第 3 章 数据链路层

本章在介绍数据链路层概念的基础上，以以太网与 WiFi 为例，系统地讨论局域网的基本概念、MAC 层协议的工作原理，并从计算机体系结构与操作系统的角度介绍计算机接入网络的原理与实现技术。

本章学习要求

- 掌握数据链路层的基本概念。
- 掌握误码率的定义与差错控制方法。
- 理解 MAC 层协议的基本概念。
- 掌握 IEEE 802.3 协议与以太网工作原理。
- 掌握 IEEE 802.11 协议与 WiFi 工作原理。

3.1 差错产生的原因与差错控制方法

3.1.1 设计数据链路层的原因

在讨论数据链路层的基本概念与协议之前，首先需要讨论一个问题：为什么要设计数据链路层？这个问题可以从以下 3 个方面来回答：

(1) 物理线路由传输介质与通信设备组成。在实际物理线路的传输过程中，人们需要进行测试，计算出各种物理线路的平均误码率，或给出某些特殊情况下的平均误码率。对于电话线路，传输速率为 300～2400b/s 时，平均误码率为 10^{-4}～10^{-6}；传输速率为 4800～9600b/s 时，误码率为 10^{-2}～10^{-4}。计算机网络对数据通信的要求是误码率必须低于 10^{-9}。因此，在不采取差错控制措施的条件下，普通电话线不能满足计算机网络的要求。

(2) 设计数据链路层的主要目的是在物理线路的基础上，采取差错检测、差错控制、流量控制等方法，将有差错的物理线路改进成无差错的数据链路，以便向网络层提供高质量的数据传输服务。

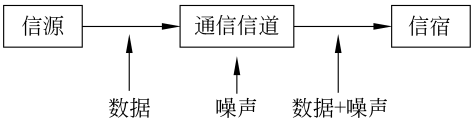
(3) 从参考模型的角度来看，物理层以上各层都有改善数据传输质量的责任，而数据链路层是其中最重要的一层。

3.1.2 差错产生的原因和差错类型

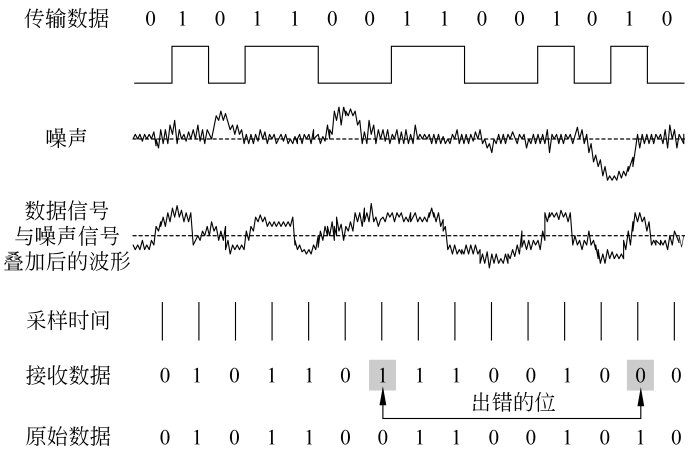
通过物理线路传输之后，接收数据与发送数据不一致的现象称为传输差错(简称差错)。

差错的产生是不可避免的。需要分析差错产生的原因与类型，研究发现是否出错及如何纠正错误的差错控制方法。

图 3-1 给出了差错的产生过程。其中，图 3-1(a)给出了数据通过信道的过程；图 3-1(b)给出了数据传输过程中的噪声影响。当数据信号在物理线路上传输时，由于物理线路上必然存在噪声，因此接收端收到的信号是数据信号与噪声信号的叠加。接收端对叠加后的信号进行判断，以便确定数据信号的二进制 0、1 值。如果信号叠加结果在电平判断时引起错误，这时就会产生传输数据的错误。



(a) 数据通过信道的过程



(b) 数据传输过程中的噪声影响

图 3-1　差错的产生过程

物理线路的噪声分为两类：热噪声和冲击噪声。热噪声是由传输介质的电子热运动而产生的。热噪声的主要特点是：时刻存在，幅度较小，强度与频率无关，但是频谱很宽。热噪声是一种随机噪声，它引起的差错是一种随机差错。冲击噪声是由外界电磁干扰引起的。与热噪声相比，冲击噪声的幅度较大，它是引起传输差错的主要原因。与数据传输中每比特的发送时间相比，冲击噪声的持续时间较长。冲击噪声造成相邻多个比特出错，它引起的差错是一种突发差错。因此，通信过程中的差错由随机差错与突发差错构成。

3.1.3　误码率的定义

误码率是指二进制比特在数据传输系统中传错的概率，用 P_e 表示，它在数值上近似等于 N_e/N。其中，N_e 为传错的比特数，N 为传输的二进制比特总数。

在理解误码率的定义时，需要注意以下几个问题：

- 误码率是衡量数据传输系统在正常工作状态下的传输可靠性的参数。在物理线路上进行传输时，数据一定会由于噪声、干扰等原因出错，差错是不可避免的，但是一

定要控制在允许的范围内。

- 对于一个实际的数据传输系统,不能笼统地说误码率越低越好,应根据实际的传输要求提出误码率要求。在数据传输速率确定之后,要求传输系统的误码率越低,则传输系统的设备就越复杂,相应的造价也就越高。
- 如果传输的数据不是二进制数,需要折合成二进制数来计算。
- 差错的出现具有随机性。在实际测量一个数据传输系统时,只有被测量的二进制比特数越多,才会越接近真实的误码率值。

3.1.4 检错码与纠错码

在数据传输系统中,检测与纠正数据传输错误的方法称为差错控制。差错控制的目的是减少物理线路上的传输错误,目前还不可能检测和纠正所有差错。在设计差错控制方法时,通常采用以下两种策略之一:

- 检错码:为传输的每个数据单元添加一定的冗余信息,接收端根据这些冗余信息发现差错,但不能确定哪个或哪些比特出错,并且不能自动纠正差错。
- 纠错码:为传输的每个数据单元添加足够的冗余信息,接收端根据这些冗余信息发现并纠正差错。

检错码虽然通过重传来纠正错误,但是工作原理简单,容易实现,因此获得广泛的使用。纠错码虽然有自己的优点,但是实现起来困难,在一般通信场景很少采用。

3.1.5 循环冗余编码工作原理

检错码主要包括奇偶校验码和循环冗余码。奇偶校验码是一种常见的检错码,它分为垂直奇偶校验码、水平奇偶校验码和水平垂直奇偶校验码(方阵码)。奇偶校验方法简单,但是检错能力较差,一般仅用于要求较低的环境。目前,循环冗余码(Cyclic Redundancy Code,CRC)是应用最广泛的检错码,具有检错能力强、实现容易的特点。

1. CRC 基本工作原理

CRC 检错方法的工作原理可以从发送端与接收端两个方面进行描述。

(1) 发送端将发送数据比特序列当作一个多项式 $f(x)$,除以一个双方预先约定的生成多项式 $G(x)$,求得一个余数多项式 $R(x)$。然后,将余数多项式加到多项式 $f(x)$之后,一起发送到接收端。

(2) 接收端将接收数据比特序列当作一个多项式 $f'(x)$,除以一个相同的生成多项式 $G(x)$,求得一个余数多项式 $R'(x)$。如果余数多项式 $R'(x)$与 $R(x)$相同,表示传输没有出错;否则,表示传输出错,并通知发送端重传数据,直至正确接收为止。

图 3-2 给出了 CRC 校验的工作原理。

CRC 校验中的生成多项式 $G(x)$由协议来规定,$G(x)$结构及检错效果是经过严格的数学分析与实验后确定的。目前,有多种生成多项式称为国际标准,例如:

- CRC-12:$G(x)=x^{12}+x^{11}+x^3+x^2+x+1$。
- CRC-16:$G(x)=x^{16}+x^{15}+x^2+1$。
- CRC-CCITT:$G(x)=x^{16}+x^{12}+x^5+1$。
- CRC-32:$G(x)=x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$。

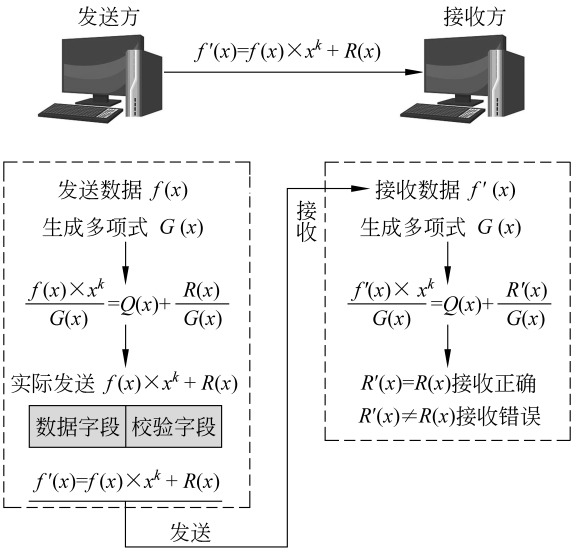


图 3-2　CRC 校验的工作原理

2. CRC 校验的工作过程

CRC 校验的工作过程如下：

(1) 发送端将发送数据比特序列 $f(x)$乘以 x^k 发送出去，其中 k 为生成多项式的最高幂值。通过这个计算将发送数据比特序列左移 k 位，用于放入余数多项式。

(2) 发送端将 $f(x)\times x^k$ 除以生成多项式$G(x)$，求得

$$\frac{f(x)\times x^k}{G(x)}=Q(x)+\frac{R(x)}{G(x)}$$

其中，$R(x)$为余数多项式。

(3) 发送端将余数多项式加到多项式 $f(x)$之后，一起发送到接收端。

(4) 接收端将接收数据比特序列 $f'(x)$采用步骤(1)与(2)的运算，求得

$$\frac{f'(x)\times x^k}{G(x)}=Q(x)+\frac{R'(x)}{G(x)}$$

其中，$R'(x)$为余数多项式。

(5) 如果余数多项式 $R'(x)$与 $R(x)$相同，表示传输没有出错；否则，表示传输出错。

3. CRC 检错方法举例

在实际生成 CRC 校验码时，通常以模二算法(即减法不借位、加法不进位)计算，这是一种异或操作。下面通过例子进一步解释 CRC 的工作原理。

在以模二算法生成 CRC 校验码时，需要注意以下几个问题：

(1) 以 CRC-12 为例，$G(x)=x^{12}+x^{11}+x^3+x^2+x+1$ 可以写为

$$G(x)=1\times x^{12}+1\times x^{11}+0\times x^{10}+0\times x^9+0\times x^8+0\times x^7+0\times x^6+0\times x^5+0\times x^4+1\times x^3+1\times x^2+1\times x^1+1\times x^0$$

CRC-12 的最高位是 x^{12}。在实际使用二进制表示时，其位数 $N=13$，用二进制表示 $G(x)$应该为 1100000001111。因此，$k=13-1=12$。

(2) 如果例子给出的生成多项式比特序列为 11001，则生成多项式应该写为

$$G(x)=1\times x^4+1\times x^3+0\times x^2+0\times x^1+1\times x^0$$

对于这个生成多项式，$N=5$，$k=5-1=4$。

下面通过一个例子来说明 *CRC* 校验码的生成过程。

(1) 发送数据为 110011(6 比特)。

(2) 生成多项式的值为 11001($N=5$、$k=4$)。

(3) 将发送数据乘以 24，求得的乘积为 1100110000。

(4) 将这个乘积除以生成多项式，按模二算法求得余数多项式为 1001。

$$\begin{array}{rl} & 100001 \quad \longleftarrow Q(x) \\ G(x)\longrightarrow 11001 & \overline{)\,1100110000} \quad \longleftarrow f(x)\cdot x^k \\ & \underline{11001} \\ & \quad 10000 \\ & \quad \underline{11001} \\ & \quad\ \ 1110 \quad \longleftarrow R(x) \end{array}$$

(5) 将余数多项式加到发送数据之后，求得带 CRC 校验码的发送数据比特序列：

$$\underbrace{110011}_{\text{发送数据比特序列}}\qquad\underbrace{1001}_{\text{CRC校验码比特序列}}$$

(6) 如果在数据传输过程中没有出错，接收端接收的数据一定能被相同的生成多项式整除：

$$\begin{array}{r} 100001 \\ 11001\ \overline{)\,1100111001} \\ \underline{11001} \\ \quad 11001 \\ \quad \underline{11001} \\ \quad\ \ 0 \end{array}$$

在实际应用中，CRC 校验码的生成与检验可用软件或硬件来实现。目前，很多超大规模集成电路芯片可实现复杂的 CRC 校验功能。

4. CRC 的检错能力

CRC 校验码的检错能力很强，除了能够检查出随机差错外，还能够检查出突发差错。突发差错是指在接收比特序列中突然出现连续几位错误。CRC 具有以下校验能力：

- 能够检查出全部离散的一位差错。
- 能够检查出全部离散的两位差错。
- 能够检查出全部奇数位差错。
- 能够检查出全部长度小于或等于 k 位的突发差错。
- 能够以 $1-(1/2)^{k-1}$ 的概率检查出长度为 $k+1$ 位的突发差错。

如果 $k=16$，CRC 校验码能检查出小于或等于 16 位的所有突发差错，并且以 $1-(1/2)^{16-1}\approx 99.997\%$ 的概率检查出 17 位的突发差错，漏检概率约为 0.003%。

3.1.6 差错控制机制

接收端通过检错码检查数据是否出错。当发现错误时，通常采用自动请求重发(Automatic Request for Repeat，ARQ)方法来纠正。

ARQ 纠错的工作过程如下：

(1) 发送端用校验码编码器为数据生成校验字段，并将数据与校验字段一起发送到接收端。为了适应 ARQ 的需求，发送端要缓存发送数据的副本。

(2) 接收端通过校验码译码器判断数据传输中是否出错。如果数据传输正确，接收端向发送端发送 ACK(传输正确)。发送端接收到 ACK 之后，不再保留发送数据的副本。如果数据传输错误，接收端向发送端发送 NAK(传输错误)。

(3) 发送端接收到 NAK 之后，将保留的数据副本重新发送，直至接收端正确接收为止。ARQ 规定了最大重发次数。如果超过最大重发次数，接收端仍无法正确接收，那么发送端停止重发，并向高层协议报告出错信息。

3.2 数据链路层的基本概念

3.2.1 链路与数据链路

链路(link)与数据链路(data link)的含义不同。图 3-3 给出了链路与数据链路的关系。

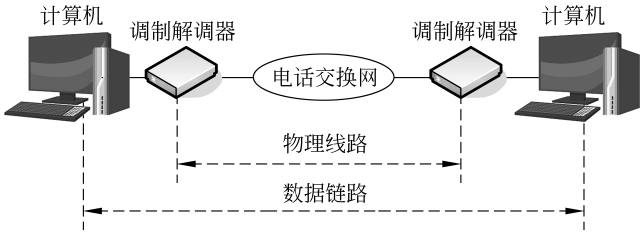


图 3-3 链路与数据链路的关系

理解链路与数据链路的区别与联系时需要注意以下几个问题：

(1) 链路是由物理线路与通信设备构成。物理线路可以是有线或无线的，用于连接相邻的两个节点。例如，连接通信双方的物理线路是电话线。为了在电话线上传输计算机的数字信号，需要用调制解调器实现数字信号与模拟信号之间的转换。这里，电话线与调制解调器构成收发双方的物理层，它就是用于完成比特流传输的链路。

