

上海交通大学

SHANGHAI JIAO TONG UNIVERSITY

学士学位论文

THESIS OF BACHELOR



论文题目：数控机床数据采集嵌入式适配器设计与开发

学生姓名：高树华

学生学号：5080209041

专 业：机械工程与自动化

指导教师：张洁（教授）

学院(系)：机械与动力工程学院

2012年5月

数控机床数据采集嵌入式适配器设计与开发

摘要

随着信息技术和网络技术的发展,我国着力推动制造业信息化,而数控机床作为制造自动化的核心设备,其数据采集对于企业实现数字化和信息化具有重要意义。但我国大多企业对数控机床并没有实现有效的数据采集。同时,嵌入式技术的发展使其在工业自动化领域发挥越来越大的作用。因此,研究利用嵌入式技术实现数控机床的数据采集具有十分重要的现实意义。

论文分析了各种数据采集方法的优缺点及其适用范围,结合数据采集和车间管理的实际需求,提出了一种基于 ARM11 嵌入式平台和机床串口通信的数据采集方案,并设计了基于嵌入式适配器的数控机床数据采集系统的 C/S 架构,进而设计了数据采集服务器的数据库物理模型。

论文在数据采集功能需求分析和数据采集系统整体架构的基础上,进行了嵌入式硬件平台的设计,并进行了嵌入式操作系统 Windows CE 的定制和移植。研究了基于数控程序内嵌宏输出指令,并利用机床串口通信实现数据采集的方法。研究了串行通信、远程数据访问和多线程等关键技术,探讨了数据采集、数据通信和数据传输等关键模块的开发,并利用 C# 语言实际开发了嵌入式适配器的数据采集应用软件。

论文在嵌入式适配器开发之后,探讨了远程客户端数据管理系统的实现方法。设计了客户端数据管理系统的各功能模块和系统的图形用户界面,研究了数据访问和数据绑定等关键技术,探讨了机床树模块和数据统计处理等关键模块的开发,并利用 C# 语言实际开发了远程客户端的数据管理系统。

论文最后进行了基于嵌入式适配器、包括服务器和远程客户端在内的数控机床数据采集系统的测试与运行,验证了嵌入式适配器和数据采集系统在数据采集、无线通信和数据统计处理等方面的可行性。

关键词: 数控机床数据采集, 串行通信, 远程数据访问, Windows CE, 嵌入式

DESIGN AND DEVELOPMENT OF THE EMBEDDED ADAPTOR FOR DATA COLLECTION OF CNC MACHINES

ABSTRACT

With the development of IT and the network technology, China begins to make efforts to promote the Manufacture Information Engineering. Since CNC machines are the core equipment of manufacturing automation, the data collection of them has a great significance for the digitization and informatization of enterprises. But most of the manufacturing enterprises in China haven't got effective data collection of CNC machines. Meanwhile, the development of embedded technology has made itself play a more important role in the field of industrial automation. Therefore, it has a very important practical significance to make research on CNC machines' data collection with embedded technology.

This paper first summarizes and compares the advantages and disadvantages of various data collection methods and their scope of application. According to the actual needs of the data collection and shop floor management, this paper proposes a data collection program based on the ARM11 embedded platform and CNC machines' serial communication. Then, the C/S structure of the entire data collection system based on the embedded adaptor is designed.

Based on the functional requirements analysis and the overall architecture of the system, the embedded hardware platform is designed and the customization and transplantation of an embedded operating system, Windows CE, is executed. Then, this paper proposes a method to make data collection of CNC machines with serial communication and inlaid macro output instruments in the NC programs. Besides, some key technologies are discussed such as serial communication, remote data access, and multi-threading. The development of some key modules such as data collection, data communication and NC programs' transmission are also expounded. Finally, the data collection application software is developed with C#.

In the next, this paper investigates the realization of the data management system on the remote client. On the first, the functional modules of the client-side data management system are designed. Then, the system's graphical user interface is designed and some key technologies including of data access and data binding are studied. In addition, the development of some key modules such as the machine tree and data processing are discussed. In the end, the client-side data management system is designed with C#.

Finally, the entire data collection system based on the embedded adaptor is tested to verify the feasibility of this system on aspects of data collection, wireless communication and data management and so on.

Key Words: data collection of CNC machines, serial communication, RDA, windows CE, Embedded

目录

第 1 章 绪论	1
1.1 引言	1
1.2 数控机床数据采集技术综述	2
1.2.1 数控系统的组成	2
1.2.2 数控机床数据采集的原理	2
1.2.3 数控机床数据采集的任务	2
1.2.4 数控机床数据采集的方法	2
1.2.5 数控机床数据采集技术的研究现状	3
1.2.6 数控机床数据采集技术的发展趋势	4
1.3 嵌入式技术概述	4
1.3.1 嵌入式系统定义及特点	4
1.3.2 数控机床嵌入式数据采集的优点	5
1.4 课题的研究意义、研究目标及研究内容	5
1.4.1 课题的研究意义	5
1.4.2 课题的研究目标	5
1.4.3 论文的主要内容	5
1.5 本章小结	6
第 2 章 基于嵌入式适配器的数控机床数据采集系统的总体设计	7
2.1 数据采集系统的功能需求分析	7
2.1.1 数据采集、储存、查询和统计处理功能	7
2.1.2 数控程序的存储、管理和传输功能	7
2.1.3 车间设备及用户信息管理功能	7
2.2 数据采集方法的选择	8
2.2.1 各数据采集方法的优缺点及适用范围	8
2.2.2 基于串口的数控机床数据采集方法	8
2.3 基于嵌入式适配器的数据采集系统的总体架构	9
2.3.1 数据采集系统总体结构设计	9
2.3.2 嵌入式适配器处理器的选择	10
2.3.3 嵌入式适配器硬件平台设计	11
2.3.4 嵌入式操作系统的选择	12
2.3.5 数据库结构设计	12
2.1 本章小结	14
第 3 章 嵌入式适配器的实现	15
3.1 嵌入式适配器的硬件平台	15
3.1.1 嵌入式处理器 S3C6410	15
3.1.2 飞凌 OK6410 开发板	15
3.2 嵌入式操作系统 Windows CE 的定制与移植	16
3.2.1 Windows CE 的体系结构	16
3.2.2 Windows CE 的定制和移植	17
3.2.3 嵌入式数据库 SQL Server Compact 3.5	17

3.3	嵌入式适配器数据采集应用软件的功能模块设计	18
3.3.1	数据采集应用软件各功能模块的划分	18
3.3.2	数据采集应用软件各功能模块的主要功能	18
3.4	嵌入式适配器数据采集应用软件的开发	20
3.4.1	Windows CE 嵌入式开发环境的搭建	20
3.4.2	数据采集应用软件的开发语言	20
3.4.3	数据采集应用软件的关键技术	21
3.4.4	数据采集应用软件关键模块的实现	24
3.5	本章小结	30
第 4 章	客户端数据管理系统的实现	31
4.1	客户端数据管理系统的功能模块设计	31
4.1.1	数据管理系统各功能模块的划分	31
4.1.2	数据管理系统各功能模块的主要功能	31
4.2	客户端数据管理系统的关键技术	33
4.2.1	服务器数据库 SQL Server 2008 及 ADO.NET 数据访问技术	34
4.2.2	数据绑定技术	34
4.3	客户端数据管理系统的界面设计	35
4.3.1	登录窗体	35
4.3.2	主窗体	36
4.3.3	企业日历配置窗体	36
4.3.4	管理功能窗体	36
4.4	客户端数据管理系统关键模块的实现	37
4.4.1	机床树的实现	37
4.4.2	数据统计处理模块的实现	41
4.5	基于嵌入式适配器的数据采集系统的测试与运行	45
4.5.1	嵌入式适配器与服务器无线通信测试	45
4.5.2	嵌入式适配器数据采集测试	46
4.5.3	客户端数据管理系统数据查询与管理功能测试	48
4.5.4	客户端数据管理系统数据统计处理功能测试	50
4.6	本章小结	50
第 5 章	总结与展望	52
	参考文献	53
	谢辞	55

第1章 绪论

1.1 引言

众所周知，制造业是综合国力和国民经济的重要基础。21 世纪，围绕以知识为基础的创新能力竞争，一场以信息技术为基础的全球制造业革命正在波澜壮阔地展开。而随着我国科技能力和生产能力的提高，中国正在成为全球制造业的中心，但是中国只是制造业大国，还不是强国。因此，国家确定通过信息化带动工业化的国策，推动制造企业实施制造业信息化。

制造业信息化（MIE，Manufacture Information Engineering），将信息技术、自动化技术、现代管理技术与制造技术相结合，可以改善制造企业的经营、管理、产品开发和生产等各个环节，提高生产效率、产品质量和企业的创新能力，降低消耗，带动产品设计方法和设计工具的创新、企业管理模式的创新、制造技术的创新以及企业间协作关系的创新，从而实现产品设计制造和企业管理的信息化、生产过程控制的智能化、制造装备的数控化以及咨询服务的网络化，全面提升我国制造业的竞争力。

制造执行系统是制造业信息化的一个重要构成。制造执行系统（MES，Manufacturing Execution System）是近此年来在国际上迅速发展、面向车间层的生产管理技术与实时信息系统，美国先进制造研究机构 AMR（Advanced Manufacturing Research）将 MES 定义为“位于上层的计划管理系统与底层的工业控制之间的面向车间层的管理信息系统”^[1]。MES 可以为用户提供一个快速反应、有弹性、精细化的制造业环境，帮助企业减低成本、按期交货、提高产品的质量和提高服务质量。适用于不同行业，能够对单一的大批量生产和既有多品种小批量生产又有大批量生产的混合型制造企业提供良好的企业信息管理^[2]。而这实现这一切功能的基础就是现场的数据采集，只有准确、及时地采集到底层的现场数据信息，才能使 MES 发挥其应有的作用。

随着制造业和国民经济的快速发展，数控机床在实际生产中的应用越来越广泛，汽车、船舶、工程机械、航空航天等行业的生产车间越来越多地采用数控机床，数控机床已经成为 MES 系统底层监管和控制的对象。当前大多制造企业正纷纷采用分布式数控（DNC，Distributed Numerical Control）系统对车间内数控设备进行管理。基于以太网和 TCP / IP 的 DNC 系统，以其所具有的易于联网、传输速度快、实现成本低等优势，成为研究的热点并广泛应用。但目前我国国内大多数 DNC 系统并不具有数据采集的功能，从而造成数控机床与上层的管理系统存在信息不对称的现象：即数控机床可以接受上层管理系统的指令，但不能及时反馈实际的生产状态信息给上层管理信息，信息流是单向的，没有形成有效的闭环信息流。而缺乏有效的底层信息支持，上层管理系统就难以做出高效的生产计划及调度安排，导致数控机床的利用率低，降低了制造企业的竞争力，也妨碍了制造信息化的进程。

因此，需要建立有效的数控机床数据采集系统，从而将数控机床设备运行及工件加工信息及时采集并传递到 MES 上层管理系统中，保证生产过程的透明，并且时刻处于管理系统的有效控制之下，从而使得管理系统可以及时调整及安排生产计划，提高数控机床的整体利用率，进而提高整个制造企业的经济效益和竞争力，推动我国制造业信息化的发展。

另外，随着嵌入式技术的发展，已经有很多成功的应用，并且智能手机和平板电脑的热卖也推动了嵌入式技术的发展及普及，嵌入式技术已经成为工业控制和生产管理研究的一个热点。所以本课题研究数控机床数据采集嵌入式适配器的设计与开发。

1.2 数控机床数据采集技术综述

1.2.1 数控系统的组成

数控机床的数控系统由输入/输出装置、计算机数控（Computer Numerical Control, CNC）装置、可编程控制器（Programmable Logic Control, PLC）及其接口电路、主轴伺服驱动装置、进给伺服驱动装置以及检测装置等组成，如图 1-1 所示^[3]。

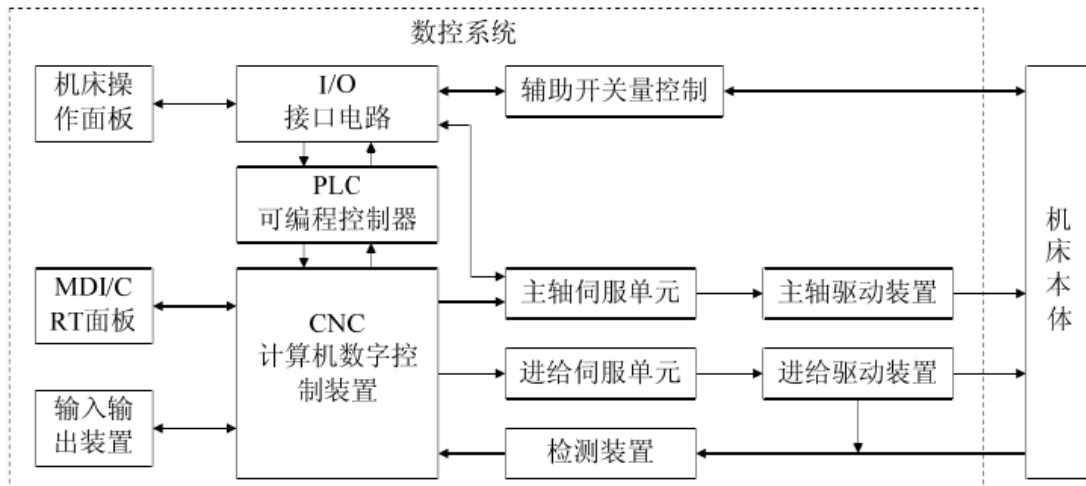


图 1-1 数控系统的组成

1.2.2 数控机床数据采集的原理

根据采集方式及采集数据的用途来划分，数控机床数据采集可以分为对两类数据的采集：第一类是机床内部储存的、表明生产加工现状及进度的状态数据，第二类是机床内部不储存的、表明机床外在运行状态的物理数据。

对于第一类数据来说，数据系统中 CNC 和 PLC 模块存储了数控机床内部的状态信息，如机床正在加工的程序名、机床主轴转速、机床刀具号、机床进给速率等，但一般不对外提供输出接口，需要采取一定的方法来获取这些信息。

对于第二类数据来说，由于这些数据在系统内部没有储存，例如刀具的切削力、刀具的温度、刀具的应力分布等，所以必须通过外接其他设备，如传感器等测量设备，搭好测量平台来测量和采集这些数据。

由于本文采集的数据主要提供给 MES 系统使用，以使管理系统做出更加高效的管理，所以本文数据采集研究仅限于第一类数据。

1.2.3 数控机床数据采集的任务

主要任务有两个：一是从数控机床内部或者外部采集数据，并且在本地或者服务器上对数据进行存储；二是把采集的数据以可视化的方式呈现出来，提供给机床管理者，或者把数据传递给更上层的管理系统，以为上层管理系统决策提供依据。

1.2.4 数控机床数据采集的方法

数控机床硬件和数控系统均在不断地发展，所以，针对不同年代生产的数控机床和不同的数控系统，其数据采集方法也有所不同。各种数据采集方法整体上可以分为两类：基于数控机床对外通信接口的数据采集和基于数控机床电气线路的数据采集。

一些比较新式的数控机床，往往会提供 DNC 接口，并且支持 OPC（OLE for Process Control）数据访问规范，对这类机床的数据采集，可以设计基于 OPC 的数据采集方案，通过标准的 DNC 接口，根据 OPC 数据访问规范编写相应的应用软件，即可实现数据采集。

此外,一些高档机床还提供了标准的以太网接口,也可以软件二次开发来实现数据采集。

对于老式数控机床来说,大多不具有 DNC 接口,也没有网络接口,而 PLC 系统也往往是封闭,即不对外提供 PLC 接口。由于对数控机床硬件改造比较困难,而且可能损坏昂贵的数控设备,所以可以研究机床的电气线路,在相关节点处布置外接线路,通过采集标志节点处的电平信号,来获取相对应的数据信息,实现数据采集功能。

此外,不管是老式机床还是新式机床,大多都具有 RS232 串口,所以通过串口来采集数控机床数据的方法也比较普遍。

1.2.5 数控机床数据采集技术的研究现状

1.2.5.1 国外研究现状

主流的数控系统是德国的 SIEMENS、日本的 FANUC 以及美国的 HAAS 等,所以国外对数控机床数据采集研究起步较早,研究也更加成熟。

美国上升(ASCENDANT TECHNOLOGIES)科技公司开发了用于机械加工车间的 extremeDNC 软件,此软件可应用于多种数控机床,并可以大多数主流数控系统进行通讯。extremeDNC 软件可以达到最高 230KBPS 的串口通信速率,并且可以同时连接多台数控机床。此外,extremeDNC 还可以通过以太网与服务器相连,实现数控文件的发送与接收,同时还能监视和远程控制数控设备,从而实现对数控设备的加工信息、位置信息以及其他辅助信息的采集,并可以在网上实时发布采集到的数据^[4]。

西门子开发的软件 WINPCIN,通过串口通信的方式,实现了数控机床参数数据、数控程序数据、误差补偿数据和偏置数据等数据的采集。

在现在的高档机床中,一些数控厂商提供的 DNC 软件中会自带数据采集功能,如 FAGOR 的 windnc 软件、DNC Work 公司的 NetWorked DNC 软件、日本 FANUC 公司的 DNC2 系统等。

巴西圣保罗大学的 Nunzio M.Torrisi 等人借助 OPC (OLE for Process Control) 技术进行了机床数据采集,并且实现了数控机床的远程监控,同时,利用 OPC 技术可以使通信周期相对更短^[5]。OPC 是一种工业标准,它的出现为基于 Windows 的应用程序和现场过程控制应用建立了桥梁。

巴西圣保罗大学的 F. Ferraz Jr 等人则通过利用因特网,并且安装 A.E Sensor 和 LVDT Sensor 等传感器,检测刀具与工件接触的声音信号,对具有开放式结构(Open Architecture)的数控机床实现了数据采集和远程监控,可以监测机床的运行状况以及刀具的磨损状况,并将相关加工、生产以及技术参数信息实时发布到网上以进行共享^[6]。

英国巴斯大学的 Sanjeev Kumar 等人则指出传统的 G、M 代码提供信息非常有限,而新兴的过程检查等数据采集方法则使传统的制造加工转向了网络化制造,并在文中提出了 STEP-NC 的概念,以为过程控制提供更加高层的信息,并且利用加工系统的自学习算法提出了一个 STEP 标准的框架^[7]。

1.2.5.2 国内研究现状

由于我国制造业的迅速发展以及制造自动化的迫切需求,国内对数控机床数据采集的研究也非常活跃。

上海交通大学金明冬等人采用 SIEMENS 公司的 5611 板卡,以机床 SIEMENS 840D 为例,通过 MPI 接口与 5611 板卡相连,并在操作终端安装自行开发的数据采集程序,实现了实时读取机床 SIMENS 840D 中的有关数据信息,并通过网卡将采集的数据信息传送并保存到服务器中,供 MES 和其他系统使用^[8]。

大连交通大学王姣等基于 ARM 处理器和嵌入式 Linux 系统,利用宏程序反馈的方式,实现了主轴转速、主轴进给速率、当前刀具号、加工程序号、开始加工时间和结束加工时间

等信息的采集^[9]。

南京航空航天大学的李波采用宏指令和特殊程序上报方法,研究了 FANUC、HAAS 和 SIEMENS 三种常用数控系统的信息采集方法,实现了对程序号、主轴转速、进给速率、机床开机时间、机床关机时间、零件加工开始时间、零件加工结束时间、加工零件数等信息的采集。

南京航空航天大学的徐永乐提出了一种基于 ARM9 和嵌入式 Linux 操作系统的数控机床信息采集系统方案,提出一种适用于嵌入式信息采集的存储与传输控制机制,并对 Linux2.6 内核环境下系统开发过程作了细致的探讨,开发了上位机管理软件,并进行了系统集成,实现了以开关量采集为主,兼顾模拟量和生产进度数据采集的生产现场数据采集系统^[10]。

重庆大学李文川基于 Fanuc 为用户提供的二次开发软件包 FOCAS (Faunc Open CNC Application Software),通过 Visual C++编程调用,实现了下载数控程序到 NC 机床、从机床上传 NC 程序到主机、在机床中删除 NC 程序、读取刀具位置、读取机床状态、从机床选择加工零件的 NC 程序、执行 NC 程序和机床复位等功能^[11]。

浙江大学利用 OPC 技术,通过对人机界面(HIM)软件的二次开发,针对装有 810D/840D 数控系统的数控机床,读取了其 NCK/PLC 中的数据^[12]。

1.2.6 数控机床数据采集技术的发展趋势

(1) 数控机床对外接口技术的研究

我国国内多数机床为老式的经济型机床,对外的接口只有 RS232 串行通讯接口,要实现有效数据的采集比较困难,采集成本也较高,并且也难以保证采集数据的实时性。所以,必须要研究和采用新一代通讯接口,如满足制造自动化协议(Manufacturing Automation Protocol, MAP)标准的新一代 DCN 接口,并且各机床制造商也会逐渐配套相应的数据采集软件。

(2) 数控机床通用数据采集方案的研究

由于一个企业往往会使用多种数控系统,而不同的数控系统提供的对外接口及通信协议也各不相同,这都导致了对于不同的数控系统必须设计不同的数据采集方案,即数控采集方案难以实现通用性。所以,今后将推动开放式数控系统的发展,逐步消除数控系统封闭与异构的弊端,建立统一的机床通信标准与开放的信息读取通道,使得数据采集方案标准化和通用化。

(3) 数控机床采集数据的深层挖掘

现阶段采集的数控机床数据多用于加工过程的监视,直接应用于生产调度和生产计划制定的还比较少。所以,有必要用采集的大量数据进行深层次的信息挖掘,从中提炼出更加有用的信息,并且应用于企业的管理和日常运转,将其转化为生产力的提高,从而增强企业的竞争力,使企业获得更好的经济效益。

1.3 嵌入式技术概述

1.3.1 嵌入式系统定义及特点

IEEE(国际电气和电子工程师协会)对嵌入式系统的定义:“用于控制、监视或者辅助操作机器和设备的装置”(原文为:Devices Used to Control, Monitor or Assist the Operation of Equipment, Machinery or Plants)。

