第3章 链路层协议分析

在 TCP/IP 协议族中，链路层也叫网络接口层，包含着 OSI/RM 模型的数据链路层和物理层。数据帧在这里转换成在网络传输媒体上传送的比特流，或将从传输媒体上接收的比特流组装成数据帧。

本章将着重介绍链路层最常用的以太网协议，并详细比较 DIX Ethernet V2 和 IEEE 802.3 封装的异同，对 SLIP 和 PPP 只作简单介绍，对大多数实现都包含的环回(loopback)接口驱动程序也作了介绍。实验部分要求掌握分析链路层帧的基本方法，同时熟悉 Packet Tracer 和 Wireshark 的用法，进一步掌握协议分析学习工具的功能特点和用途。

3.1 链路层的作用

为了更清楚地理解链路层的作用，需要再回顾一下 TCP/IP 协议的基本层次关系。TCP/IP 协议的层次结构如图 3-1 所示。

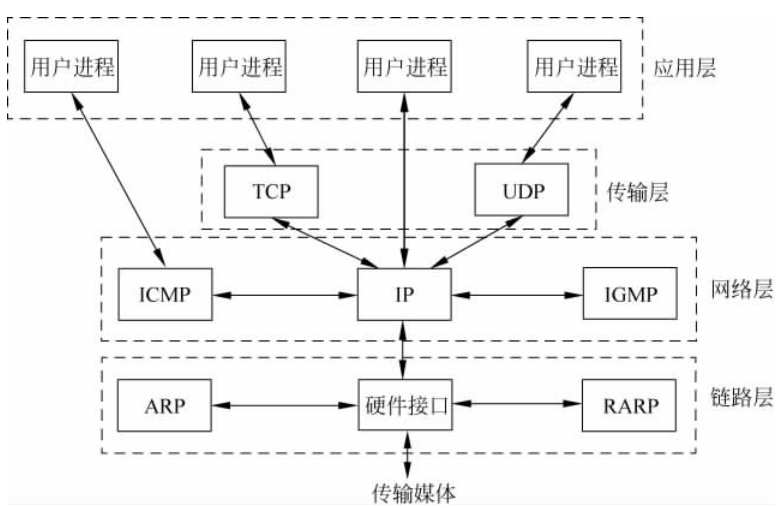

图 3-1 TCP/IP 协议的层次结构

在图 3-1 中，“硬件接口”即对应着链路层的主体。从图中可以看出，链路层主要有 3 个目的。

(1) 为 IP 模块发送和接收 IP 数据报。

(2) 为 ARP 模块发送 ARP 请求和接收 ARP 应答。

(3) 为 RARP 模块发送 RARP 请求和接收 RARP 应答。

在这里可以十分明确：链路层在各层协议中要直接打交道的就是 IP、ARP 和 RARP 3 个协议。结合第 1 章讲过的协议工作原理中封装和分用的过程，链路层帧中封装的数据体现为 IP、ARP 和 RARP 这 3 种类型的协议数据。

网络在链路层所使用的硬件不同，则会采用不同的链路层协议，如以太网、令牌环网、FDDI(Fiber Distributed Data Interface，光纤分布式数据接口)及 RS-232 串行线路等。目前 TCP/IP 能够支持多种不同的链路层协议。

在网络技术中，局域网占有非常重要的地位。按照网络拓扑结构，局域网可以分为星形、环形、总线型和树型网络，代表性的网络主要是以太网、令牌环网和令牌总线网。经过数十年的发展，特别是近年来千兆以太网和万兆以太网的飞速发展，采用 CSMA/CD(Carrier Sense Multiple Access/Collision Detection，载波侦听多路访问/冲突检测)接入方法的以太网已经在局域网市场中占有绝对优势。以太网几乎成为局域网的同义词，因此本章将以以太网作为主要的学习内容，然后介绍 SLIP(Serial Line Internet Protocol，串行线路网际协议)和 PPP(Point to Point Protocol，点对点协议)，对现今已经基本淘汰而较少使用的技术，如令牌总线网等就不再涉及。

3.2 以太网的帧结构

目前使用最为广泛的链路层协议有以太网、串行接口链路等。

3.2.1 以太网的两种主要标准

以太网是 1982 年由美国 DEC、Intel 和 Xerox 3 家公司联合制定的局域网技术标准，目前采用的是 Ethernet V2 标准，因此也称为 DIX Ethernet Ⅱ格式。在 TCP/IP 标准中，由 RFC 894(Hornig，1984)来说明，是目前最常用的局域网标准。

IEEE 802 是一个标准集，是由 IEEE(Institute of Electrical and Electronics Engineers，电气和电子工程师学会)在以太网推出后不久公布的一个局域网标准。IEEE 802 将数据链路层分为两个子层，即 LLC(Logical Link Control，逻辑链路控制层)和 MAC(Media Access Control，介质访问控制层)。IEEE 802.2 规定了 LLC 的有关内容，而 IEEE 802.3 针对整个 CSMA/CD 网络对 MAC 有具体的规定。IEEE 802 的 MAC 子层用于规定网络传输介质或网络媒体的访问，LLC 子层则用于管理两个 MAC 层地址之间的点到点的数据传输。IEEE 802.4 和 802.5 都是令牌网络有关的标准，现已较少使用。

3.2.2 以太网帧的封装结构

现在采用的以太网主要有两种不同规格的标准，分别由 RFC 894(Ethernet Ⅱ)、RFC 1042 (IEEE 802 网络)规定了两种不同形式的封装格式，如图 3-2 所示。图中帧格式下的数字表示对应字段的字节数。

从图 3-2 可以看到，两种帧格式都采用 48 位(6 字节)的目的地址和源地址，这就是硬件地址(MAC 地址)。接下来的 2 个字节在 IEEE 802 中是长度字段，是指它后续数据的字节

长度，但不包括 CRC 检验码；Ethernet Ⅱ此处是类型字段，定义了后续数据的类型。（请思考：系统是如何区分收到的帧该位置的 2 个字节是表示长度还是类型的呢？）

在 IEEE 802 帧格式中，跟随在长度后面的是 3 个字节的 802.2 LLC 结构。其中，LLC 由 DSAP（Destination Service Access Point，目的服务访问点）和 SSAP（Source Service Access Point，源服务访问点）及 Cntl 组成。DSAP 和 SSAP 通常取值相同，用于说明通信两端采用的链路层协议。如果其中封装的是 802.2 SNAP（Subnetwork Access Protocol，子网访问协议）的协议数据，则 DSAP 和 SSAP 的值都设为 0xAA（IEEE 对 DSAP 和 SSAP 的取值有专门的规定，需要时可以查阅相关资料），Cntl 字段的值设为 3。随后有 5 个字节的 SNAP 结构，包含的前 3 个字节为 org code，都置为 0。再接下来的 2 个字节的类型字段和以太网帧格式的一样。

