第5章 数据链路层

【带着问题学习】

1. 为什么要有数据链路层？
2. 数据链路层的根本任务是什么？
3. 数据链路层主要完成哪些功能？

【学习目标】

1. 理解设计数据链路层的主要目的以及在网络通信中发挥的作用和功能。
2. 掌握点对点信道中的数据链路层协议。
3. 掌握广播共享信道中的有效接入方法。
4. 理解现行主流以太网帧格式以及构建局域网的设备和二层交换工作原理。
5. 掌握局域网中交换机的 VLAN 技术应用及其原理。

【本章关键术语】

逻辑链路控制子层，介质访问控制子层，冲突域，广播域，CSMA/CD，CSMA/CA，VLAN(虚拟局域网)，帧定界，差错控制，可靠传输，MAC 帧，MAC 地址，以太网，网桥，交换机，PPP 协议

数据链路层属于网络体系结构的第二层，是在物理层的基础上，以“帧”为对象，完成相邻节点之间的数据传输。要完成此任务，除了一条必需的物理线路外，还必须有一些规程或者协议来控制这些数据的传输，以保证传输数据的正确性，为实现这些规程或协议的硬件和软件再加上物理线路就构成了“数据链路层”(Data Link Layer，DLL)。

有时常把“数据链路层”简称为“链路层”，严格说这是不对的，因为“链路”和“数据链路”不是同一个概念。“链路”是指相邻节点之间的物理线路，“数据链路”是在数据链路层上构建的逻辑链路，二者是不同的概念。当然，它们又是有联系的，逻辑链路必须建立在物理链路之上，物理链路是通信的基础。物理链路是在物理层设备(如传输介质、物理接口等)和相应的物理层通信规程作用下形成的物理线路，是固定的，不可删除的(除非物理拆除)。逻辑链路则是通信双方在需要进行通信时，在数据链路层设备和相应的通信协议作用下建立的逻辑链路。

在多段物理链路组成的网络中，由多条物理链路和多个节点设备组成，其数据链路也是分段的，如图 5-1(a)所示。虽然实际通信路径在各个节点上经过了物理层，但单从数据链路层来看，这些连接起来的数据链路段屏蔽了物理连接上的差异，构成整个数据通信的数据链路。讨论数据链路层时，只讨论水平方向的数据流动，如图 5-1(b)所示。这也是第 3 章讨论的对等层通信。

5.1 数据链路层的作用与功能

设计数据链路层的主要目的就是在原始的、有差错的物理传输线路基础上，采取差错检测与控制以及流量控制等方法，**将有差错的物理线路改进成逻辑上无差错的数据链路，以便向它的上层(网络层)提供高质量的服务**。

在“物理层”中构建了数据传输通道，那为什么还要加一个“数据链路层”的功能呢？

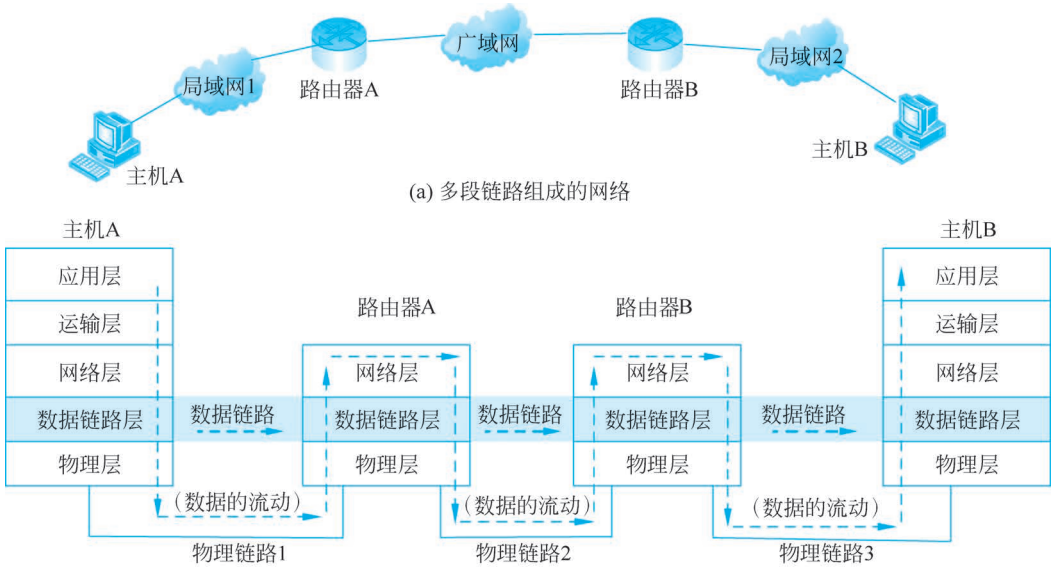


图 5-1　从数据的流动看物理链路与数据链路

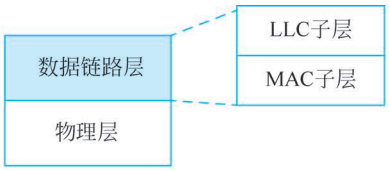
主要基于以下 3 方面的原因：

(1) 物理传输介质的多样性，导致物理层的通信规程也各不相同，通过数据链路层可以屏蔽物理层的差异性，从逻辑意义上构建一条性能稳定、不受传输介质类型影响、不受物理层通信规程影响的逻辑数据传输通道。

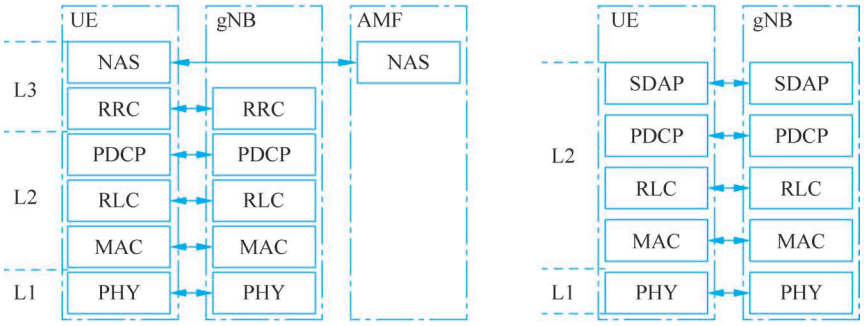
(2) 在物理层中的传输对象是一个一个的比特，数据链路层的传输对象是“帧”。数据链路层的发送方，将多个具有帧结构的数据送入物理层，形成物理层的连续比特流；在数据链路层的接收方，则需要从连续比特流中找出属于自己帧的数据。

(3) 物理层通过其通信规程考虑了数据传输的可靠性，但是从比特流的角度考虑，不能保证“帧”级的可靠，特别是在物理介质环境比较差的网络中，如无线通信、早期的电缆等环境。因此还需要在数据链路层考虑“帧”级的可靠传输甚至流量控制等。从“帧”级保证相邻节点之间的可靠传输，为上层(网络层)提供可靠的底层通信基础。

需要注意的是，在不同的网络体系结构中，数据链路层的结构和所包括的功能并不完全一样。在所有计算机网络体系结构中都直接或间接地包含了数据链路层。在 OSI/RM 体系结构和 TCP/IP 五层模型中，有明确的数据链路层；在 TCP/IP 四层模型中，数据链路层的功能包含在网络接口层中(实际是把物理层和数据链路层合并成为网络接口层)；在局域网体系结构中，数据链路层又分为两个子层：逻辑链路控制(Logical Link Control，LLC)子层和介质访问控制(Medium Access Control，MAC)子层，如图 5-2(a)所示；在 5G 空口协议栈的控制面，数据链路层分为 MAC 子层、无线链路控制(Radio Link Control，RLC)子层、分组数据汇聚协议(Packet Data Convergence Protocol，PDCP)子层，用户面协议栈中还增加了服务数据适配协议(Service Data Adaptation Protocol，SDAP)子层，如图 5-2(b)所示。



(a) 以太网数据链路层的两个子层



(b) 5G空口协议栈分层结构

图 5-2 不同系统中的数据链路层结构

数据链路层位于“网络层”之下，所以设计它的理想功能是向它的上层（网络层）提供透明、可靠的数据传输服务。其中，“透明”是指要使在数据链路层中所传输的数据在内容、格式及编码上都没有限制，也就是即便使用一些特殊用途的控制字符也能像正常的数据一样发送；“可靠的”传输是指使数据从发送方无差错地通过数据链路传输到目的接收方。

5.1.1 帧定界和透明传输

数据链路层位于网络层和物理层之间。在发送方，数据链路层是接收来自网络层的数据分组；在接收方，数据链路层是接收来自物理层的比特流。数据链路层的对等实体之间在水平方向进行逻辑通信的协议数据单元（PDU），称为“帧”（Frame），如图 5-3 所

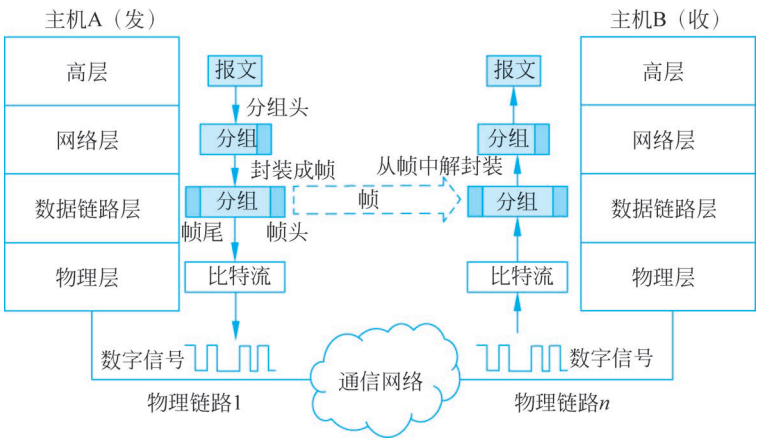


图 5-3 帧是数据链路层对等实体间的协议数据单元

示。数据链路层的成帧功能包含：一是将来自网络层的数据分组封装成数据帧，二是将来自物理层的一个个比特流组装成数据帧。这是数据链路层最基础的功能之一。

1. 发送方数据帧的封装

网络层传输的协议数据单元称为“数据包”(packet)，数据链路层中传输的协议数据单元是“帧”。当网络层的数据包通过层间原语送到数据链路层后，需加上数据链路层的协议控制信息和校验信息，即在每个包的前部加上一个帧头部，在包的结尾处加上一个帧尾部，把网络层的数据包作为帧的数据(净荷)部分，就构成一个完整的“数据帧”。帧头中含有帧的开始符，帧尾中含有结束符，分别作为帧的起始和结束标志，也就是帧的边界，如图 5-4 所示。

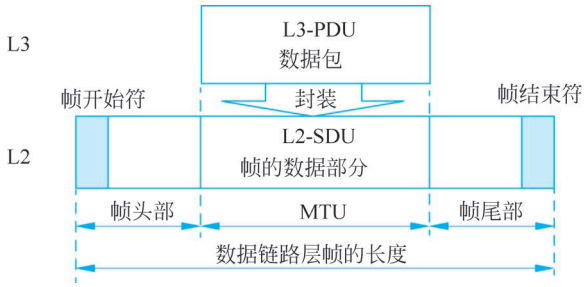


图 5-4　网络层数据封装成帧

由数据包封装成的数据帧，其大小是受限制的，即它的大小是受对应的数据链路层协议的最大传输单元(Maximum Transmission Unit，MTU)限制，而 MTU 是受物理层相关参数限制的，如以太网数据链路层的 MTU 是 1500B(不包含数据链路层的帧头和帧尾)，也就是网络层的 PDU(含网络层的协议头)最大不能超过 1500B。

2. 接收方数据帧的组装

发送方的数据帧封装是以网络层的数据包为单位，添加数据链路层的头尾即可，比较好理解。而接收方从物理层来的数据是连续的比特流，如何从连续的比特流中找出数据帧的边界呢？(先思考，看看自己能否设计一些方法，再看书中的介绍。)

接收方数据帧的组装称为**帧定界**，目的是让接收方能从接收到的二进制比特流中区分出帧的起始点和结束点。为了接收方能判别帧的边界，需要发送方配合，先在发送方设计好定界规则，接收方再根据该规则来判断。下面讨论几种常用的帧定界方法的基本原理，后面讨论具体数据链路层协议时再结合这里的原理进行应用分析。

1) 字节计数法

以一个专门的字段来标识当前数据帧内字节数的帧定界方法。

这种方法使用一个字段来标明本帧内的字节数。当接收方的数据链路层读到字节计数值时，就知道了本帧的长度，由此确定帧结束的位置，如图 5-5 所示。该方法中，“帧长度计数”字段非常重要，一旦它出错，后面所有的帧都会错乱。更严重的是，一旦错乱，无法恢复。

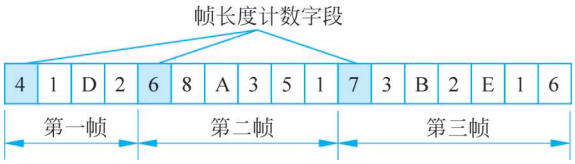


图 5-5　字节计数法

2）字符填充的首尾定界符法

这种方法是针对面向字符型的协议的，用一些特定的字符来界定一帧的开始和结束。如 DEC 公司的 DDCMP 协议选 ASCII 表中的“SOH”（注意，不是 SOH 三个字母，是 ASCII 码表中的二进制编码为 00000001 的控制字符）作为帧的开始标志，用“EOT”作为帧的结尾标志，如图 5-6 所示。IBM 公司的 BSC 协议选“SYN”作为帧的开始标志，用“ETX”作为帧的结尾标志。

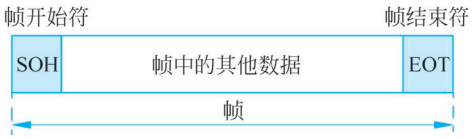


图 5-6　字符填充的首尾定界符法

此时，又出现一个新问题：如果帧中的数据部分含有与帧头和帧尾标志（“SOH”和“EOT”）相同的特定字符，则会被误判为帧的首尾定界符。该如何解决？

这个问题称为帧的**透明传输**问题，就是即使帧数据有帧头和帧尾标志相同的字符（这时它是用户的数据），也要能够正常地在帧中传输。

解决的方法是：当数据中出现“SOH”或“EOT”这类标志字符时，则发送时在其前面插入一个转义字符“ESC”（二进制编码 00011011），在接收方的数据链路层接收时先删除这个转义字符然后再把数据送往网络层。这种方法称为字节填充（byte stuffing）或字符填充（character stuffing）。

此时又出现新的问题：如果转义字符“ESC”也出现在数据当中呢？用同样的办法来解决，就是在转义字符的前面再插入一个转义字符。这样，当接收方收到连续的两个转义字符时，删除前面一个即可，如图 5-7 所示。

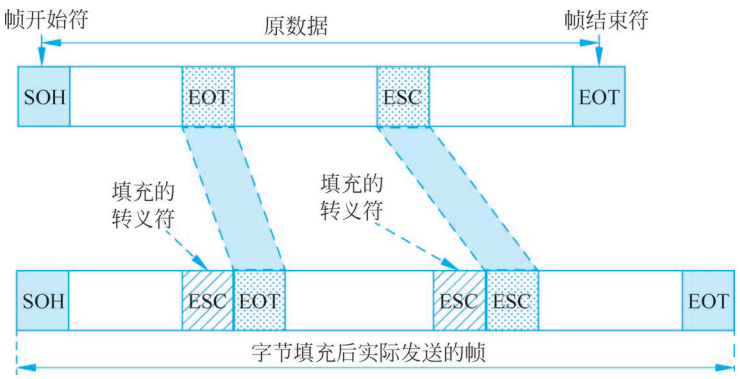


图 5-7　用转义符实现透明传输

3）比特填充的首尾标志法

这种方法针对面向比特型协议。通过在帧头和帧尾各插入一个特定的比特串来标识一个数据帧的起始与结束，这个帧头、帧尾特定比特串称为帧标志符。一般用 01111110 来作帧标志符，当传输的比特流为 10101111 时，组装成帧后就是 0111111010101111101111110。

它也同样要解决透明传输问题，观察 01111110 标志符，它的特征是有连续 6 个“1”，因此它采用零比特填充的方法，也称插零删零法：若数据中出现任何连续 5 个“1”，发送方在其后插入一个“0”，在接收方，数据链路层收到物理层送来的比特流后进行逆操作，每收到连续 5 个“1”的比特位则删除其后所插入的“0”比特，由此恢复原始数据。例如，传输的数据比特流为 10001111110101，下画线部分与标志符相同，发送方组装成帧后并插入“0”后变成 011111101000111110101010111110，再发送出去，如图 5-8 所示。

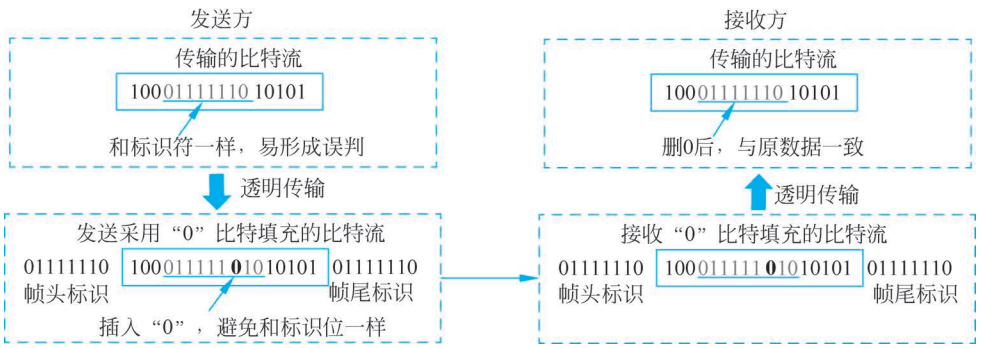


图 5-8　比特填充的方法举例

比特填充的帧定界方式很容易由硬件来实现，并且性能优于字符填充方式。其典型代表是 ISO 的 HDLC 协议，其标志符为 01111110。

4）物理层编码违例法

该方法是在物理层采用特定的比特编码方法时采用。例如曼彻斯特编码方法将数据“1”编码成“高-低”电平对，将数据“0”编码成“低-高”电平对，那么“高-高”电平对和“低-低”电平对在数据比特中是违例的，因此可以借用这些违例编码序列来界定帧的起始与终止。这种方法不需要任何填充技术，便能实现数据的透明性。但它的代价是曼彻斯特编码需要两倍的带宽。

5）校验字段搜索法

该方法应用在 ATM 的帧定界中，由于 ATM 的信元头只有 5 字节，没有空间做帧定界的标志字段，巧妙地利用信头差错校验（Header Error Check，HEC）字段与其他 4 字节的关系和 ATM 信元 53 字节定长的特点来实现帧定界。具体方法是对收到的比特流进行逐位 HEC 检验，使用 HEC 算法检测连续 5 字节的内容，如果满足 HEC 规定的算法，则确定为信元边界。一般需要连续若干 53 字节中的前 5 字节满足 HEC 算法才认为找到信元边界。