嵌入式系统是以应用为中心,以计算机技术为基础,软件硬件可裁剪,适应应用系统,对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。

嵌入式系统的主要特点是^[13]:

- (1) 嵌入式系统面向用户、面向产品、面向应用，其处理器各方面均受到应用要求的制约；
- (2) 系统内核小、功耗低、实时性好；
- (3) 系统软硬件结合度高；
- (4) 具有多种通讯接口；
- (5) 嵌入式系统的软件一般具有较长生命周期；
- (6) 嵌入式处理器的发展也体现出稳定性。

1.3.2 数控机床嵌入式数据采集的优点

现有的数控机床数据采集系统大多基于工控 PC 机，占地面积大，成本高，并且仅用于数据采集会浪费相当的资源。而随着 DSP、ARM 技术的发展，基于嵌入式平台下的数据采集系统正得到越来越多的应用，嵌入式系统实时性强、可靠性高、体积小并且成本低，非常适合工业生产环境的应用，成为 PC 平台之外一个重要的选择方案。

如果采集嵌入式系统及嵌入式技术来进行数控机床的数据采集，将会实现小型化甚至微型化，并且有利于降低成本，适应各种工业环境。而且随着嵌入式技术的不断发展，嵌入式 CPU 和操作系统将获得更加强健的性能，其硬件和软件性能配置将会满足更多工业领域的应用要求，必将成为今后发展的一个趋势。

综合以上分析，本课题采用嵌入式技术进行数控机床的数据采集，即设计与开发数控机床数据采集嵌入式适配器。

1.4 课题的研究意义、研究目标及研究内容

1.4.1 课题的研究意义

本课题主要研究基于嵌入式适配器实现数控机床的数据采集，研究意义有如下几点：

- (1) 嵌入式适配器可以相对降低数控机床数据采集的成本，并实现小型化和智能化，而且可以提供良好的图形用户界面以方便用户操作，同时嵌入式适配器的功能易于扩展和二次开发。
- (2) 实现数控机床数据采集，有利于企业管理层掌握底层机床运行状态，从而做出更好的生产计划安排，从而提高机床利用率和企业的经济效益。
- (3) 机床数据反映机床的实现运行状况，可以作为考核车间工人的标准之一，提高工人的生产积极性。

1.4.2 课题的研究目标

针对传统的数控机床在远程通信及数据采集方面的不足，为满足制造系统信息化和 MES 系统的要求，本课题提出以下三个研究目标：

- (1) 基于无线 WIFI 网络实现数控机床嵌入式适配器和上位机的通信。
- (2) 基于嵌入式系统实现数控机床多种类型状态数据的采集。
- (3) 基于数据库技术实现对采集数据的存储、查询和初步处理等数据管理操作。

1.4.3 论文的主要内容

本论文主要研究了数控机床数据采集嵌入式适配器设计与开发，比较了各种数据采集方法，并详细阐述了利用宏指令实现机床串口数据采集的原理，实现了嵌入式适配器的硬件平台和软件系统的设计及开发，并对客户端数据管理系统进行了详细的设计和开发，从而完成了一个包括嵌入式适配器和服务器、客户端数据管理系统的数据采集系统的整体开发，最后进行了系统的测试和运行。本文的主要内容安排如下：

第 1 章分析了我国制造业发展和企业数控车间的现状，指出了数控系统数据采集的重要

性,并且概述了数控机床数据采集的原理和数据采集的研究现状以及发展趋势,接下来介绍了嵌入式技术的一些概念。最后针对我国企业车间的现状引出了课题的选题及主要研究内容。

第2章阐述了数控机床数据采集系统的总体设计。首先对数控机床数据采集系统进行了需求分析,并且对现有的数据采集方法进行了分析和比较,确定了基于串口的数控机床数据采集方法。在此基础上,设计了数控机床数据采集系统的整体结构(C/S架构),并选择了嵌入式适配器的处理器和操作系统,进而设计了采集方案以及数据库的结构。

第3章阐述了嵌入式适配器的具体实现方法。以OK6410开发板配合其他必需扩展模块作为硬件平台,进行了嵌入式操作系统Windows CE 6.0的订制和移植,并介绍了嵌入式数据库SQL Server Compact 3.5,进而对运行在嵌入式操作系统的信息采集应用软件进行了功能模块设计,讲解了数控系统的宏采集指令、串口通信技术和远程数据访问等关键技术,并阐述了系统关键模块的开发。最后在.NET Compact Framework平台基础上,利用C#语言进行了数据采集软件的实际开发。

第4章阐述了客户端管理系统的具体实现方法。首先对SQL Server 2008数据库做了简要介绍,并且说明了如何配置网络环境以实现远程数据访问。进而对运行在客户端的数据管理系统进行了功能模块设计,讲解了利用ADO.NET实现数据访问和数据绑定等关键技术,设计了系统的整体界面。接着详细阐述了系统关键模块的开发,并利用C#语言进行了数据管理系统的实际开发。最后对整个数据采集系统进行了通信功能和数据采集功能的测试,并介绍了嵌入式适配器和上位机客户端的运行。

第5章是总结与展望,对论文中已经完成的主要工作进行了总结,并且根据研究过程中出现的问题以及获得的经验,从系统的实时性、普遍适用性和数据深入挖掘等角度展望后续研究工作。

1.5 本章小结

本章结合我国推行制造业信息化以及MES系统的背景,阐述了数控机床数据采集的重要意义。在此基础上,介绍了数控机床数据采集的基本原理以及当前使用较多的数据采集方法,并预测了数据采集的发展趋势。接下来,介绍了嵌入式技术相关概念,并且阐明了利用嵌入式技术进行数控机床数据采集的优势,从而引出了本课题的选题背景,并说明了本课题的研究目标。最后,介绍了本论文的各章主要内容。

第2章 基于嵌入式适配器的 数控机床数据采集系统的总体设计

2.1 数据采集系统的功能需求分析

基于嵌入式适配器的数据采集系统最基本的功能是数据采集功能,同时还要对采集到的数据进行储存,并且允许用户查询数据以及修改、增删数据等。数据采集系统还应该对数控程序进行管理,例如存储、查询数控程序以及传输数控程序等。在企业车间中使用时,数据采集系统还应具有对数据库中的车间和设备信息进行管理的功能。具体功能需求分析介绍如下:

2.1.1 数据采集、储存、查询和统计处理功能

数据采集功能主要负责实现采集数控机床的状态信息和零件加工信息。其中,状态信息有机床开机时间、机床关机时间、程序加工开始时间、程序加工结束时间、切削开始时间、切削结束时间、主轴转速、进给速率、机床坐标、刀具号等;零件加工信息主要包括零件号、程序号、零件已加工数量、设定的零件总加工数量、零件加工时间等。

数据储存功能则负责将采集到的数据按照一定规则和格式储存到数据库中,以便后期的查询和处理。

数据查询功能则可以根据用户指定的条件,如采集时间的范围,来查询并显示出用户想要的数据库。

数据统计处理功能则属于较为上层的功能,主要由客户端完成,负责对采集到的数量巨大的数据按照一定公式进行处理,从而得出对企业车间生产有一定参考意义的结论,比如机床利用率、机床待机时间比率等,并以数字、表格或者图形的形式来显示这些结果,从而给企业的上层管理者提供决策和制定生产计划的依据。

2.1.2 数控程序的存储、管理和传输功能

基于嵌入式适配器的数据采集系统应该实现数控程序的双向传输,即由机床到嵌入式适配器和由嵌入式适配器到机床传输数控程序。数控程序存储在嵌入式适配器数据库中,用户可以直接选择所需的数控程序,并将数控程序传输到数控机床上以进行加工。

数控程序的管理功能则需要实现对数据库中数控程序的管理,如对数据库中的数控程序进行查询、修改、删除和增加等操作,并可以实现数据库中数控程序和本地数控程序文件的相互转化。

2.1.3 车间设备及用户信息管理功能

企业中一般有多个制造车间,而每个车间中也放置多台数控机床,并且这些数控机床可能具有不同的型号、设备名称和数控系统等信息,这些信息都储存在数据库中。基于嵌入式适配器的数据采集系统应该提供对这些信息的管理功能,如更改车间、设备名称,或者根据实际情况变动对数据库中的车间和设备信息进行更新,如删除或者增加相应的车间、设备信息等。

作为一个完备的数据采集系统,应该具有一定的安全措施,即需要用户登录以对用户进行认证,并且对用户设置不同的权限,不同权限的用户可以执行不同的操作。

2.2 数据采集方法的选择

在绪论中，对现在常见的多种数控机床数据采集方法做了简要介绍，主要分为两类：即通过机床通信接口采集和通过外接机床电气电路进行采集。下面将对数控机床的数据采集方法进行更加详细的说明和比较。

2.2.1 各数据采集方法的优缺点及适用范围

由于数控机床型号、生产年代及数控系统不同，所以对不同的数控机床需要采用不同的数据采集方法。表 2-1 阐述了常用的数控机床数据采集方法的基本操作及其适用范围：

表 2-1 数控机床数据采集方法的比较

采集方法	操作	优点	缺点
数据采集卡采集	在机床电气线路设置采集点，引出电压信号进行数据采集。	适用各种机床，包括无对外接口的老机床。	数据采集卡采集的信号数量有限，必须依赖 PC 机，成本高。
外接 PLC 数据采集	采集机床 PLC 系统中储存的状态参数	若有 PLC 对外接口，则操作简单，可采集数据多	多数机床并不对外提供 PLC 读取端口，资料缺乏，难以找到所需的运行参数。
MAP 接口、DNC 接口数据采集	利用机床对外接口与采集主机通信互联	操作简单，具有很强的实时性	只有新式高档机床才有此接口，不适用于老旧机床
工况数据采集器	购买此硬件设备使用即可	可自行定义采集项	成本高，实时性差，需要人员参与
机床串口采集	利用宏输出指令通过串口输出机床状态数据并采集	操作简单，成本低，扩展性强，可实现自动采集	可采集数据的多少取决于数控系统的开放性
利用机床自带的集成管理软件	机床生产提供的软件具有数据采集功能	简单易用	部分高档机床才提供此类管理软件，且成本很高

分析上表可以看出：

- (1) 直接通过数控机床的对外接口进行数据采集是最方便、高效的，但这种方法仅适用于新式高档机床，而我们大多机床是上世纪八九十年代的产品，所以并不适用。
- (2) 通过外接电气线路，进而用数据采集卡采集，是一种适用于各种机床的普通方法，但必须对数控机床的线路有充分了解，需要厂家提供资料和协助，而且数据采集卡采集的信号数量有限，成本较高。
- (3) 通过数控机床串口进行数据采集，也是一种适用于绝大多数机床的普遍方法，因为现在使用的机床不管是什么档次，几乎都配备了 RS232 串口，并且此方法简单、易于实施。在下一节中将对此方法进行详细介绍。

2.2.2 基于串口的数控机床数据采集方法

从上一节的比较中可以看出，利用数控机床串口结合宏输出指令进行数据采集是一种较为通用的方法。数控机床中通常使用的串口有 RS232、RS422、RS485 等串行接口，其中最常用的为 RS232 串行接口。

上海交通大学工程训练中心内配备有较多数控机床，现对中心内的数控机床状况进行了相关调查，结果如表 2-2 所示：

表 2-2 上海交通大学工程训练中心数控机床状况统计

型号	数量	数控系统	通讯接口
法那科	18	FANUC Oi Mate-MC	RS232 串行接口
凯恩帝 100	16	KND CNC Series 100M	RS232 串行接口
NC 程序传输方式	有线局域网	PLC 系统对外状况	封闭

从表中数据可见，上海交大工程训练中心的数控机床均具有 RS232 串行接口，而机床的 PLC 系统是对外封闭的。

同时查阅资料得知，在我国企业制造车间内，FANUC、HAAS 和 KND 系统数控系统占据很大份额，所以本课题采用基于数控机床串口的数据采集方法。

2.2.2.1 RS232 串行接口简介

RS-232-C 接口（又称 EIA RS-232-C）在各种现代化自动控制装置上应用十分广泛，是目前最常用的一种串行通讯接口。它是在 1970 年由美国电子工业协会（EIA）联合贝尔系统、调制解调器厂家及计算机终端生产厂家共同制定的用于串行通讯的标准。该标准规定采用一个 25 个脚的 DB25 连接器，对连接器的每个引脚的信号内容加以规定，还对各种信号的电平加以规定，一般只使用 3~9 根引线。

2.2.2.2 基于串口的数据采集的基本原理

FANUC 和 HAAS 等数控系统均支持包含宏指令的数控程序，而在其宏指令中有一类特殊的指令用于从串口输出指定的数据。利用嵌入式适配器的硬件及软件平台与数控机床进行串行通信，进而采集数控机床串口输出的数据并进行储存，从而可以通过此方法实现 FANUC、HAAS 及其他多数支持宏输出指令的数控机床的数据采集。如图 2-1 所示。

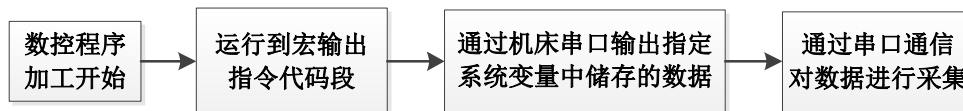


图 2-1 基于串口的数据采集的基本原理

2.3 基于嵌入式适配器的数据采集系统的总体架构

2.3.1 数据采集系统总体结构设计

本数据采集系统使用基于机床串口和宏输出指令的数据采集方法，采用包括嵌入式适配器、服务器和远程客户端在内的系统架构，即客户机/服务器的 C/S（Client/Server）架构。嵌入式适配器主要负责数据的采集，并将采集到的数据上传到服务器，客户端数据管理系统应用软件主要负责数据的管理及处理，总体架构模型图如图 2-2 所示。

在通信方式上，数控机床与嵌入式适配器通过串口进行串行通信，以完成数据采集工作，并将采集到的数据暂时储存到嵌入式适配器的数据库中。在合适的时候，各嵌入式适配器与上位机服务器通过无线 Wifi 网络进行通信，并将采集到的数据上传到服务器的数据库中。服务器数据库接收所有嵌入式适配器上传来的数据，并对这些数据进行统一的储存。而远程客户端则可以通过网络连接到服务器，利用客户端的数据管理系统对服务器数据库中的数据进行查询、修改、增添、删除等管理操作，而且可以对大量数据进行统计和分析，从而得出反映机床效率的相关参数，以供企业管理层参考。

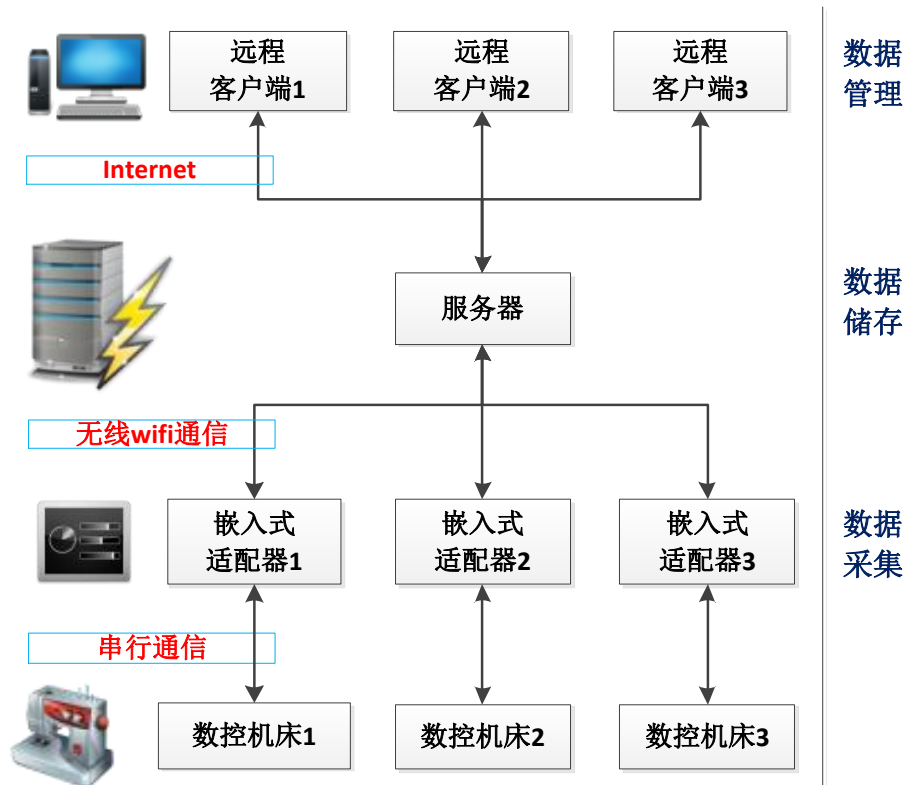


图 2-2 数据采集系统的总体架构

2.3.2 嵌入式适配器处理器的选择

嵌入式适配器处理器的选择需要考虑处理器的运算能力、外围资源是否满足应用要求、是否满足处理器自身要求的工作环境、处理器工作稳定性、成本以及开发的难易程度等因素。目前常见的嵌入式处理器有单片机、数字信号微处理器 DSP (Digital Signal Processor) 和 ARM (Advanced RISC Machines) 处理器。三者的比较如表 2-3 所示^[14]：

表 2-3 嵌入式适配器处理器的比较

处理器	适用领域	成本	是否支持操作系统	开发难易
单片机	适用于工业上简单的测控领域	低	不支持	功能简单，开发较容易
DSP	适用于数字信号处理，如 FFT、数字滤波	较高	不支持	功能较复杂，开发较困难
ARM	X86 体系的通用处理器，常用于智能终端	较高	支持	功能复杂，开发较困难

ARM 处理器是英国 ARM 公司推出的著名产品。ARM 微处理器体系结构目前被公认为是嵌入式应用领域领先的 32 位嵌入式 RISC 微处理器结构。目前，ARM 微处理器主要包括 ARM7、ARM9、ARM9E、ARM10、ARM11、SecurCore，以及 Inter 的 StrongARM/Xscale 等系列，其中 ARM9 系列相比于 ARM7 具有重要的改进。例如 ARM9 时钟频率提高，至少在 200MHz 以上，这意味着其处理速度更快，并且具有 MMU(Memory Management Unit，内存管理单元)，意味着 ARM9 可以移植任何嵌入式操作系统^[15]。

由于本课题研究的数据采集系统需要运行数据采集应用程序，并且要求有良好的图形用户界面，由上表分析结果可知，本课题应该采用 ARM 嵌入式处理器。

2.3.3 嵌入式适配器硬件平台设计

根据数据采集系统的功能需求，要实现数据采集，嵌入式适配器硬件平台需要对外有串口、无线接口等通信接口，嵌入式适配器的硬件平台整体框架如图 2-3 所示：

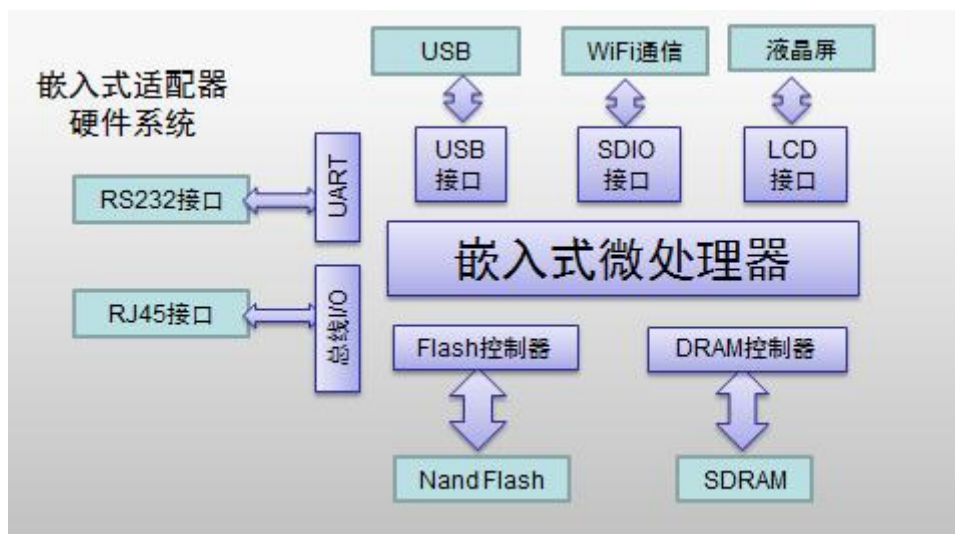


图 2-3 嵌入式适配器硬件平台整体框架

在嵌入式适配器硬件平台的设计中，主要对外接口或者模块，如 RS232 接口、RJ45 接口、USB 接口、SDIO 接口以及 Nand Flash 等的作用如表 2-4 所示：

表 2-4 嵌入式适配器主要接口、模块介绍

接口或者模块	功能
RS232 接口	串口，实现与数控机床的串行通信，是数据采集的基本通道
RJ45 接口	以太网接口，可实现有线网络通信，以及开发过程中文件传输
USB 接口	可外接 U 盘，实现文件数据的传输
SDIO 接口	安全数字输入输出接口，用于外接 Wifi 网卡
LCD 接口	外接液晶屏，实现良好的图形用户界面
SDRAM	同步动态随机存储器，即内存，新一代存储器，存储容易大，读写速度快
Nand Flash	Flash 存储器的一种，读写速度快，适用于大量数据的存储，可用于系统中大量采集数据的储存

根据数据采集系统的功能需求及核心硬件平台对外提供的接口，可知，所需的主要外围扩展器件如表 2-5 所示：

表 2-5 嵌入式适配器需要的扩展器件介绍

扩展器件	作用
NandFlash 存储卡	实现数据的存储，在考虑成本的基础上，容量应该尽可能大。
带触摸功能的液晶屏	实现应用程序图形用户界面的显示，由于嵌入式适配器不适合配备键盘，所以提供触摸功能支持用户的输入和交互。
Wifi 无线网卡	实现嵌入式适配器和上位机服务器的无线通信。
RS232 串行通信 9 针、25 针线 缆	实现嵌入式适配器和数控机床的串行通信
以太网网线	可在开发过程中与开发机进行文件交互，也可以实现嵌入式适配器的有线上网。

