第 3 章　网络体系结构

所要讲述的问题如下：

1. 应用层的功能及实现模式
2. 传输层的功能及实现策略
3. 网络层的功能及实现模式、路由算法、拥塞控制算法
4. 数据链路层的差错控制方法、协议
5. 物理层的功能、协议

3.1.1 应用层功能

为网络应用访问网络环境提供各种应用服务。

网络应用：满足用户需要的各种具体应用，通常是各种**应用程序**（AP）

应用服务：为支持各种网络应用而实现的公共服务，通常体现为各种**应用协议**。

具体功能：

1. 应用管理
   - ① 参数初始化
   - ② 应用进程创建、维护
   - ③ 资源分配、回收
   - ④ 安全控制
2. 系统管理【监督、报告、统计等】

3.1.2　应用对服务的要求

(1)可靠的数据传输

(2)足够的带宽

(3)定时

3.1.3 应用层抽象模型

3.1.4 应用层的服务模式：C/S vs P2P

3.1.4 应用层的服务模式：C/S vs P2P

3.2.0 传输层的目的

在两个应用实体之间实现可靠的、透明的、有

效的数据传输，使高层用户在相互通信时不必关

心通信子网的细节。

可靠：传输层处理并隔离低层的错误

透明：高层用户不涉及点对点间通信的任何细节

有效：全双工、尽量高效

3.2.1 主要功能

3.2.1 主要功能（con’t)

3.2.2 服务质量（QoS）

3.2.3 传输层基本策略

3.2.3 传输层基本策略
3.2.3 传输层基本策略
3.2.3 传输层基本策略
3.2.3 传输层基本策略

流量控制

流量控制
流量控制

流量控制

3.2.4 传输服务原语

3.2.4 传输服务原语
3.2.4 传输服务原语

3.3.2 网络层的实现方法-数据报与虚电路

2.数据报的实现

数据报服务

二、虚电路

虚电路服务

2. 虚电路的实现

2. 虚电路的实现
2. 虚电路的实现

虚电路是逻辑连接

虚电路表示这只是一条逻辑上的连接，分组都沿着这条逻辑连接按照存储转发方式传送，而并不是真正建立了一条物理连接。

请注意，电路交换的电话通信是先建立了一条真正的连接。因此分组交换的虚连接和电路交换的连接只是类似，但并不完全一样。

三、 虚电路与数据报的比较

3.3.3 路由选择算法

一、静态路由选择算法

条件：不考虑网络的状态

(1)随机路由选择算法（随机走动法）

算法思想：当数据包到达一个节点后，该节点**随机选择一条输出线**转发该数据包。

①完全随机法，假定与该节点相连的链路有 N 条,则产生一个从 1 到 N 之间的随机数 i，把数据包从第 i 条输出线上转发；

②轮选法，即对所有输出线排序，每来一个数据包，依次选一输出线转发表。

缺点：可能将所收到的数据包又从输入线上转发出去，即将**数据包原路返回**。

解决办法：采用**计程法**，即在数据包中增加一个字段，记录包经过的节点的数目。

(2)扩散路径选择算法 (洪泛法)

算法思想：当某个节点收到一个不是发给它的分组时，就**向所有与此节点相连的链路转**

发出去。当然，**不能再把这个分组发到它刚刚离开的那个节点**。

缺点：会产生大量的重复包，包的数目可能会呈指数规律增加。结果导致网络出现拥塞。

(2)扩散路径选择算法（洪泛法）

解决办法：

① **采用站计数法**，每个包中增加一个站点**计数字段**，初值设为从源节点到目的节点的**路径长度**(最多节点数)。数据包每经过一个节点，站计数器减 1。当该值变为 0 时，若还未到达目的节点，就丢弃该数据包。

② **首次登录法**，每个包中增加一个序号字段，在每个节点设置一张表记录首次到达本节点的包的次序。**当收到一个包时**，检查相应**源节点**发送的**该包是否首次到达节点**。若是，则登录序号，并扩散转发；否则，丢弃该包。

