专 业 学 位 硕 士 学 位 论 文

晶澳太阳能SPC软件系统的设计与实现

The Design and Implementation of SPC Software System for JAsolar

作 者 姓 名： 李斌

工 程 领 域： 软件工程

学 号： 41317206

指 导 教 师：

完 成 日 期：

大连理工大学

Dalian University of Technology

大连理工大学学位论文独创性声明

作者郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究和工作所取得的成果。据我所知，除本文中已经给予注明的引用内容和致谢的地方除外，本论文不包含其他的个人或集体已发表的研究成果，也不包含其他已申请的学位或其他用途曾使用过的成果。与我一起工作的同志对本研究所做来的贡献均已在论文中做了明确的说明并表示了谢意。

若有不实之处，本人愿意承担相关法律责任。

学位论文题目： 晶澳太阳能SPC软件系统的设计与实现

作 者 签 名 ： 日期： 年 月 日

摘 要

目前我公司质量管理体系中质量相关数据分散，采用人工统计，费事费力，而且不能全面反应实际情况，缺少系统化管理，质量事故未能及时预测，缺乏对质量数据的收集和分析。为进一步提高产品品质，预防质量事故，提高劳动生产率．经过公司高层决定由我负责进行SPC系统的设计开发和实施， SPC是一种品质工具，主要是对生产线上收集数据，通过统计分析的手法去发现生产中的异常现象，提前采取预防措施。

SPC系统的设计与开发，在开发过程中运用了B/S架构，面向对象的开发方法，并结合软件工程开发理论，从SPC系统实际出发，对在软件开发过程中所需要解决的问题进行深入分析，通过对国内外发展现状进行描述，并对课题中运用到的关键技术进行详细介绍；从系统需求分析着手分析，通过JA SPC 系统开发，实现系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视功能功能。再通过系统详细分析设计，用E-R图对系统进行数据库设计，并对每个模块用时序图进行分析、最终实现了系统，通过运用黑盒测试方法，完成系统测试功能。

本系统主要由系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视功能模块组成。对SPC系统的开发，达到对公司产品质量的监控，提高了公司产品质量管理，达到了课题的要求。

关键词：质量管理；SPC；管理系统

The Design and Implementation of SPC Software System for JAsolar

Abstract

At present ,JAsolar quality management system in the quality related data dispersion, the manual statistics, cumbersome effort, and can reflect the actual situation, the lack of systematic management, quality accident failed to timely prediction lack of quality data collection and analysis. To further improve product quality, prevent quality accidents, improve labor productivity. After the company decided by the company's high level of development and implementation of SPC system, SPC is a quality tool, mainly on the production line to collect data, through statistical analysis of the approach and to find abnormal phenomena in production, to take preventive measures in advance.

The Design and Implementationg of SPC system, in the development process of the B/S architecture, object-oriented development method, and combined with software engineering development theory, from the SPC system, the actual situation of the software development process, the need to solve the problem, through to the domestic and international development process, and to analyze the key technologies, system requirements analysis, system design, SPC SPC system, data collection, reporting, SPC analysis, Ocap system, alarm monitoring functions. Through the system detailed analysis and design, using E-R chart to design the system, and analyze the sequence diagram of each module, and finally realize the system, through the use of black box testing method, complete system testing function.

This system is mainly composed of the system, SPC plan and rule setting, data collection, report, SPC analysis, Ocap system, alarm monitoring function module. The development of SPC system to achieve the quality of the company's products, improve the company's product quality management, to meet the requirements of the project.

Keywords：Quality management;SPC;Management system

目 录

[摘 要 I](#_Toc428365036)

[Abstract II](#_Toc428365037)

[1 绪论 - 1 -](#_Toc428365038)

[1.1 课题研究背景和意义 - 1 -](#_Toc428365039)

[1.2 现状分析 - 2 -](#_Toc428365040)

[1.3 论文工作和章节安排 - 2 -](#_Toc428365041)

[2 关键技术 - 3 -](#_Toc428365042)

[2.1 .NET简介 - 3 -](#_Toc428365043)

[2.2 ORACLE 数据库 - 4 -](#_Toc428365044)

[2.3 B/S与C/S - 4 -](#_Toc428365045)

[2.4 三层架构 - 5 -](#_Toc428365046)

[3 系统需求分析 - 7 -](#_Toc428365047)

[3.1 业务需求描述 - 7 -](#_Toc428365048)

[3.2 功能性需求分析 - 7 -](#_Toc428365049)

[3.2.1 系统设置需求分析 - 7 -](#_Toc428365050)

[3.2.2 SPC计划及规则设定需求分析 - 8 -](#_Toc428365051)

[3.2.3 数据收集需求分析 - 9 -](#_Toc428365052)

[3.2.4 报表需求分析 - 10 -](#_Toc428365053)

[3.2.5 SPC分析需求分析 - 10 -](#_Toc428365054)

[3.2.6 OCAP系统需求分析 - 11 -](#_Toc428365055)

[3.2.7 警报监视需求分析 - 12 -](#_Toc428365056)

[3.3 非功能性需求 - 12 -](#_Toc428365057)

[4 系统设计 - 14 -](#_Toc428365058)

[4.1 系统功能模块总体设计 - 14 -](#_Toc428365059)

[4.2 系统功能模块详细设计 - 14 -](#_Toc428365060)

[4.2.1 系统设置管理设计 - 14 -](#_Toc428365061)

[4.2.2 SPC计划及规则设定设计 - 15 -](#_Toc428365062)

[4.2.3 数据收集设计 - 16 -](#_Toc428365063)

[4.2.4 报表设计 - 16 -](#_Toc428365064)

[4.2.5 SPC分析设计 - 17 -](#_Toc428365065)

[4.2.6 OCAP系统设计 - 17 -](#_Toc428365066)

[4.2.7 警报监控设计 - 18 -](#_Toc428365067)

[4.3 数据库设计 - 18 -](#_Toc428365068)

[4.3.1 数据库设计原则 - 18 -](#_Toc428365069)

[4.3.2 数据库概念设计实现 - 19 -](#_Toc428365070)

[4.3.3 数据库逻辑设计实现 - 19 -](#_Toc428365071)

[5 系统的实现 - 25 -](#_Toc428365072)

[5.1 系统设定管理的实现 - 25 -](#_Toc428365073)

[5.2 SPC计划和规则设定的实现 - 28 -](#_Toc428365074)

[5.3 数据收集的实现 - 29 -](#_Toc428365075)

[5.4 报表的实现 - 30 -](#_Toc428365076)

[5.5 SPC分析的实现 - 33 -](#_Toc428365077)

[5.6 Ocap系统的实现 - 35 -](#_Toc428365078)

[5.7 警报监视的实现 - 38 -](#_Toc428365079)

[6 系统测试 - 40 -](#_Toc428365080)

[6.1 系统测试环境 - 40 -](#_Toc428365081)

[6.2 系统测试方法 - 40 -](#_Toc428365082)

[6.3 系统测试用例 - 41 -](#_Toc428365083)

[6.3.1 系统设置测试用例 - 41 -](#_Toc428365084)

[6.3.2 SPC计划和规则设定测试用例 - 42 -](#_Toc428365085)

[6.3.3 数据收集测试用例 - 42 -](#_Toc428365086)

[6.3.4 报表测试用例 - 43 -](#_Toc428365087)

[6.3.5 SPC分析测试用例 - 43 -](#_Toc428365088)

[6.3.6 Ocap系统测试用例 - 43 -](#_Toc428365089)

[6.3.7 警告监视测试用例 - 44 -](#_Toc428365090)

[6.4 系统测试结论 - 44 -](#_Toc428365091)

[结 论 - 45 -](#_Toc428365092)

[参 考 文 献 - 46 -](#_Toc428365093)

[攻读硕士学位期间发表学术论文情况 - 47 -](#_Toc428365094)

[致 谢 - 48 -](#_Toc428365095)

[大连理工大学学位论文版权使用授权书 49](#_Toc428365096)

# 1 绪论

## 1.1 课题研究背景和意义

晶澳太阳能科技有限公司是全球领先的高性能太阳能光伏产品制造商。扬州电池生产基地作为电池生产制造基地，对太阳能电池片的质量管理相当严格，质量管理手段和方法显得很重要，目前公司质量管理体系中质量相关数据分散，采用人工统计，费事费力，而且不能全面反应实际情况，缺少系统化管理，质量事故未能及时预测，缺乏对质量数据的收集和分析。我公司有专门的质量管理部门，严格遵循质量体系管理规定，为进一步提高产品品质，预防质量事故，提高劳动生产率。经过公司高层决定由我负责建立项目小组进行晶澳（扬州）太阳能科技有限公司的SPC系统的开发和实施。

统计过程控制SPC是英文StatisticsProcessControl的缩写，SPC应用统计学的方法，分析生产过程的样本统计数据，以此数据来判断生产过程中的波动是否处于可接受的状态，分析并在必要时候，调整过程中的参数，从而降低产品质量特性值过多地偏离我们设定的目标值，使整个生产过程维持在仅受偶然因素影响的稳定受控状态，从而提高过程的效能[1]。SPC技术是一种统计控制的技术，它所研究的对象是过程，方法是统计，即对过程采用统计的方法来进行控制，是一种借助数理统计方法的过程控制的工具[2]。SPC的本质就是对生产过程中样本的质量特性数据进行统计分析和做出相应的评价，根据反馈的信息来区分生产过程当中产品质量的正常波动和异常波动，以便对整个过程的异常及时提出预警，提醒管理人员采取措施及时进行相关操作消除异常，恢复生产过程的稳定使过程维持在仅受随机因素影响的受控状态下，从而提高整个生产过程的产品的质量和过程的效能，所体现的是一切以事实为依据的科学管理的思想[3]。

SPC系统有很好的实用价值，我公司是专业的太能能电池片生产基地，电池片质量作为我公司产品的核心竞争力，质量管理手段和方法显得很重要，目前公司质量管理体系中质量相关数据分散，采用人工统计，费事费力，而且不能全面反应实际情况，缺少系统化管理， 质量事故未能及时预测，缺乏对质量数据的收集和分析，为进一步提高产品品质，预防质量事故，因此公司决定进行SPC系统的开发和实施，实现对生产线上收集数据，通过统计分析的手法去发现生产中的异常现象，提前采取预防措施。通过对实时数据的收集和分析也便于事后分析和总结，我们公司希望通过SPC系统的开发和实施进一步提高质量管理水平，从而提高产品的品质，增强企业核心竞争力。

SPC系统实施提高太阳能电池片的质量，对社会有很好的社会效益，太阳能作为绿色新能源，符合节能环保低碳理念，应用太阳能光伏系统对社会具有多方面的好处，包括遏制化石能源的过度消耗，环保，保证能源安全，提高对空间面积的利用率等。太阳能电池片是主要的太阳能发电核心设备之一，电池片的质量关乎太阳能的有效利用，通过SPC系统的开发和实施可以提升公司质量管理，从而提高电池片的质量，从而更大程度的利用低碳绿色新能源。

## 1.2 现状分析

我公司作为专业的太能能电池片生产基地，电池片质量作为我公司产品的核心竞争力，质量管理手段和方法显得很重要，目前公司质量管理体系中质量相关数据分散，采用人工统计，费事费力，而且不能全面反应实际情况，缺少系统化管理，质量事故未能及时预测，缺乏对质量数据的收集和分析。如果我们采用先进的质量管理技术和工具在条件不变的情况下质量就可以得到明显的改进，SPC正是这种行之有效的工具[4]。

## 1.3 论文工作和章节安排

通过晶澳太阳能SPC软件系统的设计与实现，在开发过程中运用了B/S架构，面向对象的开发方法，并结合软件工程开发理论，从SPC系统实际出发，对在软件开发过程中所需要解决的问题进行深入分析，通过对国内外发展现状进行描述，并对课题中运用到的关键技术进行详细介绍；从系统需求分析着手分析，通过JA SPC 系统开发，实现系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视功能功能。再通过系统详细分析设计，用E-R图对系统进行数据库设计，并对每个模块用时序图进行分析、最终实现了系统，通过运用黑盒测试方法，完成系统测试功能。

本文的研究涉及到以下几个方面：

（1） 本文通过对晶澳太阳能SPC软件系统的设计与实现，实现对生产线上收集数据，通过统计分析的手法去发现生产中的异常现象，提前采取预防措施。通过对实时数据的收集和分析也便于事后分析和总结，我们公司希望通过SPC系统的开发和实施进一步提高质量管理水平，从而提高产品的品质，增强企业核心竞争力。

（2） 本文运用过.NET工作平台实现系统的设计，在此过程中以软件工程为指导思想，对系统进行了需求分析、详细分析、系统实现、系统测试等一系列相关工作，在公司现有的质量管理思想理论基础上实现本SPC系统。