2.3.4 嵌入式操作系统的选择

2.3.4.1 主流嵌入式操作系统比较

当今主流的嵌入式操作系统主要有以下三个：嵌入式 Linux/uClinux、Vxworks 和 Windows CE。关于这三个嵌入式操作系统的详细比较如表 2-6 所示：

表 2-6 三种主流嵌入式操作系统的比较^[16]

嵌入式操作系统	可裁剪性	硬件兼容	通信支持	开发环境	实时性	数据库支持
Windows Embedded CE	可裁剪相对低，内核相对较大	好	支持多种通信方式，可以方便的与桌面通信	类似桌面 Windows 操作系统开发环境，容易上手	高	支持 SQL Server Compact 等
嵌入式 Linux	可裁剪性高，内核小	中	支持多种通信方式	要采用 Linux 的开发环境，开发难度较大	高	支持 MySQL、SQLite 等
VxWorks	可裁剪性高，内核小	中	支持多种通信方式	开发资料少，开发难度较大	很高	支持 Express 嵌入式数据库

2.3.4.2 Windows CE 嵌入式操作系统简介

微软公司于 1996 年 11 月发布了公司的第一个嵌入式操作系统：Windows CE 1.0，2001 年初，微软发布了 Windows CE.NET 4.0，开始了嵌入式操作系统对 .NET Compact Framework 的支持；2006 年 11 月，微软公司发布了 Windows Embedded CE6.0，在这个版本中，核心源代码被 100% 完全开放，性能得到了很大改善，如支持同时 32000 个并发进程，单个进程最大虚拟内存 2GB，提供实时数据链接，增强了对多媒体的支持等^[17]。

Windows CE 主要有以下一些特点：

- (1) 高可靠性和高稳定性。Windows CE 相对于桌面 Windows 版本要稳定得多。
- (2) 硬件良好兼容性，这是微软公司产品的一贯优良传统。
- (3) 支持多种通信。
- (4) 数据库支持。有着 SQL Server 的专门支持，并且支持 ADO.NET。
- (5) 安全性较高，高级电源管理等。
- (6) 容易上手的开发环境，其开发集成到了 Microsoft Visual Studio 之中。

鉴于 Windows CE 的以上优点，考虑到系统的兼容性及开发难易程序，以及系统的功能改进和功能扩展等方面，本课题选用 Windows CE 6.0 嵌入式操作系统。

2.3.5 数据库结构设计

数据库系统是数据采集系统的重要组成部分，承担着数据存储和数据通信的工作。根据企业车间日常管理、运行的实际需要，并且结合数据采集系统自身的功能需求，设计了数据库中相应的表结构，将采集到的机床数据、数控程序、车间信息、设备信息、员工信息、用户信息等都储存到数据库中，并在系统应用程序界面中提供对这些数据的查询、修改、增加、删除和进一步处理等操作。

上位机服务器后台数据库采用微软公司的 SQL Server 2008，嵌入式适配器后台数据库采用适用于嵌入式操作系统的 SQL Server Compact 3.5。根据实际需要，设计了各个数据表，并设置了数据表的主键以及各数据表的外键关系。

图 2-4 和图 2-5 是本系统数据库的采集模块物理模型图和管理模块物理模型图（其中

PK、FK 分别表示主键和外键)。

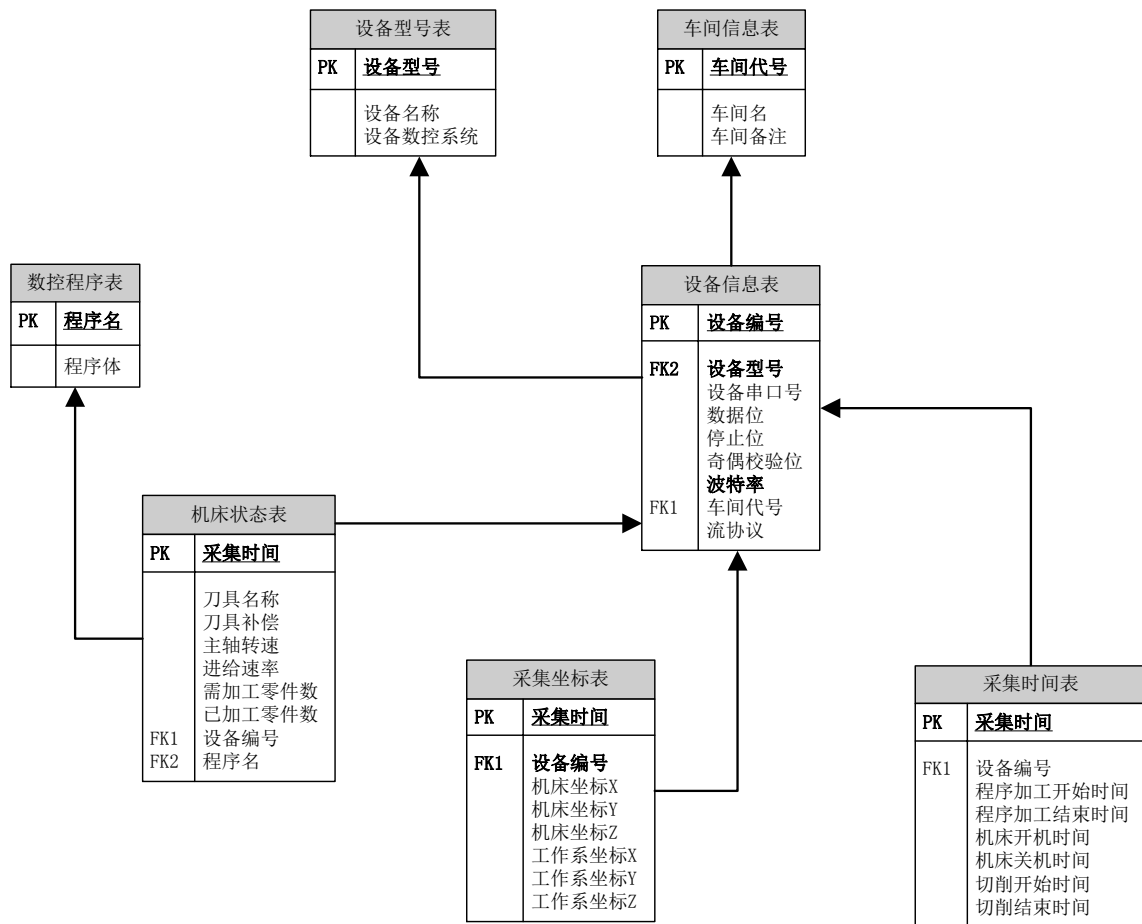


图 2-4 数据库采集模块物理模型图

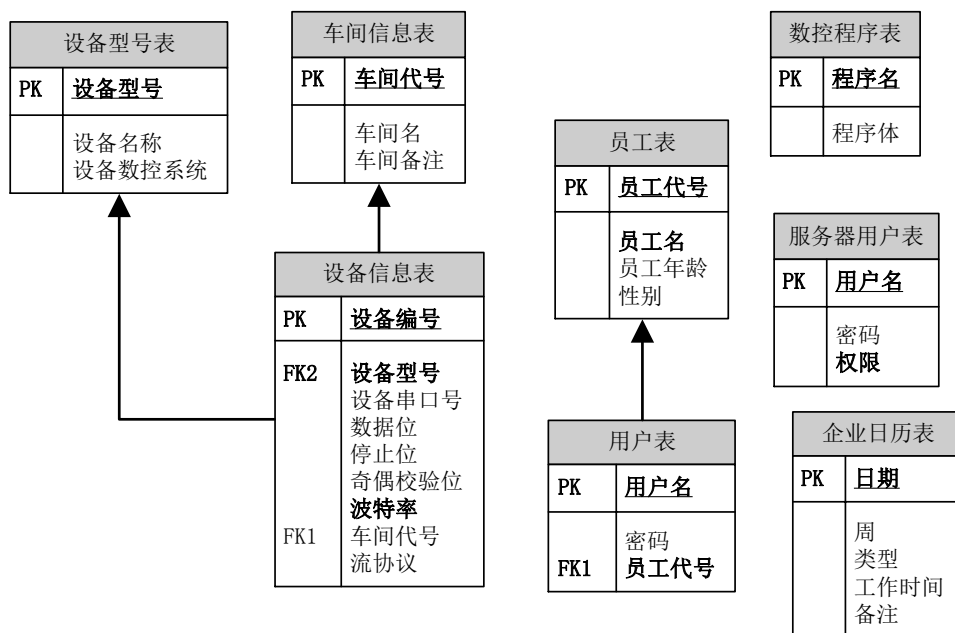


图 2-5 数据库管理模块物理模型图

2.1 本章小结

本章首先对数控机床数据采集系统进行了功能需求分析，讲解了系统所需的各项功能。进而在比较各种数据采集方法优缺点和适用范围的基础上，确定了本课题的数据采集方法。然后对基于嵌入式适配器的数据采集系统进行了总体架构设计，并设计了嵌入式适配器的硬件平台，选择了嵌入式处理器和嵌入式操作系统。最后，根据实际需要，对系统所需的数据库进行了结构设计。

第3章 嵌入式适配器的实现

3.1 嵌入式适配器的硬件平台

在上一章系统设计中，确定了嵌入式适配器采用 ARM11 处理器，并设计了嵌入式适配器的平台平台。根据设计要求，嵌入式适配器具体采用 ARM11 芯片 S3C6410，并以飞凌 OK6410 开发板为原型，实现了嵌入式适配器的硬件平台。

3.1.1 嵌入式处理器 S3C6410

与先前普遍使用的 ARM9 处理器的 5 级流水线相比，ARM11 拥有一条具有独立的 load-store 和算术流水的 8 级流水线，在同样工艺下，ARM11 处理器的性能与 ARM9 相比大约提高了 40%。ARM11 执行 ARMv6 架构的指令，ARMv6 指令包含了针对媒体处理的单指令流多数据流（SIMD）扩展，采用特殊的设计，以改善视频处理性能。为了能够进行快速浮点运算，ARM11 增加了向量浮点单元。所有这些结构上的提高，都是 ARM9 处理器不可比拟的^[18]。

S3C6410 是由韩国三星公司推出的一款低功耗、高性价比的 RSIC 处理器，它基于 ARM11 内核 (ARM1176JZF-S)，可广泛应用于移动电话和通用处理等领域；S3C6410 为 2.5G 和 3G 通信服务提供了优化的硬件性能，内置强大的硬件加速器：包括运动视频处理、音频处理、2D 加速、显示处理和缩放等；同时，S3C6410 包含了优化的外部存储器接口，该接口能满足在高端通信服务中的数据带宽要求^[19]。由于以上突出的性能表现，著名的苹果公司手机 IPHONE 就是基于 S3C6410 处理器。

3.1.2 飞凌 OK6410 开发板

本课题基于河北保定飞凌嵌入式技术有限公司的 OK6410 开发板进行硬件平台的设计和开发。开发板外观如图 所示，图 3-1 中标出了开发板了主要对外接口。

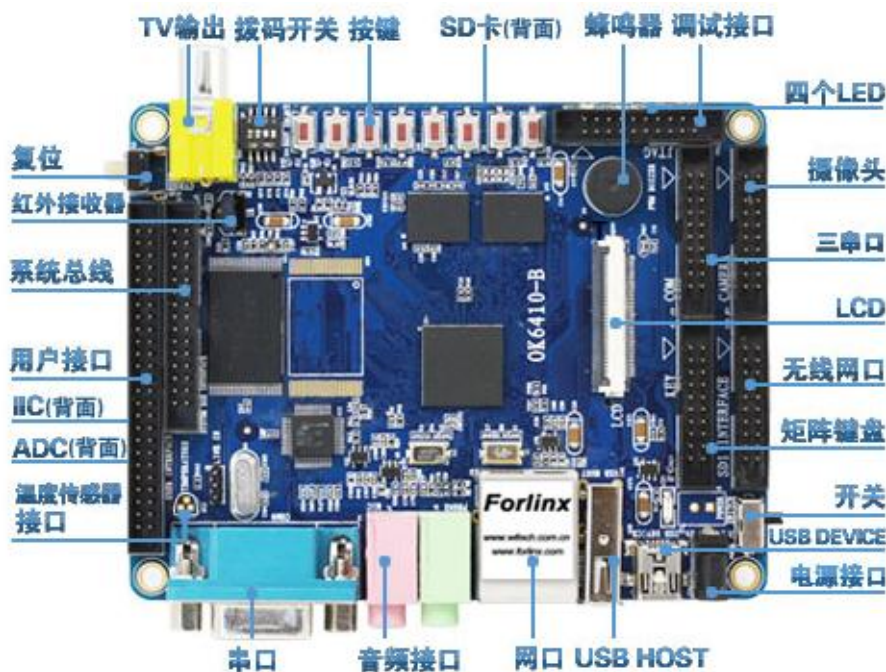


图 3-1 OK6410-B 型开发板

OK6410 开发板基于三星公司最新的 ARM11 处理器 S3C6410，拥有强大的内部资源和视频处理能力，可稳定运行在 667MHz 主频以上，支持 Mobile DDR 和多种 NAND Flash。OK6410 开发板上集成了多种高端接口，如复合视频信号、摄像头、USB、SD 卡、液晶屏、以太网，并配备温度传感器和红外接收头等。OK6410 开发板主要硬件配置、对外接口设计如表 3-1 所示。

表 3-1 OK6410 开发板主要硬件资源及对外接口介绍

硬件资源和对外接口	说明
S3C6410 处理器	主频 553MHz/667MHz
Mobile DDR 内存	256M
Nand Flash	2G
4 个串口	1 个 RS232 串口、3 个 TTL 串口
1 个 100M 网口	采用 DM9000AE
1 个 USB HOST 接口	可外接 U 盘、鼠标、键盘等
无线 WIFI 接口	可外接 WIFI 无线网卡
触摸板接口	支持 4 线电阻式触摸板
LCD 接口	支持 3.5 寸、4.3 寸、5.6 寸等 TFT LCD
3 个“10*2”插针扩展口	可扩展矩阵键盘、AD/DA 转换和 TTL 串口等

本课题的嵌入式适配器主要用于串行通信以采集数据，需要 RS232 串口；嵌入式适配器与上位机服务器进行无线通信，需要无线 WIFI 接口；嵌入式操作的移植、开发过程中文件的传输需要 U 盘或者网线等，需要网口和 USB HOST 接口。飞凌 OK6410 开发板提供的对外接口及处理器满足课题需要。

另外，需要购置串口扩展板以扩展串口，需要 USB 转串口线缆以实现开发过程中嵌入式适配器和笔记本电脑的文件同步，需要 DB9、DB25 串口通信接口线以实现嵌入式适配器与数控机床、嵌入式适配器与台式电脑的串行通信（开发过程中使用）。

3.2 嵌入式操作系统 Windows CE 的定制与移植

3.2.1 Windows CE 的体系结构

在进行嵌入式操作系统 Windows CE 的定制和移植前，需要了解 Windows CE 的体系结构，以便在定制操作系统时准确选择所需的模块。Windows CE 的体系结构如图 3-2 所示^[20]。

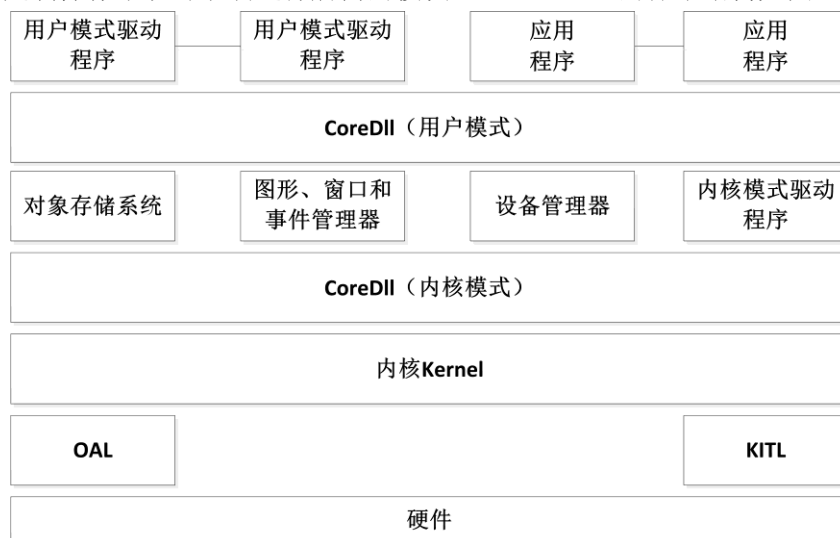


图 3-2 Windows CE 体系结构

Windows CE 采用了典型的分层结构，各层的作用简介如表 3-2 所示。

表 3-2 Windows CE 体系中各层的作用

层	作用
硬件	一般为 X86 体系结构的 IBM-PC 及其兼容机，为解决硬件，板载支持包 BSP 负责解决不同硬件的差异，以支持操作系统。
OAL	OEM 适配层，提供中断处理、电源管理、时钟、通用 I/O 服务等服务。
KITL	内核独立传输层，支持不同类型的通信。
内核 Kernel	提供最基本的底层功能，如进程、线程、内存管理。
CoreDLL	提供 Core OS 服务，使应用程序可以访问系统的计算资源，如文件系统和内存等。
文件系统	支持文件系统及存储对象的管理
图形、窗口和事件管理器	提供图形界面和用户接口
设备管理器	配合注册表管理资源
驱动程序	管理硬件设备

3.2.2 Windows CE 的定制和移植

与桌面 Windows 操作系统的标准系统（如 Windows XP、Windows 7 等）不同，嵌入式操作系统运行的硬件环境资源相对少得多，所以一般要对系统进行精简，只保留需要的模块，以使系统所占空间和所需的资源最少。

Windows CE 嵌入式系统的定制过程如图 3-3 所示。

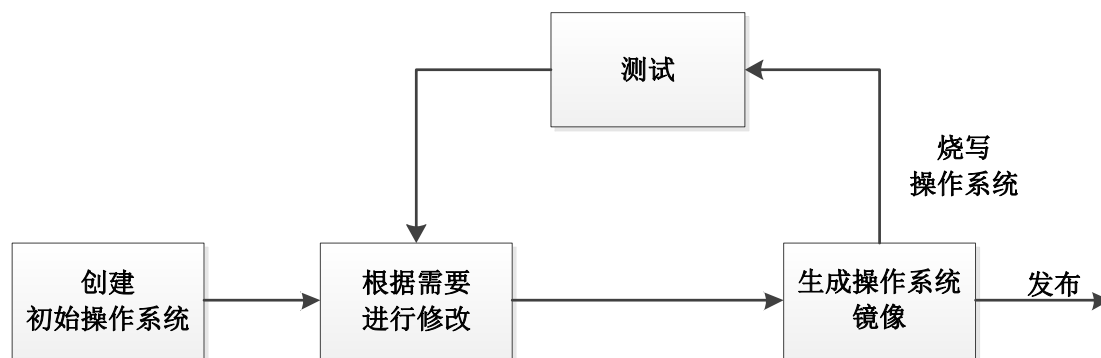


图 3-3 Windows CE 的定制

定制和移植 Windows CE 操作系统所需的工具是微软推出的 Platform Builder for CE6.0，它是集成到微软 Visual Studio 2005 中的一个插件。使用此工具定制 Windows CE 操作系统的详细流程如下：首先使用 Platform Builder for CE6.0 的操作系统创建向导创建一个操作系统，然后按照实际需要对其中进行一些组件的添加或者删除操作，修改完成后，就可以生成操作系统（OS）映像了。接下来，将 OS 映像下载到目标设备上进行测试，并反馈结果。若不满意结果，就对先前创建的操作系统进行进一步修改，并再次测试反馈。

3.2.3 嵌入式数据库 SQL Server Compact 3.5

由于数据采集系统需要数据库的支持，在嵌入式操作系统 Windows CE 定制和移植成功后，还需要在嵌入式适配器平台上安装嵌入式数据库。本系统选用了微软公司针对 Windows CE 推出的嵌入式数据库 SQL Server Compact 3.5。

SQL Server Compact 是一种功能强大的轻型关系数据库引擎，通过支持熟悉的结构化查询语言（SQL）语法以及提供与 SQL Server 一致的开发模型和 API 使得开发桌面应用程序变得非常容易，专为需要轻型、进程内关系数据库解决方案以便在台式机和移动设备上开

发应用程序并在其上进行部署的开发人员而设计^[21]。就本课题而言，主要用到 SQL Server Compact 的特色功能如下：压缩的数据库引擎和强大的查询优化器、支持合并复制与远程数据访问、支持 .NET Compact Framework 数据访问接口、支持 ADO.NET 和 SQL 语法的子集等。

SQL Server Compact 3.5 的安装比较简单，首先在开发机上安装 SQL Server 2008（或者 SQL Server 2005），在安装目录下找到 sqlce.wce5.armv4i、sqlce.repl.wce5.armv4i、sqlce.dev.CHS.wce5.armv4i 这三个文件，然后复制到嵌入式适配器上安装即可。

3.3 嵌入式适配器数据采集应用软件的功能模块设计

3.3.1 数据采集应用软件各功能模块的划分

根据数据采集系统的功能需求分析和整体 C/S 架构的要求，嵌入式适配器的数据采集应用软件功能模块划分如图 3-4 所示。

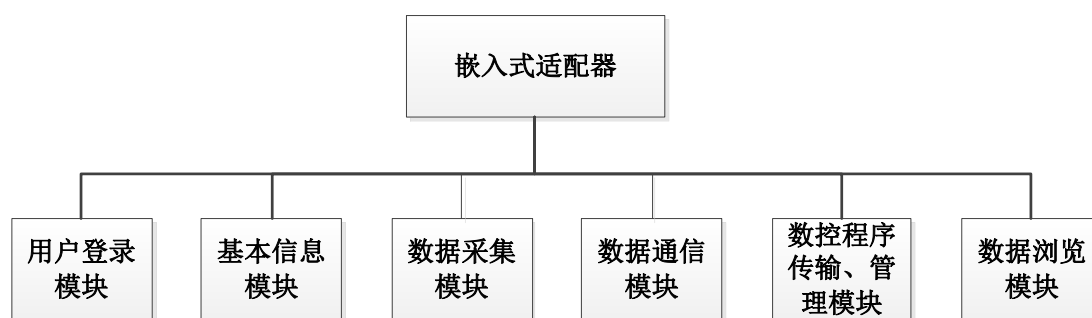


图 3-4 嵌入式适配器数据采集应用软件功能模型图

3.3.2 数据采集应用软件各功能模块的主要功能

3.3.2.1 用户登录模块

主要用户登录功能，以限制软件的使用权限。要求用户输入用户名和密码，并与储存在数据库中的用户信息进行核对，以判断用户是否可以成功登录。

3.3.2.2 基本信息模块

当用户登录进嵌入式数据采集软件系统时，当从数据库读取此用户的相关信息，并以列表的形式显示出来。由于整个数据采集系统是 C/S 架构，所以用户需要选择此嵌入式适配器所连接数控机床的车间及设备编号，以便向服务器上传数据时包含车间、设备信息，从而区分从不同数控机床所采集的数据。为方便用户，用户可以选择保存车间及设备信息，从而不用每次都选择。当用户保存好车间及设备信息后，下次登录时，基本信息窗口将展示出所在车间及设备的详细信息，供用户查看及确认。基本信息模块功能模型如图 3-5 所示。