♥**【技术思想启发】** 从几种帧定界方法中可以看出，实现同样的帧定界功能，具体的实现技术多种多样，可见技术的灵活性。同时，这几种方法都是结合帧自身的特点，因地制

宜地利用其特征来找到具体方法的，充分体现了具体问题具体分析的威力。

5.1.2 差错检测

差错检测是各层通信协议的重要内容，也是做差错校正的基础。差错检测的基本思路是：按某种约定的算法对数据进行计算，将计算结果作为校验字段附加在数据帧中，一同传输到接收方，接收方根据同样的算法进行计算，如果与收到的校验字段一致，则判定数据正确；反之则判定数据有错误。常用的差错检测方法有奇偶校验码(Parity Check Code，PCC)、累加和校验(CheckSum)、循环冗余校验(Cyclic Redundancy Check，CRC)等。

奇偶校验码是奇校验码和偶校验码的统称，是最基本的检错码。它是由 $n-1$ 位信息元和 1 位校验元组成，可以表示为(n，$n-1$)。如果是奇校验码，则在附加上一个校验元以后，码长为 n 的码字中“1”的个数为奇数个；如果是偶校验码，则在附加上一个校验元以后，码长为 n 的码字中“1”的个数为偶数个。具体采用何种校验由双方事先约定好。

累加和校验是 IP、UDP、TCP 中使用的校验和算法，原理是将要校验的全部数据按 16 位作为一个分组(不足 16 位的在前面补零)，然后将所有 16 位的二进制数据相加。如果遇到进位则进位，将高于第 16 位的进位值加到最低位上，最后将按位取反，得到校验和。

循环冗余校验是通信、计算机、嵌入式系统中广泛应用的一种校验码，具有很好的检错能力。其基本思想是双方先选定一个“生成多项式”作为 CRC 计算的除数，将要发送的数据与生成多项式进行“模 2 除法”运算，所得到的余数就是该帧的 CRC 校验码，也称为 FCS(帧校验序列)。需要注意的是，余数的位数比除数位数只能少一位，哪怕余数前面位是 0，甚至是全为 0(即刚好整除时)都不能省略。带有 FCS 的数据帧到达接收方后，把接收到的数据帧与生成多项式进行“模 2 除法”运算，如果刚好整除，则认为该帧在传输过程中没出错；否则认为在传输中出现了差错。发送方和接收方的操作如图 5-9 所示。

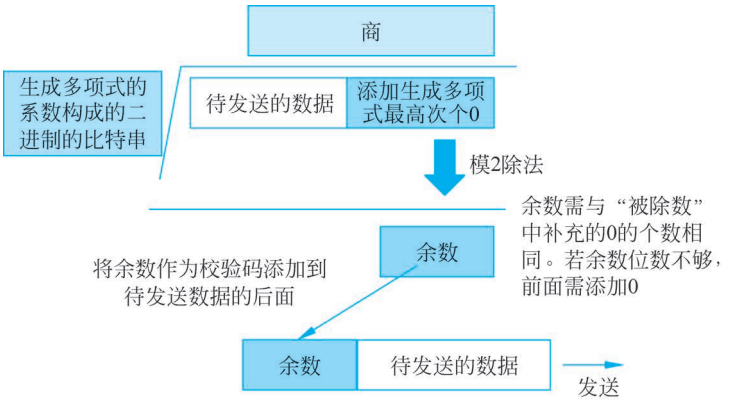
特别说明：模 2 除法与算术除法类似，它既不向上位借位，也不比较除数和被除数的相同位数值的大小，只以相同位数进行相除，相当于二进制中的逻辑异或运算。也就是对应位作比较，对应位相同则结果为 0，不同则结果为 1。

下面以一个例子来说明整个过程。现假设收发双方约定好的生成多项式为 $G(X)=X^4+X^2+1$，求出待发送数据 10111010 的 CRC 校验码。

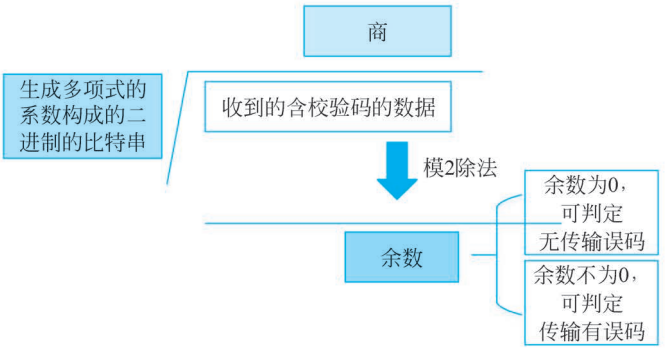
(1) 首先把生成多项式转换成二进制的比特串。由 $G(X)=X^4+X^2+1$ 可知，它一共是 5 位，根据生成多项式的含义，即只列出二进制值为 1 的位，也就是第 4 位、第 2 位和第 0 位的二进制均为 1，其他位为 0，该生成多项式的二进制比特串即为 10101。

(2) 因为生成多项式的最高次是 4，所以需要在待发送的数据后面先添加 4 个 0，作为发送方模 2 除法的“被除数”，生成多项式的二进制比特串作为“除数”，两者进行模 2 除法，如图 5-10 所示。

(3) 将计算出“余数”作为 CRC 校验码附加在待发送数据 10111010 后面，得到新的帧 101110101110(若余数位数不够 4 位，需要在余数前用 0 补充)，把这个帧发送到接收方。



(a) 发送方的CRC计算



(b) 接收方的CRC计算

图 5-9　CRC 操作原理

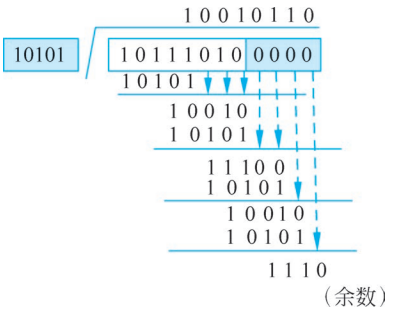


图 5-10　发送方 CRC 举例

（4）接收方收到这个数据后，把这个含校验码的数据作为模 2 除法的“被除数”，再用生成多项式的二进制比特串作为“除数”进行模 2 除法的余数检验。若余数为 0，则认为传输无误码；若余数不为 0，则认为传输出现错误。

由于生成多项式 $G(X)$ 直接关系到循环冗余校验的漏检率。在实际应用中，常采用以下几种已成为国际标准的生成多项式 $G(X)$：

$$\mathrm{CRC}-16=X^{16}+X^{15}+X^{2}+1$$

$$CRC-CCITT=X^{16}+X^{12}+X^{5}+1$$

$$CRC-32=X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1$$

采用上述生成多项式进行循环冗余检验，出现漏检的概率就非常低。需要特别注意的是循环冗余校验要求生成多项式必须包含最低次项 X^0。

5.1.3 可靠传输


视频

可靠传输不是数据链路层必需的功能，也不是数据链路层独有的功能。从物理层到应用层，每一层都可以实现可靠传输，具体由各层希望提供什么功能以及其下面各层提供的传输质量来决定。物理层关注的差错是比特级错误，一般采用前向纠错方式，通过物理层编码来实现可靠传输。物理之上各层关注的不是比特级的错误，而是关注分组（帧、包）的错误，包含分组内容错、分组丢失、分组重复、分组失序多方面，一般采用反馈重发方式进行纠错。在计算机网络中，网络层采用 IP 方式，不提供可靠服务。传输层既可以提供可靠传输服务，也可提供不可靠传输服务。数据链路层，针对通信质量良好的有线信道，不提供可靠型的传输协议；针对通信质量较差的无线信道，数据链路层提供基于反馈重发机制的可靠传输服务，如图 5-11 所示。

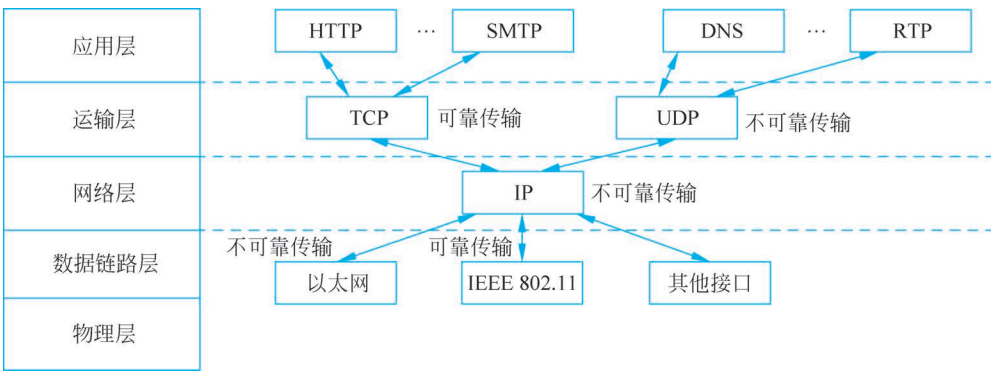


图 5-11 网络体系结构中的不可靠传输服务和可靠传输服务

1. 可靠传输应考虑的问题

可靠传输的目标是：发送方发来的信息，接收方无丢失、无差错、不重复、不失序地接收。要达到这个目标，需要考虑很多问题。

- 发送方要考虑的问题：以多快的速度发送帧？帧之间的间隔应该是多少？如何确认对方是否收到数据帧？如何确定接收方收到的数据是正确的？发出去的数据丢失后如何处理？接收方收到的数据出错了如何处理？
- 接收方要考虑的问题：能否及时处理接收到的数据？如何判断接收到的数据是正确的？数据正确与否如何告诉发送方？收到的数据是否重复？收到的数据顺序是否与发送方一致？
- 传输过程中要考虑的问题：传输的时延有多大？会不会拥塞？传输的数据会不会出错？传输的数据会不会丢失？数据出错了，丢失了，怎么办？

只有解决了这些问题，才能达到可靠传输的目标。解决的方法是设计合理的通信机制和通信协议，这给通信协议的设计带来很大的复杂性，这也是运输层 TCP 协议比 UDP 协议复杂很多的原因。

2. 通信协议设计分析

为了解决这些问题，需先明白为什么有这么多问题，其根源是运输层之下的通信不是理想的通信信道。先假设数据链路上的数据传输完全是“理想”的，假定：

(1) 传输数据不会出错也不会丢失；

(2) 接收方接收数据的缓存区容量是无穷大的，即发送方无论以何种速率发送数据，接收方都来得及接收。

在这两个假设的理想情况下，不需要任何通信机制和协议，发送和接收的任意信息都可以做到无丢失、无差错、不重复、不失序，如图 5-12 所示。

但是这两个完全“理想”的条件是不可能满足的，下面逐一进行还原分析。

1) 停止-等待协议

先保留第一个假定条件，去除第二个假定条件，即数据不会出错也不会丢失，但接收方缓冲区的容量是有限的。此时，可能出现的问题是，若发送方还是随意发送数据，其速率超过接收方的接收速率，则接收方的缓冲区会被逐渐占满，后续发送的数据，来不及处理也来不及暂存，就会出现数据丢失。

此时要解决的问题是，发送方根据接收方的缓冲区空间大小来控制发送数据的速率和节奏，也就是流量控制。当接收方缓冲区空间为零时，暂停发送，待接收方处理了数据，腾空一些缓冲区后，再根据缓冲区大小发送相应大小的数据。

这就需要一个通信协议来控制，接收方要及时地通知发送方是否可以发送数据，可以发送多大的数据，发送方根据该信息控制自己的发送数据量。最简单的方法就是发送方每发送完一个数据帧就必须停止下来，等待接收方发来确认(Acknowledgement，ACK)信息或否认(Negative-Acknowledgement，NAK)信息，再根据回应消息进行下一步动作。该协议称为简单的停止-等待协议(Stop-and-Wait，SW)，如图 5-13 所示。

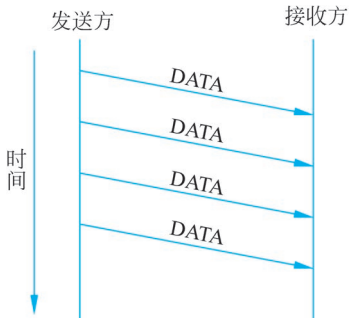


图 5-12　不需要任何协议控制的传输

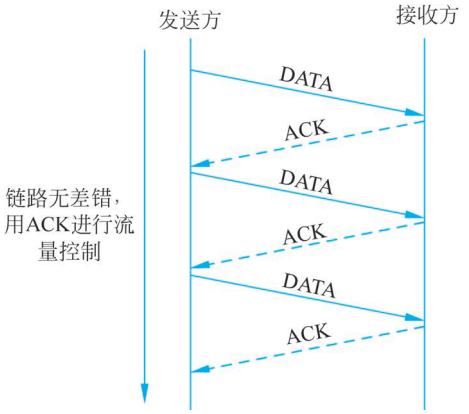


图 5-13　简单的停止-等待协议

但是在数据链路上传输数据时，不仅接收方的缓存区不可能是无限大的，同时在数据链路上的数据传输也存在差错和丢失的情况。也就是说，去掉前面两个假定条件后，需要进一步考虑数据出错或丢失后如何处理。

可以采取的方法是：

接收方发现数据出错（如何判断数据出错，参见 5.1.2 节），接收方向发送方回送一个表示出错的应答报文（NAK），发送方重新发送原数据。

数据丢失可能会出现两种情况：一是数据报文丢失，二是应答报文丢失。不管是哪种丢失情况，发送方都不会收到接收方的应答报文，无法根据应答报文做下一步动作。此时的解决办法是，利用双方共有的时间变量，即设置超时机制，规定在多少时间内没有收到应答报文，则自动重新发送原数据报。

此时又出现了新的问题：如果是应答报文丢失，那么接收方会收到两个相同的重复数据报文，接收方如何判断是重复报文？解决的办法是，给每个数据报文增加序号，接收方根据序号来进行判断。

增加超时重传和序号后的协议就是实际的停止-等待协议，如图 5-14 所示。

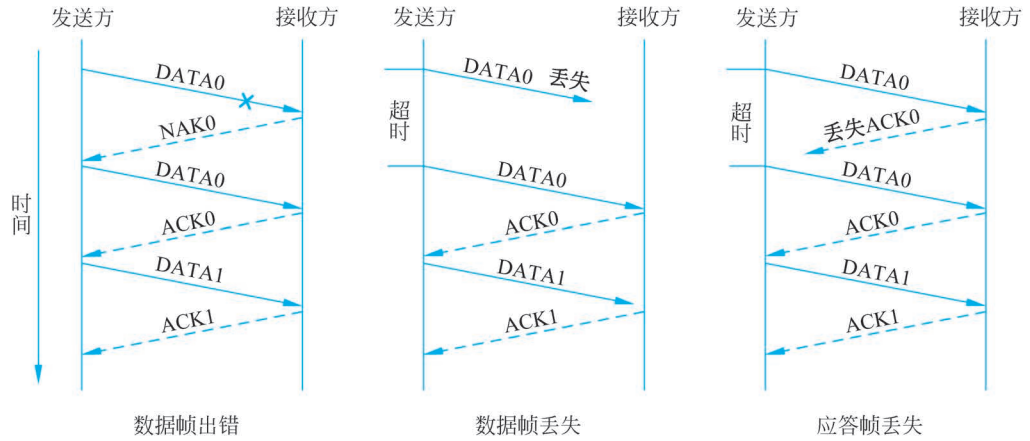


图 5-14　实际的停止-等待协议

由此，设计出了能实现可靠传输要求的通信协议，停止-等待式自动重复请求（Automatic Repeat Request，ARQ）协议，其要点是：

（1）发送方发送数据报后，启动计时器，在规定时间内收到接收方发来的肯定确认 ACK 后，再次发送数据。如果在规定时间内没收到确认报文 ACK，则重发本次的数据报（有的系统能反馈否定确认报文 NAK，可直接反馈 NAK 让发送方重传）。

（2）发送方发送数据报后，仍需在缓冲区中保留该数据，以便出错时重传。只有收到确认报文 ACK 后才从发送缓冲区中清除该数据。

（3）接受方收到数据报后，根据差错检测方法判断收到的数据报是否正确，若正确则回应 ACK，若错误则直接丢弃。

(4) 为保证接收报文不失序和判别重复报文,发送报文和回应报文都设置有序号字段,回应报文中的序号指示的是接收方期望收到的下一个序号。

停止-等待式 ARQ 协议的优点是协议简单,但存在很大的缺点,就是信道利用率太低,其计算方法如图 5-15 所示。信道利用率是指有效发送数据的时间占整个发送周期的比率。

- T_D:发送数据报的时延(发送时延)。
- RTT:收发双方之间的往返时间。
- T_A:发送确认报文的时延(忽略发送方和接收方对数据报文的处理时延)。

信道利用率 $U=[T_D/(T_D+RTT+T_A)]\times 100\%$。

例题:一个 100Mb/s 网络,每个数据报 1500B,收发双方相距 500km,则发送时延 $T_D=1500\times 8/(100\times 10^6)=0.12ms$,$RTT=500\times 10^3/(2\times 10^8)\times 2=5ms$,确认报文一般只有几十字节,其发送时延 T_A 可忽略。

信道利用率 $U=[0.12/(0.12+5)]\times 100\%=2.3\%$。通信信道很长时间都是空闲的,其利用率很低。考虑数据报重传,则信道利用率更低。

2) 连续 ARQ 协议

为了克服停止-等待式 ARQ 协议信道利用率太低的缺点,设计了连续 ARQ 协议,它采用流水线传输方式,发送方连续发送多个数据报,不用每发一个数据报都要停止等待确认,而是对按序达到的最后一个数据报发送确认报文,表示这个报文序号之前的所有报文都正确接收,如 ACK3 表示 3 号以前的报文(即 0～2 号)都正确接收,以累积确认方式提高信道效率,如图 5-16 所示。

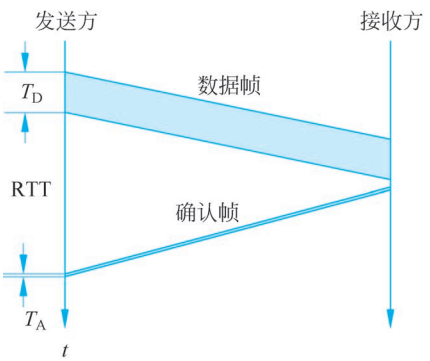


图 5-15　停止-等待协议的信道利用率

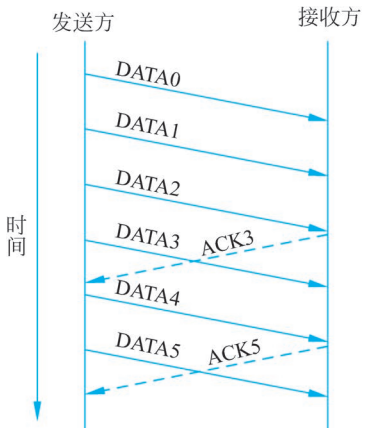


图 5-16　连续 ARQ 协议

ARQ 的关键是数据报文出错后如何处理,针对数据报文出错后不同的处理方法,连续 ARQ 又分为回退 N 式 ARQ 和选择重传 ARQ。

(1) 回退 N 式 ARQ。

回退 N 式 ARQ 的工作原理是:接收方收到的数据报文出错时,直接丢弃该报文,不

回送 ACK，收到后续序号数据报文，即使数据正确，也直接丢弃，因为它们不是接收方期望序号的报文，并回送带有期望序号的 ACK 报文，发送方收到该 ACK 后，从该序号开始重新发送后续数据报。

