

# 基于 ARM9+Linux 轨道交通门控器测试设备的研制

陈明学<sup>1,3</sup>, 左思渊<sup>2</sup>

(1. 南京邮电大学通信与信息工程学院, 南京 210003; 2. 东南大学信息科学与工程学院, 南京 210088; 3. 南京工程学院, 南京 211167)

**摘要:** 目前轨道交通门控器定期检修维护尚无相应现场检测仪, 现场手工测试效率低、准确性不高, 并且要求对门控器设计原理很熟悉。该文在分析总结手工测试原理和业务逻辑的基础上, 成功研制了基于 ARM9 + Linux 的移动式轨道交通门控器现场检修维护设备, 使检测人员可以现场判定故障所在的具体元器件, 对故障作出定量分析, 介绍了系统结构、功能以及实现中的关键技术。

**关键词:** 轨道交通门控器; Linux 实时性; GPRS; YAFFS 文件系统

## Design of Railcar Electro-door Control Unit Test Equipement Based on ARM9&Linux

CHEN Mingxue<sup>1,3</sup>, ZUO Siyuan<sup>2</sup>

(1. College of Telecommunications & Information Engineering, Nanjing University of Post and Telecommunications, Nanjing 210003;

2. College of Information Science and Engineering, Southeast University, Nanjing 210088; 3. Nanjing Institute of Technology, Nanjing 211167)

**【Abstract】** At present, there is not the equipment to test railcar electro-door control unit on the spot. The efficiency and veracity is very low by hand-test, and tester is required to know well the design of railcar electro-door control unit. Based on the analysis of the principle and logic by hand-test, this paper designs the equipment with embedded CPU and Linux, so test man can easily make a quantitative analysis. The paper introduces the equipment's structure, function and main technique.

**【Key words】** Railcar electro-door control unit; Linux real time; GPRS; YAFFS files system

轨道交通门控器是应用于地铁、轻轨、干线火车塞拉门的控制器, 负责塞拉门的开门、关门、检测、通信, 其可靠性是至关重要的。在使用过程中, 需要定期检测维护, 传统的检测方式是手工方式, 将门控器拆下来检测, 如果确认是轨道交通门控器出现故障, 则将其寄回生产厂家检修, 需要花费大量时间; 轨道交通门控器技术含量高, 逻辑复杂, 在发现问题时如何让检测人员现场判定故障所在的具体元器件, 对故障作出定量分析, 并且在现场维修一些简单的故障后, 如何对维修结果再进行现场整体测试以防出现其它故障隐患, 就要求测试人员必须对城市轨道交通门控器的原理非常了解, 并且测试中不能有任何遗漏错误。

目前, 国内尚无相应的轨道交通门控器现场检测仪, 国外也只有基于 PC 机的检测仪, 因其体积过大不适合携带和 PC 机对外界环境适应能力的局限性而不适合现场检测, 因此现场定期检测使人们希望有一种能提高检测效率和检测结果准确性, 同时又能降低对测试人员的专业知识要求难度的现场检测仪。笔者在和南京某轨道交通门控器生产厂家的合作过程中, 在与其技术人员一起分析总结手工测试原理和业务逻辑的基础上, 根据业务需求和适用环境, 采用嵌入式系统和源码开放的 Linux, 成功研制出了基于 ARM9 + Linux 的移动式轨道交通门控器现场检修维护设备, 使检测人员可以轻松高效、准确可靠地完成日常检修工作。

### 1 系统设计

#### 1.1 现行检测原理分析

在传统的现场手工测试过程中, 需要为轨道交通门控器输入要求的电流或电压模拟现实运行过程中的条件, 引发门

控器内部的程序, 控制塞拉门的动作, 同时门控器会将采集到的状态信息反映到一些 LED 灯的开关状态上, 然后使用万用表对轨道交通门控器一些相关位置的电阻、电容、电压、电流进行测试, 结合 LED 灯的开关状态所反映的信息, 来对门控器作出故障定性分析。若出现故障, 寄回生产厂家后, 由专家使用计算机和门控器内部的 CPU 进行通信, 获取门控器内部采集到的状态信息以及运行中间数据, 同时结合万用表的手工测试结果, 根据对电路图原理的理解进行相关测试, 逐步推导, 从而找出具体的出故障的元器件。