(2) 在没有采取差错控制机制的链路上传输比特流可能出错。设计数据链路层的目的就是为了发现和纠正传输中可能出现的差错，将有差错的链路变成无差错的数据链路。数据链路由实现数据链路层协议的硬件、软件与链路构成。

3.2.2 数据链路层的主要功能

数据链路层主要包括以下几个功能。

1. 数据链路管理

当收发双方开始进行通信时，发送端需要确认接收端已做好准备。为了做到这一点，收

发双方必须事先交换必要的信息,以便建立数据链路;在数据传输过程中,需要维护好数据链路;在通信结束之后,需要释放数据链路。数据链路管理功能主要包括数据链路的建立、维护与释放。

2. 帧同步

数据链路层传输的数据单元是帧。物理层的比特流是封装在帧中传输的。帧同步是指接收端能够从接收的比特流中正确判断出一帧的开始与结束。

3. 流量控制

如果发送端发送的数据量超过物理线路的传输能力,或者超出接收端的帧接收能力,这时将造成链路拥塞。因此,数据链路层必须提供流量控制功能。

4. 差错控制

为了发现和纠正链路上的传输差错,将有差错的链路变成无差错的数据链路,数据链路层必须提供差错控制功能。

5. 透明传输

如果传输的帧数据中出现某些控制字符的比特序列,那么有必要采取适当的措施,避免接收端将这些比特序列误认为控制字符。数据链路层必须保证帧数据可以包含任意比特序列,即保证帧传输的透明性。

6. 寻址

在点-多点链路连接的情况下,数据链路层要保证将每帧传送到正确的接收端。因此,数据链路层也需要提供寻址能力。

3.2.3 数据链路层与网络层、物理层的关系

1. 数据链路层与网络层的关系

在OSI参考模型中,数据链路层处于网络层与物理层之间。网络层的功能是为联网计算机之间的通信寻找一条好的传输路径。图3-4给出了数据链路层与网络层的关系。如果主机A需要向主机B传送数据,则主机A的网络层启动路由选择算法,找出一条可到达主机B的传输路径(例如路由器1、2、3)。这条传输路径由多段链路组成。

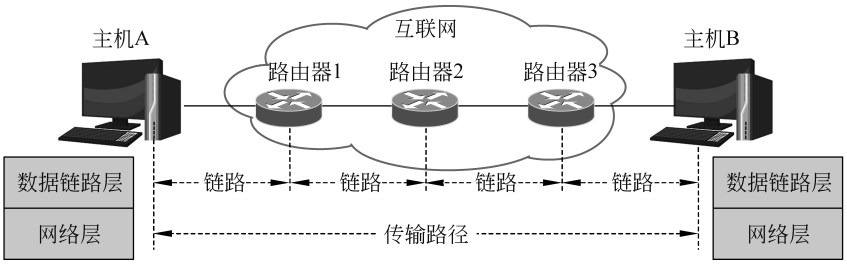


图3-4 数据链路层与网络层的关系

理解数据链路层与网络层的关系时需要注意以下两个问题:

- 网络层路由选择算法找出的传输路径一般是由多段链路组成的。如果数据链路层能够保证网络层数据经过每段链路传输时都不会出错,则网络层数据经过多段链路传输也不会出错。因此,数据链路层要为保证网络层数据传输的正确性提供服务。

- 由于数据链路层的存在，网络层无须知道物理层具体使用哪种传输介质与设备、采用模拟通信还是数字通信方法、使用有线信道还是无线信道。只要接口关系与功能不变，物理层采用的传输介质与设备的变化对网络层就不会产生影响。因此，数据链路层要为网络层屏蔽物理层传输技术的差异性提供服务。

2. 数据链路层与物理层的关系

数据链路连接建立在物理线路连接之上。在物理层完成物理线路连接并提供比特流传输能力的基础上，数据链路层才能够传输数据链路层协议数据单元——帧。数据链路层协议软件控制数据链路建立、帧传输与释放过程，并通过流量控制与差错控制来保证数据在数据链路上的正确传输，为网络层提供服务。图 3-5 给出了数据链路层与物理层的关系。

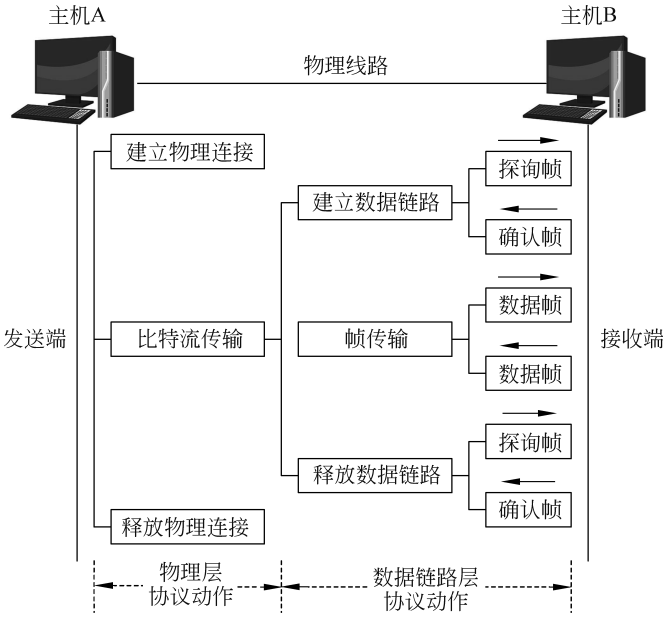


图 3-5　数据链路层与物理层的关系

理解数据链路层与物理层的关系时，需要注意以下几个问题：

- 主机 A 与主机 B 之间要传输数据，首先需要建立物理线路连接。
- 在建立物理线路连接之后，才能够传输比特流；只有传输比特流之后，才能够传输数据链路层的控制帧；控制帧通过协商来建立数据链路。
- 数据帧在数据链路上传输。
- 在数据帧传输结束之后，数据链路层的控制帧通过协商来释放数据链路。
- 在数据链路释放之后，物理线路连接应该还存在，最后才释放物理线路连接。
- 在释放物理线路连接之后，主机 A 与主机 B 之间的通信关系才完全解除。

从以上讨论中可以看出，数据链路层在物理层比特流传输功能的基础上为网络层提供服务。

3.2.4　数据链路层协议的发展与演变

为了实现数据链路层的功能，就需要制定相应的数据链路层协议。20 世纪 70 年代，计

算机网络采用的物理层技术不够成熟,数据传输速率低,误码率高,因此需要制定比较复杂的数据链路层协议来弥补物理层的缺陷。了解数据链路层协议的发展与演变,需要从点-点链路与局域网的数据链路层协议两个方向入手。

1. 点-点链路数据链路层协议的发展

早期计算机网络主要是广域网。在广域网中,连接计算机与路由器以及路由器之间的物理层线路主要是点-点链路,如电话线、电缆与光纤。最早用于点-点链路的数据链路层协议是面向字符型协议。它的特点是利用已定义好的一种标准字编码(如ACSII码或EBCDIC码)的一个子集来执行数据链路层的通信控制功能。典型的面向字符型协议是二进制同步通信(Binary Synchronous Communication,BSC)协议。面向字符型协议有3个明显缺点:一是不同类型的计算机的控制字符可能不同;二是不能实现透明传输;三是协议效率低。针对这些缺点,人们提出了面向比特型协议。典型的面向比特型协议主要包括高级数据链路控制(High-level Data Link Control,HDLC)协议与点-点协议(Point-to-Point Protocol,PPP)。

