

# 一种基于以太网的计算机网络原理实验方案

陆卫忠,刘文亮,班建民,奚雪峰

(苏州科技学院 电子与信息工程系,江苏苏州 215011)

**摘要:**计算机网络是电子信息类专业的核心课程,也是一门理论性和实践性较强的课程。作者通过多年的教学实践,设计了一套基于以太网的计算机网络原理实验方案,并采用.Net(C#)技术加以实现。通过该实验方案,可使学生加深对计算机网络体系结构的理解,同时提高了基于.Net平台下的应用系统的设计能力,从而为基于计算机网络环境的系统集成和实现相关应用系统奠定良好的基础。

**关键词:**计算机网络;实验;面向对象;以太网;C#

**中图分类号:**TP393.02

**文献标志码:**A

**文章编号:**1005-2909(2007)03-0124-04

计算机网络是电子信息类专业的核心课程,也是一门理论性和实践性较强的课程。为方便学生在学习过程中尽快掌握该课程的精髓,把所学的理论知识与实际应用紧密结合,需要通过一系列课程实验才能实现这一目标。笔者经过多年的教学实践,探索出了一套理论与实践紧密结合,引导学生自己实践的有效方法。此方法立足于对数据通信过程中数据流的传输及变换过程和网络协议实现方法的描述,可使学生加深对计算机网络体系结构的理解,进一步掌握计算机网络协议实现的技术,从而为实现基于计算机网络环境的系统集成和相关应用系统奠定良好的基础。

## 一、计算机网络原理实验模型

参照 OSI 的参考模型,将网络原理实验模型分为物理层、数据链路层、网络层、运输层及应用层 5 个部分。其物理模型是借助于以太网局域网实现的,因此只要有局域网,无需外加其他设备即可进行相应项目的实验。

由于实际的各层协议非常复杂及实验课时所限,实验方案中将各层协议的功能进行了必要的简化。同时为便于理解各层协议的功能及实现的基本方法,采用面向对象的分析和设计方法为实验系统定义了物理层、数据链路层、网络层、运输层的功能和相关的类及其成员,确定实验系统相关的各层实体,并运用开发工具 Microsoft Visual Studio 2005(编程语言为 C#)实现了本系统。

## 二、协议的实现方法

### (一)网络实体类设计

实验系统中协议的实现是从抽象网络实体类 NetEntity 开始设计的,逐步扩

收稿日期:2007-05-21

基金项目:建设部研究开发项目 04-2-205,苏州科技学院教改项目

作者简介:陆卫忠(1964-),男,江苏常熟人,苏州科技学院电子与信息工程系副教授,主要从事计算机网络与嵌入式系统研究。

展到物理层、数据链路层、网络层和运输层。协议中涉及的主要对象及其关系分别是 PHLEntity(物理层实体类)、DLLEntity(数据链路层实体类)、NLEntity(网络层实体类)和 TLEntity(运输层实体类)。在运输层之上实现了文件传输功能,相应的实体对象是 FTPClient 和 FTPServer。通过上下层实体间通信消息类(NetMessage,并由它派生各层的服务访问点 SAP)传递消息。

对等层的通信过程采用协议状态图描述,根据状态图设计相应的实体类成员函数。

## (二)通信的实现方法

网络中相邻层间相互通信的实现方法有两种。一种称为进程/协议模型,每个协议由一个独立的进程实现,当一条消息向协议栈的上方或下方移动时,它被从一个进程/协议传送到另一个进程/协议。另一种称为进程/消息模型,它把每一个协议当作是一段静态编码并把它同消息联系,当网络送来一条消息时,操作系统调度一个进程,使之负责在协议图中向上移动。在每一层,调用实现该协议的过程,然后调用实现下一个协议的过程,以此类推。本方案中协议的实现就是采用这种方法,通过在进程间传送消息来实现数据的传送。各层之间通信传递的消息是由 NetMsg 类派生的对象,含有由一个上行消息队列和一个下行消息队列组成的 SAP。

## 三、网络原理实验内容

### (一)物理层实验

本实验中,物理层是借助现有以太网技术,即以以太网作为隧道,来传输本实验系统的数据。

#### 1. 虚拟网卡

实现物理层实体通信的虚拟网卡是利用 Windows 平台下免费的 WinPCap(Windows Packet Capture)软件来实现的,它为 Win32 应用程序提供了访问网络底层的能力。同时设计一个封装了 WinPCap API 函数的动态链接库文件(TunDev.DLL)供本系统使用,实现通过以太网卡进行数据的发送和接收。

