if (sizeCnt == 0)

sizeCnt = 1;

StreamWriter sw = newStreamWriter(saveDlg.FileName, false, Encoding.Default, size \* sizeCnt);

foreach (DataColumn column in ds.Tables[0].Columns)

{

string name = column.ColumnName.Substring(column.ColumnName.IndexOf("\_") + 1);

sw.Write(name + ",");

}

sw.WriteLine();

foreach (DataRow row in ds.Tables[0].Rows)

{

foreach (DataColumn col in ds.Tables[0].Columns)

{

sw.Write(row[col.ColumnName].ToString() + ",");

} sw.WriteLine();

}

sw.Flush();

sw.Close();

Functions.ShowMessageBox("导出完成!", Const.EVENT\_INFO);

}

}

7. 导出X-Bar图至Excel文件的实现。

如图4.3所示：

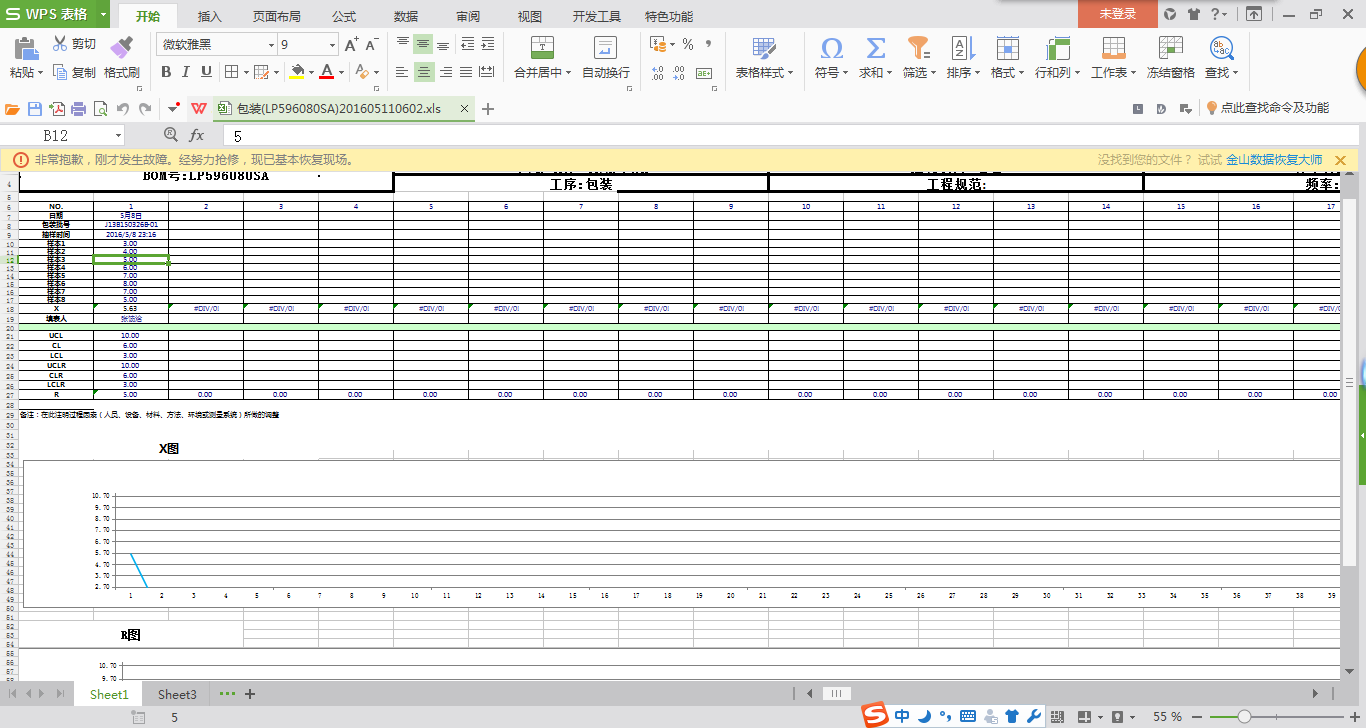


图4.3 X-Bar图excel模板的实现结果图

Fig .4.3 X-Bar figure excel template design

### 4.2.2 预控图数据查询与导出到EXCEL设计与实现

1.功能简介

（电芯）控制项目管理：在此功能模块中，管理员可以为电芯控制项目新增控制项目，同时也可以进行相应的数据维护（修改、删除）。在添加新项目时，需要指定项目类型，类型包含两种，分别是X-Bar图和预控图。同时也需要指定工序和样本容量和频率，用于后期的统计。

2.功能数据基本要求

预控图数据查询与导出模块中。管理员需提供控制项目、工序、流水线、BOM号和车间，以及查询的时间范围。同时在导出的时候，需要提供设备名称。除了设备名称字段需要管理员手动输入外，其他数据都可通过下拉列表进行选定。当管理员查询到控制图数据后，发起导出事务，程序会通过已经查出的数据产生预控图进行导出。

3.程序内部功能实现流程

在程序内部，管理员录入好数据后，点击添加按钮，客户端程序会调用web服务层的创建控制项目的方法，同时将录入好的数据封装到模型中，一并传递到服务层添加函数中，服务层会调用数据库访问层的添加方法，将数据插入到数据库中，如果添加成功，添加函数会返回新添加的控制项目的基本信息，经由服务层传递到客户端进行结果显示。当点击导出按钮后，程序调用导出程序，将前台的数据进行封装传递至导出程序中，导出程序加载预控图模板构建预控图，然后导出至excel文件中。如图4.4所示：