（3） 本系统是多部门共同合作使用，每个部门都具体相应的权限进行系统控制，部门之间相互协作完成相关工作。

（4） 本系统基于B/S的开发模式，运用三层架构实现了系统，用户只需要在有网络环境下就可以随时随地进行系统操作。

# 2 关键技术

## 2.1 .NET简介

随着互联网技术的高速发展，Web编程不是一件简单的任务，传统应用程序开发拥有很多结构化语言支持的完好编程模型，而Web应用程序开发混合了标记语言、脚本语言、服务器平台等技术，其中.NET技术得到了广泛的应用。当前微软公司对.NET的定义是：“.NET技术是微软公司面向XML Web服务的平台；它使用一种统一的、个性化的方式将人员、信息和设备紧密地联系在一起；.NET核心技术包.NET Framework以及数据访问技术ADO.NET”。.NET 框架包括一套可被开发者用于任何编程语言的类库。在此之上是许多应用程序模板，这些模板为开发网络站点和网络服务提供了高级组件和服务[5]。Web工作方式以其方便、快捷、易于实现，已深受学术界和工业界的关注，成为企业获取竞争胜利的一项关键技术[6]。

ASP.NET是建立在.NET框架平台上的完全面向对象的系统，.NET框架的体系结构包括五大部分，分别为程序设计语言及公共语言规范（CLS）、应用程序平台（ASP.NET及Windows应用程序等）、ADO.NET及类库、公共语言运行库（CLR）、程序开发环境。

.NET Framework类库包括asp.net、Web Services、WinForms、ADO.net和XML类、基本框架类。.NET Framework是服务与类的集合，在.NET应用程序和操作系统之间以层的形式存在，封装了调试和安全服务等早期各种程序设计语言所具备的许多基本功能。.NET Framework由.NET Framework类库和公共语言运行时两个主要组件组成，.NET Framework类库包括了各任何.NET语言协同工作的类库；共语言运行时提供了异常处理、安全、调试和语言版本支持等功能。

.NET编写的程序被编译两次，第一次编译很慢，主要从.NET源码到语言编译器，而第二次编译则较快，由语言编辑器到MSIL+元数据再到CLR然后是机器代码。Microsoft中间语言MSIL的能帮助语言实现互操作，IL不是字节代码，但很接近字代码。因此，执行应用程序时，IL到机器码的转换速度非常快，CPU无关的指令集，通过CLR将MSIL转为具体CPU代码[7]。.NET编辑过程，如图2.1所示：



图2.1 .NET编辑过程

Fig. 2.1 .NET editing process

## 2.2 ORACLE 数据库

数据库技术是作为数据处理的一门技术而发展起来的，所研究的问题是如何科学地组织和存储数据，如何高效地获取和处理数据，在数据库中采用数据模型来抽象、表示和处理现实世界中的数据。数据库类型主要分为关系数据库与非关系数据库，常用的关系数据库，如Oracle、 SQL Server、MySQL等。

ORACLE数据库系统是美国甲骨文股份有限公司提供的以分布式数据库为核心的一组软件产品，是目前最流行的客户/服务器(CLIENT/SERVER)或B/S体系结构的数据库之一。ORACLE数据库是目前世界上使用最为广泛的数据库管理系统，作为一个通用的数据库系统，它具有完整的数据管理功能；作为一个关系数据库，它是一个完备关系的产品；作为分布式数据库它实现了分布式处理功能[8]。ORACLE数据库的版本很多，本系统采用ORACLE 9iR2。

## 2.3 B/S与C/S

C/S又称Client/Server或客户/服务器模式。服务器通过采用高性能的PC、工作站或小型机，并采用大型数据库系统，如Oracle、Sysbase、Informix或SQL Server。C/S的优点程序在执行过程中的验证、判断，客户端响应速度非常快，当客户端将数据操作成功后，将结果传给服务端，在数据传给服务端时，还可以对数据进行压缩成数据流的方法，传输速度进一步提升。缺点也比较明显，客户端需要安装专用的客户端软件或者相应操作系统，对客户端操作系统有一定的限制，适用于局域网，若要在广域网上运行，则需要通过Web Service支持。

B/S是B表示浏览器，如IE、火狐、谷歌，S表示服务器，服务器需要安装数据库，B/S是通过Internet进行数据交互，90%以上的设置都在服务器上与程序开发上，用户只要能上网就可以进行操作，非常方便。

B/S最大的优点就是对客户端维护为零维护，系统扩展也非常容易，新功能编写测试成功后，直接传到服务器上，就可以使用新功能了，不需要用户下载客户端才能操作，再配合脚本语言，如JavaScript、Jquary等，就可以把网页效果做得特别华美。B/S缺点也比较明显，对网络要求比较高，程序样式受浏览器兼容限制，服务器配置要求高。本系统就是采用B/S结构进行编写的。

B/S结构的关键是不需要开发和安装客户端程序，而是客户端上的浏览器通过浏览Web服务器上的Web应用程序，把程序的执行结果显示到客户端的浏览器上。

Web应用程序图如图2.2所示：

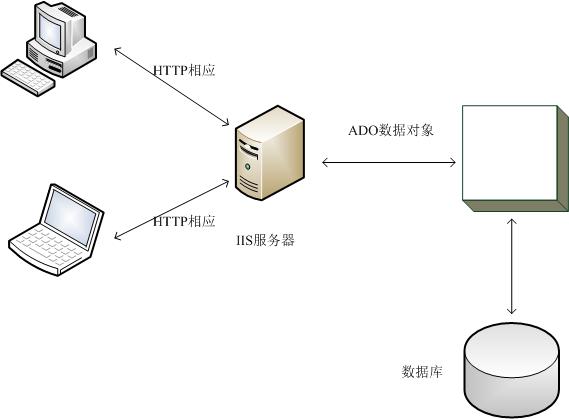


图2.2  Web应用程序图

Fig. 2.2 Web application diagram

## 2.4 三层架构

伴随着对软件复用技术和构件化思想进行深入的研究，目前对于管理信息系统的设计也不再像之前采用的“程序=数据结构+算法”设计，而更多采用的是对软件的框架和业务流程的综合研究与设计[9]。

本系统采用三层架构，完美的三层架构是修改表现层而不用去修改逻辑层，修改逻辑层不用去修改数据层。我们运用三层结构的目的是使项目结构更加清楚，分工更加明确，有利于后期的维护与升级。

三层架构包括了表现层、业务逻辑层与数据访问层。

（1） 表现层（USL）：主要表示WEB方式，用于接受用户请求，以及数据的返回，不客户端供应程序的访问。

（2） 业务逻辑层（BLL）：主要负责对数据层的操作，主要提供对数据层操作的方法。

（3） 数据访问层（DAL）：完成执行业务层BLL中的方法，对各个数据文件的操作。

系统开发的总体架构，三层架构如图2.3所示。

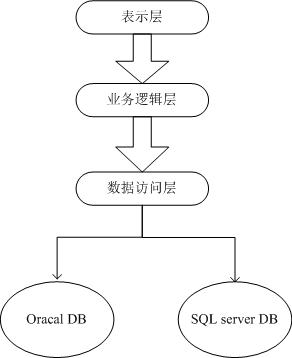


图2.3 三层架构技术架构图

Fig. 2.3 Three-tier technical architecture of diagram

# 3 系统需求分析

## 3.1 业务需求描述

生产过程中的质量控制对一个制造型企业来说有重大的意义。所谓生产过程的质量控制，就是将正常质量波动限制在一个合理的范围以内，杜绝异常质量波动的发生[10]。通过对生产过程进行细致的分析，根据反馈的信息数据及时发现系统性因素出现的征兆，并采取相应的措施消除其影响，使其过程维持在仅受随机性因素影响的受控状态，以达到控制质量、提高质量管理水平、降低质量成本的目的。SPC系统能提供及时在线监控，消除异常因素，减少异常波动，已提高制程能力，降低内外部失效成本，规范现有的生产流程。

SPC系统的设计与实现，主要解决SPC开发实施过程中软件系统技术和管理问题。从不同的角度来看，SPC可以被叫做统计质量控制（从质量管理角度），主要任务是检测产品是不是符合质量的指标，如果检测不合格我们所规定的质量指标，我们会进行查找影响其不符合的因素，积极采取措施来修复其过程或优化其设计。SPC也可以被叫做统计性能监控（从过程性能角度），主要任务是及时发现过程中的故障，及时的进行查找，找出故障的隐患，从而采取控制措施避免事故发生[11]。

## 3.2 功能性需求分析

### 3.2.1 系统设置需求分析

JA SPC系统设置包括用户组设定、用户设定、更改密码、车间设定、区域设定、原因/处理方案设定、检测项目设定、邮件组设定、公告设定、系统登录信息。

系统设置管理功能用例图如图3.1所示：

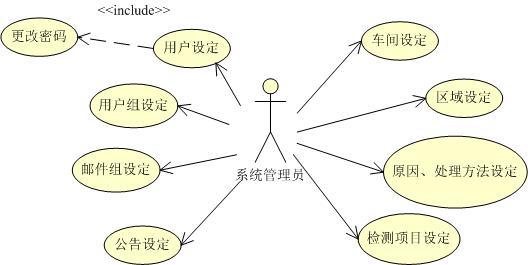


图3.1 系统管理员用例图

Fig. 3.1 System administrators use case diagram

### 3.2.2 SPC计划及规则设定需求分析

（1） SPC设置（SPC Setup）

工程师要设定的SPC管制计划，晶澳内使用到的SPC Rule等都可以通过“SPC Setup”提供的功能来做。

（2） Job Setup

Job是JA SPC里的一个概念，一个Job需要指定监控哪个Control Item，用哪种Control Chart，并指定Spec Limit和Control Limit, 使用哪些SPC Rule，此外还有一些接收数据的条件。当数据收集时，如果一笔数据符合此Job的条件，这笔数据就进入此Job,就会按Job定义中的Control Chart和SPC Rule来calculate和check rule, 如果违反Rule，就会产生Alarm，并触发Action。Job是JA SPC中最重要，也是最核心的概念。Job Setup提供的功能有：new job（新建job）,edit job(修改job), delete job(删除job),copy job(复制job),move job(把job从一个Area移到另一个Area中)，另外还有View job(用于查看Job setting), View History(查看Job修改历史记录)。具体功能包括：Job的新建、修改、删除、编辑、移动、历史记录、导出、查询等；Job状态设置；Job监控条件的设置；SPC控制状态设置；管制项目设置；管制项目的公式编辑（支持加、减、乘、除、括号、SUM、AVG、SIG、MAX、MIN、LOG等运算符号）；管制图设置；管制图表细项的设置（规格、控制、警告、Outlier界限设置）；Job所使用的SPC Rule设定以及违反规则后的Action，可细到每一个Rule单独设置Action。

（3） NonRT Setup

用于设定“非产品数据”的采集计划（或叫做Offline），包括设备相关的数据和环境数据。支持定义“自定义属性”。具体功能包括：定义NonRT的层级关系，定义各层级节点的属性。

本模块定义了数据录入方式（手工录入、文件导入、RS232获取数据）；定义数据采集的类型：计数型、计量型；定义文件导入时的Parser；定义数据的样本编号与子组数。

（4） Rule Setup

JA SPC中不拘泥于Western Electric Rules 和 Nelson Rules，系统中的Rule都由用户自定义，可以指定Rule的ID，Name等，而且包括在标准Rule基础上的一些变种，都可以通过Rule Setup灵活的定义。具体功能包括： SPC Rule的新增、修改、删除；SPC Rule子规则的设置。

用例图如图3.2所示：

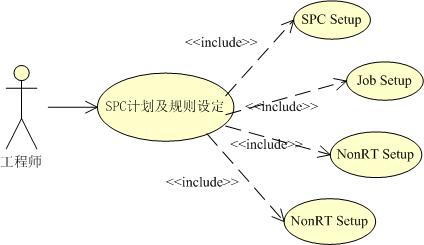


图3.2 工程师用户用例图

Fig. 3.2 Engineer use case diagram

### 3.2.3 数据收集需求分析

（1） RT Service

RTService是JA SPC提供的一个标准的Web Service，是产品相关数据进入SPC的入口。

RTService主要负责替MES做OOS/OOC检查。RT是Real Time的缩写，因为MES需要等待RTService的检查结果来决定是否要hold lot or hold equipment,所以此RT是针对MES来说的。

其处理流程如下：

MYCIM EDC模块将某个站点（分为Process 和Measurement两种类型）上收集的信息（包括lot, Process, Equipment, Measurement Data等）传送给RTService;RTService处理此笔数据再根据设定条件（预先在SPC系统中设定，见job setup）决定是否接收此笔数据，如果不符合条件，则返回MES特定信息；如果接收，接着进行OOS/OOC的检查（OOS指检查raw data有没有Out Of Spec limit; OOC是检查资料有没有违反SPC Rule）;将有无违反OOS/OOC，是否要hold lot等信息返回给MES；MYCIM EDC模块接收RTService返回的信息，决定下一步的动作。