#### 2. 物理层类设计

(1) 抽象网络实体类 NetEntity。NetEntity 类需要用到的动态连接库:TunDev.DLL,NetEntity 类实现了数据的接收与发送:

```
class NetEntity
{
    public int VirtualNetCard_Open( string GWAd-
```

```
dr);
//打开设备,等待接收数据,其中 GWAddr 为
本地 IP 地址(例如:192.168.1.50);
public uint VirtualNetCard_Read(int fd, byte[]
ip_buf, int ip_len);
//从网卡上读取数据,返回读取到的数据的总
数,其中 fd 为打开设备时返回的句柄;
public void VirtualNetCard_Write( int fd, byte[]
ip_buf, int ip_len);
//将数据发送到网卡上;
public void VirtualNetCard_Close( int fd); //关
闭设备;
}
```

(2) 服务访问点及消息类的设计。服务访问点(SAP)的类名是 NetSap,它由一个上行队列(UPQueue)和一个下行队列(DOWNQueue)组成。由于 C#中已经实现了队列(Queue),因此只需要进行如下定义:

```
public class NetSap
{
    public Queue UPQueue = new Queue(); //上
    行队列
    public Queue DOWNQueue = new Queue();
    //下行队列
}
```

上下层间通信的消息类 NetMsg 可定义如下:

```
public class NetMsg
{
    public MsgTyp msgType; //消息类型
    public NetSap Sap = new NetSap(); //服务访
    问点
}
```

(3) 物理层实体类——PHLEntity。物理层的实体类通过继承 NetEntity 类的功能,实现数据的发送与接收。定义如下:

```
class PHLEntity : NetEntity
{
    static int fd = -1; //打开设备时返回的句柄
    public NetMsg PhyMsg = null;
    public PHLEntity(); //构造函数
    public int Open( string GWAddr); //打开设备,
    GWAddr 为网卡 IP 地址
```

```

public void Send(byte buffer[],int length);//
将数据发送到网卡上
public uint Receive (byte[] buffer); //接收网
卡上的数据
public void Close(); //关闭设备
public byte Error(byte); //物理层模拟错误
}

```

## (二) 数据链路层实验

该实验是在已经实现的物理层基础上完成:(1)设计简易可行的数据链路层帧格式以及差错校验方案。(2)设计并实现“停止-等待”协议。(3)设计并实现“连续重传 ARQ”的滑动窗口协议。数据链路层上帧的总长不超过 1470 字节,定义在 Frame 类中,格式如表 1。

表 1 数据链路层格式

帧类型	源地址	目标地址	帧长	帧的序号	帧的数据部分	帧校验和
(1Byte)	(2Byte)	(2Byte)	(2Byte)	(1Byte)	(最大 1460Byte)	(2Byte)

其中:目标地址 0xffff 时为广播地址,帧校验和可以根据需要设计校验方法及范围。

数据链路层的实体类主要成员定义如下:

```

class DLLEntity : PHLEntity
{
PHLEntity PHL = null;
public NetMsg DLLMsg = null;
Frame Dataframe = null;
public DLLEntity(); //构造函数
public void Send(Frame frm); //向物理层发送

```

表 2 网络层格式

源地址	目标地址	标识	分片标志	偏移量	数据长度	校验和	数据部分
(4Byte)	(4Byte)	(1Byte)	(1Byte)	(4Byte)	(2Byte)	(2Byte)	(最大 1442Byte)

网络层的实体类主要成员定义如下:

```

class NLEntity : DLLEntity
{
DLEntity DLL = null;
public NetMsg NLMsg = null;
public NLEntity(); //构造函数
public void Send(Packet packet); //发送分组
public void Receive(Packet pkt); //接收分组
public bool FindRoute(byte[] destaddr, byte[]
route) //查找路由表
public void MakeFragment(byte[] Message) //
分片

```

帧

```

public void Receive(Frame frm); //从物理层接
收帧
public ushort makeChecksum(Frame frm); //根
据校验方法生成校验码
public void makeACKFrame(Frame frm); //生成
ACK 帧
public void makeDataFrame(Frame frm); //生成
DATA 帧
}

```

数据链路层协议采用滑动窗口协议。用 LAR、LFS、LFR 分别代表发送方最近收到的 ACK 序号、最近发送的帧和接收方期待的下一帧的序号。发送窗口的大小为 SWS,接收方窗口的大小为 1。发送的帧暂存在缓冲区 SendBuffer 中,通过在主程序中的超时定时器的定时事件,发送暂存的帧。通过改变接收窗口的大小,实现连续重传 ARQ 协议。