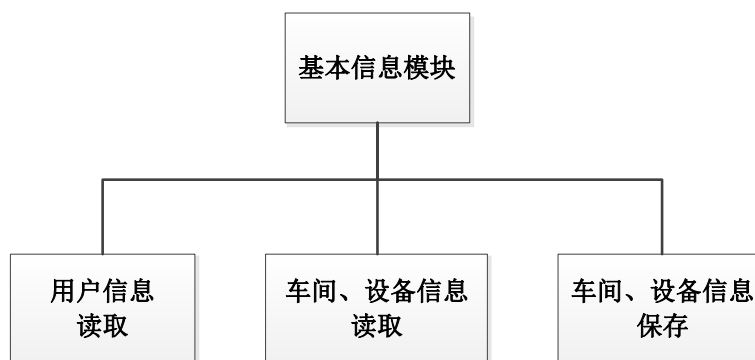


图 3-5 基本信息模块功能模型图

3.3.2.3 数据采集模块

本系统采用基于数控机床串口及宏输出指令的数控机床数据采集方法，数据采集模块主要负责与数控机床进行串口通信以采集数据。不同的机床具有不同的串口参数，所以本模块可以设置串口通信参数，如波特率、数据位、停止位、握手协议等。

另外，由于通过宏输出指令不能从机床串口输出开机和关机时间，所以机床开机、关机时间需要人工输入。本模块还提供了控制数据采集的功能，即开始数据采集和停止数据采集。数据采集模块功能模型如图 3-6 所示。

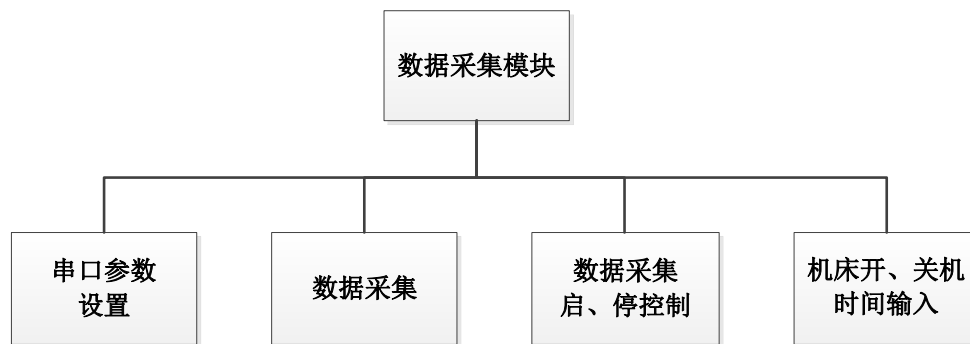


图 3-6 数据采集模块功能模型图

3.3.2.4 数据通信模块

数据通信模块主要负责嵌入式适配器与上位机服务器的通信工作。本数据采集系统利用 SQL Server 数据库的远程数据访问功能（RDA）来实现嵌入式适配器与上位机服务器的数据通信，即可以把嵌入式适配器新采集的机床数据上传到服务器数据库中，也可以从服务器数据库中下载用户、员工、设备、车间、数控程序等数据信息。关于 RDA 功能，将在后面的关键技术中进行介绍。数据通信模块需要用户输入服务器 IP、服务器虚拟目录、服务器数据库用户名和密码。数据通信模块功能模型如图 3-7 所示。

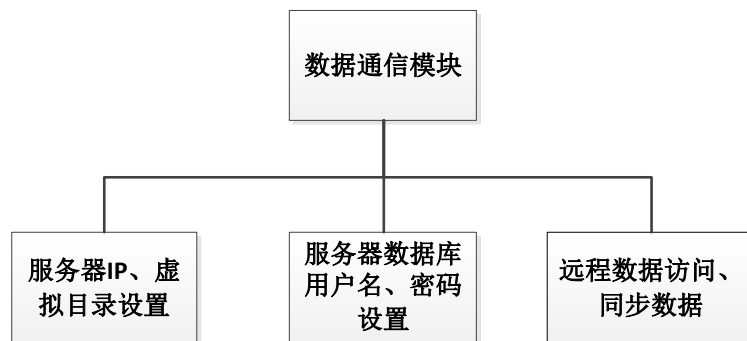


图 3-7 数据通信模块功能模型图

3.3.2.5 数控程序管理、传输模块

本模块主要实现从嵌入式适配器本地数据库中读取指定数控程序、浏览数控程序、编辑数控程序并下载数控程序到数控机床中。由于本系统采用宏指令来采集数据，为方便用户，本模块还负责自动添加宏采集指令，即用户只需选择需要采集的数据，如程序加工开始时间、切削时间、刀具号等，本模块会自动添加相应的宏采集指令到数控程序中，而不需要用户手工添加。数控程序管理、传输模块功能模型如图 3-8 所示。

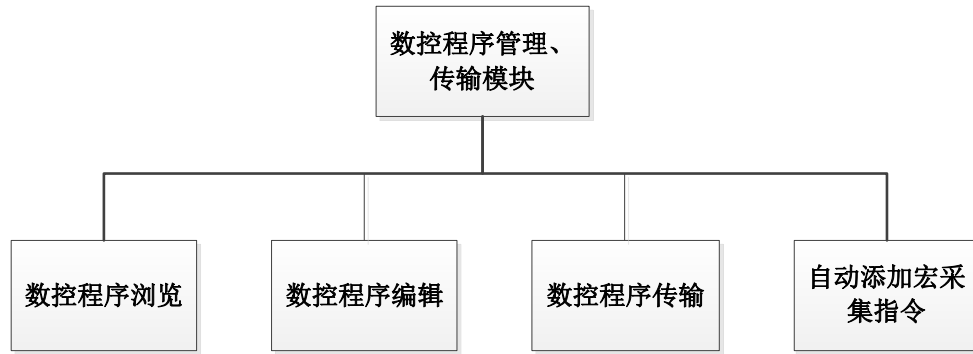


图 3-8 数控程序管理、传输模块功能模型图

3.3.2.6 数据浏览模块

此模块主要负责用户对嵌入式适配器本地数据库中相关数据的查询，可以查询之前采集到的机床数据，如采集时间表、采集坐标表、机床状态表，也可以查看企业所有的车间、设备信息。为保证数据的有效性和数据的安全性，本模块只提供查询功能，不允许用户在嵌入式适配器上对这些数据进行修改。

3.4 嵌入式适配器数据采集应用软件的开发

3.4.1 Windows CE 嵌入式开发环境的搭建

Windows CE 的开发包括两大方面，一是操作系统开发，二是应用程序开发。本课题主要进行应用程序的开发。要搭建 Windows CE 的开发环境，包括硬件平台和软件平台的搭建。硬件平台包括目标设备平台（即本文中使用的 OK6410 开发板）和开发平台（即通常使用的桌面计算机）。软件平台主要包括 Visual Studio 和 Platform Builder，前者主要用于开发应用程序，后者主要用于定制嵌入式操作系统。Windows CE 嵌入式开发环境所需资源如图 3-9 所示。

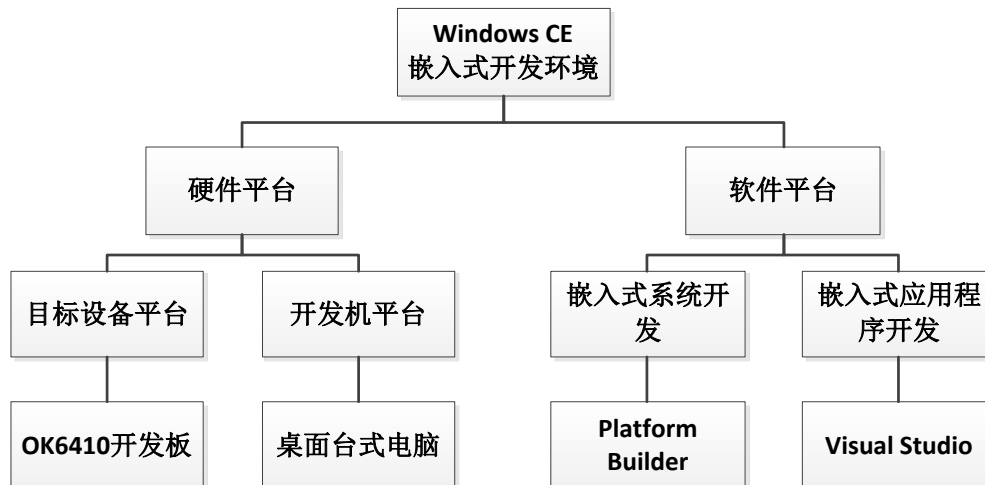


图 3-9 Windows CE 开发环境的搭建

3.4.2 数据采集应用软件的开发语言

Windows CE 嵌入式操作系统支持 .Net 框架，为提高开发效率和增强系统性能，采用了基于 .NET 平台的 C# 程序开发语言。

NET Framework 又称 .Net 框架。是由微软开发，一个致力于敏捷软件开发（Agile software development）、快速应用开发（Rapid application development）、平台无关性和网络透明化的软件开发平台。.NET 平台包含许多有助于互联网和内部网应用迅捷开发的技术，

是微软公司继 Windows DNA 之后的新开发平台。使用 .NET 来开发应用程序的好处是，一是支持跨平台，不必为每一种设备都编写一个不同的版本程序，二是提供大量类的支持，提高了开发效率^[22]。

C# (C Sharp) 是微软 (Microsoft) 为 .NET Framework 量身订做的程序语言，微软公司在 2000 年 6 月发布的一种新的编程语言。C# 拥有 C/C++ 的强大功能以及 Visual Basic 简易使用的特性，是一种最新的、面向对象的编程语言。它使得程序员可以快速地编写各种基于 .NET 平台的应用程序^[23]。

由于 Windows CE 操作系统支持 .NET Compact Framework，为提高软件开发效率，本数据采集系统使用 C# 语言进行编写。

3.4.3 数据采集应用软件的关键技术

在嵌入式适配器数据采集应用软件的开发过程中，需要实现现在数控程序中镶嵌宏采集指令的功能，在利用串口进行数据采集时需要用到串行通信技术和多线程技术，而在嵌入式适配器与服务器的无线通信中则需要数据库远程数据访问技术。下面将阐述这些关键技术。

3.4.3.1 宏采集指令

本数据采集系统通过在原有的数控程序中镶嵌宏采集指令来进行数据采集，其原理是：当数控机床运行到宏采集指令时，将把宏指令中指定的地址变量储存的数据从串口输出来，从而使得嵌入式适配器可以通过串口采集到这些数据。对于不同的数控系统来说，其宏采集指令中所用的地址变量是不同的，可以输出的数据多少也有所不同。一般来说，FANUC 系统和 HAAS 系统的宏采集指令是一致的，下面将主要针对 FANUC 系统进行阐述。

一段完整的宏采集指令如下所示：

```
POPEN
DPRNT [ a #b[c d].....]
PCLOS
```

POPEN：打开机床串口指令，以建立与外部设备的连接。

PCLOS：关闭机床串口指令，以断开与外部设备的连接。

DPRNT：数据输出指令，可以指定输出的变量标识符、输出的系统变量以及数值位数等。

语句 DPRNT 中：a 表示指定输出的变量标识符，与一般程序语言中变量命名规则一样，可以使用数字或者字母；b 表示要输出的数控系统的地址变量；c, d 指定要输出的数据位数，c 表示输出数值的整数位数，d 表示输出数值小数点后的位数。

FANUC 系统的地址变量表如表 3-3 所示^[24]。

表 3-3 FANUC 系统的地址变量表

地址变量	说明
#3000	报警信息，当#3000 中有 0~99 间的某一值时，程序停止并报警
#3001	一个以 1 毫秒为增量的计时器
#3002	一个以 1 小时为增量、当循环启动灯时亮的计时器
#3011	以十进制数字表示的当前日期。如 20120516 表示 2012 年 5 月 16 日
#3012	以十进制表示的当前时间。如 212610 表示 21 点 26 分 10 秒。
#3901	已加工的零件数
#3902	设定的需加工零件数
#4109	当前主轴进给速率
#4111	刀具长度补偿
#4115	正在加工的零件程序号

地址变量	说明
#4119	当前主轴转速
#4120	当前使用的刀具号
#5021~#5023	当前主轴相对于机床坐标系的坐标位置 (X, Y, Z)。
#5041~#5043	当前主轴相对于工件坐标系的坐标位置 (X, Y, Z)。

下面是一段宏采集指令的例子：

```

    POPEN
    DPRNT [TIME #3012[60]]
    PCLOS
    
```

此指令先打开机床串口，然后输出当前时间“TIME083025”(即上午 8 点 30 分 25 秒)，随后关闭机床串口。

HAAS 数控系统没有 FANUC 系统对外开放，可以输出的地址变量比较少，而 SIEMENS 数控系统虽然也有系统地址变量，但不支持宏输出指令，需要人工操作输出。本课题主要以 FANUC 系统研究为主，在此不再详述 HAAS、SIEMENS 系统的宏指令数据采集。

3.4.3.2 串行通信技术

顾名思义，串行通信即把数据逐位 (bit) 按顺序发送或者接收。典型地，串口用于 ASCII 码字符的传输。串行通信的特点是，数据位是按顺序传输的，最少只需要一根传输线即可完成。ARM 内部含有全双工异步 (UART) 通信串行口，最简单的串行通信使用 3 根线即可完成：地线、发送线、接收线，可实现同时发送和接收数据。

串口通信最重要的参数是波特率、数据位、停止位和奇偶校验。对于两个进行通信的端口，这些参数必须匹配才能正常进行通信。这四个参数的说明如表 3-4 所示。

表 3-4 串行通信主要参数表

参数	说明
波特率	衡量通信速度的参数。它表示每秒钟传送的位 (bit) 的个数
数据位	衡量通信中实际数据位的参数，每个数据包中数据位可以为 5、7 或者 8 位。一个数据包包括开始/停止位、奇偶校验位和数据位。
停止位	表示单个包的最后一位，典型的值为 1, 1.5 和 2 位。
奇偶校验位	一种简单的数据正确性检验方式，可以设置为奇校验、偶检验，也可以没有校验位。

串行端口的本质功能是作为 CPU 和串行设备间的编码转换器。当数据从 CPU 经过串行端口发送出去时，串口把字节数据转换为串行的二进制位；在接收数据时，串口把串行的二进制位被转换为字节数据。在 Windows 环境中，串行端口是系统资源的一部分^[25]。

在 C#编程中，使用 SerialPort 类来实现串行通信。SerialPort 类的属性可以设置串行通信参数，如串行端口号、数据位、停止位、握手协议等，并且提供了 Read、ReadLine、ReadTo 等多种读取串口数据的方法和 Write、WriteLine 等写入串口数据的方法。

3.4.3.3 数据库远程访问技术 RDA

远程数据访问技术 (RDA, Remote Data Access) 是实现嵌入式适配器 SQL Server Compact 3.5 数据库和服务 SQL Server 2008 数据库数据同步更新的方法。

通过 SQL Server Compact 3.5 中的远程数据访问功能，应用程序可以访问远程 SQL Server 数据库表中的数据，并将该数据存储在本地 SQL Server Compact 3.5 数据库表中。随后，该应用程序可以读取和更新本地 SQL Server Compact 3.5 数据库表。SQL Server Compact 3.5 可以有选择地跟踪对本地表所做的所有更改。稍后，该应用程序可以将本地表

中更改的记录更新回 SQL Server 表中^[26]。

在 SQL Server Compact 3.5 中,从 SQL Server 表向本地 SQL Server Compact 3.5 表传送数据称为请求数据。将本地 SQL Server Compact 3.5 表中所做的更改传播回 SQL Server 表称为推送数据。

应用程序还可以使用 RDA 来提交 SQL 语句,以便可以在远程 SQL Server 数据库上执行操作。例如,应用程序可以提交 SQL 语句,向远程 SQL Server 表中插入、更新或删除记录。应用程序可以调用任何不返回记录集的 SQL 语句。这包括远程系统上的存储过程。

RDA 使用 Microsoft Internet 信息服务 (IIS) 作为服务器上的 SQL Server 数据库和设备上的 SQL Server Compact 3.5 数据库之间的通信机制。SQL Server Compact 3.5 数据库引擎、SQL Server Compact 3.5 客户端代理和 SQL Server Compact 3.5 服务器代理协同工作才可以完成 RDA^[27]。其关系图如图 3-10 所示。

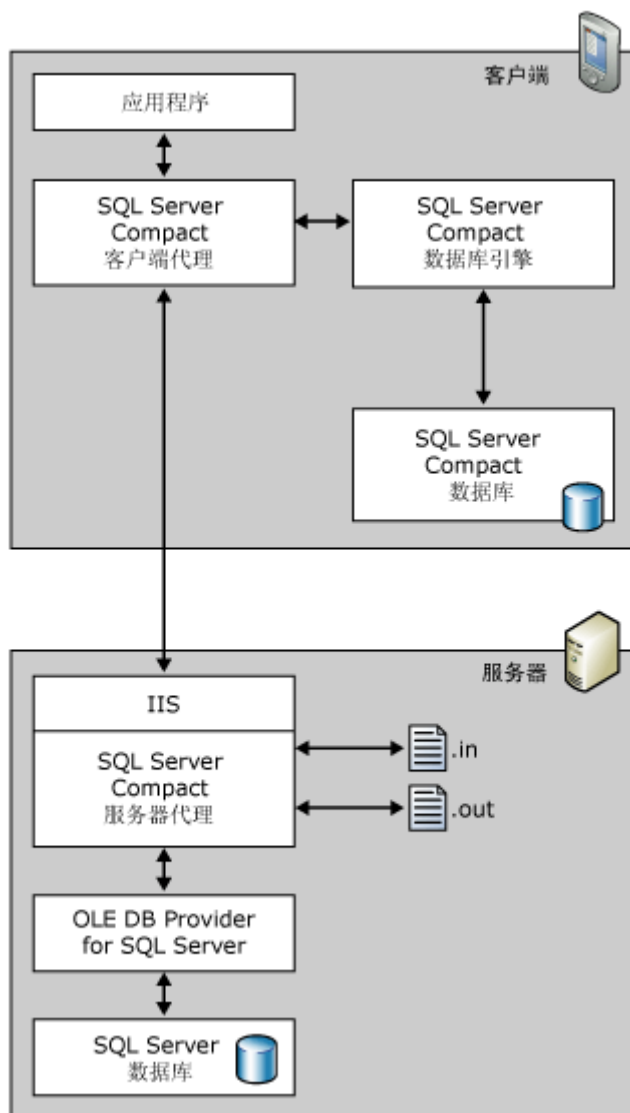


图 3-10 RDA 通信原理图

在 C#编程中,可以通过 SqlCeRemoteDataAccess 类来实现数据库的远程数据访问,对象的 Pull 和 Push 方法对应请求数据和推送数据。

3.4.3.4 多线程技术

在操作系统中,当一个应用程序开始运行后,就创建了一个进程,而一个进程可以包含

一个或者多个线程。线程是程序中的一个执行流，每个线程都有自己的专有寄存器(栈指针、程序计数器等)，但代码区是共享的，即不同的线程可以执行同样的函数。

通常来说，一个应用程序有一个主线程和多个后台线程，主线程一般是窗体界面线程，即用来响应用户的操作，而后台线程则用来进行耗时较长的数据处理等操作。在本数据采集系统中，由于采集数据的需要，必须采用多线程技术。否则，当采集数据时，界面将不再有任何响应。系统界面响应是主线程，当开始采集数据后，要在嵌入式适配器建立一个串口监视线程，以监视数控机床是否有数据输入到串口，同时，当采集到数据后，还要建立一个数据处理线程，以便进行数据的分类格式化和储存等操作。

在 C#编程中，多线程管理是由 Thread 类来实现的，其提供了开始一个新线程、终止一个线程及线程挂起、暂停等的操作。

3.4.4 数据采集应用软件关键模块的实现

3.4.4.1 数据采集模块的实现

如上节所述，数据采集模块是有多个线程来实现的。根据数据采集模块的功能要求，设计用户界面如图 3-11 所示，实现了串口参数设置、串口采集和提交机床开机、关机时间的功能。

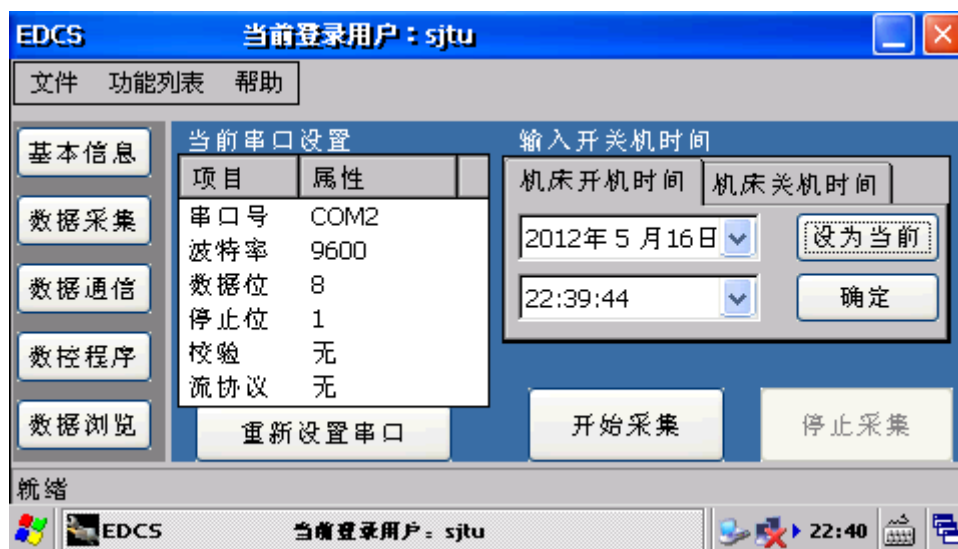


图 3-11 数据采集模块界面

A. 串行通信的实现

首先，要定义一个串行通信类 SerialPort 类的对象 sp，然后根据根据数控机床的串口参数来设置 sp 对象的串口参数，以实现以数控机床的串行通信，部分代码如下：

```
//串口参数设置
sp.PortName = "COM2"; //串行端口号
sp.BaudRate = 9600; //波特率
sp.DataBits = 8; //数据位
sp.StopBits = StopBits.One; //停止位
sp.Parity = Parity.None; //奇偶校验位
sp.Handshake = Handshake.None; //握手协议
```