用途：

- 广播；
- 高可靠性环境、拓扑不稳定；
- 分布式数据库并行更新；
- 评价其它路由算法；
- 无线网；
- 辅助路由信息的传送，如传输链路状态信息。

洪泛法在军用网中很有用，因为它有很好的稳健性。

(3) 固定路由法

算法思想：当节点收到数据包后，检查目的地，然后在输出线**选择表**中查找到该目的节点的主路径输出线并从该输出线上转发数据包。**表中可以规定多条输出线**。算法速度快、开销小。

可以根据链路度量值进行选择。

缺点：固定路径由**网管人员指定**，一旦网络本身出现故障或其它原因导致拓扑结构发生变化，则原来指定的路径就可能走不通，数据包无法到达目的地，必须重新指定路径。

用途：永久虚电路（PVC）。

(4) 分散通信量法

算法思想：是事先在每个节点的**内存中设置**一个路由表，但此路由表中给出**几个可供采用**的输出链路，并且对每条链路赋予一个**概率**。当一个分组到达该节点时，此节点即产生一个从 0.00 到 0.99 的随机数，然后按此随机数的大小，查表找出相应的输出链路。 **例如：分组到 K 站，目的站为 B。**

目的站	经过	概率	经过	概率	经 过 概率
A	M	0.50	L	0.40	N 0.10
B	M	0.35	N	0.35	L 0.30
C	N	0.65	M	0.25	P 0.10
E	N	0.55	P	0.30	M 0.15
D	P	0.45	N	0.30	M 0.25
…	…		…		…

这种方法与固定路由相比，可使网内的通信量更加平衡和得到较小的平均分组时延。

二、动态路由选择算法（自适应）

I. 孤立自适应路由选择算法

II.分布式自适应路由选择算法

②距离信息传播

④距离向量法举例 1(con’t)

④距离向量法举例 1

④距离向量法举例 1(con’t)

④距离向量法举例 1(con’t)

④距离向量法举例 2

⑤问题

(2)链路状态路由算法

工作过程：P81-（黄）

2）问题

三、层次路由选择算法

四、广播路由选择算法

五、组播路由选择算法

3.3.4 拥塞控制算法

一、概念

含义：指在通信子网中有太多的包存在，使得网络的性能降低，甚至不能工作的状况

现象：包丢失（是拥塞的征兆，是判断拥塞的标准之一）

拥塞的原因

① 处理器速度太慢

② 线路容量限制

③ 节点输出包的能力小于输入包的能力

④ 网络流量分布不均衡

对资源需求的总和 > 可用资源

←→资源不足【CPU、缓冲区、线路等】

有时，正是拥塞控制本身成为引起网络性能恶化甚至发生死锁的原因

拥塞控制与流量控制
拥塞控制的目标
拥塞控制所起的作用
二、拥塞控制原理——开环与闭环原理
②闭环原理
三、拥塞控制一般方法
(1)拥塞预防方法
② 合理分配缓冲区法
③ 通信量整形法
b. 漏桶算法
c. 令牌桶算法
令牌桶算法与漏桶算法的区别
(2) 拥塞抑制方法
阻塞包算法
阻塞包算法
阻塞包算法
缺点
2) 负载丢弃法

为什么需要数据链路层:

■ 存在**传输差错**的可能性，并且数据接收方可能要**调整数据到达的速率**，因此完全靠同步和接口技术本身还是不够的。有必要在每个通信设备中再增加一个控制层，由它提供如**流量控制、差错检测**以及**差错控制**等功能。

数据链路层的主要任务是要保证物理上相邻的两个节点之间进行可靠的数据传输。

■ 链路(link)：是一条无源的点到点的物理线路段，中间没有任何其他的交换节点。
♦ 一条链路只是一条通路的一个组成部分。
■ 数据链路(data link) 除了物理线路外，还必须有**通信协议**来控制这些数据的传输。若把实现这些协议的**硬件**和**软件**加到链路上，就构成了数据链路。
♦ 现在最常用的方法是使用适配器（即网卡）来实现这些协议的硬件和软件。
♦ 一般的适配器都包括了数据链路层和物理层这两层的功能。
■ 当**采用复用技术**时，一条链路上可以有多条数据链路。
数据链路层像个数字管道
■ 常常在两个对等的数据链路层之间画出一个数字管道，而在这条数字管道上传输的数据单位是帧。
■ 早期的数据通信协议曾叫作通信规程(procedure)。因此在数据链路层，规程和协议是同义语。