根据轨道门控器硬件电路设计的原理, 门控器的电路可以分成几大部分, 各部分之间相对比较独立, 例如该公司的 EDCU2036 型号门控器就可以分为电阻电容模块、电源模块、通信模块、数字 IO 模块、电机驱动模块等 5 个大的部分; 同时, 各个部分内部也因具体控制的功能不同而可以将元器件分成一些功能组, 在具体的某一功能组元器件内部就具有相互推导的关系, 例如图 1 就是电源模块中最简单的一个功能组内的元器件相互推导关系: 采集 Q13.C-SGND, U35.3-SGND, ENVDD-GND, VDD-SGND 4 个电压信号值, 以判断 Q13、U35、U12 等一组元器件是否有故障的判断逻辑。在分析总结了城市轨道交通门控器的原理和以前手工测试逻辑的基础上, 笔者和技术人员将其测试逻辑原理表示为可以用计算机实现的逻辑。

**作者简介:** 陈明学(1974 -), 男, 博士, 主研方向: 无线信号处理和嵌入式应用; 左思渊, 本科

**收稿日期:** 2006-06-15 **E-mail:** MXChen@njit.edu.cn

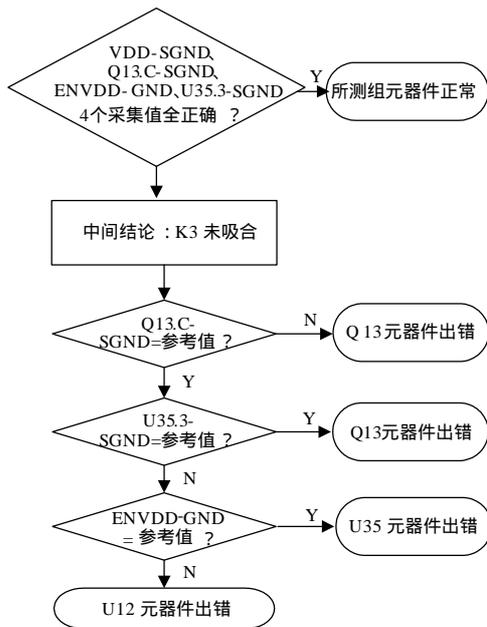


图1 简单测试逻辑示例

## 1.2 系统结构

整个系统根据功能分为两个大的部分：现场检测仪和位于生产厂家的数据库服务器。数据库服务器主要实现数据的管理和接收、显示测试现场的图像数据供专家分析；移动检测仪主要完成现场测试、摄取塞拉门周围图像以及通过GPRS网络发送图像数据功能。

考虑到为了在测试不同型号的门控器时能最大化地共享测试硬件，以节约硬件成本，所以在设计过程中采用模块化的思想，将检测仪系统分为核心设备、图像摄取模拟、GPRS通信模块、信号同步转换采集板等部分，在逻辑上将检测仪系统中与具体门控器相关的电路集中在信号同步转换采集板上，从而在测试不同型号的轨道交通门控器时只需要修改信号同步转换采集板和不同的软件设置就可以了。

(1)检测仪的核心设备。嵌入式系统因其体积小、功耗低、软硬件可裁剪、性价比高等特点成为便携式仪器的硬件首选。本系统为了满足仪器便携性，采用了 ARM9 系列嵌入式 CPU，三星公司的 S3C2410，主频为 206MHz。所以系统核心部分由一块体积很小的电路板组成，上面集成了 ARM9 系列嵌入式 CPU 三星公司的 S3C2410、NandFlash、SDRAM 等主要芯片和一些外围扩展电器，它带有 6 路 8 位 AD 数据采集接口，16 位数字量输入口，16 位数字量输出口，两个 RS232 接口和一个 RS485 接口，一个可以外接 U 盘的 USBHOST 接口，系统可以通过 U 盘与数据库服务器通信（测试数据和控制程序等）。

(2)图像摄取模块。因为在现场检测时，塞拉门表现出某一故障，它可能涉及到机械、电子、门控器本身 3 方面的故障，即使是熟练的检测人员也可能出现不能明确判断是否是因为门控器发生故障，这时就需要生产厂家的专家来检测了，所以为了在出现这种疑难杂症时，也能通过专家的知识解决问题，系统集成了 CMOS 摄像模块，可以由现场检测人员通过摄像头将塞拉门相关的图像资料传送回厂家，由专家来根据测试数据和图像数据来诊断故障。

