

# 嵌入式高分辨率图像采集系统的设计与实现

王振煜, 葛万成

(同济大学 电子与信息工程学院, 上海 200092)

**摘要:**随着信息技术的发展,人们对图像采集系统图像分辨率的要求也越来越高,针对图像采集系统对图像精度的要求,提出了在3.5英寸嵌入式单板机平台上使用高分辨率数码相机采集现场图像数据并通过Internet传输到远程终端的设计方案,实现了高分辨率图像数据的自动获取和实时传输及显示,并具体给出了图像采集单元的软件实现和远程监控端的软件实现方法。

**关键词:**嵌入式单板机;数码相机;XP Embedded;套接口

中图分类号:TP391.41

文献标识码:A

文章编号:1673-629X(2008)02-0167-03

## Design and Implementation of an Embedded High Resolution Image Acquisition System

WANG Zhen-yu, GE Wan-cheng

(School of Electronics and Information Engineering, Tongji University, Shanghai 200092, China)

**Abstract:** With the development of information technology, resolution requirement of image acquisition system is increasingly higher. Aiming at the high resolution requirement of image acquisition system, presents a design based on the platform of 3.5-inch embedded subcompact board computer, using digital camera for image acquisition and internet for data transmission, achieving the automatic acquisition, real-time transmission and display of high-resolution images, and gives the implementation of software in detail both in image acquisition unit and remote monitoring unit.

**Key words:** subcompact board computer; digital camera; XP embedded; socket

### 0 引言

图像采集系统正从模拟采集向数字采集发展,其中数字图像采集系统中的嵌入式图像采集系统由于其优越的性能越来越受到人们的关注。随着嵌入式技术的发展和互联网的普及,利用Internet作为传输媒介的嵌入式图像数据采集及传输系统也越来越流行。在对图像精度要求比较高的特定工控系统中,为了实现高清晰度的图像监控,满足数字图像数据高速传输的要求,提出了在3.5英寸嵌入式单板机平台上采用高像素数码相机,基于以太网的嵌入式数字图像监控系统。数码相机可获取高分辨率图像数据。3.5英寸嵌入式单板机具有便携的体积和丰富的资源,其运行速度快,稳定性高,可以与外围硬件设备和软件系统实现无缝连接,提高系统的开发速度,同时也更易于功能的扩展。同时采用的Windows XP Embedded作为一种以

组件化的形式展示Windows的强大优势的嵌入式操作系统,它使开发者可以快速地构造先进可靠的嵌入式设备<sup>[1]</sup>。由于基于Win32编程模型,Windows XP Embedded使开发者通过使用类似Visual Studio.NET的开发工具和最普通的个人电脑硬件便可以开发出相应的应用程序而大大缩短了开发时间。

### 1 系统硬件结构

图1描述了利用嵌入式单板机开发的图像采集系统的框图。该系统由图像监控单元和图像采集单元组成。图像监控单元由PC机担当。采集单元由Olympus公司的SP-500UZ数码相机和研扬公司的Gene-8310 3.5英寸嵌入式单板机组成。SP-500数码相机具有10倍光学变焦镜头和600万像素高分辨率CCD。Gene-8310嵌入式单板机外扩了USB接口,提供10/100Mbps以太网接口,支持CF卡存储。图中,嵌入式单板机上运行裁减所得的XP Embedded操作系统,控制数码相机获取图像数据,将数据缓存后发送到网口,再通过Internet将数据传输给监控端PC机,