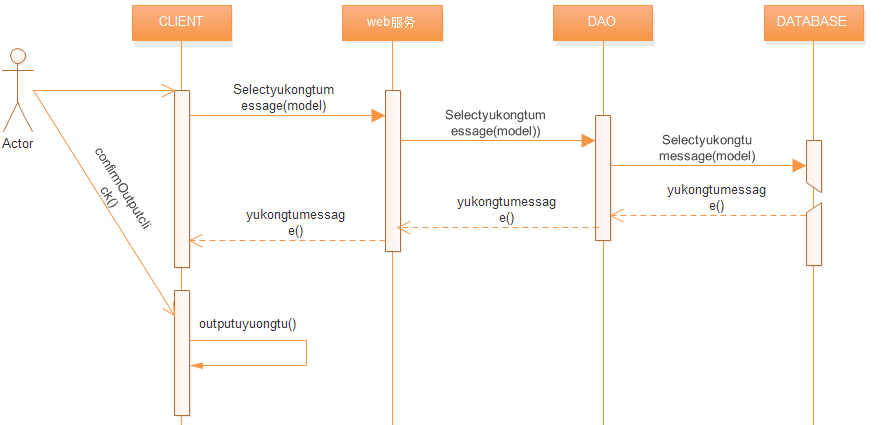


图4.4 预控图数据查询与导出时序图

Fig .4.4 Mapping data query and export timing diagram

4.功能模块界面实现。

如图4.5所示：

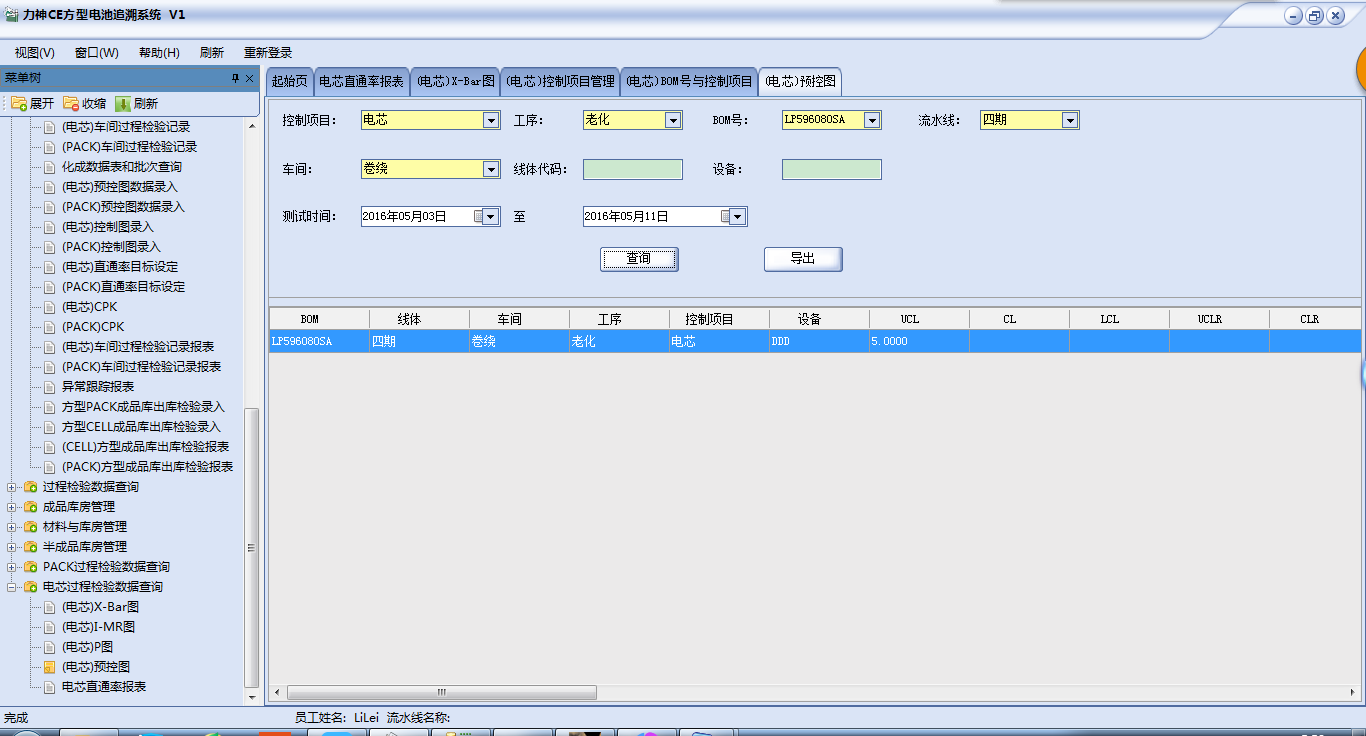


图4.5 预控图数据查询与导出界面的实现结果图

Fig .4.5 Mapping data query and export interface design

5.代码及其说明

for (int i = 0; i < dgv.Columns.Count; i++)

{

if (!(dgv.Columns[i].Visible == false || dgv.Columns[i].HeaderText == "选择" || dgv.Columns[i].HeaderText == "操作"))

{

dtlm.Columns.Add(i.ToString() + "\_" + dgv.Columns[i].HeaderText.ToString(), typeof(string));

}

}

//对list全部记录进行遍历

for (int row = 0; row < dgv.Rows.Count; row++)

{

DataRow drlm = dtlm.NewRow();

int i = 0;

for (int col = 0; col < dgv.Columns.Count; col++)

{

if (!(dgv.Columns[col].Visible == false || dgv.Columns[col].HeaderText == "选择" || dgv.Columns[col].HeaderText == "操作"))

{

string value = dgv.Rows[row].Cells[col].FormattedValue.ToString().Trim().Replace(",", "/");

if (isNumeric(value))

{

drlm[i] = "　" + value;

}

else

{

drlm[i] = value;

}

if (dgv.Columns[col].HeaderText == "批次号")

{

drlm[i] = dgv.Rows[row].Cells[col].FormattedValue.ToString().Trim().Replace(",", "/");

}

else

{

drlm[i] = dgv.Rows[row].Cells[col].FormattedValue.ToString().Trim().Replace(",", "-");

}

i++;

}

}

dtlm.Rows.Add(drlm);

}

ds = SplitTable(dtlm, 60000);

SaveFileDialog saveDlg = newSaveFileDialog();

saveDlg.Filter = "CSV文件(\*.csv)|\*.csv";

saveDlg.FileName = ds.Tables[0].TableName + DateTime.Now.ToString("yyyyMMddHHMMss");

if (saveDlg.ShowDialog() == DialogResult.OK)

{

int size = 1024;

int sizeCnt = (int)Math.Ceiling((double)ds.Tables[0].Rows.Count / (double)2000);

if (sizeCnt == 0)

sizeCnt = 1;

StreamWriter sw = newStreamWriter(saveDlg.FileName, false, Encoding.Default, size \* sizeCnt);

foreach (DataColumn column in ds.Tables[0].Columns)

{

string name = column.ColumnName.Substring(column.ColumnName.IndexOf("\_") + 1);

sw.Write(name + ",");

}

sw.WriteLine();

foreach (DataRow row in ds.Tables[0].Rows)

{

foreach (DataColumn col in ds.Tables[0].Columns)

{

sw.Write(row[col.ColumnName].ToString() + ",");

}

sw.WriteLine();

}

sw.Flush();

sw.Close();

Functions.ShowMessageBox("导出完成!", Const.EVENT\_INFO);