为了实现图像采集和压缩功能，经比较后采用 D192 嵌

入式摄像模块(Pictos 公司产品)，由于该模块集成了数字信号处理和 CMOS VGA 图像传感器，大大简化了硬件设计，并提高了系统工作的可靠性，它可以将拍摄的 RGB 源图像转化成 MJPEG 压缩格式，并通过 8 位数据口并行输出。帧速率、输出图像大小、格式可编程调整，使用两个串口控制工作状态，设定工作模式。

(3)GPRS 通信模块。在系统中通过 GPRS 网进行数据传输需要使用 GPRS 模块，系统采用的是 Motolra 公司的 G20 GPRS 无线通信模块。微处理器通过串口使用 AT 指令集与 GPRS 模块通信，主要包括串口发送(TX)、串口接收(RX)以及微处理器与 GPRS 之间一些硬件握手信号。为了简化微处理器的控制，在满足系统功能要求的情况下，系统硬件设计时仅使用了数据载波检测(DCD)和终端准备就绪(DTR)信号。

(4)信号同步转换采集板。该部分集成了一些端子板、集成电路、模拟传感器、用于连接的电缆线、用于卡放轨道交通门控器的装置、探针；工作时，将卡放装置卡紧轨道交通门控器，使探针接触到需要测试的位置上，采集到信号，然后将轨道交通门控器的信号通过电器转换成检测仪能接收识别的信号，如将一些电压阈值转换成数字 IO 信号，或将一些模拟信号同比例缩小，以利于检测仪识别。

在测试时，移动检测仪与门控器之间通过检测仪的总线扩展接口引线、信号同步转换采集板的模拟信号和数字信号的采集引线相连，具体系统结构如图 2 所示。

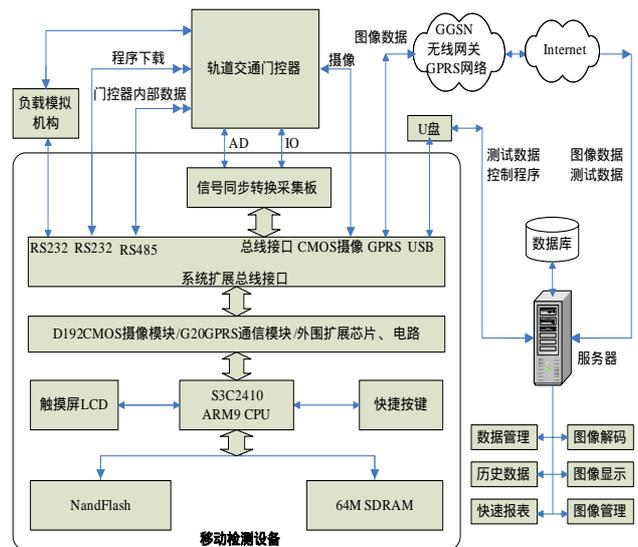


图2 系统结构

## 1.3 系统软件功能描述

移动轨道交通门控器检测仪主要功能为：(1)用户操作终端，使用快捷按键或触摸笔来操作控制测试过程；(2)采集处理城市轨道交通门控器的测试数据并根据测试逻辑来进行判断和控制；(3)存储测试数据；(4)通过 RS232 接口将门控器所用的程序下载到门控器内以更新门控器内的程序；(5)提取门控器采集到的内部错误信息；(6)通过 CMOS 摄像将塞拉门的图像资料传送回生产厂家，由专家来指导诊断故障；(7)通过 U 盘作为中介、系统本身作为 U 盘两种方式与外部数据库服务器进行数据交互，即将测试数据汇总到服务器，以及从服务器中下载门控器工作时所需要的控制程序。

具体测试时，测试的方式可以分为 3 种方式：(1)全自动测试方式，在测试仪安装好之后，通过仪器的快捷按键可以

一键完成全部功能的测试；(2)自动测试某一部分功能元器件，主要适用于维修后对维修局部进行复测，比如只单独测试电机驱动模块等；(3)全手工测试，主要是在检测仪本身元器件如探针等出现故障时，由系统提供判断测试结果和数据管理的功能。

在自动测试时，测试仪器采集到某一项测试数据后，以文字的形式、图像曲线的方式将测试数据和标准参考值显示给用户查看，如果测试数据超过了误差范围，系统将以醒目的红色字体提醒检测人员；当所有项目测试结束后，系统会给出整体测试结果以及各子项目的测试结果，如果有故障出现，会以醒目颜色的字体列出故障发生的子项目名称和具体出故障的元器件，让用户能明了地看出测试结果或出现故障的具体元器件名称。