以图 5-17 为例，发送方从 DATA0 开始连续发送数据，接收方接收 DATA0 正确，回送 ACK1（注意，不是回送 ACK0，而是 ACK1，是接收方所期望收到的 DATA1 的序号）。接收方接收 DATA1 正确，回送 ACK2。接收方接收 DATA2 错误，直接丢弃，不回送 ACK，此时接收方仍然期望收到正确的 DATA2。后续接收到 DATA3，不是其期望的 DATA2，也直接丢弃，并回送 ACK2，告诉发送方期望收到 DATA2，后续的 DATA4、DATA5 也是同样处理。若发送方一直收不到 DATA2 的肯定确认，ACK3 出现超时，则发送方知道 DATA2 出错了，于是从 DATA2 开始全部重传。

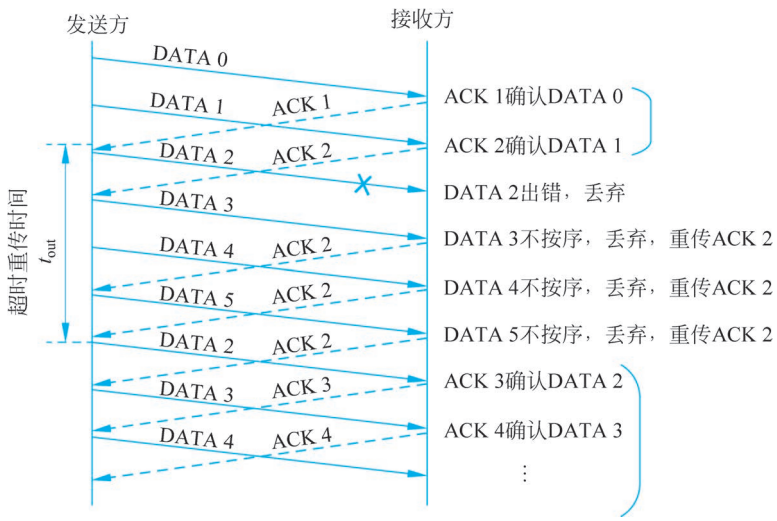


图 5-17　回退 N 式 ARQ

（2）选择重传 ARQ。

选择重传 ARQ 需要与回退 N 式结合起来使用，当接收方检测到错误，它就发送一个否定确认报文 NAK。NAK 可以触发该帧的重传操作，而不需要等到相应的计时器超时，因此协议性能得到提高。但此时接收到的帧会出现乱序情况，因此要求接收方具有数据帧排序能力，以及数据帧存储能力，用来缓存传输正确但顺序不连续的帧。

选择重传 ARQ 对正确接收的帧，接收方反馈 ACK，对有错误的帧，假设第 i 帧出错，那么接收方反馈 NAKi 表示第 i 帧出错，那么发送方就只是重发第 i 帧。如在图 5-18 中，发送方连续发送数据，其中 DATA2 数据错误，但不影响 DATA3、DATA4 的接收，接收方发现 DATA2 出错，则回送 NAK2，当发送方收到 NAK2 后，单独重发 DATA2。

最后总结一下可靠传输采用的机制及其相关用途，如表 5-1 所示。表中的“数据”可指数据链路层的帧，或是网络层的分组，或是传输层的报文。

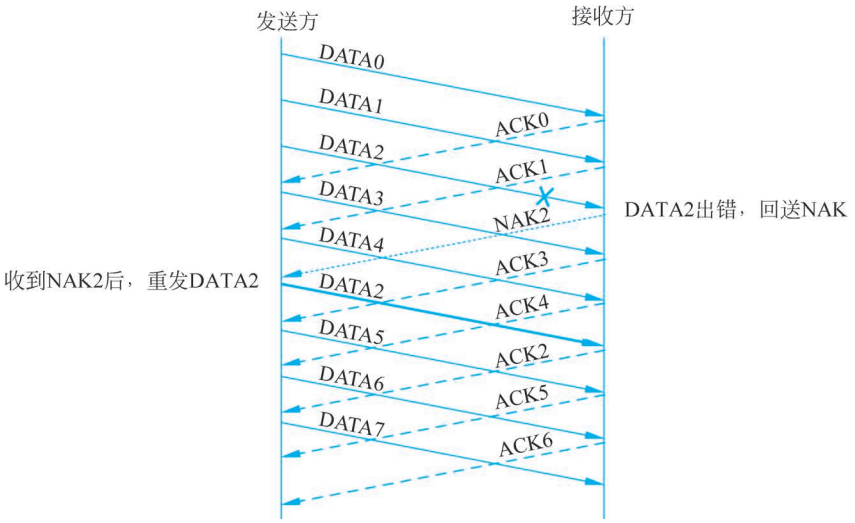


图 5-18　选择重传 ARQ

表 5-1　可靠传输机制及其相关用途说明总结

机　　制	用 途 说 明
校验和	用于在数据接收方发现数据是否在传输中出现错误，即“检错”
确认	用于接收方告诉发送方传输的数据已被正确地接收了。确认可以是逐个确认或者是累计确认，这取决于具体协议。确认报文中通常包含有被确认数据的序号
否定确认	用于接收方告诉发送方传输的数据未被正确地接收。否定确认报文中通常包含未被正确接收的数据的序号
序号	用于为从发送方流向接收方的数据按顺序编号。根据接收方所接收数据的编号可检测出丢失的分组；或者发现重复传输的冗余数据
定时器	因为数据（发送方发送的数据或者接收方响应的数据）在传输过程中的丢失，用于超时/重传发送方的数据。但同时会导致发送方重复传输数据，而使接收方产生冗余数据副本。但因为对数据采用“序号”编码，可以在接收方发现重复的数据
流水线传输方式	发送方可连续发送一定序号范围内的数据（未收到这些数据的确认之前），使发送效率提高。发送方的一定序号范围由“发送窗口”大小进行限制，数据的接收也需要满足传输的数据序号在“接收窗口”限制范围内的要求。发送窗口的限定的数据序号范围受接收窗口的变化影响
窗口	用于限制发送流量控制的一种机制。窗口大小可根据接收方接收和缓存数据的能力以及网络中拥塞的程度等情况来综合设定

5.1.4　流量控制

流量控制同可靠传输一样，**不是数据链路层必需的功能，也不是数据链路层独有的功能**。许多高层协议中也提供流量控制功能，只不过流量控制的对象不同。对于数据链路层来说，流量控制的是**相邻两节点**之间传输链路上的流量；对于传输层来说，流量控制的是从**源到最终目的端**之间的流量。

为什么需要进行流量控制？主要原因是发送方的数据发送速率高于接收方的数据

接收速率，接收方来不及接收，会造成数据丢失。流量控制实际上是对发送方数据传输速率的控制，使其数据发送速率不超过接收方所能承受的数据接收能力。

流量控制思路也很简单，就是接收方根据接收数据的情况，给发送方反馈一个信号，发送方根据这个信号进行发送速率的控制。具体的方法有两种：一种是基于反馈信息的流量控制方法，另一种是基于窗口值的流量控制方法。

1. 基于反馈信息的流量控制方法

在5.1.3节中已分析，为解决接收方缓冲区的有限容量问题，采用了两种通信方式：一种是停止-等待方式，另一种是连续ARQ方式。在停止-等待方式下，接收方回复的ACK反馈信息控制着发送方的发送速率，自然可以保证发送速率小于接收速率，但其效率太低。在连续ARQ方式中，对流量控制的方式是设置专门用于流量控制的反馈帧，如早期的SS7号信令中的回送SIB(Status Indication Busy)信令，以太网中回送的Pause信号，都是表示接收方忙，通知发送方暂停发送。

2. 基于窗口值的流量控制方法

这种方法采用滑动窗口方式，在确认帧中回送允许发送方可以发送的数据量大小，限制发送方的数据传输速率，接收方无须对每一帧数据都发回确认帧。这种方法与传输层的流量控制类似，不同的是数据链路层的流量控制是针对点对点的通信系统，而传输层是针对端到端的系统。

这种基于滑动窗口机制的流量控制具体可参见7.4.6节。

5.1.5 数据链路层的实现

图5-19展示了一个典型的主机体系结构。数据链路层的主体部分是网络适配器(Network Adapter)或称网络接口卡(Nerwork Interface Card，NIC)，俗称“网卡”。网络适配器的核心是链路控制器，该控制器通常是一个实现了多种链路层功能(如成帧、链路接入、差错检测等服务)的专用芯片。

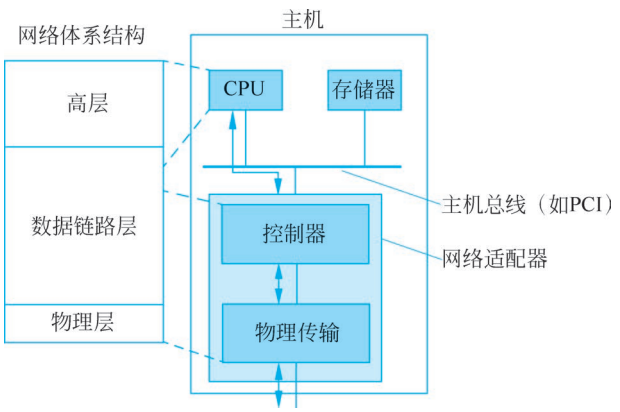


图5-19 在主机中的网络适配器

网络适配器通过主机内的总线与主机中的其他部件连接，从图5-19可以看出，大部分数据链路层的功能是在硬件中实现的，但其他功能是靠主机CPU上的软件实现的，包

括发送时组装链路层寻址信息、激活控制器硬件、接收时响应控制器中断、处理差错条件和将数据帧向上传递给网络层等。


视频

5.2 点对点信道中的协议

数据链路层存在两类信道：点对点信道和广播信道。

点对点信道是由一个节点与另一个节点连接起来的链路，用于建立点对点通信，它所采用的是点对点协议，如 HDLC、PPP、PPPoE。

广播信道是一个节点与多个节点连接建立起来的链路，用于建立点对多点通信，它所采用的通常是点对多点协议，如以太网 IEEE 802 系列协议、WLAN 协议等。在一对多的广播信道上会连接多个主机，因此必须使用专用的共享信道协议来协调这些主机的数据发送，其过程相对比较复杂。

所谓点对点信道，是指两个没有经过任何中间设备的节点（网卡或网络设备端口）之间构成的信道，如图 5-20 所示。点对点信道需要封装点对点类型的数据链路层协议，点对点两端的节点只与对方节点进行连接，不存在寻址和介质争用问题。

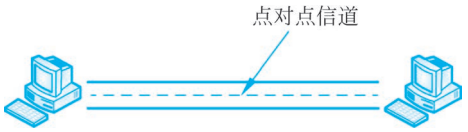


图 5-20　点对点信道示意图

5.1 节介绍的各种数据链路层功能，都会通过数据链路层的协议来实现。数据链路层中常见的协议可分为两大类：面向字符的链路层协议和面向比特的链路层协议，几类协议如表 5-2 所示。

表 5-2　数据链路层协议分类说明

类　型	主要协议名称	提出者
面向字符的链路层协议	**BSC**（Binary Synchronous Communication，二进制同步通信协议）	IBM
	DDCMP（Digital Data Communications Message Protocol，数字数据通信消息协议）	DEC
	SLIP（Serial Line Internet Protocol，串行线路互联网协议）	
	PPP（Point-to-Point Protocol，点对点协议）	IETF
面向比特的链路层协议	**SDLC**（Synchronous Data Link Control，同步数据链路控制）	IBM
	ADCCP（Advanced Data Communication Control Protocol，高级数据通信控制协议）	ANSI
	HDLC（High-level Data Link Control，高级数据链路控制）	ISO
	LAP（Link Access Procedure，链路访问规程）	CCITT
	PPP（Point-to-Point Protocol，点对点协议）	IETF
	PPPoE（PPP over Ethernet，以太网上的点对点）	IETF

为什么会有这么多种数据链路层协议呢？主要因为基于点到点信道的广域网采用非常多形式的物理连接方式，比如有点到点专线形式的、采用电路交换方式接入的以及采用分组交换方式连接的等情况，如图 5-21 所示。

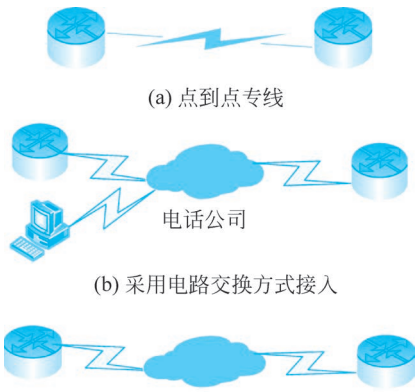


图 5-21 多形式的物理连接

5.2.1 SDLC 和 HDLC 协议

在面向比特的同步传输协议中，数据块是作为比特流，而不是作为字符流来处理的，所以称之为面向比特的同步传输。在面向比特的同步传输中，每个数据块的头部和尾部都用一个特殊的比特序列（如 01111110）来标记数据块的开始和结束。最具有代表性的协议是 IBM 的同步数据链路控制（SDLC）协议、国际标准化组织（ISO）的 HDLC 协议、美国国家标准协会 ANSI 的 ADCCP。

SDLC 协议是一种适用于 IBM 的系统网络体系结构（Systems Network Architecture，SNA）的数据链路层协议，属于单链路规程；HDLC 协议是一种在同步网上传输数据、面向位的数据链路层协议，属于多链路规程。国际标准化组织 ISO 的 HDLC 协议是对 IBM 的 SDLC 协议进行了改进和标准化的协议，其格式如图 5-22 所示。所有面向比特的数据链路控制（DLC）协议均采用统一的帧格式，且不论是数据还是控制信息均以帧为单位传送，如图 5-22 所示。

图 5-22 SDLC 和 HDLC 帧格式

HDLC 帧格式中包含用于定界的标志字段、地址字段、控制字段、信息字段和帧校验序列（FCS）字段。

1. 标志字段 F（Flag）

HDLC 协议规定，所有信息传输必须以一个标志字段 F（Flag）开始，以同一个标志字段结束。标志字段的值均为 01111110，用于帧定界。

2. 地址字段 A（Address）

地址字段占 8 位或 16 位（即 1 字节或 2 字节），通常采用 8 位长度。当地址字段值为

00000000 时，为空地址，不能分配给任何站点，用于测试数据链路的状态；当地址字段值为 11111111 时，为广播地址，可以把一个数据帧发送到同一网段中的其他所有站点。

3. 控制字段 C(Control)

该字段占 1 字节，是用来实现 HDLC 协议的各种控制信息，并标志是否是数据(因为它可以标志不同的帧类型)。根据控制字段的取值，可以把 HDLC 帧划分为三大类，即信息帧 I(Information)、监控帧 S(Supervisory)和无编号帧 U(Unnumbered)。

4. 信息字段

该字段包含了用户的数据信息和来自上层的各种控制信息。在信息帧和某些无编号帧中具有该字段，并且可以是任意长度的比特序列。

5. 帧校验序列字段(FCS)

该字段可使用 16 位 CRC，对两个标志字段之间的整个帧内容进行校验。CCITT 和 ISO 推荐使用的生成多项式为 $G(X)=X^{16}+X^{12}+X^5+1$；IBM SDLC 使用的生成多项式为 $G(X)=X^{16}+X^{15}+X^2+1$。

在通信线路质量比较差的年代，在数据链路层采用能实现可靠传输的 HDLC 协议，在当时是比较流行的。但现在 HDLC 协议已很少使用，因为现在点对点有线链路的误码率已经非常低，常采用比 HDLC 协议实现方法更简单的点对点协议(Point-to-Point Protocol，PPP)。

5.2.2 PPP 传输协议

PPP 是一种应用更广泛的广域网数据链路层协议。如在使用调制解调器(Modem)进行拨号连接时就需要用到它、路由器设备间的 Serial 口之间的连接也要封装这个协议。Internet 对 PPP 的正式标准是 RFC 1661 和 RFC 1662，主要针对 Internet 用户的计算机通过点对点链路接入某个网络服务提供商(ISP)进而接入 Internet，也广泛应用于广域网中路由器之间的专用线路。

1. PPP 简介

在点对点链路上，最早使用的数据链路层协议是一个在串行线路上对 IP 分组进行封装的简单的面向字符的协议——串行线路互联网协议(Serial Line Internet Protocol，SLIP，该协议用于帮助用户通过电话线和调制解调器接入 Internet)，但是 SLIP 有许多根本无法适应当时网络技术发展和应用需求的不足，如：

- 连接速率低。
- 不能自动分配 IP 地址，通信的双方都事先必须知道对方的 IP 地址。
- 没有协议类型字段，不具备同时处理多种网络层协议的能力。
- 没有校验字段，无法判断接收数据是否正确。

PPP 是在 SLIP 的基础上发展起来的点对点数据链路层协议，可解决 SLIP 协议存在的全部问题。它既支持面向字节的异步链路，也支持面向比特的同步链路，它的特性包括：

- 在连接速率上可远高于 SLIP。
- 具有验证协议，具有更高的网络安全性。
- 设置协议类型字段，支持多种上层协议。
- 设置帧校验字段，可实现无差错接收。
- 提供了一整套方案来解决链路建立、维护、拆除、上层协议协商、认证等问题。

PPP 既支持面向字节的异步链路，也支持面向比特的同步链路。

2. PPP 帧结构

PPP 的帧格式如图 5-23 所示，它借用了 HDLC 的帧格式。

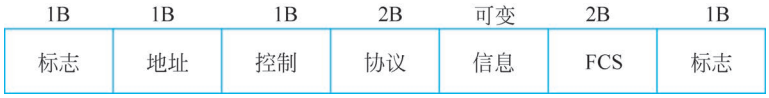


图 5-23 PPP 帧格式

PPP 帧格式以 HDLC 帧格式为基础，做了很少的改动，增加了协议类型字段。它们的主要区别是：PPP 是面向字节的(帧的长度必须是字节的整数倍)，而 HDLC 是面向比特位的(帧的长度可以是任意个比特)。

标志：用来标志帧的开始和结束，占 1 字节(8 位)，值固定为 01111110(0x7E)，与 HDLC 帧中的一样。

因为 PPP 既支持物理层的异步链路，也支持物理层的同步链路，因此其协议既可以是面向字节的，也可以是面向比特的，导致它的透明传输也既要支持字节填充方式，也要支持比特填充方式(详见 5.1.1 节)。

当 PPP 采用面向字符的异步链路时，其“转义字符”为 0x7D，具体做法是将信息字段中出现的每一个 0x7E 字节转变成 2 字节序列(0x7D，0x5E)；如果信息字段中出现一个 0x7D 的字节，则转变成 2 字节序列(0x7D，0x5D)。如果信息字段中出现 ASCII 码的控制字符，则在该字符前加一个 0x7D 字节。