}

6.预控图模板实现。

如图4.6所示：

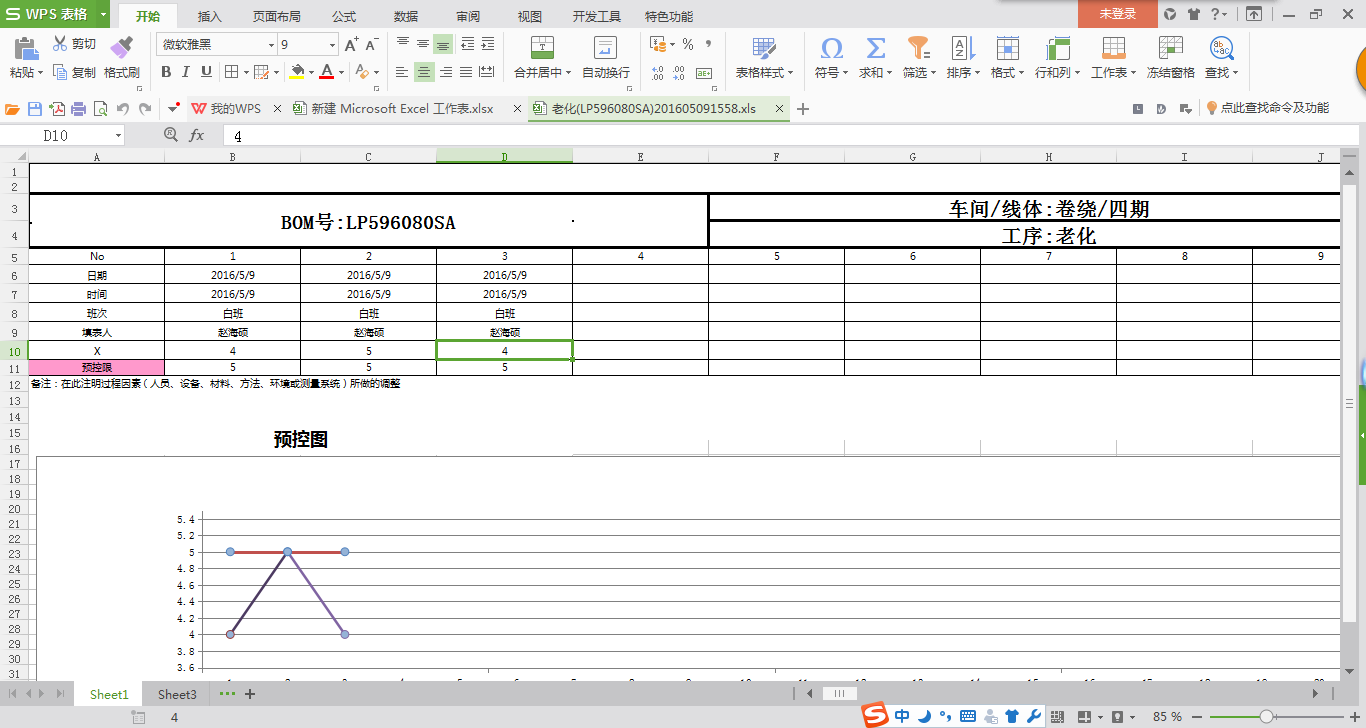


图4.6 预控图excel模板的实现结果图

Fig .4.6 Mapping excel template design

### 4.2.3 控制项目添加设计与实现

1.功能简介

（电芯）控制项目管理。在此功能模块中，管理员可以为电芯控制项目新增控制项目，同时也可以进行相应的数据维护（修改、删除）。在添加新项目时，需要指定项目类型，类型包含两种，分别是X-Bar图和预控图。同时也需要指定工序和样本容量和频率，用于后期的统计。

2.功能数据基本要求

控制项目管理中，管理员首先需要确定添加的控制项目的内容，当管理员点击添加按钮后，界面会显示需要添加的控制项目的基本信息的选项，如控制项目名称，类型，样本容量，频率，工序和工程范围。其中，控制项目名称、样本容量，频率工程范围需要管理员自行输入，控制项目的类型和工序是已有的数据，需要管理员来选择，无需自定义。

3.程序内部功能实现流程

在程序内部，管理员录入好数据后，点击添加按钮，客户端程序会调用web服务层的创建控制项目的方法，同时将录入好的数据封装到模型中，一并传递到服务层添加函数中，服务层会调用数据库访问层的添加方法，将数据插入到数据库中，如果添加成功，添加函数会返回新添加的控制项目的基本信息，经由服务层传递到客户端进行结果显示。如图4.7所示：

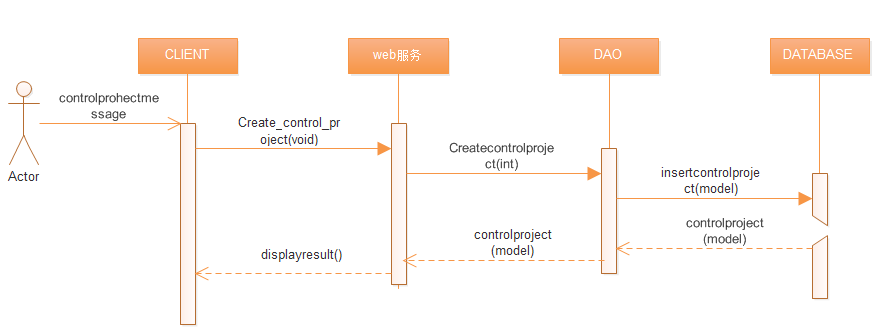


图4.7 控制项目添加时序图

Fig .4.7 Control project to add timing diagram

4.功能模块界面的实现。

如图4.8所示：

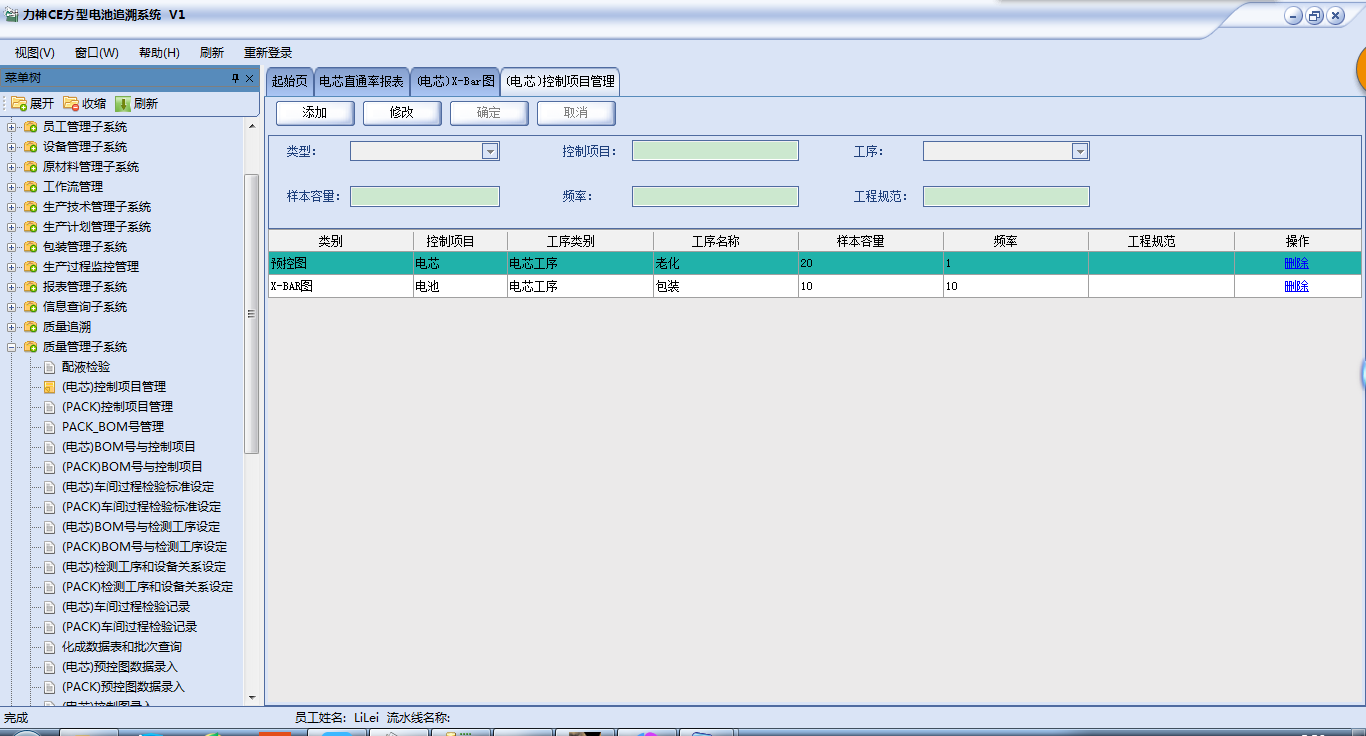


图4.8 控制项目添加界面的实现结果图

Fig. 4.8 Control project to add interface design

5.代码及其说明

if (checkInput())