当用户点击开始采集时，开启串口，即 sp.Start()。当用户点击停止采集时，关闭串口，即 sp.close()。

当有数据到达接收数据缓冲区时，将会引发 SerialPort 对象的数据接收事件 DataReceived()事件，进而在事件处理方法中采集并处理接收数据缓冲区的数据即可。

B. 多线程的实现

在本模块中，采用三个线程来实现数据的采集及处理、储存。其中，线程 1 为主线程，用于界面的响应；线程 2 为串口监视线程，用于监视串口是否有数据到达，当有数据到达时，按要求读取串口数据；线程 3 为数据处理线程，负责对采集到的数据进行分类，并储存到对应的数据表中。数据采集模块多线程框图如图 3-12 所示。

其中，数据监视线程是由 SerialPort 对象自行创建的，即默认的数据接收事件是在辅助线程（即数据监视线程）上引发的。而线程 3 数据处理线程是程序开发中自行设计的。

在进行串行通讯时，数据是连续发送的，所以为了正确区分多条数据，要规定一个数据帧的协议。通常情况下，一个数据帧包括帧头、有效数据、帧尾三部分。在本模块中，POPEN 打开机床串口时，会发送控制字符 DC2（ASCII 码值 18），而 PCLOS 关闭机床串口时，会发送控制字符 DC4（ASCII 码值 20）^[28]。所以，本模块一个数据帧使用 DC2 字符作为帧头，DC4 字符作为帧尾，中间为机床系统地址变量中的有效数据。

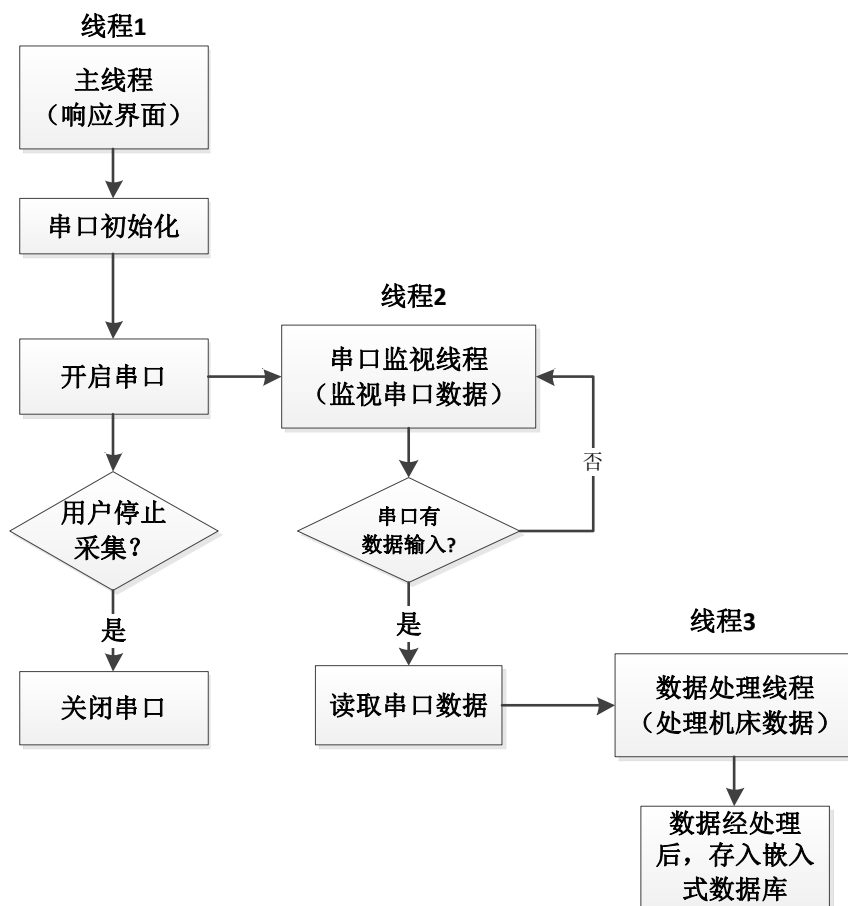


图 3-12 数据采集模块多线程框图

结合多线程及数据帧的定义，即可在 SerialPort 对象的数据接收事件 DataReceived 事件中编写嵌入式适配器数据采集的程序，下面是部分程序代码：

```

//1个字节到来即引发DataReceived事件
sp.ReceivedBytesThreshold = sp.ReadBufferSize;
int startByte = sp.ReadByte();
if (startByte == 18)//说明到了POPEN语句, 要开始发送数据了
{

```

```

//开始读数据，一直读到DC4，即PCLOS发出的字符
strReceive=sp.ReadTo(((char)20).ToString());
try//开始数据处理线程
{
    dataProcessingThread = new Thread(new ThreadStart(dataProcess));
    dataProcessingThread.Start();
}

catch(Exception ex)
{
    MessageBox.Show("开始数据处理线程发生错误" + ex.Message);
}
}

```

C. 数据储存的实现

在进行数据采集时，根据用户需要，可以采集如表 3-3 中所示的任何一个地址变量对应的数据。根据所设计的数据库的结构，不同的数据要储存在三个数据表中：即采集时间表、采集坐标表和机床状态表。而宏采集指令中的 DPRNT 命令是按照指定的数值位数以连续的数字字符串输出数据的，所以为了标明所接收的数据是哪种类型，需要为不同的数据类型指定不同的变量名。

在线程 3 数据处理线程中，会根据变量名对数据进行分类，以确定所对应的数据类型，随后会按照数据表中对应字段格式的要求，对数据进行格式化，并将数据储存到对应的数据表中。流程如图 3-13 所示。

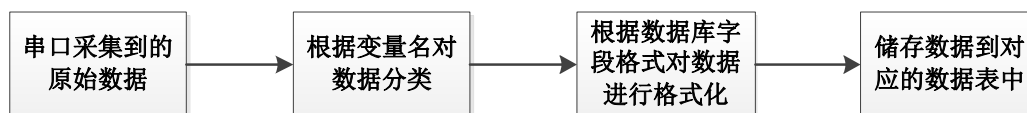


图 3-13 串口数据处理线程的工作流程

以采集程序加工开始时间为例，其宏采集指令如下：

```

POPEN
DPRNT[PROSTART#3011[80]#3012[60]]
PCLOS

```

假定程序加工开始时间为 2012 年 5 月 17 日 16 点 09 分 30 秒，则按照上述指定的格式，串口采集到的原始数据为：

```
PROSTART20120517160930
```

根据变量 PROSTART，判断采集到的数据为程序加工开始时间，在数据库中为 datetime 类型，进而将这个数据格式化，成为“2012-05-17 16:09:30”，然后将其储存到“采集时间表”的“程序加工开始时间”字段中。即完成了对采集数据的分类处理和储存工作。

3.4.4.2 数据通信模块的实现

根据数据通信模块的功能需求，设计用户界面如图 3-14 所示。实现了远程数据访问信息配置及数据表下载和上传的功能。



图 3-14 数据通信模块界面

数据通信模块通过下载服务器数据库中的数据表并把储存有最新机床采集数据的数据表上传到服务器数据库，实现了服务器和嵌入式适配器的数据同步更新。应用的关键技术为 SQL Server 数据库的远程数据访问（RDA），在程序中是通过 `SqlCeRemoteDataAccess` 类来实现远程数据访问的。首先，要对 `SqlCeRemoteDataAccess` 对象进行初始化，指定必须的信息，如远程服务器的 IP、用户名、密码及服务器数据库和本地嵌入式数据库的连接字符串等，随后便可以调用 `SqlCeRemoteDataAccess` 对象的相关方法来实现数据表的下载和上传。`SqlCeRemoteDataAccess` 类的主要方法如表 3-5 所示^[29]。

表 3-5 `SqlCeRemoteDataAccess` 类的主要方法

方法	说明
Pull	从远程 Microsoft SQL Server 数据库下载数据，并将这些数据存储在本地 Microsoft SQL Server Mobile 数据库的一个表中。
Push	将 SQL Server Mobile 中跟踪的提取表中的更改传送回原始 SQL Server 表。这些更改分别使用各自的事务应用到服务器。
SubmitSql	提交将在远程服务器上 Microsoft SQL Server 中的数据库上执行的 SQL 语句

由于通过无线通信上传和下载数据时，如果数据量比较大，用时将比较长，所以为减少用户等待时间，数据的上传和下载都将建立一个新线程，在后台运行，不对用户的界面操作产生影响。

下载数据表的 Pull 方法定义如下：

```
public void Pull (
    string localTableName,
    string sqlSelectString,
    string oleDBConnectionString,
    RdaTrackOption trackOption
)
```

其中最后一个参数 `trackOption` 指示 SQL Server CE 是否跟踪对提取的表所做的更改，在本模块中设置其为跟踪全部更改。这样，在采集到新的数控机床数据后，调用 `Push` 方法进行数据上传时，只需要上传表中更新的数据，会大幅降低网络流量，减少上传数据所需时间。

3.4.4.3 数控程序传输模块的实现

根据数控程序传输模块的功能需要，设计用户界面如图 3-15 所示。用户可以选择本地嵌入式数据库中的数控程序，并在右侧的数控程序编辑框中对其进行编辑。接下来用户可以选择要采集的机床数据类型，如程序加工开始时间、程序加工结束时间、当前刀具号、当前主轴转速、当前已经加工的零件数、设定的总加工零件数、刀具长度补偿以及当前主轴坐标等程序会自动添加相应的宏采集指令。最后，用户将添加指令后的数控程序下载到数控机床中，则数控机床运行数控程序时，即可从机床串口输出用户想要采集的数据，然后利用嵌入式适配器进行采集即可。

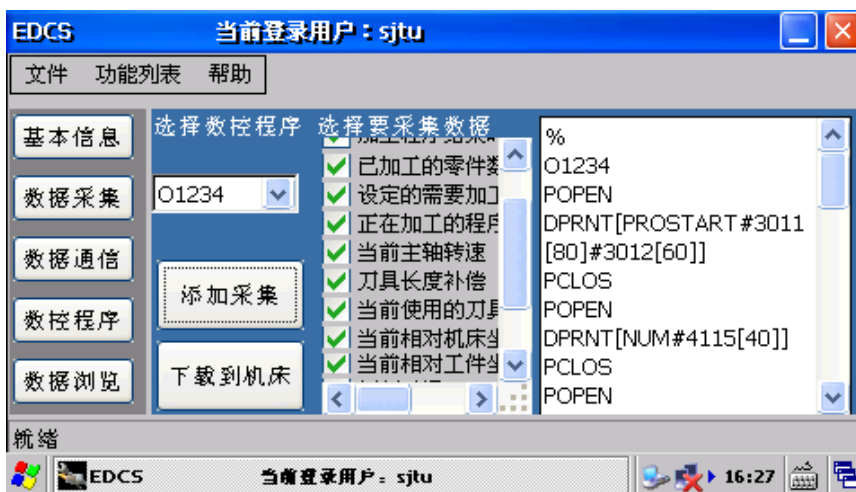


图 3-15 数控程序传输模块界面

A. 添加宏输出指令

添加采集指令时，要分析每一行 NC 程序（数控程序）中关键指令，如 G 表示切削、O 代表程序号、S 表示主轴转速、T 表示刀具号等，然后根据用户选择的采集数据类型来添加相应的宏采集指令。如用户选择了程序加工开始时间，则只要在含有程序号（指令为 O）的数控程序下一行添加宏输出指令，输出此刻的时间，即得到了程序加工开始时间。添加宏输出指令的流程图如图 3-16 所示。

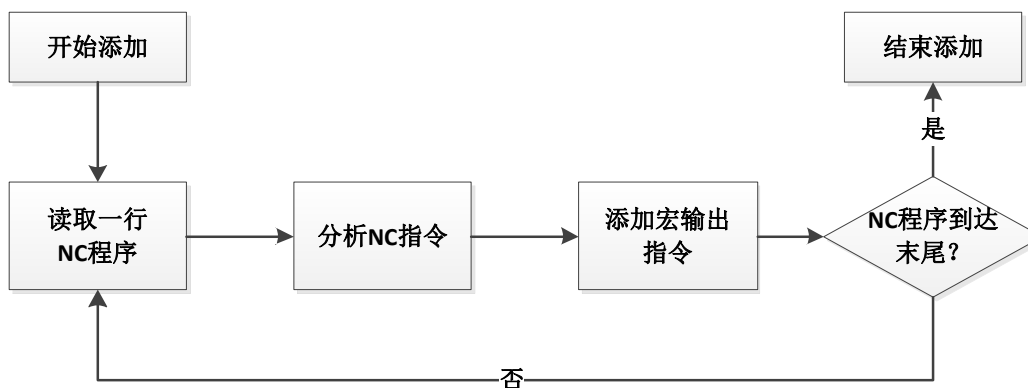


图 3-16 添加宏采集指令的流程图

以下是添加程序加工时间宏输出指令：

```

string[] NCText = textBoxNC.Text.Split(new char[] { '\r', '\n' });
ArrayList list = new ArrayList();
foreach (string s in NCText)
{
    if (s != string.Empty) list.Add(s);
}
    
```

```

for (int i = 0; i < list.Count; i++)
{
    if ((!isMacro(list[i].ToString()))&&
        list[i].ToString().IndexOf("O")>-1)
    {
        if (checkBoxStartTime.Checked)//程序加工开始时间
        {
            list.Insert(i + 1, "POPEN\r\nDPRNT[PROSTART#3011[80]"
                + "#3012[60]]\r\nPCLOS");
            i++;
        }
    }
}

```

B. 传输速率匹配

在嵌入式适配器向数控机床传输 NC 程序时,可能存在两者通信速率不匹配的问题。即嵌入式适配器发送数据的速率快于数控机床接收数据的速率,虽然数控机床有一定的数据接收缓冲区,但如果两者速率差过大,仍会造成数据溢出和数据丢失发生。为防止数据丢失,在嵌入式适配器向数控机床传输 NC 程序时,采用软件握手协议 XON/XOFF 来匹配两者的传输速率。

XON/XOFF 是一种流控制协议(通信速率匹配协议),用于数据传输速率大于等于 1200b/s 时进行速率匹配,方法是控制发送方的发送率以匹配双方的速率。以数控机床和嵌入式适配器的串行通信为例,当数控机床接收缓冲区数据量达到某一个阈值时,就会发送 XOFF 控制字符,以通知嵌入式适配器停止发送;当数控机床接收缓冲区内数据被读取而有空间,传输可以恢复后,就会发送 XON 字符,以通知嵌入式适配器继续发送。通常, XON 采用 ASCII 字符集中的控制字符 DC1 (ASCII 码值为 17), XOFF 采用 ASCII 字符集中的控制字符 DC3 (ASCII 码值为 19) [30]。

当需要嵌入式适配器向数控机床发送 NC 程序时,首先要把数控机床调到准备接收 NC 程序的状态,然后点击“下载到机床”按钮,即可实现 NC 程序的传输。传输 NC 程序的流程图如图 3-17 所示。

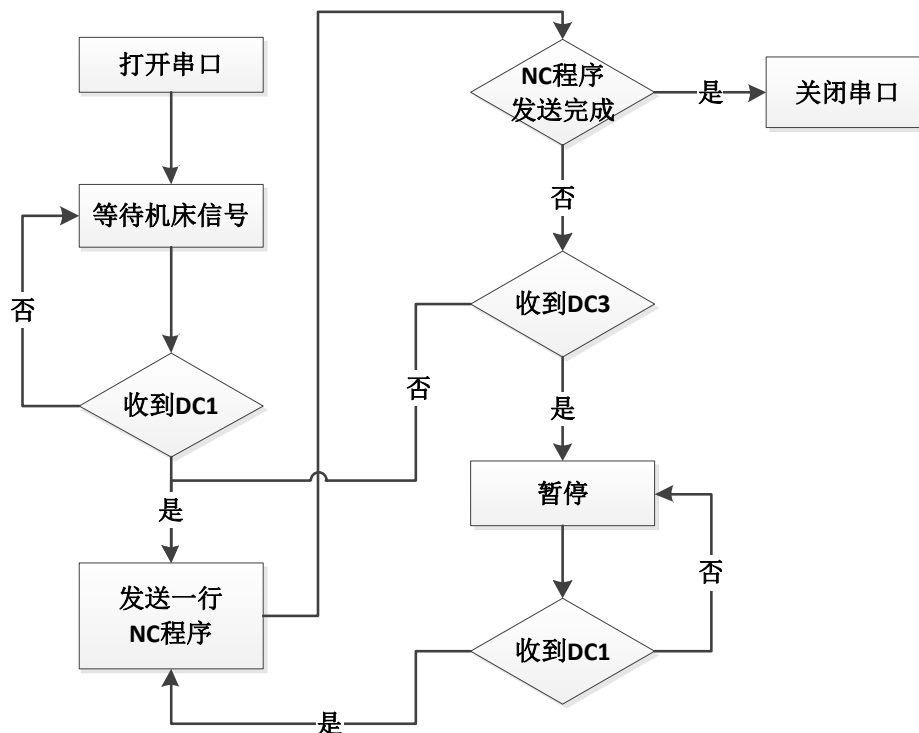


图 3-17 嵌入式适配器 NC 程序传输流程

在实际的串行通信编程中，只要设置 SerialPort 串口通信对象的 Handshake = 属性为 Handshake.XOnXOff，即可实现采用 XON/XOFF 通信协议进行串行通信，以满足数控机床和嵌入式适配器的传输速率匹配。部分串行通信代码如下：

```
if (sp.Handshake != Handshake.XOnXOff)
    sp.Handshake = Handshake.XOnXOff; //握手协议
StringReader sr = new StringReader(textBoxNC.Text);
try
{
    sp.Open();
    sp.DiscardOutBuffer();
    while (sr.Peek() > -1)
    {
        //读取一行NC程序
        string line = sr.ReadLine();
        //转化为字节编码
        byte[] sendBuffer = Encoding.ASCII.GetBytes(line + "\r\n");
        //发送一行NC程序
        sp.Write(sendBuffer, 0, sendBuffer.Length);
    }
}
```

3.5 本章小结

本章首先介绍了嵌入式处理器 S3C6410 和飞凌 OK6410 开发板，在配置扩展模块后，实现了嵌入式适配器的硬件平台。然后讲解了嵌入式操作系统 Windows CE 的定制和移植，并介绍了嵌入式数据库 SQL Server Compact 3.5。接下来，设计了嵌入式适配器数据采集应用软件的功能模块，讲解了各功能模块的主要功能。最后，搭建了开发环境，研究了开发数据采集应用软件所需的关键技术，并讲解了关键模块的实现。最后，利用 C#语言进行了嵌入式适配器数据采集应用软件的实际开发。

第4章 客户端数据管理系统的实现

4.1 客户端数据管理系统的功能模块设计

4.1.1 数据管理系统各功能模块的划分

根据基于嵌入式适配器的数据采集系统的功能需求分析,结合 C/S 的架构特征,客户端数据管理系统负责对各个嵌入式适配器上传到服务器的数据进行管理工作,如数据查询和数据统计、处理等,同时还要对车间、设备、用户、数控程序等信息进行管理。客户端的数据管理系统既可以对嵌入式适配器采集的数据进行深入的统计和分析,又可以为嵌入式适配器提供必须的基本信息,如车间、设备信息和数控程序等。客户端数据管理系统的功能模块划分如图 4-1 所示。

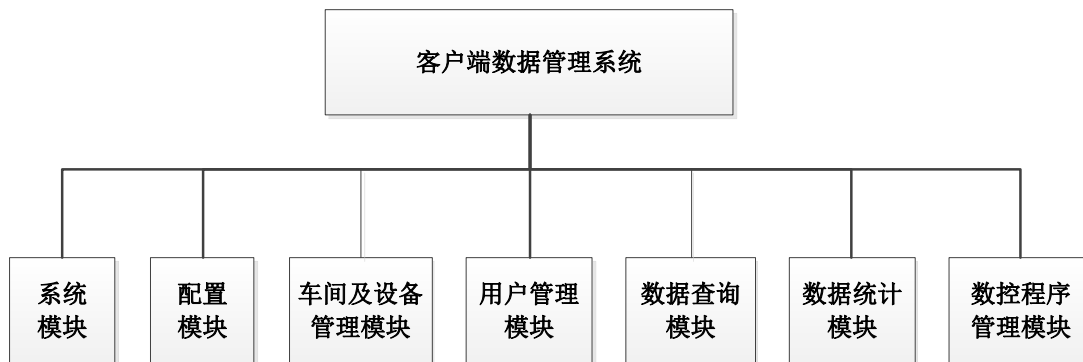


图 4-1 客户端数据管理系统功能模型图

4.1.2 数据管理系统各功能模块的主要功能

4.1.2.1 系统模块

系统模块主要负责对系统整体的相关操作,如系统登录、系统注销、系统锁定等。系统模块的功能模型如图 4-2 所示。

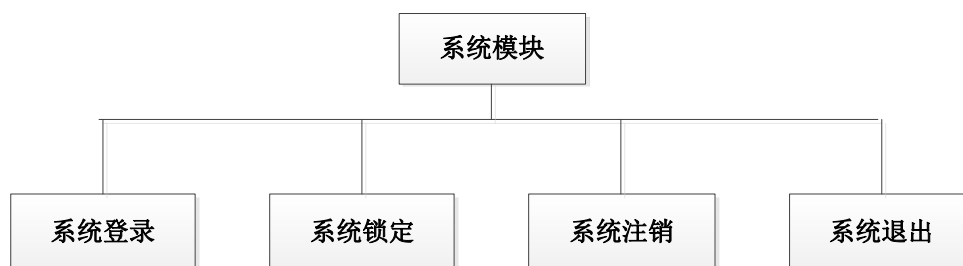


图 4-2 系统模块的功能模型图

4.1.2.2 配置模块

主要包括数据库的配置和企业日历配置功能。数据库的配置是为了满足远程客户端连接服务器的需要,需要输入服务器的 IP、数据库名、用户名及密码以连接服务器数据库。而企业日历配置则是出于数据统计的需要,为计算出表征机床效率的有关参数,需要根据企业日历来计算企业的上班時間。配置模块功能模型如图 4-3 所示。