■ **数据链路层最重要的作用（目的）就是：通过一些数据链路层协议（即链路控制规程），把不可靠的传输线变为可靠的传输线。**

■ **或者：将物理层为传输原始比特流而提供的可能出差错的链路改造成为逻辑上无差错的数据链路。**

■ **信道类型：**点点信道、广播信道
■ **方法：**成帧传送
数据链路层的简单模型
数据链路层的简单模型(续)

数据链路层的主要功能： P89一（黄）

（1）链路管理 （2）帧同步 （3）流量控制
（4）差错控制 （5）透明传输 （6）寻址
3.4.1 数据链路层功能

1.成帧（封装成帧、帧同步）
封装成帧(framing)就是在一段数据的前后分别添加首部和尾部，然后就构成了一个帧。确定帧的界限。
首部和尾部的一个重要作用就是进行**帧定界**。

保证帧同步的方法：

①字符界定法：特定标志字符
②字符计数法：开始字符、长度字段
③位串界定法：
01111110，位填充
帧长度： $L_d^{opt} \approx SQRT(L_h/p_b)-L_h$
例： $L_h=48b$， $p_b=4\times10^{-5}$， $L_d^{opt} \approx 1000b$

2.透明传输

所谓**透明传输**就是无论所传数据是什么样的比特组合，都不会影响数据传输的正常进行。
透明传输举例
解决透明传输问题
发送端的数据链路层在数据中出现控制字符“SOH”或“EOT”的前面插入一个转义字符“ESC”(其十六进制编码是 1B)。
字节填充(byte stuffing)或字符填充(character stuffing)——接收端的数据链路层在将数据送往网络层之前删除插入的转义字符。
如果转义字符也出现数据当中，那么应在转义字符前面插入一个转义字符。当接收端收到连续的两个转义字符时，就删除其中前面的一个。
用字节填充法解决透明传输的问题

3. 流量控制

限制发送方的数据流量，使其发送的速率

必须使接收方来得及接收。高层也有流量控制。如运输层是端到端的流量控制，而本层是相邻两节点之间的流量控制。

4. 差错控制

①错误类型

位错

帧错：丢失、重复、顺序错

②控制方法

CRC 检错，ARQ 纠错

5.数据链路管理

建立、维持、释放数据链路（连接）

6. 寻址

在多点连接的情况下，必须保证每一帧都能送到正确的目的站。接收方也应当知道发送方是哪一个站。（用网卡地址）

3.4.3 数据链路层差错控制方法

3.4.4 基本链路控制规程

当两个主机进行通信时，发方和收方的数据链路层分别设有一个**发送缓冲区**和一个**接收缓冲区**。

在全双工通信，则在每一方都要同时设有发送缓冲区和接收缓冲区。

- 因为通信线路上数据是以比特流的形式串行传输的，在计算机内部数据的传输则是以字节（或多字节）为单位传输的。所以必须在计算机的内存中设置一定容量的缓冲区，以便解决数据传输速率不一致的矛盾。

不需要数据链路层协议的数据传输（完全理想化的数据传输）

在这种条件下不需要数据链路层协议

假定 1：链路是理想的传输信道，所传输的任何数据既**不会出差错**也**不会丢失**。

假定 2：不管发方以多快的速率发送数据，**收方总是来得及接收**，并及时上交主机。

满足上述**第二个条件的代价是**：接收缓冲区的容量为无限大，或者是接收速率与发送速率绝对精确相等。实际情况很难做到。

一、停－等机制

1.单工停—等协议（具有最简单流量控制的数据链路层协议）

- **保留第一个假定，去掉第二个假设**。为了使接收方的接收缓冲区在任何情况下都不会产生溢出，**就是发方每发送一帧就暂停下来，等待接收方发来的确认信号，然后才发下一帧。**
- **这种方法的流量控制是由收方控制的，也是计算机网络中流量控制的一个基本方法。**
- **优点**：简单；
- **缺点**：效率低。