（2） NonRT DataCollection

按照NonRT Setup的设定，QC/OP利用NonRT Data Collection收集设备相关数据，系统支持多种数据数据采集方式：手工录入，文件导入。

详细用例图如图3.3所示：

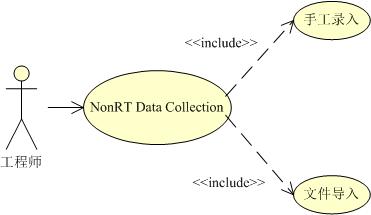


图3.3 工程师用户用例图

Fig. 3.3 Engineer use case diagram

### 3.2.4 报表需求分析

主要包括CPK报表、在线CPK报表查询、离线CPK报表查询。在Job Setup / Alarm Monitor等处，除了常见的查询条件外，JA SPC还提供了Advanced Query功能，可以指定较复杂的查询条件及其组合来查询数据。

### 3.2.5 SPC分析需求分析

（1） SPC 控制图分析（Control Chart）

控制图是美国贝尔通信研究所得休哈特(Walter Stewhart)博士于1924年首先推出的, 是用来监视、控制质量特性值随时间推移而发生变异的图表[12]。用控制图的方式展示数据是常见的分析方法，JA SPC提供了丰富的查询，筛选条件。ChartPoint显示为不同的颜色和形状代表了不同的状态：蓝色：正常点(Normal)、红色: 异常点（Alarm）、绿色：已处理过的异常点（Fixed）、灰色：违反Rule，但并不要求产生Alarm。三角形（圆点为通常的形状）：此点上加有评论。

在Chart和ChartPoint上可进行20余种操作，比如：

Switch Chart Type: 同样的数据可在Control Chart、Histogram、Normal Prob等之间切换显示。

Ad.Recalculate: 可以对历史数据，重新指定Control Chart Type,ControlLimit，Rule等以达到重新计算和检查Rule的目的。

Switch X axis label: 变更X坐标的标签, 可以选择Index、批号、Customer Lot、量测日期/时间、客户、Package、Device、P1EQ、操作员等等；

（2） 控制线重新计算

JA SPC提供定时自动重新计算控制功能（系统默认为每3个月重新计算），用户能够预先选择需要重新计算控制线的监控项目，系统定时自动计算出控制上下线等信息，并且可以将计算结果方便的应用到实际监控中。具体功能包括：重新计算设定、设置数据过滤器、重新计算结果查询、计算结果应用。如图3.4所示：

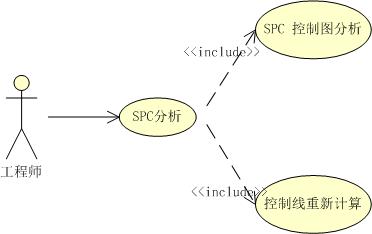


图3.4 工程师用户用例图

Fig. 3.4 Engineer use case diagram

### 3.2.6 OCAP系统需求分析

OCAP系统主要包括OCAP设定和Ocap报表两个功能模块。用例图如图3.5所示：

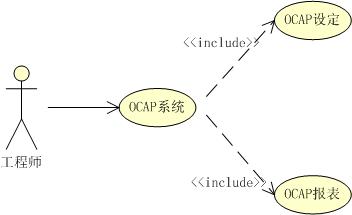


图3.5 工程师用户用例图

Fig. 3.5 Engineer use case diagram

### 3.2.7 警报监视需求分析

当有异常发生时，JA SPC系统中会产生一个Alarm，警报监视主要提供了两类功能:

一类功能用于监控厂内发生的异常情况，通过它可以了解到过去的一周中每天在每个Station发生了多少Alarm,有多少已经处理过了，有多少还没处理。每天的晨会时这些信息都一目了然,并且还可以导出到Excel，方便进一步使用。

第二类功能用于记录处理异常情况时记录的信息，比如异常的原因，处理措施等。

Alarm的产生可触发相关的Mail通知发送，用户可以通过Mail查看警报的详细内容，并在Mail中通过链接直接看到相关的控制图表。

## 3.3 非功能性需求

所谓非功能性需求，是指软件产品为满足用户业务需求而必须具有且除功能需求以外的特性[13]。非功能性需求，主要是为辅助功能性需求而出现的，有时候非功能性需求作用甚至超过了功能性需求，两者相辅相成。

（1） 系统的性能

公司各部门人员同时使用系统，是否出现网络缓慢、延迟或者打不开了应用界面等情况。本系统要求，用户在使用系统的时候，不应该出现页面打不开，打开速度缓慢，延时过长，这方面需要从程序算法、数据库设计、服务器配置、服务器带宽四个方面进行考虑和优化。

（2） 系统操作的方便性

在系统操作时，系统要考虑到界面、用户的使用习惯的角度进行设计，讲求人性化，同时需求配置完整的系统使用说明文档，说明文档包括使用步骤说明，界面截图和使用时的注意事项。

（3） 系统的安全性

使用B/S架构时，尤其重要的是考虑到系统的安全性，B/S架构本身是服务器/浏览器的架构模式，安全性相当不稳定，这时需要从程序上进行防SQL注入的设定，所谓SQL注入攻击，就是利用SQL注入技术来实施的网络攻击[14]。同时为加固网络安全在网络上配置一款硬件防火墙。

（4） 系统的可管理性

系统的可管理性是指系统必须能被配置、部署、监控和优化以确保其在我们的生产环境中工作良好。主要包括联机帮助、用户登陆管理、软件版本管理、系统公告等内容可被管理。联机帮助主要提供给用户的在线操作手册，知道用户在想使用本系统。用户登录管理主要提供实时显示在线用户的情况，实时了解系统使用用户的情况。软件版本管理有助于软件系统的升级，版本的优化。系统公告主要是实现在线部门的信息共享和传递。

# 4 系统设计

## 4.1 系统功能模块总体设计

在对系统需求分析的基础上，JA SPC 系统包括系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视7个功能模块，如图4.1所示。

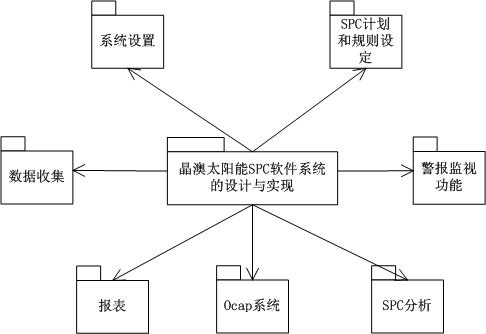


图4.1 系统功能结构图

Fig. 4.1 System function structure chart

## 4.2 系统功能模块详细设计

### 4.2.1 系统设置管理设计

系统设置包括用户组设定、用户设定、更改密码、车间设定、区域设定、原因/处理方案设定、检测项目设定、邮件组设定、公告设定、系统登录信息功能。

（1） 用户组设定、用户设定都是对系统使用人员进行分配相关操作功能与使用的基本功能，都包括增加、修改、查询、删除功能。

（2） 密码修改主要用于修改登录用户的密码，修改密码包括原密码、新密码、重复新密码3项，输入原口令的目的是为了防止其他工作人员使用其它人员的电脑，进行密码修改，本系统登录采用了Cookie进行数据存储，登录成功后，默认有效登录时间为24小时。

（3） 车间设定和区域设定是系统针对不同车间不同区域进行的设定，每个车间对应不同的操作区域。

（4） 原因和处理方案设定是针对起因代码列表和行动代码列表进行设置。包括代码和代码描述。

（5） 检测项目设定主要是设置检测项目已经项目的描述。

（6） 邮件组设定是系统对邮件通知群组的设定，每个群组可包括若干的邮件地址。

（7） 公告的设定是系统首页公告的内容设定，包括公告的起始结束时间，公告内容等。

（8）系统登录信息是系统记录和展示目前登陆的用户信息。

### 4.2.2 SPC计划及规则设定设计

（1） SPC设置设计

工程师要设定的SPC管制计划，晶澳内使用到的SPC Rule等都可以通过“SPC Setup”提供的功能来做。

（2） Job Setup设计

Job是JA SPC里的一个概念，一个Job需要指定监控哪个Control Item，用哪种Control Chart，并指定Spec Limit和Control Limit, 使用哪些SPC Rule，此外还有一些接收数据的条件。当数据收集时，如果一笔数据符合此Job的条件，这笔数据就进入此Job,就会按Job定义中的Control Chart和SPC Rule来calculate和check rule, 如果违反Rule，就会产生Alarm，并触发Action。Job是JA SPC中最重要，也是最核心的概念。Job Setup提供的功能有：new job（新建job）,edit job(修改job), delete job(删除job),copy job(复制job),move job(把job从一个Area移到另一个Area中)，另外还有View job(用于查看Job setting), View History(查看Job修改历史记录)。具体功能包括：Job的新建、修改、删除、编辑、移动、历史记录、导出、查询等；Job状态设置；Job监控条件的设置；SPC控制状态设置；管制项目设置；管制项目的公式编辑（支持加、减、乘、除、括号、SUM、AVG、SIG、MAX、MIN、LOG等运算符号）；管制图设置；管制图表细项的设置（规格、控制、警告、Outlier界限设置）；Job所使用的SPC Rule设定以及违反规则后的动作，可细到每一个Rule单独设置动作。

（3） NonRT Setup设计

用于设定“非产品数据”的采集计划（或叫做Offline），包括设备相关的数据和环境数据。支持定义“自定义属性”。具体功能包括：定义NonRT的层级关系；定义各层级节点的属性；定义数据录入方式（手工录入、文件导入、RS232获取数据）；定义数据采集的类型：计数型、计量型；定义文件导入时的Parser；定义数据的样本编号与子组数。

（4） Rule Setup设计

JA SPC中不拘泥于Western Electric Rules 和 Nelson Rules，系统中的Rule都由用户自定义，可以指定Rule的ID，Name等，而且包括在标准Rule基础上的一些变种，都可以通过Rule Setup灵活的定义。

Rule Setup设计具体功能包括：SPC Rule的新增、修改、删除；SPC Rule子规则的设置。

### 4.2.3 数据收集设计

（1） RTService设计

RTService是产品相关数据进入SPC的入口，主要负责替MES做OOS/OOC检查；RT是Real Time的缩写，因为MES需要等待RTService的检查结果来决定是否要hold lot or hold equipment,所以此RT是针对MES来说的。

其处理流程如下： MYCIM EDC模块将某个站点（分为Process 和Measurement两种类型）上收集的信息传送给RTService，RTService处理此笔数据再根据设定条件（预先在SPC系统中设定，见job setup）决定是否接收此笔数据，如果不符合条件，则返回MES特定信息；如果接收，接着进行OOS/OOC的检查;将有无违反OOS/OOC，是否要hold lot等信息返回给MES；MYCIM EDC模块接收RTService返回的信息，决定下一步的动作。

（2） NonRT DataCollection设计

按照NonRT Setup的设定，QC/OP利用NonRT Data Collection收集设备相关数据，系统支持多种数据数据采集方式：手工录入、文件导入。

### 4.2.4 报表设计

报表模块主要包括CPK报表、在线CPK报表查询、离线CPK报表查询。主要设计包括：

（1） CPK报表查询提供条件有报表名称，车间代码，Job Type,Status。报表的设定包括新增、修改和删除，报表设定可选择内容包括报表名称、车间代码、报表类型、报表日期、状态、机台、Job Type、管控计划代码。

（2） 在线CPK报表查询，提供在线的数据的CPK报表的查询，包括报表名称，车间代码，Job Type,Status。生成和汇出在线的CPK报表。

（3） 离线CPK报表查询，提供离线的数据的CPK报表的查询，包括报表名称，车间代码，Job Type,Status。生成和汇出离线的CPK报表。

另外在Job Setup 、Alarm Monitor等处，除了常见的查询条件外，JA SPC还提供了Advanced Query功能，可以指定较复杂的查询条件及其组合来查询数据，为用户提供更多各种数据需求的报表。

### 4.2.5 SPC分析设计

（1） SPC 控制图分析设计

用控制图的方式展示数据是常见的分析方法，JA SPC提供了丰富的查询，筛选条件。ChartPoint显示为不同的颜色和形状代表了不同的状态：蓝色：正常点(Normal)、红色: 异常点（Alarm）、绿色：已处理过的异常点（Fixed）、灰色：违反Rule，但并不要求产生Alarm、三角形（圆点为通常的形状）：此点上加有评论。

（2） 控制线重新计算设计

JA SPC提供定时自动重新计算控制功能（系统默认为每3个月重新计算），用户能够预先选择需要重新计算控制线的监控项目，系统定时自动计算出控制上下线等信息，并且可以将计算结果方便的应用到实际监控中。具体功能包括：重新计算设定、设置数据过滤器、重新计算结果查询、计算结果应用。

