

西华大学

硕士学位论文

基于ARM&WinCE的刀具状态监测数据处理平台设计

姓名：徐全

申请学位级别：硕士

专业：机械电子工程

指导教师：胡丹

20090401

基于ARM&WinCE的刀具状态监测数据处理平台设计

机械电子工程专业

研究生 徐全

指导老师 胡丹

刀具状态的精确监测是保证金属切削加工过程顺利进行的关键，因此研制准确、可靠且成本低廉的刀具状态监测系统一直是研究人员所追求的目标。在众多刀具状态监测方法中，声发射监测技术，以其信号直接来源于切削区，具有灵敏度高、响应快，能有效避开低频干扰等优点，非常适用于刀具状态监测。

围绕如何获取高信噪比的刀具状态信号特征，拟结合嵌入式技术，构建准确、稳定、低成本的实时刀具状态监测与辨识系统。给出了基于 ARM & WinCE 平台的刀具状态监测系统数据处理平台软硬件初步解决方案。作为课题的前期研究，本文主要进行了以下工作：

- (1) 分析了声发射信号与刀具磨损状态的相关性，验证了利用声发射信号进行刀具状态监测的可行性；
- (2) 确定刀具状态监测系统的整体方案，包括系统整体架构、软硬件设计方案。ARM & WinCE 构成本系统的数据处理与显示平台，EVC 为图形界面应用程序开发工具；
- (3) 构建了数据处理与显示平台。选用 MagicARM2410 实验开发平台，简化了硬件设计；根据系统的功能需求，进行 ARM 平台的接口设计、操作系统和必要的驱动程序的剪裁及移植；
- (4) 完成了数据处理与显示应用软件设计。系统软件包括界面模块、数据管理模块、数据处理模块、图形及结果显示模块、参数设置模块等，其中数据处理模块主要包括小波消噪、小波包分解特征提取等算法；
- (5) 实现了 ARM & WinCE 平台与 PC 机的实时可靠通讯。

关键词：刀具状态监测， 声发射， WinCE， 小波消噪， 小波包

Design of Tool Condition Monitoring Signal Processing Platform Based on ARM&WinCE

Major: Mechatronic Engineering

M.E. Candidate: XuQuan

Supervisor: HuDan

The precise monitoring of the tool condition is the key to guarantee the metal cutting procedure. There is always the researchers' target that how to develop an exact, reliable and low cost tool monitoring system. Compared with the other tool monitoring technologies, the acoustic emission technology is more suitable to the tool monitoring because it has good performances such as, high sensitiveness, respond quickly, and can avoid low frequency disturbances.

According to those reasons, this paper research how to avoid the defects and utilize acoustic emission signal's performances and eventually establish a simple, reliable and low cost tool wear monitoring and recognize system. Proposed a set of hardware and software solutions based on ARM&WinCE. Primarily works for the following:

- (1) Describes the correlation of the acoustic emission signal to tool wear condition, and it is proved that the tool condition is corresponding to the acoustic emission signal;
- (2) Blue print for the tool monitoring system is studied, including the structure of the whole system, the designing proposal of hardware and software. Data processing and display platform are made up of ARM and WINDOWS CE. EVC(Embedded Visual C++) is used as graphical interface development tool;
- (3) Establishes the platform of the signal processing and display. To simplify the hardware design, the MagicARM2410 experimental development platform is selected. In accordance with the requirements of system performance, the interface for ARM is designed. And the operating system and the software of drive is tailored and transplanted to ARM;

(4) Carries out the application software design of the signal processing and display platform. Base on the platform of EVC, the paper carries out the modules of the settings of parameters, management of data, signal processing and display et al; The Wavelet and Feature extraction Algorithm is primarily stressed on;

(5) Real-time and reliable communication is realized between WINDOWS CE operating system on ARM and Personal Computer.

Keywords: Tool Condition Monitoring, Acoustic Emission, WinCE, Wavelet Denoising, Wavelet Packet

声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得西华大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示感谢。

本学位论文成果是本人在西华大学读书期间在导师指导下取得的，论文成果归西华大学所有，特此声明。

作者签名：徐全 09年6月16日
导师签名：胡丹 09年6月16日

西华大学

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，西华大学可以将本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、缩印或扫描等复印手段保存和汇编本学位论文。

本学位论文属于

- 1、保密，在 年解密后适用本授权书；
- 2、不保密 适用本授权书。

(请在以上内划√)

学位论文作者签名：徐全 指导教师签名：胡丹
日期：2009. 6. 16 日期 09. 6. 16

1 绪论

1.1 刀具状态监测的研究目的及意义

在传统的机加工环境中，以人为主体的，对于机械制造过程工况状态的判断很容易进行，通常是对加工过程中产生的异常噪音或者所得产品的质量对工况进行估计。随着柔性制造系统和集成制造系统的发展，机械加工的自动化程度日益提高，要求加工系统能自动地对生产过程中所出现的故障进行有效的在线监测。作为机加工过程中最基本的加工要素加工刀具，由于切削加工条件的多样性、切削参数的多变性、刀具破磨损的随机性和复杂性等因素使得刀具的监测成为整个生产过程监测的重要环节。一旦刀具严重磨损或破损，将给生产带来不同程度的损失，对于严重的损失，会引起系统总体技术经济效益的下降。因此，刀具状态的实时、在线监测是实现机械制造自动化及保证生产顺利进行的关键。

1.2 刀具状态监测的发展现状、趋势以及问题的提出

通常，一个完整的刀具状态监测系统由研究对象(刀具)、加工条件、传感器检测、信号处理、特征信号提取及模式识别等模块组成，如图 1.1 所示。

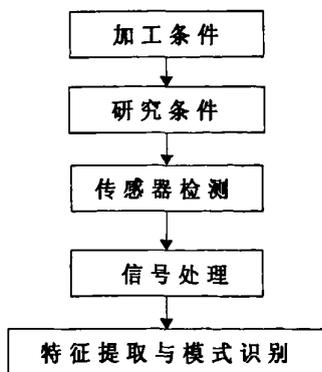


Fig1.1 Tool condition monitoring system configuration

图 1.1 刀具状态监测系统框图

设计一个监测系统必须首先确定研究对象，比如铣刀的磨损监测等；其次确定监测系统的实际监测范围，理想情况下希望监测系统在任意加工条件下都能准确检测刀具破磨损状态，即在线监测；监测系统的传感器模块包括传感器的选择、安装以及信号的放大和传送；信号处理模块通过研究信号的时域特性和频域特性，确定提取反映刀具破磨损状态的特征量；特征提取及模式识别模块利用合适的数学方法选择能够反映刀具状态变化的敏感特征，通过人工智能的分析方法，实现对刀具状态的分类和刀具破磨损的精确监测。

在功能上，刀具状态监测系统主要由传感器数据采集和特征提取及模式识别两部分组成。刀具状态监测技术也从这两个方向分别进行各种方法的研究。

根据数据采集采用传感器的种类，目前，国内外所采用的自动监测刀具状态的方法可归为两大类，一类为直接监测法，直接法是直接监测刀具形状、位置等参量，一般只能离线监测，它包括接触电阻、光学法、光导纤维和电视摄像等；另一类为间接监测法，间接法是间接测量切削过程中的某些参数，将它与给出的正常切削参数进行比较，从而推断出刀具的磨损程度或者刀具要发生破损的先兆，达到监测的目的，它包括切削力、扭矩、主轴电机功率、切削振动、切削温度、工件尺寸、工件表面粗糙度、声发射、切削声信号的测量法等^{[1]-[10]}。

声发射检测技术作为一门成熟的检测方法广泛地应用在石油化工工业、电力工业、材料试验、民用工程、航天和航空工业、金属加工、交通运输业等领域，而且应用领域正在生物等领域延伸。

声发射监测法的研究是自 70 年代开始的，各国相继开展了广泛的研究。在切削加工时，金属材料将发生变形和断裂，声发射(Acoustic Emission 简称 AE)不同于可听声。它是材料的某个局部区域受外力或内应力作用发生塑性变形或破坏之际，将其内部储存的应变能以瞬态弹性波的形式迅速释放出来的一种物理现象。当材料受力时，首先发生弹性变形，此变形是以弹性应变能储存在材料中，使其内部存在微观结构的缺陷，导致了局部应力集中，造成不稳定应力。当这种不稳定应力分布状态下的应变能积累到一定程度时，材料出现快速相变、裂纹等现象，并在此过程中释放出应变能。其中部分由表面能量吸收产生新裂痕部分增加塑性变形区域部分以弹性应力波的形式快速向外释放出来，这种能量释放现象称为声发射。它是材料缺陷正在扩散的指示器，是检测材料损伤，

特别是早期损伤的有力工具。

声发射监测法的基本传感器是一个高频压电传感器，相应频率带宽为几 KHz 到 1MHz。当刀具破损或磨损时，AE 信号在时域或频域将发生变化，以此来判断刀具的状态。目前虽然有许多国家着手 AE 刀具监测装置的研究开发，但真正能用于生产中的产品还不多，原因在于，由于 AE 传感器的灵敏度较高，切削过程中除了工件材料剪切变形形成切屑时产生的 AE 信号是判断刀具剧烈磨损或破损的依据外，切屑之间、切屑与工件、机床的碰撞以及轴承、齿轮的运动和周围的噪声都会产生干扰信号。由于干扰信号的强度与剪切区 AE 信号的强度相差并不是太大，因此如何从这些合成信号中提取刀具破损、磨损信息是 AE 监测技术中关键且困难的问题。

基于声发射信号的刀具状态监测技术，国内外的很多科研院所都对其开展了研究，并取得了一定的成果。

I.Tansel^[11] 等人利用声发射信号来监测刀具磨损；文献^[12]用声发射信号建立一个综合刀具监测系统；Pan Fu^[13]同时采用声发射信号、力信号、振动信号的多特征融合技术，较好地实现了刀具磨损量的自动分级；清华大学利用固体材料在发生变形、断裂和相变时迅速释放变能而产生的声发射信号进行刀具监测。王忠民^[67]研制成功了可用于刀具磨损监测的高灵敏度双叠片式流体声发射传感器，满足自动化加工过程中刀具磨损状态在线自动监测。这些研究都表明，声发射检测技术应用在刀具状态监测领域有着别的监测技术无法比拟的效果。

综上，声发射技术是一种动态的无损检测技术，主要优点如下^[14,67]：

(1) 是动态的无损检测，可以在结构或部件实际操作或运转中受力情况下进行实时检测和监督，反映物体缺陷的动态信息，这是声发射检测技术与其它检测技术的根本区别。声发射技术所具有实时性和在线性为声发射检测技术的迅速发展和广泛应用提供了有力保证；

(2) 声发射技术是一种整体状态检测技术。通过按一定阵列布置少量固定不动的传感器，声发射仪就可获得被检测对象中生源在检测过程中的一切活动信息，可以直接检测到试件本身的弹性波，并可确定声源的位置，由传感器加以接收，这为实际检测工作带来极大方便；

(3) 声发射探头工作时不需移动、操作简便、灵敏度高、不受材料种类的限制，适用范围广泛。除了极少数材料外，黑色金属、有色金属、复合材料、塑

料、木材、岩石和混凝土等各种固体材料都有声发射现象，因此都适合采用声发射检测技术；

(4) 声发射检测可以避开切削过程中振动和音频噪声污染严重的低频区，在所感兴趣的频率范围内灵敏度较高；能抵御一定范围内切削用量的变化；同时，由于检测信号反映的是金属内部晶格的变化情况；故有可能较早地对刀具破损情况进行预报。

利用声发射检测技术检测刀具的切削状态，信号中包含刀具各种切削状态特征。所采集到的信号适合于计算机控制；可以在原理工作区检测刀具切削状态信号，从而减少仪表的损坏的危险；并且声发射信号的频率很高，远远超过机械振动和环境噪声的频率，这都有利于提取到有效的刀具切削状态信息。

在数据处理、特征提取及模式识别方面，计算机技术、人工智能技术及嵌入式技术的飞速发展，推动了刀具状态监测技术的进一步发展。目前，各种信号处理及模式识别的算法也应用到刀具状态监测中，并且，被认为是刀具状态监测最具前途的领域。诸如，数字滤波、FFT、小波分析、小波包分析、支持向量机、模糊推理、神经网络、遗传算法以及专家系统^[65]等等。但是，有些算法只是适用于某一特定方面，有些还只是处于理论研究或者实验室阶段，离实际运用还有一段距离。而且，这些算法大多只能在高性能的 PC 机上实现，也就是说很难达到在线、实时地监测，并且成本也较高。总体来说，还有很长的路要走。本文拟在 ARM & WinCE 嵌入式开发平台实现小波消噪、小波包特征提取等算法，构建准确、稳定、低成本的实时刀具状态在线监测与辨识系统。

1.3 课题来源及主要研究内容

1.3.1 课题来源

本课题在四川省科技厅重点项目“支持向量机在刀具状态监测中的应用”（项目编号：07ZA118）下展开。实验室前期做了大量特征提取及模式识别算法研究，本文为在 ARM&Windows CE 平台构建刀具状态监测系统的前期研究，给出基于 ARM&Windows CE 的刀具状态监测系统数据处理平台的初步软硬件解决方案，为进一步完成系统设计奠定基础。

1.3.2 本文研究的主要内容

本课题研究的目的是实现刀具状态的自动化监测与分析，并将测量的结果显示出来，以便帮助用户及时了解刀具的磨损状态，必要时将重要数据传送到上位机进行模式识别等进一步处理，便于今后的统计和总体分析。本课题的主要工作内容如下：

- (1) 分析声发射信号与刀具磨损状态的相关性，验证利用声发射信号进行刀具状态监测的可行性。
- (2) 确定刀具状态监测系统数据处理平台的整体方案，包括系统整体架构、软硬件设计方案。ARM & WinCE 构成本系统的数据处理与显示平台，EVC(Embedded Visual C++)为图形界面应用程序开发工具；
- (3) 构建数据处理与显示平台。选用 MagicARM2410 实验开发平台，简化了硬件设计；根据系统的功能需求，进行 ARM 平台的接口设计、操作系统和必要的驱动程序的剪裁及移植；
- (4) 数据处理与显示应用软件设计。系统软件包括界面模块、数据管理模块、数据处理模块、图形及结果显示模块、参数设置模块等，其中数据处理模块主要包括小波消噪、小波包分解特征提取等算法；
- (5) 实现 ARM & WinCE 平台与 PC 机的实时可靠通讯。

2 刀具监测系统的总体方案设计

2.1 系统总体框架

刀具状态监测系统分为三个模块：数据采集模块、数据处理与显示模块、模式识别模块。数据采集模块负责切削声发射信号的采集，并传送到数据处理与显示模块，关系到整个系统的精度和抗干扰能力；数据处理与显示模块，是系统软件开发平台，在 ARM&WinCE 平台上构建，负责信号的处理(如小波消噪、信号特征提取等)，并将结果显示出来，得到实时的刀具磨损状态，同时进行必要的备份，以便必要时传送到 PC 端，是本文研究的重点；模式识别模块 (PC 端)，接受从数据处理与显示模块传来的数据并做诸如模式识别等进一步的处理与分析。系统最终完成时功能模块如图 2.1 所示，本文进行了整体方案的论证与设计，重点研究了数据处理与显示模块，给出了其初步软硬件解决方案，但并未对数据采集模块及模式识别模块做详细论述。

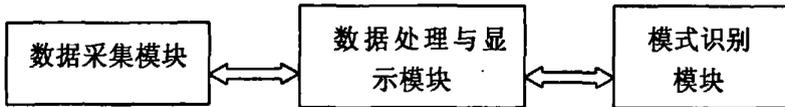


Fig.2.1 System function schematic drawing

图 2.1 系统功能模块示意图

2.2 系统硬件功能结构

整个系统硬件功能结构如图 2.2 所示：

系统的核心是 ARM9 开发平台，它负责采集到信号的处理与显示以及与 PC 机的可靠通信，具有很强的实时性、可靠性和抗干扰性。本设计利用声发射传感器获得刀具状态信号，经必要的前置放大、初级滤波，然后传送到数据处理显示平台。数据处理显示平台将获得的数据进行相关处理与实时显示波形，对刀具进行实时状态监测。同时进行必要的备份，以便必要的时候传输到监控中心供统计分析。

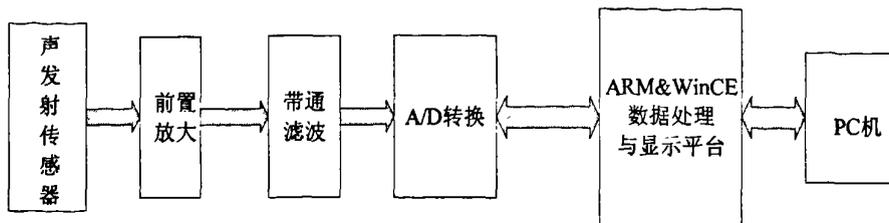


Fig.2.2 System hardware schematic diagram

图 2.2 系统硬件功能结构

2.3 数据处理与显示平台软件功能结构

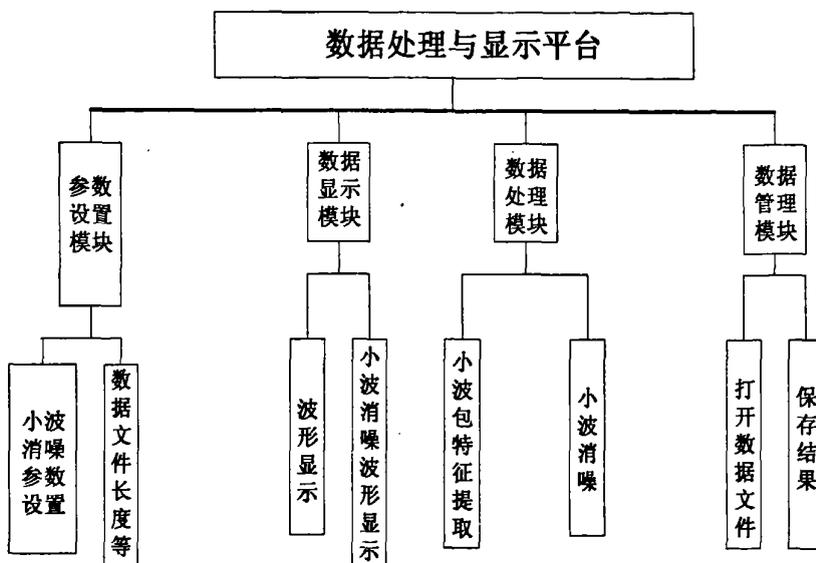


Fig.2.3 Software platform of signals processing and display

图2.3 数据处理与显示平台软件功能结构

数据处理与显示平台软件功能结构如图 2.3 所示，主要由以下几个部分组成：

- ①界面模块。主要响应用户的各种操作命令，包括输入必要的数据、显示波形、计算结果，以及启动、停止系统等；
- ②数据处理模块。根据设置的参数，进行数据文件的小波消噪、消噪后信号的特征提取；
- ③显示模块。它提供信号时域波形显示、小波分解与重构后的各层逼近信

号及细节信号显示;

④数据管理模块。打开保存的波形文件,供小波消噪等使用,并将分析结果及计算结果保存,必要时将保存的结果上传到监控中心,便于统计分析等;