2.具有最简单流量控制的算法

在发送节点：

(1) 从主机取一个数据帧。

(2) 将数据帧送到数据链路层的发送缓存。

(3) 将发送缓存中的数据帧发送出去。

(4) 等待。

(5) 若收到由接收节点发过来的信息(此信息

的格式与内容可由双方事先商定好)，则

从主机取一个新的数据帧，然后转到(2)。

具有最简单流量控制的算法（续）

在接收节点：

(1) 等待。

(2) 若收到由发送节点发过来的数据帧，

则将其放入数据链路层的接收缓存。

(3) 将接收缓存中的数据帧上交主机。

(4) 向发送节点发一信息，表示数据帧已

经上交给主机。

(5) 转到(1)。

两种情况的对比（传输均无差错）

3. 有噪声信道的停—等协议（实用的停止等待协议）

这种情况是**去掉上述两个假定**来考虑传输问题。

既然信道是不可靠的，在传输中就会出现错误。

(1) 数据帧出现差错

在数据帧中加上了循环冗余校验(CRC)，接收端很容易校验出收到的数据帧是否有差错。

(2) 数据帧丢失

数据帧丢失造成接收方接收不到信息，所以它不会向发送方发确认信号，这样发送方就不能发送下一帧。于是出现了死等现象，也就是**死锁**。

解决这一问题的办法是加设置超时定时器。

(3) 确认帧丢失

数据帧正确接收，返回的**确认帧丢失**，造成发方在规定的超时时间片内收不到确认信息。发方就会重发这一帧，因此接收方收到了**相同**

的两个帧，造成上一层收到**重复**的数据(重复帧)。

解决的办法是在发送方对**每一数据帧进行编号**。

停止等待协议的算法

■ 可不使用否认帧（实用的数据链路层协议大都是这样的），而且确认帧带有序号 n。

■ 按照习惯的表示法，ACKn 表示“第 n - 1 号帧已经收到，现在期望接收第 n 号帧”。

ACK1 表示“0 号帧已收到，现在期望接收的下一帧是 1 号帧”；

ACK0 表示“1 号帧已收到，现在期望接收的下一帧是 0 号帧”。

在发送节点：

(1) 从主机取一个数据帧，送交发送缓存。

(2) V(S)←0。 {发送状态变量初始化}

(3) N(S)←V(S)。 {将发送状态变量的数值写入发送序号}

(4) 将发送缓存中的数据帧发送出去。

(5) 设置超时计时器。

(6) 等待。 {等待以下(7)和(9)这三个事件中最先出现的一个}

(7) 收到确认帧 ACKn，

若 n = 1 - V(s)，则：

从主机取一个新的数据帧，放入发送缓存；

V(S)←[1 – V(S)]，{更新发送状态变量，变为下一个序号}

转到 (3)。

(8) 若收到否认帧 NAK，则转到(4) 。{重传数据帧}

(9) 若超时计时器时间到，则转到(4)。 {重传数据帧}

在接收节点：

(1) V(R)←0。{接收状态变量初始化，其数字等于欲接收的数据帧的发送序号}

(2)等待。

(3)当收到一个数据帧，就检查有无产生传输差错(CRC)

若结果正确，则执行后续算法；

否则转到(8)

(4) 收到一个数据帧；

若 N(S) = V(R)，则执行(5)；{收到发送序号正确的数据帧}

否则丢弃此数据帧，然后转到(7)。

(5) 将收到的数据帧中的数据部分送交上层软件

（也就是数据链路层模型中的主机）。

(6) V(R)←[1 – V(R)]{更新接收状态变量，准备接收下一个数据帧}

(7) n←V(R)；

发送确认帧 ACKn，转到(2)。

(8) 发送否认帧 NAKn，转到(2)。

停止等待机制时间关系

重传时间

超时重传：数据帧发送完毕后若经过了一段时间还没有收到应答帧，就重传这个数据帧。

超时时间为

$$t_{out} = t_p + t_{pr} + t_a + t_p + t_{pr}$$

假定 t_{pr} 和 t_a 都远小于 t_p，则

$$t_{out} \approx 2t_p$$

两个发送成功的数据帧之间的最小时间间隔是

$$t_T = t_f + t_{out} = t_f + 2t_p$$

信道利用率

没有传输错误，传输 N 帧所需要的时间为 $T_D = N t_T = N（t_f+2t_p）$

信道的利用率 η：

$$\eta = N t_f / T_D = N t_f / (N t_f + 2 N t_p) = t_f / (t_f+2t_p) = 1/(1+2\alpha)，$$

$$\alpha = t_p / t_f$$

吞吐量

数据帧**出错的概率为 p**，应答帧不出现错误，重传次数不受限制

正确传送一帧所需的平均时间 t_{av} 为

$t_{av}=t_T（1+$一个帧的平均重传次数$）$

一帧的平均重传次数=

{1×P[重传次数为 1]+2×P[重传次数为 2]