在图 3-2 中标示出了链路层中封装的主要 3 种协议的类型标识的取值。0x0800 表示帧承载的是 IP 报文，0x0806 表示帧承载的是 ARP 报文，而 0x8035 表示帧承载的是 RARP 报文。RFC 5342 对以太网帧格式中的“类型”字段的更多取值有相应的规定，需要时可以查阅。

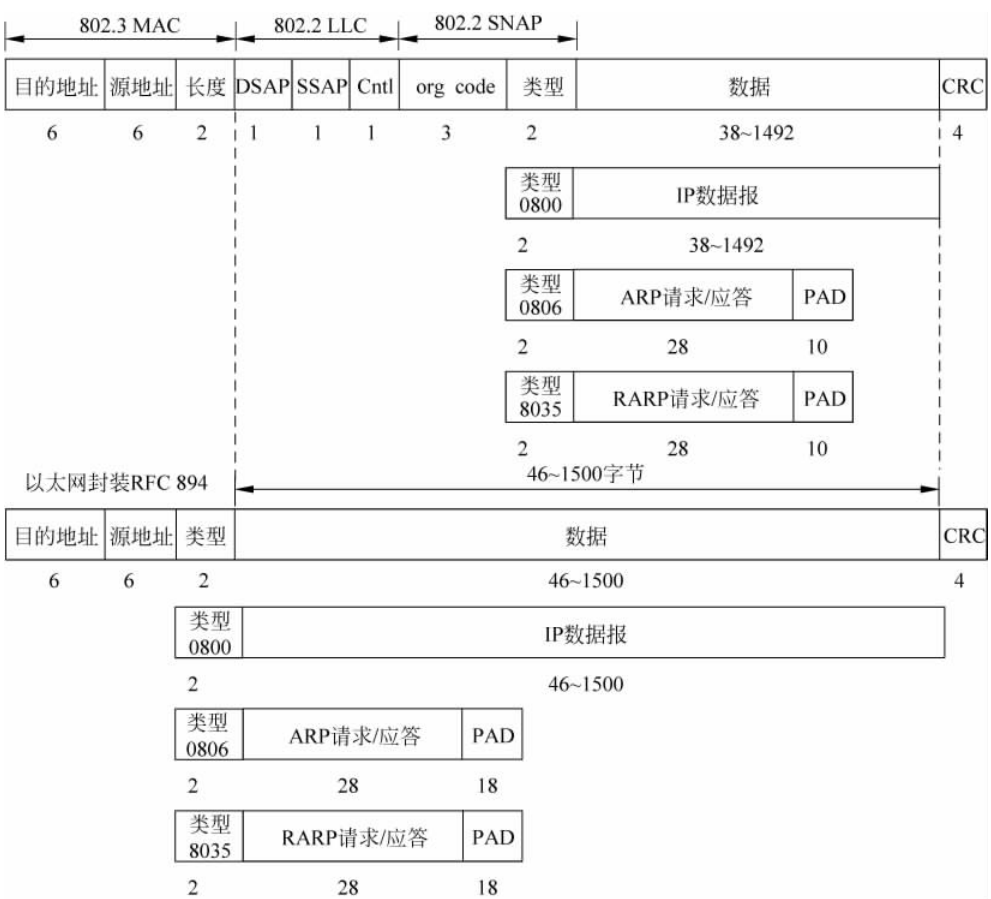

图 3-2　IEEE 802.2/802.3（RFC1042）和以太网（RFC 894）的封装格式

大多数应用程序的以太网数据包都采用 Ethernet Ⅱ格式的帧来封装(如 HTTP、Telnet、FTP、SMTP、POP3 等应用),执行 STP(Spanning Tree Protocol,生成树协议)的交换机之间的 BPDU(Bridge Protocol Data Unit,网桥协议数据单元)采用 IEEE 802.3 SAP 帧(即 802.3 MAC 和 802.2 LLC),VLAN Trunk 协议 802.1Q 和 CDP(Cisco Discovery Protocol,Cisco 发现协议)采用 IEEE 802.3 SNAP 帧。

CRC 字段用于帧内字节差错的循环冗余码检验,它也被称为 FCS(Frame Check Sequence,帧检验序列)。

IEEE 802 标准定义的帧和 Ethernet Ⅱ的帧都有最小和最大长度要求。IEEE 802 标准规定帧的数据部分最少要有 38 字节,以太网则规定最少为 46 字节。如果不足最小长度,则协议要求用插入填充(pad)字节的方式来补足。最大长度要求就是通常所说的 MTU(Maximum Transmission Unit,最大传输单元),IEEE 802 和 Ethernet Ⅱ分别是 1492 和 1500 字节。

在传输媒体上实际传送的比特流中还要在如图 3-2 所示的帧序列前多出 8 字节的前导字节(7 个字节的前同步码和 1 个字节的起始帧定界符),用作帧收发的同步控制。这里没有标注出来是因为只有链路层硬件接口(如网卡)正确地从网络链路上接收到能够识别处理的比特流数据且没有差错并组装成帧后,才会由链路层协议栈来处理。或者说,不能够识别的或错误的比特流都丢弃了。因而,在各种协议分析器捕获的数据中都不会看到帧前导字节,甚至是校验字节。Cisco Packet Tracer 模拟方式显示的帧有时会给出前导字节。

3.3 串行接口的链路层协议

在串行线路上对 IP 数据报进行封装的常见形式有 SLIP 和 PPP。当然,这两个协议不只用于数据通信网络中,也可以和许多其他的串行通信协议一样用于工业控制、家用电器等微型或小型系统间的数据传输,目前在嵌入式系统中也有应用。

3.3.1 SLIP

SLIP 是一种在串行线路上对 IP 数据报进行封装的简单形式,在 RFC 1055 中有详细的描述。SLIP 适合具有最常见的 RS-232 串行口的计算机系统或高速调制解调器接入 IP 网络使用。

SLIP 帧的格式如图 3-3 所示。

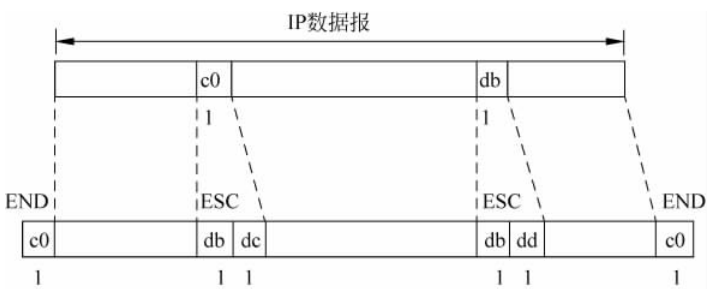

图 3-3 SLIP 报文的封装结构

IP 数据报以一个称为 END(0xc0)的特殊字符结束,同时为了防止数据报到来之前的线路噪声被当成数据报内容,大多数实现在数据报的开始处也会传一个 END 字符。