系统同时提供了多种查询统计功能，可以查看设定参数之内的各门控器测试结果，系统列出各门控器总体判断结论和各子项目的判定结论，可以用触笔点击查看具体的某一历史记录的实测数据。系统也可以统计出具体时间段或某型号门控器的某一部分出现故障的几率，让设计者、生产商、用户了解到故障易发生的元器件名称。

## 2 系统关键技术的研究

### 2.1 Linux 实时性的改善

目前，自制仪器多选用源码开放的Linux作为操作系统。但Linux作为通用操作系统，其进程调度机制并非专为工业实时性而设计，定时器只提供 10ms 的调度粒度<sup>[1]</sup>，较难满足工业级的数据采集等硬实时性要求，在实现过程中，需要对其实时性进行改善。改善实时性的方法很多，有系统内核调用法、细化时钟粒度、构建实时内核等方法<sup>[2]</sup>，都能改善Linux的某一方面实时性，但存在难以兼顾软硬两方面实时性的问题，笔者通过构建双内核的方式可以针对不同的实时要求给予其不同的响应处理，并在移动式轨道交通门控器检测维护仪中得到较好的实现。

#### (1)改善 Linux 实时性的方法研究

改善实时性的方法很多，有系统内核调用法、细化时钟粒度、构建实时内核等方法。但调用Linux内核一些高实时性函数的系统内核调用法因Linux内核函数的适应面而使该方法在工业应用中并不太适合；使用修改Linux内核定时参数 HZ 的初值来细化调度时钟粒度的方法在实现时很多时候反而会增加系统开销<sup>[3]</sup>，也很难解决工业数据采集中的实际问题；通过构建实时内核，直接修改Linux调度器，在Linux内核中创建可抢占点，从而将Linux内核改成一个可抢占任务的调度内核<sup>[4]</sup>。这种方法可以很好地改善Linux的实时性，但修改Linux内核的任务调度器，改为可抢占式的任务调度器要求编程人员对Linux内核要非常熟悉，所以难度较大，调试过程比较繁琐，并且难以保证最终的内核的稳定性。

#### (2)双内核结构

在工业应用中，除了与硬件相关的功能对实时性要求高之外，其它的文件管理、数据操作等功能对实时性的要求Linux完全能胜任，且更稳定，所以采用双内核结构，在保持Linux内核的完整性的同时，外加一层实时任务调度器的壳，则能满足软硬两方面实时要求，结构如图3所示。

双内核结构包含了实时内核、Linux基本内核、硬件抽象层、私有API函数集。其中实时内核截取硬件中断，并且根据实时内核调度器对任务的实时要求进行判断，决定调用Linux基本内核来处理任务还是调用私有API函数来完成该

任务，起到一个重定向的作用；Linux基本内核处理非实时任务，如文件操作等实时性要求不高的任务，实时任务和非实时任务之间信息的交换要通过管道(FIFO)或共享内存(MBUFF)来实现；系统通过硬件抽象层进行硬件管理，把基本内核和实时内核结合在一起，其中一个内核的改变，不会影响另一个内核的执行；私有API函数集是完成具体的工业应用、数据采集等实时要求高的任务的一些函数集，供实时内核调用。在系统运行时，内核将Linux基本内核作为优先级最低的一个任务来运行，只有在没有实时任务运行的时候才予以调度。

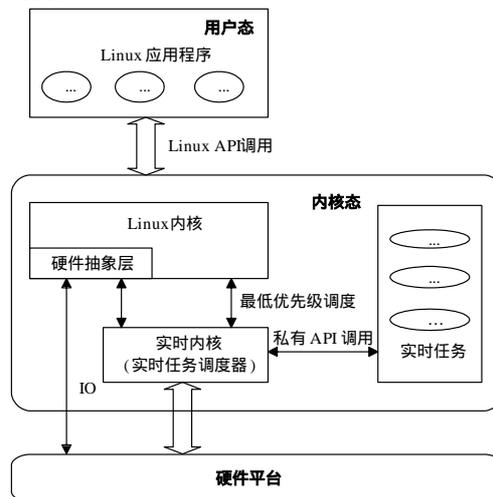


图3 实时内核框架

#### (3)双内核运行原理

为了判断任务的实时性，从而调用不同的处理函数，实时内核内有一个关键组件：中断描述符表(Interrupt Descriptor Table, IDT)，它定义了一套指针用来处理中断例程作为系统同硬件之间的抽象接口，主要用来截取硬件中断，并且依据实时内核调度器的需求，重定向为Linux基本内核任务还是调用私有API函数来完成实时任务。硬件抽象层本身定义了一个结构，使得Linux基本内核中断处理函数能够很容易地被私有的实时处理函数所替代，这样，当实时内核通过硬件抽象层激活后，新的IDT表则合法。