+3×P[重传次数为 3+]…}

={1×P[第 1 次发送出错]×P[第 2 次发送成功]

+2×P[第 1，2 次发送出错]×P[第 3 次发送成功]

…}

$=p(1-p)+2p^2(1-p)+3p^3(1-p)+\cdots$

$t_{av}=t_T+(1-p)\sum i p^i t_T=t_T/(1-p)$

$i=1,2,3,\cdots,\infty$

最大吞吐量 $\lambda max = 1/t_{av} = (1-p)/t_T$

二、连续 ARQ 机制（Go Back n）

在发送完一个数据帧后，不是停下来等待应答帧，而是可以**连续再发送若干个数据帧**。

接收端每次接收一个帧，且**按顺序接收**，收到正确帧后发送应答帧。

如果此时收到了接收端发来的应答帧，还可以接着发送数据帧。

发送端对**出错的数据帧进行重发是自动进行的**，所以这种差错控制体制简称为 ARQ（Automatic Repeat reQuest），意思是**自动请求重发。**

连续 ARQ 协议的工作原理

连续 ARQ：

(1) 接收端只按序接收数据帧。收到错帧后丢弃，重复发送已发送过的最后一个确认帧（防止确认帧丢失）。

(2) 发送节点在每发送完一个数据帧时都要设置该帧的超时计时器。如果收到确认帧，就将计时器清零。若超时而未收到确认帧，就重传相应的数据帧及其后续帧。

问题：连续发送多少帧（数据帧编号占开销）？

重发哪些帧？

三、滑动窗口的概念

发送窗口的概念：**发送窗口中的序列号代表已发送了的但尚未确认的帧**。如果发送窗口大小为 N，则需要有 N 个缓冲区来保存尚未确认的帧。

接收窗口的概念：**接收窗口的序列号是期望接收的帧**。任何不在窗口内序号的帧都要丢弃。

全双工信道数据流分析

在全双工通信中，数据是**双向通信**。从 A 机器到 B 机器的数据帧和从 A 到 B 的确认帧混在一起，接收方通过查看到达的帧头部的 kind 字段，区别是数据帧还是确认帧。

捎带技术：将**确认帧附加到发送的数据帧上。**

捎带技术优点：开销少，减少“帧到达”中断次数，占接收方的缓冲区少。

捎带技术缺点：捎带技术比单独确认复杂（链路层无法预知下一个分组何时到来），**等待时间不能超过发送方的时间间隔**，否则，帧会重发。采用折衷方法，**设置一个时间片，如果分组很快到来，就采用捎带技术，否则就单独发确认帧。**

接收端设置接收窗口

- 在接收端只有当收到的数据帧的发送序号**落入**接收窗口内才允许将该数据帧**收下**。
- 若接收到的数据帧落在接收窗口**之外**，则一律将其**丢弃**。
- 在连续 ARQ 协议中，接收窗口的大小 $W_R = 1$。

◆ 只有当收到的帧的序号与接收窗口一致时才能接收该帧。否则，就丢弃它。

◆ 每收到一个序号正确的帧，接收窗口就向前（即向右方）滑动一个帧的位置。同时发送对该帧的确认。

滑动窗口的重要特性

- 只有在接收窗口向前滑动时（与此同时也发送了确认），发送窗口才有可能向前滑动。
- 收发两端的窗口按照以上规律不断地向前滑动，因此这种协议又称为滑动窗口协议。
- 当发送窗口和接收窗口的大小都等于 1 时，就是停止等待协议。

发送窗口的最大值

- 当用 n 个比特进行编号时，若接收窗口的大小为 1，则只有在发送窗口的大小 $W_T \le 2^n - 1$ 时，连续 ARQ 协议才能正确运行。
- 例如，当采用 3 bit 编码时，发送窗口的最大值是 7 而不是 8。