当 PPP 采用面向比特的同步链路时，使用比特填充(插零删零法)来实现透明传输。

地址：用来标志对方节点地址的，占 1 字节(8 位)。由于 PPP 是点对点协议，明确知道对方节点，在实际通信中，所以无须知道对方的数据链路层地址，此地址字段实际上没有什么意义。在 PPP 帧中，此地址字段固定为 11111111(0xFF)标准广播地址。这与 HDLC 帧中的地址字段是不一样的。

控制：占 1 字节(8 位)，此字段在 PPP 帧中也没有什么意义，值固定为 00000011(0x03)。

协议：用来指示在信息字段中封装的数据类型，占 2 字节(16 位)，这是与 HDLC 区别最大的地方，是 PPP 协议新增的字段。该字段值的取值和含义如表 5-3 所示。

表 5-3 协议字段的取值和含义

字 段 数 值	表示信息字段数据包类型
0x0021	IP 数据包
0xC021	链路控制协议 LCP 数据包
0x8021	网络控制协议 NCP 数据包

续表

字 段 数 值	表示信息字段数据包类型
0xC023	PAP 安全性认证数据包
0xC223	CHAP 安全性认证数据包
0x0029	Apple Talk 协议数据包

信息(Information)：来自上层的有效数据，可以是任意长度，不超过 1500 字节。

帧校验序列(FCS)：使用 16 位的循环冗余校验计算信息字段中的校验和，以验证数据的正确性。

3. PPP 的组成

PPP 由 3 部分组成，链路控制协议(LCP)、网络层 PDU 封装到串行链路的方法以及网络控制协议(NCP)，如图 5-24 所示。

图 5-24　PPP 的组成

网络控制协议(Network Control Protocol，NCP)包含多个协议，其中的每一个协议分别用来支持不同的网络层协议。如 TCP/IP 中的 IP、Apple 公司的 AppleTalk 以及 Novell Netware 网络操作系统的 IPX 等。

网络层 PDU 封装到串行链路的方法是指网络层 PDU 作为 PPP 帧的数据载荷被封装在 PPP 帧中传输。网络层 PDU 的长度受 PPP 的最大传输单元 MTU 的限制。

链路控制协议(Link Control Protocol，LCP)是用来建立、配置、测试数据链路的连接以及协商一些选项。

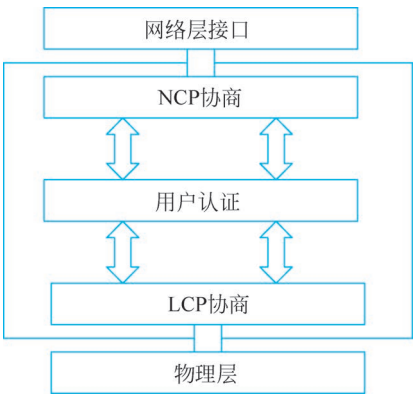


图 5-25　PPP 栈层次结构

它的协议栈层次结构如图 5-25 所示。

物理层建立连接后，LCP 协商一些 PPP 参数，建立数据链路层的连接，用户认证完成用户的身份认证。NCP 协商进行网络层参数配置。

4. PPP 的工作过程

在 PPP 通信中，链路不是始终连接的，所以需要在 PPP 通信前，通信双方协商建立链路连接，只有链路成功建立后，数据才能进行传输，在数据传输完成后，拆除已建立的链路。

整个过程分为 6 个阶段，即链路静止(Link Dead)阶段、链路建立(Link Establish)阶段、鉴别(Authenticate)阶段、网络层协议(Network-Layer Protocol)阶段、链路打开(Link Open)阶段和链路终止(Link Terminate)阶段。在不同的阶段进行不同协议的协商，只有前面阶段的协议协商出结果后，才能转入下一个阶段协议的协商，各阶段的转移关系如图 5-26 所示。

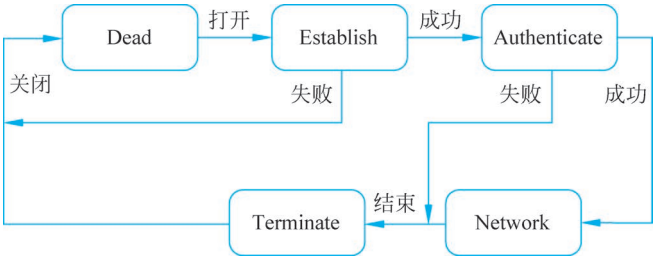


图 5-26　PPP 状态转移关系

(1) 在链路静止阶段，用户与 ISP 的路由器之间没有物理连接，当用户拨号接入 ISP 时，路由器的调制解调器对拨号做出确认，并建立一条物理连接，进入链路建立阶段。

(2) 然后 PPP 通过封装了 LCP 的 PPP 帧与接口进行协商，内容包括工作方式、认证方式、压缩方式和最大传输单元等。协商成功则进入鉴别阶段，并保持 LCP 的激活状态，此时则建立好数据链路层的链路；若协商失败则返回链路静止阶段，LCP 的状态为关闭。

(3) 在鉴别阶段，对请求连接的用户进行身份认证，以确定采用密码认证协议(Password Authentication Protocol，PAP)还是质询握手认证协议(Challenge Handshake Authentication Protocol，CHAP)进行身份认证，或者是不采用任何身份认证方式。如果认证成功，则进入网络层协议协商阶段；如果身份认证失败，则进入链路终止阶段，拆除链路，返回到链路建立阶段，LCP 状态转为关闭。

(4) 在网络层协议协商阶段，使用封装了 NCP(网络控制协议)的 PPP 帧与对应的网络层协议进行协商，并为用户分配一个临时的网络层地址，然后进入链路打开阶段，可以正常通信。

(5) 在链路打开阶段，PPP 链路将一直保持激活通信，通信完毕后有明确的 LCP 或 NCP 关闭请求，或发生某些外部事件(如用户的干预)，进入链路终止阶段，NCP 释放网络层连接，收回原来分配出去的 IP 地址。接着，LCP 释放数据链路层连接。最后释放的是物理层的连接，返回到链路静止阶段。

5. PAP/CHAP 身份认证

在 PPP 的鉴别阶段，可采用 PAP 或者 CHAP 对连接用户进行身份认证，以防止非法用户的 PPP 连接。如果双方达成一致，也可以不采用任何身份认证方式。

PAP 身份认证过程非常简单，是一个二次握手机制，整个认证过程即两步，即：被认证方发送认证请求，然后是认证方给出认证结果。它的特点是：用于身份认证的用户名

和密码在网络上以明文方式传输，所以这种方式并不是一种安全有效的认证方式。PAP可分为单向和双向认证。双向方式是PPP链路的两端同时扮演客户端和服务器双重角色，两端都需向对方发送认证请求，同时对对方发来的认证请求进行认证。

CHAP认证采用三次握手机制，整个过程经过3个步骤：

（1）认证方要求被认证方提供认证信息；

（2）被认证方提供认证信息；

（3）认证方给出认证结果。

CHAP认证相对PAP认证来说更加安全，因为身份认证信息在网络上传输时通过MD5等摘要加密协议产生随机密钥，而且这个密钥具有时效性，原密钥失效后会随机产生新的密钥。即便在通信过程中，密钥被非法用户截获并破解，也不能用于后面的通信截取。与PAP认证一样，CHAP认证也可以是单向或者双向的。如果是双向认证，则要求通信双方均要通过对对方请求的认证，否则无法在双方建立PPP链路。

6. PPPoE传输协议

PPP要求进行通信的双方之间是点到点的关系，不适于广播类型的以太网和另外一些多点访问类型的网络。在宽带接入技术的发展背景下，需要在广播式的网络上建立、维持各主机与访问集中器之间点对点的关系，每个主机与访问集中器之间能建立唯一的点到点的会话。由此，PPP也衍生出了新的应用，比如在ADSL（Asymmetrical Digital Subscriber Loop，非对称数据用户环线）接入方式中，PPP派生出符合宽带接入的PPPoE（PPP over Ethernet）、PPPoA（PPP over ATM）等协议。PPPoE是将PPP封装在以太网框架中的一种网络隧道协议，提供远程的多个用户主机接入功能，并且能够提供数据传输的计费数据，解决用户上网收费等实际应用问题，因而被运营商广泛应用于接入网络。PPPoA是在ATM网络上运行PPP来管理用户认证的方式，它的原理和作用与PPPoE基本相同。

PPPoE利用以太网资源，允许在以太网广播域中的两个以太网接口间创建点对点隧道通信，在以太网上运行PPP来进行用户认证接入的方式如图5-27所示。

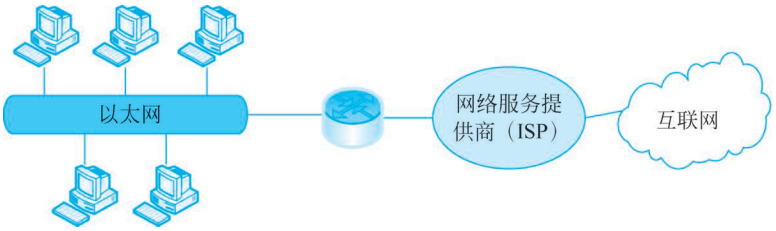


图5-27　基于以太网的PPP认证场景

既然PPPoE的协议是运行在以太网上的，因此其帧格式也建立在以太网帧格式基础之上，将PPPoE的PDU数据直接封装到以太网帧中，如图5-28所示。

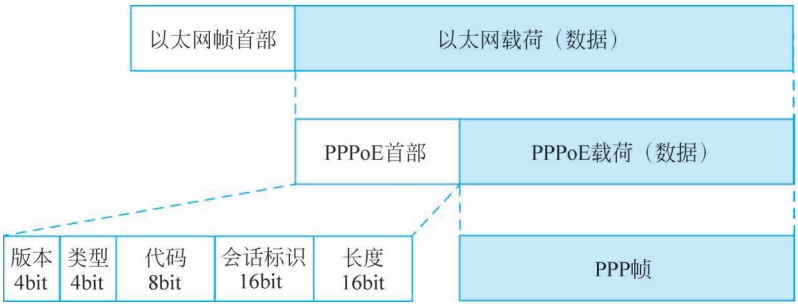


图 5-28　PPPoE 的封装

5.3　广播信道的链路接入协议

广播信道是一个信道被多种设备（主机）所共享，存在物理介质共享的情况，如图 5-29 所示，属于共享介质型网络。在广播信道中，一个节点主机发送的数据可以同时被多个节点主机接收到，对应的链路是“广播链路”。最典型的广播型数据链路层协议就是经常用的以太网协议和 WLAN 协议。在广播型网络中，一个用户发送的一个广播包可以通过交换机广播到局域网的所有节点上。

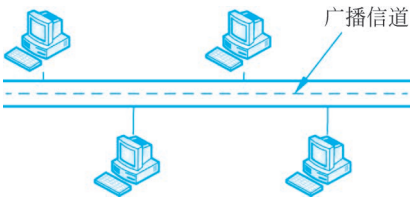


图 5-29　广播信道示意图

由 5.1 节可知，以太网的数据链路层包含介质访问控制子层。介质访问控制对于广播型网络（以太局域网、WLAN 等）非常重要，因为它要解决广播信道中的介质争用问题。MAC 子层仅在广播型网中有用，而对于点对点网络中就没有存在的意义，因为点对点网络中不需要寻址，也不存在一般意义上的信道争用问题。对 MAC 子层的学习主要适用于局域网，而在广域网中的路由器之间基本都是点对点连接的，不存在寻址和介质争用问题。

为解决链路层的寻址问题，首先要理解链路层寻址中的一个重要概念：MAC 地址。

5.3.1　MAC 地址

以太网的数据链路层分为 LLC 子层和 MAC 子层，所以数据链路层有两种不同的数据帧——LLC 帧和 MAC 帧。这里所说的帧是“MAC 帧”。对于物理层来说，最终的数据链路层的帧也是 MAC 帧。

局域网中与接入的各种传输介质相关的问题都放在 MAC 子层来解决，还负责实现无差错数据传输。MAC 子层的主要功能包括：MAC 帧的封装与解封装、实现和维护各种 MAC 协议、比特流差错检测、MAC 寻址等。由于在 IEEE 802 系列局域网标准中有不同规定的 MAC 子层协议（对应于不同的网络标准），所以其 MAC 帧的格式也各不相同，但无论是哪一种 MAC 帧协议，都具有 MAC 地址，以便于在局域网内部实现二层的寻址。

MAC 地址用于在广播信道中识别互连的节点。但 MAC 地址实际上标识的是网络

中具体的物理节点(主机或者路由器),它更像是一个“名字”。这些标识通常“刻写”在物理节点的网络适配器(即网卡)中,也就是说,MAC 地址实际标识的是网络中具体物理节点的网络接口(一个具体的物理节点可以包含多个网络接口)。MAC 地址通常是被固化在网卡的 ROM 中,一个重要的性质就是不允许有相同地址的两块适配器。那各个国家和地区的不同公司在生产适配器时,如何能保证这个 MAC 地址是唯一的呢?

MAC 地址由 IEEE 的注册管理机构负责管理,长度为 6 字节(48 位),具有扁平化结构,其结构如图 5-30 所示。

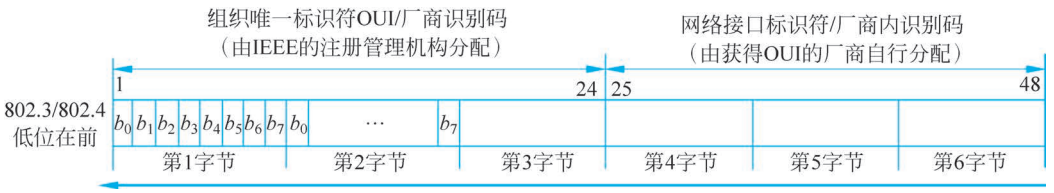


图 5-30　MAC 地址格式

其中,前 3 个字节是组织唯一标识符(Organizationally Unique Identifier,OUI),由生产网络设备的厂商根据需要向 IEEE 的注册管理机构进行申请(购买);后 3 个字节是获得 OUI 的厂商自行分配的网络接口标识符,只要保证生产出的网络设备没有重复地址即可。在 MAC 地址的第 1 字节中的 b_0 位为 0 时,表示单播地址,此时的帧称为“单播帧”;b_0 位为 1 时为多播地址,此时的帧称为“多播帧”。第 1 字节中的 b_1 位为 0 时,表示全局管理地址,由 IEEE 分配,可确保全球没有相同的 MAC 地址,网络设备厂商向 IEEE 购买的 OUI 都属于全局管理地址;b_1 位为 1 时为本地管理地址,可由网络管理员分配的 MAC 地址。

- 全球单播地址,即 $b_1=0$ 且 $b_0=0$ 时,是由厂商生产网络设备时固化在设备中的。
- 全球多播地址,即 $b_1=0$ 且 $b_0=1$ 时,是交换机和路由器等网络设备所支持的多播地址,用于特定功能。
- 本地单播地址,即 $b_1=1$ 且 $b_0=0$ 时,是由网络管理员分配可以覆盖网络接口的全球单播地址。
- 本地多播地址,即 $b_1=1$ 且 $b_0=1$ 时,是由用户对网卡编程实现,以表明其属于哪些多播组。48 比特是全 1 的 MAC 地址就是本地多播地址的一个特例,此时的帧称为“广播帧”。

MAC 地址的表示方法为:每 4 个比特写成一个十六进制的字符(共 12 个字符),每 2 个字符为一组,共 6 组,各组之间用短线连接,也可以用冒号,或者将每 4 个字符分为一组,共 3 组,各组之间用点连接,如图 5-31 所示,其中,X、Y 分别是十六进制的字符。

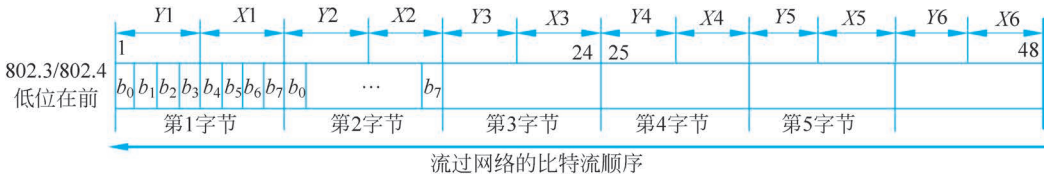


图 5-31　MAC 地址的表示方法

应特别注意，在网络的比特流传输中，每个字节是先要发送高位，再发送低位。

（1）各组之间用短线连接表示 MAC 地址：$X1Y1$-$X2Y2$-$X3Y3$-$X4Y4$-$X5Y5$-$X6Y6$，如 FF-FF-FF-FF-FF-FF。

（2）各组之间用冒号表示 MAC 地址：$X1Y1$:$X2Y2$:$X3Y3$:$X4Y4$:$X5Y5$:$X6Y6$，如 FF:FF:FF:FF:FF:FF。

（3）每 4 个字符分为一组表示 MAC 地址：$X1Y1X2Y2$. $X3Y3X4Y4$. $X5Y5X6Y6$，如 FFFF. FFFF. FFFF。

网卡从网络上每收到一个帧，就可以检查帧首部中的目的地址字段，再按以下情况进行处理：

- 如果目的 MAC 地址与自身网卡上固化的全球单播地址相同，则接受该帧，说明此帧正是发给自己的网络接口的。
- 如果目的 MAC 地址是广播地址，则接受该帧，说明此帧是广播帧，自己的网络接口正处于一个广播域（即一个广播帧最大的传输范围）中。
- 如果目的 MAC 地址是网卡支持的多播帧，则接受该帧，说明自己的网络接口正属于某个多播组。
- 除了上述情况外，通常情况是要丢弃该帧。

由此可以看出，虽然发送数据帧的节点处于广播型共享信道中，但是网卡依据 MAC 地址是否匹配来决定接受或丢弃帧，从而在广播型网络中实现一对一通信。

另外，网卡还可被设置为一种特殊的工作方式，即“混杂方式”（Promiscuous Mode），该方式是指只要是共享介质上传下来的数据帧，混杂模式工作的网卡都会接收，不管帧的目的 MAC 地址是什么。网络维护和管理人员由此可以方便地对局域网上的流量进行监视和分析。然而，混杂方式就像一把“双刃剑”，很多黑客利用这种方式非法获取网络用户的口令。所以，在公共的共享信道场合，使用终端设备时，要注意个人的信息和 MAC 地址泄露的安全问题。

5.3.2 介质争用

由于在广播型共享信道中存在介质争用情况，会形成“冲突/碰撞域”。冲突域就是可能发生介质访问冲突的节点范围。用两种局域网常用到的设备——集线器（Hub）和交换机（Switch），来帮助理解冲突域中数据传输原理。