{

return;

}

//插入参数设置 seq字段的取值在设置状态时进行

CP = new Client.ServerGMControlProject.MdGMControlProject();

Functions1.FormToEntity(groupBox2, CP);

//CP.PROCESSESID = this.cmbPROCESSNAME.SelectedValue.ToString();

CP.PROCESSNAME = this.cmbPROCESSNAME.Text;

DataSet ds = DL.SelectProcessID(this.cmbPROCESSNAME.Text.ToString());

CP.PROCESSESID = ds.Tables[0].Rows[0]["PROCESSESID"].ToString();

CP.CPTYPE = int.Parse(\_type);

CP.CONTROL\_NAME = this.txtCONTROL\_NAME.Text;

CP.SAMPLE\_CAPACITY = this.txtSAMPLE\_CAPACITY.Text;

CP.FREQUENCY = this.txtFREQUENCY.Text;

CP.STANDARD = this.txtSTANDARD.Text;

int nResult = 0;

if (this.status == Const.STATUS\_INSERT)

{

if (\_type == Const.GM\_CELL\_TYPE)

{

CP.CPTYPE = int.Parse(Const.GM\_CELL\_TYPE);

int ret = serverCommand.CheckControlType(CP);

if (ret >= 1)

{

Functions.ShowMessageBox("电芯工序中一个“控制项目”名称只会有一种图，不会有多种图！", Const.EVENT\_INFO);

return;

}

CP.PROCESS\_TYPE = "电芯工序";

nResult = serverCommand.insertControlInfo(CP);

}

if (\_type == Const.GM\_PACK\_TYPE)

{

CP.CPTYPE = int.Parse(Const.GM\_PACK\_TYPE);

int ret = serverCommand.CheckControlType(CP);

if (ret >= 1)

{

Functions.ShowMessageBox("PACK工序中一个“控制项目”名称只会有一种图，不会有多种图！", Const.EVENT\_INFO);

return;

}

CP.PROCESS\_TYPE = "PACK工序";

nResult = serverCommand.insertControlInfo(CP);

}

//nResult = serverCommand.insertControlInfo(CP);

}

else

{

//日志参数

CP = (ServerGMControlProject.MdGMControlProject)Functions.setLogInfo(CP, this, Share.ShareConst.LOG078, Share.ShareConst.LOG\_UPDATE);

//nResult = serverCommand.updateControlInfo(CP);

}

if (this.status == Const.STATUS\_UPDATE)

{

if (\_type == Const.GM\_CELL\_TYPE)

{

CP.CPTYPE = int.Parse(Const.GM\_CELL\_TYPE);

CP.PROCESS\_TYPE = "电芯工序";

nResult = serverCommand.updateControlInfo(CP);

}

if (\_type == Const.GM\_PACK\_TYPE)

{

CP.CPTYPE = int.Parse(Const.GM\_PACK\_TYPE);

CP.PROCESS\_TYPE = "PACK工序";

nResult = serverCommand.updateControlInfo(CP);

}

//nResult = serverCommand.updateControlInfo(CP);

//Functions.CheckReturnFlag(nResult);

// this.setState(Const.STATUS\_INIT);

//return;

}

//

if (nResult == -9999)//存在重复记录

{

Functions.ShowMessageBox(sz\_MSG001, Const.CHECK\_INFO);

return;

}

Functions.CheckReturnFlag(nResult);

if (nResult != -1)

{

if (\_type == Const.GM\_CELL\_TYPE)

{

CP.CPTYPE = int.Parse(Const.GM\_CELL\_TYPE);

//CP.PROCESS\_TYPE = "电芯工序";

this.dataGridView1.DataSource = serverCommand.SelectInfo(CP).Tables[0];

}

if (\_type == Const.GM\_PACK\_TYPE)

{

CP.CPTYPE = int.Parse(Const.GM\_PACK\_TYPE);

//CP.PROCESS\_TYPE = "PACK工序";

this.dataGridView1.DataSource = serverCommand.SelectInfo(CP).Tables[0];

}

}

setState(Const.STATUS\_INIT);

### 4.2.4 控制项目与BOM号绑定设计与实现

1. 功能简介

（电芯）BOM号与控制项目管理：在此功能模块中，主要是进行BOM号与控制项目的绑定。在进行添加时，需要指定统计图类型和工序，选定控制项目和BOM号绑定，同时需要设定控制图中心线（CL）、统计图波动上限（UCL）和波动下限(LCL)以及工程规范上限和下限，设定预控图的中心线（CL）。

2. 功能数据基本要求

BOM号与控制项目绑定中，管理员首先需要获取BOM号基本信息，以及控制项目的基本信息，然后管理选定需要绑定的BOM号和控制项目名称，这两部分无需手动输入，只需从下拉列表框中选择即可。在完成BOM号和控制项目的选定后，还需要提供工序、类型和控制图需要的相关信息，如上限，下限，中心线等基本内容，还有工程范围上限和工程范围下限。除了工序和类型需要从下拉框选择外，其他内容，管理员可以根据需要手动输入相关的值。

3. 程序内部功能实现流程

在程序内部，当管理员发起BOM号与控制项目绑定事务后，程序内部首先会初始化基本信息，如BOM号列表、控制项目列表、类型列表和工序列表。其次管理员录入好数据后，点击添加按钮，客户端程序会调用web服务层的创建控制项目的方法，同时将录入好的数据封装到模型中，一并传递到服务层添加函数中，服务层会调用数据库访问层的添加方法，将数据插入到数据库中，如果添加成功，添加函数会返回新添加的控制项目的基本信息，经由服务层传递到客户端进行结果显示。如图4.9所示：