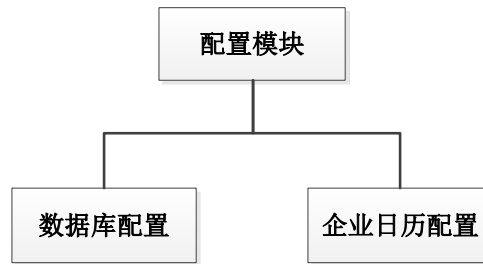


图 4-3 配置模块功能模型图

4.1.2.3 车间及设备管理模块

此模块主要负责数据库中车间及设备信息的管理，即管理三个数据表：车间信息表、设备信息表、设备型号表。若用户有足够的权限，可以对三个表中的相关信息进行修改、增加及删除操作。服务器数据库更新后，只要嵌入式适配器重新下载这些数据表，即可实现嵌入式适配器有关信息的更新。车间及设备管理模块的功能模型如图 4-4 所示。

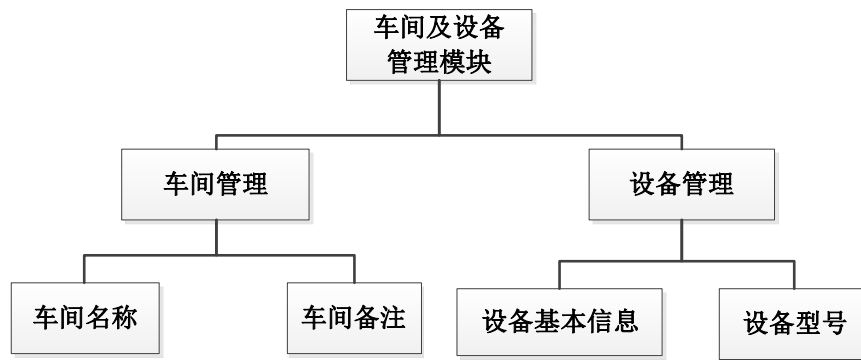


图 4-4 车间及设备管理模块功能模型图

4.1.2.4 用户管理模块

此模块主要负责对用户信息进行管理。客户端管理系统中设定了两种用户权限，即一般用户和管理员用户。一般用户只能够修改自己的用户信息，如用户名和用户密码。而管理员用户则可以管理所有用户的信息，可以对用户信息进行修改，删除或者增加某个用户。用户管理模块的功能型如图 4-5 所示。

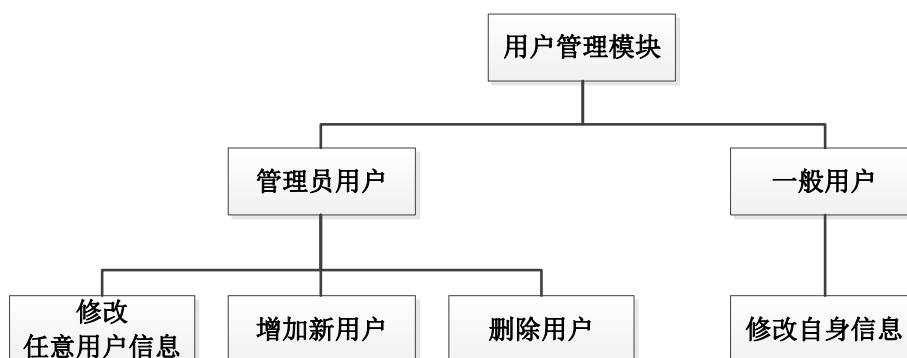


图 4-5 用户管理模块

4.1.2.5 数据查询模块

此模块主要对各个嵌入式适配器采集到的机床数据进行查询。用户可以指定要查询的表、要查询的日期区间等条件，进行详细的数据查询。当用户有管理员权限时，可以对查询到的数据进行修改，并提交修改后的数据到数据库中。用户要查询的信息在本模块中分为两类，一是采集到的机床数据，二是设备、车间等管理层数据。数据查询模块的功能模型如图 4-6

所示。

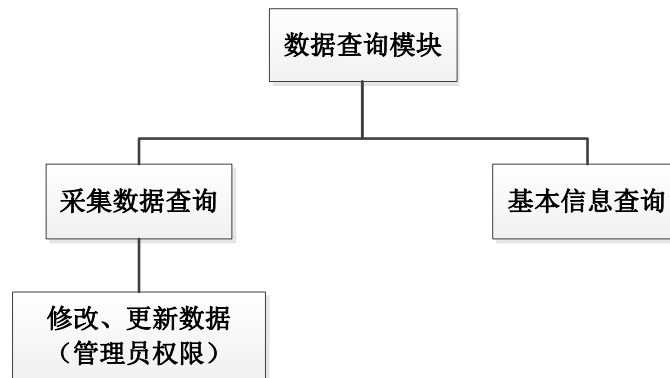


图 4-6 数据查询模块功能模型图

4.1.2.6 数据统计处理模块

本模块主要用于对采集到的机床数据进行挖掘和统计，计算出反映机床效率和利用率的相关参数，以代用户参考。本模块可以根据用户选择的统计日期区间，以表格或者饼状图的形式展示统计结果，并可以保存结果为本地图片文件。数据统计处理模块功能模型图如图 4-7 所示。

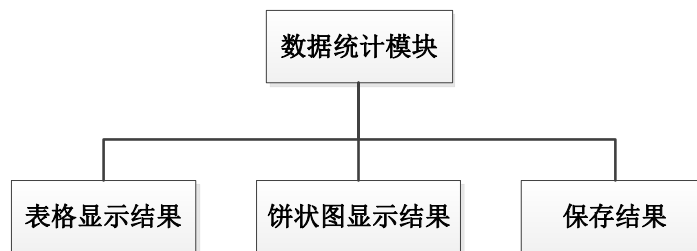


图 4-7 数据统计模块功能模型图

4.1.2.7 数控程序管理模块

本模块主要负责对数据库的数控程序进行管理，如查询数据库中数控程序、编辑数控程序、保存成本地数控程序、添加本地数控程序到数据库等。数控程序管理模块的功能模块如图 4-8 所示。

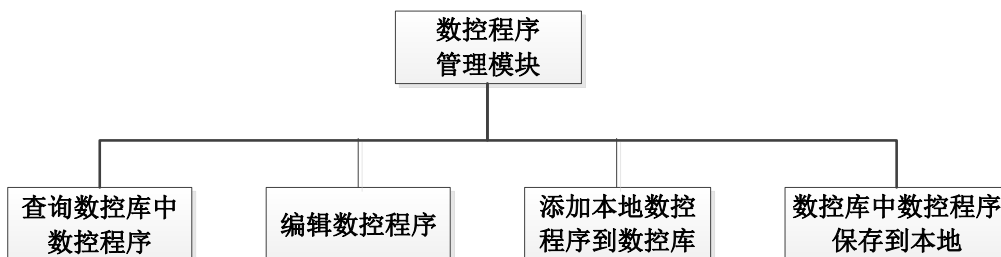


图 4-8 数控程序管理模块功能模型图

4.2 客户端数据管理系统的核心技术

客户端数据管理系统的主要功能都依赖于与数据库的交互和操作，需要用到数据库数据访问及数据绑定技术，同时嵌入式适配器利用远程数据访问（RDA）实现了与服务器的通信，需要在服务器电脑上进行一定的环境配置。下面将介绍 SQL Server 2008 数据库，并阐述数据访问和数据绑定等 ADO.NET 技术。

4.2.1 服务器数据库 SQL Server 2008 及 ADO.NET 数据访问技术

4.2.1.1 SQL Server 2008 简介

SQL Server 2008 数据库系统是微软公司新近推出的基于 C/S 模式的大型分布式高性能关系数据库管理系统，与 Windows 操作系统高度集成，是一个全面的、集成的、端到端的数据解决方案，它为企业中的用户提供了一个更安全可靠和更高效的平台^[31]。

SQL Server 2008 使用 Transact-SQL 语句在 Server 和 Client 间传递请求，把所有的工作负荷分解成在 Server 机器上的任务和 Client 机器上的任务。本课题研究的数据采集系统服务器将以 SQL Server 2008 作为后台数据库，以实现大量数据的管理和存储工作。

4.2.1.2 远程数据访问 (RDA) 环境设置

首先，要在服务器电脑上开启 IIS (Internet Information Services, 互联网信息服务)，IIS 服务是由微软公司提供的基于运行 Microsoft Windows 的互联网基本服务。对于 Windows XP 系统来说，需要下载 IIS 安装包并安装 IIS，对于 Windows 7 系统来说，已经内置了 IIS，只需要选择开启即可。

然后需要设置一个虚拟目录，在“控制面板”打开“Internet 信息服务”，建立一个虚拟目录，设定其允许权限为读取、写入、运行脚本和执行，并连接到本地一个共享文件夹即可。

最后，向虚拟目录中复制一个 SQL Server CE 服务器代理的副本，即 sqlcesa35.dll^[32]。

4.2.1.3 ADO.NET 数据访问技术

ADO.NET 的名称起源于 ADO (ActiveX Data Objects)，是一个在 .NET 平台上使用的提供数据访问的构架和方法，它允许以多种方式从多种数据源读取、处理和写入多种数据。从基本角度来讲，ADO.NET 是一个面向对象的类库，提供了大量类来对数据库进行操作，可以访问包括 SQL Server、Access 和 Oracle 数据库在内的多种数据库，并支持访问嵌入式数据库如 SQL Server Compact Edition。通过使用 ADO.NET 数据访问技术，可以以更加简洁易懂的代码来实现数据的访问，并且可以提高数据编程的抽象水平^[33]。

在本课题的程序设计中，由于是针对 SQL Server 数据库进行操作，所以使用了 ADO.NET 中专门针对 SQL Server 的类。这些类在命名空间 System.Data.SqlClient 中，主要使用的类如表 4-1 所示。

表 4-1 ADO.NET 部分类列表

类	说明
SqlConnection	主要用于连接 SQL Server 数据库
SqlCommand	主要用于向数据库提交 SQL 语句命令
SqlDataReader	主要用于按顺序读取数据库中的记录
SqlDataAdapter	主要表示用于填充 DataSet 和更新 SQL Server 数据库的一组数据命令和一个数据库连接
DataSet	主要用于把数据库中指定数据放于内存中，并对数据进行相关操作

在嵌入式适配器的程序设计中，只要换成专门面向 SQL Server CE 的命名空间 System.Data.SqlServerCE，并采用其中的类，即可实现相同功能。

4.2.2 数据绑定技术

.NET 平台提供了数据绑定技术来进行数据源中数据的显示和更新。从一个 Windows 窗体的角度来看，“数据绑定”是一种把数据绑定到一种用户界面元素（控件）的通用机制。在 Windows 窗体中有两种数据绑定类型：简单绑定和复杂绑定。在本课题研究中主要使用了复杂绑定类型^[34]。

复杂数据绑定是把一个基于列表的用户界面元素（比如 ComboBox、Grid）绑定到一个数据实例列表（比如 DataTable）的方法，而 DataTable 则类似数据库中的一个数据表，所以

利用此技术，可以向用户展示数据库的数据，并且可以把用户对数据的更改传递回数据库以更新数据库中数据。

当数据进行绑定后，数据流的流向有三种情况，可以通过 Binding 对象的 Mode 属性对此加以控制。三种数据流向如图 4-9 所示。其中 OneWay 指数据只从数据源流向绑定目标，用户对绑定目标的更改不会影响到数据源；而 TwoWay 指数据可以从数据源流向绑定目标，也可以从绑定目标流向数据源，用户可以更新数据源；OneWayToSource 则指数据只能从绑定目标流向数据源，用户可以更新数据源，但数据源发生的更改不会反映给绑定目标。



图 4-9 数据绑定后的三种数据流向^[35]

本程序设计中主要使用了 DataGrid 列表控件和数据源 DataSet 的绑定，并根据实际需要来设置数据流的方向。

4.3 客户端数据管理系统的界面设计

本管理系统界面布局采用较标准的 Windows 窗体界面风格，即由菜单栏、工具栏和客户区组成。其中为充分利用客户区空间，采用选项卡的形式来布置客户区，用户单击不同的选项卡，将显示不同的窗体，以完成不同的功能。

本管理系统由多个窗体组成，除主窗体外，还包括登录窗体，以及一些进行功能处理的弹出子窗体，如用户管理窗体、车间设备管理窗体等。下面是本管理系统中部分窗体的用户界面设计。

4.3.1 登录窗体

用户进入系统前，必须先进行登录，以保证数据的安全性。用户权限分为两类，即一般用户权限和管理员用户权限。当用户成功登录后，系统会根据数据库中的储存的用户信息，自动判断当前登录用户的权限，并禁用或者启用系统的某些功能。

总体来说，一般用户权限只能浏览数据和统计数据；而管理员用户除了浏览数据和统计数据外，还可以对数据进行管理操作，如修改、增加、删除数据等。图 4-10 是登录界面。



图 4-10 客户端数据管理系统登录界面

4.3.2 主窗体

主窗体完成主要的数据库操作，如数据查询、数据统计和数控程序管理等。主窗体左下角以一树形控件展示企业中车间及设备的分布情况，即“机床树”，其可以清楚地展示企业中所有车间和设备的分布情况，并方便用户对车间和设备信息进行选择和修改操作，如用户选择机床树中某一设备，则右侧选项卡中相应设备信息也会自动改变，并且机床树具有右键菜单功能。主窗体如图 4-11 所示。



图 4-11 客户端数据管理系统主窗体

4.3.3 企业日历配置窗体

企业日历配置窗体主要用于制订企业当年的日历，并且保存信息到数据库中，从而作为数据统计时的时间依据。

企业日历配置窗体可以设置每天的标准工作时间、特殊工作时间（如加班）、是否有双休日、调班以及当年的节假日安排等，如果可以准确地计算出每天的实际工作时间。企业日历配置窗体如图 4-19 所示。

4.3.4 管理功能窗体

除以上窗体外，还有一些弹出窗体实现对数据库中数据的管理功能。当用户单击菜单栏中某一项或者工具栏中的快捷图标时，便会弹出这些窗体，如设备管理窗体、车间管理窗体、用户管理窗体等。图 4-12 是管理员用户管理窗体的界面，图 4-13 是设备管理窗体的界面，其余窗体界面在此不再详述。

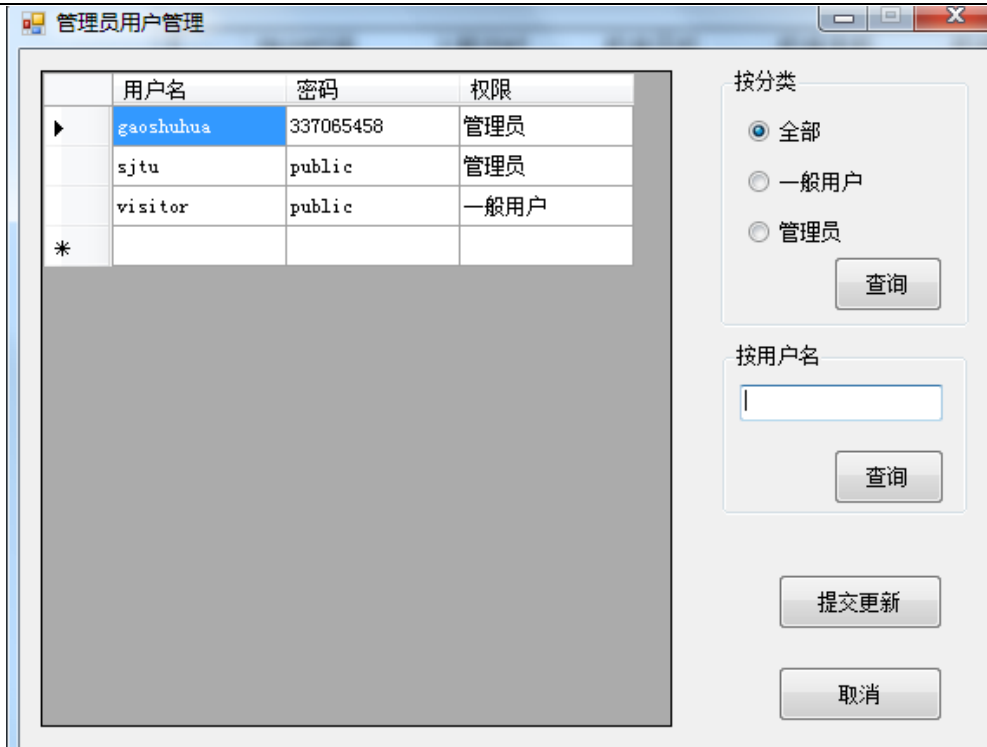


图 4-12 管理员用户管理窗体界面



图 4-13 设备管理窗体界面关键模块的实现

4.4 客户端数据管理系统关键模块的实现

4.4.1 机床树的实现

本系统中的机床树主要实现了两个功能：一是直观显示车间和设备信息，表现企业中车

间和设备的分布情况，并方便用户选择某一车间或者设备；二是具有右键菜单功能，用户可以直接在右键菜单中对相应车间和设备信息进行修改，比如更改车间名称、更改设备串口配置信息、删除或者增加车间和设备等，相比于利用菜单和弹出窗体来管理车间、设备信息更加方便和快捷。

4.4.1.1 机床树的初始化

在窗体的加载过程中，会从数据库中“车间信息表”、“设备信息表”中读取相关信息，并加到机床树的对应节点之中。初始化机床树的流程如图 4-14 所示：

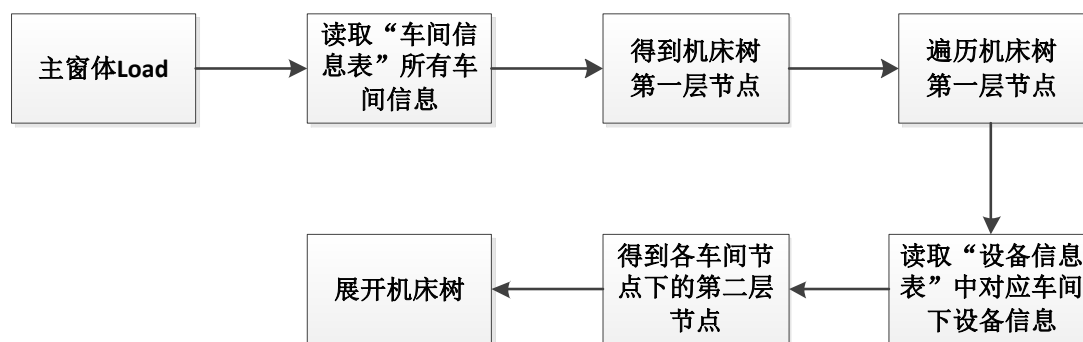


图 4-14 机床树初始化流程

下面是机床树初始化的部分代码：

```

//连接数据库
connection = new OleDbConnection(connectionStr);
connection.Open();
command = connection.CreateCommand();
command.CommandText = "SELECT 车间代号, 车间名 FROM 车间信息表";
dataReader = command.ExecuteReader();
while (dataReader.Read())//添加车间信息
{
    string workshop = dataReader.GetString(0) + "-" +
        dataReader.GetString(1);
    treeViewMachine.Nodes[0].Nodes.Add(workshop);
    treeViewMachine.Nodes[0].Nodes[i].ImageIndex = 1;
    i++;
}
dataReader.Close();
if (treeViewMachine.Nodes[0].Nodes.Count==0) return;
//添加各车间的机床信息
foreach (TreeNode tn in treeViewMachine.Nodes[0].Nodes)
{
    i = 0;
    command.CommandText = "SELECT 设备编号 FROM 设备信息表 "
        + "WHERE 车间代号=' "
        + tn.Text.Substring(0, tn.Text.IndexOf("-")) + "' ";
    dataReader = command.ExecuteReader();
    while (dataReader.Read())
    {
        tn.Nodes.Add(dataReader.GetString(0));
    }
}
    
```

```

tn.Nodes[i].ImageIndex = 2;
i++;
}
dataReader.Close();
}
treeViewMachine.ExpandAll(); //展开所有节点

```

4.4.1.2 机床树右键菜单的实现

当用户右键单击机床树时，系统会弹出右键菜单，并根据用户右键单击的节点是车间节点还是设备节点决定弹出的右键菜单具体项目。设备节点的右键菜单提供了更改设备编号、更改设备型号、更改设备串口信息(如串口号、数据位、停止位、奇偶校验位和握手协议等)、更改所属车间编号和删除此设备的功能；车间节点的右键菜单提供了更改车间代号、更改车间名称、更改车间备注的功能。当用户鼠标单击机床树时，系统的处理流程如图 4-15 所示。