1. 理解冲突域

首先看集线器，在它中间的背板是以总线方式连接各个端口的，如图 5-32 所示。

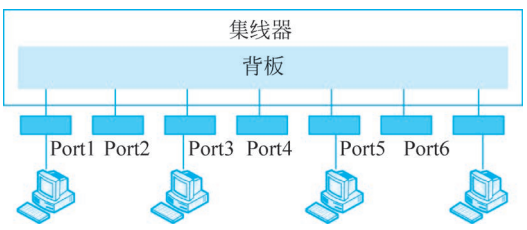


图 5-32 集线器端口与背板的总线连接示意

从图 5-32 中可以看出，在集线器中连接的所有用户都是通过共享一个背板信道进行通信的，连接的用户同时发送数据时，都会发生冲突碰撞，所以说由集线器连接起来的用户就构成一个冲突域。采用集线器构建的局域网络，也称之为“共享式局域网”。

交换机的背板与端口的连接方式不同于集线器，在背板上形似有一个交换矩阵，如图 5-33 所示。

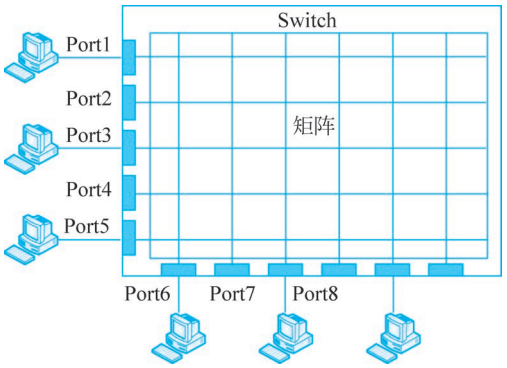


图 5-33　交换机端口与背板交换矩阵连接示意图

通过交换矩阵，就可以实现任何两个端口间的通信都有一条专用的通道，不同通信节点间的通信不存在介质争用现象，所以交换机的各个端口之间不再是冲突域。采用二层交换机构建的局域网络，称为“交换式局域网”。

如果交换机的某个端口连接集线器进行用户扩展，同时连接多个用户进行通信，这时也存在介质访问冲突问题。所以，交换机每一个端口还是一个冲突域。

冲突域是有大小的，就是共享同一介质的节点数量。如在集线器中，端口数量越多，共享同一背板信道的用户越多。如果进行了集线器级联，则更是扩大了冲突域，冲突域越大，发生介质争用的可能性就越高，可能造成的冲突就越大。另一方面，冲突的大小与集线器或交换机背板带宽的大小有关系，即背板带宽越高，数据通过的速率就越高，发生冲突的可能性就越低，冲突也就越小。

2. 介质争用解决办法

介质争用会导致冲突，冲突自然会导致数据的丢失或者变形。但是，不同的网络，对于共享介质、解决争用问题的方法是不一样的。解决介质争用的方法称为介质访问控制，有多种方法，归纳起来如图 5-34 所示。

静态划分信道的方法在第 1 章中讲过，是从多种物理资源上进行划分的方法，其中从时间资源划分的，又分为同步时分复用和异步时分复用，同步时分复用是静态划分的方式，异步时分复用是动态划分的方式。第 3 章讲的电路交换属于 STDM 方式，分组交换属于动态接入控制中的排队方式。

动态接入控制，都是从时间上划分的，即时间上不固定，动态地占用介质的访问时间。它又细分为受控接入方式和随机接入方式。

受控接入方式有一个控制方。其中，轮询方式由控制方根据轮询策略（按循环顺序或优先级顺序）不断重复询问每个站点是否有信息要发送、是否需要接入介质；预约方式是用户事先向控制方提出预约时段，即每一轮数据的发送都是事先安排好的；排队方式

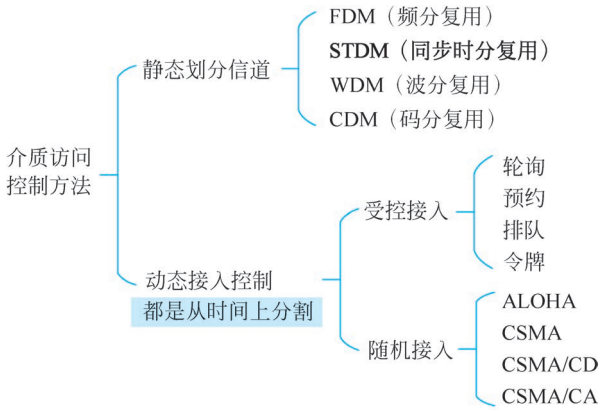


图 5-34　介质访问控制方法

是在访问介质前依次在缓冲器中排队等候；令牌方式是以令牌为控制方，站点间循环传递这个令牌（一种特殊的控制数据），只有收到令牌的站点才能发送数据。

随机接入方法在以太局域网和 WLAN 无线局域网中广泛应用。随机接入协议有几十种，这里只分析一些典型的随机接入协议，即 ALOHA 协议、CSMA 协议、CSMA/CD 协议和 CSMA/CA 协议。

5.3.3　ALOHA 与 CSMA

ALOHA 协议于 1970 年设计，1985 年和 2009 年对 ALOHA 协议进行了改进。虽然这个协议已不再使用，但它激励后来者修改 ALOHA 协议，发明了广泛使用的 CSMA/CD 协议。

ALOHA 协议是一种“想发就发”的传输策略，节点有信息时立即发送，如遭遇碰撞，则随机延迟一段后重发。当有大量节点有很多数据帧要传输时，ALOHA 协议的信道利用率很低，最大效率仅为 $1/(2e)\approx0.18$。为提高 ALOHA 协议的信道利用率，后来改进为时隙 ALOHA，即数据在每个时隙的开始时刻发送，但这要求所有的节点需要保持同步，信道最大效率为 $1/e$（约 0.37），相对于 ALOHA，效率提升了一倍。

在 ALOHA 中，节点数据是否传输完全取决于本节点，不受共享信道中其他节点的影响，核心是“想发就发”，这必然会导致大量冲突，特别是在一个共享信道中有大量要发送数据的节点时。因此，后来改进了 ALOHA 协议，增加了发送前的载波侦听，这就是 CSMA，即 CSMA＝ALOHA＋载波侦听。CSMA 使发送数据的节点在发送前先进行信道侦听，根据信道的忙闲状态确定是否进行数据发送，这里有 3 种策略，如表 5-4 所示。

表 5-4　CSMA 侦听信道后的处理方式

侦听的信道状态	1-坚持 CSMA	非坚持 CSMA	p-坚持 CSMA
信道空闲	马上发送	马上发送	p 概率马上发送 $1-p$ 概率等到下一个时隙再发送
信道忙	继续坚持侦听	等一个随机时间后再侦听	继续侦听，信道空闲再以 p 概率发送

视频

5.3.4 CSMA/CD 介质访问控制原理

CSMA 发送数据时，发生冲突后还是要继续把数据帧发送完，造成了信道浪费。因此继续改进，在 CSMA 增加载波侦听的基础上，再增加数据发送后的数据碰撞检测，以判断发出的数据是否遇到冲突/碰撞，发明了带有冲突检测的载波侦听多路访问（Carrier Sense Multiple Access with Collision Detection，CSMA/CD）协议，即 CSMA/CD = CSMA+冲突检测。

CSMA/CD 广泛应用于以太网中，它是标准以太网、快速以太网和千兆以太网中都采用的介质争用处理协议（万兆以太网由于采用全双工通信，就不再需要这种半双工模式下的技术）。人们已经提出了 CSMA 和 CSMA/CD 的许多变种，这里只考虑 CSMA/CD 中最重要和最基本的一些特性。

1. CSMA/CD 原理

CSMA/CD 在共享传输介质环境的半双工通信模式下使用，其介质访问控制原理包括侦听、发送、检测、冲突处理 4 个处理内容，对应的 4 个步骤是：**先听后发，边发边听，冲突停发，随机延迟后重发**，其过程如图 5-35 所示。

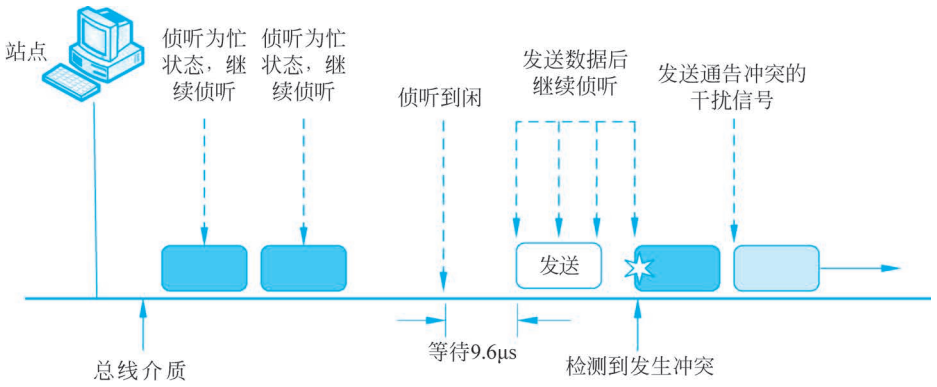


图 5-35 CSMA/CD 介质访问控制示例

（1）当一个站点想要发送数据时，它首先要检测总线介质上是否有其他站点正在传输，即侦听介质是否空闲（即“先听”）。

（2）如果信道忙，则继续侦听，直到侦听到介质状态为空闲；侦听到介质状态为空闲时，站点开始发送数据（即“后发”），如图 5-35 所示，等待 9.6μs（原 10Mb/s 的以太网的帧间间隔，相当于 96 比特的发送时间）后再发送，这样做是为了使刚刚收到数据帧的站点的接收缓存来得及清理，做好接收下一帧的准备。

（3）在发送数据的同时，站点继续侦听总线介质（即“边发边听”），确保在发送过程中没有与其他站点发送的数据发生碰撞。（为什么先听后发还会发生碰撞，其原因详见后面对争用期的分析）。

（4）若检测到有冲突，则立即停止发送数据（即“冲突停发”，也就是说，一旦出现冲突，立即停发），同时发送一个用于加强冲突的阻塞（JAM）信号，以便使网络上其他所有

站点都知道网上发生了冲突，不再接收原来的帧。

(5) 本站点然后等待一个随机时间(该时间根据“二进制指数退避算法”来计算)，且在信道为空闲时，再重新发送数据(即“随机延迟后重发”)。

2. 争用期

在 CSMA/CD 协议中，监听信道为空闲后再发送数据，为什么在发送数据的过程中仍然可能会遭遇碰撞呢？这是因为物理信号在传播时存在传播时延，两个站点同时检测到介质空闲，然后开始传输数据，由于传播时延，各方都要等到其他站点的数据传播到自己这里时才能检测到双方的数据实际在半路上发生了碰撞。

那么发送数据帧的站点，最迟要经过多长时间，才能检测出自己发送的帧与其他站点发送的帧产生了碰撞呢？

假设两个站点之间的单程端到端传播时延为 τ，当两端同时发送数据时，则在 $\tau/2$ 时刻相遇，在 τ 时刻双方都检测到对端的信号，知道发生冲突，如图 5-36(a)所示。当两端不同时发送数据时，如图 5-36(b)所示，在 A 端信号发出 $\tau-m$ 时刻(B 端信号发出 m 时刻)相遇，则 A 在 $2\times(\tau-m)$时刻知道发生冲突，B 在 $2m$ 时刻知道发生冲突。极端情况是 m 趋于零，即最长在 2τ 时刻(也是端到端往返时间)，双方都能检测到对端的信号，知道发生碰撞。因此，2τ 被称为“争用期”或者“碰撞窗口”，只要是经过争用期 2τ 这段时间后还没有检测到碰撞，就可以肯定这次发送不会发生碰撞。在争用期时间内，是可能发生碰撞的，具体是多长时间，取决于两个站点之间的距离。显然，总线的长度越长(τ 越大)，网络中站点越多，发生碰撞的概率就越大。

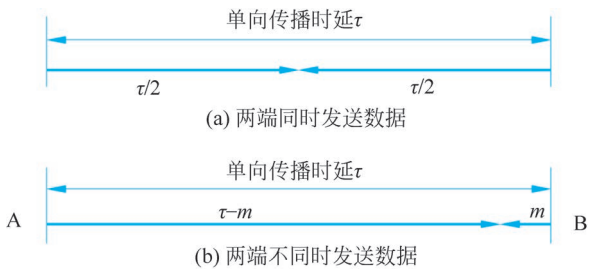


图 5-36 传播时延与争用期

3. 最小帧长

前面分析出了争用期 2τ，但具体的传播时延不便于测量，工程上需要简化。于是把 2τ 转换为在 2τ 内能传输的数据量，对于设计 CSMA/CD 时的 10Mb/s 共享总线以太网，最长距离为 5km，传播时延为 25μs，争用期 2τ 为 50μs，对 10Mb/s 的传输速率，可传输 500 比特(62.5 字节)，取靠近 2^n 的字节数，即 64 字节(反推回去，相当于争用期 2τ 取 51.2μs)。

这样，某个站点在发送帧后，如果帧的前 64 字节未遭遇碰撞，那么帧的后部分也不会发生碰撞；反之，发送站点一旦检测到碰撞就立即终止帧的发送，所以此时已发送的数据量一定是小于 64 字节。因此，接收站点收到长度小于 64 字节的帧，就可以判定这是一个遭遇了碰撞而异常终止的无效帧，便将其丢弃。因此以太网规定，以太网 PDU 帧

(包含帧头)的最小帧长为 64 字节。

4. 最大帧长

从效率角度来说,数据帧的载荷长度希望越大越好,这样可以提高帧的传输效率。但是,如果不限制数据载荷的长度上限,使帧的总长度太长,则会造成广播型共享以太网中的其他主机总是拿不到共享信道的使用权,并且还可能导致接收站点的接收缓冲区无法装下该帧而产生溢出。因此,以太网的 MAC 数据帧的最大长度规定为 1518 字节,其中首部和尾部开销是 18 字节,数据载荷的最大长度是 1500 字节。

5. 退避算法

广播型共享以太网中各站点在检测到发送数据帧遭遇碰撞后,采用"二进制指数退避算法"来选择退避的随机时间。先想一想这个随机时间与哪些因素有关,如何确定这个时间。

假设随机延迟时间为 T,则 $T=\text{random}[0,X]$,X 大,则 T 的可选值多;X 小,则 T 的可选值少。若有 4 个站点,每个站点平均发 2 个包,需要 10 次发包机会,X 取 8,肯定还会出现很多次碰撞,若 X 取 80,则碰撞概率很低,但又出现很多浪费的时间。因此,希望 X 的取值与网络负载(站点数和发包的量)一致。负载小时,X 取值小;负载大时,X 取值大。那如何判断负载量大小呢?

工程师们巧妙地利用了负载量与冲突的关系,冲突次数多,说明负载量大;反之则小。因此该问题转化为根据冲突次数来判断负载量大小。

解决了基本思路,再来具体看看二进制指数退避算法的算法过程。

(1) 确定基本退避时间为争用期 2τ(类似时隙 ALOHA,数据只在每个时隙的开始时刻发送)。

(2) 随机延迟时间 $T=\text{random}[0,1,2,\cdots,(2^k-1)]\times 2\tau$。

参数 k 与冲突次数有关,规定 k 不能超过 10,$k=\text{Min}[\text{重传次数},10]$。在重传次数大于 10,小于 16 时,$k$ 不再增大,一直取值为 10。也就是重传次数大于 10 以后,都是从 $0\sim 2^{10}-1$ 个 2τ 中随机选择一个作为延迟时间。当冲突次数超过 16 次后,就表明同时发送帧的站点太多,以至于连续产生碰撞,因此只能丢弃并放弃重传而报告上层处理。

6. CSMA/CD 的亮点与不足

CSMA/CD 的最大亮点是它在侦听到有冲突发生时,可以立即中止数据帧的发送,快速地终止被破坏的帧,从而节省整个系统的时间和信道资源;并且发送一个阻塞信号,以强化冲突,使其他站点更容易检测到有冲突的发送,不再同时发送数据了,可避免更多的帧发送冲突。

使用 CSMA/CD 的缺点是,如果一个站点需要发送大量的数据时,这时可能出现一个站点长时间控制整个局域网的情况,当网络中通信数据量较大时,这种方式仍然有可能引起较大的冲突。

在共享型广播网络中,站点使用 CSMA/CD 协议只能是尽量避免碰撞冲突,并在出现碰撞时做出退避后重发的处理,但无论如何也不能完全避免冲突/碰撞。

5.3.5 CSMA/CA 介质访问控制原理

虽然 CSMA/CD 协议成功地应用于有线连接的局域网，但无线局域网不能简单地搬用 CSMA/CD 协议。针对无线局域网传输特点需制定出适合无线网络共享信道的介质访问控制协议。在无线局域网中，由于无线通信本身的特点，存在隐藏站问题和暴露站问题（简单说，隐蔽站问题是检测不到碰撞，但会产生碰撞；暴露站问题是，检测到了碰撞，但不影响自己与其他站的通信），而碰撞检测无法解决这两类问题。在无线局域网发送中，一旦发生碰撞，会导致整个信道资源的严重浪费，因此，无线局域网应当尽量减少碰撞的发生。

为此，IEEE 802.11 局域网使用带有冲突避免的载波侦听多路访问（Carrier Sense Multiple Access with Collision Avoid，CSMA/CA）协议，目标是尽量减少碰撞发生的概率，同时还采取以下措施：

- IEEE 802.11 局域网在使用 CSMA/CA 的同时，还使用停止-等待方案，是因为无线信道的通信质量不如有线信道，因此无线站点每次通过无线局域网发送完一帧后，就要等收到对方的确认帧后才能继续发送下一帧，通过数据链路层确认来实现可靠传输。
- 采用虚拟载波监听机制，目的是让源站把它要占用信道的时间（包括目的站发回确认帧所需要的时间）及时通知给所有其他站，以便使其他站在这一段时间都停止发送帧。
- 允许源站对信道进行预约。即源站在发送数据前先发送一个短的发送请求帧（Request To Send，RTS）和目的站响应一个允许发送帧（Clear To Send，CTS），以确保源站和目的站之间的通信不会收到干扰。

1. IEEE 802.11 的 MAC 层

无线局域网标准 IEEE 802.11 的 MAC 层是包括两个子层，即：分布协调功能（Distributed Coordination Function，DCF）和点协调功能（Point Coordination Function，PCF），如图 5-37 所示。它通过协调功能（DCF）来确定基本服务集 BSS（是无线局域网最小构件）中的移动站，在什么时间能发送数据或接收数据。

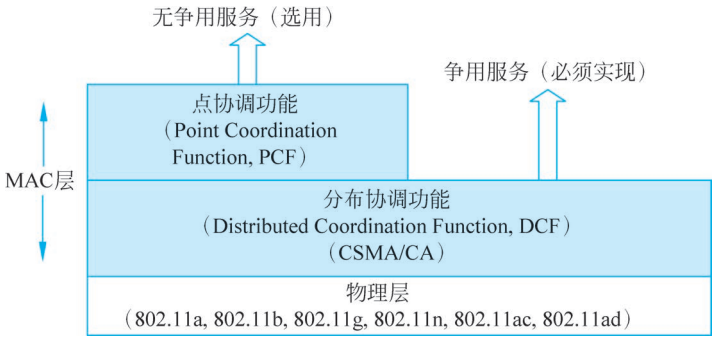


图 5-37 IEEE 802.11 的 MAC 层

- DCF 子层使用一种简单的 CSMA 算法，但不包括冲突检测功能。
- PCF 是一个集中式 MAC 算法，该算法用于提供无争用服务。PCF 建立在 DCF 之上，并利用 DCF 特性来保证它的用户接入，它是在 DCF 之外实现的一个可供选择的访问方式。