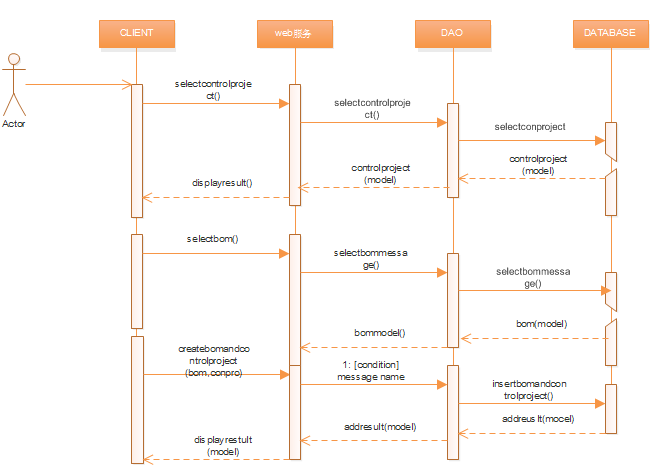


图4.9 BOM号与控制项目绑定时序图

Fig. 4.9 BOM number and control project binding timing diagram

4. 功能模块界面实现。

如图4.10所示：

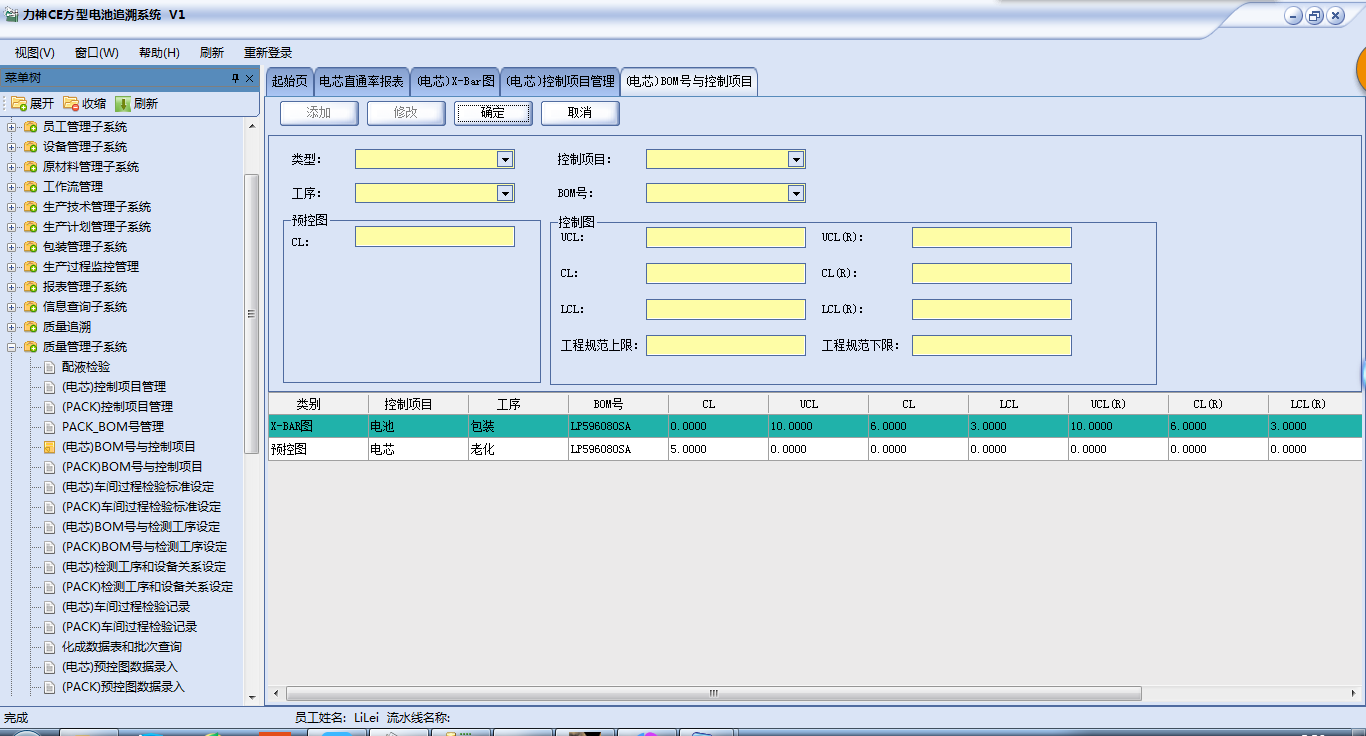


图4.10 BOM号与控制项目绑定界面的实现结果图

Fig .4.10 BOM number and control project binding interface design

5. 代码及其说明

//画面输入验证

if (CheckInput() == false)

return;

mdcontrol = new Client.ServerGMBomAndControl.MdGMBomAndControl();

Functions1.FormToEntity(this.groupBox4, mdcontrol);

mdcontrol.CONSEQ = this.cmbPROCESSNAME.SelectedValue.ToString();

this.cmbInspectLock.SelectedIndex = 0;//默认先设置为不锁定 20141104

mdcontrol.INSPECT\_LOCK = this.cmbInspectLock.Text.Trim(); //20130510

mdcontrol.CONTROL\_TYPE = this.cmbCONTROL\_TYPE.Text.Trim();//类型

int returnValue = 0;

//添加状态时

if (this.status == Const.STATUS\_INSERT)

{

if (\_type == Const.GM\_CELL\_TYPE)

{

mdcontrol.CPTYPE = int.Parse(Const.GM\_CELL\_TYPE);

int ret = serverCommand.CheckInsert(mdcontrol);

if (ret > 0)

{

Functions.ShowMessageBox("电芯工序相同设定已经存在，请确认！", Const.EVENT\_INFO);

return;

}

//CP.PROCESS\_TYPE = "电芯工序";

returnValue = serverCommand.InsertTalbe(mdcontrol);

}

if (\_type == Const.GM\_PACK\_TYPE)

{

mdcontrol.CPTYPE = int.Parse(Const.GM\_PACK\_TYPE);

int ret = serverCommand.CheckInsert(mdcontrol);

if (ret > 0)

{

Functions.ShowMessageBox("PACK工序相同设定已经存在，请确认！", Const.EVENT\_INFO);

return;

}

returnValue = serverCommand.InsertTalbe(mdcontrol);

}

else

{

returnValue = serverCommand.UpdateTable(mdcontrol);

}

Functions.CheckReturnFlag(returnValue);

this.dataGridView1.DataSource = serverCommand.GetShowList(mdcontrol).Tables[0];

SetState(Const.STATUS\_INIT);

### 4.2.5 控制图数据录入的设计与实现

1. 功能简介

电芯）控制图数据录入：此功能模块中，主要是维护电芯的控制图数据。包括对控制图的数据的添加修改和删除操作，为后期控制图的产生提供数据支持。

2. 功能数据基本要求

控制图数据录入模块中，管理员发起控制图数据录入事务后，界面提供的数据接口包含了类型，工序、控制、BOM号、流水线、车间和操作时间、操作人、确认人、班次等数据录入下拉列表以及样本值录入表。管理员需要手动录入的是样本值和批次号，其他的值可由下拉列表中进行选定。

3. 程序内部功能实现流程

在程序内部，管理员录入好数据后，点击添加按钮，客户端程序会调用web服务层的创建控制项目的方法，同时将录入好的数据封装到模型中，一并传递到服务层添加函数中，服务层会调用数据库访问层的添加方法，将数据插入到数据库中，如果添加成功，添加函数会返回新添加的控制项目的基本信息，经由服务层传递到客户端进行结果显示。如图4.11所示：