（3） 其它数据导入设计

系统还提供一些工具，比如Offline Import功能可以将Excel中的数据用控制图的方式显示。在导入数据前需要预先设定导入数据的管制项目、规格上下限、控制上下线、使用的图表类型、文件的存放路径等等。

### 4.2.6 OCAP系统设计

OCAP系统主要包括OCAP设定和Ocap报表。其中OCAP的设定包括处理、 历史纪录、汇出、生成、删除。OCAP报表提供的查询条件包括Ocap状态、Ocap代码、Ocap单号、流程单号、Ocap类型、CreateTime。

OCAP系统将用户目前OCAP单类型、内容、处理流程，实现系统电子化管理。并在各处理环节中，以电子邮件方式提醒各相关处理责任单位做相应处理。

OCAP由SPC RTService Check OOS/OOC后自动实时开出。对用户来讲，几乎没有间隔时间，就可以收到OCAP Alarm邮件，和在“实时处理状态看板UI中”看到该OCAP单和状态。

详细流程图如图4.2所示：

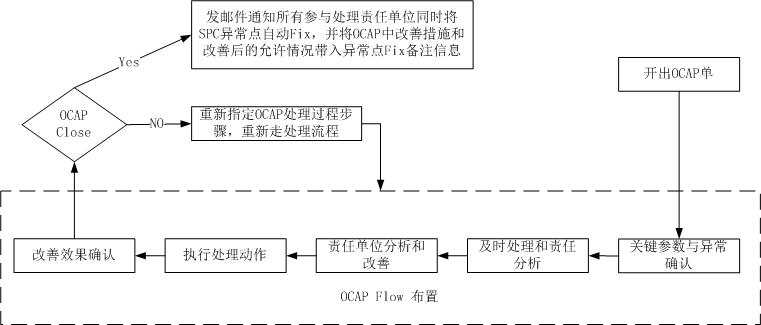
 图4.2 Ocap管理流程图

Fig. 4.2 Ocap management flow chart

### 4.2.7 警报监控设计

当有异常发生时，JA SPC系统中会产生一个Alarm，警报监控主要提供了两类功能:

一类功能用于监控厂内发生的异常情况，通过它可以了解到过去的一周中每天在每个Station发生了多少Alarm,有多少已经处理过了，有多少还没处理。每天的晨会时这些信息都一目了然,并且还可以导出到Excel，方便进一步使用。

第二类功能用于记录处理异常情况时记录的信息，比如异常的原因，处理措施等。

Alarm的产生可触发相关的Mail通知发送，用户可以通过Mail查看警报的详细内容，并在Mail中通过链接直接看到相关的控制字符。

## 4.3 数据库设计

### 4.3.1 数据库设计原则

数据库设计在整个设计中处于非常关键和重要的位置，若在系统搭建完成后，再进行数据库表结构更改，不但会影响到数据库设计性能，也能引发系统的稳定性。若系统正常投入使用时，发现因为客户、产品等使用的不断增长，而造成数据因为量大，而造成系统运行缓慢，从而造成系统二次开发，即浪费公司的人力、物力，又加大了新开发系统的成本。所以在数据库设计时要考虑以下原则[16]：

（1） 数据库设计时要考虑到数据量的大小，如果数据量大的情况下，要考虑到索引的设置。

（2） 设计数据库时要考虑到数据使用频率，使用频率高的数据放在同一表中，使用频率低的数据放在另一个表中。

（3） 设计数据库时要考虑尽量避免两表或两表以上的数据查询。

（4） 设计数据库时要考虑到数据库表的可扩展性与数据挖掘。

### 4.3.2 数据库概念设计实现

数据库概念设计实现主要通过E-R图来描述，E-R图主要包括了实体、属性、联系。数据库设计实现主要用E-R图来显示数据之间的关系，E-R图直接提供了实体型、属性与联系的方法。

用户管理E-R图，用户组信息（用户组ID、用户组描述、更新人、更新时间、创建人、创建时间），用户信息（用户ID、密码、邮件、部门、名称、状态、更新人、更新时间、创建人、创建时间、车间、用户描述、密码提醒），权限（权限ID、权限名称、权限备注）。

用户管理E-R图如图4.3所示：

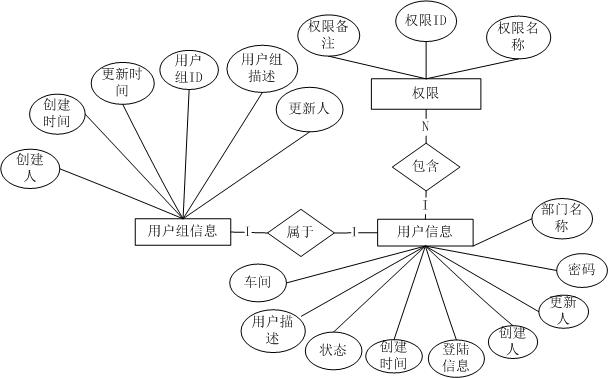


图4.3 用户管理E-R图

Fig. 4.3 User management chart E-R diagram

### 4.3.3 数据库逻辑设计实现

（1） 用户表(spc\_user)

用户信息表主要用于存储用户登录使用的账户信息，主要包括用户ID、密码、邮件、部门、名称、状态、更新人、更新时间、创建人、创建时间、车间、用户描述、密码提醒、是否登录其中ID为主键。如表4.1所示：

表4.1 账户信息表

Tab. 4.1 Account information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| USER\_ID | VARCHAR2(16) | 否 | ID |
| PASSWORD | VARCHAR2(16) | 是 | 密码 |
| EMAIL | VARCHAR2(64) | 是 | 邮件 |
| DEPARTMENT | VARCHAR2(64) | 是 | 部门 |
| USER\_NAME | VARCHAR2(32) | 是 | 名称 |
| STATUS | VARCHAR2(16) | 是 | 状态 |
| UPDATE\_BY | VARCHAR2(16) | 是 | 更新人 |
| UPDATE\_TIME | DATE | 是 | 更新时间 |
| CREATE\_BY | VARCHAR2(16) | 是 | 创建人 |
| CREATE\_TIME | DATE | 是 | 创建时间 |
| DEFAULT\_FAB | VARCHAR2(16) | 是 | 车间 |
| USER\_DESC | VARCHAR2(4000) | 是 | 用户描述 |
| PWD\_UPDATE\_TIME | DATE | 是 | 密码提醒 |
| LOGIN\_FLAG | VARCHAR2(1) | 是 | 是否登录 |
| STATUS | VARCHAR2(16) | 是 | 状态 |

（2） 用户组信息表(spc\_usergroup)

用户信息表主要包括：用户组ID、用户组描述、更新人、更新时间、创建人、创建时间，如表4.2所示：

表4.2 用户组信息表

Tab. 4.2 Usergroup information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| USERGROUP\_ID | VARCHAR2(16) | N | 用户组ID |
| USERGROUP\_DESC | VARCHAR2(256) | Y | 用户组描述 |
| UPDATE\_BY | VARCHAR2(16) | Y | 更新人 |
| UPDATE\_TIME | DATE | Y | 更新时间 |
| CREATE\_BY | VARCHAR2(16) | Y | 创建人 |
| CREATE\_TIME | DATE | Y | 创建时间 |

（3） 用户组用户信息表(spc\_usergroup\_user)

用户组用户信息表主要包括用户ID和用户组ID，如表4.3所示：

表4.3 用户组用户信息表

Tab. 4.3 Usergroup\_user information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| USERGROUP\_ID | VARCHAR2(16) | N | 用户组ID |
| USER\_ID | VARCHAR2(16) | N | 用户ID |

（4） Ocap单表(Ocap表)

Ocap单表主要包括：OCAPID、OCAP类型、对应JOB、OCAP单状态、更新时间、创建人、创建时间等信息，如表4.4所示：

表4.4 OCAP表

Tab. 4.4 OCAP table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| OCAP\_ID | VARCHAR2(32) | N | OCAPID |
| OCAP\_NO | VARCHAR2(32) | Y | OCAPNO |
| OCAP\_TYPE | VARCHAR2(8) | Y | OCAP类型 |
| JOB\_ID | VARCHAR2(512) | Y | 对应JOB |
| FAB\_ID | VARCHAR2(16) | Y | 车间 |
| EQPT\_ID | VARCHAR2(32) | Y | 设备ID |
| MATERIAL\_LOT | VARCHAR2(64) | Y | 工单 |
| LOT\_ID | VARCHAR2(32) | Y | 批次号 |
| CURRENT\_STEP | NUMBER | Y | 当前步骤 |
| NEXT\_STEP | NUMBER | Y | 下一步骤 |
| CREATE\_TIME | DATE | Y | 创建时间 |
| UPDATE\_BY | VARCHAR2(32) | Y | 创建人 |
| OCAP\_STATUS | VARCHAR2(16) | Y | OCAP单状态 |
| PRODUCT | VARCHAR2(64) | Y | 产品 |
| RAW\_DATA | CLOB | Y | 原始数据 |
| RETURN\_MSG | VARCHAR2(2000) | Y | 返回信息 |
| JOB\_TYPE | VARCHAR2(32) | Y | Job类型 |
| PASTE | VARCHAR2(32) | Y | PASTE |
| TANK | VARCHAR2(64) | Y | TANK |
| SEND\_MAIL\_FLAG | VARCHAR2(1) | Y | 是否发送mail |

（5）SPC Job表

SPC Job信息表主要包括车间ID、JobID、Job版本、Job名称、Job类型、Job状态、控制状态、区域ID、产品列表等。

SPC Job信息表如表4.5所示：

表4.5 SPC Job信息表

Tab. 4.5 Job information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| FAB\_ID | VARCHAR2(32) | N | 车间ID |
| JOB\_ID | VARCHAR2(128) | N | JobID |
| JOB\_VER | NUMBER(3) | N | Job版本 |
| JOB\_NAME | VARCHAR2(128) | N | Job名称 |
| JOB\_TYPE | VARCHAR2(16) | N | Job类型 |
| JOB\_STATUS | VARCHAR2(16) | N | Job状态 |
| CONTROL\_STATUS | VARCHAR2(16) | N | 控制状态 |
| AREA\_ID | VARCHAR2(32) | N | 区域ID |
| PROD\_LIST\_1 | VARCHAR2(256) | Y | 产品列表1 |
| M1\_OPERATION | VARCHAR2(32) | Y | 工步 |
| M1\_EQ | VARCHAR2(512) | Y | 设备 |
| M1\_MFG\_AREA | VARCHAR2(8) | Y | 设备区域 |
| LOT\_TYPE | VARCHAR2(64) | Y | 批次 |
| EDC\_LEVEL | VARCHAR2(16) | Y | 收集级别 |

（6）SPC\_Alarm报警表(spc\_Alarm)

SPC\_Alarm报警表包括报警ID、控制状态、对应job、时间等信息如表4.6所示：

表4.6 报警信息表

Tab. 4.6 Alarm information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| ALARM\_ID | VARCHAR2(128) | N | 报警ID |
| CONTROL\_STATUS | VARCHAR2(16) | N | 控制状态 |
| ALARM\_TIME | DATE | N | 时间 |
| JOB\_ID | VARCHAR2(128) | N | 对应job |
| JOB\_VER | NUMBER(3) | N | job版本 |
| JOB\_NAME | VARCHAR2(128) | N | job名称 |
| P1\_EQ | VARCHAR2(64) | Y | P1\_EQ |
| CHART\_TYPE | VARCHAR2(16) | Y | 图类型 |
| ENTRY\_SEQ | NUMBER(20) | N | 顺序ID |
| LOT\_ID | VARCHAR2(64) | Y | 对应lot |
| PROD\_1 | VARCHAR2(256) | Y | 产品 |
| PROD\_2 | VARCHAR2(256) | Y | 工艺 |
| M1\_OPERATION | VARCHAR2(32) | Y | 工序类型 |
| AREA\_ID | VARCHAR2(32) | N | 所在区域 |
| FIXED\_FLG | CHAR(1) | Y | 是否解决 |
| UPDATE\_TIME | DATE | Y | 更新时间 |
| OCAP\_ID | VARCHAR2(16) | Y | 对应Ocap单 |
| VIOLATED\_INFO | VARCHAR2(4000) | Y | 报警明细 |