⑤参数设置模块。打开文件前设置好数据文件的长度、小波分解的层数等参数,小波分解与重构后,可以选择要显示的逼近信号及细节信号的层数。

3 刀具状态信号特征提取

本文中，对采集到的声发射信号进行特征量的提取可以分为两个步骤：

- (1) 声发射信号的预处理。对采集到的声发射信号利用小波分析进行消噪。
- (2) 声发射信号特征量的提取。通过小波包分解将各个频带的能量特征提取出来。

3.1 声发射信号处理方法

声发射信号是一种复杂的波形，包含着丰富的声发射源信息。声发射信号处理方法是声发射检测技术中的重要环节，如何选择合适的信号处理方法来分析声发射信号，从而获取正确描述被检测对象的特征，是声发射检测技术发展中的难点和热点。目前，声发射信号处理方法有参数分析、频谱分析、小波分析等。参数分析方法的应用比较成熟，但从理论上讲，后面两种方法能够提供比参数分析更多的信息量。

参数分析法是分析声发射信号最普遍也是历史最悠久的方法，通过对测得的声发射信号进行初步的处理和整理，变换成不同的声发射参数来对声发射源的特征进行分析和处理。声发射参数一般是通过硬件来获取，常用的声发射参数有振铃计数、能量、上升时间、脉冲持续时间、幅度分布、事件等^[14]。尽管声发射参数能提供声发射源特征的相关信息，但是由于参数的取舍存在较大的随意性，声发射技术应用试验条件不同，研究对象不同，选用参数不同，得到的对声发射源的评价也会不同^[15,16]，这些都是影响参数分析法在声发射技术中应用的一个重要因素。而且参数分析比较适合容易找到参数值的瞬态声发射信号，而刀具状态的声发射信号为连续型的声发射信号声发射，因此在本论文中不考虑使用参数分析。

频谱分析是声发射信号处理中常用的波形分析方法，通过把声发射信号从时域转换到频域，在频域中研究声发射信号的特征，通过分析声发射信号的频域分布特性提取能反映和突出声发射源信息的特征。大量的文献报导，人们在建立声发射信号的频谱分布特性与声发射源信息的对应关系方面做了大量的研究工作^[15-19]，并取得了许多有意义的结论。比如 De Groot 在对环氧复合材料四

类损伤模式的声发射信号的特征研究中发现不同损伤模式所产生的声发射信号的频谱分布各不相同，每一类损伤模式对应一定的特征频率分布范围。大量的实践表明声发射信号的频谱能在一定的程度上反映声发射源的特征，不同声发射源的声发射信号的特征可以通过频谱分布信息表现出来^[17]。但是对信号进行频谱分析是建立在一个隐含的前提条件之上，即被分析的信号是周期的平稳信号。而声发射信号是一种频率和统计特征均随时间变化的非平稳随机信号，频谱分析无法对于某个时段的声发射信号的特征进行分析，因此频谱分析不是分析声发射信号特征的有效方法^[18]。

小波分析是一种时频分析方法，它的特点是对信号进行变时窗分析，即对信号中的低频分量采用较宽的时窗，对高频分量采用较窄的时窗，这个特点使得小波分析在时域和频域同时具有良好的局部分析特性，这就非常适合分析具有瞬态性和随机性的声发射信号^[17]。1999年 Suzuki^[20]用快速傅立叶变换、短时傅立叶变换和小波分析等三种分析方法对复合材料微观破坏模式的声发射检测信号进行研究，认为小波分析能提供更多的声发射源信息；2001年 Terchi^[23]对用于声发射信号处理的维纳滤波、短时傅立叶分析、频谱分析、小波变换等信号处理方法进行了全面的综述，指出小波变换是分析瞬态、非平稳、随机信号的有用工具。

3.2 声发射信号的小波消噪

从加工系统中采集到的信号普遍含有各种噪声，它们来自被测信号源本身、传感器、调理电路、外界干扰等。为了获得准确的信号，除了在硬件上采用相应的滤波电路之外，软件上也必须采用一定的算法来降噪。传统的去噪方法等价于信号通过一个低通或带通滤波器，滤掉噪声频率成分。但对于呈现出明显非平稳性的刀具状态检测信号，采用传统的方法处理有着明显的局限性，表现在它不能将有用信号的高频部分和由噪声引起的高频干扰加以有效地区分。若低通滤波器太窄，则在滤波后，信号中存在大量的噪声，若低通滤波器太宽，则将一部分有用信号当作噪声而滤掉了，并且由于加工环境的复杂性，被监测的信号伴随着大量噪声，早期故障特征信号的信噪比低，对于短时低能的瞬态信号，在低信噪比的情况下，经滤波器的平滑，不仅信噪比得不到较大的改善，

而且破损特征信号即突变奇异点的信息也被模糊掉了,从而难以实现非平稳随机信号的信噪分离。而小波变换具有良好的时频局部化特性,可将信号分解到多个互不重叠频段,用小波进行信号的消噪可以很好地保存信号中的尖峰和突变部分即能够有效区分信号突变部分和噪声,从而非常适用于刀具状态检测信号的消噪^[45,46]。

3.2.1 小波分析算法原理

从数学上定义小波:若 $\Psi(t)$ 是一个可测的、平方可积的函数,即具有有限能量 $\Psi(t) \in L^2(R)$,这里 $L^2(R)$ 为 $\Psi(t)$ 的矢量空间, R 为实数集),若 $\Psi(\omega) = \int_{-\infty}^{\infty} \Psi(t) e^{-j\omega t} dt$ 是 $\Psi(t)$ 的傅里叶变换,且满足 $C_{\Psi} = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty$,则称 $\Psi(t)$ 是基本小波或母小波函数^[41,42]。

$\Psi(t)$ 满足: $\int_{-\infty}^{\infty} \Psi(t) dt = 0$ (波动性); $\int_{-\infty}^{\infty} \Psi^2(t) dt < \infty$ (衰减性)

给定平方可积的信号 $x(t)$,则 $x(t)$ 的小波变换 (wavelet transform, WT) 定义为

$$WT_x(a,b) = \frac{1}{\sqrt{a}} \int x(t) \Psi\left(\frac{t-b}{a}\right) dt = \int x(t) \Psi_{a,b}(t) dt = \langle x(t), \Psi_{a,b}(t) \rangle \quad (3.1)$$

式中,时移 b 的作用是确定对 $x(t)$ 分析的时间位置即时间中心,尺度因子 a 的作用是把基本小波 $\Psi(t)$ 作伸缩。即 a 和 b 联合起来确定了对 $x(t)$ 分析的中心位置及分析时间宽度。可以形象地打个比喻,使用镜头观察目标 $x(t)$ (亟待分析的信号), $\Psi(t)$ 代表镜头所起的作用(例如滤波或卷积); b 的作用相当于镜头相对于目标平行移动, a 的作用相当于镜头向目标推进或远离。

小波分解是对信号一层一层进行分解的,每一层的分解都是将原信号分解到两个时频空间中,这两个时频空间内的数据相当于原信号分别经过两个滤波器(一个为低通滤波器,一个为高通滤波器)后,再减点抽样而得到的两个包含不同频率成分的时频信号,再下一层的分解则是对经过低通滤波器的时频空间再进行同样的分解^[41]。

小波变换的逆过程就是小波重构。由小波系数能够完全重构出原函数 $f(x)$,

其重构公式为

$$f(x) = \frac{1}{C_\Psi} \int \int_{R^+ \times R^+} \frac{1}{a^2} W_f(a,b) \Psi_{a,b}(x) da db \quad (3.2)$$

$$C_\Psi = \int_R |\Psi(\omega)|^2 |\omega|^{-1} d\omega$$

式中 $\Psi_{a,b}(x)$ —— 由母小波生成的依赖于参数 a, b 的小波基函数;

$\Psi(\omega)$ —— $\Psi(x)$ 的傅里叶变换

由重构公式可见, 与傅立叶分析相类似, 基于小波变换的信号分析同样是将信号分解成小波基, 以此构成级数来逼近信号函数。不同的是小波基是通过平移和伸缩构成的, 因此具有良好的时频局部化性质。同时小波函数族在若 (R) 中是稠密的, 其线性组合在若 (R) 中也是稠密的, 因此可通过小波函数的线性组合达到最佳的逼近性能来逼近函数。小波分析的主要优点在于能够分析信号的局部特征, 实现分辨叠加在信号上的微小突变^[42]。

3.2.2 小波基函数和分解尺度的选择

由第 3.2.1 节小波函数的定义, $\Psi(t)$ 称为一个基本小波, 小波基函数是小波变换中的一个重要因素, 用小波对声发射信号进行分解首先就要选择合适的小波基和分解尺度^[43]。因为采用不同的小波基函数对同一个信号进行小波变换后获取的信号特征参数也不相同。因此小波基函数的选取对准确提取信号的特征很重要, 是对信号进行小波变换必须考虑的一个关键问题。

按照下列的标准, 小波基函数具有不同的类型

(1) 支撑长度: 当时间和频率趋向无穷大时, 小波基函数 $\Psi(t)$ 和尺度函数 $\Phi(t)$ 从一个有限值收敛到 0 的长度。支撑长度越长, 一般需要耗时更多。支撑长度越短, 时域局部特性越好。

(2) 正交性: 正交性好的小波能在信号重构中获得较好的平滑效果, 减少量化或舍入误差的视觉影响。

(3) 对称性: 对称的小波基函数可以避免信号小波变换的分解和重构过程对信号产生畸变, 因为该小波对应的滤波器具有线性相位的特点。

(4) 消失矩: 要求小波基 $\Psi(t)$ 的 n 阶原点矩为 0。即消失矩越大, 就使更多的小波系数为零或者产生尽量少的非零小波系数, 这样有利于信号消噪, 但在一般情况下消失矩越大, 支撑长度也越长, 计算量也会增加。表 3.1 列出常用的小波基及其主要性质。

根据刀具状态声发射信号的要求, 通过比较分析, 本文选择 Haar 小波、Daubechies 小波、Coiflets 小波, 供小波消噪时选择使用。

表 3.1 常用小波基的主要性质

Table 3.1 The main property of some typic wavelet function

小波基函数	紧支撑性	正交性	对称性	离散小波变换	消失矩
Haar 小波	有	有	对称	可以	1
Daubechies 小波	有	有	近似	可以	N
Coiflets 小波	有	有	近似	可以	2N
Morlet 小波	无	无	对称	不可以	—

3.2.3 小波分解消噪原理

在实际的工程中, 有用信号通常表现为低频信号, 或是一些比较平稳的信号, 而噪声信号则通常表现为高频信号。一个含噪的一维信号模型可以表示成如下形式

$$x(n)=s(n)+u(n) \quad (3.3)$$

$x(n)$ —含噪信号; $s(n)$ —有用信号; $u(n)$ —噪声信号

在这里以一个简单的噪声模型详细加以说明, 即认为 $u(n)$ 为零均值且服从高斯分布的随机序列, 即服从 $N(0, \sigma_u^2)$ 分布。对 (3.3) 式两边作小波变换, 有

$$WT_x(a,b)=WT_s(a,b)+WT_w(a,b) \quad (3.4)$$

即两个信号和小波变化等于各个信号小波变换的和。

再令 $u(n)$ 是零均值、独立分布的平稳随机信号, 记 $u=(u(0)u(1)\dots u(N-1))^T$, 显然 $E\{uu^T\}=\sigma_u^2 I=Q$, 式中, $E\{\cdot\}$ 代表求均值运算, Q 是 u 的协方差矩阵。

令 W 是小波变换矩阵, 对正交小波变换, 它是正交阵。分别令 x 和 s 是对应 $x(n)$ 和 $s(n)$ 的向量, 向量 X, S 和 U 分别是 $x(n), s(n)$ 和 $u(n)$ 的小波变换, 即 $X=Wx, S=Ws, U=Wu$ 。由式 (3.5), 有 $X=S+U$ 。令 P 是 U 的协方差矩阵, 由于

$$E\{U\}=E\{Wu\}=WE\{u\}=0$$

$$\text{所以 } P=E\{UU^T\}=E\{Wuu^T W^T\}=WQW^T \quad (3.5)$$

因为矩阵 W 是正交阵, 且 $Q=\sigma_u^2 I$, 所以 $P=\sigma_u^2 I$

由此, 得到重要结论: 平稳白噪声的正交小波变换仍然是平稳的白噪声。由该结论可知, 对加性噪声模型, 经正交小波变换后, 最大程度的去除了 $s(n)$ 的相关性。其能量将集中在少数的小波系数上而这些系数既是在各个尺度下的模极大值。而噪声 $u(n)$ 经正交小波变换后仍然是白噪声, 因此, 其小波系数仍然是互不相关的, 它们将分布在各个尺度下的所有时间轴上。这一结论就为抑制噪声提供了理论依据, 即在小波变换的各个尺度下保留那些模极大值点, 而将其它点置零, 或是最大程度的减小, 然后利用处理后的小波系数作小波重构, 即可达到制噪声的目的^[43]。对信号消噪实质上是抑制信号中的无用部分, 恢复信号中用部分的过程。

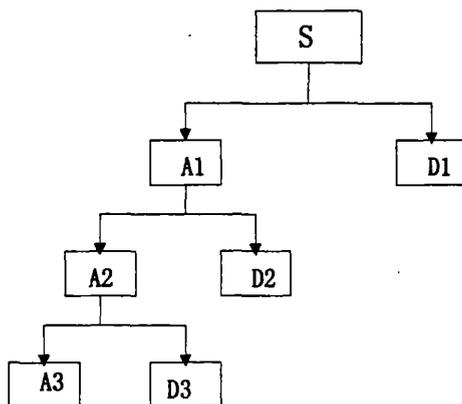


Fig3.1 Structure of three layer wavelet

图 3.1 三层小波分解结构图

一般地, 一维信号小波消噪的具体过程分为如下三个步骤:

(1) 一维信号的小波分解。选择一个小波并确定分解的层次 N , 然后对信号进行分解计算;

(2) 小波分解高频系数的闭值量化。对各个分解尺度下的高频系数选择一

个阈值进行软阈值化处理;

(3)一维小波重构。根据小波分解的最低层低频系数和各层高频系数进行一维小波重构。

这三个步骤中,关键在于是如何选择阈值及如何进行阈值量化,在某种程度上,它关系到信号消噪的质量。

以三层小波分解为例,其分解结构简图如图 3.1 所示。

3.3 小波包分解提取信号特征量

3.3.1 小波包分解基本原理

如图 3.1 所示,小波分解每次只对信号的低频部分进行分解,高频部分保留不动,因此高频部分频率分辨率差。固然这样对去除高频噪声是有效的,但是,声发射信号本身频率较高,提取信号的高频特征比较困难。小波包(wavelet packet)分解对此进行了改进,它同时可在低频和高频进行分解,自适应地确定信号在不同频段的分辨率^[73]。

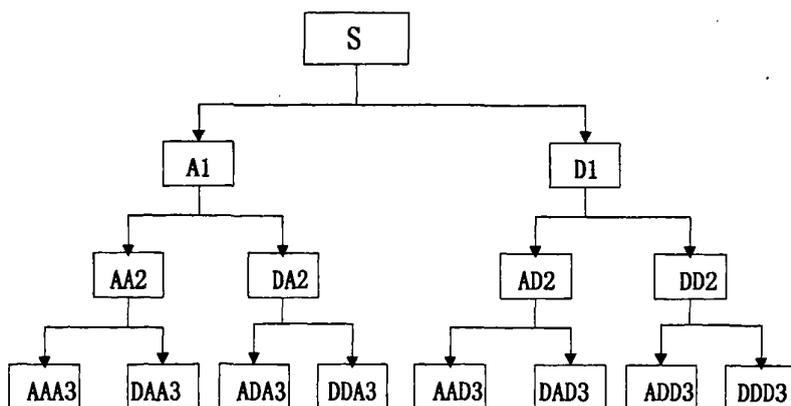


Fig3.2 Structure of three layer wavelet packet

图 3.2 三层小波包分解结构图

在实际应用中,往往需要提高高频带信号的频率分辨率,小波包分解给出了解决问题的途径。小波包分解能够为信号特征量提取提供一种更加精细的分析方法。它在全频带对信号进行多层的频带划分,集成了小波分解所具有的良好

好的时频局部化优点，继续对小波分解没有再分解的高频频带作进一步的分解，小波包技术实现了将信号无冗余、无疏漏、正交地分解到独立的频带内，从而提高频率分辨率，图 3.2 为三层小波包分解结构图。A 表示低频，D 表示高频，末尾的序号数表示小波包分解的层数（也即尺度数）。

分解具有关系式：

$$S=AAA3+DAA3+ADA3+DDA3+AAD3+DAD3+ADD3+DDD3$$

3.3.2 小波包分解与重构算法

小波包的定义、性质本文不做详细论述，下面给出小波包的分解和重构算法^[74]。设 $g_j^n(t) \in U_j^n$ ，则 $g_j^n(t)$ 可表示为

$$g_j^n(t) \in \sum_l d_l^{j,n} u_n(2^j t - l)$$

小波包分解算法由 $\{d_l^{j+1,n}\}$ 求 $\{d_l^{j,2n}\}$ 与 $\{d_l^{j,2n+1}\}$

$$\{d_l^{j,2n}\} = \sum_k a_{k-2l} d_k^{j+1,n}$$

$$\{d_l^{j,2n+1}\} = \sum_k b_{k-2l} d_k^{j+1,n}$$

小波包重构算法 由 $\{d_l^{j,2n}\}$ 求 $\{d_l^{j,2n+1}\}$ 与 $\{d_l^{j+1,n}\}$

$$d_l^{j+1,n} = \sum_k [h_{1-2k} d_k^{j,2n} + g_{1-2k} d_k^{j,2n+1}]$$

3.3.3 声发射信号能量特征提取

小波分析能把信号分解成不同频率范围的分量，由于不同类型的声发射信号包含的信息成分不同，经过小波包分解后，声发射信号中的信息成分在各个分解尺度的不同频带分量中的分布也不同。从统计学的角度考虑，信号在不同频带上信息分布情况可以通过各频带上信号的能量反映出来。因此，可提取小波包分解后各频带上的能量信息作为声发射信号刀具磨损状态的特征^[75]。

信号 $x(n)$ 经过 j 尺度的小波包分解可分解成 $J=2^j$ 个频率范围分量，即有

$$x(n) = A_0x(n) + D_1x(n) + D_2x(n) + \dots + D_{j-1}x(n) \quad (3.6)$$

各频段分量的能量:

$$E_0^A x(n) = \sum_{n=1}^N (A_0x(n))^2$$
$$E_j^D x(n) = \sum_{n=1}^N (D_jx(n))^2 \quad j=1,2,\dots,J-1 \quad (3.7)$$

$E_0^A x(n)$ 表示信号在分解尺度 j 上的低频分量, $E_j^D x(n)$ 表示信号在分解尺度 j 上某高频段的能量。

信号的总能量为:

$$E_x(n) = E_0^A x(n) + \sum_{j=1}^J E_j^D x(n) \quad (3.8)$$

4 数据处理与显示平台硬件设计

4.1 平台总体结构

作为课题的前期研究,本文选用 MagicARM2410 为刀具状态监测系统数据处理与显示平台硬件开发平台,简化硬件设计。系统以 ARM9 为核心,平台的构成框图如图 4.1 所示,该平台是通过 A/D 获得各传感器数据,对检测过程进行间接控制,对检测数据做相关处理,实时显示波形及计算结果,并对结果进行实时存盘,必要时传到监控中心做进一步的处理和分析。