收稿日期:2007-06-28

**作者简介:**王振煜(1983-),男,硕士研究生,主要从事信号与信息处理方面的研究;葛万成,教授,博导,主要从事信号与信息处理方面的研究。本文受德国罗德与施瓦茨公司资助。

由 PC 机完成图像数据的实时接收和自动显示。反之,PC 机到嵌入式单板机的数据传输同样如此<sup>[2]</sup>。

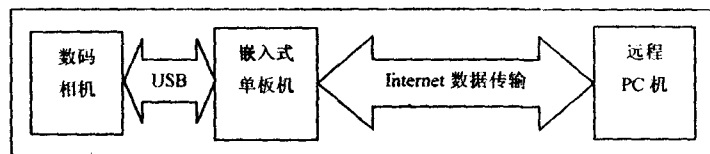


图 1 利用嵌入式单板机开发的图像传输系统框图

## 2 采集单元软件实现

### 2.1 操作系统的实现

由于 Windows XP Embedded 是一个可定制的操作系统,其包含有超过 10000 个独立的特性组件可供开发者选择以达到管理和减小定制的设备镜像大小的最优功能特性,因此可以只选择那些图像采集应用程序所必需的组件,而忽略其它的常用组件。在此简单介绍定制过程的步骤。首先使用了 Target Analyzer,这是 Windows Embedded Studio 工具包中专门为 Windows XP Embedded 设计的新特性,它会对目标硬件平台进行分析,自动检测这些系统并收集有关硬件的信息。在使用了 Target Analyzer 后,将这个工具收集到的可扩展标记语言数据导入到 Target Designer 中,Target Designer 也包含在 Windows Embedded Studio 工具包中,它可以自动构建可对目标平台提供支持的基础操作系统映像。系统定制人员可以使用 Target Designer 来选择额外的组件,可以只选择它所需要的特性,如 Win32 API、TCP/IP 堆栈、多进程和多线程、USB 端口、鼠标和键盘驱动等。在检查了配置的相关性并生成了操作系统映像后,就拥有了为安装准备就绪的操作系统映像,笔者定制的映像大小在 100MB 左右,随后可以对映像进行进一步的功能性测试。完成后,对 CF 卡进行引导处理,包括 SYS 命令和 BOOTPREP 命令的使用。然后将定制所得的操作系统映像复制到 CF 卡中,装载到嵌入式单板机上,顺利通过 First Boot Agent 后完成了操作系统的实现。

### 2.2 数码相机的 SDK 开发

目前购买的数码相机都附带了丰富的 PC 端应用程序,一些数码相机制造商还为开发人员提供了 SDK (Software Development Kit) 资源,便于有关人员利用数码相机进行图像采集和监测方面的开发工作。Olympus 公司的 SP-500UZ 数码相机提供的 SDK 函数主要如下<sup>[3]</sup>:

- (1) 初始化 SDK 资源函数: `_InitControl()`。
- (2) 相机初始化函数: `_Connect()`, `_Disconnect()`。
- (3) 相机控制函数: 有 `_GetPropCameraID()`, `_`

`Capture()`, `_SetPropOpticalZoom()`, `_SetPropFocusMode()`, `_setFlashMode()`, `_SetPropResolution()`。

(4) 图片管理函数: 主要有 `_GetPropPicCount()`, `_SetPropCurrentPicture()`, `_GetPropPicSize()`, `_GetPropIndexSize()`, `_GetPropFileName()`。

(5) 文件传递函数: 主要有 `_GetPicture()`, `_ErasePicture()`, `_GetIndex()`, `_Preview()`。

通过使用 SDK 资源编程,可以控制相机自动采集和存储图像数据,一个简单的拍摄过程如下:

```

#include "ryecommon.h"
//初始化相机 SDK 类
m_Rye._InitControl();
//获取连接的相机数
long m_nNumCameras = m_Rye._GetPropCameraCount();
//连接到相机
m_Rye._Connect(m_nCurrentCamera);
//拍摄一幅相片
m_Rye._Capture(m_nCurrentCamera);
//定位到所拍摄的图像
m_Rye._SetPropCurrentPicture(m_nCurrentCamera, m_Rye._GetPropPicCount(m_nCurrentCamera));
//文件的存储和传输操作
.....
//断开与相机的连接
m_Rye._Disconnect(m_nCurrentCamera);
  
```

### 2.3 数据传输功能的实现

数据传输部分是指将采集来的图像数据进行打包并且通过以太网发送出去。嵌入式单板机作服务器端软件运行的平台,由于采用了基于 Win32 编程模型的 XP Embedded 操作系统,可以直接使用 MFC 中的 CSocket 类开发图像数据的发送软件<sup>[4]</sup>。

图像数据的发送过程中采用 500 字节每包的方式。为了实现图像数据的连续采集和发送,将发送环节加入到连续采集控制程序的内循环中进行处理。该步骤的简单实现为在获得连接请求后,数码相机拍摄一帧图像并完成在指定文件夹中的存储后,进行图像数据的读取和发送。其程序流程如图 2 所示。

## 3 监控单元软件实现

### 3.1 图像数据的接收

监控单元软件运行在 PC 机上,使用 Visual C++ 开发的客户端程序,其功能包括创建与服务器端的连接,发送控制命令,接收和显示图像数据,其中数据的接收和显示最为关键。在 Windows 下使用 Socket 类

开发的数据接收程序,其基本结构如下:

- (1) CSocket.Create()//创建套接字函数。
- (2) CSocket.Connect("IP", Tcpport)//连接服务器,需要指定 IP 地址和端口号。
- (3) CSocket.Send(Buffer, BufferSize)//数据发送函数。
- (4) SocketSrv.Receive(Buffer, BufferSize)//接收数据函数。
- (5) SocketSrv.Close()//关闭连接。