(7) 在线数据信息表（spc\_sample\_RT）

在线数据信息表主要包括JobID、序列号、批次、时间、控制类型、job版本、工艺、参数、产品、Rate、更新时间、接收时间、产品ID等信息。如表4.7所示：

表4.7 在线数据信息表

Tab. 4.7 RT information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| JOB\_ID | VARCHAR2(128) | N | JobID |
| P1\_EQ | VARCHAR2(64) | Y | P1\_EQ |
| ENTRY\_SEQ | NUMBER(20) | N | 序列号 |
| LOT\_ID | VARCHAR2(64) | Y | 批次 |
| P1\_REWORK\_COUNT | NUMBER(2) | Y | P1\_REWORK\_COUNT |
| M1\_REWORK\_COUNT | NUMBER(2) | Y | M1\_REWORK\_COUNT |
| M1\_TIME | DATE | N | 时间 |
| CONTROL\_STATUS | VARCHAR2(16) | N | 控制类型 |
| JOB\_VER | NUMBER(3) | N | job版本 |
| LOTS\_IN\_BATCH | VARCHAR2(512) | Y | LOTS\_IN\_BATCH |
| PROD\_1 | VARCHAR2(256) | Y | 工艺 |
| PROD\_2 | VARCHAR2(256) | Y | 参数 |
| PROD\_3 | VARCHAR2(1024) | Y | 产品 |
| PROD\_4 | VARCHAR2(256) | Y | Rate |
| UPDATE\_TIME | DATE | N | 更新时间 |
| RECEIVE\_TIME | DATE | Y | 接收时间 |
| CHART\_PNT\_TYPE | VARCHAR2(16) | Y | CHART\_PNT\_TYPE |
| PRODUCT\_ID | VARCHAR2(64) | Y | 产品ID |
| CUSTOMER | VARCHAR2(64) | Y | CUSTOMER |
| DEVICE | VARCHAR2(64) | Y | DEVICE |
| M1\_EQ | VARCHAR2(64) | Y | 设备ID |
| M1\_OPERATOR | VARCHAR2(64) | Y | 操作员 |
| ALARM\_ID | VARCHAR2(128) | Y | 报警ID |
| OCAP\_ID | VARCHAR2(16) | Y | OCAP\_ID |
| USL | NUMBER | Y | 标准上线 |
| SL | NUMBER | Y | 标准线 |
| LSL | NUMBER | Y | 标准下线 |
| DATA\_TYPE | VARCHAR2(16) | Y | 数据类型 |
| RAW\_DATA\_DETAIL | CLOB | Y | 原始数据 |
| WORDER | VARCHAR2(64) | Y | 工单 |
| MATERIAL\_LOT\_ID | VARCHAR2(64) | Y | 批次 |
| ROUTE | VARCHAR2(64) | Y | 工序 |
| POSITION\_ID | VARCHAR2(64) | Y | 位置 |
| LOCATION | VARCHAR2(64) | Y | LOCATION |
| TANK | VARCHAR2(64) | Y | TANK |

（8）离线数据信息表（spc\_sample\_NONRT）

离线数据信息表主要包括质量检测ID、检测人、检测时间、增加时间、修改时间信息。如表4.8所示：

表4.8 离线数据信息表

Tab. 4.8 NONRT information table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 可否为空 | 说明 |
| JOB\_ID | VARCHAR2(128) | N | JobID |
| ENTRY\_SEQ | NUMBER(20) | N | 序列 |
| MEASURE\_TIME | DATE | N | 时间 |
| JOB\_VER | NUMBER(3) | N | job版本 |
| RAW\_DATA | VARCHAR2(4000) | N | 原始数据 |
| UPDATE\_TIME | DATE | N | 更新时间 |
| OPERATOR\_ID | VARCHAR2(64) | Y | 操作人 |
| USL | NUMBER | Y | 标准上线 |
| SL | NUMBER | Y | 标准线 |
| LSL | NUMBER | Y | 标准下线 |
| DATA\_TYPE | VARCHAR2(16) | Y | 数据类型 |
| KEY\_VALUE\_LIST | VARCHAR2(4000) | Y | 数据列表 |
| IS\_ACCUMULATE | CHAR(1) | Y | IS\_ACCUMULATE |
| IS\_PASTDATA | CHAR(1) | Y | IS\_PASTDATA |
| CONTROL\_STATUS | VARCHAR2(16) | N | CONTROL\_STATUS |
| MONITOR\_EQ | VARCHAR2(1024) | Y | MONITOR\_EQ |
| MATERIAL\_ID | VARCHAR2(1024) | Y | MATERIAL\_ID |
| DEVIATION\_VALUE | VARCHAR2(4000) | Y | DEVIATION\_VALUE |
| LOT\_ID | VARCHAR2(64) | Y | LOT\_ID |

5 系统的实现

## 5.1 系统设定管理的实现

系统设定管理包括用户组设定、用户设定、更改密码、车间设定、区域设定、原因处理方案设定、检测项目设定、邮件组设定、公告设定、系统登录信息的功能。

JA SPC系统登录界面输入项有4项，包括UserID，Password，Fab，Language。UserID组成格式为字母、数字与下划线组成，长度为6-25位；Password由字母、数字与下划线组成，长度为6-25位；Fab是下拉框，包含公司内所有的车间。登陆界面如图5.1所示：

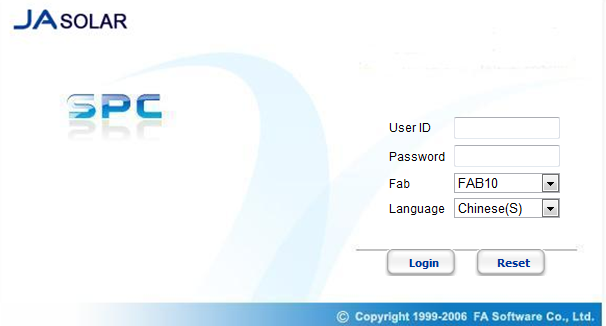


图5.1 用户登录界面

Fig 5.1 Login interface

JA SPC系统登录成功后，进入到系统的主界面，系统采用iFrame框架式的结构，基于iFrame的ASP数据处理方式只需要子窗口Form提交的数据结果，不需要维持并重新发送当前页面的数据状态，避免了Session的使用，因而性能得到了提升[15]。采用iFrame框架重载页面时不需要重载整个页面,只需要重载页面中的一个框架页(减少了数据的传输，加快了网页下载速度)。页面大都使用了自定义控件，自定义控件通常用于重复运用较多的部分，将重复运用次数较多的部分进行抽离，组成一个具有个性意义的控件。.net自定义控件优点:可以重复运用、可以进行自定义控件传参，即从父页面中获取到的参数可以传给自定义控件，减少了自定义控件再次去数据库中查找一次，用户自定义控件可以使开发人员能够根据应用程序的要求，方便的定义和编写控件，开发所使用的编程技术将与编写的Web窗体的技术相同，只要开发人员对自定义控件进行修改，就可以将使用此控件的页面的所有控件都进行修改。

系统主界面如图5.2所示：



图5.2 系统主界面

Fig. 5.2 System interface

用户登录的核心代码如下：

string userID = txtUserID.Text;

string password = txtPassword.Text;

SetFavorites();

User user = new User();

user.UserID = userID;

user=SystemServiceProxy.CRUDUser(userID,user,FAS.SPC.ServiceImp.

ObjectServiceReq.OperationTypeDefine.READ);

//Set UserID, FabID, Language

Session[SessionNames.USER\_ID] = userID;

Session[SessionNames.DEPT] = user.Department;

//Auto redirect to Login.aspx if user doesn't login(refer to web.config)

FormsAuthentication.SetAuthCookie(userID, false);

user.LastLoginTime = System.DateTime.Now;

user.LastLoginComputer=Page.Request.UserHostName==null? Page.Request.UserHostAddress : Page.Request.UserHostName;

if (UtilObjectCache.LogonUsers.Contains(Session.SessionID))

{

UtilObjectCache.LogonUsers.Remove(Session.SessionID);

}

UtilObjectCache.LogonUsers.Add(Session.SessionID, user);

……

用户组设定，用户设定，车间设定，区域设定都是对系统使用人员进行分配相关操作功能与使用的基本功能，都包括增加、修改、查询、删除功能。更改密码主要用于修改登录用户的密码，修改口令包括原口令、新口令、重新新口令3项，输入原口令的目的是为了防止其他工作人员使用其它人员的电脑，进行密码修改，本系统登录采用了Cookie进行数据存储。原因/处理方案设定是设定异常发生原因。

检测项目设定是设定SPC检查的项目。邮件组设定、公告设定、系统登录信息即设定邮件，公告，系统信息。

用户组增加的核心代码如下：

if ("New".Equals(actionType))

{

UserGroup readUserGroup = SystemServiceProxy.CRUDUserGroup(Convert.ToString(Session[SessionNames.USER\_ID]), userGroup, ObjectServiceReq.OperationTypeDefine.READ);

if (readUserGroup != null)

{

UtilJSMessage.MessageBox(this, UtilCulture.GetString("User.IsExist"));

return;

} SystemServiceProxy.CRUDUserGroup(Convert.ToString(Session[SessionNames.USER\_ID]), userGroup, ObjectServiceReq.OperationTypeDefine.CREATE);

} ……

用户组增加或修改如图5.3所示：

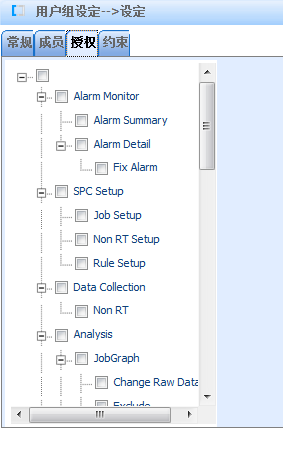


图5.3 用户组主界面

Fig. 5.3 UserGroup interface

## 5.2 SPC计划和规则设定的实现

SPC计划和规则设定包括管控计划设定、管制规定设定。管控计划设定可根据类型，区域，管控计划ID等进行查询。管制规定设定主要是设定一些生产过程中用到的规则。

管控计划查询的核心代码如下：

Query q = null;

DataSet d = QueryDA.LoadDefaultQueryIDAndNameDS(userID, fabID, moduleID);

if (d != null && d.Tables[0].Rows.Count > 0)

{

q = Query.CreateQuery(d.Tables[0].Rows[0]["MODULE\_ID"].ToString());

q.JobType = d.Tables[0].Rows[0]["JOB\_TYPE"].ToString();

q.QueryID = d.Tables[0].Rows[0]["QUERY\_ID"].ToString();

q.QueryName = d.Tables[0].Rows[0]["QUERY\_NAME"].ToString();

}

return q; ……

查询界面如图5.4所示：

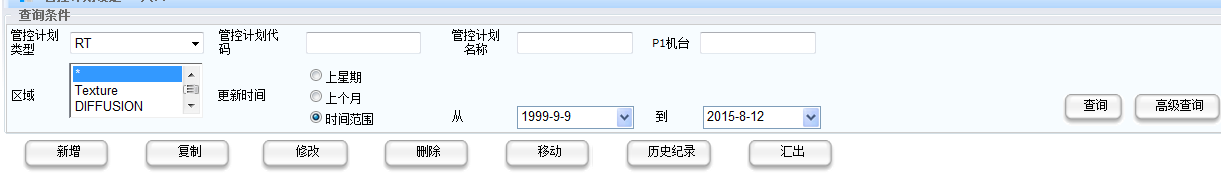


图5.4 查询界面

Fig. 5.4 Query interface

可通过管控计划类型，管控计划代码，名称，P1机台，区域，更新时间段联合查询。查询列表如图5.5所示：



图5.5 查询列表界面

Fig. 5.5 Query list interface

## 5.3 数据收集的实现

RTService数据收集是产品相关数据进入SPC的入口，主要负责替MES做OOS/OOC检查；因为MES需要等待RTService的检查结果来决定是否要hold lot or hold equipment,所以此RT是针对MES来说的。

详细流程图见图5.6所示：

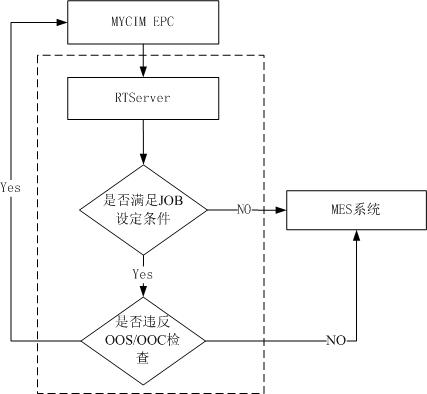


图5.6 实时数据收集流程图

Fig. 5.6 RT data collection flow chart

离线数据采集主要是通过手工采集离线信息。显示可以采集的类型的部分核心代码如下：

DataTable dt = SPCSetupServiceProxy.GetNonRTChildNodeDS(objectID, Convert.ToString(Session[SessionNames.FAB\_ID])).Tables["SPC\_NONRT\_OBJECT"];

string namePath;

string[] path = DataCollectionServiceProxy.GetNonRTObjectPath(objectID, out namePath);

string displayPath = "";

if (!String.IsNullOrEmpty(namePath))

{

string[] name = namePath.Split(new char[] { '\\' }, StringSplitOptions.RemoveEmptyEntries);

for (int i = 0; i < path.Length; i++)

{

if (i < name.Length)

{

name[i] = "<a href=\"NonRTObjectList.aspx?objectid=" + path[i] + "\">" + name[i] + "</a>";