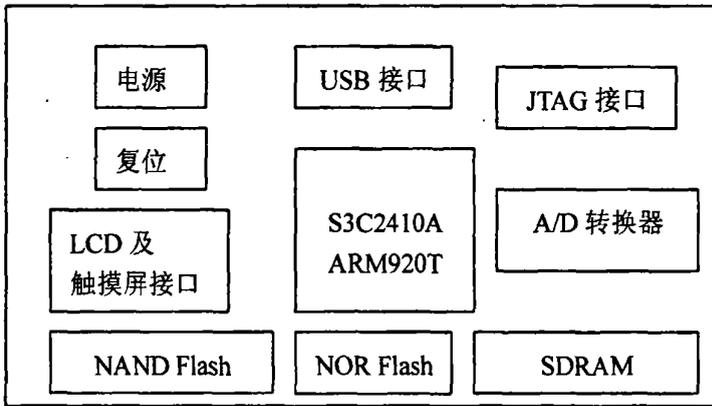


Fig.4.1 Frame of the data processing and display platform

图 4.1 数据处理与显示平台框图

4.2 微处理器选型

在 32 位微处理器里面,ARM 系列的处理器应用最为典型和广泛,它具有低功耗、性能稳定和应用前景广等特点,在实时监控等领域均得到重视和研究。由于 ARM 微处理器有多达十几种的内核结构,几十个芯片生产厂家,以及千变万化的内部功能配置组合,在实际应用中,需合理选择芯片,以适应不同的应用领域。本监测系统希望嵌入操作系统以减少应用软件的开发时间,增强系统的灵活性,所以选择了 ARM9 系列的处理器,因为支持哈佛体系结构,采用五级流水线,执行指令效率非常高,并且有全功能的 MMU,所以它不但支持 $\mu\text{c/os-II}$ 、Linux 等操作系统,而且支持嵌入式 Windows CE 操作系统。在厂

家的选择方面,本系统采用了三星公司的 ARM920T 系列的 S3C2410 芯片,它支持基本的外设接口,如:彩色 TFT LCD、USB、UART 等,并支持 MMC 和 S 等标准的外部插卡。它具有的最大优点是能支持 NAND FLASH 启动,从而能够使整个系统的综合成本降低,具有较高的性价比。S3C2410 是三星公司的一款 32 位 RISC 嵌入式处理器。为了尽可能减少系统的整体成本,S3C2410 包括了以下组件:16KB 的指令和数据 CACHE,处理虚拟存储器管理的 MMU,支持 STN 和 TFT 的 LCD 控制器,NAND FLASH 的系统引导器,SDRAM 控制器,触摸屏接口,USB 接口,PLL 时钟产生器等。同时 S3C2410 的 ARM920T 内核采用 0.18 微米标准 CMOS 技术制造的,它采用了一种全新的总线设计,即 AMBA(ADVANCED MICROCONTROLLER BUS ARCHITECTURE)。ARM920T 内核实现了 MMU\AMBA 总线和分离的指令和数据 CACHE 数据结构。集成在片上的功能包括:

- 内核 1.8V 供电,3.3V 外部 I/O 接口供电;
- 16KB 的指令 CACHE 和 16KB 的数据 CACHE·MMU;
- 外部存储器控制器(SDRAM 控制器和外部片选信号产生逻辑);
- LCD 控制器并有一个通道的 DMA,能支持高达 4K 的 STN 和 256K 的 TFT;
- 4 通道带有外部请求引脚的 DMA;
- 3 通道的 DART 和 2 通道的 SPI;
- 1 通道的多主 IIC-BUS/1 总线和 1 个通道的 IIS-BUS 总线控制器;
- SD 存储器控制接口;
- 2 个 USB 主口和 1 个通道的 USB 从口;
- 4 通道的 FWM 定时器和 1 个通道的内部定时器;
- 带有看门狗定时器;117 个通用 I/O 口,包括 24 个外部中断源;
- 4 种电源模式:一般模式、低速、休眠、掉电;
- 内部 ADC. 8 通道 10 位;
- 触摸屏接口;
- 片上 PLL 时钟发生器。

由以上的芯片功能配置,可以看出它提供的完整的系统外围使得 S3C2410 能够最大的减少系统的整体成本和减少另外的设备组件。选用其作为监测系统

的 CPU,在保证监测性能的基础上,还可以增强系统的灵活性,从而使系统具有较强的可扩展和可移植性。在其上嵌入操作系统,还可以有效地缩短软件开发周期,降低开发成本。

4.3 平台的功能及接口电路设计

(1) S3C2410 的功能设计: 作为 S3C2410 芯片的 CPU 内核, ARM920T 核由 ARMTDMI、存储管理单元(MMU)和高速缓存三部分组成。其中, MMU 可以管理虚拟内存, 高速缓存由独立的 16KB 地址和 16KB 数据高速 Cache 组成, 从而具有更高的指令和数据处理能力。图 4.2 为 ARM920T 的内核结构。S3C2410 处理器有多个通用 I/O 端口及控制端口, 并且其内部集成了 LCD 控制器, 在围绕其进行硬件设计时, 外围部件可以较方便地挂接在系统总线上; 同时它还支持嵌入式操作系统运行, 并支持小波分析等算法。在嵌入式操作系统之上, 可以运行声信号的识别程序, 以及其它的应用程序, 它能够实现声信号的预处理、小波滤波、特征提取、训练和识别等任务。

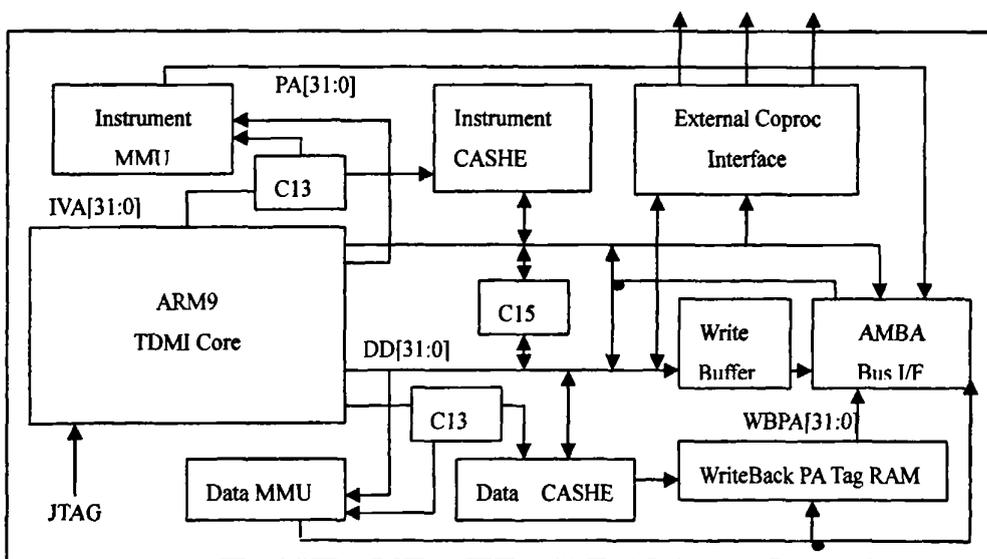


Fig4.2 ARM920T Core structure diagram

图 4.2 ARM920T 的内核结构

(2) 电源与复位电路: 监测系统的供电电源仅提供 5V 和 3.3V 电压, 因此

需要进行电源转换才能满足系统需要，电源转换电路如图所示(+12V 转换电路如图 4.3 所示、3.3V 转换电路如图 4.4 所示)。监测系统的供电电源电压 5V 经 SPX1117—33 芯片转换后可得 3.3V 的工作电压供主板和核心板工作使用，另外电源还涉及电容滤波电路。复位电路主要完成系统的上电复位和系统在运行时用户的按键复位，主要由芯片 MAX811 构成，实现对电源电压的监控和手动复位。系统的复位电路如图 4.5 所示。

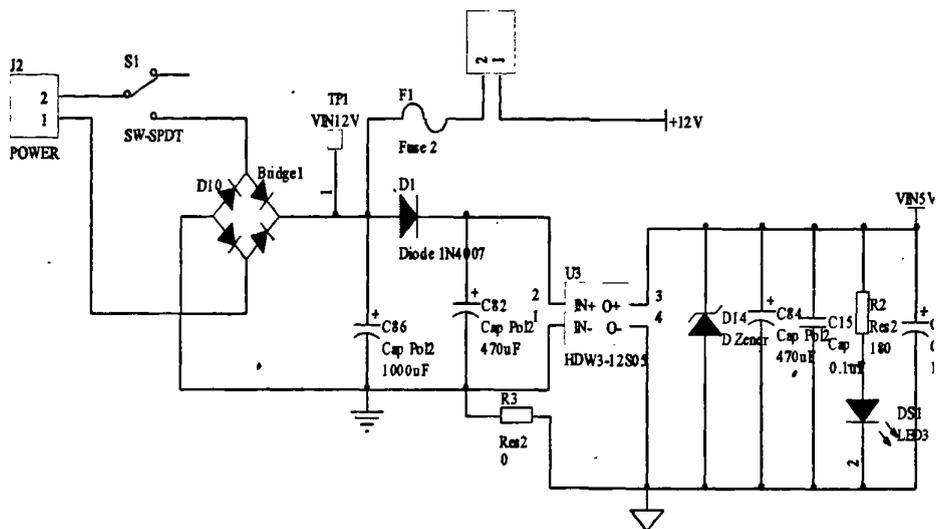


Fig4.3 +12V Conversion circuit

图 4.3 +12V 转换电路

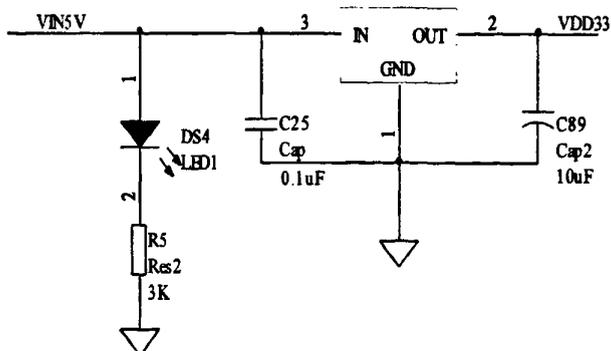


Fig4.4 3.3V Conversion circuit

图4.4 3.3V转换电路

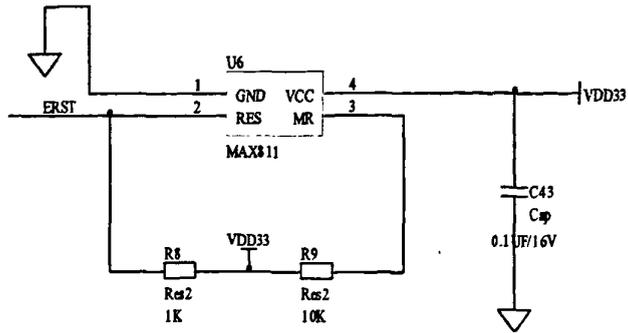


Fig4.5 System reset circuit

图4.5 系统复位电路

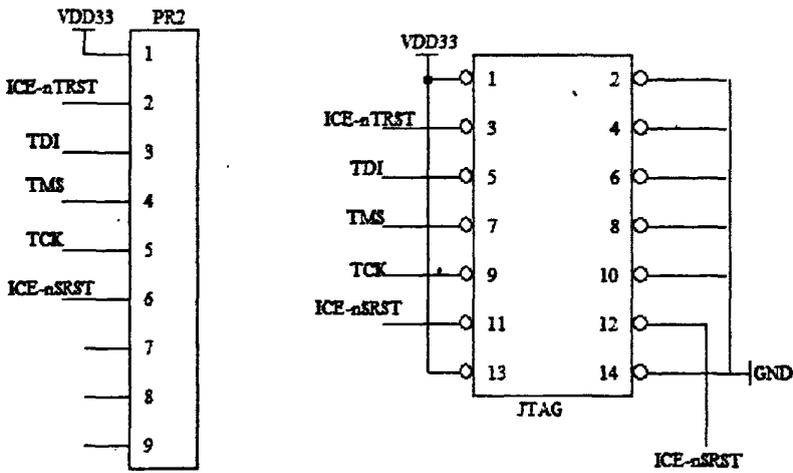


Fig4.6 JTAG Interface circuit

图4.6 JTAG接口电路

表4.1 14针JTAG接口的定义

Table4.1 14Pin JTAG interface

引脚	名称	描述
1, 13	VCC	接电源
2, 4, 6, 8, 10, 14	GND	接地
3	nTRST	测试系统复位信号
5	TDI	测试数据串行输入
7	TMS	测试模式选择
9	TCK	测试时钟
11	TDO	测试数据串行输出
12	NC	未连接

(3) 系统存储电路：系统采用NOR Flash 和 NAND Flash两种Flash存储器以及SDRAM存储器配合使用。其选型及接口电路设计在此不再详述。

(4) JTAG接口电路：JTAG(Joint Test Action Group, 联合测试行动小组)是一种国际标准测试协议，主要用于芯片内部测试及对系统进行仿真、调试。JTAG技术是一种嵌入式调试技术，它在芯片内部封装了专门的测试电路访问口，通过专用的JTAG测试工具对内部节点进行测试。目前大多数比较复杂的器件都支持JTAG协议，如 ARM、DSP、FPGA等器件。JTAG接口电路如图4.6所示。通过JTAG接口可以对芯片内部所有部件进行访问，因而它是开发调试嵌入式系统的一种简洁高效的手段。本监测系统采用了14针的JTAG接口，其引脚功能定义如表4.1所示

(5) A/D转换电路：S3C2410具有8路10位ADC转换器，其参考电压由芯片的Vref引脚输入，MagicARM2410实验箱上采用3.3V参考电压，电路图如图4.7所示。实验箱主板提供了两路直流电压测量电路，可调电阻W1、W2用于调整ADC的输入电压。

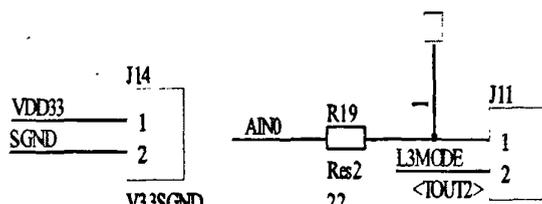


Fig4.7 A/D conversion circuit

图4.7 A/D转换电路

(6) LCD及触摸屏设计：由于S3C2410内部带有LCD控制器，所以LCD可以作为外设直接挂在系统总线上。从CPU的LCD控制器出来的信号线包括8跟数据线和5跟控制线，这些信号线经过两片74HC245芯片隔离后连接到LCD模块。LCD模块接口电路如图4.8所示。触摸屏接口电路如图4.9所示。

(7) USB接口电路设计：S3C2410具有2路USB v1.1接口，一路固定为HOST接口(下行口)，另一路可配置为HOST或Device接口。这两个下行口对于芯片的引脚为DN0、DP0和DN1、DP1，其中DN1、DP1引脚与USB设备端(即Device)得引脚复用。USB主机接口如图4.10所示，USB设备接口如图4.11所示。