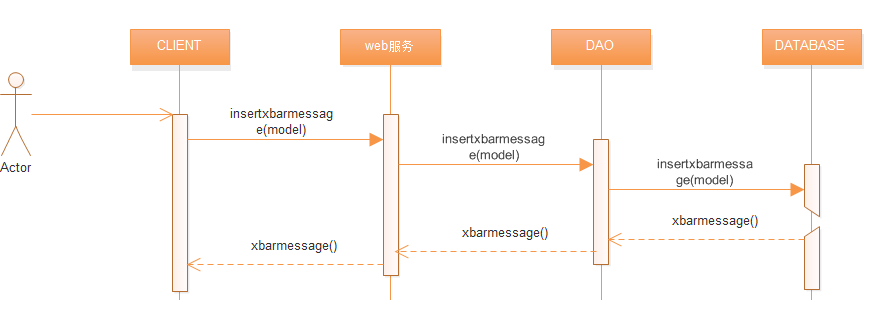


图4.11 控制图数据录入时序图

Fig 4.11 Control chart data entry timing diagram

4.功能模块界面实现。

如图4.12所示：

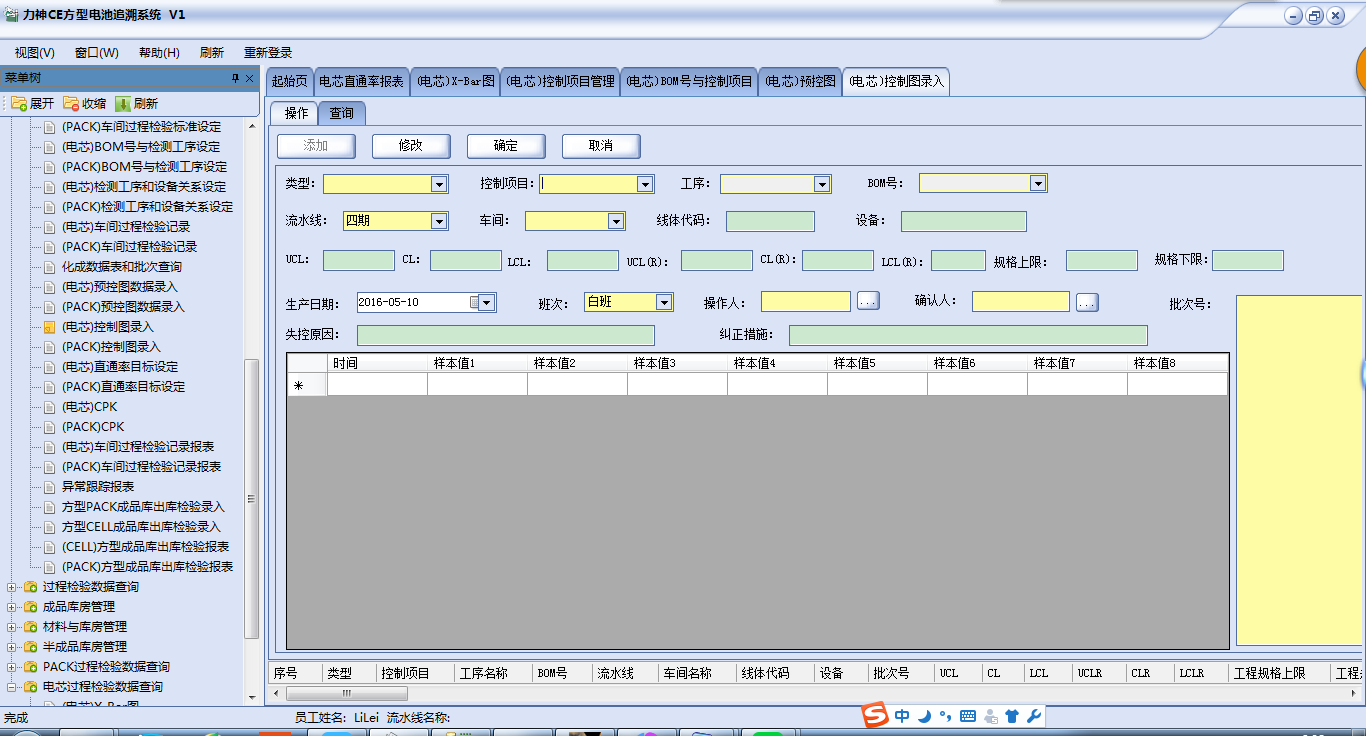


图4.12 控制图数据录入界面实现结果图

Fig 4.12 Control chart data entry interface design

5.代码及其说明

if (checkInput())

{

return;

}

if (string.IsNullOrEmpty(this.txtLINECODE.Text.ToString()) &&string.IsNullOrEmpty(this.txtEquipment.Text.ToString()))

{

Functions.ShowMessageBox("设备与线体二者至少应填写一个!");

return;

}

//checkLINECODEDEVICE();

//校验相同条件下的线体或设备是否重复

XRDI.TYPE = this.cmbCONTROL\_TYPE.Text;

XRDI.PROJECT = this.cmbCONTROL\_NAME.Text;

XRDI.PROCESS = this.cmbPROCESSESNAME.Text;

XRDI.BOM = this.cmbBOMNO.Text;

XRDI.LINE = this.cmbPipeLine.Text;

XRDI.WORKSHOP = this.cmbTemplate.Text;

XRDI.TEST\_DATE = this.calTestTime.Value.ToString("yyyy-MM-dd") + "" + System.DateTime.Now.ToString("HH:mm:ss");

XRDI.CLASSNAME = this.cmbSHIFT.Text;

XRDI.LINECODE = this.txtLINECODE.Text;

XRDI.DEVICE = this.txtEquipment.Text;

if (!this.txtLINECODE.Text.Equals(linecode) || !this.txtEquipment.Text.Equals(equipment))

{

DataSet ds = serverCommand.queryLINECODEDEVICE(XRDI, \_type);

if (ds != null)

{

if (ds.Tables[0].Rows.Count > 0)

{

Functions.ShowMessageBox("已经存在相同条件下的线体或设备，请确认!");

return;

}

}

}

if (this.dataGridView2.Rows.Count == 0 || this.dataGridView2.Rows.Count == 1)

{

Functions.ShowMessageBox("没有样本值，无需插入数据!");

return;

}

if (this.dataGridView2.Rows[0].Cells[1].Value == null)

{

Functions.ShowMessageBox("第一个样本值为空，不能插入!");

return;

}

for (int n = 0; n <this.dataGridView2.Rows.Count - 1; n++)

{

if (this.dataGridView2.Rows[n].Cells[0].Value != null)

{

if (!string.IsNullOrEmpty(this.dataGridView2.Rows[n].Cells[0].Value.ToString()))

{

if (this.dataGridView2.Rows[n].Cells[1].Value == null)

{

Functions.ShowMessageBox("样本时间不为空，所在行的第一个样本值不能为空!");

return;

}

if (string.IsNullOrEmpty(this.dataGridView2.Rows[n].Cells[1].Value.ToString()))

{

Functions.ShowMessageBox("样本时间不为空，所在行的第一个样本值不能为空!");

return;