}

}

displayPath = string.Join("=>", name);

}

lblObjectList.InnerHtml = displayPath;

if (objectID == long.Parse(Session[SessionNames.FAB\_OBJECT\_ID].ToString()))

{

btnBack.Visible = false;

} ……

可采集数据列表如图5.7所示：

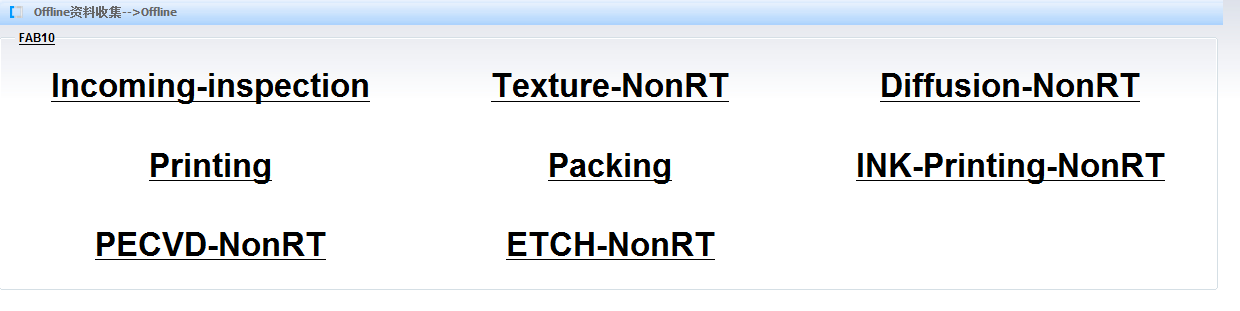


图5.7 实时采集数据列表界面

Fig. 5.7 RT DataCollection list interface

具体采集部分核心代码如下：

NonRTDataCollection nonRTDC = new NonRTDataCollection();

NonRTDC = SaveDC(nonRTDC);

TextBox tb = (TextBox)Page.FindControl("frmNonRTVDC").FindControl("tblNonRTDC").FindControl("txtFilePath");

List<string> sChartType = new List<string>();

List<string> sVoliatedRule = new List<string>();

bool oosFlag = false;

……

## 5.4 报表的实现

报表模块中包含CPK报表、在线CPK报表查询和离线CPK报表查询。CPK报表就是设定制程能力综合指标报表。

CPK报表流程图见图5.8所示：

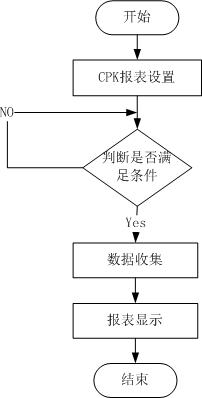


图5.8 CPK报表流程图

Fig. 5.8 CPK report flow chart

报表设定的部分核心代码如下：

CPKReport report = BuildModel();

string userID = (string)Session[Util.SessionNames.USER\_ID];

if ("New".Equals(actionType))

{

if (ReportServiceProxy.HasObject(report))

{

tilJSMessage.MessageBox(this, UtilCulture.GetString("CPKReport.ReportName") + UtilCulture.GetString("E44064") + "[" + report.ReportName + "]");

return;

}

ReportServiceProxy.CRUDCPKReport(userID, report, FAS.SPC.ServiceImp.ObjectServiceReq.OperationTypeDefine.CREATE);

}

else if ("Edit".Equals(actionType))

{

ReportServiceProxy.CRUDCPKReport(userID, report, FAS.SPC.ServiceImp.ObjectServiceReq.OperationTypeDefine.UPDATE);

}

UtilJSMessage.MessageBox(this, UtilCulture.GetString("E44061"));

CPKReportController.ReportEntryShow();……

报表设定的界面如图5.9所示：



图5.9 报表设定界面

Fig. 5.9 Report set interface

CPK报表查询部分核心代码：

集合查询条件：

List<ICriterion> list = new List<ICriterion>();

if (!String.IsNullOrEmpty(ddlReportName.Text))

{

list.Add(Expression.Like("ReportName", ddlReportName.Text.Trim(), MatchMode.Anywhere));

}

if (!String.IsNullOrEmpty(ddlFabID.Text))

{

list.Add(Expression.Like("FabID", ddlFabID.Text.Trim(), MatchMode.Anywhere));

}

if (!String.IsNullOrEmpty(ddlJobType.Text))

{

list.Add(Expression.Like("JobType", ddlJobType.Text.Trim(), MatchMode.Anywhere));

}

if (!String.IsNullOrEmpty(ddlStatus.Text))

{

list.Add(Expression.Like("Status", ddlStatus.Text.Trim(), MatchMode.Anywhere));

}

criterions = list.ToArray();

Session[SessionNames.QUERY\_CRITERION] = criterions;

查询后显示代码:

criterions = (ICriterion[])Session[SessionNames.QUERY\_CRITERION];

orders = (Order[])Session[SessionNames.QUERY\_ORDER];

count = ReportServiceProxy.GetCPKReports(criterions, orders, 0, 0).Length;

int pageSize = Constants.PageSize;

int pageCount = (count / pageSize) + ((count % pageSize == 0) ? 0 : 1);

PageIndex = (pageCount < PageIndex) ? 1 : PageIndex;

DataTable dt = null;

查询页面如图5.10所示：

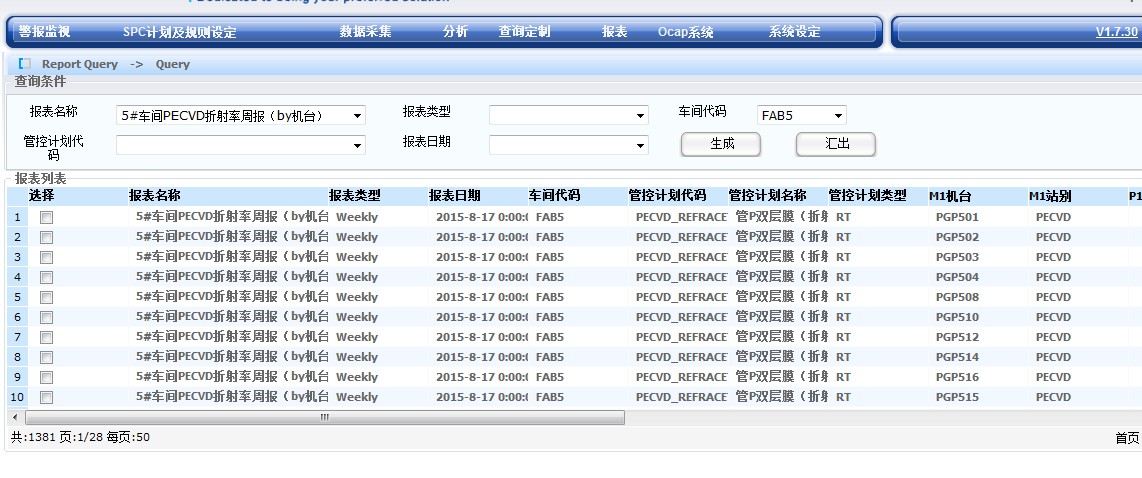


图5.10 查询界面

Fig. 5.10 Query interface

## 5.5 SPC分析的实现

SPC分析包括：管制计划图、重新计算设定和重新计算结果。管制计划图可查看详细的数据信息。SPC分析主界面如图5.11所示。

重新设定管控计划筛选条件，部分核心代码如下:

recSetupQuery = new RecSetupQuery();

recSetupQuery.Fab = (string)Session[Util.SessionNames.FAB\_ID];

recSetupQuery.JobType = cmbJobType.SelectedValue.ToString();

List<string> list = new List<string>();

if (cmbJobType.SelectedValue.Equals(ObjectTypes.JobTypes.RT.ToString()))

{

recSetupQuery.Area = hidAreaRT.Value;

}……

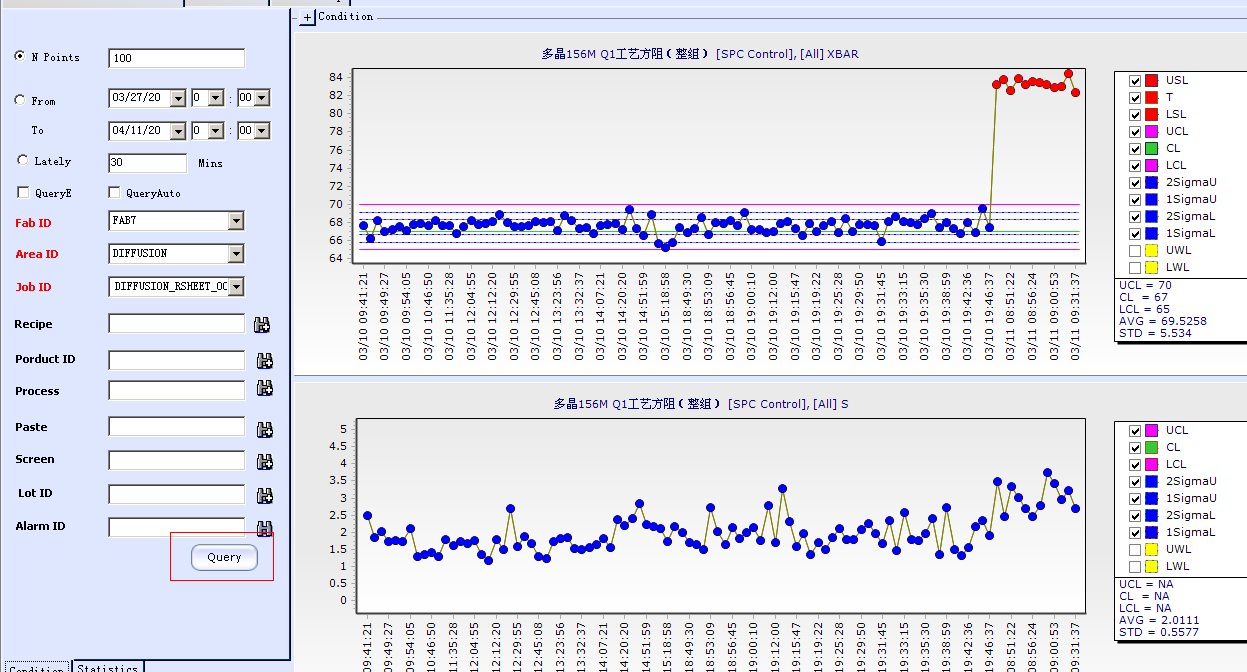


图5.11 SPC分析主界面

Fig. 5.11 SPC analysis of the main interface

重新设定管控计划筛选条件，部分核心代码如下:

recSetupQuery = new RecSetupQuery();

recSetupQuery.Fab = (string)Session[Util.SessionNames.FAB\_ID];

recSetupQuery.JobType = cmbJobType.SelectedValue.ToString();

List<string> list = new List<string>();

if (cmbJobType.SelectedValue.Equals(ObjectTypes.JobTypes.RT.ToString()))

{

recSetupQuery.Area = hidAreaRT.Value;

}……

重新设定管控计划页面如图5.12所示：



图5.12 重新设定管控计划界面

Fig. 5.12 Reset the control plan interface

重新计算结果即重新计算修改筛选条件后的结果，代码如下

//先筛选

recResultsQuery = new RecResultsQuery();

recResultsQuery.Fab = (string)Session[Util.SessionNames.FAB\_ID];

recResultsQuery.JobType = cmbJobType.SelectedValue.ToString();

List<string> list = new List<string>();

if (cmbJobType.SelectedValue.Equals(ObjectTypes.JobTypes.RT.ToString()))

{

recResultsQuery.Area = hidAreaRT.Value;

}

……

//后应用

操作界面如图5.13所示：

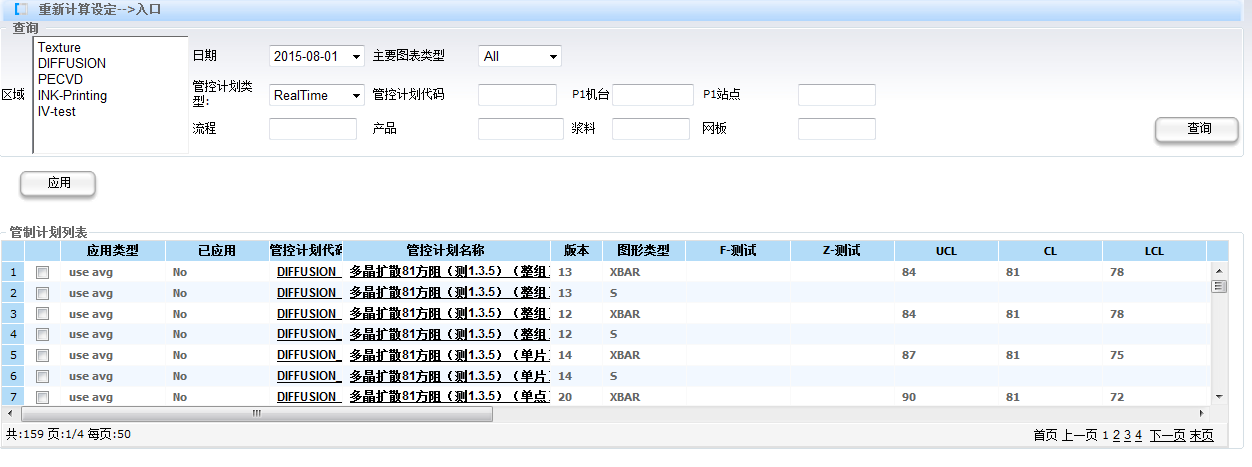


图5.13 重新计算结果界面

Fig. 5.13 Recalculated results interface

## 5.6 Ocap系统的实现

Ocap系统包含Ocap设定和Ocap报表。Ocap设定指异常发生时采取措施的计划设定。Ocap报表指的是异常发生时采取措施的计划查询。

查询的主要代码：

OcapQuery OcapQuery = new OcapQuery();