如果 IP 报文中某个字符为 END,那么就要连续传输 2 个字节的 0xdb 和 0xdc 来取代这个 END。0xdb 这个特殊字符被称为 SLIP 的 ESC 字符(转义字符)。

如果 IP 报文中某个字符为 SLIP 的 ESC 字符,那么就要连续传输 2 个字节的 0xdb 和 0xdd 来取代它。这个方式其实就是一种字符填充的方式,在串行线路上传输的总字节数会增加。

SLIP 是一种简单的封装方法,虽然简便,但有以下缺陷。

(1) 每一端必须知道对方的 IP 地址,否则不能通信。

(2) 没有办法把本地 IP 地址通知给另一端,可以看到帧中没有专门的地址字段。

(3) 没有在数据帧中加入校验和,如果 SLIP 传输的报文受线路噪声影响而发生错误,则只能通过上层协议来发现,这样上层协议必须提供某种形式的校验。

目前 SLIP 已经被 PPP 所取代,因为 PPP 有许多更好的特点,并且不需要在连接建立前进行 IP 地址的配置。但由于 SLIP 有非常小的包装头,因此在微控制器中它仍是首选的封装 IP 包的方式。

3.3.2 PPP

PPP 是支持点到点连接的一种通信协议,既支持数据为 8 位和无奇偶校验的异步模式,也支持面向比特位的同步连接,提供对从局域网到广域网的数据链路封装支持。

RFC 1661 给出了 PPP 的详细规范,主要包括以下内容。

(1) 支持同一链路上同时使用多种协议的封装方法。事实上,PPP 支持各种主要网络协议的封装,包括 IP、NetBEUI、AppleTalk、IPX、SNA 以及其他更多的协议。

(2) 采用一个特殊的 LCP(Link Control Protocol,链路控制协议)来建立、配置、测试乃至终止链路,协商任何点到点链路的特性。

(3) 针对封装的不同网络协议,采用 NCP(Network Control Protocol,网络控制协议)来完成点对点通信设备之间网络层通信所需参数的配置,它通过协议域来区分数据域中净载荷的数据类型。RFC 1332 和 RFC 1877 描述了一个用于 IP 的 NCP,称为 IP 控制协议,它用于协商发送方的 IP 地址、DNS 服务器的地址以及在可能情况下使用的压缩协议。

PPP 的封装和组帧技术基于 ISO 的 HDLC(High-level Data Link Control,高级数据链路控制)协议,因此数据帧封装格式非常类似于 HDLC。PPP 帧结构如图 3-4 所示。

每个 PPP 数据包的开始和结束都有一个 0x7E 的数据标志。在开始标志后,紧跟两个 HDLC 常量:地址常量 0xFF 和控制常量 0x03。

接下来是协议字段,长度通常为 2 字节,表示信息字段中包含的是哪种协议以及它的处理信息。正如图 3-4 中标示的一样,0x0021 表示信息字段是一个 IP 数据报,0xC021 表示信息字段是 LCP 的内容,0x8021 则表示信息字段是 NCP 的内容。

信息字段的长度最多为 1500 字节。

然后是一个长度为 2 个字节的循环冗余检验码,以检测数据帧中的错误。

由于标志字符的值是 0x7E,因此当该字符出现在信息字段中时,类似于 SLIP 的字符填充,PPP 也需要对它进行转义。具体实现过程如下。

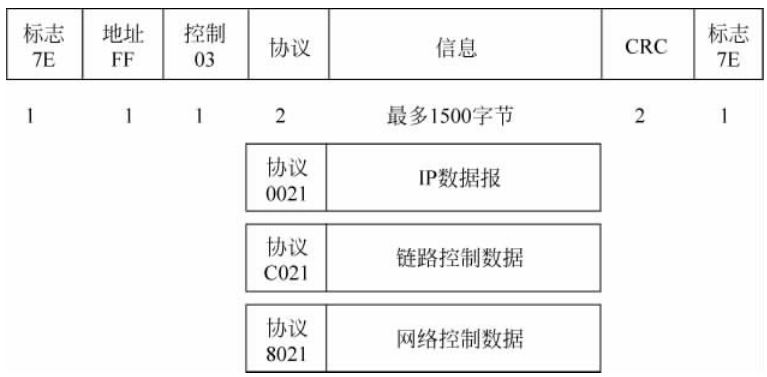

图 3-4　PPP 数据帧结构

(1) 当遇到字符 0x7E 时,需连续传送两个字符: 0x7D 和 0x5E,以实现标志字符的转义。

(2) 当遇到转义字符 0x7D 时,需连续传送两个字符: 0x7D 和 0x5D,以实现转义字符的转义。

(3) 默认情况下,如果字符的值小于 0x20(如 ASCII 控制字符),一般都要进行转义。例如,遇到字符 0x01 时需连续传送 0x7D 和 0x21 字符(这时第 6 个比特位取补码后变为 1,而前面两种情况均把它变为 0)。这样做是防止它们出现在双方主机的串行接口驱动程序或调制解调器中,因为它们有时会把这些控制字符解释成特殊的含义。另一种可能是用 LCP 来指定是否需要对这 32 个字符中的某些值进行转义。默认情况下是对所有的 32 个字符都进行转义。

当 PPP 用于同步通信时,如综合业务数字网(ISDN)、同步光纤网(SONET)的链路,则使用了一种更快速、更有效的比特填充技术,而不是字符填充。这时任何连续的 6 个 1 序列(考查用作标志的 0x7E)都可以通过在 5 个 1 之后插入 1 个 0 来进行转义。该方法支持这种链路类型中潜在非法值的更有效编码,因而使 PPP 成为 TCP/IP 中最流行的点到点协议。PPP 还支持多链路实现,即将多个相同宽度的数据通道合并。

PPP 在工作时为建立点对点链路上的通信连接,发送端首先发送 LCP 帧,以配置和测试数据链路。在 LCP 建立好数据链路并协调好所选设备后,发送端发送 NCP 帧,以选择和配置一个或多个网络层协议。当所选的网络层协议配置好后,便可以将各网络层协议的数据包发送到数据链路上。配置好的链路将一直处于通信状态,直到 LCP 帧或 NCP 帧明确提示关闭链路,或有其他的外部事件发生。在链路建立和数据传输的过程中,信息字段的内容还可以分出代码(code)、标识符(ID)和长度(length)等字段,以满足不同协议的工作要求。这里不再深入阐述,更具体的内容请参考有关资料。

要指出的是,在点到点链路上因为只有两方参与通信,并不需要寻址。PPP 提供了一种管理两点间会话的有效方法,同时不同于广域网上使用的 X.25、frame relay(帧中继)等数据链路层协议,PPP 提供了两种可选的身份认证方法: PAP(Password Authentication Protocol,口令验证协议)和 CHAP(Challenge Handshake Authentication Protocol,挑战握手验证协议),从而更好地保证了网络通信的安全性。