在运行过程中，当中断到来时，实时内核判断它是基本内核中断还是属于实时中断，分别进行处理。若是基本内核中断，如果当前有实时任务正在运行，则只是设置一下中断标志位，悬挂此中断，如果没有实时任务运行，则调用基本内核的中断处理程序；如果此中断是实时中断，就直接调用相对应的实时中断处理程序。在这个过程中，实时内核实际上是把Linux内核当作它的一个低优先级的实时任务进行处理，由于Linux核心优先级最低，且可被其他实时任务抢占，因此运行在用户态下的Linux应用程序不会影响运行在内核态下的实时任务的实时性能，保证了数据采集的高实时性。这样既可以充分利用Linux的特性，又能解决实时性的要求。

### 2.2 GPRS 通信的建立

使用GPRS进行数据的无线传输可以通过2种移动分组路由，路由可以分为内部PLMN数据传送路由和互联PLMN数据传送路由<sup>[5]</sup>，由此相应也就有2种传输方法：(1)在移动GPRS网内进行点对点的数据传输，要求收发双方都包含GPRS模块；(2)使用GPRS模块登录GGSN网关，通过无线上网的方式将数据传送到远端PC上。

对于第(1)种方式，要求图像接收端的PC机上必须装有

GPRS 接收模块,而第(2)种方式,只要求接收端的 PC 机能连接到 Internet 网络上即可,为了增强图像接收的 PC 机的灵活性,使专家可以在任何计算机上安装相应的软件即可接收现场的检测仪发回的图像数据,所以在系统中采用的是第(2)种 GPRS 通信方式。

通过 Internet 网络通信的方式,由于 Linux 具有 TCP/IP 协议栈的支持,在初始化程序中完成网络连接后,系统直接使用操作系统的 API 函数实现 GPRS 模块和 GGSN 之间的协商通信。主要过程是首先 GPRS 模块通过拨“\*99#”连接 GGSN,接下来与 GPRS 网关进行通信链路协商,协商过程遵守 LCP(Link Control Protocol)、PAP(Password Authentication Protocol)和 IPCP(Internet Protocol Control Protocol)等协议,一旦协商完成,链路就已经创建,得到分配的 IP 地址,就可以按协商的标准进行 IP 报文传输了。编程的一个难点是微处理器登录 GGSN 并与网关通过 LCP、PAP、IPCP 协议进行协商的过程。

因为系统程序中只有处理请求(REQ)、同意(ACK)和拒绝(NAK)3种帧,其它链路问题都通过程序重新拨号来解决,所以整个协商过程都得用这3种帧来完成。协商程序描述如下:在拨号连接成功后,GGSN 首先返回一个 PAP REQ 数据帧,程序发送一个空 LCP REQ 帧,以强迫方式进行协议协商,随后 GGSN 发送 LCP 设置帧,程序拒绝所有的设置并请求验证模式,GGSN 选择 CHAP 或 PAP 方式验证,程序接受 PAP 方式,然后进行 PAP 验证用户名和密码过程,在 GPRS 中用户名和密码都设置为空,如果验证成功,GGSN 会返回 IPCP 报文分配动态 IP 地址,这样就完成了 GPRS 模块与 GGSN 的协商过程。协商完成后进入通信阶段,同时也开始按流量计费,这时微控制器向 GGSN 发送的所有包含 IP 报文的数据报都会被传送给 Internet 网中相应的 IP 地址,而远端所有向微处理器 IP 地址发送的报文也都会经 GPRS 网传送到微处理器上,从而完成微处理器的无线上网数据传输。