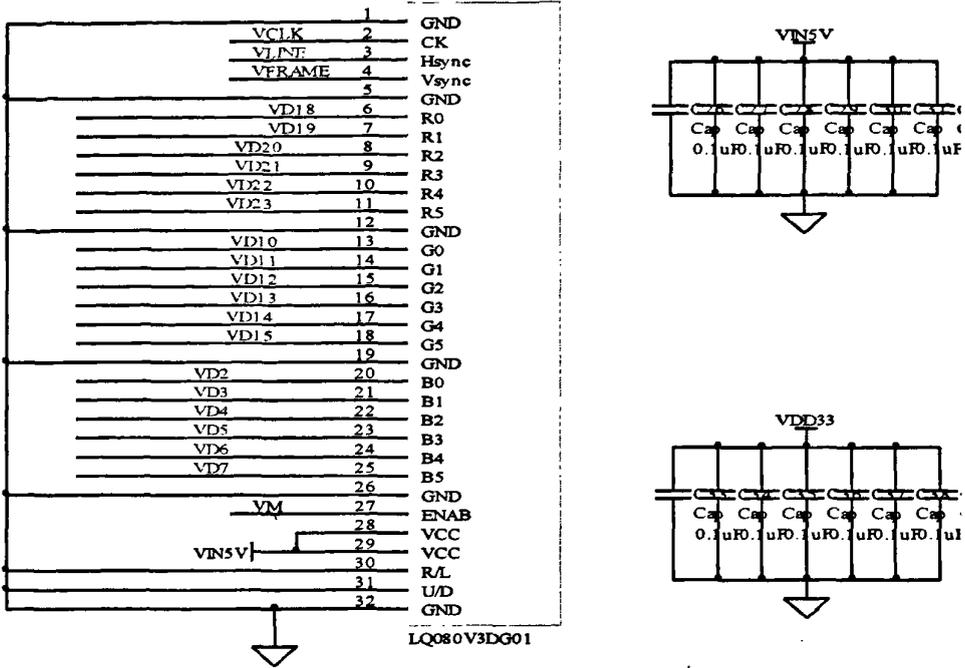


Fig4.8 LCD Module interface circuit

图4.8 LCD模块接口电路

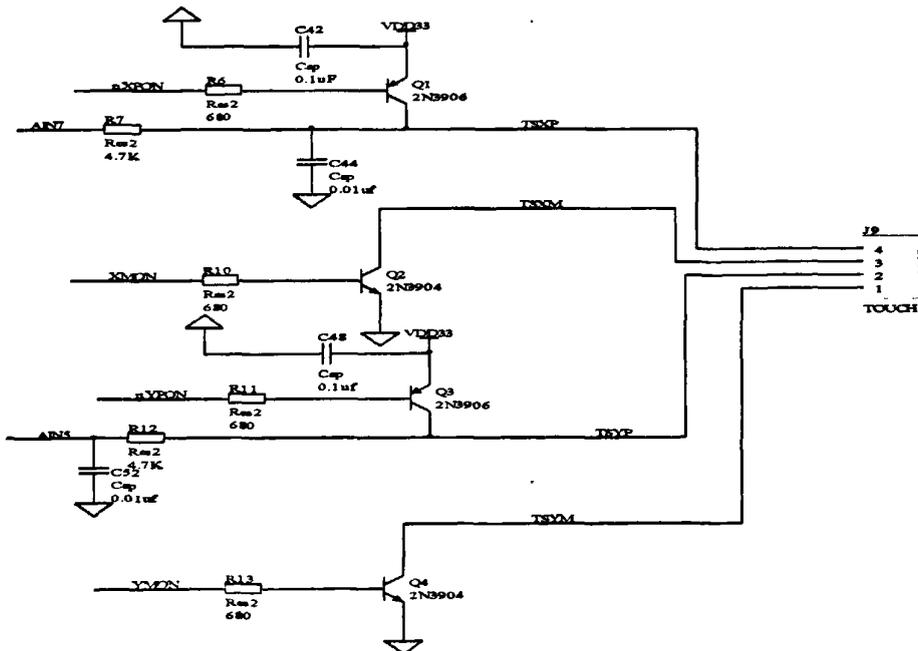


Fig4.9 Touch screen interface circuit

图4.9 触摸屏接口电路设计

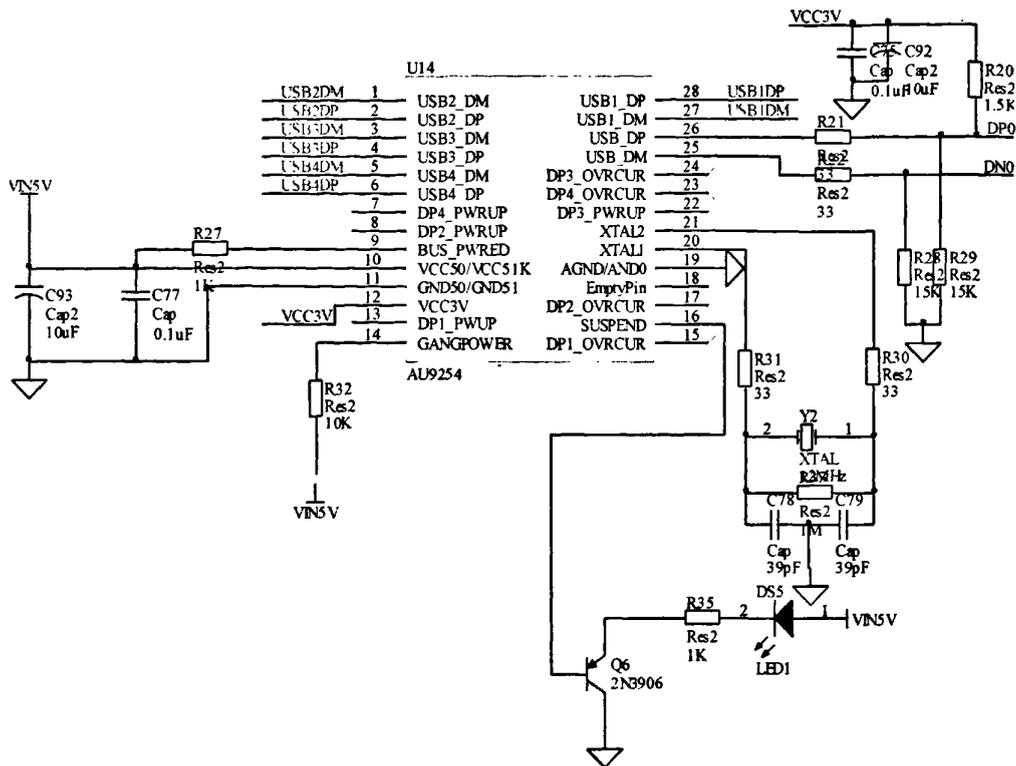


Fig4.10 USB host interface
图4.10 USB主机接口

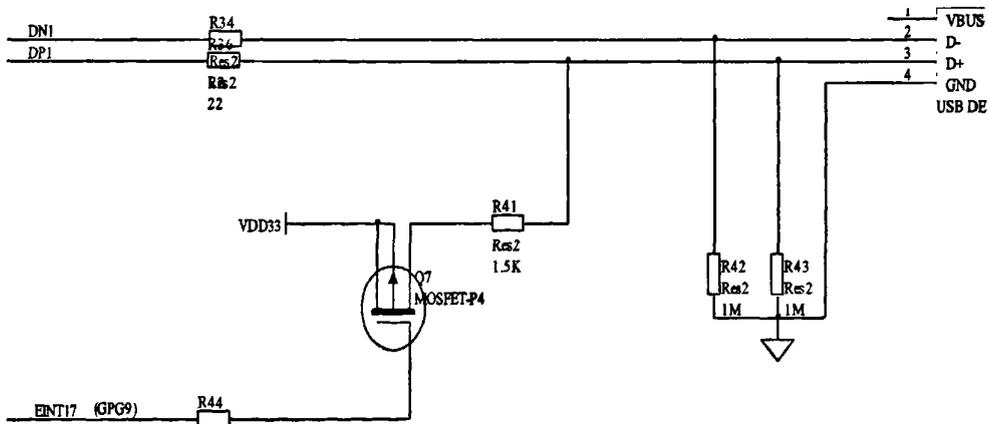


Fig4.11 USB device interface
图4.11 USB设备接口

5 数据处理与显示软件平台设计

在刀具状态监测系统中，为了提高数据的处理速度及精度，并且给用户一个良好的人机界面，选用 MagicARM2410 实验箱作为数据处理和显示操作界面的硬件平台。选择嵌入式 Windows CE 操作系统作为操作系统平台。并在此基础上利用 eMbedded Visual C++ (EVC) 开发图形用户界面、数据处理算法等等。

5.1 Windows CE 开发平台的构建

5.1.1 Windows CE 特点

嵌入式操作系统是一种支持嵌入式系统应用的操作系统软件，它是嵌入式系统极为重要的组成部分，通常包括与硬件相关的底层驱动软件、系统内核、设备驱动接口、通信协议、图形界面及标准化浏览器等。嵌入式操作系统具有通用操作系统的基本特点，如能够有效管理越来越复杂的系统资源；能够把硬件虚拟化，使得开发人员从繁忙的驱动程序移植和维护中解脱出来；能够提供库函数、驱动程序、工具集以及应用程序。与通用操作系统相比，嵌入式操作系统在系统实时高效性、硬件的相关依赖性、软件固态化以及应用的专用性等方面具有较为突出的特点^[33,34]。

常用嵌入式操作系统有 VxWorks、Embedded Linux、Windows Embedded 等。VxWorks 是风河公司(WindRiver)开发的一款商用实时操作系统。VxWorks 因其良好的可靠性和卓越的实时性，已广泛应用在通信、军事、航空、航天等高端技术及实时要求极高的领域中；嵌入式 Linux(Embedded Linux)的出现，给整个嵌入式世界带来了无限生机。Linux 具有内核可裁剪、效率高、稳定性好、移植性好、源代码开放等优点，还内含了完整的 TCP/IP 网络协议栈，很适合在嵌入式领域应用。与传统的实时操作系统 (RTOS) 相比，采用嵌入式 Linux 这样开放源码的操作系统的另外一个好处是，Linux 开发团体会比 RTOS 的供应商更快地支持新的 IP 协议和其他协议。例如，用于 Linux 的设备驱动程序要比用于商业操作系统的设备驱动程序多。因此，越来越多的公司和个人正在把 Linux 作为嵌入式操作系统的首选。

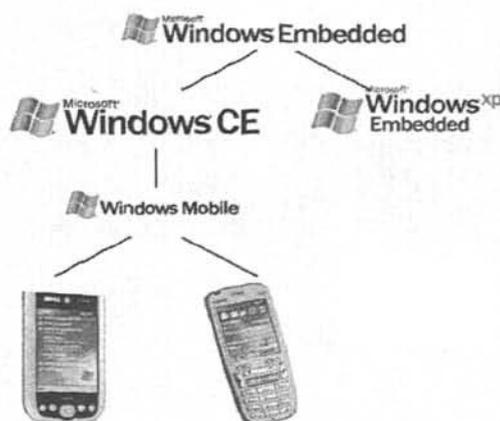


Fig.5.1 Windows embedded family

图 5.1 Windows embedded 家族

Windows CE 是微软公司推出的一个功能强大的嵌入式实时操作系统，它具有很多优秀的性能。该系统的模块化设计，使开发人员可以根据需求定制设备，这种定制有助于快速开发功能各异的产品与服务。Windows CE 提供一系列的集成定制和开发工具，使得开发嵌入式软件的门槛大大降低。Windows CE 已经越来越多的被开发厂商支持与使用，迅速成为嵌入式操作系统主流平台之一。基于 Windows CE 的 Windows Mobile 使得全球范围内智能手机平台市场得以快速的发展，Windows CE 还在 PDA、机顶盒、嵌入式通信设备（诸如网关路由器等）、客户端电子设备、专用工业控制器（如机器人、生产线控制等）、医疗设备、测量仪器、信息家电、与 GPS 和 GIS 结合的导航设备等中得到广泛的应用。

Windows CE 具有以下特点：

(1) 精简的模块化操作系统

Windows CE 是高度模块化的嵌入式操作系统，正因如此，用户可以为了满足特定的要求而对操作系统进行定制。在用户定制的操作系统中，不需要的模块可以被拿走，只有所需的模块才会被包含进来。

Windows CE 的可裁减性使其体积也非常小。一个最小的可运行的 Windows CE 内核只占 200KB 左右；增加网络支持需要 800KB；增加图形界面支持需要大概 4MB；增加 Internet Explorer 支持需要额外的 3MB。这样就可以充分适应

一些硬件资源不足的嵌入式设备的要求。对于 Windows CE 的可定制性有这样形象的比喻：如果要盖一座房子，那么桌面 Windows 相当于买一座大楼里的商品房，房间和装修一应俱全，而且这座大楼里的所有房间样式都差不多。而 Windows CE 相当于买了一堆水泥、砖块、沙子等材料，可以用这些材料自己盖一座大楼，也可以用它们盖一个小狗窝。

值得注意的是，体积小并不意味着 Windows CE 的功能单一。如果把 Windows CE 所有的可选功能都加进来，完全可以和 Windows 9X 系列相媲美。

(2) 多硬件平台支持

嵌入式系统的专用性特点决定了嵌入式系统的硬件设备必定是多种多样的。为了适应嵌入式系统的要求，Windows CE 支持在多种不同的 CPU 硬件平台上运行，包括 x86, ARM, MIPS 等嵌入式领域主流的 CPU 结构。

(3) 支持有线的和无线的网络连接

Windows CE 不但支持传统的有线网络连接，还支持各种无线网络标准，包括蓝牙、红外及 802.11 等。可基于 Windows CE 构建有扩展性的无线平台，将移动设备彼此连接或连接到现有的设备上；也可通过网络进行远程登录、验证和管理，或为设备上的应用程序和服务提供更新。

(4) 稳健的实时性支持

实时性是指能够在限定的时间内执行完规定的功能，并对外部的异步事件做出反应的能力。实时性的强弱以完成规定功能和做出响应时间的长短来衡量。提高硬件的处理能力可以在一定程度上提高计算机控制系统的实时性，但是当硬件确定后，控制系统的实时性主要由操作系统来决定。

Windows CE 是一个实时操作系统。实时支持功能在以下几个方面提升了 Windows CE 的性能：①支持嵌套中断②允许更高级别的中断首先得到响应，而不是等待低级别的中断服务线程（IST）完成③更好的线程响应能力④对高级别的 IST 的响应时间的上限要求更加严格，在线程响应能力方面的改进，可以帮助开发人员掌握线程转换的具体时间，并通过增强的监控能力和对硬件的控制能力帮助开发人员创建更好的嵌入式应用程序⑤更多的优先级别，256 个优先级别可以使开发人员在控制嵌入式系统的时序安排方面有更大的灵活性⑥更强的控制能力，对系统内的线程数量的控制能力可使开发人员更好的掌握调度程序的工作情况。

(5) 丰富的多媒体和多语言支持

丰富的多媒体支持是 Windows CE 的一大特性，基于 DirectX API 和 Windows Media 的技术可以提供高性能的视频、音频、流式多媒体和 3D 图形处理服务。这些功能可满足大部分多媒体娱乐和游戏的要求。

同时，Windows CE 是基于 Unicode 的，可支持国际语言，这样就可以针对特定的市场调整产品。它可以为那些想创建本地化操作系统版本的 OEM 提供本地化支持。

(6) 强大的开发工具

与其他嵌入式操作系统相比，Windows CE 为开发人员提供了友好的开发工具支持。这些工具可帮助开发人员简化开发流程并提高开发效率。

对于 Windows CE 的应用程序开发人员，可选的开发工具有 eMbedded Visual C++和 Visual Studio.NET；对于操作系统定制设计人员，可使用 Platform Builder。Platform Builder 是一个集成操作系统的“构建—调试—发布”三者为一体的集成开发环境。

此外，Windows CE 还提供了多种模拟器，它们可以模拟硬件设备，使开发人员无需拥有真实的硬件，即可进行部分 Windows CE 下的开发。

5.1.2 Windows CE 5.0 介绍

微软公司于 1996 年 11 月发布了 Windows CE 的第一个版本。它的设计目的是使熟悉桌面 Windows 的用户易于上手和使用。Windows CE 5.0 是微软于 2004 年 7 月推出的 Windows CE 的最新版本。与以前版本相比，Windows CE 5.0 的变化主要集中在以下几个部分：

(1) 操作系统增强

在硬件驱动方面，Windows CE 5.0 新增了对 USB 2.0 的支持，包括 USB 2.0 Host 和 USB 2.0 Client。

在图形方面，Windows CE 新增了对 Direct 3D Mobile 的支持，Direct 3D Mobile 可用来开发嵌入式设备上的 3D 图形应用程序。此外，Windows CE 还增加了对图片格式的支持，操作系统可处理 GIF 和 JPEG 等常见的图片格式。

在内核层面，Windows CE 5.0 支持的系统中断从 32 个增加到 64 个。此外，

还增加了可变的时钟调度。这允许 OEM 按需产生时钟中断，而不是现在的每毫秒都产生一个中断。

(2) 统一的构建系统

Windows CE 5.0 中把使用 Platform Builder 构建操作系统与使用命令行构建操作系统进行了统一。在以前的 Platform Builder 中，使用命令行构建与使用 Platform Builder 构建采用的是两种不同的机制。这给开发人员造成的一定的困难。在 Windows CE 5.0 中，Platform Builder 集成开发环境只是命令行界面的简单封装，使用 Platform Builder 与使用命令行构建操作系统没有任何功能上的区别。

(3) 高质量的 BSP

在 Windows CE 5.0 中，微软对板级支持包(BSP)的结构做了非常大的改变。将 BSP 的功能提炼为一些小的库文件，并且对 BSP 的目录及文件等都做了限定。这样不仅简化了 BSP 的开发，而且相比以前的 BSP 更加模块化，结构更加清晰。

此外，在 Windows CE 5.0 中，微软共享了 250 万行 Windows CE 操作系统的源代码，占整个 Windows CE 代码的 70%左右。开发人员可以在微软的 Shared Source License 协议的许可下使用这部分源代码。

5.1.3 基于 Windows CE 的嵌入式系统开发流程

与通常 PC 机上应用程序开发不同，通常如果要开发一个嵌入式系统，软件开发和硬件开发都是要考虑的内容。基于 Windows CE 的嵌入式系统开发同样会涉及到硬件和软件两方面的开发。图 5.2 描述了一般基于 Windows CE 的嵌入式系统开发流程。

(1) 硬件设计

首先，要确定系统所运行的硬件平台。这涉及根据具体的应用选择合适的硬件。有一点需牢记，嵌入式系统的硬件设计与通用 PC 的硬件设计不同。由于嵌入式系统通常是专用的系统，嵌入式系统的设计强调的是“够用”而不是“功能强大”。也就是说，在可实现应用功能的前提下，尽量去掉用不到的接口和外设，以降低成本。

(2) 让 Windows CE 运行起来

得到硬件后，下一步是让 Windows CE 运行在硬件平台上。这必须拥有针对这块硬件的板级支持包(BSP)。BSP 是操作系统和硬件板卡的重要交互接口。如果硬件是从 OEM(原始设备提供商)处采购，那么通常都会提供 Windows CE 的 BSP、默认的运行时映像和 SDK。否则就要自己开发 BSP。

(3) 定制操作系统

下一步的工作是决定是否进行操作系统定制，这也完全取决于应用的需求，如果从 OEM 处获得的默认的运行时映像不能满足应用的需求，那么就需要自己定制操作系统。

操作系统定制是通过 Platform Builder 工具来完成的。使用 Platform Builder 可根据具体的需要选择操作系统功能组件，然后生成操作系统的运行时映像。

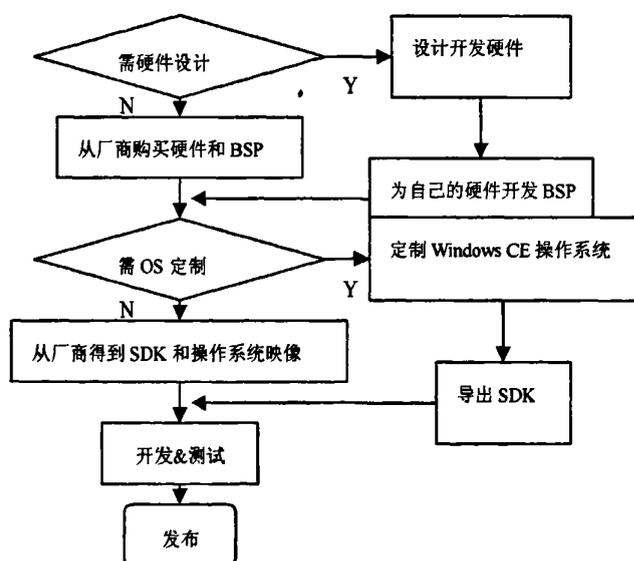


Fig5.2 Windows CE development flow

图5.2 Windows CE 开发流程

(4) 应用程序开发

当硬件与操作系统都已经具备后，所剩的工作就是为自己的平台开发一些必要的应用程序。这一步骤跟通常的 Windows 下的应用程序开发没有太大的区别。惟一不同的是，在 Windows CE 下编写的应用程序既可以象桌面 Windows

那样通过安装包的形式进行安装,也可以把应用程序作为操作系统的—个组件,打包进入操作系统的运行时映像中。

5.1.4 Windows CE 的层次体系结构

Windows CE 属于比较典型的微内核操作系统。在内核中仅仅实现进程、线程、调度及内存管理等最基本的模块,而把图形系统、文件系统及设备驱动程序等等都作为单独的用户进程来实现。这样做显著增加了系统的稳定性和灵活性。

什么是微内核?微内核是一个小型的操作系统核心,它为模块化扩展提供基础。微内核的基本原理:只有最基本的操作系统功能才能放在内核中。不是最基本的服务和应用程序在微内核之上构造,并在用户模式下执行。图 5.3 表明了单体内核操作系统和微内核结构操作系统结构的区别。

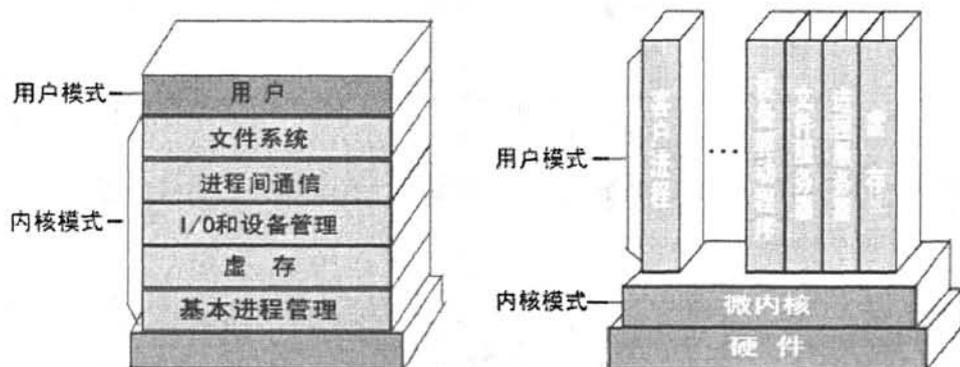


Fig.5.3 Monolithic kernel OS (UNIX) and micro kernel OS (Windows CE)

图 5.3 单体内核操作系统 (UNIX) 和微内核结构操作系统 (Windows CE)

从图中可以看到微内核结构用一个水平分层的结构代替了传统的纵向分层的结构。在微内核外部的操作系统部件被当作服务器进程实现,它们可以借助通过微内核传递信息来实现相互之间的交互。图 5.4 表明了 Windows CE 的系统架构。