OcapQuery.FabId = (string)Session[Util.SessionNames.FAB\_ID];

if (cmbOCapStatus.SelectedIndex == 0)

{

OcapQuery.OcapStatus = Model.OcapStatus.Wait;

}

else

{

if (cmbOCapStatus.SelectedIndex == 1)

{

OcapQuery.OcapStatus = Model.OcapStatus.Close;

}

else

{

OcapQuery.OcapStatus = Model.OcapStatus.Delete;

}

}……

OCAP查询界面如图5.14所示：

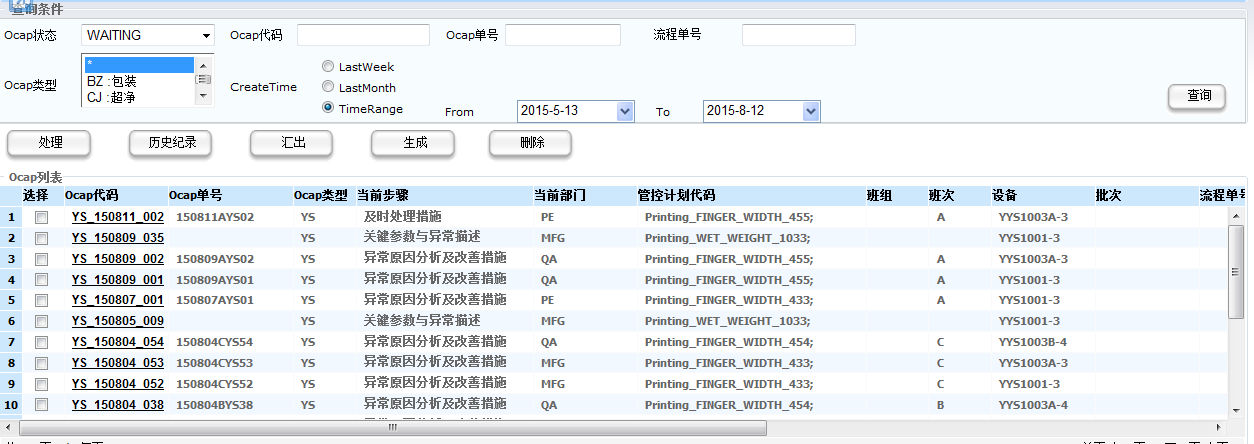


图5.14 OCAP查询界面

Fig. 5.14 OCAP Query interface

生成到Excel代码：

string tempFileName = "";

string saveFileName = "";

string ocapID = (string)UtilWebGrid.GetRowValue(row, "OcapID");

Excel.Application objApp = null;

Excel.Workbooks objBooks = null;

Excel.Workbook objBook = null;

Excel.Sheets objSheets = null;

Excel.Worksheet objSheet = null;

//准备数据

FAS.SPC.Model.SPCSetup.Ocap ocap = Ocap.Find(ocapID);

OcapBaseInfo ocapBaseInfo = OcapBaseInfo.Find(ocapID);

//生成报表

if (ocap != null)

{

//写数据

tempFileName = ConfigurationSettings.AppSettings["OcapReportTemplatePath"] + ocap.OcapType + ".xls";

saveFileName = ConfigurationSettings.AppSettings["OcapReportPath"] + ocap.OcapType + DateTime.Now.ToString("yyyy-MM-dd") + "\_" + (new Random()).Next() + ".xls";

object MissingValue = Type.Missing;

if (!File.Exists(tempFileName))

{

//throw new FASException(UtilCulture.GetString("E34096"));

}

objApp = new Excel.Application();

objBooks = objApp.Workbooks;

objBook = objBooks.Open(tempFileName, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value, Missing.Value);

objSheets = objBook.Worksheets;

objSheet = (Excel.Worksheet)objSheets[1];

OcapGenerate ocapGenerate = new OcapGenerate();

ocapGenerate.OcapWriteSheet(ocap, objSheet);

objBook.SaveAs(saveFileName, Missing.Value, Missing.Va

} ……

生成的Excel如图5.15所示：

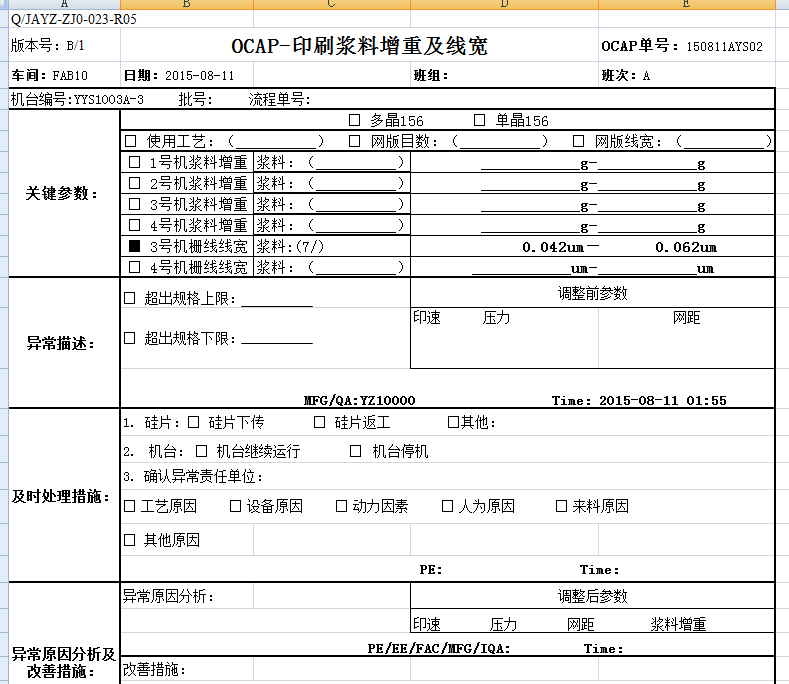


图5.15 生成EXCEL界面

Fig. 5.15 To excel interface

## 5.7 警报监视的实现

警报监视包括警报汇总和警报明细。可查看每个区域报警的详细信息。警报监视汇总查询核心代码如下：

DateTime dateFrom = DateTime.Parse(sdateFrom);

DateTime dateTo = DateTime.Parse(sdateTo);

TimeSpan days = dateTo - dateFrom;

for (int i = 0; i < days.Days + 1; i++)

{

grdData.Columns[i + 2].Hidden = false;

string colName = (dateTo.AddDays(-i-1)).ToShortDateString();

grdData.Columns[i + 2].Header.Caption = colName;

grdData.Columns[i + 2].DataType = "System.Int32";

}

int rowCount = Int32.Parse(hRowCount.Value);

for (int i = 0; i < rowCount; i++)

{

DataRow dr;

string area = grdData.Rows[i].Cells[0].ToString();

for (int j = 0; j < days.Days + 1; j++)

{

DateTime day = DateTime.Parse(grdData.Columns[2 + j].Header.Caption);

DateTime dayBegin = DateTime.Parse(day.ToShortDateString() + TimeBegin);

DateTime dayEnd = DateTime.Parse(grdData.Columns[2 + j].Header.Caption + TimeEnd).AddDays(1);

DataRow[] foundRows = dt.Select("ALARM\_TIME >= '" + dayBegin + "' and AREA\_ID ='" + area + "' and ALARM\_TIME <= '" + dayEnd + "'");

int count = foundRows.Length;

grdData.Rows[i].Cells[j + 2].Value = count;

string areaName = (string)grdData.Rows[i].Cells[0].Value;

string colName = grdData.Columns[2 + j].Header.Caption;

int num = (i + 100) \* 100 + 2 + j;

grdData.Rows[i].Cells[j + 2].TargetURL = "javascript:hyperlink(" + num + ")";

}

}

lblShowDate.Text = "Duration " + dateFrom.ToString("yyyy-MM-dd") + TimeBegin + " To " + dateTo.ToString("yyyy-MM-dd") + TimeEnd;

DateTime dateFrom = DateTime.Parse(sdateFrom);

DateTime dateTo = DateTime.Parse(sdateTo);

TimeSpan days = dateTo - dateFrom;

for (int i = 0; i < days.Days + 1; i++)

{

grdData.Columns[i + 2].Hidden = false;

string colName = (dateTo.AddDays(-i-1)).ToShortDateString();

grdData.Columns[i + 2].Header.Caption = colName;

grdData.Columns[i + 2].DataType = "System.Int32";

}

int rowCount = Int32.Parse(hRowCount.Value); ……

警报监视操作界面如图5.16所示：

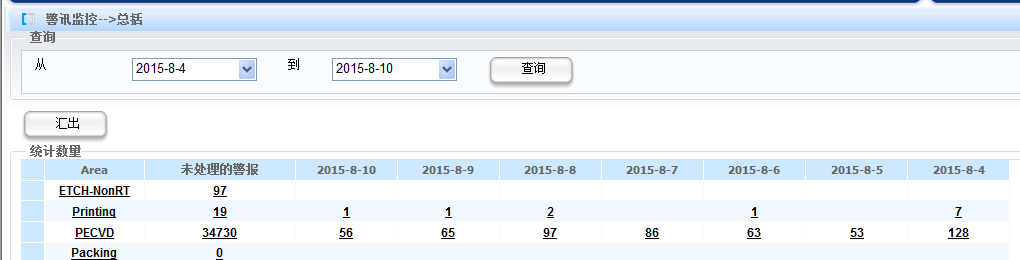


图5.16 警报监视界面

Fig. 5.16 Alarm monitor interface

6 系统测试

## 6.1 系统测试环境

系统测试环境：测试环境是为了完成软件测试工作的软件、硬件环境。

网络环境：公司网络10M；

软件环境：IE 8浏览器、火狐浏览器、谷歌浏览器、360浏览器、操作系统Windows 2003 Server以上；

硬件环境：双核以上CPU、内存1G以上、硬盘120G以上。

## 6.2 系统测试方法

系统测试不仅是检测软件的整体行为表现，也是对软件开发设计的再确认，是对集成后的产品和解决方案进行测试。系统测试的目标是验证软件产品是否满足系统需求规格说明书的要求[17]。系统测试是测试的三级层次中最接近日常测试实践的测试，它是根据预期来评估产品而不是根据规格说明或标准来评估。系统测试一般由独立测试人员或测试小组来执行，通常采用黑盒测试。

黑盒测试又称为功能测试。黑盒测试是为了确认每个实现了的功能是否符合用户的功能需求，他不涉及程序内部的实现[18]。测试时，不知道系统内部功能结构，也不知道系统内部算法与代码构造。

黑盒测试有两个显著的优点：

（1） 黑盒测试与软件具体实现无关，使用如果软件实现发现了变化，测试用来依然可以使用。

（2） 设计黑盒测试用来可以和软件实现同时进行，因此可以压缩项目总的开发时间 。

黑盒测试主要测试的错误有：性能错误、外部数据访问错误、界面错误、不正确功能[19]。

系统测试常用方法有功能测试、压力测试、性能测试、可用性测试。

（1） 功能测试

功能测试目标是为了检查应用程序是否达到了它预期的要求，判断依据是通过行为/功能说明书，测试用例包括一系列的原子能系统功能。任何程序都可以看作是将从输入定义域取值映射到输出值域的函数，将系统看成黑盒，又称之为黑盒测试，黑盒测试实现是不需要了解的，只需要知道输入和预期输出。

（2） 性能测试

通过使用用户的角度观察，系统支撑强度，在正常、峰值以及异常负载条件下，测试系统和各项性能指标，通过自动化的测试工具模拟进行。性能测试的目的是为了评估系统的能力、识别体系中的弱点、系统调优、验证可伸缩性和可靠性。性能测试的内容包括负载测试、容量测试、网络性能测试。