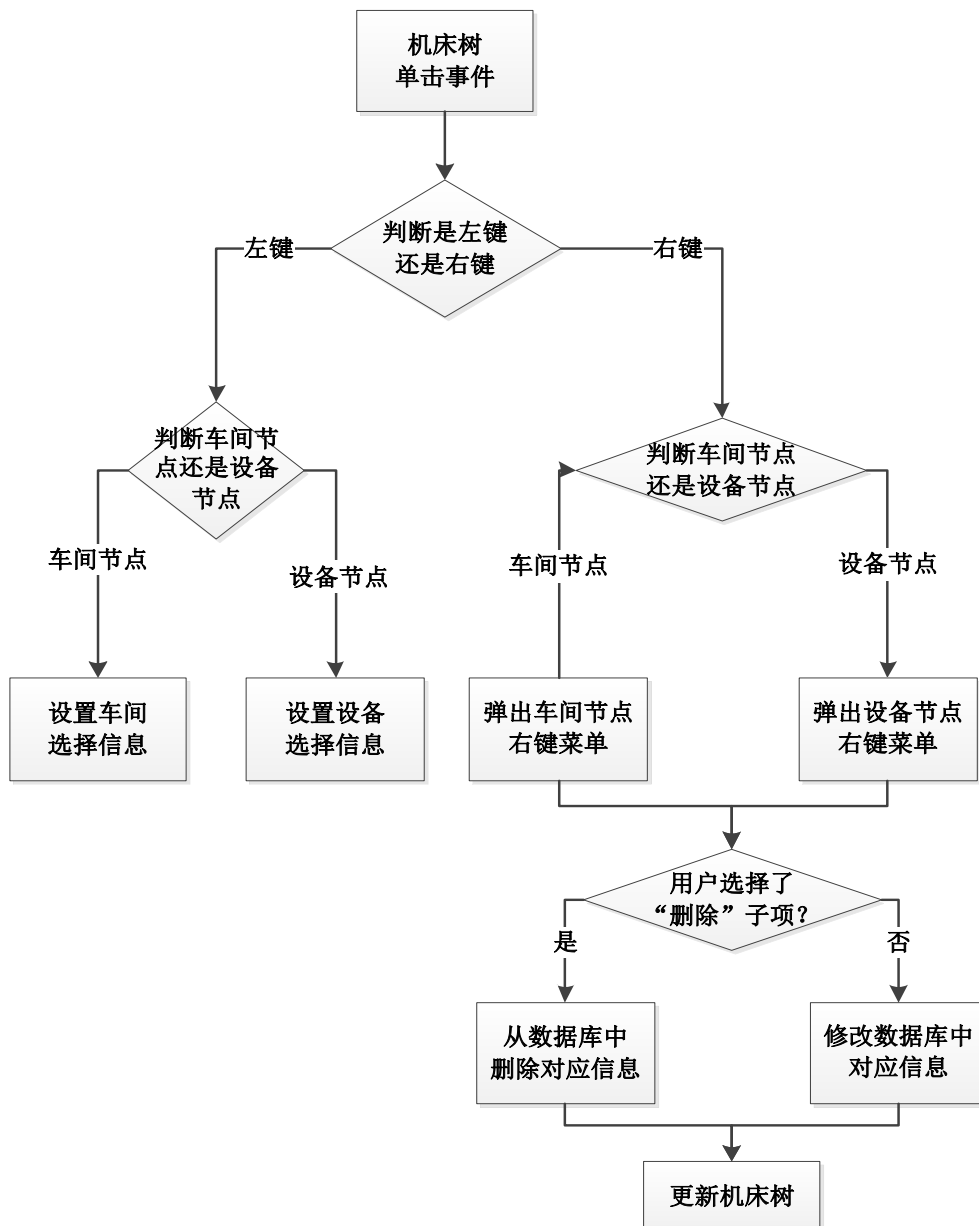


图 4-15 机床树鼠标单击事件处理流程图

当用户右键单击车间节点或者设备节点时，弹出右键菜单分别如图 4-16 和图 4-17 所

示。右键菜单中选择子项后进行的操作实质上是对数据库的信息进行修改、增加或者删除的操作，所涉及到的数据表有“设备信息表”、“设备型号表”和“车间信息表”。当用户选择右键弹出菜单中某一子项并进行操作后，数据库中相关数据将进行更新，数据管理系统的相关模块，如机床树模块和数据查询模块，也会对此做出反应，即重新读取数据库中经更新后的数据，并进行界面的更新，保证界面显示与数据库中的数据更新相一致。以用户右键单击某车间节点并选择“删除此车间”选项为例，对应的部分程序代码如下：

```

if (rightButtonNode == null) return;
if (((ToolStripItem)sender).Text == "删除此设备")
{
    if (DialogResult.No == MessageBox.Show("确定要删除此设备吗?"
        + "与此设备相关的数据也将删除。", "警告",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning))
        return;
    try
    {
        connection = new OleDbConnection(connectionStr);
        connection.Open();
        command = connection.CreateCommand();
        command.CommandText = "DELETE FROM 设备信息表 WHERE "
            + "设备编号='" + rightButtonNode.Text + "'";
        command.ExecuteNonQuery();
        //MessageBox.Show("修改成功! 重新登入系统时生效");
        FormMain_Load(sender, e);
    }
    catch (InvalidOperationException ex)
    {
        MessageBox.Show("删除设备时发生错误: \r\n" + ex.Message);
    }
    finally
    {
        command.Dispose();
        connection.Close();
    }
}
return;

```

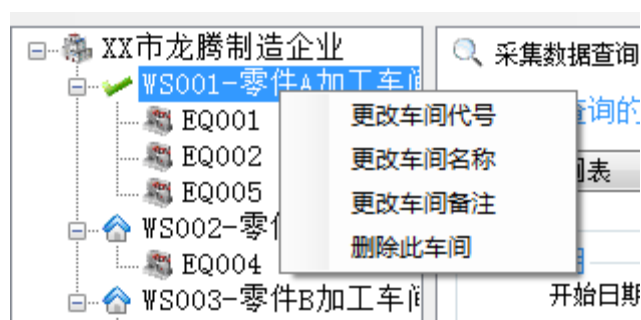


图 4-16 右键单击车间节点界面

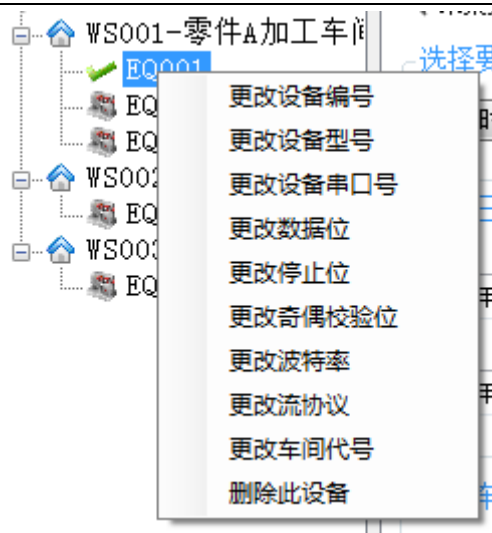


图 4-17 右键单击设备节点界面

4.4.2 数据统计处理模块的实现

4.4.2.1 统计公式及模块界面

数据处理模块主要实现对服务器数据库中数据的初步统计和处理。当各个嵌入式适配器采集到数据并上传到服务器的数据库后，这些数据本身并没有表达出有用的信息或者结论。为此，此模块主要对“采集时间表”进行了分析，得出了以下两个参数^[36]。

- (1) 设备负荷率 P_1 ：反映出设备的忙闲情况。

$$P_1 = T_O / T_S \quad (4-1)$$

T_O ：机床开机时间

T_S ：机床工作时间（即车间上班时间）

- (2) 设备运行有效负荷率 P_{0e} ：真实反映设备的使用情况，通过该统计量可以间接反映出操作工人的实际工作情况。

$$P_{0e} = T_P / T_O \quad (4-2)$$

T_P ：机床程序运行时间

T_O ：机床开机时间

通过以上两个负荷和利用率参数的计算，就可以从繁杂的机床采集数据中提炼出有用的信息和结论，向管理者反映车间中机床设备的开机时间在整个工作时间中所占的比率以及机床开机后真正用于加工的时间所占的比率。准确的统计数据将使管理者能够掌握各车间机床的实际运行情况，从而能够更加高效地管理和调整各车间的机床，并且依据统计结果做出合适的生产计划和调度安排，也可以一定程度上了解各车间员工的工作情况，给员工的业绩评测提供一定的依据。图 4-18 是数据统计处理模块的界面。

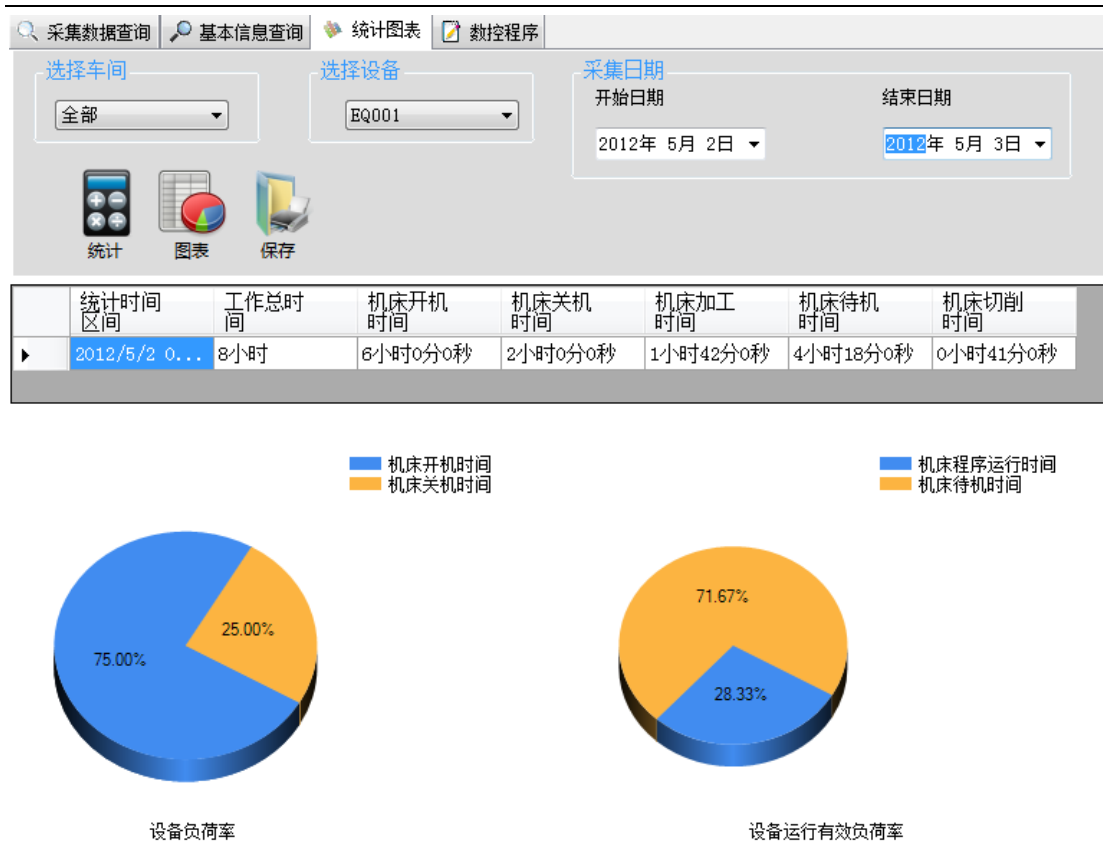


图 4-18 数据统计处理模块的界面

用户操作时，需要先选择车间及设备，然后选择要统计的日期区间。“统计”按钮以表格形式显示统计结果，“图表”按钮以饼状图形式显示统计结果，并计算出相关比率。而“保存”按钮允许用户将饼状图保存成本地图片文件。

4.4.2.2 企业日历配置的实现

在公式（4-1）中，计算设备负荷率 P_1 时，需要机床工作时间作为基数。而实际的机床工作时间计算比较复杂，主要是因为企业车间可能因为假期、周末或者调班、加班等而导致企业车间的每天工作时间并不固定，难以简单地用每天标准工作时间乘以日期区间得出选定日期区间的总工作时间。为此，在系统中设置了企业日历配置模块，允许企业提前设置好一年中每天的具体工作时间，或者将临时调整的工作时间输入到数据库中以进行更新。这样，当进行选定日期区间的机床工作时间统计时，便可以统计到实际的工作时间，从而得到准确的统计结果。企业日历配置窗体的界面如图 4-19 所示。



图 4-19 企业日历配置窗体

用户只需要按照界面上的说明和提示，进行相应的日历设置或者调整，然后添加到数据库中的“企业日历表”即可。而“查询”功能则可以查询某一天或者全年的日历安排。

以向数据库添加节假日为例，首先用户要选择或者输入节假日名称，然后选择节假日的放假日期区间，即可添加到数据库中。在程序代码中，首先要连接数据库，然后选择日期在节假日区间内的记录，最后对这些记录的工作时间、类型和备注等进行更新即可。可以利用 Command 对象来提交 SQL 语句完成这些操作，部分代码如下：

```
//格式化开始日期
string start = string.Format("{0}{1:D2}{2:D2}",
    dateTimePicker1.Value.Year,
    dateTimePicker1.Value.Month, dateTimePicker1.Value.Day);
//格式化结束日期
string end = string.Format("{0}{1:D2}{2:D2}",
    dateTimePicker2.Value.Year,
    dateTimePicker2.Value.Month, dateTimePicker2.Value.Day);
```



```

try
{
    connetion = new SqlConnection(connectionString);
    connetion.Open();
    command = connetion.CreateCommand();
    command.CommandText = "UPDATE 企业日历表 SET 类型='节假日',"
        + "工作时间=0,备注='"+ comboBoxHoliday.Text.Trim() + "'";
        + "WHERE 日期>='"+ start + "' AND 日期<='"+ end + "'";
    command.ExecuteNonQuery();
}
catch (Exception ex)
{
    MessageBox.Show("出现错误" + ex.Message);
}

finally
{
    command.Dispose();
    connetion.Close();
}

```

4.4.2.3 统计图表的实现

向用户展示统计结果表格和饼状图的前提是，统计出所需的各项时间参数，包括①工作总时间、②机床开机时间、③机床关机时间、④机床加工时间、⑤机床待机时间等。各项时间参数的统计方法如表 4-2 所示。

表 4-2 各项时间参数的统计方法

参数名	所需数据表	具体操作
工作总时间	企业日历表	1) 根据用户选定的统计日期区间,选择表中符合条件的记录。 2) 对各条记录中的“工作时间”进行求和,得总工作时间。
机床开机时间	采集时间表	1) 根据用户选定的日期区间,选择中表中符合条件的记录。 2) 计算各条记录中“机床关机时间”-“机床开机时间”。 3) 对上述差值进行求和
机床加工时间	采集时间表	1) 根据用户选定的日期区间,选择中表中符合条件的记录。 2) 计算各条记录中“程序加工结束时间”-“程序加工开始时间”。 3) 对上述差值进行求和
机床关机时间	无	工作总时间-机床开机时间
机床待机时间	无	机床开机时间-机床加工时间

在程序编制时，得到上述各个时间参数的流程图如图 4-20 所示。

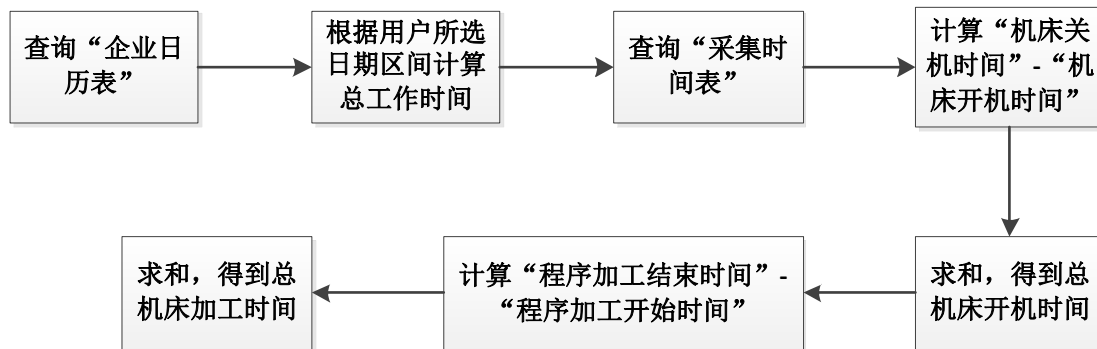


图 4-20 计算时间参数的流程图

界面的数据表格展示可以使用 DataGridView 控件,用上述时间参数构建一个 DataTable 数据表对象作为其数据源。而饼状图的显示,可以使用微软公司推出的 MSChart 控件,此控件可以方便地绘制各种图表,如柱状图、饼状图和曲线图等,并支持平面和 3D 视觉功能。

实现饼状图展示统计结果的部分代码如下:

```

//待机时间=开机时间-加工时间
TimeSpan machineStandBySpan = _machineTime - _programTime;
DataTable chartTable = new DataTable();
chartTable.Columns.Add("时间类型");
chartTable.Columns.Add("秒数", typeof(float));
DataRow newRow = chartTable.NewRow();
newRow["时间类型"] = "机床开机时间";
newRow["秒数"] = _machineTime.TotalSeconds;
chartTable.Rows.Add(newRow);
newRow = chartTable.NewRow();
newRow["时间类型"] = "机床关机时间";
newRow["秒数"] = totalWorkHours * 3600 - _machineTime.TotalSeconds;
chartTable.Rows.Add(newRow);
//dataGridViewStatisticsTable.DataSource = chartTable;
chart1.Series[0].Label = "#PERCENT{P}"; //百分比形式
chart1.Series[0].Points.DataBind(chartTable.DefaultView,
    "时间类型", "秒数",
    "LegendText=时间类型,YValues=秒数,Tooltip=秒数");
chart1.Series[0].ChartType = SeriesChartType.Pie;
chart1.Series[0].ToolTip = "#LEGENDTEXT:#VAL{C} MILLION";
chart1.ChartAreas[0].Area3DStyle.Enable3D = true;
chart1.DataBind(); //绑定数据源
  
```

4.5 基于嵌入式适配器的数据采集系统的测试与运行

整个数据采集系统开发完成之后,需要依据第 2 章中的功能需求分析对系统进行测试,以验证其是否实现所需的设计功能。下面对嵌入式适配器进行了通信和数据采集两个方面的测试,对客户端数据管理系统进行了数据查询、数据管理和数据统计处理两个方面的测试。每项测试分为测试设备、测试环境、测试步骤及测试结果四部分内容。

4.5.1 嵌入式适配器与服务器无线通信测试

测试设备: 嵌入式适配器适配器、无线 Wifi 网卡、台式电脑。

测试环境: 实验室环境, 带宽 100Mbps 的无线 Wifi 网络。

测试步骤:

(1) 打开台式机及嵌入式适配器, 并运行嵌入式适配器的数据采集软件。

(2) 输入嵌入式适配器软件的用户名和密码，以进行登录。如图 4-21 所示。

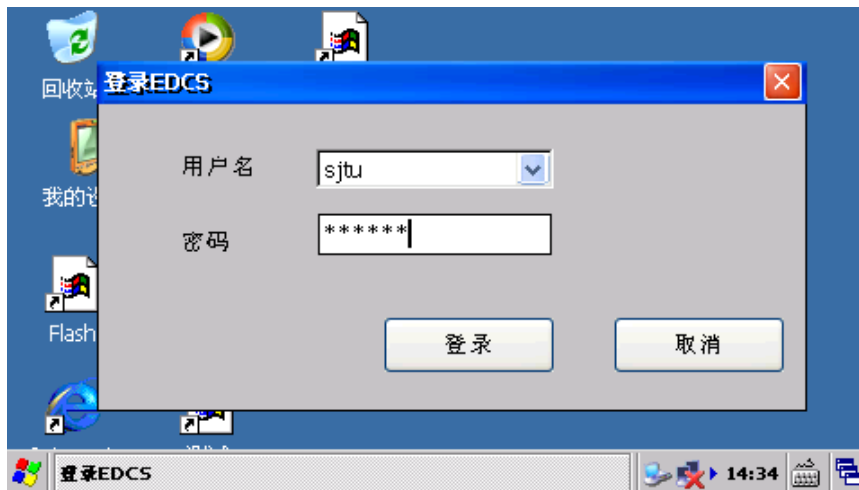


图 4-21 嵌入式适配器数据采集软件登录界面

- (3) 进行“数据通信”界面，并输入服务器 IP、虚拟目录、用户名和密码。
- (4) 选择某一数据表，如“采集时间表”，然后点击“上传”。
- (5) 选择某一数据表，如“设备信息表”，然后点击“下载”。如图 4-22 所示。



图 4-22 数据通信测试界面

测试结果：嵌入式适配器和服务器均可以接收到对方更新的数据并可以查询这些数据，说明嵌入式适配器和服务器可以成功地实现无线通信和远程数据访问。

4.5.2 嵌入式适配器数据采集测试

测试设备：嵌入式适配器适配器、带 RS232 串口的 FANUC 系统数控机床、串行通信线

测试环境：工程训练中心

测试步骤：

- (1) 用串行通信线连接数控机床和嵌入式适配器的串口。其串口接线表如表 4-3 所示。

表 4-3 嵌入式适配器和数控机床串口接线表

嵌入式适配器 (9 针串口)		数控机床 (25 针串口)	
针号	缩写	针号	缩写
2	RXD	2	TXD
3	TXD	3	RXD

嵌入式适配器 (9 针串口)		数控机床 (25 针串口)	
针号	缩写	针号	缩写
5	GND	7	GND
6, 8, 20 针连接; 4, 5 针连接			

- (2) 打开嵌入式适配器的数据采集软件, 进行“数据采集”界面。手动提交机床的开机时间。如图 4-23 所示。

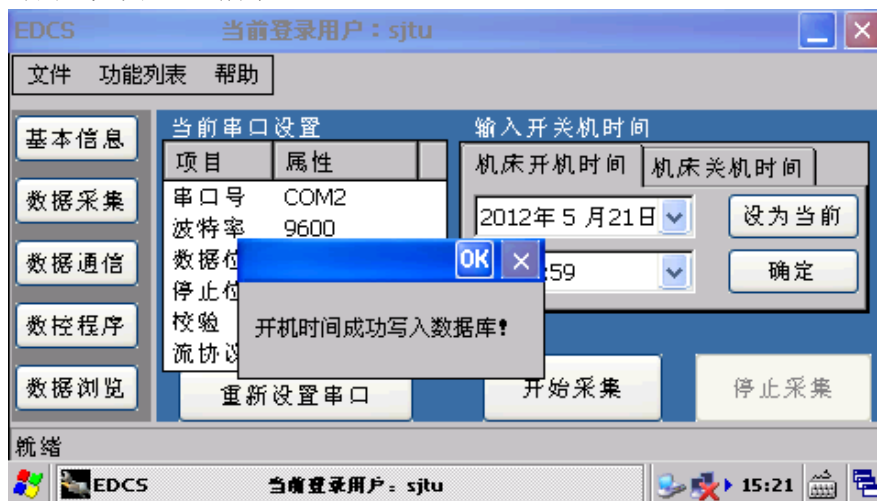


图 4-23 提交机床开机时间界面

- (3) 根据数控机床的串口参数设置嵌入式适配器的串口参数, 如图 4-24 所示。



图 4-24 嵌入式适配器串口参数设置界面

- (4) 设置好串口参数后, 点击“开始采集”开始数据采集。数控机床则开始运行内嵌有宏采集指令的数控程序。
 (5) 数控程序运行结束后, 点击“停止采集”以停止嵌入式适配器的数据采集。
 (6) 机床关机时, 手动提交机床关机时间。