总的来看,PPP 相比 SLIP 具有显著的优点: PPP 支持在单根串行线路上运行多种协

议，而不只是 IP；每一帧都有循环冗余校验；通信双方可以进行 IP 地址的动态协商；LCP 可以对多个数据链路选项进行设置；提供安全支持。同时，PPP 仍然保持了成本低、传输稳定等特点。

3.4 MTU

链路层数据帧的最大长度就是 MTU。注意，MTU 是指帧的净载荷部分，不包括帧的头部、尾部及控制用字段。

前面已经指出以太网和 IEEE 802.3 的数据帧的长度限制，其 MTU 分别是 1500 和 1492 字节，如图 3-2 所示。

不同类型的网络数据帧的长度都有一个上限。如果 IP 层有一个数据包要传送，而且 IP PDU 的长度比链路层的 MTU 要大，那么 IP 层就需要进行分片，即把数据报分成若干片，使得每一片都小于 MTU，这样才能通过链路层来封装传送。IP 分片的过程将在以后的章节中讨论。

表 3-1 所示列出了一些典型的 MTU 值，表的内容来自于 RFC 1191。其中，“点到点（低时延）”是指 SLIP 和 PPP 在低时延情况下的逻辑链路限制，这时减少每一帧的字节数可以降低应用程序的交互时延，从而为交互应用提供足够快的响应时间。

表 3-1 几种常见的 MTU

网 络 类 型	MTU 字节
超通道	65 535
4Mb/s 令牌环(IEEE 802.5)	4464
FDDI	4352
以太网	1500
IEEE 802.3/802.2	1492
X.25	576
点对点(低时延)	296

在 RFC 1055 中，SLIP 的 MTU 是 1006 个字节。在 Windows 2000 的实现中，SLIP 的 MTU 设置为 1500 个字节，以满足和以太网的互联。

目前 PPP 默认的 MTU 是 1500 个字节，这个长度对于基于以太网的互联十分理想。其实，通过 LCP 在对等实体之间协商，PPP 可以在通信中使用更大或更小的 MTU，依据是它们所连接的网络的类型不同。此时 PPP 能够处理更大的帧，如 9216 字节。

可以用 netstat 命令查看并打印出网络接口的 MTU。

和 MTU 直接相关的另一个重要概念是路径 MTU。如果两台主机之间的通信要通过多个网络，那么每个网络的链路层就可能有不同的 MTU。这时重要的并不是两台主机各自所在网络的 MTU，而是连接两台主机的所有网络中的最小 MTU，称为路径 MTU，即两台主机的通信路径上最小的 MTU。这个数值直接影响着在整个通信过程中数据包是否需要分片。

路径 MTU 不一定是常数，它取决于通信时选择的路由。由于路径的选择不一定是对称的，因此路径 MTU 在通信的两个方向上不一定是一致的。

RFC 1191 描述了路径 MTU 的发现机制，即确定路径 MTU 的方法。在后面的章节中将采用这种发现方法来完成确定路径 MTU 的实验。

3.5 环回接口

环回接口(loopback interface)是一种特殊的逻辑网络接口。

绝大多数产品都支持这种形式的逻辑接口，以允许运行在同一台主机上的客户程序和服务器程序通过 TCP/IP 通信。A 类网络号 127 就是为环回接口预留的。根据惯例，大多数系统把 IP 地址 127.0.0.1 分配给这个接口，并命名为 localhost，这个地址也称为回送地址。回送地址主要用于网络软件测试及本机进程间通信。例如，“ping 127.0.0.1”用来测试本机中的 TCP/IP 协议是否正常工作。

其另一个作用是某些 C/S 模式的应用程序在运行时需调用服务器上的资源，一般要指定服务器的 IP 地址，但当该程序要在同一台机器上运行而没有别的服务器时，就可以把服务器的资源装在本机，服务器的 IP 地址设为 127.0.0.1，同样也可以运行。

对于大多数习惯用 localhost 来指代服务器的表示来说，实质上就是指向 127.0.0.1 这个本地 IP 地址。在 Windows 系统中，它成了 127.0.0.1 的别名。对于网站建设者，经常用 localhost 指向一个表示自己的特殊 DNS 主机名。

环回接口对路由器来讲是一个逻辑的虚拟接口，方便用于测试目的，因为该接口总是开启的，可作为一台路由器的管理地址，作为动态路由协议 OSPF、BGP 的 Router Id。

图 3-5 所示是环回接口处理 IP 数据报的简单过程。从中可以看到一个传给环回接口的 IP 数据报不能在任何网络上出现。无论什么程序，一旦使用环回地址发送数据，环回驱动程序会立即把数据返回给协议栈中的 IP 输入函数，而不进行任何网络传输。

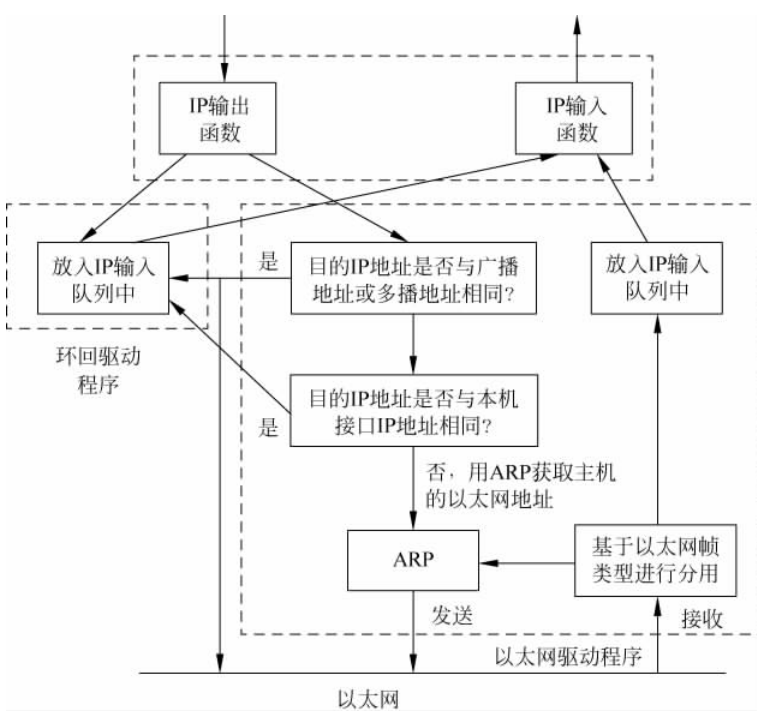

图 3-5 环回接口处理 IP 数据报的过程

图 3-5 中需要指出的要点如下。

(1) 传给环回地址的任何数据均作为 IP 输入。

(2) 传给广播地址或多播地址的数据报复制一份传给环回接口,然后送到以太网上。这是因为广播传送和多播传送的对象也包括主机本身。

(3) 任何传给该主机 IP 地址的数据均送到环回接口。

在图 3-5 中,另一个隐含的意思是送给主机本身 IP 地址的 IP 数据报一般不出现在相应的网络上。因为借助环回地址,主机保证了处理发送给自己的 IP 数据报。