（3） 可用性测试

可用性测试首要目标是为了测试应用程序是否能够达到用户的需求，同时感觉使用起来明确而方便，它需要监测应用程序的界面布局、支持组件（包括安装程序、文档和帮助系统）等方法，系统操作过程中人的要素。

（4） 压力测试

Web 应用程序的压力测试是这样的一个过程：首先通过分析Web 应用程序中的应用场景来制定详细的测试计划然后构建一个接近真实的运行环境，再按照需要测试的计划内容，模拟多个使用用户对程序进行并发访问生成测试的结果，最后再通过分析Web 应用程序在多个用户并发的条件下的性能和稳定性并提交相应的测试报告 [20]。我们通过压力测试来获得系统负荷增加时用户真实的体验，从而确定运行该应用程序硬件的最大负荷，从而决定在将应用程序推广至实际应用中之前，是否必要对硬件的配置进行升级，根据系统的运行时间为程序的使用者确定可接受的运行性能的阈值，确保系统在预期的运行最大负荷时，系统性能的阈值仍然处于可接受的水平。压力测试加载方式包括分段加载压力测试、连接加载压力测试、在大数据量的基础上进行常规数据量测试。进行压力测试时需要对应和程序进行监测，主要监测客户端和服务器的应用程序的运行状态；对操作系统的监测，包括对CPU利用率、内存使用情况、硬盘读写情况；对数据库进行监测，应用程序主要访问数据库，并对数据库进行大量操作，因此对数据库的监测是十分重要的，对数据库的监测主要是使用SQL语句查询系统表进行观察，以及使用数据库提供的一些管理工具。

## 6.3 系统测试用例

晶澳太阳能SPC软件系统包含系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视功能。SPC系统测试分为7项测试，测试目的是为了验证各设定功能是否能达到要求。

### 6.3.1 系统设置测试用例

系统设置测试主要测试用户组设定、用户设定、更改密码、车间设定、区域设定、原因/处理方案设定、检测项目设定、邮件组设定、公告设定、系统登录信息。

系统设置测试用例如表6.1所示：

表6.1 系统设置测试用例

Tab. 6.1 System setup test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| 用户组设定  用户设定  更改密码  车间设定  区域设定  原因处理方案设定  检测项目设定  邮件组设定  公告设定  系统登录信息 | 点击新增、修改、删除进行操作  点击新增、修改、删除进行操作  点击更改密码，进行密码修改操作  点击新增、修改、删除进行操作  点击新增、修改、删除、查询操作  点击新增、修改、删除、查询操作  点击新增、修改、删除进行操作  点击新增、修改、删除、查询操作  点击新增、修改、删除进行操作  点击更新进行操作 | 完成新增、修改、删除功能  完成新增、修改、删除、查询功能  完成密码的修改功能  完成车间新增、修改、删除功能  完成新增、修改、删除、查询功能  完成新增、修改、删除、查询功能  完成新增、修改、删除功能  完成新增、修改、删除、查询功能  完成新增、修改、删除功能  完成系统登录信息点击更新功能 | 通过  通过  通过  通过  通过  通过  通过  通过  通过  通过 |

### 6.3.2 SPC计划和规则设定测试用例

SPC计划和规则设定测试主要管制计划设定测试和管制规则设定测试。管制计划设定测试完成管制计划的新增、复制、修改、删除、移动、汇出、查询功能的测试。管制规则设定测试完成管制规则的新增、修改、删除功能的测试。SPC计划和规则设定测试用例如表6.2所示：

表6.2 SPC计划和规则设定测试用例

Tab. 6.2 SPC setup test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| 管制计划设定  管制规则设定 | 点击新增、复制、修改、删除、移动、汇出、查询进行操作  点击新增、复制、修改、查询进行操作 | 完成管制计划的新增、复制、修改、删除、移动、汇出、查询功能。  完成管制规则的新增、修改、删除功能。 | 通过  通过 |

### 6.3.3 数据收集测试用例

数据收集主要是完成数据的收集的功能。

测试用例如表6.3所示：

表6.3 数据收集测试用例

Tab. 6.3 Data collection test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| 数据收集 | 点击进入数据类型选择，  点击收集进行操作 | 完成数据收集功能 | 通过 |

### 6.3.4 报表测试用例

报表包括CPK报表设定，新增、修改、删除、查询功能，报表测试用例如表6.4所示：

表6.4 报表测试用例

Tab. 6.4 Report test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| 报表 | 点击进入CPK报表设定，点击新增、修改、删除、查询进行操作 | 完成CPK报表的新增、修改、删除查询功能。 | 通过 |

### 6.3.5 SPC分析测试用例

SPC分析测试包括控制图分析测试和重新计算测试。其中控制图分析测试包括测试图形导出，分析，图形转换功能等功能，重新计算设定测试数据过滤器设定和查询功能的测试。SPC分析测试用例如表6.5所示：

表6.5 SPC分析测试用例

Tab. 6.5 SPC analysis test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| 控制图分析  重新计算设定 | 点击静茹控制图分析，点击趋势图资料分析、各类辅助分析图转换、数据导出进行操作  点击进入重新计算设定数据过滤器设定、查询进行操作 | 完成图形导出，分析，图形转换功能。  完成数据过滤器设定和查询功能。 | 通过  通过 |

### 6.3.6 Ocap系统测试用例

Ocap系统测试包括Ocap设定测试和Ocap报表测试。其中Ocap设定测试包括完成Ocap的处理、历史记录、汇出、生成、删除、查询功能测试，Ocap报表测试包括完成Ocap报表的查询和汇出功能测试。Ocap系统测试用例如表6.6所示：

表6.6 Ocap系统测试用例

Tab. 6.6 Ocap system test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| Ocap设定  Ocap报表 | 点击进入Ocap设定，点击处理、历史记录、汇出、生成、删除、查询进行操作  点击进入Ocap报表点击查询、汇出操作 | 完成Ocap的处理、历史记录、汇出、生成、删除、查询功能。  完成Ocap报表的查询和汇出功能。 | 通过  通过 |

### 6.3.7 警告监视测试用例

警告监视测试包括警报汇总和警报细项测试。警讯监控汇总包括完成警报的查询和汇出功能；警讯监控细项主要包括完成警讯的查询修正和汇出功能。

警告监视系统测试用例如表6.7所示：

表6.7 警告监视测试用例

Tab. 6.7 Alarm test material

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项 | 操作内容 | 预期结果 | 结果 |
| 警报汇总 | 进行警讯监控汇总操作。  进行警讯监控细项操作。 | 完成警报查询和汇出功能。  完成警讯查询、修正、汇出功能。 | 通过  通过 |

## 6.4 系统测试结论

通过各功能模块的功能测试，晶澳太阳能SPC软件系统包含系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视功能均能正常使用。

从测试结果中可以看出，本系统通过测试用例进行测试，系统在程序上基本能够达到系统的目标。

结 论

本论文主要介绍了晶澳太阳能SPC软件系统的设计与实现过程，首先，本论文对该系统的开发背景、国内外发展情况进行详细介绍，确定了开发该系统的意义和必要性。然后通过需求分析，对系统的技术与系统架构进行描述，同时对系统进行子模块划分，并详细描述了每个子模块的功能及子模块包括的内容，最终实现了系统，并对系统进行用例测试。

本系统采用了.net 4.0、Oracle10g、B/S三层架构设计模式， 在.net 4.0中主要用到了自定义控件、数据绑定控件、标准的服务器控件、Ajax控件；Oracle10g用到了子查询、左链接、右链接、内链接查询，同时为了让数据查询加快，还用到了聚簇索引与非聚簇索引。

本系统功能主要系统系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视管理模块共7个功能模块，通过系统需求分析，将使用用户分成系统管理和工程师人员，根据这些工作人员进行用户功能用例图进行分析；在系统功能模块详细设计中，将系统设置、SPC计划和规则设定、数据收集、报表、SPC分析、Ocap系统 、警报监视管理模块共7个功能模块进行一个个分析，将以流程图来陈述功能模块，同时还根据这些功能模块进行了数据库设计，在数据库设计中采用了E-R图；在系统详细实现中，分别对这7个功能模块使用时序图进行描述，同时展示了部分功能实现界面与核心代码；最后进行了对各个功能的测试，使用了黑盒测试法，并给出了测试用例。

因为时间有限，在本课题中还欠缺许多功能上的考虑，没有深入的完成一个多功能的大系统，同时在网络安全与压力测试上也存在着一定的问题。这个可以在后续进行二次开发与进一步完善。通过对晶澳太阳能SPC系统的设计与实现，基本上满足了课题的需求，提高公司整体质量管理水平。

参 考 文 献

[1] 张勇.锂锰扣式电池三参数自动检测系统研究[D]:(硕士学位论文).河北:河北工业大学,2006.

[2] 陶靖轩,刘春雨.统计过程控制的进一步讨论[J].中国计量学院学报,2007,18(1):62-65.

[3] 吴成锋.统计过程技术在质量控制中的应用[J].价值工程,2008,27(4):86-88.

[4] 钟飞,郑一维,黄蔚莉,等.统计过程控制在制管质量控制中的应用[J].焊管,2006,29(6):32-36.

[5] 艾迪明..NET框架体系结构[J].计算机工程与应用,2003,39(2):174-176.

[6] Matthew MacDonald,Adam Freeman, Mario Szpuszta. ASP.NET 4 高级程序设计(第4版)[M].北京:人民邮电出版社,2011.

[7] Alex Mackey.Introducing .NET 4.0:With Visual Studio 2010[M].美国:Apress,2010.

[8] 严含,陈甫.Oraclellg虚拟列应用[J].科学咨询,2012,13(22):69.

[9] 高扬.基于.NET平台的三层架构软件框架的设计与实现[J].计算机技术与发展,2011,21(2):77-80.

[10] 陶学恒,刘勇.基于Web的生产统计过程控制系统中产品质量分析技术的研究与开发[J].组合机床与自动化加工技术,2004,46(12):28-29.

[11] 张杨.基于粒子群优化的工业过程独立成分分析方法研究与应用[D]:(硕士学位论文).沈阳：东北大学,2010.

[12] 何桢,卢晋,刘晓亮,等.集成质量管理系统IQM2.0的设计与开发[J].工业工程,2007,10(6):696 -700.

[13] 张宏升.软件架构的非功能性需求指标和区域化支持[J].电脑知识与技术,2011,07(9):2085-2086.

[14] 王云,郭外萍,陈承欢,等.Web项目中的SQL注入问题研究与防范方法[J].计算机工程与设计,2010,31(5):976-978.

[15] 陈冈.构造基于Iframe的ASP数据处理方式[J].现代计算机（专业版）,2003,20(11):86-88.

[16] David M.Kroenk,David J.Auer.数据库原理(第5版)[M].北京:清华大学出版,2011.

[17] 淡海英.软件项目中系统测试概述[J].工业仪表与自动化装置,2014,44(6):15-16.

[18] 万年红,李翔.软件黑盒测试的方法与实践[J].计算机工程,2000,26(12):91-93.

[19] Molyneaux.应用程序性能测试的艺术[M].北京:机械工业出版社,2009.

[20] 肖芝.面向服务在线实验平台的研究[D]:(硕士学位论文).武汉:华中科技大学,2006.

攻读硕士学位期间发表学术论文情况

致 谢

时间转瞬而过，在这段在职攻读研究生的几年时间里，我在大工各位老师的悉心指点下，我在学业上的进步空间不断地增强，以前在实践工作中碰到的问题，在学习过程中，得到了理论支持，让我明白了如何解决问题理论基础。

本课题在我的导师与学校指导老师的悉心指导和殷切关怀下完成的，感谢导师对我的谆谆教导与面对学术一丝不苟的精神，深深地印入我的脑海，让我在学习、研究、工作、生活等各方面都有指导意义。在编写课题期间，我遇到了许许多多的问题，都是以前在工作中从来没有碰到的领域，特别是软件程序开发上，通过大工学习过程中，学到了放多的理论知识，但一到真正去实践的时候，就发现完成课题需要的知识包括的内容非常多，比如说，如何进行需求分析、详细分析、数据库设计、程序编码、包括界面如何实现、系统测试等。这些通过多年工作经验的校内指导老师的指导下，让我学习到了许多的知识，再次对导师进行感谢，同时感谢在写论文过程中帮助我和晶澳同事和朋友们。

最后，感谢诸位老师认真审阅我的论文，让我在学术上进一步提高。

大连理工大学学位论文版权使用授权书

本人完全了解学校有关学位论文知识产权的规定，在校攻读学位期间论文工作的知识产权属于大连理工大学，允许论文被查阅和借阅。学校有权保留论文并向国家有关部门或机构送交论文的复印件和电子版，可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印、或扫描等复制手段保存和汇编本学位论文。

学位论文题目： 晶澳太阳能SPC系统的设计与实现

作 者 签 名 ： 日期： 年 月 日

导 师 签 名 ： 日期： 年 月 日