为了避免碰撞，IEEE 802.11 还规定，所有的站点在完成发送后，必须等待一段很短的时间并继续监听，才能发送下一帧。为尽可能地避免冲突发生，IEEE 802.11 局域网还要求，各站点不能不间断地一直发送。每发完一帧都得经过一个特定的时长才能继续发送下一帧(使用停止-等待方案)，并把这个特定的时长称作帧间间隔(Inter Frame Space，IFS)。

帧间间隔的长短取决于站点要发送的帧的类型，采用不同优先级别的帧(低优先级的帧需要等待较长的时间，高优先级的帧需要等待的时间较短，这样高优先级的帧可优先获得发送权)来减少发生碰撞的机会。至于各种帧间间隔的具体长度，则取决于所使用的物理层特性。

IEEE 802.11 共规定了 4 种长度的帧间间隔，分别是 SIFS(Short Inter Frame Space)、PIFS(PCF Inter Frame Space)、DIFS(DCF Inter Frame Space)和 EIFS(Extended Inter Frame Space)，它们之间的长度关系如图 5-38 所示。

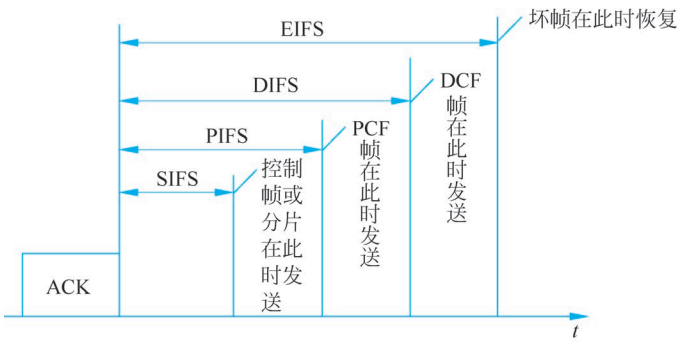


图 5-38　IEEE 802.11 4 种帧间间隔的长度及关系

先来看看以下各种帧间间隔的作用：

(1) SIFS(短帧间间隔)，是最短的帧间间隔，用来分隔属于一次对话的各帧。在这段时间内，一个站应当能够从发送方式切换到接收方式。使用 SIFS 的帧类型有较多种(如 ACK 帧、CTS 帧等)。例如，用于请求发送 RTS 帧和允许发送 CTS 帧之间、允许发送 CTS 帧和 DATA 帧之间、DATA 帧和确认帧 ACK 之间。

(2) DIFS(分布协调功能帧间间隔)，长于 SIFS。在 DCF 方式中，DIFS 是用来发送数据帧和控制管理帧的。

(3) PIFS(点协调功能帧间间隔)比 SIFS 长一个时隙时间(slot time)。它主要用于在刚开始使用 PCF 功能时，使站点能够尽快得到发送权，但是 PIFS 只能够工作于 PCF 模式。

(4) EIFS(扩展的帧间间隔)。EIFS 是最长的 IFS，主要在站点接收到坏帧时使用。

在 CSMA/CA 协议中，根据本站下一帧的功能和类型来确定 IFS 的类型。高优先级

的数据帧享有优先发送的权利，因此要选择长度较短的 IFS。低优先级的数据帧就必须要选择长度相对长的 IFS。这样在低优先级帧间隔时间未结束时，高优先级帧已被发送到信道，且信道变为忙状态，所以低优先级帧只好退避等待。这样就可以尽量避免碰撞的发生。所以，站点当前所使用的 IFS 的类型就直接表明了即将要发送的数据帧的优先级别，即 SIFS＞PIFS＞DIFS＞EIFS。若当前等待时间为 SIFS，则说明即将发送的数据帧优先级别最高，享有优先发送的权利；若当前等待时间为 EIFS，则说明即将发送的数据帧具有最低的优先级别，必须等其他站点的数据帧发送完成后才能发送。

2. CSMA/CA 工作原理

CSMA/CA 协议的工作原理如图 5-39 所示。

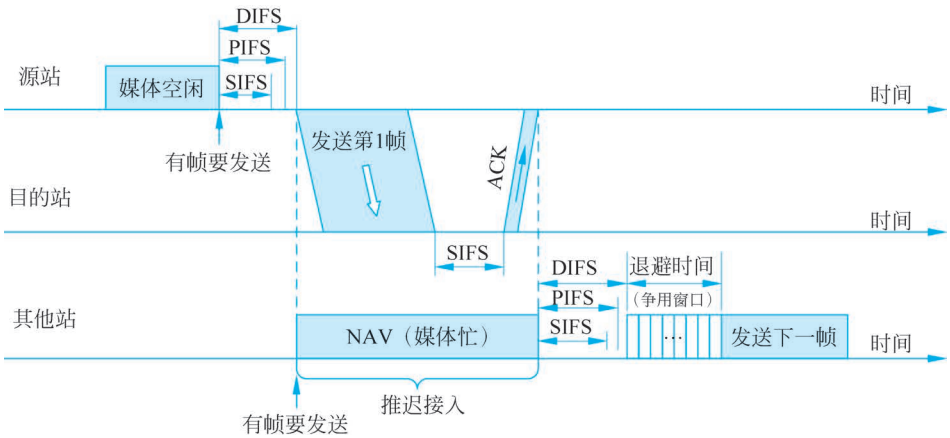


图 5-39　CSMA/CA 协议的工作原理

首先，要发送数据的站要先检测信道。采用的方式可能是下面的方式之一：

(1) 在 IEEE 802.11 标准中规定了在物理层的空中接口进行物理层的载波监听(Physical Carrier Sense)，即基于物理层的载波监听信道忙否。通过收到的相对信号强度是否超过一定的门限数值就可判断是否有其他的移动站在信道上发送数据。

(2) 由 MAC 层的虚拟载波监听(Virtual Carrier Sense)机制指出信道是否繁忙。即当某个站检测到正在信道中传送的帧首部的“持续时间”字段时，就调整自己的网络分配向量(Network Allocation Vector，NAV)，即指出必须经过多长时间才能完成帧的这次传输，才能使信道转入空闲状态。

当源站发送它的第 1 个 MAC 帧时，若检测到信道空闲，则在等待一段时间 DIFS 后就可发送。等待 DIFS 间隔是考虑到其他可能的站有更高优先级的帧要发送。现在假设没有其他高优先级的帧(不采用 SIFS 或 PIFS)要发送，源站就发送自己的数据帧。

若源站发送了自己的数据帧，目的站也正确收到此帧，则经过短帧间间隔 SIFS 后，向源站发送确认帧 ACK。若源站在规定时间内没有收到确认帧 ACK，就必须重传此帧，直到收到确认帧为止，或者经过若干次的重传失败后放弃发送。

其他站通过虚拟载波监听调整自己的网络分配向量，即信道被占用的时长。在 NAV 时间段内，若其他站也有数据帧要发送，则必须推迟发送。在 NAV 时间段结束后，

在经过一个 DIFS，然后还要退避一段随机时间后才能发送帧。

无线局域网 IEEE 802.11 采用的停止-等待方案，是一种可靠传输方式，CSMA/CA 流程如图 5-40 和图 5-41 所示，其中图 5-40 是 CSMA/CA 无信道预约的 CSMA/CA 工作流程，图 5-41 是有信道预约的 CSMA/CA 工作流程。

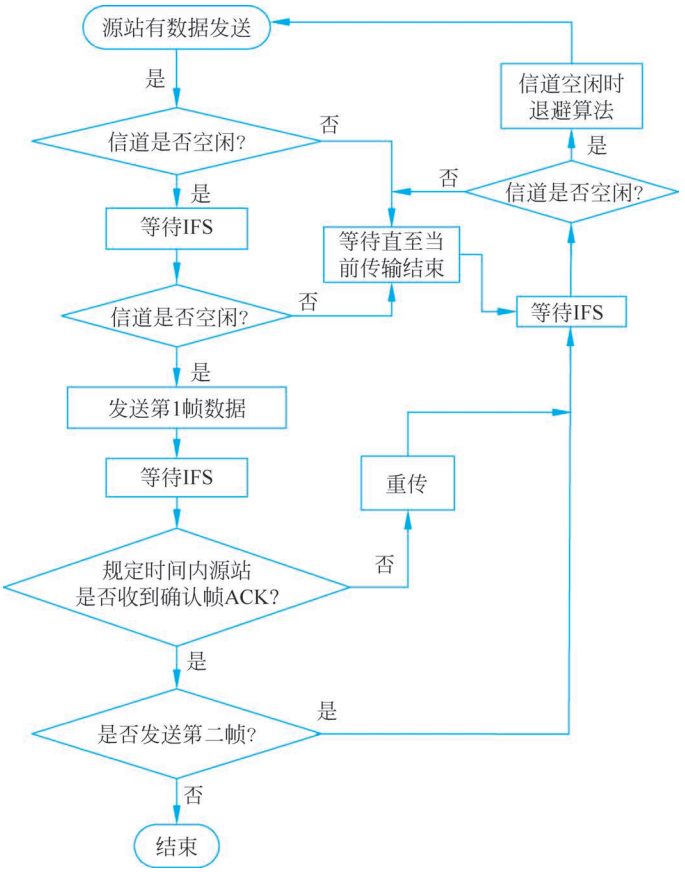


图 5-40　无信道预约的 CSMA/CA 流程

需要说明的是：

（1）源站的有数据不是成功发送完上一数据帧之后立即连续发送的数据或者重传数据，是指源站的第 1 帧数据。

（2）图 5-40 和图 5-41 中的 IFS 是根据本站下一帧的功能和类型来确定其具体类型的。

（3）应用退避算法是因为检测到信道忙，也就要暂停退避计时器。只有信道闲时，退避计时器才能进行倒计时，当退避计时器时间减少到零，并且信道空闲时，源站才能发送整个帧，并等待确认（请思考该退避算法是否区别于 CSMA/CD 的退避算法）。

（4）若规定时间内未收到确认 ACK，会再次使用 CSMA/CA 协议争用接入信道，进行重传确认，但是若经过若干次的重传失败后就会放弃发送。

（5）存在不使用退避算法的情况是：检测到信道空闲，并且这个数据帧是站点想发

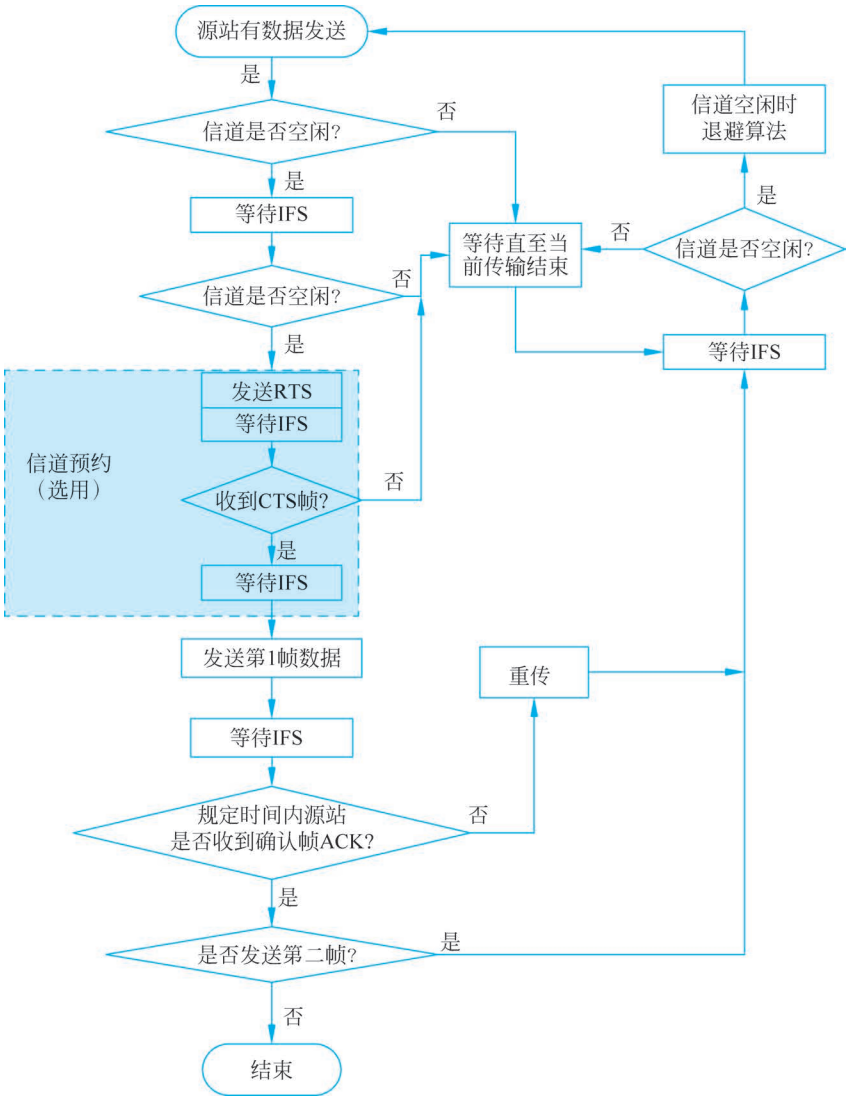


图 5-41　有信道预约的 CSMA/CA 流程

送的第 1 个数据帧。

（6）在如下情况必须使用退避算法。

- 在发送第一个帧之前检测到信道处于忙的状态；
- 每一次的数据帧重传；
- 每一次的成功发送后要再发送一帧。

（7）虽然使用 RTS 帧和 CTS 帧进行信道预约会带来开销，但 RTS 帧和 CTS 帧都很短，发生碰撞的概率以及产生的开销都很小。在发送传输时延长的数据帧前，采用很小的代价对信道进行预约是值得的。对 RTS 帧和 CTS 帧的使用在 IEEE 802.11 局域网中是可选择性的。

5.4 局域网标准及以太网帧格式

在局域网中，绝大多数标准都是由 IEEE(电气和电子工程师协会)制定的 IEEE 802 系列标准。在网络中曾经应用最广的是以太局域网标准 IEEE 802.3。后来 IEEE 802 系列标准被 ISO 接受为正式的国际标准，标准名为 ISO 8802。

视频

5.4.1 IEEE 802 系列局域网标准

IEEE 802 协议包含多种子协议，把这些协议汇集在一起即 IEEE 802 协议集。IEEE 802 标准委员会，致力于研究局域网和城域网的物理层和 MAC 层中的服务和协议，对应 OSI 网络参考模型的物理层和数据链路层。IEEE 802 系列的常见标准如表 5-5 所示。

表 5-5 IEEE 802 系列的常见标准

标 准 名 称	作　　用
IEEE 802.1	局域网概述，体系结构，网络管理和网络互连
IEEE 802.2	逻辑链路控制 LLC
IEEE 802.3	CSMA/CD 总线介质访问控制子层与物理层规范
IEEE 802.4	令牌总线(Token Bus)介质访问控制子层与物理规范
IEEE 802.5	令牌环(Token Ring)介质访问控制子层与物理规范
IEEE 802.6	城域网(MAN)介质访问控制子层与物理规范
IEEE 802.10	局域网安全性规范
IEEE 802.11	无线局域网访问控制方法和物理层规范
IEEE 802.14	协调混合光纤同轴(HFC)网络的前端和用户站点间数据通信的协议
IEEE 802.15	无线个人网技术标准，其代表技术是蓝牙(Bluetooth)

视频

5.4.2 以太网帧格式

以太网中，MAC 帧是在数据链路层实体间交换的协议数据单元(PDU)。不同的组织机构定义了多种不同的 MAC 帧，IEEE 802.3 是其中一种，但使用最多的是 DIX Ethernet Ⅱ标准(也称 Ethernet V2 标准)。因此这里只介绍使用得最多的 Ethernet Ⅱ的 MAC 帧格式，帧格式如图 5-42 所示。

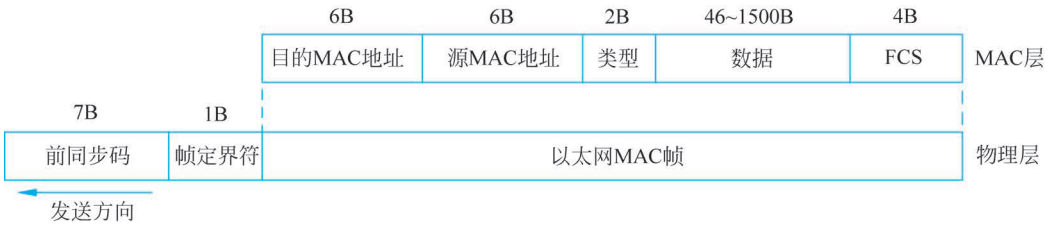


图 5-42 Ethernet Ⅱ MAC 帧格式

1. 目的 MAC 地址/源 MAC 地址

目的 MAC 地址(Destination Address，DA)和源 MAC 地址(Source Address，SA)字

段各占 6 字节，分别用于标识接收站点的 MAC 地址和发送站点的 MAC 地址，地址可以是单播 MAC 地址，也可以是组播地址或者广播 MAC 地址。

2. 类型(Type)

该字段占 2 字节，指出帧中数据字段中的数据类型，数值总大于 1536(即 0x0600)(数值≤1500 则为 IEEE 802.3 以太网帧的长度字段)。类型字段用来表明数据字段中的内容是由上一层的哪个协议封装的，以便将收到的 MAC 帧的数据字段中的内容交给上一层的相关协议，一些典型的类型字段数值及说明如表 5-6 所示。

表 5-6 典型的类型字段数值及说明

类型编号(十六进制)	协　　议
0000～05DC	IEEE 802.3 长度字段
0800	IPv4 协议
0806	地址解析协议(Address Resolution Protocol，ARP)
8037	IPX(Novell NetWare)
809B	AppleTalk(EtherTalk)
8100	IEEE 802.1Q 用户 VLAN
814C	基于以太网的 SNMP(SNMP over Ethernet)
8191	NETBIOS/NetBEUI
86DD	IPv6 协议
8847～8848	多协议标签交换(Multi-Protocol Lable Switching，MPLS)
8863	PPPoE 发现阶段
8864	PPPoE 会话阶段
9000	Loopback(配置测试协议)

3. 数据(Data)

该字段装载上层来的 PDU，其长度范围为 46～1500 字节。如果数据不够 46 字节，则需要在数据字段后面加上填充字段(46 是由最小帧长 64 字节减去 MAC 帧首尾 18 字节得到)。

4. 帧校验序列(Frame Check Sequence，FCS)