3.6 小结

(1) TCP/IP 协议族中和链路层协议有直接联系的协议有 3 个,即 IP、ARP 和 RARP,这体现在链路层的功能描述中。

(2) 目前链路层使用最多的局域网协议是以太网,包括 Ethernet V2 和 IEEE 802 两种。Ethernet V2 标准规定的链路层帧结构和 IEEE 802 标准规定的链路层帧结构既有相同的地方,也有不同之处。相同之处主要是 MAC 地址形式,不同之处主要体现在 IEEE 802.2 对 LLC 的规定中。

(3) SLIP 和 PPP 协议是串行链路中的重要协议,其帧结构为适应串行通信有特别的设计,如帧内容的字节填充方式。

(4) 不同类型的链路层对 MTU 有不同的规定,Ethernet V2 标准规定的 MTU 是 1500 字节,IEEE 802 是 1492 字节。

(5) 大多数实现都提供环回接口,传送给环回接口的数据不会出现在网络上。访问这个接口可以通过特殊的环回地址。

3.7 习题

1. 除了图 3-2 所示给出的 3 种以太网帧类型,是否还有其他帧类型? 从什么地方可以查阅到以太网帧格式中的“类型”字段是怎样分配的?

2. 如果读者的主机是通过 ADSL 的 PPPoE 拨号上网的,请尝试在系统上网时捕获拨号连接通信的 PPPoE 帧并进行分析。

3. 如果读者的主机系统有 netstat 命令,如何用它来确定系统上的接口及其 MTU?

4. 主机中的环回地址通常为 127.0.0.1,能够采用其他地址来表示环回地址么?

实验

实验 3-1 DIX Ethernet V2 帧格式分析

1. 实验说明

分别通过在 Packet Tracer 和 Wireshark 中查看分析链路层的以太网帧,进一步学习捕

获查看网络通信信息的方法。同时通过对 DIX Ethernet V2 帧的分析，进一步巩固对链路层帧结构的理解和掌握。

2. 实验环境

Windows 操作系统及联网环境（主机有以太网网卡并连接局域网或 Internet），安装有 Packet Tracer 6.0 和 Wireshark 1.10。

3. 实验步骤

(1) 在 Wireshark 中捕获分析以太网帧。

步骤 1　启动 Wireshark。

在 Windows 中启动 Wireshark，选定本地网络接口并启动抓包。

步骤 2　构造网络通信信息。

启动浏览器浏览网页或运行网络应用程序（如运行 QQ 或 ping 命令等），在本机网络接口中产生网络通信信息。

步骤 3　分析捕获的帧。

在 Wireshark 中查看捕获的以太网帧，并结合 3.2 节的内容分析以太网帧结构。图 3-6 所示为在 Windows 主机浏览器中访问 Internet 网站时捕获的以太网帧示例，实验时请依据实际捕获的帧进行分析。注意，分析帧中的每一个域以及取值，观察帧的长度和 MAC 地址的构成，如 48 位 MAC 地址中 LG 和 IG 比特的含义、Type 字段的取值。