由于 Windows CE 是个微内核操作系统,其基本功能都被放在多个独立的进程 (EXE) 里面实现。在运行的时候,这些进程大致有如下几个:内核

(NK.EXE)、图形系统 (GWES.EXE)、对象存储 (FILESYS.EXE)、设备管理。

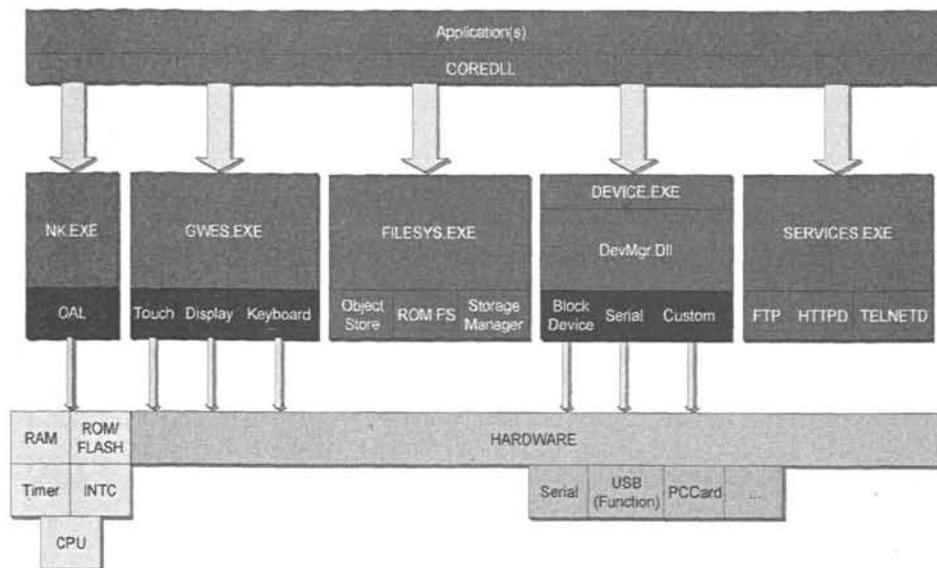


Fig.5.4 Windows CE system structure

图 5.4 Windows CE 系统架构

系统 (DEVICE.EXE)、服务 (SERVICES.EXE)。由于 Windows CE 是一个可裁剪的嵌入式操作系统，所以未必所有的 Windows CE 系统都具有上述几个进程，只有 NK.EXE 和 GWES.EXE 是所有系统中必须的。

NK.EXE 由 NK.LIB 和 OAL.LIB 两部分组成。微软提供了预先编译的 NK.LIB 库文件，它的代码与 CPU 指令体系结构相关而与具体的外设无关。OAL.LIB 是 OEM 层中的 OAL 代码编译后的输出。NK.EXE 具体功能有：处理器调度内存管理、进程创建加载、系统内的通信机制、中断处理、为其他部分提供核心调用例程。

图形窗口事件系统(GWES.EXE)负责操作系统中与图形界面相关的部分，管理所有的图形界面处理以及用户输入，显示驱动也是由它来管理的。它是 Windows CE 系统中最为高度组件化的模块之一。GWES.EXE 实现的功能基本上是桌面 Windows 中的 USER32 和 GDI32 的集合。

FILESYS.EXE 是在运行时负责对象存储的进程。对象存储包括文件系统、

Windows CE 数据库和系统注册表三部分。文件系统包括 RAM 文件系统和 ROM 文件系统，它们可以为应用程序提供永久存储服务；Windows CE 数据库是对流式文件的更高级抽象，它提供了结构化的数据存储，数据库中的数据以记录的形式存在，进一步方便了应用程序管理数据；系统注册表负责存储系统和应用程序的配置信息。

设备管理系统 (DEVICE.EXE) 负责加载、卸载和管理所有不被 GWES.EXE 管理的驱动程序。大多数功能都放在了 devmgr.dll 中，由它提供所有的与驱动相关的函数实现，在启动的时候通过注册表加载驱动程序。

SERVICES.EXE 与 DEVICE.EXE 非常类似，但是它们没有管理真实的硬件，它们提供一些后台的处理或者为应用程序提供高级的功能。它负责加载系统服务的进程 (FTP, TELNET, HTTP)，Windows CE 提供单独的 API 管理服务，用户也可以自己添加服务。

Windows CE 是一个基于优先级的抢占式多任务操作系统。在 Windows CE 中，每个运行着的程序都是一个进程。在一个进程中可以包含多个线程。Windows CE 调度系统负责对系统中的多个线程进行调度，调度是基于优先级的。此外 Windows CE 还提供了多种方法供多个线程进行同步，多个进程之间相互通信。

5.1.5 Windows CE 系统、驱动剪裁及移植

本系统实验阶段采用的是 Magic2410 开发板，因为购买了硬件，所以厂商提供了系统移植所需的 Bootloader、Eboot、NK.BIN 等文件。具体步骤如下：

(1) 分区

开发板里有两块存储器，NorFlash 和 NandFlash。前者安装有类似于电脑主板上的 BIOS 系统，分区只会擦除 NandFlash 里面的所有数据。连接好串口，打开超级终端，上电启动开发板，进入如下的 BIOS 功能菜单，选择功能号[x] 开始对 NandFlash 进行分区即可。

(2) 安装 Bootloader

打开 DNW 程序，接上 USB 电缆，如果 DNW 标题栏提示[USB: OK]，说明 USB 连接成功，这时根据菜单选择功能号[v]开始下载 supervivi；点击“USB

Port->Transmit”选项，并选择打开文件 supervivi 开始下载；下载完毕，BIOS 会自动烧写 supervivi 到 Nand Flash 分区中，并返回到主菜单。



Fig5.5 BIOS menu

图 5.5 BIOS 菜单

(3)安装 Eboot

在 BIOS 主菜单中选择功能号 [e]，开始下载 Eboot；点击“USB Port->Transmit”选项，并选择文件 Eboot_2440.nb0 开始下载。

(4)安装 Windows CE 内核映象

在 BIOS 主菜单中选择功能号 [w]，开始下载 WINCE 内核；点击“USB Port->Transmit”选项，并选择打开相应的内核文件 NK.BIN 开始下载；烧写完毕，Windows CE 会自动运行。

5.1.6 Embedded Visual C++开发工具

微软为应用程序的开发人员提供了三种开发工具：Embedded Visual C++、Platform Builder、Visual Studio 2005。前两个用来开发本机代码，后一个用来开发基于 .NET Compact Framework 的托管 C++代码。Windows CE 下应用开发模型如图 5.6 所示。

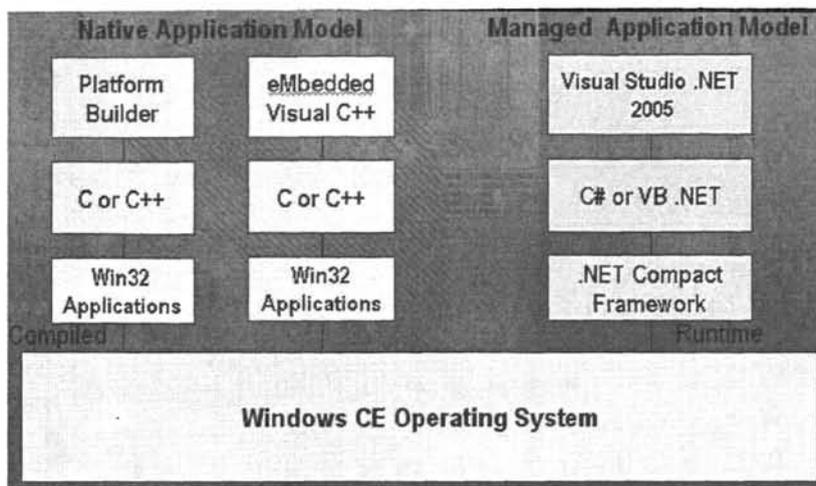


Fig 5.6 Application development model under Windows CE

图 5.6 Windows CE 下应用开发模型

(1) Visual Studio 2005

Visual Studio 2005 可使用 C#, VB.NET 开发基于 .NET Compact Framework 1.0 和 .NET Compact Framework 2.0 的托管代码,也可以使用 C/C++ 语言来开发本机代码,它具有完整的开发环境,只要在目标平台上安装 .NET compact framework 即可。Visual Studio 2005 优点在于易于调试和测试,它的集成开发环境 (IDE) 强化了对图形界面设计的支持。开发人员可以所见即所得的方式设计 Pocket PC 和 Smartphone 的应用程序,这样就避免了以前在 IDE 中无法知道目标设备的确切分辨率而导致的图形界面问题。在 Visual Studio 2005 中,所有的 Windows CE 模拟器都是基于 ARMV4 体系结构的。这对于开发 Pocket PC 和 Smartphone 的应用程序来说有非常重要的意义,因为现在市场上所有的 Pocket PC 和 Smartphone 设备的 CPU 都是清一色的 ARM 处理器。这样在模拟器下运行的代码无需重新编译即可在真实的 Pocket PC 和 Smartphone 上运行。

(2) Platform Builder^{[48][49]}

Platform Builder 主要作用是定制 Windows CE 操作系统,但是也可以使用它进行应用开发。它提供了创建和调试 Windows CE 映像 NK.BIN 的集成开发环境,如向导和工具栏,支持活动模板库(ATL)、微软基本类库(MFC)和 Visual Basic,为支持的处理器家族提供编译器、内核调试器以及各种远程调试工具。通过以太网或者串、并行口与目标平台连接,并将 Windows CE 操作系统映像

文件 NK.BIN 下载到目标平台。下载完成之后，Windows CE 在目标平台启动。在开发工作站上，可以使用 Platform Builder 提供的调试工具查看 Windows CE 的运行情况。在 Platform Builder 中选择 File->New Project or File 可打开新建项目对话框，如图 5.7 所示。WCE Application 和 WCE Console Application 分别对应 Windows CE 下的图形应用程序和命令行应用程序。

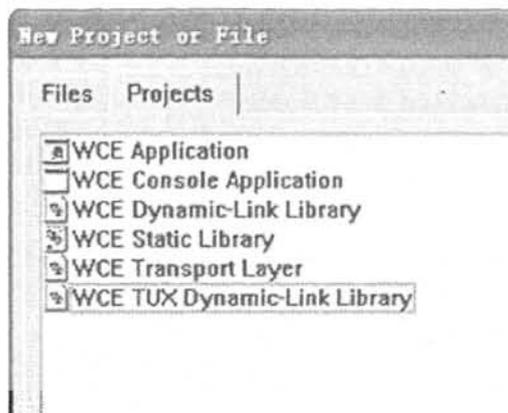


Fig5.7 Platform Builder's new project guide

图 5.7 Platform Builder 新建项目向导

Platform Builder 支持 Win CE App / DLL / Lib，使用 Win32 API 开发，应用程序将被链接到操作系统（OS）内核中，这样就避免了应用程序的安装部署。它的缺点在于一旦修改代码，必须再重新编译镜像，容易调试 OS 内核但不容易调试应用程序。

(3) Embedded Visual C++ (EVC)

从此工具的名字上就可以看出 Embedded Visual C++ 与桌面的 Visual C++ 关系密切，它们的功能基本相同，连一些快捷键都是一样的，熟悉 VC6 的人可以很快上手。由于实验室技术积累的原因，本系统选用了 EVC 作为开发工具。

图 5.8 是 EVC 集成开发环境的界面，除了特定工具栏外，它与 VC 的集成开发环境类似。

图中界面组成从上到下依次是菜单条、标准工具栏、特定工具栏、项目文件列表（中左）、文本编辑区域（中右）、信息提示区域。特定工具栏中可选择设备类型、仿真器类型和 CPU 类型。

集成开发环境包含的组件主要有：文本编辑器、资源编辑器、编译器、平台管理器、调试器、代码浏览器、类向导、应用程序向导、远程工具、在线帮助。

下面通过一个简单的绘图程序进一步体会一下 EVC 的特性。

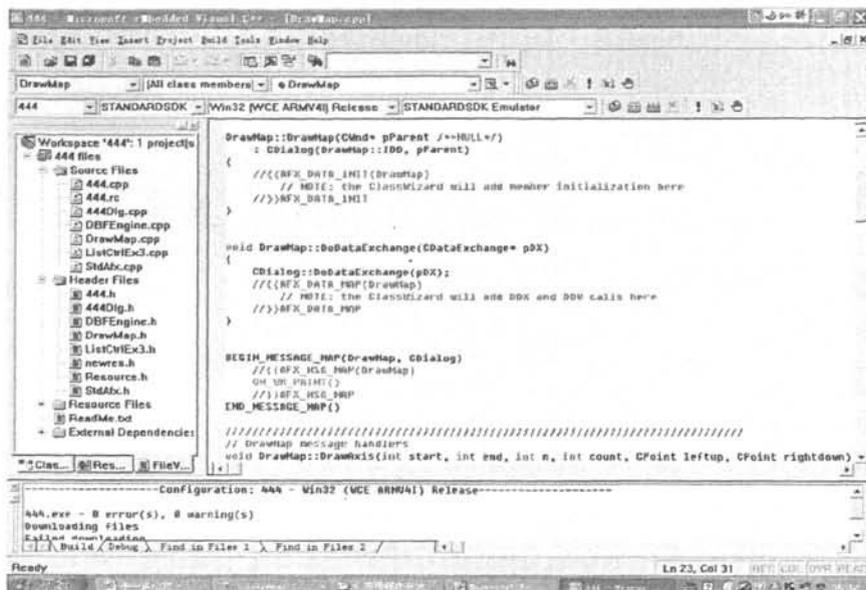


Fig 5.8 EVC integrated development environment

图 5.8 EVC 集成开发环境

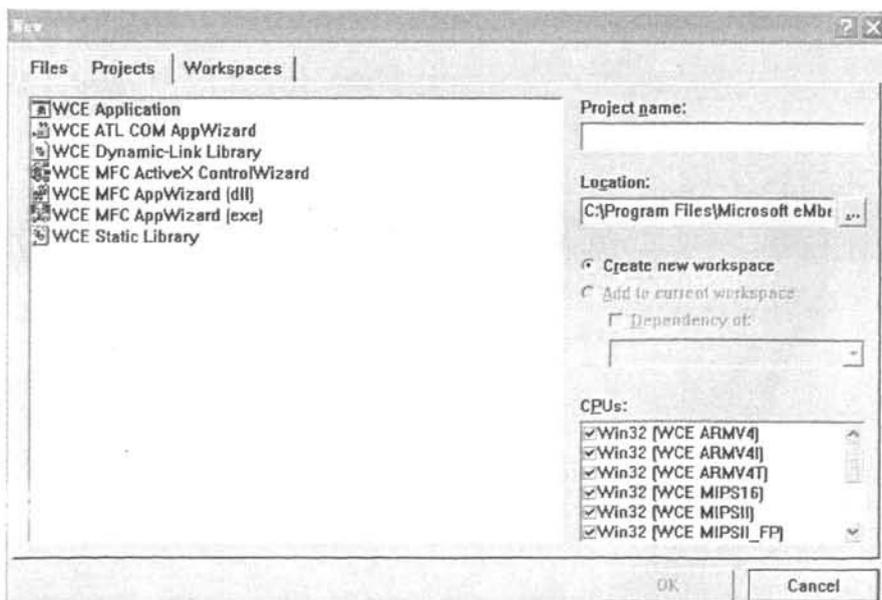


Fig 5.9 Build a new project with EVC

图 5.9 用 EVC 生成一个新项目

来看下面的程序，该程序用来画两条线和一个矩形。

```
void CDrawGraphView::OnDraw(CDC* pDC)
{
    CDrawGraphDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    CRect crect;
    CPen *pOldPen;
    CPen *pMyPen = new CPen();
    int MaxX;
    int MaxY;
    GetClientRect(&crect);           //得到窗口区域
    MaxX=crect.Width ();
    MaxY=crect.Height ();
    //创建新画笔
    pMyPen->CreatePen (PS_SOLID,1,RGB(0,0,0));
    pOldPen = pDC->SelectObject (pMyPen);
    pDC->MoveTo (0,MaxY/2); //设置绘线点为水平
    pDC->LineTo (MaxX,MaxY/2);//画一条水平线
    pDC->MoveTo (MaxX/2,0);
    pDC->LineTo (MaxX/2,MaxY);//画一条中垂线
    CBrush *pOldBrush;           //保存系统原有画刷
    CBrush brushGreen(RGB(0,255,0)); //创建一个绿色画刷
    pOldBrush=pDC->SelectObject (&brushGreen);//选择绿色画刷
    crect.SetRect (0,0,MaxX/2,MaxY/2); //设置绘图区
    crect.DeflateRect (30,30);     //将绘图区缩小 30 个像素
    pDC->Rectangle (crect);       //画矩形
    pDC->ExtTextOut(MaxX/4,MaxY/4,ETO_CLIPPED,NULL,_T("Green
Rectangle"),NULL);//输出说明性文字
}
```

从中可以看出 EVC 与 VC 的两点不同:

(1) 标识常量字符串用的是 `_T("Green Rectangle")`(或用 `L"Green Rectangle"`), 而不是 `"Green Rectangle"`;

(2) 用 `ExtTextOut()` 输出字符串, 而不是用 `TextOut()`(也可以用 `DrawText()`)。

第一点说明 EVC 中使用的是 UNICODE 字符集, 它与 ANSI 字符集有三点不同。a. ANSI 是一个字节, UNICODE 是两个字节; b. 声明: `CHAR->WCHAR` or `TCHAR`; c. 字符串函数: `strlen()->wcslen()`, `strcmp()->wcscmp()`。第二点说明 EVC 中使用的 MFC 类库及其类成员函数与 VC 中不完全相同。MFC 是一个基于 C++ 语言的面向对象的程序设计框架, 它大大简化了 Windows 编程工作。它的优点在于: MFC 提供了一个标准化的结构, 开发人员可从一个较高的起点编程, 从而节省大量的时间; 提供了大量的代码, 程序的控制主要由 MFC 框架完成。Windows CE 中的 MFC 对桌面的 MFC 做了一定的改动, 增加了一些 Windows CE 特有的类, 并且删除了对其他一些类的支持。

除了上述两点外, EVC 还有以下的不同点:

(1) 引入了平台管理器的概念

在桌面 WINDOWS 平台中, 没有平台的概念, 因为 VC 开发的程序就是在桌面版的 WINDOWS 上运行。但在 WINCE 下, 操作系统是定制的, 所以开发都是针对某一平台。可以在新建项目的时候选择针对的平台(如图 4.9 的右下角)。在 Tools 下选择 `Configure Platform Manager` 弹出的列表是该系统上已经安装好了的平台 SDK, 你可以选择你要开发的平台 SDK。你也可以删除, 添加平台, 设置其中的属性, 测试与该设备的连接等。

(2) 远程工具系列

由于 Windows CE 下的应用程序开发并不是在 PC 机上运行, 而是在 CE 设备上运行, 如果想通过 PC 查看 Windows CE 设备中的某些信息, 就要借助于远程工具了。EVC 提供了十个远程工具, 依照功能可以分为三类: 用来进行测试的 `Remote Heap Walker`、`Remote Process Viewer`、`Remote Spy`; 用来进行性能监测的 `Remote Call Profiler`、`Remote Kernel Tracker`、`Remote Performance Monitor`; 用来进行远程信息管理的 `Remote File Viewer`、`Remote Registry Editor`、`Remote System Information`、`Remote Zoom-in`。图 5.10 对功能进行了具体的介绍。

工具名称	描述
Remote Call Profiler	Provides profiling and analysis tools within a graphical user interface (GUI) that enable you to identify algorithmic bottlenecks in your applications.
Remote File Viewer	Displays a hierarchical view of the file system on a target device.
Remote Heap Walker	Displays information about heap identifiers and flags for processes that are running on a target device.
Remote Kernel Tracker	Displays a representation of the execution of the operating system (OS) on a target device.
Remote Performance Monitor	Measures the performance of a target device.
Remote Process Viewer	Displays a list of processes and threads running on a target device.
Remote Registry Editor	Displays the registry for a target device and enables you to manage the registry.
Remote Spy	Displays a list of windows opened on a target device and messages generated by those windows.
Remote System Information	Displays system settings and properties for a target device.
Remote Zoom-in	Displays an image from the screen of a target device.