1.一位滑动窗口

3.选择重传（Selective Repeat）协议

为提高信道的利用率，可只**重传出现差错的数据帧**或者是**计时器超时的数据帧**。一旦重传帧收到，就可与原先已收到的**暂时存放在缓冲区中的帧一起**，按正确的顺序送网络层，且**只对最高序号的帧**进行确认。

选择重传 ARQ 协议

- 可**加大接收窗口**，先收下发送序号**不连续**但仍处在接收窗口中的那些数据帧。等到所缺序号的数据帧收到后再一并送交主机。
- 选择重传 ARQ 协议可**避免重复传送那些本来已经正确**到达接收端的数据帧。
  - 付出的代价是在**接收端**要设置具有相当容量的缓存空间。**接收窗口的大小应大于 1。即：发、收窗口为 N。**
  - 发送方：一次发送 N 帧，等待应答。对 NAK 帧和超时帧重传，待 N 个帧都收到

ACK 后，窗口推进 N。

- 接收方：对收到的帧进行检错、排序，对每个帧给出应答（ACK/NAK）；N 帧都应答 ACK 后窗口推进 N。

■ 对于选择重传 ARQ 协议，若用 n 比特进行编号，则接收窗口的最大值受下式的约束：

$W_R + W_T \leq 2^n$ ，W_R最大值： $W_T = W_R = 2^n/2$

4. 部分重传（Go-Back N，GBN ）

4. 部分重传（Go-Back N，GBN ）

部分重传

窗口的变化时机

重传时只发送已发送的部分？

四、滑动窗口对效率的影响

四、滑动窗口对效率的影响

信道利用率

■ 由于每个数据帧都必须包括一定的**控制信息**(如帧的序号、地址、同步信息以及其他的一些控制信息)，即使连续不停地发送数据帧，信道利用率(扣除全部的控制信息后的数据率与信道容量之比)也不可能达到 100 %。