测试结果: 可以在嵌入式适配器的嵌入式数据库中查询到新采集的机床数据, 如图 4-25 所示。说明数据采集成功。



图 4-25 机床采集数据查询界面

4.5.3 客户端数据管理系统数据查询与管理功能测试

测试设备：台式电脑

测试环境：实验室，台式电脑上装有 .NET Framework 3.5

测试步骤：

- (1) 输入用户名和密码，以管理员用户身份登录客户端数据管理系统。如图 4-26 所示。

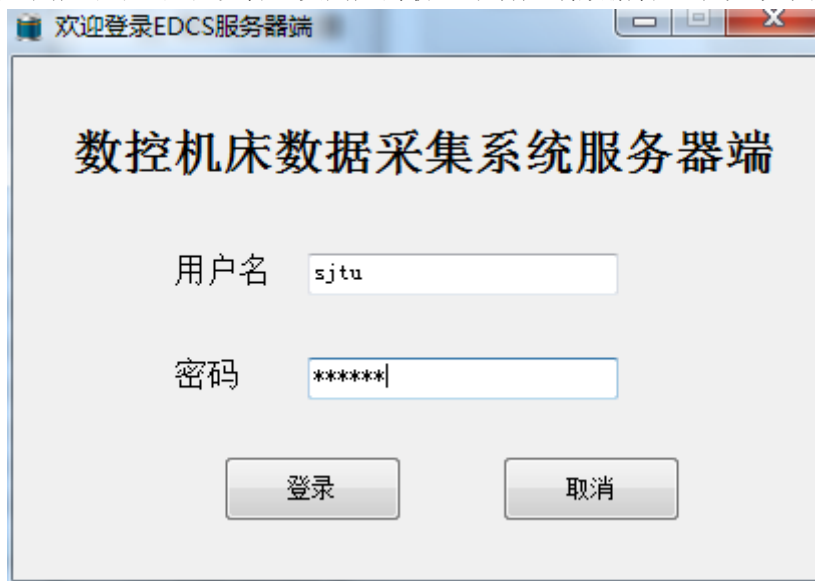


图 4-26 客户端数据管理系统登录界面

- (2) 以查询“采集时间表”为例，首先选择要查询的表，然后选择查询的时间区间，最后选择车间和设备。其中车间和设备也可以在机床树中选择。即可实现数据查询。如图 4-27 所示。

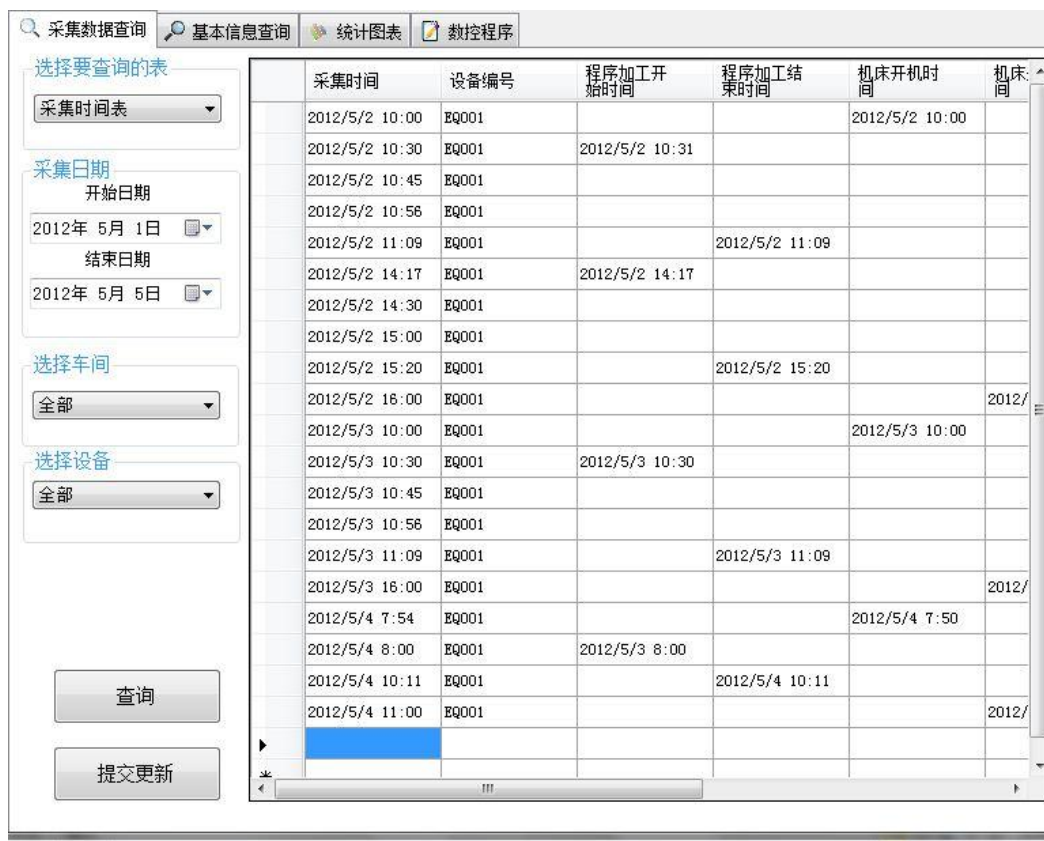


图 4-27 客户端数据管理系统采集数据查询界面

- (3) 单击工具栏或者菜单中的“设备管理”图标，弹出设备管理窗口。在此窗口中可以实现设备信息的增删以及修改，并将更新保存到数据库中。如图 4-28 所示。



图 4-28 客户端数据管理系统设备管理窗体界面

测试结果：成功实现对机床采集数据以及车间、设备、用户信息等数据的查询和增删、修改等操作。

4.5.4 客户端数据管理系统数据统计处理功能测试

测试设备：台式电脑

测试环境：实验室，台式电脑上装有 .NET Framework 3.5 并联网

测试步骤：

- (1) 输入用户名和密码，以管理员用户身份登录客户端数据管理系统。
- (2) 进行“统计图表”页面，选择车间、设备以及采集的日期区间。依次点击“统计”及“图表”按钮。如图 4-29 所示。

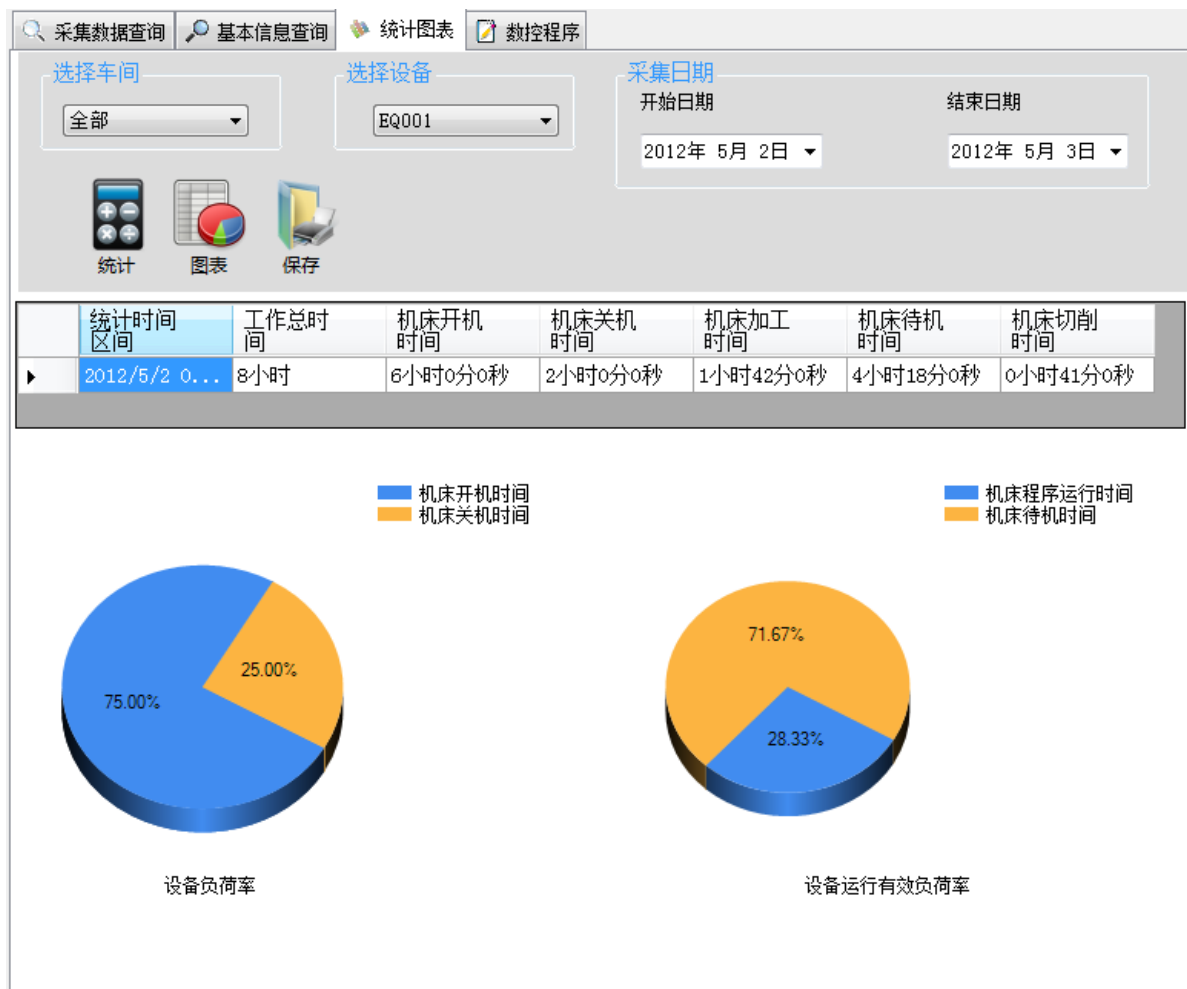


图 4-29 客户端数据管理系统统计图表界面

- (3) 点击“保存”按钮，弹出保存文件对话框，选择保存路径后，将图表保存成本地文件。

测试结果：成功实现对机床采集数据的初步统计处理，并可以得到以图表形式呈现的相关效率参数。

4.6 本章小结

本章首先设计了客户端数据管理系统的各功能模块，介绍了其主要功能。在此基础上，进行了数据管理系统开发关键技术的研究，并设计了数据管理系统良好的图形用户界面，讲解了关键模块的具体实现。最后，利用 .NET 平台和 C# 语言，实际开发了运行了远程客户端

的数据管理系统。在包括嵌入式适配器、服务器和远程客户端数据管理系统在内的整个数控机床数据采集系统开发完成后，对整个数据采集系统进行了测试与运行，并介绍了具体的测试步骤与测试结果，验证了系统的可行性。

第5章 总结与展望

制造业信息化的五个特征是：管理数字化、设计数字化、生产过程数字化、制造装备数字化以及以前四个特征为基础的企业数字化。而数据采集技术是实现管理数字化和生产过程数字化的支撑技术，是 DNC 以及 MES 系统实施的前提，也是最终实现企业数字化并推动我国制造业信息化的重要力量。

由于目前我国大多数数控机床并不具备数据采集功能，从而影响了企业的数字化管理，降低了数控机床的利用效率。结合当前迅速发展的嵌入式技术，在分析现有各种数控机床数据采集技术的基础上，本文提出了基于嵌入式平台和机床串口、利用数控程序内嵌宏输出指令来实现数据采集的数控机床数据采集方法，进行了硬件平台和软件系统的设计开发工作。具体工作如下：

- (1) 总结和分析了当前各种数据采集方法，并结合嵌入式平台制定了基于宏采集指令和机床串口的数控机床数据采集方案，在此基础上设计了数控机床嵌入式数据采集系统的总体架构。
- (2) 设计了嵌入式硬件平台，并根据实际需要，选择了合适的处理器、对外接口以及其他扩展模块，最后选择了嵌入式操作系统，并进行了操作系统的定制和移植。
- (3) 以 SQL Server 2008 和 SQL Server Compact 3.5 作为后台数据库，设计了数据库的结构，并利用远程数据访问技术(RDA)实现了嵌入式适配器与服务器的无线通信。
- (4) 利用 C#语言和.NET 平台技术，进行了嵌入式适配器数据采集软件的开发，并结合串口通信技术实现了对机床开关机时间、程序加工开始时间、程序加工结束时间、主轴转速、主轴坐标等一系列机床数据的采集。
- (5) 利用 C#语言和.NET 平台技术，进行了远程客户端数据管理系统的开发，实现了对各个嵌入式适配器所采集数据的查询、统计功能，得出对企业管理层有用的效率参数，并实现了对用户和数控程序等信息的管理功能。

本系统经过多次调试和测试，各部分基本满足了设计要求，但由于开发时间及个人技术水平所限，系统中还存在较多不足的地方有待进一步的研究与完善。主要包括：

- (1) 系统的实时性有待提高。通过机床串口和宏输出指令，嵌入式适配器可以实现数据的实时采集。嵌入式适配器向上位机传送数据是利用远程数据访问技术(RDA)。由于在程序设计中，RDA 一般是在更新多条数据记录后，一次性上传更新的数据，以减轻系统负担和网络流量，但这样也导致服务器端数据库难以实时获得采集到的机床数据。可考虑采用直接利用基于 TCP/IP 协议的网络通讯来实现数据的传输。
- (2) 本数据采集系统主要适用于提供宏输出指令的数控系统，一定程度上缺乏通用性。
- (3) 客户端数据管理系统对大量机床数据的处理较为简单，有必要对这些数据进行深入挖掘，以获得更多、更具有参考意义的信息和结论，从而为管理者提供更加全面的生产信息。

参考文献

- [1] MESA International. WhitePaper3. Pittsburgh: Manufacturing Execution Systems Assoc, 1997.
- [2] 刘一臻. 机械制造公司生产管理信息系统的分析与设计[D],[硕士学位论文]. 大连: 大连理工大学, 2005.
- [3] 肖士利. 数控机床状态数据实时采集与监视系统的研究开发[D],[硕士学位论文]. 南京: 南京航空航天大学, 2008.
- [4] Hiromitsu Shimakawa, George Ido. Real-Time Reaction in Supervisory Control According to Data Freshness [J]. Real-Time System, 2001, (3): 197~201.
- [5] Nunzio M. Torrisi, Joao F.G. Oliveira. Remote control of CNC machines using the CyberOPC communication system over public networks[J]. Int J Adv Manuf Technol, 2007, 23: 125~129.
- [6] Ferraz Jr.F, Coelho R.T. Data acquisition and monitoring in machine tools with CNC of open architecture using Internet[J]. International Journal of Advanced Manufacturing Technology, 2005, 26(1): 90-97.
- [7] Sanjeev Kumara, Aydin Nassehia, Stephen T. Newmana 等. Process control in CNC manufacturing for discrete components: A STEP-NC compliant framework[J]. Robotics and Computer-Integrated Manufacturing, 2007, 23(6): 667-676.
- [8] 金明东. 制造执行系统 MES 中数据采集与传送的研究应用[D],[硕士学位论文]. 上海: 上海交通大学, 2007.
- [9] 王姣, 宫政. 基于嵌入式系统的数控机床信息采集与监测技术的研究[J]. 机床与液压, 2010, 38(8): 90-92.
- [10] 徐永乐. 基于 ARM9 的数控机床数据采集系统的研究与开发[D],[硕士学位论文]. 南京: 南京航空航天大学, 2011.
- [11] 李文川. 数控机床群控系统状态监测及系统集成研究与实施[D],[硕士学位论文]. 重庆: 重庆大学, 2008.
- [12] 付林云. MES 中数控车间数据采集系统的研究和应用[D],[硕士学位论文]. 浙江: 浙江大学, 2008.
- [13] 童大鹏, 张礁等. 嵌入式系统技术研究及其开发实例[J]. 山西电子技术, 2004, (4): 5-7.
- [14] DSP、ARM 和单片机的区别汇总. <http://hi.baidu.com/anothertwo/blog/item/dc7e1e39412183e23a87ce86.html>.
- [15] 马中梅等. ARM 嵌入式处理器结构与应用[M]. 北京: 北京航空航天大学出版社, 2002: 6-9.
- [16] 陈立定, 杨俊辉, 肖力扬. 三种通用嵌入式操作系统的分析与比较[J]. 微计算机信息, 2009, 25(4-2): 89-91.
- [17] 薛大龙等. Windows CE 嵌入式系统开发从基础到实践[M]. 北京: 电子工业出版社, 2008: 24-25.
- [18] OK6410 开发板硬件手册[Z]. 河北: 河北保定飞凌嵌入式技术有限公司, 2011 年.
- [19] 三星 S3C6410 用户手册(中文版)[Z], 2010 年.
- [20] Clarke K.S.P, Kershaw D. System design of a Windows CE ARM based micro-controller[C]. Proceedings - IEEE International Conference on Computer Design: VLSI in Computers and Processors, 1998: 236-241.
- [21] Microsoft SQL Server 2005 Compact Edition 概述[Z]. <http://www.microsoft.com/downloads/zh-cn/details.aspx?displaylang=zh-cn&FamilyID=85e0c3ce-3fa1-453a-8ce9-af6ca20946c>

3#QuickDetails.

- [22] Zerzelidis A. Requirements for a real-time .NET framework[J]. ACM SIGPLAN Notices, 2005,40(2): 41-50.
- [23] 殷泰晖等著.C#编程从基础到实践[M].北京:电子工业出版社,2007:5-6
- [24] FANUC 0 系统编程说明书[Z].北京 FANUC 机电有限公司.2003.
- [25] Jindal Varun. PC-to-PC communication via RS-232 serial port using C[J]. Electronics World, 2006,112(1837): 25-29.
- [26] 杨雪婧,张志彪.SQL Server 2005 Mobile 同步技术初探[J].电脑知识与技术,2010,6(20):5413-5414,5416.
- [27] Qi Xiangyang, Lin Shuzhong, Cui Hui. Research on the remote data collection based SQL server. Journal of Wuhan University of Technology, 2006,28(1): 828-832.
- [28] 王召鹏,陈国金.利用宏指令实现数控机床状态采集方法的研究[J].现代制造工程,2006,(11):134-136.
- [29] SqlCeRemoteDataAccess 方法[Z]. [http://msdn.microsoft.com/zh-cn/library/system.data.sqlserverce.sqlceremotedataaccess_methods\(v=vs.80\)](http://msdn.microsoft.com/zh-cn/library/system.data.sqlserverce.sqlceremotedataaccess_methods(v=vs.80)).
- [30] 李长星. VB 环境下数控系统和上位机的通信实现[J].西安石油学院学报(自然科学版), 2002,17(4):74-76.
- [31] 崔群法.SQLServer2008 从入门到精通[M].北京:电子工业出版社,2009:8-10.
- [32] 薛大龙等.Windows CE 嵌入式系统开发从基础到实践[M].北京:电子工业出版社,2008:24-25.
- [33] Castro Pablo, Melnik Sergey, Adya Atul. ADO.NET entity framework: Raising the level of abstraction in data programming[C]. Proceedings of the ACM SIGMOD International Conference on Management of Data,2007: 1070-1072.
- [34] 马娟. .NET 数据绑定技术的分析与研究[J].信息安全与技术,2011,(11):47-48,51.
- [35] 数据绑定概述[Z]. <http://msdn.microsoft.com/zh-cn/library/ms752347.aspx>.
- [36] 李波.基于串口的 DNC 信息采集系统的研究[D],[硕士学位论文].南京:南方航空航天大学,2007.

谢辞

值此论文完成之际，我的本科求学生涯也即将画上句号。在此，向所有关心、帮助我的学院领导、老师、同学和我的父母致以诚挚的谢意。

首先，要感谢我的指导老师张洁教授。严格来说，这是我第一次做正规的研究和写正规的论文。而张老师除了在课题定位和知识补充上给予我实际的帮助外，更重要的是教会了我应该如何做科研、如何进行一个课题的系统流程。正所谓授人以鱼不如授人以渔，现在来看，实际的知识固然重要，但学习新知识、探索新领域的方法则更为珍贵，而张老师则教给了我这些方法。另外，张老师严谨有序的治学态度和一丝不苟的科研精神，也让我感受到一位做科研者应有的态度，让我受益匪浅。总之，在实验室跟张老师做毕业设计的这一学期，为毕业之后的继续读研深造奠定了良好的基础。

感谢跟随张老师的各位研究生、博士生学长和学姐，特别要感谢杨俊刚、吕佑龙和夏至三位学长。在做毕业设计期间，由于个人能力及知识所限，遇到了一些较大的困惑和困难，正是由于各位学长的帮助和自己的刻苦努力，才能够顺利完成整个毕业设计。鉴于自己也要读研，我一定会传承各位学长无私助人的精神，并将其发扬光大。

感谢所有被引用文献的作者，他们的研究工作使我开阔了视野，获得了思路，而我则站在巨人的肩膀，努力向前迈出了新的一步。

感谢 F0802002 的各位同学以及所有支持、关心我的朋友们，我们的友情会成为一生的财富。

感谢我的父母和哥哥、姐姐，他们在生活上给予我支持，在精神上给予我鼓励，他们无私的关爱是我在学业上不断探索前进的无限动力。祝我的父母和哥哥、姐姐永远身体健康，幸福快乐！

最后，在本文结束之际，向所有为我的论文提出宝贵意见的评阅专家们表示衷心的感谢！