Fig 5.10 Remote tools

图 5.10 远程工具

(3) 通过 ActiveSync 与设备交互，调试开发

ActiveSync 提供了一种支持，用来同步基于桌面 Windows 系统和基于 CE.NET 的便携式系统间的数据，它使用时间标记(stamps)和用户参数来跟踪两个系统间的数据改变，并传送相应的数据使得每个系统上的数据都是最新的，过时的或者冗余的数据会被删除。ActiveSync 支持 USB, 红外, 调制解调器, 以及 Ethernet 等多种连接。在 EVC 下也可以使用该软件来与远端设备进行同步调试与运行。在远程调试的时候，你可以在远程设备运行的程序上设断点。同在 VC 下的程序一样，远程设备程序会在断点处停下，并与本地 EVC 的 debugger 同步信息（系统与应用程序堆栈的快照等）。ActiveSync 的界面如图 5.11 所示。

(4) 模拟器

在没有硬件设备的条件下，可以使用 EVC 自带的模拟器，基本上可以达到和真实机器上一样的运行效果。

表 5.1 更清楚的反映 EVC4.0 和 VC6.0 的不同点。

处理器调试每次都要下载最新编译的可执行文件到处理器上，模拟器调试会占用大量的电脑资源而且速度较慢，因此在本设计的过程中采用的是 EVC 开发的常用做法，即在 VC 下编译执行得到满意的结果后，再移植到 EVC 上，只需要改动相应的函数即可。在 EVC 上编译生成 EXE 文件下载到 ARM 处理

器上，就可以运行了。



Fig5.11 ActiveSync frame

图 5.11 ActiveSync 界面

表 5.1 EVC4.0 和 VC6.0 的不同点^[33]

Table 5.1 Difference between EVC4.0 and VC6.0

VC6.0	EVC4.0
Win32 API	Win32 API 的子集
C 运行时库	标准 C 运行时库的子集
C/C++编译器	为 x86 模拟器提供的 x86 C/C++编译器 为 ARM 设备提供的 CLARM C/C++编译器 为 MIPS 和 MIPSFP 设备提供的 CLMIPS C/C++编译器 为 Power PC 设备提供的 CLPPC C/C++编译器 为 SuperH 设备提供的 SHCL C/C++编译器
通用开发工具	使用和 Windows 环境类似的开发工具，还附加了一些为 Windows CE 设备提供的远程开发工具，并具有能生成 Windows CE 格式资源的资源编辑器
支持微软基础类 (MFC)	Windows CE MFC 是标准 MFC 类库的子集
支持活动模板库 (ATL)	支持大多数 ATL 类
支持标准模板库 (STL)	不支持 STL

5.2 数据处理与显示平台软件总体结构

在刀具状态监测系统中，数据处理与显示软件总体结构流程如图5.12所示。软件运行后，设置好数据文件的长度，即可打开待分析的数据文件，同时将原始曲线显示出来；设置好小波分解层数、选择使用的小波类型以后，即可进行小波分解与重构，并将各层逼近曲线及细节信号显示出来；最后用小波包分解进行声发射信号的特征提取。用EVC开发的系统界面如图5.13所示。

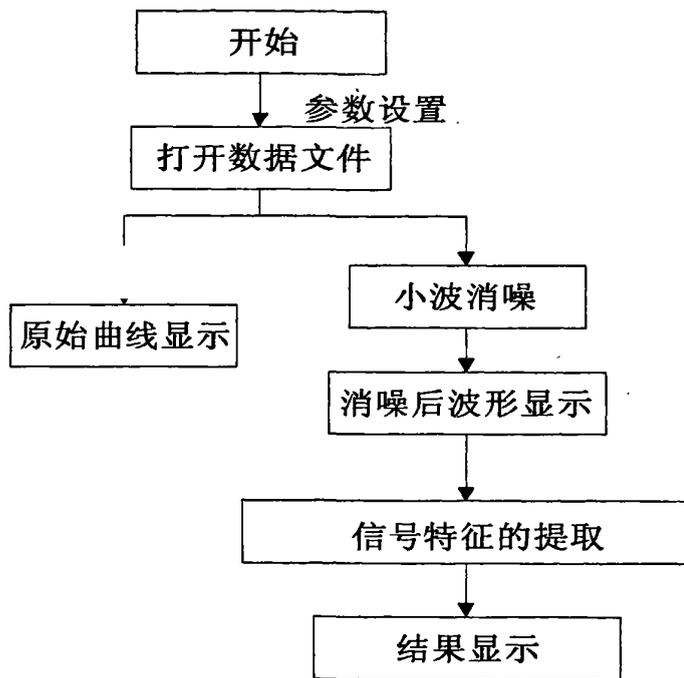


Fig5.12 The overall structure of the software process

图5.12 软件总体结构流程图

在进行小波重构时，系统会弹出如图5.14所示的对话框，供设置小波重构各层加权系数。

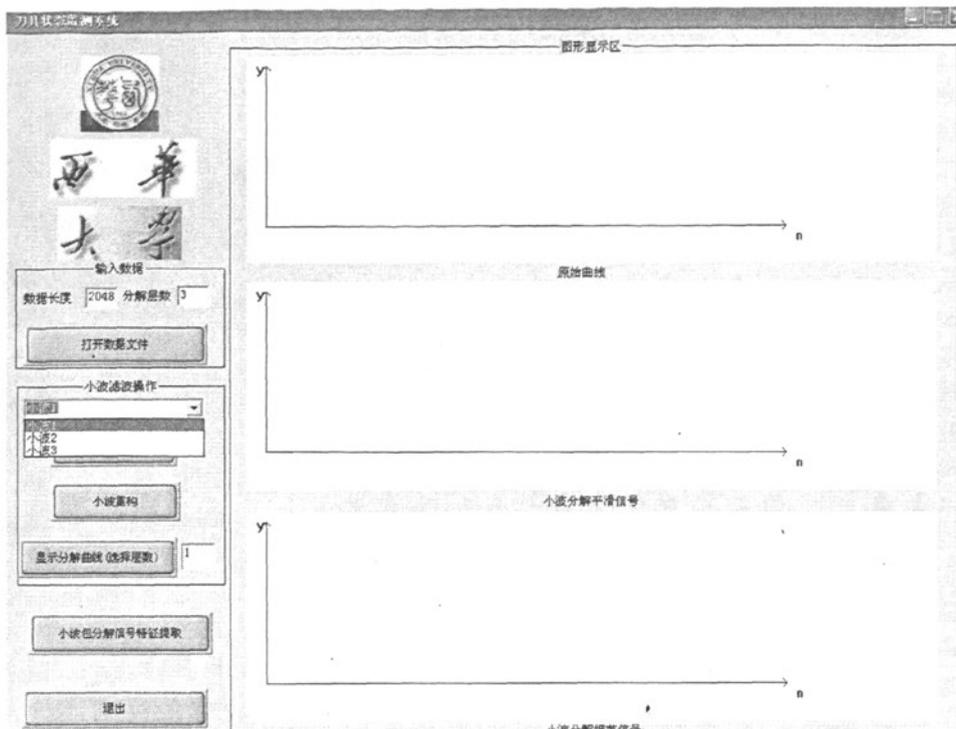


Fig5.13 Data processing and display system main interface

图5.13数据处理与显示系统界面

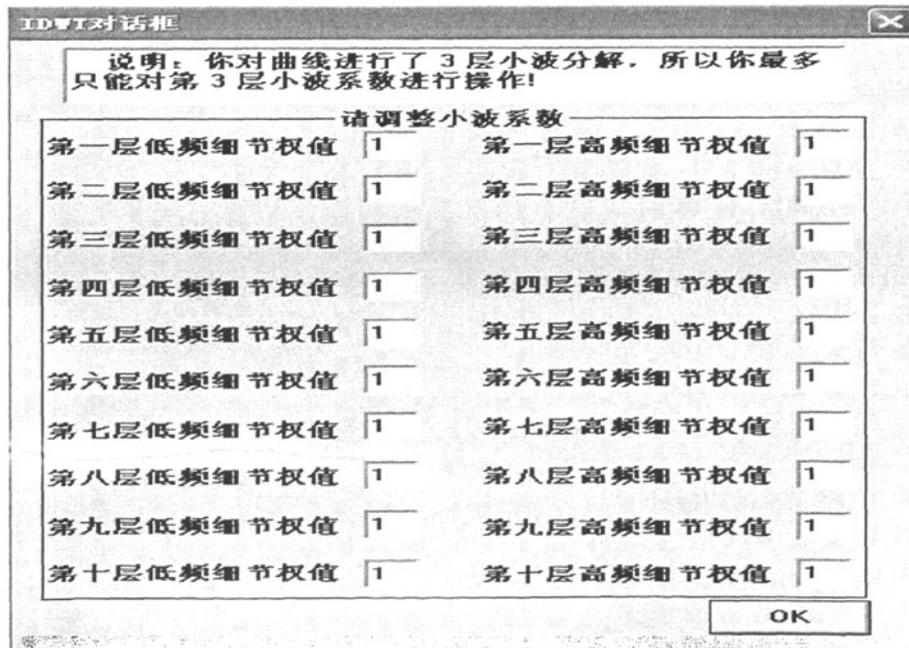


Fig5.14 Parameters settings dialog of Wavelet reconstruction

图5.14 小波重构各层系数设置对话框

5.3 数据处理模块设计

5.3.1 小波消噪 EVC 具体实现

据 3.2.2 节所述, 本文提供三种小波基供小波分解时选择, 即 Haar 小波(小波 1)、Daubechies 小波(小波 2)、Coiflets 小波(小波 3)。在 EVC 中, 是通过构造小波类来实现小波分解与重构算法的, 从而达到去除噪声的目的。小波类 CWaveletDlg 包含在“WaveletDlg.h”头文件中, 下面列出 CWaveletDlg 部分成员函数及成员变量:

```
class CWaveletDlg : public CDialog
{
public:
    double gauss(double mean, double sigma, long *seed);
    double uniform(double a, double b, long int *seed);
    void PaintChonggou(CDC *pDC);
    void PaintAfterDWT(CDC *pDC2, CDC *pDC3);
    void PaintInputData(CDC *pDC);
    CString inputdatafile;
    int DataLength;int J;int Jtmp;
    int lMaxCount,lMaxCount2;double mean;double sigma;
    virtual BOOL OnInitDialog();
    afx_msg void OnSysCommand(UINT nID, LPARAM lParam);
    afx_msg void OnPaint();
    afx_msg HCURSOR OnQueryDragIcon();
    afx_msg void OnInputData();
    afx_msg void OnDWT();//小波分解
    afx_msg void OnPaintAfterDWT();
    virtual void OnOK();
    afx_msg void OnIDWT();小波重构
    afx_msg void OnInputNoise();
```

```
//}}AFX_MSG  
DECLARE_MESSAGE_MAP()  
};
```

下面列出小波分解函数 void OnDWT()与重构函数 void OnIDWT()的部分代码:

(1) 小波分解

```
void CWaveletDlg::OnDWT()  
{  
    if(INPUTDATA==false)  
    {  
        AfxMessageBox("请先输入数据!");  
        return;  
    }  
    UpdateData(true);  
    DataLength=m_DataLength;  
    J=m_J;  
    int i,j,wlen,sca[20],flag[20];  
    int m;//分解的层数  
    char d_name[16]="temp\\d0.txt",c_name[16]="temp\\c0.txt";  
    m=J;  
    j=DataLength;  
    flag[0]=0;  
    for(i=0;i<=m;i++)  
    {  
        flag[i+1]=flag[i]+j;  
        sca[i]=j;  
        j=j/2;  
    }  
    CString strPgmName;  
    strPgmName = m_combo;
```

```
strPgmName.MakeUpper();    //将文件名转成大写
if (strPgmName == "小波 1")
{
    wlen=6;//小波长度
    DWT(g,h,wlen,c,d,m,sca);
}
if (strPgmName == "小波 2")
{
    DWT1(c,d,m,sca);
}
if (strPgmName == "小波 3")
{
    DWT3(c,d,m,sca);
}
ofstream ostrm,ostrmc,ostrmd;
ostrm.open( c_name);
for(i=0;i<DataLength;i++)
{
    ostrm<<c[i]<<endl;
}
ostrm.close();
for(i=1;i<=J;i++)
{
    c_name[6]++;
    d_name[6]++;
    ostrmc.open( c_name);
    ostrmd.open( d_name);
    for(j=0;j<sca[i];j++)
    {
        ostrmc<<c[flag[i]+j]<<endl;
```

```
        ostrmd<<d[flag[i]+j]<<endl;
    }
    ostrmc.close();
    ostrmd.close();
}
AFTERDWT=true;
}
(2) 小波重构:
void CWaveletDlg::OnTIDW()
{
    if(AFTERDWT==false)
    {
        AfxMessageBox("请先进行小波正变换!");
        return;
    }
    UpdateData(true);
    Jtmp=m_Jtmp;
    int i,j,wlen,sca[20],flag[20];
    int m;//分解的层数
    int tmpDataLength;
    int tmp;
    m=J;
    j=DataLength;
    flag[0]=0;
    for(i=0;i<=m;i++)
    {
        flag[i+1]=flag[i]+j;
        sca[i]=j;
        j=j/2;
    }
}
```

```
for(i=0;i<sca[m];i++)
{
    c[flag[m]+i]=c[flag[m]+i];
}
// 创建对话框
CDlgIDWT dlgPara;
// 初始化变量值
memcpy(dlgPara.c,c,sizeof(double)*SHUZU);
memcpy(dlgPara.d,d,sizeof(double)*SHUZU);
dlgPara.J=J;
dlgPara.DataLength=DataLength;
// 显示对话框, 提示用户设定平移量
if (dlgPara.DoModal() != IDOK)
{
    // 返回
    return;
}
// 获取用户的设定
memcpy(c,dlgPara.c,sizeof(double)*SHUZU);
memcpy(d,dlgPara.d,sizeof(double)*SHUZU);
// 删除对话框!!!
delete dlgPara;
CString strPgmName;
strPgmName = m_combo;
strPgmName.MakeUpper();    //将文件名转成大写
if (strPgmName == "小波 1")
{
    //小波长度
    wlen=6;
    //小波重构
```

```
        IDWT(g,h,wlen,c,d,m,sca);
    }
    if (strPgmName == "小波 2")
    {
        IDWT1(c,d,m,sca);
    }
    if (strPgmName == "小波 3")
    {
        IDWT3(c,d,m,sca);
    }
    PAINTCHONGGOU=true;
    PAINTAFTERDWT=false;
    OnPaint();
}
```

5.3.2 小波包分解提取信号能量特征 EVC 实现

据本文 3.3 节所述，小波包分解与重构和小波分解重构有相似之处，故程序设计思路与小波消噪一致，下面以三层小波包分解为例，给出小波包分解提取信号能量特征的程序流程，如图 5.15 所示。

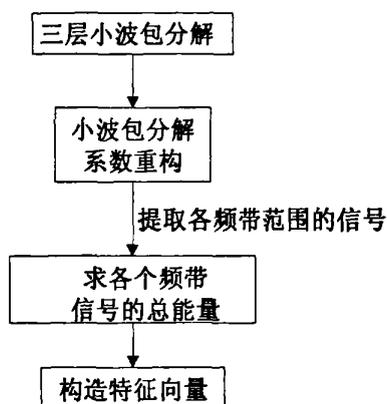


Fig 5.15 Flow diagram of wavelet packet transform

图 5.15 小波包分解特征提取流程图

在 EVC 中,小波包分解特征提取算法是通过设计 Wavelet packet 来实现的,本文作为刀具状态监测系统最终实现的前期研究,目前其 EVC 实现处于调试阶段,在此本文不详细论述。

5.4 显示模块

图形显示是一件较为费时的操作,在实时波形显示时,一定要确保绘图所需的时间小于数据更新的时间间隔,否则先后两次的图形可能会叠加在一起,造成显示混乱。为了提高显示的速度,我们没有直接在窗口上绘出波形图,而是用兼容位图的方法解决这一问题,即先在内存中创建一个与显示区域(一个 static 控件)设备上下文(DeviceContext)兼容的位图,其大小与控件完全相同。图形设备接口(GDI)是在 Windows CE 平台上处理图形图像的方法,负责系统与绘图程序之间的信息交换,以及处理所有 Windows CE 程序的图形输出。GDI 使程序员无需关心硬件设备及设备驱动,就可以将应用程序的输出转化为硬件设备上的输出,大大方便了开发工作。

设备环境(DC)是 GDI 的重要组成部分,是 GDI 创建用来代表设备连接的数据结构,为程序和物理设备间提供连接。DC 包含一组绘图属性设置,涵盖了从背景色、字体到位置、绘图模式、影射模式等诸多方面。使用任何 GDI 输出函数之前,都必须为设备创建一个 DC。

微软 MFC 类库按照面向对象原则封装了几个 DC 类,都是由基类 CDC 派生的。应用程序进行图形图像处理的一般步骤如下:

(1) 将该 GUI(图形用户界面)对象选入 DC 中,同时保存 DC 中原来的 GUI 对象。

(2) 确定所使用的坐标系及映射方式。

(3) 进行图形图像或文字处理(通常在 OnDraw()函数中进行)。

(4) 释放所使用的 DC。

下面以绘制小波分解后的波形图为例列出程序代码如下:

```
void CWaveletDlg::PaintAfterDWT(CDC *pDC2, CDC *pDC3)
{
    UpdateData(true);
```

```
Jtmp=m_Jtmp;
int i,j,k;
int lengthX,lengthY;
.....
//绘制 x 轴刻度
if(DataLength<100)
{
    DataLengthtmp=DataLength-DataLength%10;
    lengthX=int( (DataLengthtmp)/10 );
    jiangeX=int(540/lengthX);
    for (i = 0; i <= lengthX; i ++ )
    {
        pDC2->MoveTo(20+i*jiangeX, 180);
        pDC2->LineTo(20+i*jiangeX, 186);
        str.Format("%d",10*i);
        pDC2->TextOut(20+i*jiangeX, 183, str);
        pDC3->MoveTo(20+i*jiangeX, 180);
        pDC3->LineTo(20+i*jiangeX, 186);
        str.Format("%d",10*i);
        pDC3->TextOut(20+i*jiangeX, 183, str);
    }
}
else
{
    DataLengthtmp=DataLength-DataLength%10;
    lengthX=int( (DataLengthtmp)/10 );
    jiangeX=int(540/lengthX);
    for (i = 0; i <= lengthX; i +=5)
    {
        pDC2->MoveTo(20+i*jiangeX, 180);
```

```
pDC2->LineTo(20+i*jiangeX, 186);
str.Format("%d",10*i);
pDC2->TextOut(20+i*jiangeX, 183, str);
pDC3->MoveTo(20+i*jiangeX, 180);
pDC3->LineTo(20+i*jiangeX, 186);
str.Format("%d",10*i);
pDC3->TextOut(20+i*jiangeX, 183, str);
}
}
//绘制 y 轴刻度
lengthY=IMaxCount2-(int)IMaxCount2%10;
lengthY=int(IMaxCount2+10)/10;
jiangeY=(int)(180/lengthY);
for (i = 1; i < lengthY; i+=1)
{
    pDC2->MoveTo(20, 180-i*jiangeY);
    pDC2->LineTo(16, 180-i*jiangeY);
    str.Format("%d",10*i);
    pDC2->TextOut(2, 180-i*jiangeY, str);
    pDC3->MoveTo(20, 180-i*jiangeY);
    pDC3->LineTo(16, 180-i*jiangeY);
    str.Format("%d",10*i);
    pDC3->TextOut(2, 180-i*jiangeY, str);
}
jiangeX=int(540/DataLength);
tmpDataLength=0;
for(i=1;i<=Jtmp;i++)
{
    tmp=1<<i;
    tmpDataLength+=DataLength*2/tmp;
```

```
    }  
    for (i = 1; i <DataLength/tmp; i ++)  
    {  
        // 10 的倍数  
        pDC2->MoveTo(20+540*i/DataLength,  
90-80*c[i+tmpDataLength-1]/lMaxCount2);  
        pDC2->LineTo(20+540*(i+1)/DataLength,  
90-80*c[i+tmpDataLength]/lMaxCount2);  
        pDC3->MoveTo(20+540*i/DataLength,  
90-80*d[i+tmpDataLength-1]/lMaxCount2);  
        pDC3->LineTo(20+540*(i+1)/DataLength,  
90-80*d[i+tmpDataLength]/lMaxCount2);  
    }  
}
```