最初,由于物理层的误码率高,因此需要设计复杂的数据链路层协议,以弥补物理层存在的缺陷。1974年,ISO在IBM公司的HDLC基础上制定了数据链路层协议ISO 3309。随着物理层通信大量使用光纤,传输速率高,误码率低,数据链路层不再需要复杂的协议,HDLC逐渐被PPP协议所取代。1994年,RFC 1661、RFC 1662与RFC 1663详细说明了PPP的帧结构、差错处理、IP地址动态分配、身份认证等。PPP除了包括链路控制协议(Link Control Protocol,LCP)之外,还有涉及网络层的网络控制协议(Network Control Protocol,NCP)。这样设计PPP的主要原因是:早期计算机网络的网络层除了IP协议之外还有NetWare IPX等多种协议。随着IP成为网络层主流协议,PPP再有NCP协议就显得多余了。尽管如此,PPP在很长一段时间内仍在应用,始终未找到一种协议可取代PPP。

这种情况在高速以太网出现之后发生了变化。对于千兆以太网(Gigabit Ethernet,GE)与10GE、40GE、100GE物理层,除了局域网物理层(LAN PHY)标准之外,还需要制定远距离、点-点光纤线路的广域网物理层(WAN PHY)标准,如1000Base-ZX(单模光纤,最大长度为70km)、10GBase-ZR(单模光纤,最大长度为80km)等。因此,用高速以太网取代PPP已是大势所趋。其优点主要表现在两个方面:一是互联网络与远距离光纤链路都使用以太网协议,组网方便,协议运行效率高;二是光纤链路物理层采用高速以太网的广域网物理层标准,可获得很高的数据传输速率。

目前,宽带接入技术(如ADSL、线缆调制解调器、FTTx)使用PPPoE(PPP over Ethernet,运行在以太网上的PPP)。1999年,RFC 2516文档给出了PPPoE的内容。PPPoE是将PPP帧封装在以太网帧中,使多个使用点-点线路的PPP用户可以共享一条高带宽的以太网链路。

2. 局域网数据链路层协议的发展

20世纪80年代,广域网技术的成熟与微型机的广泛应用推动了局域网技术研究的发展。早期的局域网主要是令牌环网,如1972年美国加州大学研究的Newhall环网和1974年英国剑桥大学研究的Cambridge Ring环网,随后出现以以太网为代表的总线型局域网。无论是环形局域网还是总线型局域网,都存在多台联网主机需要共享传输介质发送和接收数据的多路访问问题。如果有多台联网主机同时争用公共传输介质,那么就会产生冲突,导

致数据发送失败。因此，有必要研究分布式介质访问控制（Medium Access Control，MAC）算法。局域网的数据链路层的研究重点是 MAC 算法。以太网采用的是带冲突检测的载波侦听多路访问（Carrier Sense Multiple Access with Collision Detection，CSMA/CD）控制算法，而令牌总线网和令牌环网采用的是令牌控制算法。

20 世纪 80 年代，局域网领域出现以太网与令牌总线网、令牌环网三足鼎立的局面，并且各自形成了国际标准。20 世纪 90 年代，以太网开始被业界认可和广泛应用。进入 21 世纪，IEEE 802.3 标准与以太网成为局域网领域“一枝独秀”的主流技术，高速以太网、交换以太网与无线以太网成为研究重点。目前，广泛应用的无线以太网——WiFi 的 MAC 层介质访问控制算法与 IEEE 802.11 标准，MAC 层采用的是带冲突避免的载波侦听多路访问（CSMA with Collision Avoidance，CSMA/CA）算法。

因此，IEEE 802.3 的 MAC 层 CSMA/CD 与 IEEE 802.11 的 MAC 层 CSMA/CA 控制算法与实现技术是数据链路层学习的重点。

3.2.5 局域网参考模型与协议标准

1. 局域网参考模型

图 3-6 给出了 OSI 参考模型与 IEEE 802 参考模型的对应关系。

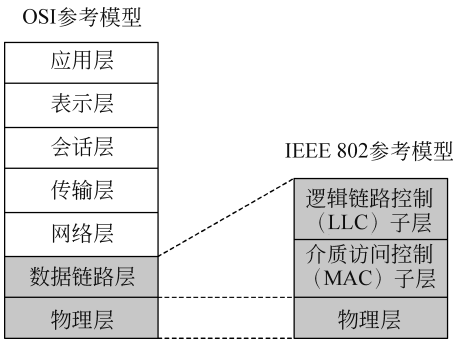


图 3-6　OSI 参考模型与 IEEE 802 参考模型的对应关系

1980 年 2 月，IEEE 成立致力于局域网标准化的 IEEE 802 委员会。IEEE 802 标准的研究重点是解决局部范围内计算机的组网问题。因此，研究者仅需面对 OSI 参考模型的数据链路层与物理层，而网络层及高层不属于局域网研究范围。

最初，局域网领域有 3 类典型技术与产品，即以太网、令牌总线网与令牌环网。市场上有很多不同厂家的局域网产品，其数据链路层与物理层协议各不相同。面对这样的复杂局面，需要为多种局域网技术建立一个共用的协议模型，IEEE 802 标准将数据链路层划分为两个子层：逻辑链路控制（Logical Link Control，LLC）子层与介质访问控制（MAC）子层。不同局域网的 MAC 子层和物理层采用不同协议，而在 LLC 子层必须采用相同协议。LLC 子层将 MAC 帧封装到统一结构的 LLC 帧中。LLC 子层与物理层采用的传输介质、MAC 方法无关，网络层不考虑低层采用的传输介质、MAC 方法与拓扑构型。

从目前局域网的实际应用情况来看，几乎所有办公自动化应用的局域网环境（例如企业网、办公网、校园网）都采用以太网，局域网中是否使用 LLC 子层已变得不重要，很多硬件和

软件厂商已经不再使用LLC协议,而直接将数据封装在MAC帧中。网络层的IP协议直接将分组封装到以太网帧中,整个协议处理过程变得简洁,因此人们已很少讨论LLC协议。目前,教科书与文献也不再讨论LLC协议的问题。

2. IEEE 802协议标准

1) IEEE 802协议标准的分类

为了研究不同的局域网标准,IEEE 802委员会成立了一系列工作组(Work Group,WG)或技术行动组(Technical Action Group,TAG),它们制定的标准统称为IEEE 802标准。随着局域网技术的发展,目前活跃的工作组是IEEE 802.3WG、IEEE 802.11WG、IEEE 802.15WG等。

IEEE 802委员会公布了很多标准,这些标准可以分为3类:

- 定义了局域网体系结构、网络互联以及网络管理与性能测试的IEEE 802.1标准。
- 定义了逻辑链路控制子层功能与服务的IEEE 802.2标准。
- 定义了不同介质访问控制技术的相关标准。

2) 介质访问控制标准的发展

不同介质访问控制技术的相关标准曾多达16个。随着局域网技术的发展,一些过渡性技术在市场的检验中逐步被淘汰或很少使用,当前应用最多、正在发展的标准主要有4个,其中3个是无线网络标准(如图3-7所示)。这4个网络协议标准如下:

- IEEE 802.3标准:定义以太网的MAC子层与物理层标准。
- IEEE 802.11标准:定义无线局域网的MAC子层与物理层标准。
- IEEE 802.15标准:定义近距离无线个域网的MAC子层与物理层标准。
- IEEE 802.16标准:定义宽带无线城域网的MAC子层与物理层标准。

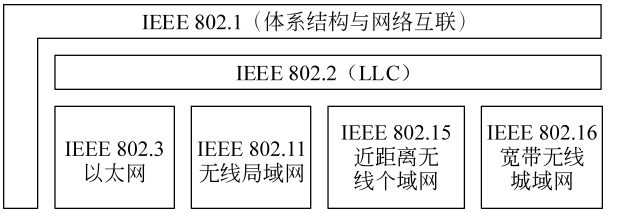


图3-7 IEEE 802协议结构

3. 局域网技术发展趋势

根据局域网技术研究与应用情况,可以总结出以下几个发展趋势:

- 以太网已成为办公环境组网的首选技术,大量计算机通过以太网接入互联网。传统的共享式以太网向交换式以太网、高速以太网与无线以太网发展。
- 由于高速以太网(GE、10GE、40GE与100GE)保留了传统以太网的帧结构、最小帧长度等特征,因此与传统以太网有很好的兼容性,采用光纤、点-点的全双工方式,增大了以太网的覆盖范围。
- 高速以太网应用从局域网逐步扩大到城域网,正在向全覆盖的方向发展,从组建办公环境的局域网向组建近距离的高性能计算机集群、存储区域网、云计算中心与互联网数据中心等机房后端网络发展。

3.3 以太网与 IEEE 802.3

3.3.1 以太网技术的发展背景

1. ALOHANET 研究的背景

以太网的核心技术是共享总线的介质访问控制方法 CSMA/CD，而它的设计思想来源于 ALOHANET。ALOHANET 出现在 20 世纪 60 年代末期。夏威夷大学为实现位于不同岛屿校区之间的计算机通信研究了一种无线分组交换网。最初设计时的数据传输速率为 4800b/s，以后提高到 9600b/s。ALOHANET 中心主机是一台位于瓦胡(Oahu)岛校园的 IBM 360 主机，它要通过学校的无线通信系统与分布在各个岛屿的计算机终端通信。因此，设计这样一个无线分组网，首先要解决的问题是：如何实现多个主机对一个共享无线信道多路访问的控制。ALOHANET 规定从 IBM 360 主机到终端的传输信道为下行信道，而从终端到 IBM 360 主机的传输信道为上行信道。下行信道是一台 IBM 360 主机向多个终端广播数据，不会出现冲突；但是，当多个终端利用上行信道向 IBM 360 主机传输数据时，就可能出现两个或两个以上终端同时争用一个信道而产生冲突的情况。解决冲突的办法只有两种：一种是集中控制方法；另一种是分布控制方法。集中控制方法在系统中设置一个中心控制主机，由中心节点决定哪个终端可使用共享的上行信道发送数据，从而避免出现多个终端争用一个上行信道的冲突现象。但是，控制中心有可能成为系统性能与可靠性的瓶颈。ALOHANET 采用的是分布式控制方法。

2. ALOHANET 访问控制的工作原理

理解 ALOHANET 访问控制的工作原理时需要注意以下几个问题：

- 每台主机在发送数据之前都需要监听无线信道是否空闲。如果没有其他主机利用无线信道传输数据，信道是空闲的，那么这台主机才可以发送数据。
- 主机发送结束之后，要等待中心主机返回正确传输的确认。如果主机在规定时间内没有接收到确认，则认为出现冲突，传输失败。主机需要重新监听信道，等到空闲时才能够重新发送。
- 由于冲突的概率与主机传输数据的频繁程度相关，因此 ALOHANET 采用的分布式控制方法是一种随机访问控制方法。

3. 以太网技术的产生过程

1973 年 5 月，Robert Metcalfe 与 David Boggs 提出以太网设计方案。他们受到 19 世纪物理学家解释光在空间传播的介质——以太(ethre)这一概念的影响，将这种局域网命名为 Ethernet。1976 年 7 月，Metcalfe 与 Boggs 发表了具有里程碑意义的论文——*Ethernet*：*Distributed Packet Switching for Local Computer Networks*。连接在以太网中的每台主机都称为节点。由于以太网中不存在集中控制节点，联网的多个节点必须平等地争用发送时间，这种多个节点争用同一传输介质的控制方法属于随机争用方法。以太网的核心技术是介质存取访问控制方法 CSMA/CD。

1977 年，Metcalfe 等申请了以太网的专利。1978 年，他们开发了以太网中继器(repeater)。1980 年，Xerox、DEC 与 Intel 公司合作，首次公布以太网物理层、数据链路层规范。1981 年，公

布 Ethernet V2.0 规范。IEEE 802.3 标准是在 Ethernet V2.0 的基础上制定的,它推动了以太网技术的发展。1982 年,第一片支持 IEEE 802.3 标准的超大规模集成电路芯片——以太网控制器问世。多家软件公司开发支持 IEEE 802.3 标准的操作系统及应用软件。

20 世纪 80 年代,以太网与令牌环网和令牌总线网之间的竞争非常激烈。早期的以太网使用的传输介质是同轴电缆,传输介质造价较高,故障率也较高。1990 年,随着物理层标准 10Base-T 的推出,非屏蔽双绞线成为 10Mb/s 的以太网传输介质。在使用非屏蔽双绞线之后,以太网组网造价降低,可靠性提高,性价比提升,以太网在竞争中占据优势。同年,以太网交换机面世,标志着交换式以太网的出现。1993 年,Kalpana 公司设计了全双工以太网,改变了传统的半双工模式,将以太网带宽增加一倍。在此基础上,以光纤为传输介质的物理层 10Base-F 标准推出,使以太网最终从三足鼎立中脱颖而出。高性价比、适应于办公环境的应用使以太网得到硬件与软件厂商的广泛支持。NetWare、Windows NT Server、IBM LAN Server 及 UNIX 操作系统的支持使以太网技术逐渐进入成熟阶段。

4. 高速以太网的发展背景

图 3-8 给出了以太网的结构。主机 A～E 都连接在一条共享总线(如同轴电缆、双绞线等)上。当主机 A 向 C 发送数据信号时,电信号沿着传输介质向两个方向传播,除主机 C 之外的其他主机都能收到主机 A 发出的电信号。如果其他主机也向共享总线上发送信号,多路信号的叠加使主机 C 不能正确接收主机 A 发出的电信号,导致此次数据传输失败。介质访问控制方法保证每个主机都能公平使用共享的总线介质。

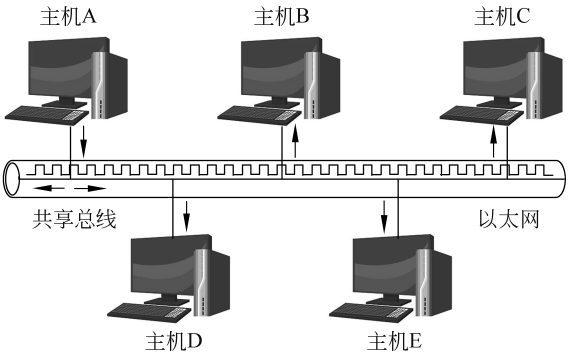


图 3-8　以太网的结构

传统的局域网技术建立在共享介质的基础上,所有网络节点共享一条公用的传输介质。介质访问控制方法用来保证每个节点都能公平使用传输介质。在网络技术讨论中,可粗略做一个估算,如果以太网中有 N 个节点,那么每个节点获得的平均带宽为 $10/N$(单位为 Mb/s)。显然,随着局域网规模的不断扩大,节点数 N 不断增加,每个节点分配的平均带宽越来越小。当网络节点数 N 增大时,网络通信负荷加重,冲突和重发次数大幅增长,传输介质利用率急剧下降,传输延迟明显增加,服务质量将显著下降。为了克服网络规模与性能之间的矛盾,人们提出了 3 种解决方案:高速、交换与互联。

1) 高速

第一种解决方案是将以太网的数据传输速率从 10Mb/s 提高到 100Mb/s,甚至是 1Gb/s、10Gb/s、40Gb/s 或 100Gb/s,这推动了高速局域网技术的发展。在这个方案中,无论局域

网传输速率提高到多少，仍保持以太网的基本特征（帧结构、最大与最小帧长度）。

这里从传播延时带宽积的角度去认识提高传输速率的必要性。评价网络性能的两个参数是传播延时与带宽。这两个参数的乘积称为传播延时带宽积，经常简称为延时带宽积。延时带宽积＝传播延时×带宽。图 3-9 给出了延时带宽积的物理意义。