}

}}

### 4.2.6 预控图数据录入的设计与实现

1.功能简介

（电芯）预控图数据录入。

2.功能数据基本要求及其实现流程。

预控图数据录入管理中，管理员发起预控图数据录入事务后，界面提供的数据接口包含了类型，工序、控制、BOM号、流水线、车间和操作时间、操作人、确认人、班次等数据录入下拉列表以及样本值录入表。管理员需要手动录入的是样本值和批次号，其他的值可由下拉列表中进行选定。。

在程序内部，管理员录入好数据后，点击添加按钮，客户端程序会调用web服务层的创建控制项目的方法，同时将录入好的数据封装到模型中，一并传递到服务层添加函数中，服务层会调用数据库访问层的添加方法，将数据插入到数据库中，如果添加成功，添加函数会返回新添加的控制项目的基本信息，经由服务层传递到客户端进行结果显示。如图4.13所示：

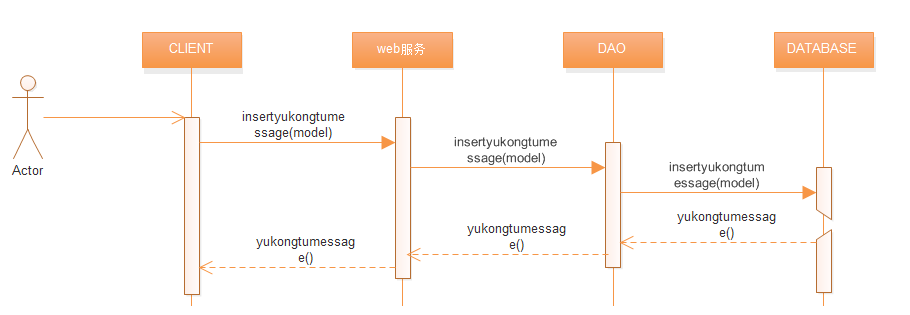


图4.13 预控图数据录入时序图

Fig. 4.13 Mapping data entry timing diagram

3.功能模块界面设计与实现。

如图4.14所示：

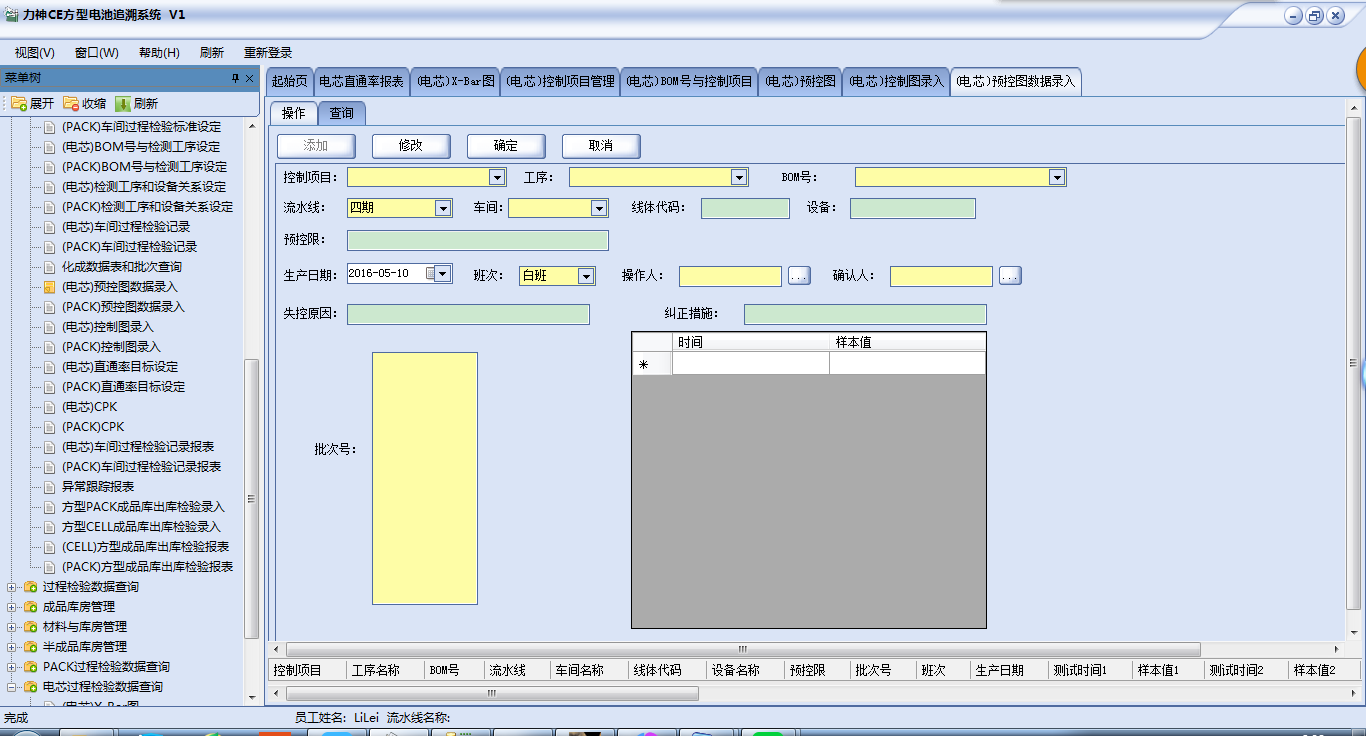


图4.14 预控图数据录入界面的实现结果图

Fig .4.14 Mapping data entry interface design

4. 代码实现与说明

//校验相同条件下的线体或设备是否重复

XRDI.PROJECT = this.cmbCONTROL\_NAME.Text;

XRDI.PROCESS = this.cmbPROCESSESNAME.Text;

XRDI.BOM = this.cmbBOMNO.Text;

XRDI.LINE = this.cmbPipeLine.Text;

XRDI.WORKSHOP = this.cmbTemplate.Text;

XRDI.TEST\_DATE = this.calTestTime.Value.ToString("yyyy-MM-dd") + "" + System.DateTime.Now.ToString("HH:mm:ss");

XRDI.CLASSNAME = this.cmbSHIFT.Text;

XRDI.LINECODE = this.txtLINECODE.Text;

XRDI.DEVICE = this.txtEquipment.Text;

//判断所填的批次号是否是改BOM号的电池

//生产日+班次和测试时间的校验

string producetime = this.calTestTime.Value.ToString("yyyy-MM-dd");

string shift = this.cmbSHIFT.Text;

string start = "";

string end = "";

for (int k = 0; k <this.dataGridView2.Rows.Count - 1; k++)

{

if (this.dataGridView2.Rows[k].Cells[0].Value != null)

{

if (!string.IsNullOrEmpty(this.dataGridView2.Rows[k].Cells[0].Value.ToString()))

{

string value = this.dataGridView2.Rows[k].Cells[0].Value.ToString();

int l = k + 1;

if (shift.Equals("白班"))

{

start = producetime + " 08:00";

end = producetime + " 20:00";

if (Convert.ToDateTime(value) >= Convert.ToDateTime(start) &&Convert.ToDateTime(value) <Convert.ToDateTime(end))