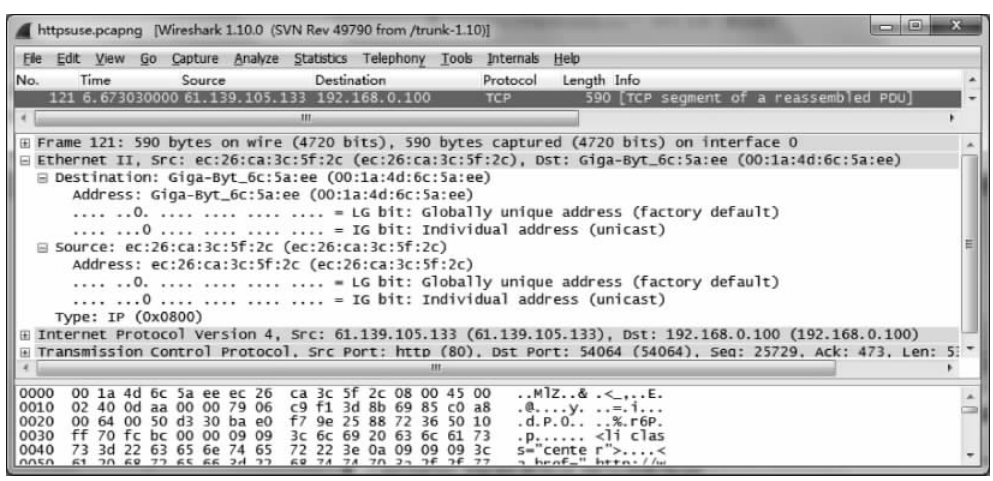

图 3-6　以太网帧结构分析示例

步骤 4　任意访问一个网站，在捕获的输入帧中查看有没有帧长度刚好为 60 字节的帧，能否看到 Padding 字段，分析其成因。

如果没有，则可以反复多次捕获帧以获取，也可以用 ping -l 限定帧长来构造短帧。有时捕获的帧中会有小于最小帧长的帧出现，这是为什么呢？

图 3-7 所示是在连通测试时捕获到有 Padding 字段及帧长为 54 字节的数据包示例，请分析其成因。

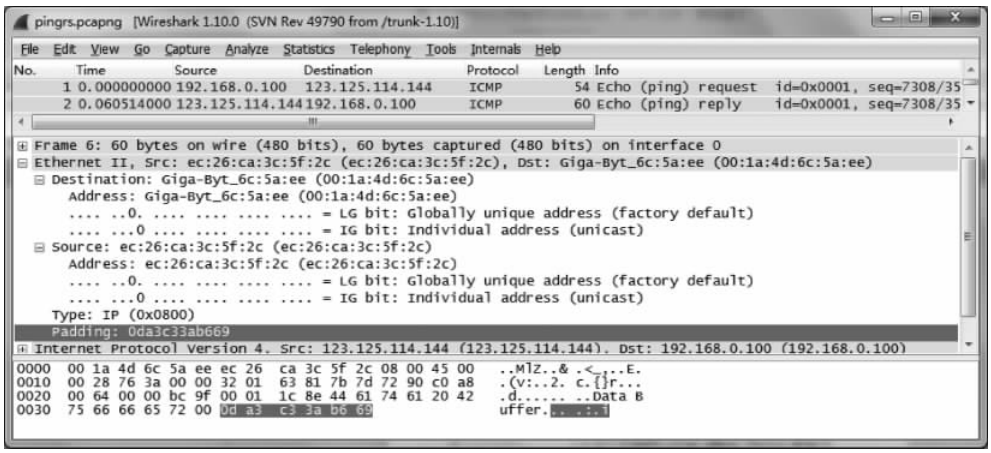

图 3-7 有 Padding 字段的以太网帧示例

(2) 在 Packet Tracer 中查看以太网帧。

步骤 1 启动 Packet Tracer,按图 3-8 所示建立一个简单的网络并相应做好 IP 地址配置,也可以打开以前建立的网络拓扑来进行实验。

图 3-8 中,PC0、PC1 和 PC2 的默认网关分别设置为指向路由器对应接口,路由器可以只配置静态路由。配置方法请参考 2.2 节。

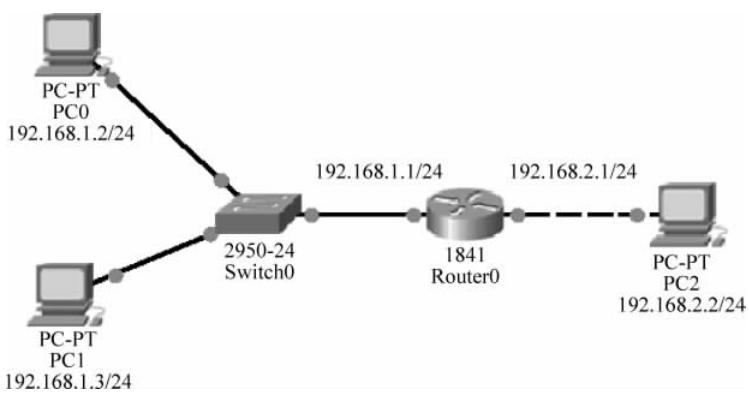

图 3-8 Packet Tracer 链路层实验

步骤 2 运行 ping 命令,查看链路层数据。

先在 Packet Tracer 工作窗口中单击 Simulation mode,切换到模拟模式,然后在网络拓扑中单击 PC0 图标,打开 PC0 的 Desktop 选项卡,单击 Command Prompt,打开命令行窗口,输入以下命令。

```
PC>ping 192.168.1.3
```

按 Enter 键后,马上就可以在 Event List 对话框中看到出现了对应的网络事件。

步骤 3 单击 Capture/Forward 按钮,会产生下一个事件,这样不断单击就可以看到 ping 程序运行中数据包传送的全部情况。

步骤 4 单击第一个事件的 Info 字段,打开 PDU Information 对话框中的 Outbound

PDU Details 选项卡，可以看到 PC0 发出的第一个 ICMP 数据包在链路层的封装情况，得到类似图 3-9 所示的数据。认真分析每一个数据域的内容。

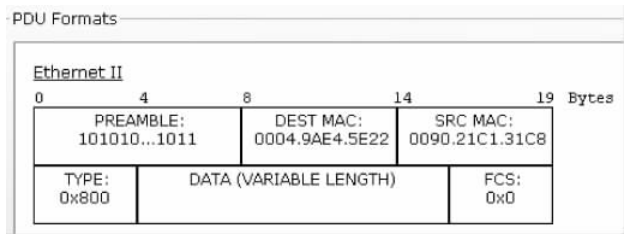

图 3-9　Packet Tracer 下查看以太网帧结构示例

4. 实验报告

记录自己的实验过程和实验结果，分析实验结果，比较说明用 Wireshark 和 Packet Tracer 捕获的以太网帧的异同，理解和掌握以太网帧结构。

5. 思考

(1) 图 3-6 所示示例中捕获的以太网帧类型为 0x0800，要怎样做才能捕获一个 0x0806 类型的帧？

(2) 图 3-9 所示显示的帧结构中的内容和实际网络中的数据有区别吗？

实验 3-2　IEEE 802 帧格式分析

1. 实验说明

分别通过在 Packet Tracer 和 Wireshark 中查看分析链路层的 IEEE 802 帧，学习了解不同的链路层帧格式。

实验中提到的 STP 在 IEEE 802.1D 文档中给出了其定义。STP 协议按照树的结构来构造网络拓扑，消除网络中的环路，避免广播风暴。CDP 是 Cisco 公司设计的专用协议，被 Cisco 公司的网络设备用来获取相邻设备的协议地址及发现这些设备的平台。

本实验的主要目的是观察 IEEE 802 帧不同于 Ethernet V2 的封装结构，对 STP 和 CDP 的具体工作原理和协议结构不作要求，需要时请参考有关书籍或资料。

2. 实验环境

Windows 操作系统及联网环境(主机有以太网网卡并连接局域网或 Internet)，安装有 Packet Tracer 6.0、Wireshark 1.10 和 GNS3(已配置好 IOS)。

3. 实验步骤

(1) 在 Packet Tracer 中捕获分析 IEEE 802 帧。

步骤 1　启动 Packet Tracer，建立如图 3-8 所示的实验网络。切换到模拟模式，直接单击 Capture/Forward 按钮，这时会看到从交换机 Switch0 发出的 STP 包。

步骤 2 任意选择一个 STP 协议包事件并单击其 Info 字段，会弹出 PDU Information 对话框，单击 Outbound PDU Details 选项卡，可以看到类似图 3-10 所示的 PDU Formats。

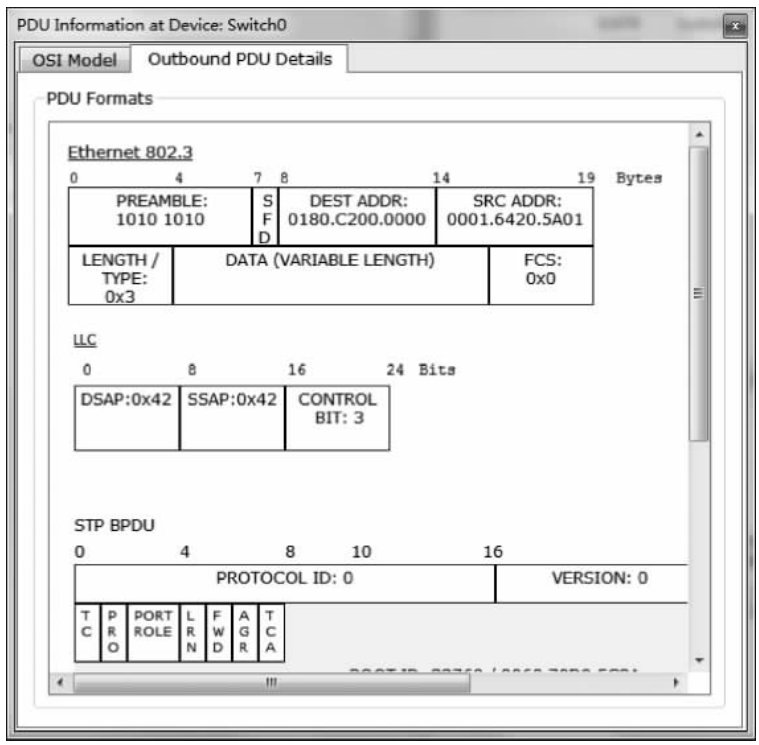

图 3-10 IEEE 802 帧结构示例

步骤 3 依据实验中 PDU Formats 实际显示的内容，对照图 3-2 认真分析帧的每一个字段。要特别注意 802.2 LLC 结构中 DSAP 和 SSAP 的取值。