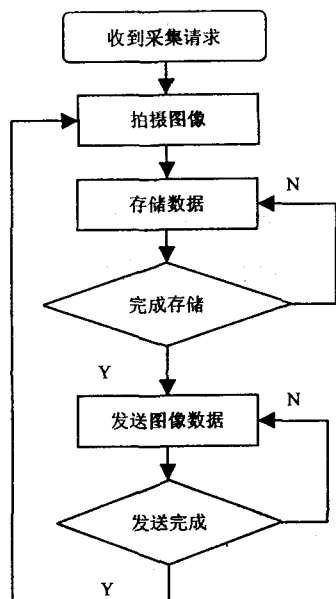


图2 发送程序流程

图像数据的接收同样采用 500 字节每包的方式,由于是连续接收,在接收过程中要做控制字符的检查<sup>[5]</sup>。JPEG 格式的图像数据是以 0xFF、0xD8 作为开始符,以 0xFF、0xD9 作为结束符的,主机应对接收的数据进行检查,当检查到当前接收到的数据包中存在结束符时则结束文件的写入,同时转入显示环节<sup>[6]</sup>。其流程如图 3 所示。

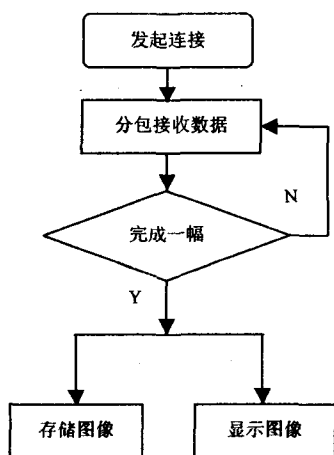


图3 接收程序流程

### 3.2 图像数据的显示

JPEG 格式图像的显示方法有很多,文中采用的是基于 CPicture 类的方式,CPicture 类具备一个 ATL 智能指针 CComQIPtr 指向 IPicture 接口,后者操纵图像对象及其属性。CPicture 类提供了常用的打包函数来调用底层的 IPicture,它允许以多种方式加载图像数据,包括从文件名加载、从 CFile 类和 CArchive 类的对象加载等,当然其实质都是调用了 OleLoadPicture 函数从包含有图像数据的流中装载图像<sup>[7]</sup>。显示程序的实现过程如下:

- (1)声明 CPicture 类的对象 MyPicture,并加载图像数据 MyPicture.Load(FileName);
- (2)通过 CWnd \* CWndPointer = (CWnd \*) GetDlgItem(int nID)来得到子窗口指针;
- (3)使用该子窗口指针初始化作为显示基础的设备内容 CDC \* pDC = CWndPointer->GetDC();
- (4)获得用户区域矩形 CWndPointer->GetClientRect(CRect &rc);
- (5)在用户窗口的矩形中显示图像 MyPicture.Render(pDC,rc)。

该显示环节嵌入在数据接收循环中,以实现连续接收和实时的显示。

## 4 结 语

提出了一套基于嵌入式单板机平台和 XP Embedded 系统实现的嵌入式高分辨率图像采集系统,在实验中该系统很好地完成了图像的采集和传输,得到的图像分辨率大,服务器可以稳定运行。由于采用了 XP Embedded 操作系统,软件开发速度得到较大提升,同时采用的嵌入式单板机具有丰富的扩展能力,可以接驳无线网卡、CDMA/GPRS 模块,在带宽满足要求的前提下,还可以采用无线网络方式传送图像数据<sup>[8]</sup>。

### 参考文献:

- [1] Reed D, Macintyre J, Berard S. Building and Deploying XP Embedded Images[DB/OL]. 2003-06. <http://msdn2.microsoft.com/en-us/library/ms838635.aspx>.
- [2] 秦 锋,袁志祥,石 磊. 基于 Web 的远程数据监控系统的设计与开发[J]. 计算机技术与发展,2006,16(5):201-207.
- [3] CAMEDIA SDK 3.3(DLL Version) Reference Manual[R]. [s.l.]:Olympus Optical Co., Ltd,2002.
- [4] 张 越. Visual C++ 网络程序设计[M]. 北京:人民邮电出版社,2006.
- [5] 刘方敏,吴永辉,俞建新. JPEG2000 图像压缩过程及原理

(下转第 172 页)

化效果,导致相位噪声总体呈恶化趋势。偏离中心频率 600kHz 处的相位噪声随着  $R_{\text{filter}}$  的增大而增大,说明此处电阻的热噪声对相位噪声的恶化占主导地位。经过仿真,确定  $R_{\text{filter}}$  取 74Ω 时比较合适。