该字段占 4 字节，是 32 位的循环冗余校验(CRC)值。

至此，前面讲的数据链路层最基础的帧定界功能还没有，以太网 MAC 层巧妙地借用了物理层做时钟同步的前导码来实现了帧定界功能，相当于把帧定界放到物理层去做了(也可以说，是物理层提前做好了各帧的分界，数据链路层就不用做这项工作了)。

前导码分成两部分：前同步码和帧开始定界符。

(1) 前同步码(Preamble)。该字段占 7 字节，由 1 和 0 交替构成，用于接收方调整其时钟频率，使它和发送方的时钟同步，实现收发双方的位同步，这是确保正确接收比特流的基础。(详见 1.5.3 节)

(2) 帧开始定界符(Start-of-Frame Delimiter，SFD)。该字段占 1 字节，前 6 位也是 1 和 0 交替构成，最后两位是连续的 1，即 10101011，表示一个帧的开始。即接收方检测到连续两位 1(即读到帧起始定界符字段 SFD 最末两位)时，便将后续的信息递交给

MAC 子层。

至此，有了帧定界符，确定了帧的开始位置，MAC 帧中没有长度字段，那帧的尾部位置如何确定呢？

以太网也把它放到物理层去实现了，规定各帧之间插入帧间间隔时间，如图 5-43 所示。以太网标准中规定最小帧间隙是 12 字节，其数据为全 1。所以，MAC 帧不需要帧结束符或长度字段。

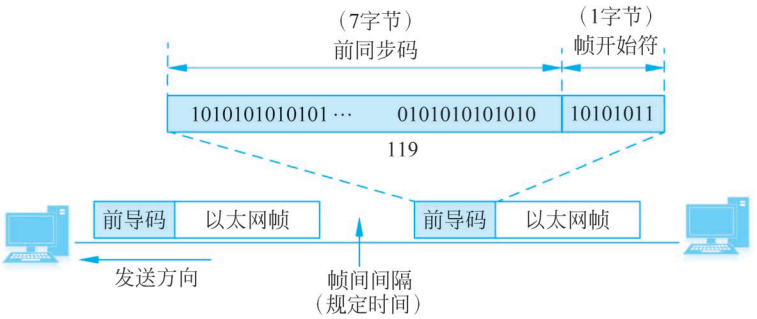


图 5-43　以太网帧的前导码

5.4.3　以太网的分类

以太网因通信介质的不同和通信速率的差异，衍生出了众多不同的以太网类型。IEEE 802.3 系列标准中发布的就有标准以太网（10M 以太网）、快速以太网（100M 以太网）、千兆以太网和万兆以太网规范，具体如表 5-7 所示。所有的以太网都有一个规范的命名，都能表示其相关的特殊含义，命名规范为：<数据传输速率（Mb/s）><信号传输模式><最大段长度（百米）>/<-介质类型>，如 10Base5、10Broad36、10Base-T、10Base-F 等。

表 5-7　以太网主要分类

以太网种类	电缆最大长度	电 缆 种 类	标　　准
10Base5	500m	粗同轴电缆，75Ω	IEEE 802.3
10Base2	185m	细同轴电缆，50Ω	IEEE 802.3
10Base-T	100m	双绞线（3 类～5 类无屏蔽双绞线）	IEEE 802.3
10Base-F	1000m	多模光纤（Multi Mode Fiber，MMF）	IEEE 802.3
100Base-TX	100m	双绞线（5 类无屏蔽双绞线/屏蔽双绞线）	IEEE 802.3u
100Base-FX	2000m	多模光纤	IEEE 802.3u
100Base-T4	100m	双绞线（3 类～5 类无屏蔽双绞线）	IEEE 802.3u
1000Base-CX	25m	屏蔽铜线	IEEE 802.3z
1000Base-SX	550m	多模光纤	IEEE 802.3z
1000Base-LX	550m/5000m	多模光纤/单模光纤	IEEE 802.3z
1000Base-T	100m	双绞线（5 类/5e 类无屏蔽双绞线）	IEEE 802.3z
10GBase-SR	300m	多模光纤	IEEE 802.3ae
10GBase-LR	10km	单模光纤	IEEE 802.3ae
10GBase-ER	40km	单模光纤	IEEE 802.3ae

续表

以太网种类	电缆最大长度	电缆种类	标准
10GBase-T	100m	双绞线(6a类无屏蔽/铝箔屏蔽双绞线)	IEEE 802.3an
10GBase-CX4	15m	双芯同轴电缆	IEEE 802.3ak
40GBase-SR4	400m	多模光纤	IEEE 802.3ba
100GBase-LR4	10km	单模光纤(采用4个波长复用,每个波长25Gb/s)	IEEE 802.3ba
100GBase-ER4	40km	并行多模光纤	IEEE 802.3ba

- 10Base5中的10表示10Mb/s数据传输速率,Base表示信号采用基带传输方式,5表示单段介质的最大传输长度为500m。
- 10Broad36表示10Mb/s数据传输速率,采用宽带系统(如有线广播方式、有线电视网络)传输,36表示端到端的最大距离为3600m。
- 10Base-T表示10Mb/s数据传输速率,采用基带传输方式,T表示传输介质为双绞线(Twisted Pairewire)。
- 10Base-F:表示10Mb/s数据传输速率,采用基带传输方式,F表示传输介质为光纤(Fiber)。

5.5 构建交换式局域网

构建局域网可以采用基于物理层和数据链路层的设备,物理层设备只是采用集线器或者中继器等构建的局域网,可以称为“共享式局域网”。带有数据链路层功能的常见设备有网卡、网桥和二层交换机,通过网桥和交换机构建的局域网通常是“交换式局域网”。

5.5.1 网卡

网卡是网络接口卡(Network Interface Card,NIC)的简称,也叫网络适配器,是设备与网络之间的桥梁,是网络中最基础的网络设备,它安装在计算机或网络设备中。

网卡工作在物理层和数据链路层,由物理层隔离变压器、物理层处理芯片、处理器、存储器、输入/输出缓存器等组成,每张网卡的MAC地址就保存在网卡的存储器中。

网卡的基本功能包括:

(1) 物理层功能——完成数据传送与接收所需的数/模转换,串/并转换、同步时钟提取、数据编码(曼彻斯特编码与译码)等,并向数据链路层设备提供标准接口。

(2) 链路管理——通过CSMA/CD或CSMA/CA协议规定的规则来实现。

(3) 数据的封装与解封——发送时将上一层传来的数据加上首部和尾部,成为以太网的MAC帧;接收时将以太网的MAC帧剥去首部和尾部,送交至上一层。

根据不同的分类依据,分成多种类型的网卡。

(1) 按网卡所支持的传输介质划分,分为有线网卡和无线网卡。有线网卡分为双绞线接口网卡和光纤接口网卡,前者连接RJ-45网线,后者连接光缆。无线网卡连接空中无线电波。

（2）按与主机相连的接口类型划分，分为 PCI 总线接口、PCMCIA 接口、PCI-X 接口、PCI-E 接口和 USB 接口的网卡。

（3）按支持的网络标准划分，分为 10Mbps/100Mbps 自适应的双绞线快速以太网卡、10Mbps/100Mbps/1000Mbps 双绞线千兆以太网卡、纯 1000Mbps 的光纤千兆以太网卡等。


视频

5.5.2 交换机

交换机（Switch）可以理解成是集线器和网桥的综合升级换代产品，它既具有集线器的集中连接功能，又具有网桥的数据交换功能；是带有“交换”功能的集线器，或者说是多端口网桥。

网桥（Bridge）是早期的只有少量端口（2～4 个端口）的基于数据链路层的网络设备，可用来连接处于不同物理位置网段的计算机（如图 5-44 所示），并具备隔离冲突域特性，比当时的集线器（Hub）性能更好。网桥与传统的二层以太网交换机有非常相似的工作原理。将在 5.5.3 节统一介绍二层交换的基本原理。

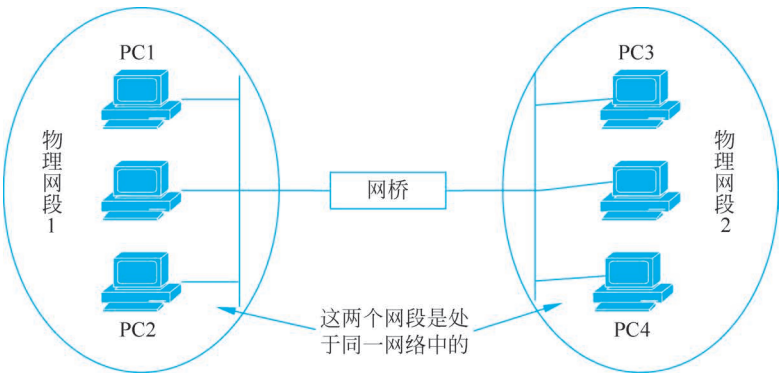


图 5-44 网桥连接两个物理网段

特别说明：图 5-44 中的“物理网段”指的是设备 IP 地址属于同一网络地址段，只是位于不同地理位置的主机。

由于交换机发展速度相当快，其应用向两个不同的方向发展，在外观上也有很大区别。小的桌面交换机和集线器类似，而大的则采用模块化结构（即内部各插槽中的通信线卡可以根据需要进行调整），机箱较大，如图 5-45 所示。

(a) 小型固定端口交换机　　(b) 模块化交换机

图 5-45 小型固定端口交换机与模块化交换机

1. 交换机与集线器和网桥的比较

二层交换机与集线器的区别主要如表 5-8 所示。

表 5-8 二层交换机与集线器的区别

区　别	交　换　机	集　线　器
在 OSI 中的工作层次	二层及二层以上	物理层
数据传输方式	有目的的、数据是可以针对目的节点发送。只有在 MAC 地址表中找不到目的地址的情况下会使用洪泛广播式传输	多次复制的洪泛广播式传输
背板信道占用方式	每个端口的收、发都是独享的背板信道带宽	共享背板中的一条信道带宽
数据通信方式	可以进行全双工数据传输	只采用半双工方式进行传输

与网桥相比，二层交换机的主要特性体现在如下方面：

(1) 具有更多的交换端口。

(2) 大多数主机直接连在交换机端口上，数据转发效率更高。使用网桥时，因为网桥端口少，为扩展网桥端口使用，网桥端口下面还要先连接集线器，再通过集线器连接主机。而交换机端口是与主机直接相连，使交换机具有更强的数据转发能力。因为交换机转发数据是基于自学习而来的地址转发表，地址转发表中基本都是主机 MAC 地址与交换机端口一一对应的映射，一对一的映射查找效率会高于一对多(比如，网桥的端口连接集线器，集线器再连接主机)的情况。

综上所述，交换机的数据通信效率要远高于集线器和网桥。从当今网络应用情况来看，交换机已经完全取代了集线器和网桥曾经在网络中的位置和作用。但集线器属于共享总线式网络，其思想在无线通信场景中依然有用。

2. 交换机的分类

交换机是一种应用非常广泛的网络设备，目前有各种不同的交换机类型。交换机的分类方式也是多种多样，常用的分类方式有如下。

1) 按照交换机功能划分

根据交换机处理数据包的能力可分为二层交换机、三层交换机和高层交换机。

二层交换机是有类似于网桥的功能，只是二层交换机相对于网桥，它具备更强的性能和更多的端口；带有 IP 路由功能的交换机就可称为“三层交换机”或者“路由交换机”。高层交换机是指能够处理 OSI 模型中 4～7 层的数据，如多台服务器前可加装的负载均衡器、有“带宽控制”的交换机、广域网加速器、可以针对特殊应用访问加速以及具有防火墙功能的交换机等。

2) 根据网络类型划分

根据交换机所应用的局域网类型可以将局域网交换机分为标准以太网交换机(10Mb/s 传输速率)、快速以太网交换机(100Mb/s 传输速率)、千兆以太网交换机(1000Mb/s 传输速率)、万兆以太网交换机(10000Mb/s 传输速率)等。

3) 按交换机结构划分

按交换机结构划分，交换机可分为固定端口交换机和模块化交换机。

固定端口就是它所带有的端口是固定的。例如,8 端口、16 端口、24 端口的。这种固定端口的交换机基本上都属于较低档次的。

模块化交换机是交换机上除了有部分固定的端口外,还可通过插入扩展模块来扩展端口数量以及所支持的传输介质、网络协议、业务类型等。

4) 按照用途分类

根据规划设计规模化网络,网络可划分为核心层、汇聚层和接入层。根据交换机在网络中所处的位置和用途可分为核心交换机(处于核心层)、汇聚交换机(处于汇聚层)、接入交换机/边缘交换机(处于接入层)。

5) 按是否支持网络管理功能划分

按交换机是否支持网络管理功能可划分为网管型和非网管型两大类。

网管型交换机可以通过交换机的控制端口(Console 口)或 Web 界面进行配置和管理。非网管型交换机则不能进行任何配置与管理,仅按照出厂的默认设置进行工作。


视频

5.5.3 二层交换原理

从实际应用上看,网桥或者二层交换机的每一个端口都可以连接一个物理网段(如图 5-44 所示),但是它们所连接的主机都是处在同一网络(子网)中。比如用交换机连接位于不同办公室或者不同楼层的主机,则可以通过使用同一网络地址的两个或多个小的物理网段,组成一个可以统一管理的大规模局域网。通常所说的“桥接”也就是网桥的作用,即连接的是同一网络(子网)中的两个物理网段。

二层交换机与网桥一样,具备自学习能力,建立自身的数据转发的依据——MAC 地址表(地址转发表),实现对发送数据的过滤转发,即二层交换机直接根据数据帧中的目的 MAC 地址,进行 MAC 地址表查表,根据转发表情况把数据发送到相应端口上。

1. MAC 地址与端口映射建立 MAC 地址表

MAC 地址表保存的是主机 MAC 地址与所连接的端口的映射,即列出哪个 MAC 地址连接的是哪个具体的物理端口。这个映射表项可以由管理员手动绑定(静态配置),也可以由交换机自动学习得到(动态获取),针对多播还可以通过各种多播协议,如 IGMP 嗅探、GMRP 协议等方式获取。

但要注意的是,MAC 地址表是存于交换机缓存中的,但交换机缓存空间是有限的,所以可以存储的 MAC 地址和端口映射表项也是有限的。当网络比较大时,交换机中的缓存空间就不能保存网络中所有节点 MAC 地址与交换机端口的映射关系。

在网络中,具有 MAC 地址的设备,是如何与连接这些设备的交换机端口建立映射,形成交换机中 MAC 地址表的?

实际上,交换机在进行对数据转发的同时,还有一个“MAC 地址与端口映射关系”的自学习过程。

(1) 交换机在最初接入网络中时,是不存在 MAC 地址与端口映射关系表项的(即 MAC 地址表为空表)。

(2) 当交换机在接收到数据帧时,会提取数据帧中的源 MAC 地址,并查询 MAC 地

址表，检查在自身的 MAC 地址表中是否有针对该源 MAC 地址的转发项？

- 如果没有，则把该源 MAC 地址和收到该源 MAC 地址的端口绑定起来（即建立映射关系），插入 MAC 地址表项中，并启动一个该项的老化定时器。这样当再接收到一个发送到该 MAC 地址的数据帧时，就不需要向所有端口广播转发，而仅向这一个对应的映射端口发送。
- 如果有，则会更新 MAC 地址表中该项的老化定时器时间，即恢复定时器初始值。

（3）当交换机在接收到数据帧时，会针对数据帧中的目的 MAC 地址，并在 MAC 地址表中进行查表。当查表没有结果时，会向交换机上其他所有端口进行广播；当接收到该帧的节点应答该帧后，便可获得数据帧中对应的目的 MAC 地址所连接的端口（即会使用上述第（2）点方法），这时交换机会把该 MAC 地址与所连端口的对应表项插入到 MAC 地址表中。

（4）如 MAC 地址表中各表项的老化定时器在限定时间内没有收到有相关 MAC 地址与端口映射关系的数据帧而进行老化时间的更新，则该 MAC 地址与端口映射关系的表项则会在 MAC 地址表中失效并被删除。

特别说明：交换机 MAC 地址表项老化定时器时间长短是可以通过操作指令进行设置的。在进行交换机组网应用时，大家可以根据情况在实践应用时尝试进行设置操作，并思考一下，当设置 MAC 地址表项的老化定时时间过长或者过短，对网络或者交换机会有什么影响？

上述说明的 MAC 地址表没有考虑 VLAN（虚拟局域网）的情形。现在的交换机一般都支持 VLAN 技术应用，所以 MAC 地址表就有了变化，由原来的主要两项对应关系（MAC 地址、交换机端口），变成主要的 3 项对应关系（VLAN ID、MAC 地址、交换机端口）。这样当接收到一个数据帧时，交换机要同时根据数据帧的目的 MAC 和 VLAN ID 两项来查询 MAC 地址表（但其二层交换基本原理仍然一样），找到相应端口将该数据帧转发出去。大家可以在支持 VLAN 技术应用的交换机上使用指令进行 MAC 地址表查看观察。

2. 二层交换基本原理

交换机的二层交换原理其实比较简单，具体流程如图 5-46 所示。

总之，当交换机收到数据帧时，交换机根据数据帧中的目的 MAC 地址，并依据自身 MAC 地址表（地址转发表）的情况，把数据从对应的端口转发到所连接的主机或者级联的其他网络设备上。

♥**【技术思想启发】** 在交换机地址转发表建立过程中可清晰地看到：交换机通过自学习，边试错，边完善，最后形成灵活的适应性很强的自动更新转发表，以控制整个交换机的工作。可以看到“先行动，边行动，边总结，在挫折中前进”的威力，也可从技术上看到“我为人人、人人为我”的互助价值。

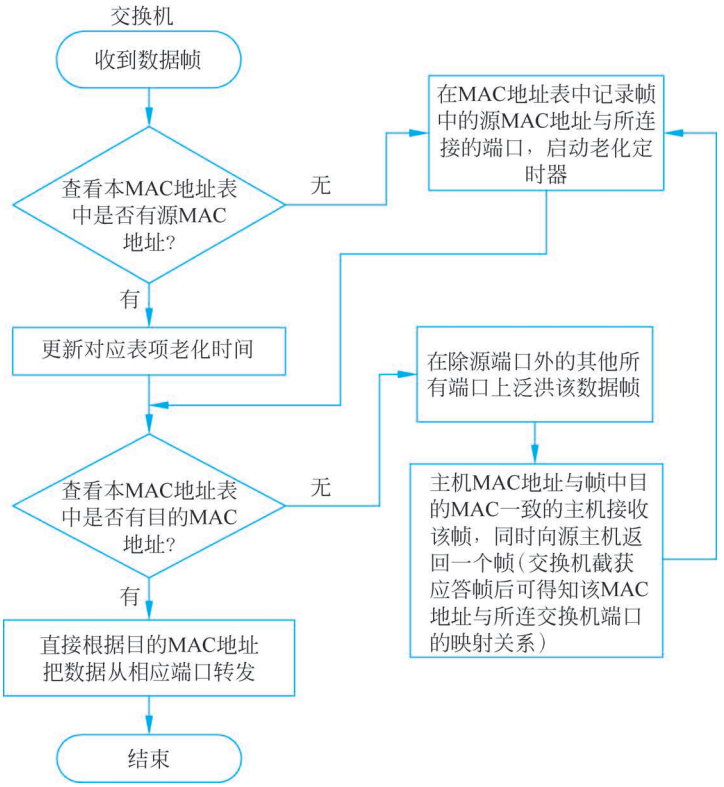


图 5-46　二层交换基本原理

5.5.4　环路检测及处理

当二层交换机连接网络出现环路时，会出现什么情况？最坏的情况是数据帧会在环路中被反复持续转发，一旦这种异常的数据帧越积越多就会导致网络瘫痪。这种情况下只能关掉网络设备的电源或者断开网络才能恢复。