## (三) 网络层实验

该实验建立在已经实现的数据链路层基础上,实现:(1)设计简易实用的分组数据包格式。(2)设计并实现类“IP”的网络层协议,路由选择采用静态路选择协议。(3)设计并实现分组的分片与重装。

网络层的分组设计一方面要考虑到路由选择的实现,即分组中应含有路由寻址所必要的信息,另一方面要考虑到分组太大时的分片与重装。设计的网络层上的数据报结构定义在 Packet 类中,总长度为 1460 字节,格式如表 2。

```

public void AssemblePkt(Packet pkt) //分片重
装
}

```

本实验采用静态路由选择算法,在每个具有路由功能的主机上,需配 2 块以上的以太网卡,通过配置文件 route\_tab.cfg 确定路由选择结果,该文件包含目的“主机”地址和转发“主机”(网卡)地址。

## (四) 运输层实验

该实验建立在已经实现的网络层基础上,实现面向连接的通信和为应用进程提供可靠服务。网络层的报文设计主要考虑到与应用进程的通信及提供面向连接的服务,格式如表 3。

表3 运输层格式

源端口 (1Byte)	目的端口 (1Byte)	序号 (4Byte)	确认号 (4Byte)	控制 (1Byte)	长度 (1Byte)	校验和 (2Byte)	数据部分 (长度不定)
----------------	-----------------	---------------	----------------	---------------	---------------	----------------	----------------

运输层的实体类继承了网络层实体类,实现连接建立、通信、连接拆除等功能,类及主要成员定义如下:

```
class TLEntity : NLEntity
{
    NLEntity NL = null;
    public NetMsg TLMsg = null;
    public byte Port;
    public TLEntity (); //构造函数
    public void Send(Message msg); //发送报文
    public void Receive(Message msg); //接收报文
    public byte Connect(byte[] destaddr, byte destport); //连接建立
    public bool Disconnect(byte); //拆除指定连接
}
```

#### (五) 应用层实验

该层实验是设计并实现一个可完成类似 FTP 功能的程序。其目的是了解应用层协议使用下层协议提供服务的方法,掌握客户/服务器模式的应用程序设计方法。

本实验中要实现简单的 FTP 功能,即具有与 FTP 服务器连接/断开、目录浏览和文件下载功能,可以处理的命令有:CD、DIR(LS)、GET、QUIT 等。

实验的 FTP 需要设计两个类,即 FTPClient 客户类和 FTPServer 服务器类,它们分别在文件“FTP-Class.cs”中定义,可以处理的命令在相应类中实现。

#### 四、结语

通过上述实验方案的实施,一方面对学习并掌握计算机网络原理起到了促进作用,加深了对网络体系结构和所依据的理论理解,另一方面也使学生在学习过程提高了应用程序的设计能力。教学过程中可以根据需要对上述实验方案的内容进行扩充,以便充分调动学生学习积极性和激发其创造性,主动参与到实验的每个环节:从方案设计到协议实现。

#### 参考文献:

- [1] 谢希仁. 计算机网络[M]. 大连:大连理工大学出版社, 2000.
- [2] Andrew S. Tanenbaum. 计算机网络(第三版)[M]. 北京:清华大学出版社, 1997.
- [3] 陆卫忠. 计算机网络实验教程[M]. 国防工业出版社, 2004.
- [4] 李名世. 计算机网络实验教程[M]. 北京:机械工业出版社, 2003.
- [5] Douglas E. Comer David L. Stevens. 用 TCP/IP 进行网际互联[M]. 北京:电子工业出版社, 2004.

## An Experimental Project of Computer Networks Principle Based on Ethernet

LU Wei-zhong, LIU Wen-liang, BAN Jian-min, XI Xue-feng

(Department of Electronic and Information Engineering, University of Science and Technology of Suzhou, Suzhou, 215011, China)

**Abstract:** Computer Networks is a core course of Electronic Information majors. It focuses on both the theory and the practicality. Through long-period teaching, the authors design an experimental project of computer networks principle based on Ethernet using C#. Students' conception of Computer Networks architecture may be intensified and their ability of designing system on .Net platform can be enhanced through the project. Thereby the foundation of system integration and related application implementation based on computer networks will be built successfully.

**Key words:** computer networks; experiment; object-oriented; ethernet; C#

(编辑 彭建国)