5.5 数据管理模块

数据管理模块负责打开数据文件和保存结果。在打开并读取数据文件时，可以采用 WINCE 提供的 `CreatdFile` 函数来实现，该函数的定义如下：

```
HANDLE CreateFile(  
LPCTSTR lpFileName,  
DWORD dwDesiredAccess,  
DWORD dwShareMode,  
LPSECURITY_ATTRIBUTES lpSecurityAttributes,  
DWORD dwCreationDisposition,  
DWORD dwFlagsAndAttributes,  
HANDLE hTemplateFile);
```

其中，`lpFileName` 用于指定要创建或者打开的文件的文件名，参数 `dwDesiredAccess` 用于指定文件的访问权限，参数 `dwShareMode` 用于指定可以给予其他进程的访问权限，`lpSecurityAttributes` 用于指定文件的访问属性等等。

同样 WINCE 也提供了关闭文件的函数，只要调用 CloseHandle 函数即可。打开文件后，WINCE 提供了 ReadFile 函数来读取文件的内容，其定义如下：

```
BOOL ReadFile(  
HANDLE hFile,  
LPVOID lpBuffer,  
.....);
```

要将小波消噪等分解结果保存到创建的文件中，调用 WriteFile 函数即可。

5.6 参数设置模块

软件运行后，在打开数据文件之前，可以进行数据长度的设置；打开数据文件以后，应先设置好小波分解的层数，选择小波的类型、显示小波分解及重构的曲线层次等参数。同时，系统也力求在友好界面的进一步设置及美化方面做大量的改进。

5.7 与上位机 USB 通信

基于 ARM& Windows CE 的嵌入式系统，可以通过 USB 通信协议实现与 PC 机 Windows 的 ActiveSync 连接。

实现 ActiveSync 连接之后，就可以很轻松地将文件传到 PC 端；同时，在开发 eMbedded Visual C++4.0 应用程序的过程中，还可以远程在线调试；另外，通过 ActiveSync 连接，也可以在 PC 端查看 Windows CE5.0 的注册表、进程运行情况、系统信息等等。

6 系统软件调试及运行结果

6.1 数据来源

在刀具的加工过程中，分别对刀具正常切削和刀具磨损两种状态下的切削信号通过声发射传感器进行数据采集。

试验条件如下^[60]：

切削速度：0.0254m/s

切削用量：0.013mm/转

切削深度：4.3mm

工件材料：AISI1018

刀片材料：TNMA432

采样频率：4MHz

滤波截止频率：100KHz—1MHz

采样点数：2048

通过试验所采集的刀具在正常切削和刀具磨损两种切削状态下的切削信号经前置放大，滤波，A/D 转换处理后的数据如表 6.1 和 6.2 所示。其中表 6.1 为正常切削时的典型数据，共采集 2048 个点；表 6.2 为磨损时的典型数据，共采集 2048 个点。

表 6.1 刀具正常切削时的典型数据

Table6.1 The typical data of tool normal cutting

NN1=[- 2.000000 - 1.000000 - 1.000000 - 3.000000 - 6.000000 - 5.000000
 - 1.000000 2.000000 1.000000 2.000000 1.000000 - 3.000000
 - 0.000000 9.000000 11.000000 9.000000 4.000000 - 4.000000
 - 8.000000 - 10.000000];

表 6.2 刀具磨损时的典型数据

Table6.2 The typical data of tool wearing

WW1=[18.000000 22.000000 18.000000 16.000000 20.000000 24.000000

24.000000 16.000000 7.000000 - 4.000000 - 15.000000 - 18.000000
 - 18.000000 - 12.000000 - 6.000000 - 5.000000 - 11.000000 - 11.000000
 - 4.000000 1.000000];

6.2 系统调试运行结果

本文在刀具正常和磨损状态下，各选择一个样本来进行软件调试实验。设置数据长度为 2048，以三层小波分解（选择小波 1，即 Haar 小波）为例，进行小波消噪。图 6.1—图 6.6 分别为刀具正常切削和刀具磨损时原始信号、小波分解与重构后各层逼近信号及细节信号曲线。

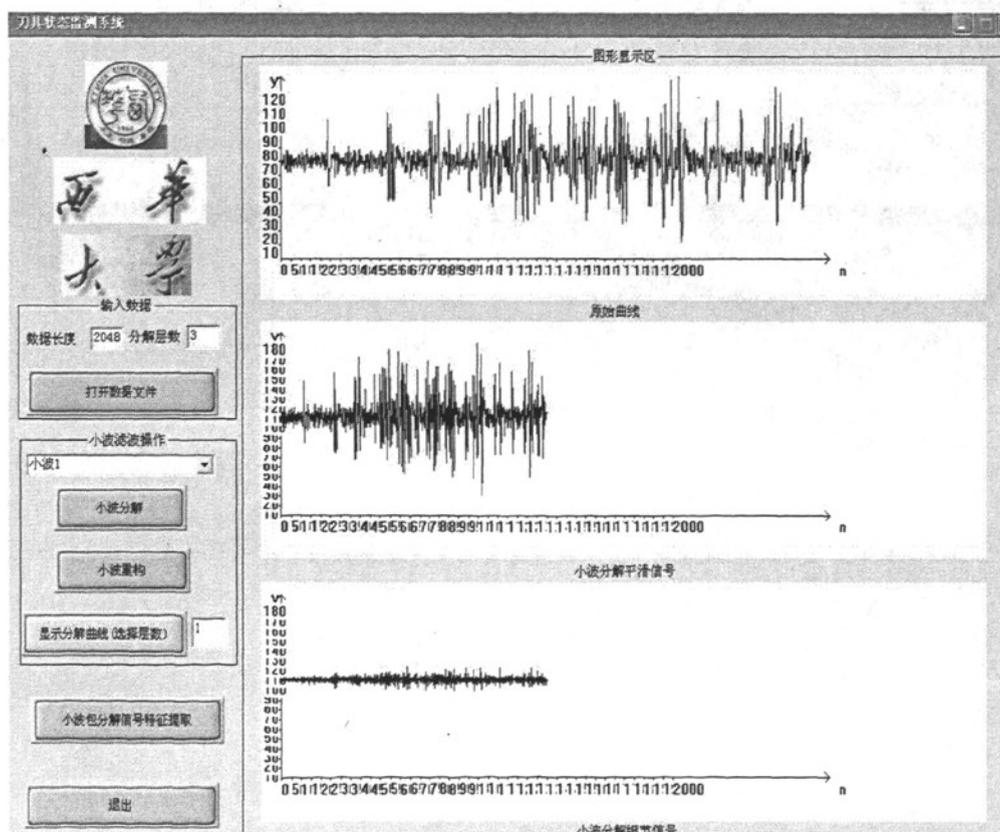


Fig6.1 Signal wavelet de-noising experiment while normal cutting (the first layer)

图 6.1 正常切削时小波消噪运行结果（第 1 层曲线）

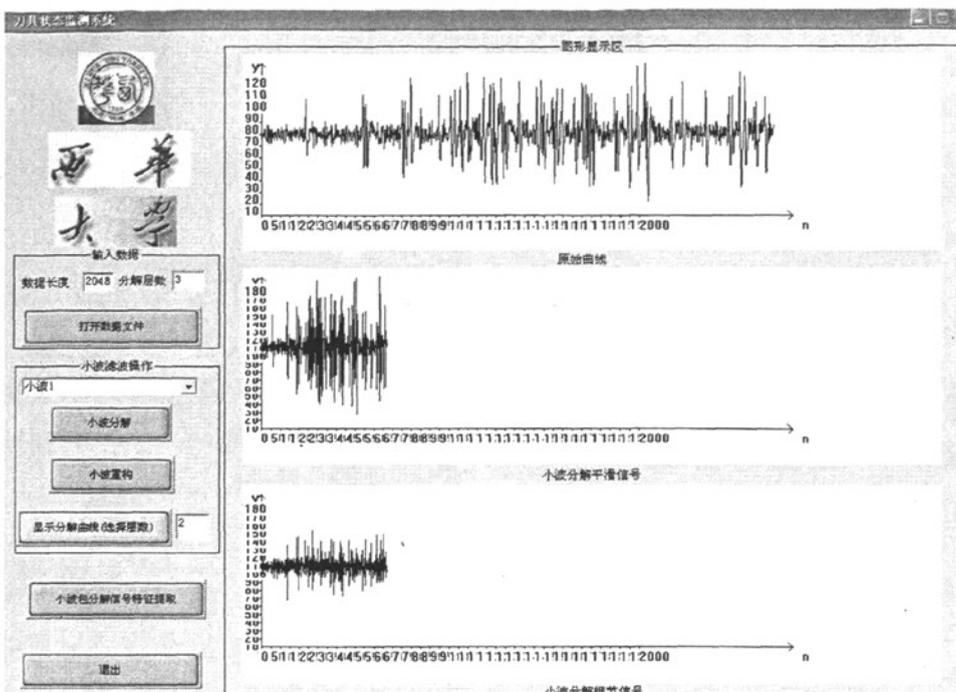


Fig6.2 Signal wavelet de-noising experiment while normal cutting (the second layer)

图 6.2 正常切削时小波消噪运行结果 (第 2 层曲线)

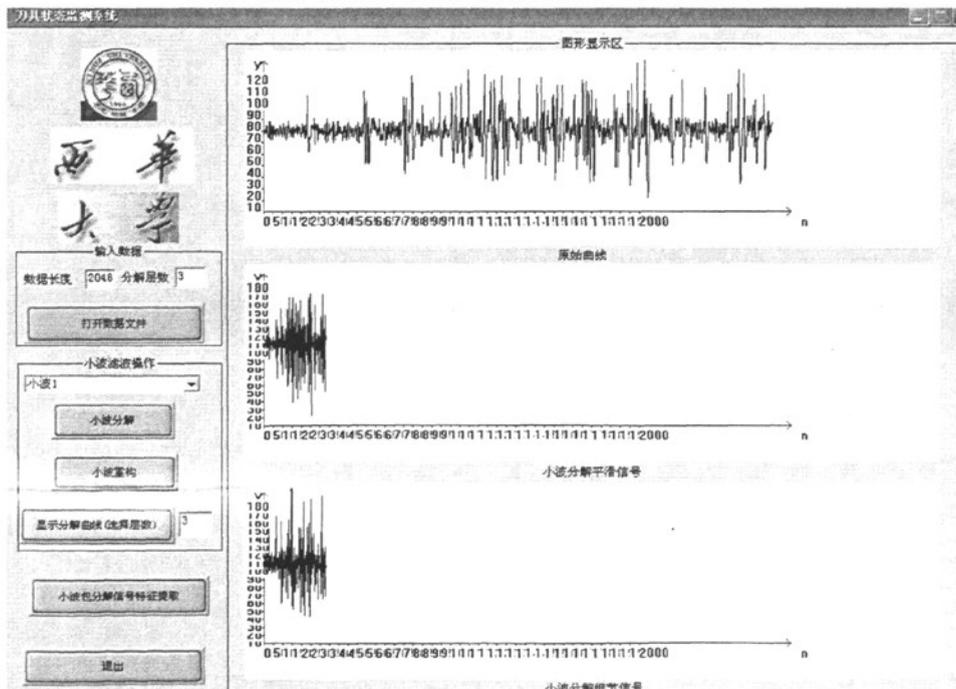


Fig6.3 Signal wavelet de-noising experiment while normal cutting (the third layer)

图 6.3 正常切削时小波消噪运行结果 (第 3 层曲线)

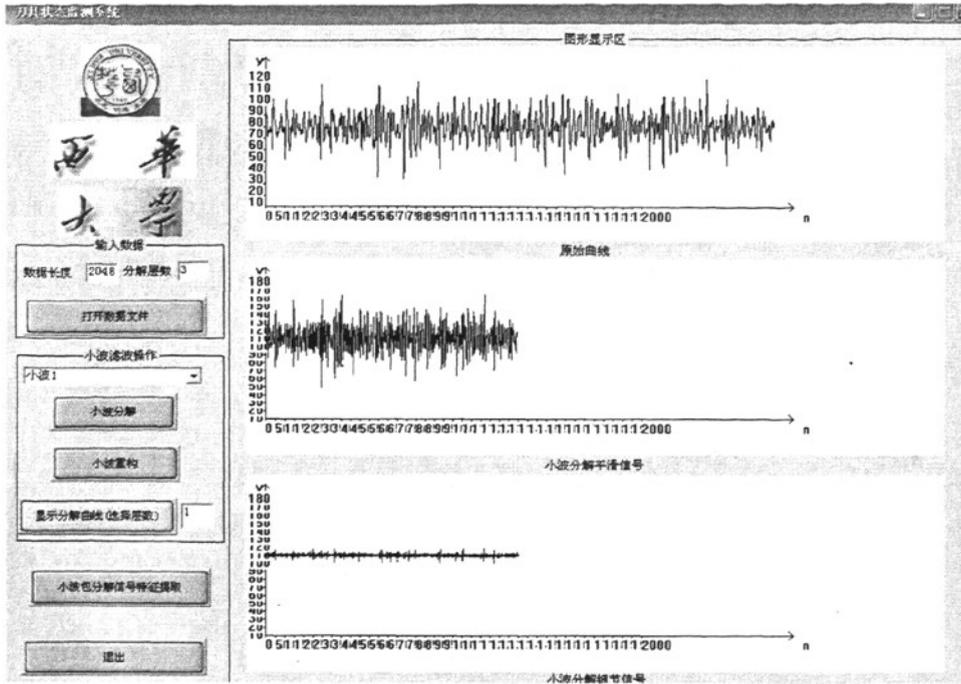


Fig6.4 Signal wavelet de-noising experiment while tool wearing (the first layer)
图 6.4 刀具磨损时小波消噪运行结果 (第 1 层曲线)

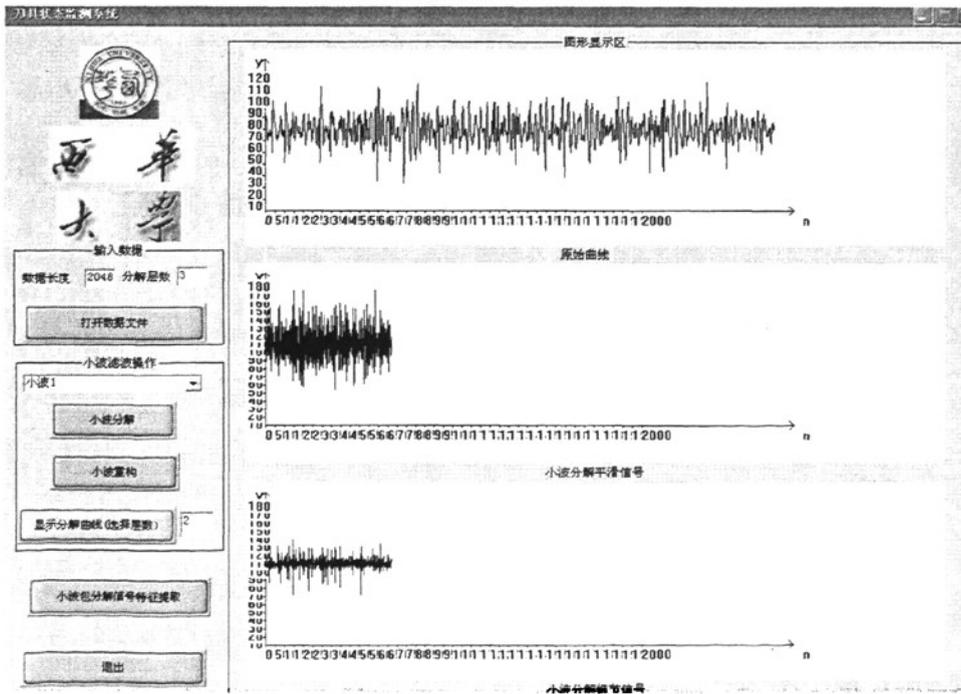


Fig6.5 Signal wavelet de-noising experiment while tool wearing (the second layer)
图 6.5 刀具磨损时小波消噪运行结果 (第 2 层曲线)

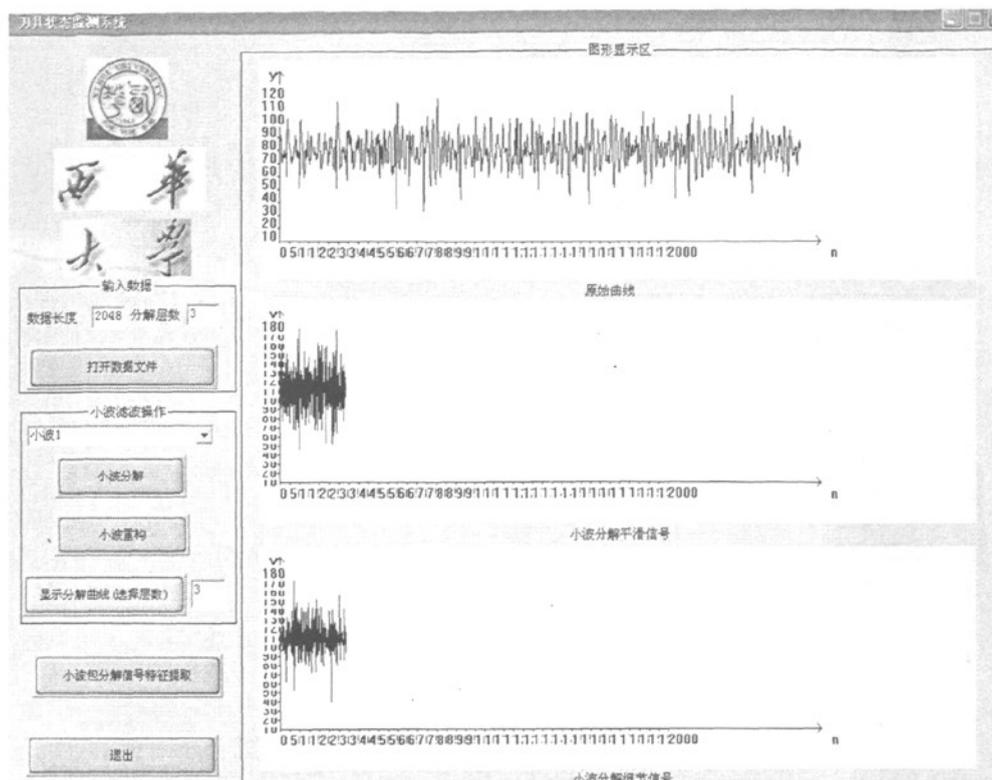


Fig.6.6 Signal wavelet de-noising experiment while tool wearing (the thirist layer)

图 6.6 刀具磨损时小波消噪运行结果 (第 3 层曲线)

从图 6.1—6.6 可以看出：刀具正常切削及磨损时，第一层的细节信号的值较小，可以认为是由传感器和状态噪声的高频分量引起的局部干扰，第二、三层的细节信号包含了主要的信号模式，其中第二层的细节信号最接近于原始曲线。

对刀具状态信号进行小波消噪后，如本文第 3 章所述，接下来就能用小波包分解提取声发射信号各频段的能量值作为特征向量。提取到的特征向量作为支持向量机分类器的输入，进行训练，进而对刀具磨损状态进行模式识别。目前小波包分解特征提取算法尚处在调试阶段。

7 总结与展望

本文研究的目的是实现刀具状态的自动化监测与分析，并将测量的结果显示出来，以便帮助用户及时了解刀具的磨损状态，必要时将重要数据传送到上位机进行模式识别等进一步处理，便于今后的统计和总体分析。本文主要做了以下工作：