STP 的配置往往和 VLAN 的配置相关联，图 3-8 所示实验拓扑中的 Cisco 2950 交换机有默认 VLAN1(尽管没有配置)，因此能够捕获 STP 包。和 VLAN 有关的具体操作请参考有关书籍或资料。

步骤 4 继续单击 Capture/Forward 按钮，直到看到交换机发出的 CDP 包。单击 CDP 协议包事件的 Info 字段，在打开的对话框中查看其 Outbound PDU Details 选项卡。

步骤 5 依据实验中 PDU Formats 实际显示的内容，对照图 3-2 认真分析帧的每一个字段。图 3-11 所示是 CDP 使用的 IEEE 802 帧的局部信息示例。

注意观察，由于有 SNAP 帧的存在，LLC 的 DSAP 和 SSAP 值为 0xAA。

(2) 用 Wireshark 捕获分析 IEEE 802 帧(选做)。

利用 GNS3 按图 3-8 所示构建高仿真的网络环境，用 Wireshark 捕获完全和真实网络环境中一样的 STP 或 CDP 帧。

这部分实验内容作为课后自学，这里不给出详细的实验过程，请参考 Packet Tracer 的实验内容和第 2 章关于 GNS3 的内容。

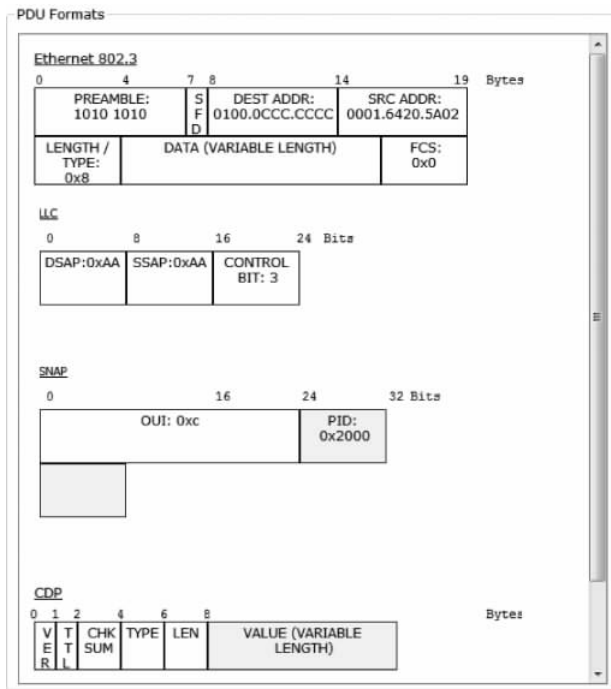

图 3-11　CDP 使用的 IEEE 802 帧的基本格式

4. 实验报告

记录自己的实验过程和实验结果，分析实验结果，比较说明 IEEE 802 帧和 DIX Ethernet V2 帧的异同。

实验 3-3　PPP 帧的观察

1. 实验说明

PPP 协议是当今网络上使用最广泛的串行链路协议。

PPPoE(Point to Point Protocol over Ethernet，以太网上的点到点协议)则是一种设计用于串行通信并为以太网进行了改造的 PPP。通过在标准 PPP 报文的前面加上以太网的报头，使得 PPPoE 提供通过简单桥接接入设备连接远端接入设备，并可以利用以太网的共享性连接多个用户主机。PPPoE 广泛用于用户通过拨号或专线方式接入 ISP 时建立点对点连接的收发数据。更多有关 PPPoE 的通信过程请查阅相关资料。

本实验通过在 Packet Tracer 中查看分析网络设备互联的 PPP 帧结构，学习了解串行链路中使用的帧格式；通过在真实上网时捕获 ADSL Modem 拨号连接时系统收发的数据包，了解链路层 PPPoE 帧格式。

2. 实验环境

Windows 操作系统及联网环境(主机有以太网网卡并连接局域网和 Internet)，安装有

Packet Tracer 6.0、Wireshark 1.10；ADSL Modem 拨号上网设备。

3. 实验步骤

(1) 在 Packet Tracer 中观察分析 PPP 帧结构。

步骤 1 启动 Packet Tracer，建立如图 3-12 所示的实验网络。先在两台路由器中添加 WIC-1T 串行接口模块，然后从 Router0 处选用串口线 DCE 连接两台路由器，配置好 IP 地址。

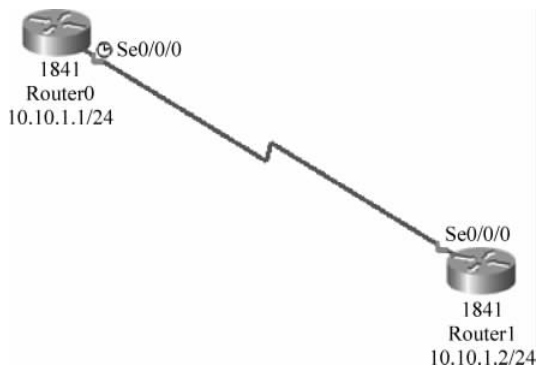

图 3-12 PPP 帧查看实验

步骤 2 将鼠标停留在 Router0 的接口处即会显示出时钟图标，表明 Router0 是 DCE，需要设置 DCE 的时钟频率。输入以下命令进行配置。

```
Router#show controllers s0/0/0          #可查看路由器是否为 DCE
Router#conf t
Router(config)#int s0/0/0
Router(config-if)#clock rate 9600       #设置串口同步时钟频率为 9600b/s
```

Router1 的串口为 DTE，不用配置时钟。实验中没有设置带宽，采用默认的 128kb/s。

步骤 3 继续配置 PPP，运行命令启用 PPP。

```
Router(config-if)#encapsulation ppp     #设置串行通信的封装方式为 PPP
Router(config-if)#^z                    #保存
Router#show int s0/0/0                  #查看设置参数
```

步骤 4 切换 Packet Tracer 到模拟模式，直接单击 Capture/Forward 按钮，这时会看到从 Router0 发出的 CDP 包。任意选择一个 CDP 协议包事件并单击其 Info 字段，查看 Outbound PDU Details 选项卡，可以看到类似图 3-13 所示的 PDU Formats。