{

}

else

{

Functions.ShowMessageBox("第" + l + "行测试时间超出" + shift + "时间范围!");

return;

}

}

if (shift.Equals("夜班"))

{

start = producetime + " 20:00";

string newstart = Convert.ToDateTime(producetime).AddDays(1).ToString("yyyy-MM-dd") + " 00:00";

end = Convert.ToDateTime(producetime).AddDays(1).ToString("yyyy-MM-dd") + " 08:00";

if ((Convert.ToDateTime(value) >= Convert.ToDateTime(start) &&Convert.ToDateTime(value) <= Convert.ToDateTime(producetime + " 23:59"))

|| (Convert.ToDateTime(value) >= Convert.ToDateTime(newstart) &&Convert.ToDateTime(value) <Convert.ToDateTime(end))

)

{

}

else

{

Functions.ShowMessageBox("第" + l + "行测试时间超出" + shift + "时间范围!");

return;

## 4.3 本章小结

本章对统计过程管理进行了总体设计（即详细设计）。通过对系统的各部分功能流程就行了系统的分析与设计。首先，绘制了功能模块的时序图，明确了本功能模块具体的流程以及各个层的实现过程。然后绘制了该部分功能模块的界面展示图。通过总体设计，给后期的编码实现提供了有效的依据和指导。

# 第五章 结论

## 5.1项目归纳与总结

### 5.1.1 项目归纳

该项目来源于天津市电子计算机研究所项目一部，是从电池行业生产制造执行系统中提炼出来的一个小型系统，项目专门针对制造型企业对产品质量检验的管理分析而设计。

该系统的开发经历了相关技术资料收集与整理，而后对实际过程中的需求进行了分析，在明确了需求后，开展了项目的概要设计。概要设计中主要对系统的结构和数据库的逻辑设计进行了详细的论述。紧接着，我们对项目进行了总体设计与实现，在此过程中，进行了各个功能模块的详细设计，进而进行了编码工作。

完成项目的编码后，我们对项目进行了测试。保证了项目的可应用性与代码的健壮性。

### 5.1.2 项目结论

此项目基本上实现了设计过程中所涵盖的功能模块，也最终实现了制造型小型企业的生产制造过程中的统计过程控制管理的基本功能。基本上满足了用户的需求。项目已经应用到了生产制造的过程中。

## 5.2 新发展趋势

当今世界一趋向智能化与多元化，传统的生产制造过程的质量管理紧紧是通过人的手动数据录入和导出分析报表来实现质量的管理。

但是，对于科技较为发达的当今世界，统计过程的数据不应该人为的手动录入，然后产生报表进行分析。这无疑有点事后诸葛。所以，新的时代背景下，新的技术支持下，统计过程的数据录入应该由计算机智能的进行切面方式采集，智能的进行预警。

## 5.3 课题尚待改进的建议

### 5.3.1 课题尚存在的问题

本课题中存在的问题有如下几点：

（1）数据录入的非智能性。系统中的数据样本采集仅仅依赖于人工的手动输入，这无疑增加了潜在的人为因素产生的误差。

（2）程序的运行效率不高。从数据的采集、收录到最后产生出统计报表，需要耗费大量的时间，这无疑会影响生产效率。

### 5.3.2 课题改进的建议

针对课题中存在的问题，有如下几点仅供参考：

（1）优化数据录入。采用计算机智能切面式数据录入。

（2）优化程序或重构程序。程序开发过程中所使用的是早期的编程语言，早期的编程技术尚不成熟，所以系统的程序应该使用当前的最新的运行效率最高的技术来优化或重构。

## 5.4 本章小结

本章对整个课题的设计与开发进行了总结，分析了当前项目所存在的问题，并对问题进行了相关的建议。

# 

# 参考文献

[1] 张正礼，王坚宁. ASP.NET4.0从入门到精通. 清华大学出版社.P1

[2] 高宏，李俊民. ASP.NET典型模块与项目实战大全. 清华大学出版社.P1

[3] 戴宗友,张伍荣,杨辉.C#程序设计实训.清华大学出版社,2009.P1

[4] 甘志,C#高级应用开发篇.清华大学出版社,2008.P2

[5] 蒋金楠.ASP.NET MVC 4框架揭秘.电子工业出版社，2013.P2

[6] 高云.jQuery技术内幕：深入解析jQuery架构设计与实现原理.机械工业出版社，2014.P2

[7] 王玫丽.浅析OA 系统(网络办公自动化)的开发.科技信息(科学教研),2008.P2

[8] 雍珣. 基于WEB 的办公自动化系统的设计与实现. 山西广播电视大学学

报,2009.P2

[9] 周德荣.基于.NET 环境的OA 系统框架设计与实现.福建电脑,2007.P3

[10] 朱玉超,鞠艳,王代勇.ASP.NET 项目开发教程.北京：电子工业出版社,2008.P2

[11] 陈伟.SQL Server 2005 数据库应用与开发教程.北京：清华大学出版社,2007.P2

[12] 尹增明.ASP.NET Web 应用开发.机械工业出版社,2008.P5

[13] 李红日,宋俊兰.用SQL 语句实现关系代数运算探讨.科技信息(学术版)，2008.P5

[14] Nielsen Paul.SQL Server 2005 宝典.北京:人民邮电出版社,2008.P5

[15] 胥光辉.软件工程方法与实践.机械工业出版社,2009.P5

# 致 谢

在本系统设计的过程中，老师为我们选定了该系统的方向，为我们分析了基本的客户需求。为我们的系统拟定了一个大致的方向，我们的团队就依着该方向进行领域分析，可行性分析和需求分析。我们为系统搭好了框架，并且老师对我们框架的细节进行完善，给予了细致的指导，给我们提出了具有建设性的建议，老师以他那渊博的知识、开阔的视野和那丰富的实战经验在写项目的时候给予了我们很大的帮助。

在这里我还要感谢我们团队里的每一个人，我们这个团队一共由留个人组成，因为我们的分工很明确，所以每个人都有各自的任务，我们团队里的每一个人都非常认真，负责的做着自己的那一份任务，没有以一个人一个人原因而耽误项目的进度。在做项目的过程中，我们虽然各自有着自己的工作，但是我们还互相帮助。并且一个团队里有多少人就有多少种思路，我们都会坐下来一起商量。就这样我们的团队从不稳定，慢慢的变的稳定。在这里我要感谢我的团队，感谢他们的兢兢业业，因为有了他们我们的项目才能够顺利，如期的完成。

在完成项目的时候，同学们给了我很多的意见和建议。同时我还参考了好多网上的资料，也给了我很大的启发。还有在这里还是要着重的感谢一下我的指导老师。感谢老师在我的专业思维及技能方面给予我的帮助，在老师的细心指导和鼎力的支持下让我变的非常有信心。就这样在这么多人的帮助下我顺利的完成了这个系统 。在这里感谢所有授我以业的老师，如果没有这么多年知识的积累，我也就没有有够的信心来完成这个系统。在这里，我非常诚恳的请老师对我的系统多加批评和指正，这样我就能够及时的修改我的系统，使我的系统更加的完善。