为此，有必要解决网络中的环路问题。具体方法有生成树和源路由方式。使用带生成树或者源路由功能的二层交换机，即便是构建了环路网络，也不会造成严重的网络瘫痪问题。而且有时需要搭建合适的环路网络，以分散网络流量，甚至在网络某处发生故障时，因为网络是环路结构，所以数据可以绕开故障点，从而提高整体网络抗灾容灾的能力。

生成树方法由 IEEE 802.1D 定义。它规定每个运行生成树协议的交换机必须每1～10 秒相互交换网桥协议数据单元(Bridge Protocol Data Unit，BPDU)，从而判断哪些端口能否使用，以便消除环路。一旦网络发生故障，就自动切换通信线路，利用那些没有被使用的端口继续进行传输。

生成树是以各自交换机为生成树的“根”，通过对网络中每个交换机端口权重的计算，从逻辑上使物理上呈现环状结构的网络变成树状结构网络。交换机端口权重是可以通过网络管理员进行设置的，可以指定哪些端口有优先权以及发生问题时该使用哪些端口。

生成树协议与设备类型没有关系，只要是具有生成树功能的设备，都具备消除环路的功能。但 IEEE 802.1D 所定义的生成树协议最大的一个问题是：网络发生故障时，网络的切换时间比较长。为此，为解决这个切换用时过长的问题，还定义了一种快速生成树协议的方法，如 RSTP(Rapid Spanning Tree Protocol)，以及结合 VLAN 技术应用的(Multiple Spanning Tree Protocol，MSTP)，它是将网络划分为多个 VLAN，并为每个 VLAN 构建一个独立的生成树。关于生成树的协议还有很多，这里不再展开说明。

5.6 VLAN 技术

VLAN(Virtual Local Area Network，虚拟局域网)对应的技术标准是 IEEE 802.1Q，是 1999 年 6 月由 IEEE 委员会正式颁布实施的，最早的 VLAN 技术是在 1996 年由 Cisco(思科)公司提出的。VLAN 属于 OSI/RM 的二层技术，是一种将物理网络划分为多个逻辑上独立的虚拟子网络的技术，但在同一 VLAN 内部是通过数据链路层(第二层)进行通信的。

5.6.1 VLAN 应用的目的


视频

早期的以太网是没有 VLAN 技术的，后来是基于什么原因开发了 VLAN 技术呢？因为在进行网络管理的时候，时常会遇到需要分散网络负载、变换部署网络设备位置、流量隔离、管理用户等情况。早期网络管理员在进行这些操作时，不得不修改网络的拓扑结构，这就意味着必须进行相关硬件线路的改造。后来应用 VLAN 技术后，就可以不必重新修改布线，只要修改交换机配置即可。当然，这样会造成物理网络结构和逻辑网络结构不一样，从而导致新的管理问题。因此，网络管理者应该加强对逻辑网段构成和物理网络运行等的设计与管理。

VLAN 的主要用途是把一个大的交换网络划分成多个小的交换网络(即分割广播域，隔离二层通信)。那为什么要这样做呢？为什么要缩小广播域呢？其实，这样做的目的是减少二层网络中广播流量对整个交换网络的影响。

在交换网络中，广播可能是经常发生并且不可避免的。在大型的交换网络中，广播流量会给整个网络带来不小的负荷，可能影响正常的数据交换。在二层网络中，随着连接设备数量的增加，广播数据也会增加，网络状况就会变得越发糟糕。这种情况下就需要一种能够将整个物理网络进行逻辑划分的技术。于是 VLAN 技术适时而出，这种技术能够有效限制广播通信的规模。它能将一个物理网络划分成一个个不同的 VLAN(逻辑段)，每个 VLAN 相当于一个小的、独立的二层交换网络(即小的广播域)，每个 VLAN 中的广播包只会在本 VLAN 中广播，广播包的影响范围和程度自然大大降低。由此看出，VLAN 是由隔离二层通信而生的，划分 VLAN 后，不同 VLAN 中的主机不能直接进行二层通信。因此划分 VLAN 不仅隔离了广播风暴，还提高了网络的安全性。

另一方面，不同 VLAN 之间的用户被隔离在不同的二层网络中，若不借助含路由功能的设备就无法实现不同 VLAN 之间的通信，因此不同 VLAN 之间用户需要通过第三层的路由功能来实现通信。实现方法有两种：一种是通过路由器实现，另一种是通过具


视频

有三层路由功能的交换机实现。


视频

5.6.2 VLAN 的划分方式

常见的 VLAN 划分方式有如下 5 种。

(1) 基于端口的方式划分 VLAN：这是最简单和基础的 VLAN 划分方式。即以静态方式将指定的接口划分到对应的 VLAN 内，那么它就固定在这个 VLAN 中了。将具有相同特征或需求的设备连接到同一个 VLAN 上。例如，将所有服务器连接到一个 VLAN，将所有打印机连接到另一个 VLAN。

(2) 基于 MAC 地址划分 VLAN：使用 MAC 地址划分 VLAN 是一种较为灵活的方式。可以根据设备的 MAC 地址来判断其所属的 VLAN，它只看用户的 MAC 地址，不把接口固定在某个 VLAN 中，通过配置交换机的 MAC 地址表，将不同 MAC 地址范围的设备划分到不同的 VLAN。

(3) 基于子网形式划分 VLAN：按照 IP 子网来划分，将相同 IP 子网号的设备划分到同一个 VLAN 中，实现对不同子网之间的流量隔离和管理。它只看用户的 IP 子网形式，比如规定一个 192.168.1.0/24 的网段划分到 VLAN 20，那么配置了该网段的 PC 连接的接口就会动态划分到 VLAN 20。这种方式常用于大规模网络环境。

(4) 基于协议划分 VLAN：根据不同的网络协议将设备划分到不同的 VLAN 中。例如，将所有 Voice over IP(VoIP)电话连接到一个 VLAN，将所有 IP 视频设备连接到另一个 VLAN，以便对不同协议的流量进行优化管理。

(5) 基于策略的方式划分 VLAN：这种方式只有华为的产品才支持，它比较类似于将 IP 与 MAC 绑定，在 IP 跟 MAC 匹配或者是 IP、MAC 和端口都匹配时才划分到对应的 VLAN。

需要注意的是，不同的交换机厂家对 VLAN 的划分方式可能有一定的差异，具体操作方法也可能不同。


视频

5.6.3 VLAN 交换机端口类型

在 VLAN 网络中，交换机端口有 Access、Trunk、Hybrid 3 种链路类型。

Access 类型端口：只能属于一个 VLAN，一般用于连接计算机。所有通过它的接口都不需要标签，数据在经过交换机后再打上标签。

Trunk 类型端口：允许多个 VLAN 通过，可以接收和发送多个 VLAN 报文，一般用于交换机之间或者交换机与路由器之间的连接。所有通过 Trunk 的数据都需要带标签，同时可以接收和发送多个 VLAN 报文。Trunk 使得一条物理线路可以传送多个 VLAN 的数据。

Hybrid 类型端口：允许多个 VLAN 通过，可以接收和发送多个 VLAN 报文，可以用于交换机之间连接，也可以用于连接用户的计算机。Hybrid 类型接口具有 Trunk 和 Access 两种接口属性的特点，Hybrid 类型接口可以接收某个或者多个 VLAN 的数据。发送数据时：Hybrid 类型端口允许多个 VLAN 的报文发送时不打标签，而 Trunk 类型

端口只允许默认 VLAN 的报文发送时不打标签。

应特别注意，无论是哪种交换机端口类型，端口中都具有 PVID(Port VLAN ID，或称 Native VLAN、本征 VLAN)属性值，该 PVID 属性值是唯一的。端口具有 PVID 属性值的重要作用是：对于交换机端口，当接收到没有 VLAN 信息(不打标签)的数据帧时，端口就根据自己的 PVID 属性值对数据帧进行 VLAN 信息的添加(即打上标签)。


视频

5.6.4 IEEE 802.1q 帧格式

VLAN 属于二层通信协议，它对数据帧进行了重新封装，添加了 VLAN 协议头信息。支持 VLAN 应用的交换机会在普通的以太网帧的“类型”字段前加上用于标记帧 VLAN 信息的 2 个字段(共 4 字节，由 TPI 和 TCI 构成)，又称“打标签”，VLAN 协议头部符合 IEEE 802.1q 标准，如图 5-47 所示。这种增加了 VLAN 信息的数据帧，通常可称为“打标签的数据帧”(Tagged Frame)，而原来普通的数据帧就是“无标签的数据帧”(Untagged Frame)。

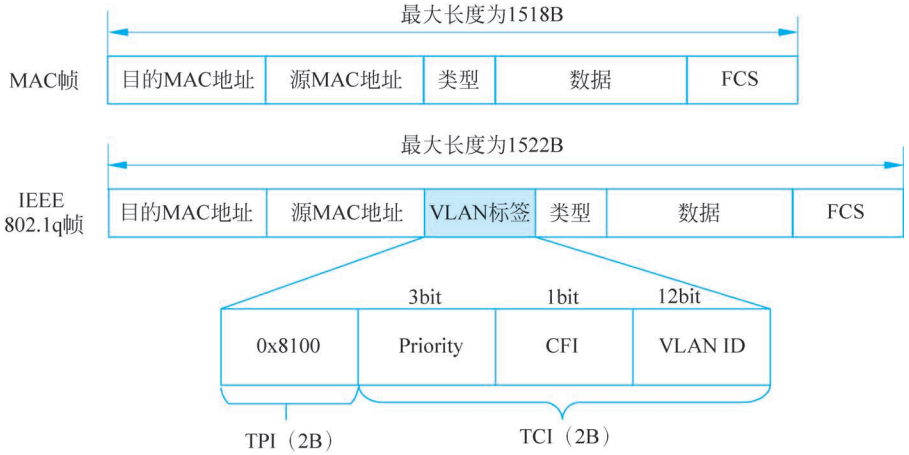


图 5-47 IEEE 802.1q 协议格式

TPI(Tag Protocol Identifier)，即标签协议标识符字段，占 2 字节(16 位)，以表明这是一个添加了 IEEE 802.1q 标签的帧(区别于未加 VLAN 标记的帧)，采用固定值 0x8100(表示封装了 IEEE 802.1q VLAN 协议)。Priority、CFI 和 VLAN ID 这 3 个字段统称为标签控制信息(Tag Control Information，TCI)，占 2 字节。

- **Priority**：用户优先级字段，占 3 位，表示 0～7 共 8 个优先级(数值越大，表示优先级越高)，主要用于确定当交换机阻塞时，优先发送哪个数据帧，也就是服务质量(Quality of Service，QoS)的应用，具体在 IEEE 802.1p 规范中详细定义。
- **CFI**(Canonical Format Indicator)：标准格式指示器字段，占 1 位，用来兼容以太网和令牌环网，标识 MAC 地址在传输介质中是否以标准格式进行封装。默认取值为 0，表示 MAC 地址以标准格式进行封装，为 1 表示以非标准格式封装。在以太网中该值总为 0。
- **VLAN ID**(VLAN Identified)：VLAN 标识字段，占 12 位，指明 VLAN 的 ID 号，

一共 4096 个(即 2^{12},但通常 VLAN ID 的最小值和最大值都不使用)。每个支持 IEEE 802.1q 协议的交换机发送出来的数据包都会包含这个域,以指明自己属于哪一个 VLAN。大部分支持 VLAN 协议的交换机都会有出厂默认 VLAN ID,其数值为 1。

视频

【本章实验】

1. 在网络仿真软件中在基于物理层设备扩展网络的基础上进行数据链路层设备的网络扩展,并融入无线局域网(WLAN)情形。实验内容:

(1) 设置用户终端 PC 的合理的 IP 地址,在网络中完成任意终端 PC 间的连通性测试(Ping 成功);

(2) 观察交换机转发表情况,说明交换机转发表的建立与更新情况。

2. 在网络仿真软件中构建如图 5-48 所示的网络拓扑,实验内容:

(1) PC1～PC4 为同网段的 IP 地址设置,所有 PC 间均可实现连通性(Ping)测试,观察交换机 LSW1 和 LSW2 的转发表;

(2)理解基于 VLAN 应用时交换机不同端口的属性特点。对交换机 LSW1 和 LSW2 相关端口进行合理配置,实现同网段同 VLAN-ID 的终端 PC 间的互联互通,不同 VLAN-ID 的终端 PC 由于交换机端口 VLAN 划分的不同而相互隔离,并观察交换机 LSW1 和 LSW2 的转发表,与要求(1)中的交换机转发表进行对比,观察并说明其中的不同。

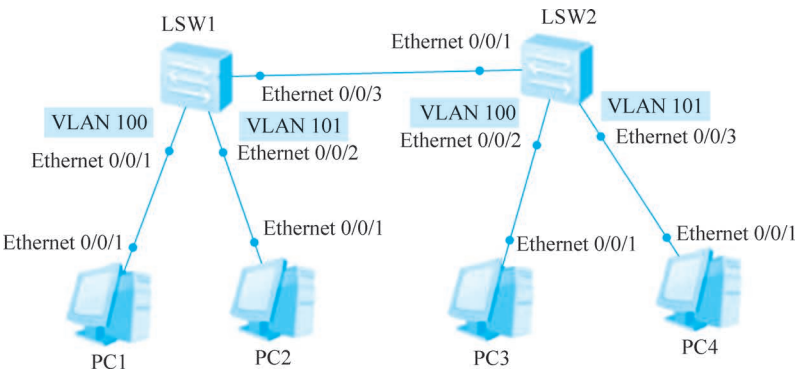


图 5-48　数据链路层实验

【本章小结】

在物理层的基础之上,数据链路层进一步保障在数据传输时的逻辑层面的数据链路可靠性,可以采取差错检测与控制以及流量控制等方法。在基于局域网的数据链路层的网络体系结构方面,可以把数据链路层进一步分成两个子层,即逻辑链路控制(LLC)子层和介质访问控制(MAC)子层,但在大部分局域网应用中,LLC 功能和协议基本无具体应用。总的来说,数据链路层的主要功能包含数据链路管理、封装成帧、透明传输和差错控制。

数据链路总体可以分为点到点型(常见于广域网中)和广播型(常见于局域网中),数据链路层的协议可分为面向比特型的和面向字符型的,不同类型的数据链路可有不同的协议支持,达成的数据链路层功能也有差异。

对于共享信道的接入情况,现在主要存在于无线局域网中。有线的共享型局域网有多种接入方法,但基本没有实用性。常见的基于以太网技术的局域网标准和数据帧格式,大多处于比较统一的情况,并且保证了前后应用上的兼容性。

局域网的搭建可以从物理层和数据链路层面进行扩展,若基于物理层设备,则主要是集线器和中继器;若基于数据链路层设备,现在主要应用的是交换机。但需要注意的是,现在的交换机类型多,可以是基于二层、三层或者高层功能的交换机。本部分主要是基于二层功能交换机进行说明,而现在大部分交换机是支持 VLAN 技术应用的,结合 VLAN 技术和网络应用需求,可以对可配置的交换机的端口进行 VLAN 有效设置,满足网络工程应用需求。

【知识对接】

1. 在 OSI 模型中,每一层都提供给多个上层使用,实现各层功能的复用,多个数据链路层共用一个物理层,那么如何从复用的物理层中提取出自己的数据?

2. 数据链路层的差错控制与物理层的差错控制有哪些不同的做法?为什么?

3. 分组网络为什么要做流量控制?电路交换需要做流量控制吗?

4. 结合 3.4.2 节的内容,分析停止-等待协议和几种 ARQ 协议分别属于哪一类应答方式。

5. 从时分复用的角度看,CSMA 属于哪种时分复用方式?

6. 如果让你设计一台交换机,你觉得应该有哪些电路模块?

【扩展阅读】

1. http://www.360doc.com/content/22/0310/13/53036841_1020905927.shtml

2. https://blog.csdn.net/weixin_45119097/article/details/127557918

3. https://cloud.tencent.com/developer/article/1339728

【思考题】

1. 为什么要设置数据链路层?

2. 设置 MAC 子层的目的是什么?对于点对点链路,有设置 MAC 子层的意义吗?为什么?

3. 交换机中 CAM 表项中设置的定时器的长短对交换机的影响是什么?

4. 在 CSMA/CA 工作原理中,站点发送 MAC 帧时检测到信道空闲,为什么还要再等待一段时间呢?

【习题】

1. 数据链路层的主要作用是什么？其特点是什么？

2. 什么是链路？什么是数据链路？两者有什么区别和联系？

3. 数据链路层要实现的基本功能包含哪几方面？为什么都必须要实现？

4. 假设发送方和接收方约定的生成多项式为 $G(X)=X^4+X^3+X^2+1$，待发送数据为 101110010，试求出 CRC 校验码。

5. 上题中，若在传输过程中，数据部分的最后一个数字 0 变成了 1，即数据为 101110011，请问接收方能否发现？若在传输过程中，数据部分的最后两个数字 10 变成了 01，即数据为 101110001，请问接收方能否发现？请根据这个例子，理解数据链路层的可靠传输。

6. 简要说明数据链路流量控制和传输层流量控制的异同。

7. 简要说明可靠传输的基本思路和想法，试着考虑是否还有其他方式解决网络中的可靠传输问题。

8. 简要说明你对 MAC 地址的理解。

9. 为什么广播链路和点到点链路要使用不同的协议？分别可以用什么协议？

10. 简述 CSMA/CD 和 CSMA/CA 协议的原理，总结这两者之间的区别，并理解为什么无线局域网中只能用 CSMA/CA，而不能直接用 CSMA/CD。

11. 使用异步传输技术的 PPP，如果接收方收到的 PPP 帧数据部分是 7D 5D 7E 27 FE 27 7D 5E 7D 5D 65 AB，那么真正的数据部分是什么？

12. 如果 PPP 使用同步传输技术，传输的比特串为 1101111101111111001，请问经过零比特填充之后得到的比特串是什么？若接收方收到的比特串为 01101111010111110100，请问发送方实际发送的比特串是什么？

13. 简述 PPP 的工作流程。

14. 什么是局域网？什么是以太网？这两者的关系是什么？

15. 典型的数据链路层设备有哪些？与物理层设备的区别有哪些？

16. 以太网交换机组网时，为什么会产生循环兜圈子的情况？如何解决这个问题？

17. 交换机有什么特点？其交换原理是什么？简述其交换表的自学习过程。怎么用它构建 VLAN？

18. 什么是冲突域和广播域？以太网交换机、集线器是否可以隔离冲突域和广播域？