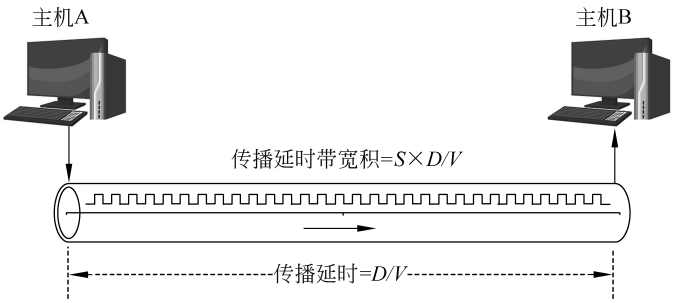


图 3-9　延时带宽积的物理意义

如果以太网的总线长度 $D=500\text{m}$，电磁波在总线中的传播速度为 $V=2\times10^8\text{m/s}$，那么主机 A 发送的信号经过 500m 的传播，到达主机 B 的传播延时为 2.5μs。如果主机 A 的发送速率为 10Mb/s，那么经过 2.5μs，在传输介质上可连续发送的比特数为 10Mb/s×2.5μs＝25b。对于总线长度为 500m、电磁波传播速度为 $2\times10^8\text{m/s}$、传输速率为 10Mb/s 的以太网，延时带宽积仅能达到 25b；速率提高到 100Mb/s，那么延时带宽积将达到 250b；速率提高到 1Gb/s，延时带宽积将达到 2500b；速率提高到 10Gb/s，延时带宽积将达到 25 000b。因此，提高延时带宽积对改善网络性能至关重要。

2）交换

第二种解决方案是将共享介质方式改为交换方式，这推动了交换式局域网（switched LAN）技术的发展。交换式局域网的核心设备是局域网交换机，支持在交换机多个端口之间同时建立多个并发连接。从这个角度，局域网被分为两类：共享式局域网（shared LAN）与交换式局域网。

3）互联

第三种解决方案是将一个大型局域网划分成多个用网桥或路由器互联的小型局域网，这推动了局域网互联技术的发展。网桥、交换机与路由器可以隔离子网之间的广播通信量。通过减少每个子网的节点数的方法，使每个局域网的网络性能得到改善。

1995 年，100Mb/s 的快速以太网（Fast Ethernet）标准发布。1998 年，1Gb/s 以太网（1 Gigabit Ethernet GE）标准发布。1999 年，GE 的产品问世，并成为局域网主干的首选方案。2002 年，10Gb/s 以太网（10 Gigabit Ethernet，10GE）标准发布。2010 年，40Gb/s 以太网（40 Gigabit Ethernet，40GE）与 100Gb/s 以太网（100 Gigabit Ethernet，100GE）标准完成。这些标准进一步增强了以太网在网络建设中的竞争优势。通常将快速以太网、1Gb/s 以太网、10Gb/s 以太网、40Gb/s 以太网与 100Gb/s 以太网分别称为 FE、GE、10GE、40GE 与 100GE，而将 10Mb/s 以太网称为传统以太网或以太网。

在高速以太网发展的同时，以太网技术进一步向无线以太网、工业以太网与电信级以太网方向发展，以太网从局域网向城域网、广域网和无线网络等领域扩展，并成为公认的主流

组网技术之一。

5. 以太网物理地址

理解以太网物理地址的概念时需要注意以下3个问题。

1) 对以太网物理地址的管理方法

48位地址称为EUI-48。EUI(Extended Unique Identifier)表示扩展的唯一标识符。按48位以太网物理地址的编码方法,可分配的以太网物理地址为2^{47}个,这个数量可保证全球任何一个以太网物理地址是唯一的。为了统一管理以太网物理地址,IEEE注册管理委员会(Registration Authority Committee,RAC)为每个网卡生产商分配以太网物理地址的前三字节,即公司标识(commpany-id),又称为机构唯一标识符(Organizationally Unique Identifier,OUI)。后三字节由网卡的厂商自行分配。

2) 以太网物理地址的表示方法

当某个网卡生产商获得一个前三字节地址分配权后,它可以生产的网卡数量是2^{24}(16 777 216)块。例如,IEEE分配给某个公司的以太网物理地址前三字节可能有多个,其中一个是020100。表示方法是在两个十六进制数之间用一个连字符隔开,即02-01-00。该公司可以为其生产的每块以太网网卡分配一个后三字节地址值,例如2A-10-C3。那么,这个网卡的物理地址应该是02-01-00-2A-10-C3,也可以写为0201002A10C3。

3) 以太网物理地址的唯一性

在网卡的生产过程中,物理地址写入网卡的只读存储器中。将这块网卡安装在任意一台计算机中,这块网卡的以太网物理地址都是02-01-00-2A-10-C3。不管这台计算机放在何处,连接到哪个局域网中,这块网卡的物理地址都是不变的,并且不会与全球任何一台计算机中的网卡的物理地址相同。

3.3.2 以太网数据发送流程分析

有人将CSMA/CD的工作过程形象地比喻成很多人在一间黑屋子中举行会议,参加会议的人只能听到其他人的声音。每个人在说话前必须倾听,只有等会场安静下来后,他才能发言。人们将发言前监听以确定是否有人发言的动作称为载波侦听;在会场安静的情况下,每人都有平等的机会讲话称为多路访问;如果同一时刻有两人或两人以上说话,大家无法听清其中任何一人的发言,这种情况称为冲突;发言人在发言过程中要及时发现冲突,这个动作称为冲突检测。如果发言人发现冲突,他需要停止讲话,然后随机延迟,再次重复上述过程,直至讲话成功。如果失败次数太多,也许他就会放弃这次发言的想法。

为了有效实现多个节点访问公共传输介质的控制策略,CSMA/CD的发送流程可以简单概括为4步:先听后发,边听边发,冲突停止,延迟重发。图3-10给出了以太网节点的数据发送流程。

1. 载波侦听过程

以太网中的任何一个节点在发送数据帧之前都要首先侦听总线的忙/闲状态。以太网网卡的收发器一直在接收总线上的信号。如果总线上有其他节点发送的信号,则曼彻斯特解码器的接收时钟一直有输出;如果总线上没有数据信号发送,则曼彻斯特解码器的接收时钟输出为0。因此,接收电路的曼彻斯特解码器的接收时钟输出能够反映出总线的忙/闲状态,如图3-11所示。