### 3 仿真结果

该电路的仿真是在 Mentor 公司的模拟 IC 设计平台 Eldo RF 下进行,工艺库采用 Chartered 0.35μm CMOS 标准工艺。

图 4 为振荡器的起振波形,可以看出振荡器从开始起振到波形稳定大约需要 28ns 的时间。输出幅度为: -1.8V~1.6V。

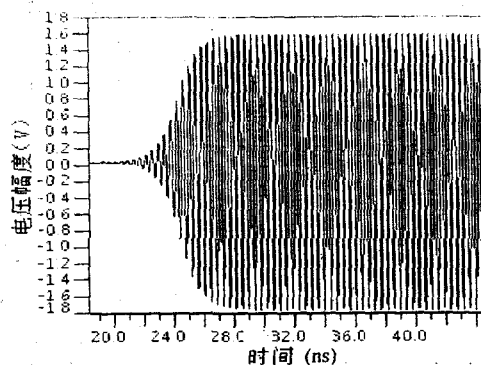


图 4 振荡器的起振波形

图 5 为振荡器的频率调谐特性曲线。在控制电压从 0.5V 变化到 3.0V 范围内,振荡器频率从 2.19GHz 变化到 2.49GHz,变化范围达到 300MHz。

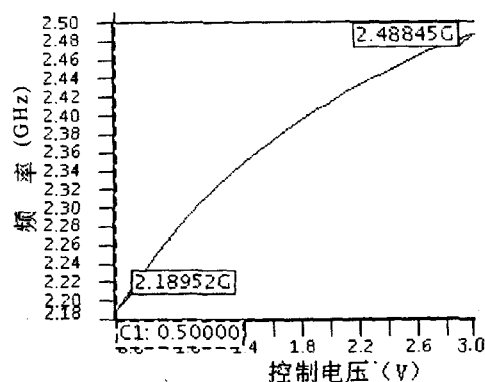


图 5 振荡器频率调谐特性曲线

图 6 是振荡器的相位噪声曲线,在偏离中心频率 600kHz 处测得的相位噪声达到 -121dBc/Hz。

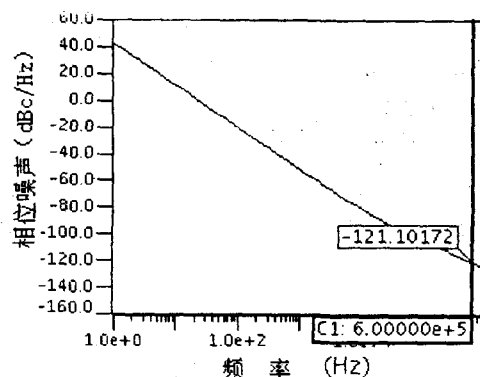


图 6 振荡器相位噪声曲线

### 4 总结

利用 Mentor 公司的模拟 IC 设计平台 Eldo RF 和 Chartered 0.35μm CMOS 标准工艺,设计了一个工作电压为 3.3V,静态电流为 12mA 的集成 CMOS 负阻 LC 压控振荡器。仿真结果表明,振荡器起振时间为 28ns,输出幅度范围为: -1.8V~1.6V,频率变化范围达到 300MHz,在偏离中心频率 600kHz 处的相位噪声为 -121dBc/Hz,验证了 RC 滤波技术对集成压控振荡器相位噪声的抑制,同时为全集成低相位噪声 CMOS 负阻 LC 压控振荡器的设计提供了一种参考电路。

### 参考文献:

- [1] Hajimiri A, Lee T H. A General Theory of Phase Noise in Electrical Oscillators[J]. IEEE Journal of Solid - State Circuits, 1998, 33(2): 179 - 194.
- [2] Lesson D B. A Simple model of feedback Oscillator noise Spectrum[J]. Proceeding of the IEEE, 1966, 54(2): 329 - 330.
- [3] Rael J J, Abidi A A. Physical processes of phase noise in differential LC oscillators[C]// IEEE Custom Integrated Circuits Conf. Orlando; [s. n.], 2000: 569 - 572.
- [4] Razavi B. A Study of Phase Noise in CMOS Oscillators[J]. IEEE Journal of Solid - State Circuits, 1996, 31(3): 331 - 343.
- [5] Hagazi E, Sjöland H, Abidi A. A Filtering Technique to low LC Oscillator Phase Noise[J]. IEEE Journal of Solid - State Circuits, 2001, 36(12): 1921 - 1930.
- [6] 拉扎维·毕查德. 模拟 CMOS 集成电路设计[M]. 陈贵灿译. 西安: 西安交通大学出版社, 2003.

(上接第 169 页)

- [5] 概述[J]. 计算机辅助设计与图形学学报, 2002, 14(10): 905 - 911.
- [6] 胡 栋. 静止图像编码的基本方法与国际标准[M]. 北京: 北京邮电大学出版社, 2003.
- [7] DiLascia P. Displaying a JPG in your MFC Application[DB/

OL]. 2001 - 10. <http://msdn.microsoft.com/msdnmag/issues/01/10/c/default.aspx>.

- [8] 常 雄, 周 旭. 基于 GPRS 的信息采集系统智能终端的设计[J]. 计算机技术与发展, 2006, 16(10): 153 - 158.