对照 3.3.2 节的内容，分析 PPP 帧的结构。

(2) 观察分析 ADSL 拨号上网的 PPPoE 帧。

步骤 1 确认实验需要的 ADSL 宽带设备及拨号上网可用，Windows 中已配置有拨号上网快捷方式。

步骤 2 先在 Windows 中启动 Wireshark，选定本地网络接口并启动抓包，随即启动宽带连接拨号，输入用户名和口令，这时可以看到捕获的拨号连接数据包。

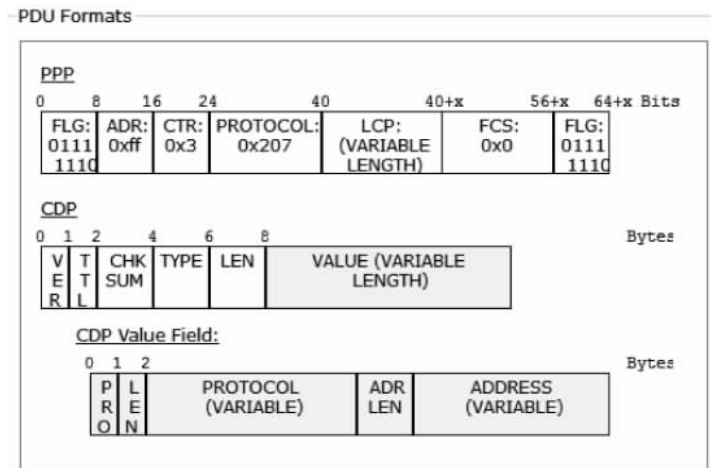

图 3-13　Packet Tracer 中 CDP 的 PPP 帧封装

步骤 3　在 Wireshark 的显示过滤器中输入“pppoed”，可以查看 PPPoE 帧信息，如图 3-14 所示。在 PPPoE 的 Discovery 阶段，以太网帧的 Type 域都设置为 0x8863。

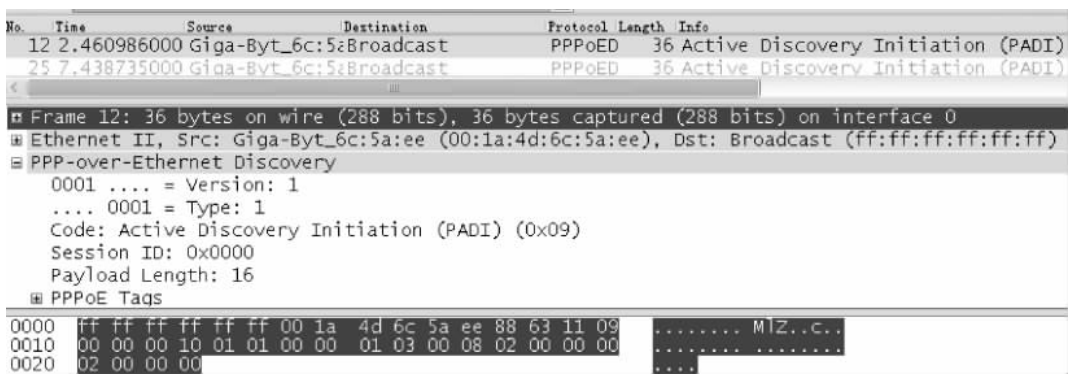

图 3-14　拨号上网的 PPPoE 帧结构示例

步骤 4　结合 PPPoE 的相关工作原理，分析自己捕获的 PPPoE 帧结构。

步骤 5　观察能否捕获 PPP 会话阶段（以太网帧的 Type 域都设置为 0x8864）的帧。PPPoE 的 payload 部分包含 0 个或多个 TAG。一个 TAG 是一个 TLV（type-length-value）结构，TAG_TYPE 域为 16 位值（网络字节序），请了解 TAG 取值的情况。

4. 实验报告

记录自己的实验过程和实验结果，分析实验结果，说明 PPP 帧的结构。

5. 思考

通过连接路由器之间的 WAN 口来观察 PPP 帧可以得到更多的内容，请思考应该如何实验。

实验 3-4 环回接口

1. 实验说明

本实验通过在 Packet Tracer 中配置路由器上的环回接口，查看了解其工作特点。

2. 实验环境

Windows 操作系统，安装有 Packet Tracer 6.0。

3. 实验步骤

步骤 1 启动 Packet Tracer，建立如图 3-15 所示的拓扑。

步骤 2 在 Router0 上配置网络接口地址，分别为 10.1.1.1/24 和 20.1.1.1/24，类似地在 Router1、Router2 和 PC1、PC2 上配置相应的 IP 地址。

在 Router0 上配置环回接口地址，命令如下。

```
Router0(config)#interface loopback 0
Router0(config-if)#ip address 17.17.1.1 255.255.255.255
```

然后配置 OSPF，命令如下。

```
Router0(config)#router ospf 1
Router0(config-router)#network 10.1.1.0 0.0.0.255 area 0
Router0(config-router)#network 20.1.1.0 0.0.0.255 area 0
```

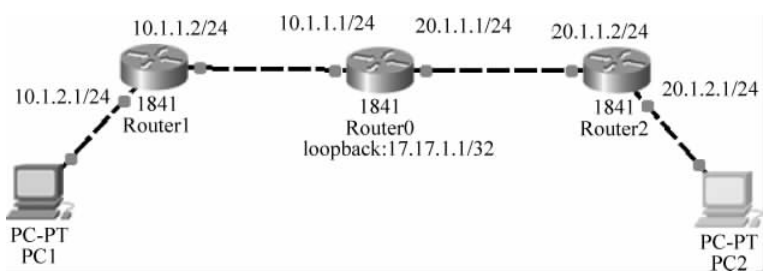

图 3-15 环回接口实验拓扑

步骤 3 配置好后，在 Router0 上执行命令“show ip protocol”，可以看到如图 3-16 所示的输出。

在图 3-16 中可以看到，Router0 上设置的环回接口地址 17.17.1.1 被当作了路由器的 Router ID。

步骤 4 在 Router0 上执行以下命令。

```
Router0#ping 17.17.1.1
```

在 Packet Tracer 中采用模拟方式可以看到 ping 发出的数据包只在 Router0 上收发，并收发成功。

```
Router#show ip protocol

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 17.17.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    10.1.1.0 0.0.0.255 area 0
    20.1.1.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    17.17.1.1            110      00:01:40
  Distance: (default is 110)
```

图 3-16　环回接口地址作为路由器的 Router ID

4. 实验报告

记录自己的实验过程和实验结果，分析实验结果，说明环回接口的特点。

5. 思考

(1) 路由器使用环回接口地址作为该路由器产生的所有 IP 包的源地址，从而提高数据的过滤效率，请参考路由器配置的有关资料了解其工作原理和特点。

(2) 在 Windows 主机上用 Wireshark 能不能抓到对 127.0.0.1 进行 ping 的数据包呢？如果抓不到，原因是什么？