### 2.3 YAFFS 文件系统的建立

移动型的数据检测仪不仅要完成检测功能,同时也存储大量检测数据,所以文件系统的建立也是非常重要。在本系统中,数据量相对比较大,选用容量较大的 NandFlash 芯片,因此必须移植适用于这种 Flash 的 YAFFS 文件系统。YAFFS 是按层次设计的日志结构文件系统,具有损耗平衡和掉保护功能,能避免意外掉电时文件系统的一致性和完整性的影响。

实现用户程序区支持 YAFFS 格式文件系统一般是把 YAFFS 配置到内核一起编译的方法,这种方法需要增加 YAFFS 源码到内核,而且还要下载较新版本的支持 YAFFS 的 MTD 驱动程序,在内核配置的过程中也要对配置文件和 Makefile 文件做大量的修改,因此工作量大且容易出错。

作者使用 insmod 命令手工加载模块,可以很简单方便地实现 YAFFS 文件系统的建立。具体实现步骤为:

(1)从网上下载 YAFFS 源码到宿主机,修改其 Makefile 文件中的内核源码路径 KERNELDIR 变量值和交叉编译工具 MAKETOOLS 变量值,配置成系统内核所在目录和所用到的交叉工具所在的目录,都使用完全路径名;

(2)在源码目录下使用 Make 命令编译,生成 yaffs.o 文件;

(3)将 yaffs.o 放于宿主机根文件系统 /root/lib 下,并与整个 /root 一起生成根文件系统映像 root.carmfs,下载到检测仪核心设备之后,使用超级终端在开发板上手动加载 yaffs.o,命令为 #insmod yaffs.o;

(4)在宿主机上建立一个名为 /usr 的目录,存放常用的用户程序和库文件,利用 mkyaffsimage 程序制作 YAFFS 映像 usr.yaffs,命令为: #/mkyaffsimage usr usr.yaffs;

(5)在检测仪核心设备上执行“mount -t yaffs /dev/mtdblk4/usr”,实现 NandFlash 设备对 YAFFS 文件系统的支持,文件系统挂载在 /usr 目录下,可在此目录上执行建立和删除目录、文件等操作,这样就建立起了 YAFFS 文件系统。

### 3 总结

目前,系统在基于 S3C2410 和 Linux-2.4.18.tar.gz 版本之上调试成功,经过和成熟的基于工控机的测试设备对比测试,测试结果完全正确。在现场检测过程中,能很好地达到现场采集处理数据的实时性,以毫秒级的实时性准确地采集到门控器输出的脉冲电压值,在 80s 之内可以可靠地测试完所有元器件是否存在故障,以及出故障的具体元器件,实时给出包含测试数据、标准参考值数据、测试判定结果的报表;能够与位于生产厂家的 PC 机建立较好的 GPRS 通信,使专家能比较清晰地查看现场的摄像;保证测试数据的存储和历史数据的方便查看,可以通过多种方式和数据库服务器方便地进行共享测试数据。达到了高效、准确地测试,同时又降低对测试人员的专业知识要求难度的目的。目前已开始批量生产,投入到实际应用之中。

#### 参考文献

- 1 毛德操,胡希明. Linux 内核源代码情景分析[M]. 杭州:浙江大学出版社,2001.
- 2 李凡,卢杜阶,邱鹏. 在工业应用中增强 Linux 实时性的方法研究[J]. 华中科技大学学报,2005,33(2): 83-85.
- 3 吴姣梅,李红艳. 改善嵌入式 Linux 实时性能的方法研究[J]. 微机计算机信息,2006,22(2): 72-74.
- 4 唐寅. 实时操作系统应用开发指南[M]. 北京:中国电力出版社,2002.
- 5 李华. 现代移动通信技术-GPRS 系统[M]. 广州:华南理工大学出版社,2001.

(上接第 225 页)

#### 参考文献

- 1 朱心雄. 自由曲线曲面造型技术[M]. 北京:科学技术出版社,2000.
- 2 Woodward C. Skinning Techniques for Interactive B-spline Surface Interpolation[J]. Computer Aided Design, 1998, 20(8): 441-451.
- 3 Hamann B, Tsai P Y. A Tessellation Algorithm for the Representation

of Trimmed NURBS Surface with Arbitrary Trimming Curves[J]. CAD, 1996, 28(6/7): 461-472.

- 4 施法中. 计算机辅助几何设计与非均匀有理 B 样条[M]. 北京:北京航空航天大学出版社,2000.

- 5 Piegl L, Tiller W. Symbolic Operators for NURBS[J]. CAD, 1997, 29(5): 361-368.