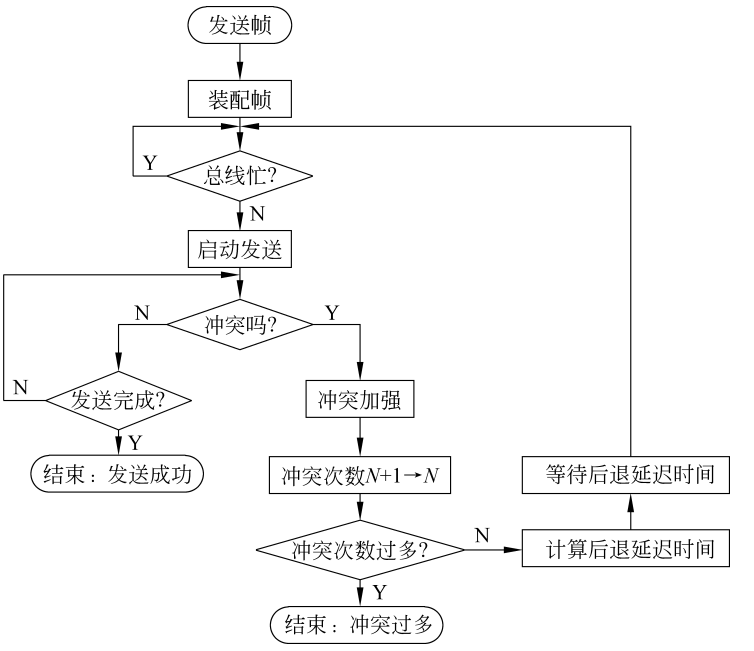


图 3-10　以太网节点的数据发送流程

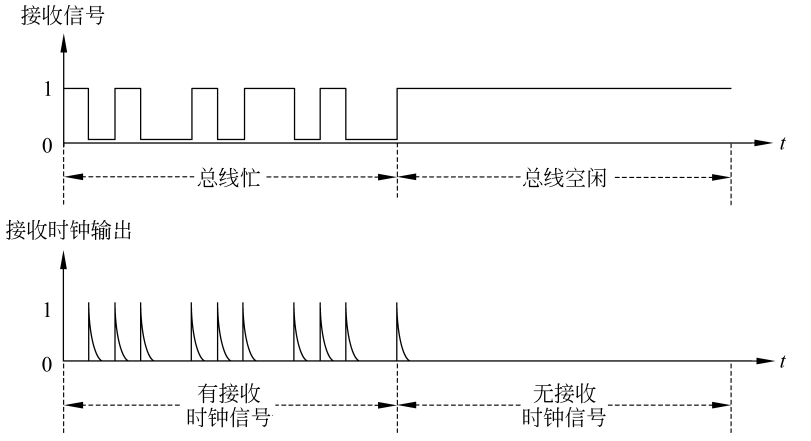


图 3-11　接收时钟输出与总线忙/闲状态

2. 冲突检测方法

载波侦听并不能完全消除冲突。数字信号以一定速度在介质中传输。电磁波在同轴电缆中传播速度只有光速的 2/3 左右，即大约 2×10^8 m/s。例如，局域网中相隔最远的两个节点 A 和 B 相距 1000m，那么节点 A 向 B 发送一帧数据要经过大约 5μs 传播延时。也就是说，在节点 A 开始发送数据 5μs 后，节点 B 才可能接收到这个数据帧。在这 5μs 的时间内，节点 B 并不知道节点 A 已发送数据，它就有可能也向节点 A 发送数据。当出现这种情况时，节点 A 与 B 的此次发送就发生冲突。因此，多个节点在公共传输介质上发送数据需要进行冲突检测。

极端的情况是：节点 A 向 B 发送了数据，当数据信号快达到节点 B 时，节点 B 也发送

了数据,此时发生冲突。当冲突的信号传送回节点A时,经过2倍的传播延时(2τ),其中$\tau=D/V$,D为总线介质的最大长度,V为电磁波在传输介质中的传播速度。在2倍的传播延时中,冲突的帧可以传遍整个缆段。整个缆段连接的所有节点都应该检测到冲突。一个缆段是一个冲突域(collision domain)。如果超过2倍的传播延时没有检测出冲突,就认为该节点已取得总线访问权。因此,$2D/V$被定义为冲突窗口(collision window)。冲突窗口是指连接在一个缆段上的所有节点都能检测到冲突的最短时间。由于以太网物理层协议规定了总线的最大长度,电磁波在介质中的传播速度是确定的,因此冲突窗口的大小也是确定的。图3-12给出了冲突域与冲突窗口的概念。

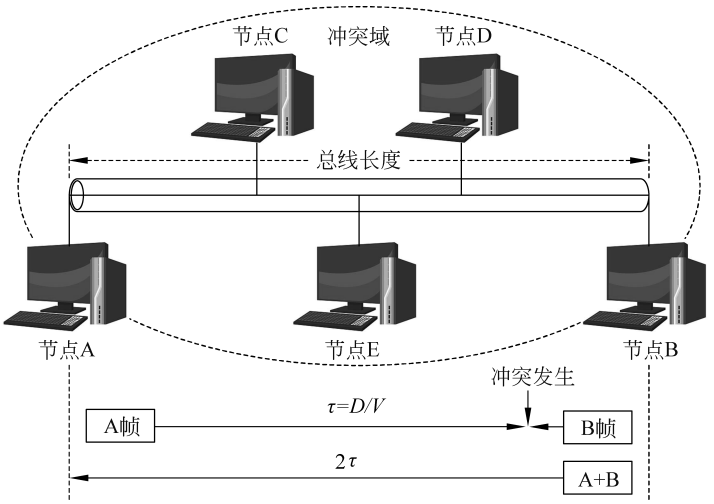


图3-12 冲突域与冲突窗口的概念

理解冲突与冲突窗口的概念时,需要注意以下两个问题。

(1) 最小帧长度与总线长度、发送速率之间的关系。为了保证任何一个节点在发送一帧的过程中都能够检测到冲突,就要求发送一个最短帧的时间都要超过冲突窗口大小。如果最短帧长度为$L_{\min}$,节点发送速率为S,则发送最短帧所需的时间为$L_{\min}/S$。冲突窗口值为$2D/V$。要求发送一个最短帧的时间都要超过冲突窗口大小,即

$$L_{\min}/S \geqslant 2D/V$$

那么,总线长度与最小帧长度、发送速率之间的关系为

$$D \leqslant VL_{\min}/2S$$

可以根据总线长度、发送速率与电磁波传播速度计算出最小帧长度。

(2) 在网络环境中如何检测到冲突。从物理层来看,冲突是指总线上同时出现两个或两个以上发送信号,它们叠加后的信号波形将不等于任何节点输出的信号波形。例如,总线上同时出现了节点A与B的发送信号,它们叠加后的信号波形将既不是节点A的信号,也不是节点B的信号。节点A与B的信号都采用曼彻斯特编码,叠加后的信号波形既不符合曼彻斯特编码的信号波形,也不是任何一路信号的波形。图3-13给出了曼彻斯特编码信号的波形叠加情况。

从电子学实现方法的角度,冲突检测可以有两种方法:比较法和编码违例判决法。比较法是指发送节点在发送帧的同时将发送信号波形与从总线上接收到的信号波形进行比

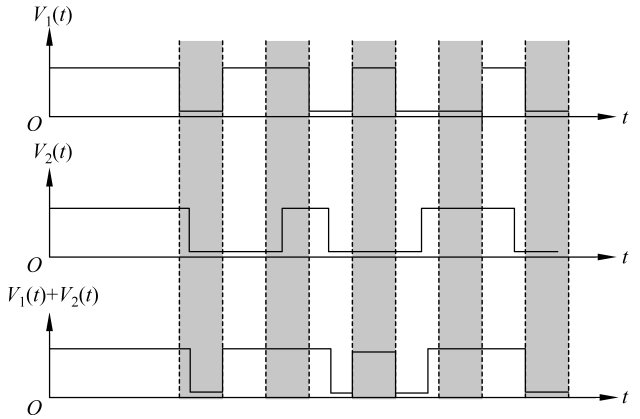


图 3-13　曼彻斯特编码信号的波形叠加情况

较。如果发送节点发现这两个信号波形不一致，表示总线上有多个节点同时发送数据，即可判定已经发生冲突。编码违例判决法是指接收节点检查从总线上接收的信号波形。如果接收的信号波形不符合曼彻斯特编码规律，就说明已经出现了冲突。

以太网协议规定的冲突窗口大小为 51.2μs。以太网传输速率为 10Mb/s，冲突窗口的 51.2μs 可发送 512b(64B)数据。64B 是以太网最小帧长度。这意味着当一个节点发送一个最小帧或一个帧的前 64B 时没有发现冲突，则表示该节点已获得总线发送权，并可以继续发送后续的字节。因此，冲突窗口又称为争用期(contention period)。

如果在发送数据过程中没有检测出冲突，在发送完所有数据之后报告发送成功，进入“接收正常”的结束状态。

3. 发现冲突时停止发送

如果在发送数据过程中检测出冲突，为了解决信道争用问题，发送节点要进入停止发送数据、随机延迟后重发的流程。第一步是发送冲突加强干扰序列(jamming sequence)信号。冲突加强干扰序列信号长度规定为 32b。这样做的目的是：确保有足够的冲突持续时间，使以太网中所有节点都能检测出冲突的存在，并立即丢弃冲突帧，减少因冲突而浪费的时间，提高信道的利用率。