- (1) 分析了声发射信号与刀具磨损状态的相关性，验证了利用声发射信号进行刀具状态监测的可行性；
- (2) 确定刀具状态监测系统数据处理平台的整体方案，包括系统整体架构、软硬件设计方案。ARM & WinCE 构成本系统的数据处理与显示平台，EVC(Embedded Visual C++)为图形界面应用程序开发工具；
- (3) 构建了数据处理与显示平台。选用 MagicARM2410 实验开发平台，简化了硬件设计；根据系统的功能需求，进行 ARM 平台的接口设计、操作系统和必要的驱动程序的剪裁及移植；
- (4) 完成了数据处理与显示应用软件设计。系统软件包括界面模块、数据管理模块、数据处理模块、图形及结果显示模块、参数设置模块等，其中数据处理模块的小波消噪、小波包分解特征提取算法是课题研究的重点，小波消噪的 EVC 程序本文已调试通过；
- (5) 实现了 ARM & WinCE 平台与 PC 机的实时可靠通讯。

综上本文将先进的嵌入式技术与刀具状态监测技术结合起来，拟研发出一种快速、精确、人机界面友好的刀具状态监测系统。不过，在 ARM & WINDOWS CE 平台上构建刀具状态监测系统，本文尚处在前期研究阶段，给出了系统的初步软硬件解决方案，但仅仅对数据处理与显示平台做了重点研究。实现了小波消噪算法，并调试通过，运行良好，为进一步的系统开发奠定了基础。传统的刀具状态监测系统，上述智能算法只能通过高性能的 PC，利用 Matlab 等工具进行仿真，并未真正应用到实际中。

当然，刀具状态监测系统是基于传感器技术、信号处理与分析，智能控制、模式识别等技术的复杂系统。本文做了一些前期研究，实现了部分功能，还需要进一步在增强系统功能、人机界面的友好性、系统设计上下很大的功夫。在

以后的工作和科研中，还需要在以下几个方面进行进一步的研究：

- (1) 多传感器的融合。刀具状态信号的监测方法通用性较差，对于同一种监测方法，不能用于刀具加工条件或加工参数改变的情况，各有优缺点和使用场合。例如当利用声发射信号进行监测时，刀具材料变化后，不得不重新修改算法，有一定得局限性，可以采用多传感器相结合来监测刀具状态信号；
- (2) 特征提取算法的完善。特征提取算法是刀具状态监测技术的核心，本文实现了声发射信号的小波消噪 EVC 开发，并对小波包分解特征提取算法做了深入的理论研究，但尚未在 EVC 下完全实现。提取刀具特征后就能用支持向量机等分类算法进行模式识别；
- (3) 刀具磨损模式识别的研究。实验室对利用支持向量机进行刀具磨损识别做了大量的理论研究，但都是在高性能 PC 机上进行仿真。在 ARM & WINDOWS CE 平台实现模式识别程序开发是系统的目标；
- (4) 系统功能的进一步扩充以及人机界面友好性的改进。在保证系统必要的性能的基础上，增强系统的界面的友好性，以使系统操作方便。最终将传统刀具状态监测系统的所有功能在 ARM & WINDOWS CE 平台实现。

参考文献

- [1] 雷彬.国外铣削刀具磨损量自动检测方法综述.军械工程学院学报, 1999, 11(2): 73-78
- [2] 刘彬, 任玉艳, 冯冀宁.数控车床刀具监测方法的研究.机械研究与应用, 2003, 16(2): 44-45
- [3] 张永清, 刘志峰.非接触式刀具磨损在线检测方法—光纤位移检测法.工具技术, 1996, 30(11): 37-39
- [4] SunXudong,LiuYuchi,ZhangGuoxiong.Photo-electric measuring system for tool edge detection by the technology of computer dynamic display moving image. Transactions of Tianjin University,1999,5(1):47-51
- [5] 熊四昌, 王亚良, 计时鸣等.基于分数布朗运动的工件表面图像分析及其在刀具状态检测中的应用.机床与液压, 2004, (4): 44-46
- [6] J.H.Lee,D.E.Kim,S.J.Lee.Statistical analysis of cutting force ratios for flank-wear monitoring.Materials Processing Technology,1998,74:104-114
- [7] Dae Kyun Bake,Tae Jo Ko,Hee Sool Kim.Real time monitoring of tool breakage in a milling operation using a digital signal processor.Materials Processing Technology,2000,100:266-272
- [8] S.E.Oraby.Monitoring of turning operation via force signals Part 1:Recognition of different tool failure forms by spectral analysis.Wear,1995,184:133-144
- [9] 高宏力, 许明恒, 傅攀.一种在线监测铣刀磨损量的新方法.中国机械工程, 2005, 16(12): 1069-1072
- [10] FREQUENCY AND TIME DOMAIN ANALYSIS OF SENSOR SIGNALS IN DRILLING INVESTIGATION ON SOME PROBLEMS ASSOCIATED WITH SENSOR INTEGRATION.Tool Manufact.35(6):795-815
- [11] Mitsuharu Shiwa. A Review for Present and Future of Acoustic Emission Examination.非破坏检查, 附: 学术演讲会讲演概要集, 2001, 50(2):91—94
- [12] Rangwala S, Dornfeld D. Integration of Sensors via Neural Networks for Detection of Tool Wear States. American Society of Mechanical Engineering Division,1987,25:109—120

- [13] <http://pac.mistrs-holding.com/noesis3.html>
- [14] 纪洪广, 张天森, 张志勇, 蔡美峰无损检测中常用声发射参数的分析与评价. 无损检测, 2001,23(3):289—291
- [15] Freeman S F. Characterization of Lamina and Inter lamina Damage in Graphite-Epoxy Composites by the Deeply Technique. *Composite Materials Testing and Design*,1981,5: 50—57
- [16] 叶琳, 张艾萍. 声发射技术在设备故障诊断中的应用. 新技术新工艺, 2000, (8): 16—19
- [17] Qing Qing Ni, Yasuharu Misada. Analysis of AE Signals by Wavelet Transform. *Journal of Society Materials Science*,1998,47(3):305—311
- [18] 刘时风. 焊接缺陷声发射检测信号谱估计及人工神经网络模式识别研究: [博士学位论文]. 北京: 清华大学机械工程系, 1996
- [19] 许凤旌, 陈积樊. 声发射技术在复合材料发展中的应用. *机械工程材料*, 1997, 21(4): 30—40
- [20] Hiroaki Suzuki, Tetsuo Kinjo, Mikio Takemoto, et al. Fracture-Mode Determination of Glass-Fiber Composites by Various AE Processing. *Progress in Acoustic Emission VIII*, The Japanese Society for NDI, 1996, 47—52
- [21] D.O'Sullivan, M. Cotterell. Temperature measurement in single point turning. *Materials Processing Technology*, 2001, 118: 301-308
- [22] HERCHANGAY. Heat transfer and life of metal cutting tools in turning. *Pergamon*, 1998, 41(3): 613-623
- [23] A Terchi, Y H J Au. Acoustic Emission Signal Processing. *Measurement and Control*, 2001, 4(8): 240—244
- [24] 王琪. 刀具磨损和破损的综合监测法. 新技术新工艺, 2003, 10: 17-18
- [25] Ali A. Houshand, Elijah Kannatey-Asibu, Gary D. Herrin. A Dynamic Model for Tool Wear Detection using Acoustic Emission. *Mechanical systems and Signal processing*, 1995, 9(4): 415-428
- [26] H.V. Ravindra, Y.G. Srinivasa, R. Krishnamurthy. Acoustic emission for tool condition monitoring in metal cutting. *Wear*, 1997, 212: 78-84
- [27] MANBIR S. SODHI, KHALIL THLIOUINE. SURFACE ROUGHNESS MONITORING

- USING COMPUTER VISION. *Tools Manufact*,1996,36(7):817-828
- [28] SCOTT A.COKER,YUNG C.SHIN.IN-PROCESS CONTROL OF SURFACE ROUGHNESS DUE TO TOOL WEAR USING A NEW ULTRASONIC SYSTEM. *Tools Manufact*.1996,36(3):411-422
- [29] Zafer Tekiner,Sezgin Yesilyurt.Investigation of the cutting parameters depending on process sound during turning of AISI 304 austenitic stainless steel.*Materials&Design*,2004, 25:507-513
- [30] J.Kopac,S.Sali.Tool wear monitoring during the turning process.*Materials Processing Technology*,2001,113:312-316
- [31] I.Daubechies, *Ten Lectures on Wavelets*. Philadelphia:SIAM,1992
- [32] W.Lawton.Tight frames of compactly supported wavelets, *J.Math.Phys*,31:1898—1901, 1990
- [33] 田东风.Windows CE 应用程序设计[M], 北京: 机械工业出版社, 2003.10
- [34] (美)[T.A.高金]Terence A.Goggin 著.Windows CE 高级开发指南[M], 北京: 电子工业出版社, 2000.3
- [35] 白伟平等.基于 ARM 的嵌入式 BootLoader 浅析[J].*微计算机信息(嵌入式与 SOC)*, 2006:99~100
- [36] (美)[C.默恩希]Chris Muench 著.Windows CE 权威指南[M], 北京: 中国电力出版社
- [37] 宋斌等.*Visual C++ 6.0 教程*[M], 北京: 希望电子出版社, 1999.11
- [38] 王浩等.*Visual C++程序开发案例分析*[M], 北京: 清华大学出版社, 2006.3, P59~112
- [39] 探索工作室.嵌入式开发圣经[M], 北京:中国青年出版社, 2002:1~27
- [40] 姜峥嵘, 基于 WindowsCE 数控系统实时控制的关键技术[J], 合肥工业大学硕士论文, 2006:1~10
- [41] 秦前清.杨宗凯.使用小波分析.西安电子科技大学出版社, 2002
- [42] 唐远炎.王玲.小波分析与文本文字识别.科学出版社, 2004
- [43] 林克正.李殿璞.基于小波变换的去噪方法.哈尔滨工程大学学报, 2000.21(4)
- [44] 李驹光, 聂雪媛等,ARM 应用系统开发详解[M],北京:清华大学出版社, 2003
- [45] 赵红怡, 武梦龙, 曹淑琴. 小波分析在突变信号检测中的应用. 北方工业大学学报, 2004, 16(1): 21-24

- [46] 林凤涛, 曹冲锋. 小波分析在滚动轴承故障诊断中的应用. 华东交通大学学报, 2004, 21(5): 120-122
- [47] 郑灵祥. 嵌入式系统设计与应用开发. 北京: 北京航空航天大学出版社
- [48] 汪兵, 李存斌, 陈鹏等. Embedded Visual C++嵌入式编程. 北京: 中国水利水电出版社
- [49] 周立功等. S3C2410&嵌入式 Windows CE 系统教程. 广州: 广州志远电子有限公司
- [50] Magic2410 用户手册. 广州志远电子有限公司
- [51] 姜峥嵘. 基于 WindowsCE 数控系统实时控制的关键技术[J], 合肥工业大学硕士论文, 2006:1~10
- [52] 张禹等. Platform Builder 集成开发环境研究[J], 微计算机信息, 2005.4
- [53] 张新房等. 使用 Platform Builder 配置 Windows CE 操作系统[J], 单片机与嵌入式系统应用, 2002.10
- [54] 田泽等. 嵌入式系统开发与应用[M], 北京: 北京航空航天大学出版社, 2005:15~25
- [55] 金敏等. 嵌入式系统[M], 北京: 人民邮电出版社, 2006:24~44
- [56] 杜春雷. ARM 体系结构与编程[M], 北京: 清华大学出版社, 2003
- [57] 嵌入式系统 BootLoader 技术内幕, <http://teiko2006.spaees.live.com/>
- [58] 张大波等. 嵌入式系统原理、设计与应用[M], 北京: 机械工业出版社, 2005.1
- [59] 何加铭. 嵌入式 32 位微处理器系统设计与应用[M], 北京: 电子工业出版社, 2006:239~253
- [60] 黄惟公. 线性判别函数对刀具监测中声发射信号的分析. 四川工业学院学报, 1995, 21-34
- [61] (印度) RajKamal 等. 嵌入式系统一体系结构、编程与设计[M], 北京清华大学出版社, 2005:304~309
- [62] 傅曦、齐宇. 嵌入式系统 Windows CE 开发技巧与实例[M], 北京: 化学工业出版社, 2004.1
- [63] 何宗健. Windows CE 嵌入式系统[M], 北京: 北京航空航天大学出版社, 2006.9
- [64] 于明等. ARM9 嵌入式系统设计与开发教程[M]. 北京: 电子工业出版社, 2006
- [65] 王忠民, 王信义, 杨大勇等. 刀具磨损状态在线监测技术, 制造技术与机床, 2000, 6: 39-41
- [66] 吴明晖等. 基于 ARM 的嵌入式系统开发与应用[M], 北京: 人民邮电出版社, 2004.6
- [67] 刘镇清, 陈广. 超声无损监测中的谱分析技术[J]. 无损检测, 2001, 23(2):85-88

- [68] 范雷雷, 施惠丰.声发射技术在设备状态检修中的应用[J].电力设备, 2004, 5(11):41-43
- [69] 张艾萍, 暴丽.声发射技术在化工设备状态检测与故障诊断中的应用[J].化工机械, 2003, 30(2):93-96
- [70] 陈新全, 李西南, 邓敏.金属切削过程中的声发射现象及其研究的意义[J].长沙电力学院学报, 2001, 16(1):75-76
- [71] 耿荣生.声发射信号处理和分析技术.无损检测, 2002, 24 (2): 157-165
- [72] 王强.基于声发射技术及小波分析的砂轮钝化状态监测方法研究.中国海洋大学硕士学位论文.2008.6: 15-16
- [73] 王奉涛等.基于小波包分解的频带局部能量特征提取方法[J].农业机械学报, 2004, 35(5): (177-180)
- [74] 谢秀娟.基于声发射和神经网络的刀具磨损状态识别研究.西南交通大学硕士学位论文, 2005: 21—28
- [75] 刘鹏惠.基于小波分析与支持向量机的刀具状态检测方法的研究.西华大学硕士学位论文, 2006: 17—25

在读期间科研成果

[1] 徐全、胡丹, 车畅等.基于 CAN 和 $\mu C/os-II$ 新型电子测量仪, 中国仪器仪表, 2008.1

[2] 徐全、胡丹, 车畅等.基于冗余 TTCAN 校园用电监测系统设计, 西华大学学报, 2008.5

[3] 顾伟, 费凌, 徐全. 基于 Net—CNC 技术的机床厂网络化改造.工业控制计算机, 2008.08

[4] 吕冬梅, 胡丹, 车场, 徐全.小波包分析和支持向量机在刀具状态识别中的应用, 信息、电子与控制技术学术会议论文集, 2008 年

[5] 2007.6 至今 四川省科技厅重点项目“支持向量机在刀具状态监测中的应用”

项目描述: 本项目拟应用支持向量机的分类思想, 通过支持向量机的分类模型, 对刀具的运行状况进行监控, 从而达到故障诊断的目的。前期主要进行算法研究, 初步进行软硬件方案设计。

[6] 2007.12~至今 支持向量机在系统辨识中的应用

项目描述: 本课题要达到的主要目标是基于支持向量机和小波分析技术, 获得新的复杂系统建模的新技术、新理论和新方法, 并将其应用到系统辨识中。

致 谢

几经彷徨求索，论文终于得以完成。回顾这一程求学路，记忆里满是老师的悉心栽培和同学的快乐相伴，在此我要向他们表达最诚挚的感谢。

三载教诲，师恩难忘！在此论文完成之际，谨向我尊敬的导师胡丹教授致以最诚挚的谢意和最崇高的敬意。胡老师高尚的人格、高度的责任心、丰富的实践经验、渊博的专业知识、宽以待人的处事态度是我终身的财富。是在胡老师的悉心指导和无私帮助下才使我的学业得以顺利完成，毕业之际，真诚的祝福胡老师健康幸福。

感谢车畅老师，正是她的鼓励和关心，帮助我克服各种困难，在此我要向我亲爱的师姐道一声真诚的谢意！同时也要感谢韦兴平老师、吕冬梅同学、安斌同学以及师弟师妹，在研究过程中，以及三年的学习生活中，给予我的帮助和支持。在读研期间，机械学院的很多老师也给了我很大的帮助，在此表示深深的感谢。

最后，我谨向导师、论文评审老师、家人、所有帮助过我的老师、同学和朋友们表示最诚挚的感谢！

作者: [徐全](#)
学位授予单位: [西华大学](#)

相似文献(2条)

1. 学位论文 [吕冬梅](#) [支持向量机在刀具故障诊断中的应用](#) 2009

加工过程中刀具的磨损将造成工件的精度及粗糙度变差,甚至造成工件的报废,机床停机等故障。因此,刀具故障诊断成为了工况监测中的一个极为关键的技术环节。

首先采用声发射技术采集刀具切削状态的数据,使用小波包分析技术提取能量谱,进行特征提取。然后阐述了支持向量机理论,基于已提取的特征,构建支持向量机分类器,用不同规模的数据进行训练测试,得到了理想的分类效果。

本文的主要内容如下:

第一章综述了刀具状态识别目的和意义,分析了国内外声发射技术和小波包分析技术的发展和研究现状。论述了统计学习理论和支持向量机理论研究和应用发展。最后给出了本文的主要研究内容,并给出论文的总体框架。

第二章着重介绍了声发射技术的原理及特点,并分析了声发射信号的特征提取方法。

第三章介绍了小波包分析技术及其特点。根据刀具声发射信号具有随机性的特点和小波包分析良好的时频局部化特性,本文给出了小波包分解提取信号特征的方法,分析结果表明了小波包分解提取的能量特征能够正确地反映刀具切削状态。

第四章介绍了统计学习理论的重点内容,简要介绍了支持向量机算法的主要思想和特点。利用小波包分解提取刀具声发射信号各频段的能量值作为特征向量,构建支持向量机分类器。通过实验证明了支持向量机在刀具状态识别中的可行性和有效性,实现了对刀具磨损状态的识别,取得了比较满意的效果。

第五章对全文的主要工作和研究成果进行了总结,并指出了有待进一步改进和研究的方向。

2. 学位论文 [孙振华](#) [刀具破损声发射监控系统的研制](#) 2002

本文主要讲述了一种刀具破损声发射监测系统的开发研制过程以及其功能构成和性能分析。

论文首先介绍了当今世界上机械制造工业的发展方向,提出了对刀具加工过程进行实时监测的重要性,通过对刀具加工过程监测技术和声发射技术的发展历程分析,得出将声发射技术应用于刀具状态监测是可行并且有效的。

其次,本文描述了系统结构,确立了以声发射传感器、数字存储示波器、工业控制计算机和监控软件为基本构成的实验及监控系统构成,通过对实验数据分析得出刀具破损的声发射信号特征波形,运用虚拟仪器技术和C++编程语言完成系统功能,详细介绍了刀具状态声发射信号的数字处理方法及应用软件的编制过程。

最后,本文分析了该刀具破损声发射监测系统的性能,通过现场实际应用实验得出系统的误报率和漏报率,总结了系统的应用情况和优缺点,对系统的改进和推广提出意见。

本文论述的研究课题其思想一方面要研制出一套系统,即刀具破损声发射监测系统,该系统在加工过程中出现破损情况下能够对刀具的破损状态及时准确地发出警报,以提醒操作工刀具已经损坏,及时换刀;另一方面又要考虑到追踪当前国际先进技术,对刀具加工状态作定性分析,为将来技术上的发展和产品的推广应用作准备。

本文链接: http://d.g.wanfangdata.com.cn/Thesis_Y1556225.aspx

下载时间: 2010年5月27日