4. 随机延迟重发

以太网协议规定一个帧的最大重发次数为 16。如果重发次数超过 16，则认为线路故障，进入“冲突过多”结束状态。如果重发次数不超过 16，则允许节点随机延迟再重发。

为了公平地解决信道争用问题，需要确定后退延迟算法。典型的算法是截止二进制指数后退延迟(truncated binary exponential backoff)算法。该算法可以表示为

$$\tau = 2^k Ra$$

其中，τ 为重新发送所需的后退延迟时间，a 是冲突窗口值，R 是随机数。如果一个节点需要计算后退延迟时间，则以其地址为初始值产生随机数 R。

节点重发后退的延迟时间是冲突窗口值的整数倍，并与以冲突次数为二进制指数的幂值成正比。为了避免延迟过长，截止二进制指数后退延迟算法限定二进制指数 k(即 2^k)的范围，定义 $k=\min(n,10)$。如果重发次数 $n<10$，则 k 取值为 n；如果重发次数 $n\geqslant 10$，则 k 取值为 10。例如，如果第一次冲突发生，则重发次数 $n=1$，取 $k=1$，即在冲突后两个时间片

后重发。如果第二次冲突发生,则重发次数 $n=2$,取 $k=2$,即在冲突后 4 个时间片后重发。在 $n<10$ 时,随着 n 的增加,重发延迟时间按 2^n 增长。当 $n\geqslant 10$ 时,重发延迟时间不再增长。由于限制二进制指数 k 的范围,则第 n 次重发延迟分布在 0 与$[2^{\min(n,10)}-1]$个时间片内,最大可能延迟时间为 1023 个时间片。在后退延迟时间到达后,节点重新判断总线忙/闲状态,重复发送流程。当冲突次数超过 16 时,表示发送失败,放弃发送该帧。

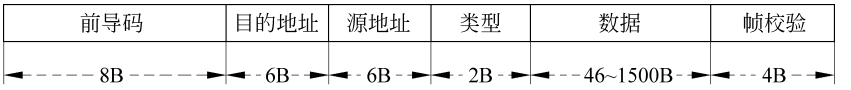
从上述讨论可以看出,任何节点发送数据都要通过 CSMA/CD 方法争取总线使用权,从准备到开始发送的等待时间不确定。因此,CSMA/CD 方法是一种随机争用型介质访问控制方法。

3.3.3 以太网数据接收流程分析

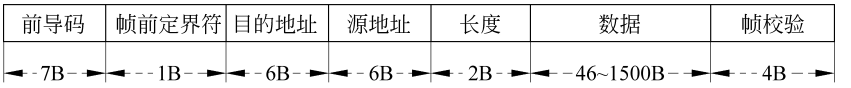
1. 以太网帧结构

为了分析以太网的数据接收流程,首先需要了解以太网帧结构。这时,需要注意 Ethernet V2.0 标准与 IEEE 802.3 标准以太网帧结构的区别。

Ethernet V2.0 标准是在 DEC、Intel 与 Xerox 公司合作研究的以太网协议的基础上改进的结果,有些文献将 Ethernet V2.0 帧结构称为 DIX 帧结构。IEEE 802.3 标准也规定了以太网帧结构,通常将它称为 IEEE 802.3 帧。DIX 帧和 IEEE 802.3 帧的结构有差异。图 3-14 给出了 DIX 帧与 IEEE 802.3 帧的结构比较。



(a) DIX帧结构



(b) IEEE 802.3 帧结构

图 3-14 DIX 帧与 IEEE 802.3 帧的结构比较

DIX 帧与 IEEE 802.3 帧的结构差异主要表现在以下两点。

1) 前导码部分

DIX 帧与 IEEE 802.3 帧在前导码部分的差异如下:

(1) DIX 帧的前 8B 是前导码,每个字节都是 10101010。接收电路通过提取曼彻斯特编码的自含时钟实现收发双方的比特同步。

(2) IEEE 802.3 帧规定 7B 前导码由 7 个 10101010 组成,之后是 1B 的帧前定界符,为 8 位的 10101011。从物理层的角度来看,由于接收电路在曼彻斯特解码时需要采用锁相电路,而锁相电路从开始接收状态到达同步状态需要 10～20μs 的时间。设置 7B 前导码的目的是保证接收电路在接收帧目的地址字段之前已进入稳定接收的状态,能够正确地接收。如果将前导码与帧前定界符结合起来看,在 62 位比特序列后出现 11,在 11 之后开始出现以太网帧的目的地址字段。

2) 类型字段与长度字段

DIX 帧与 IEEE 802.3 帧在类型字段与长度字段部分的差异如下:

(1) DIX 帧规定了一个 2B 的类型字段。类型字段表示网络层使用的协议类型。例如，类型字段值等于 0x0800，表示网络层使用 IPv4 协议；类型字段值等于 0x8106，表示网络层使用 ARP 协议；类型字段值等于 0x86DD，表示网络层使用 IPv6 协议。

(2) IEEE 802.3 帧规定对应的字段为长度字段。数据字段是网络层发送的数据部分。由于帧最小长度为 64B，帧头部分长度为 18B(6B 的目的地址字段，6B 的源地址字段，2B 的长度字段，4B 的帧校验字段)，因此数据字段最小长度为 64－18＝46B。数据字段最大长度为 1500B。因此，数据字段长度为 46～1500B，长度不是固定的。从这个角度来看，设置长度字段是合理的。

由于 DIX 帧没有设定长度字段，因此接收方只能根据帧间间隔判断一帧是否完成接收。当一帧发送结束时，物理线路上不会出现表示一帧发送结束的电平跳变。如果接收方认为已经完整地接收了一帧，那么除去最后的 4B 校验字段，就能够取出数据字段。

由于 Ethernet V2.0 标准已得到广泛应用，因此 IEEE 802.3 标准修订中采用了折中方案，将 2B 的长度字段改为长度/协议字段。同时表示长度和协议并不矛盾。以太网帧的最大长度小于 1518B，如果用十六进制表示，长度字段值一定小于 0x0600。定义的协议字段值最小为 0x0800(IP 协议)。接收方的 MAC 层可根据该字段的值来解释其含义。这样做就消除了 IEEE 802.3 与 Ethernet V2.0 标准之间存在的差异。目前，DIX 帧结构已得到广泛应用，本节将以 DIX 帧为对象分析以太网帧结构的特点。

2. 以太网帧结构分析

以太网帧结构由 6 个部分组成：前导码字段、目的地址字段、源地址字段、类型字段、数据字段和帧校验字段。

1) 前导码字段

前导码由 8 个 10101010 比特序列组成，共 8B。前导码的作用是实现收发双方的比特同步与帧同步。前导码在接收后不需要保留，也不计入帧头的长度。

2) 目的地址与源地址字段

目的地址与源地址分别表示帧的接收节点与发送节点的硬件地址。硬件地址通常称为物理地址、MAC 地址或以太网地址。地址长度为 6B。源地址必须是 48 位的 MAC 地址，目的地址可以是单播地址、多播地址或广播地址。

3) 类型字段

类型字段表示网络层使用的协议类型。

4) 数据字段

数据字段是网络层发送的数据部分。数据字段的长度为 46～1500B。加上帧头部分的 18B，以太网帧最大长度为 1518B。因此，以太网帧最小长度为 64B，最大长度为 1518B。如果发送方数据长度小于 46B，则在组帧之前要填充到最小数据长度。

在 DIX 帧中，由于没有设置长度字段，接收方不知道发送方是否对数据字段做了填充以及填充了多少个字节。如果高层使用的是 IP 协议，IP 协议分组头有总长度字段，该字段值表示发送方发送的 IP 分组长度。接收方根据总长度字段值就可以方便地确定填充字节的长度，并且删除填充字节。

5) 帧校验字段

帧校验字段采用 4B(32 位)的 CRC 校验。CRC 校验的范围包括目的地址、源地址、长