■ 当出现差错时(这是不可避免的)，数据帧的不断重传将进一步使信道利用率降低。

最佳帧长

■ 若数据帧的帧长取得很短，控制信息在每一帧中所占的比例增大，因而额外开销增大，导致信道利用率的下降。

■ 若帧长取得太长，数据帧在传输过程中出错的概率增大，重传次数增大，这也会使信道利用率下降。

■ 由此可见，存在一个最佳帧长，在此帧长下信道的利用率最高。

■ **信道利用率与帧长有关**

■ 最佳帧长在 1000～ 2000bit 之间

小结：

1.流量控制：是用于确保**发送实体发送的数据不会超出接收实体接收数据**能力的一种技术。

(1) 没有差错的流量控制：所有的帧都能成功地接收，而且这些帧在到达时也没有差错。

(2) 停止等待流量控制：可通过**目的站点发确认帧**来**控制源站点所发的数据流量**。为什么分割帧？

■ 接收方的缓存尺寸可能有限。

■ 传输的时间越长，产生差错的可能性越大，重传整个帧的可能性也越大。

■ 在局域网中，通常不希望让一个站点长时间地占有传输媒体，因为这样会导致其他发送站点的时延过长。

(3) 滑动窗口流量控制：滑动窗口流量控制比停止等待

流量控制能够达到**更高的有效性**。在滑动窗口流量控制

中，**传输链路被看作是一个管道**，它**有可能**在传输过程

中**被填满**。相反，**停止等待**流量控制中，管道中一次只

可能存在一个帧。

2.差错控制：采用自动重发请求(ARQ)机制，使不可靠

的数据链路变得可靠。

(1) 停止等待 ARQ 协议。

(2) 连续 ARQ 协议。

(3) 选择重传 ARQ 协议。

3.4.5 数据链路层协议

● **字符界定型（BSC/BISYNC）**

■ 二进制同步通信（BISYNC）是 **IBM 公司**于六十年代推出的一种通信方式。

■ BISYNC 是面向**字符**的，即传输的主要元素是字符而不是位流。

■ 先由两个同步字符，用于发送方与接收方的同步。

■ 头部开始（SOH）命令、头部，然后是文本开始（STX）命令、文本。

■ 最后是文本结束命令（EOT）及循环冗余校验（CRC）。

● **字节计数法（DDCMP）**

■ 数字数据通信报文协议 DDCMP（Digital Data Communications Message Protocol）。

■ 是 **DEC 公司推出**。

■ 从专门字段中获知该帧中随后跟随的数据字节数 。

面向字符的数据链路层协议有其许多**不足之处(BSC)：**

（1）控制规程与特定的字符编码集关系过于密切，兼容性较差。

（2）半双工的停-等协议(反馈重传)，传输效率较低，

即使物理链路可以支持全双工通信，BSC 仍然不能加以利用。

（3）只对数据部分进行差错控制，无

法检查控制部分的所有问题，可靠性较差。

针对上述的缺点，1974 年 **IBM 公司**在 SNA(System Network Architecture)的数据链路层规程采用了**面向比特**的规程 SDLC(Synchronous Data Link Control),后经 ISO 修改成为 HDLC(High-level Data Link Control)。

见 P84—谢（4）

支持任意二进制数据的传输常用的标准：

- **IBM 的同步数据链路控制规程(IBM SDLC)**
- **ISO 的高级数据链路控制规程(ISO HDLC)**
- **CCITT 的平衡型链路访问规程(CCITT X.25 LAP-B)**

一、HDLC

(IBM:SDLC->ISO:HDLC->CCITT:LAP->ISO:LAPB)

1. HDLC 的链路结构

①站类型

HDLC 通信方式中的站有三类:**主站、从(次)站**和**复合(组合)站**。

主站(Primary Station)：控制整个链路的工作，可发出命令来**确定**和**改变**链路的状态，包括确定次站、组织数据传输和链路恢复等。

从站(Secondary Station)：也称次站，指受主站控制，只能发出响应的站。

复合站(Combined Station)：具有主站和从站的功能。

其链路有两种基本配置，即**非平衡配置**和**平衡配置**。

③三种基本数据传送方式

正常响应方式 NRM：只有主站才能发起向从站的数据传输，而从站只有在主站向它发送命令帧时进行**轮询**(poll)时，才能以响应帧的形式回答主站。

异步响应方式 ARM：主站具有**初始链路，差错校正**和**逻辑拆链**功能；允许从站发起向主站的数据传输，即从站不需要等待主站发过来的命令，而是可以**主动向主站发送响应帧**。

异步平衡方式 ABM：每个复合站都可以平等地发起数据传输，而不需要得到对方复合站的允许。即任一组合站均可控制链路，主动传送数据。

2.HDLC 的帧结构　　P99—101(黄)

(1) 帧组成

(2) 各字段的意义

什么是帧同步？帧边界如何判断？

零比特填充方法。地址字段的填入方法。

- F：同步符号，“01111110”
- A：地址字段，可写入**从站**(非平衡链路)或**应答站**(平衡链路)的**地址。**全 1 是广播地址，全 0 是无效地址。地址可以扩充。

1 2 3 4 5 6 7 8　9 10　11　12….16 8n

- FCS：校验码，对 A、C、Info 字段进行循环校验。

$g(x)=x^{16}+x^{12}+x^5+1$　（CCITT 和 ISO 使用）；

$g(x)=x^{16}+x^{15}+x^2+1$（IBM 的 SDLC 使用）。

校验范围包含**地址字段、控制字段**和**信息字段**。由于帧中至少含有 A（地址）、C（控制）和 FCS（帧校验序列）字段，因此整个帧长度**应大于 32 位**。

- C：控制字段，确定 HDLC 帧的**类型**。分为三大类，即**信息帧、监督帧**和**无编号帧**，其简称分别是 I (Information)、S (Supervisory)和 U (Unnumbered)。

①信息帧(I)

传输用户数据，其控制字段的第 1 位规定为‘0’。

◆ N(S)：当前**发送的信息帧的序号**（采用模 8 计数），每发一帧，N(S)模 8 计数一次。

◆ N(R)：本站所**期望收到的帧的序号 N(R)**（采用模 8 计数）以及 **N(R) -1 号以前的帧已经收到**。有确认和捎带含义。

◆ P/F(Probe/Final)：探询/终止(比特)指示符。

- **正常响应模式(NRM)： 主站 P=1,询问次站是否有数据发送。**

如果次站有数据待发，开始发送信息帧（I）；次站可以连续发送多帧，并在最后一个 I 帧中，置 F=1,示意次站数据传输完毕。

如果次站无数据待发，直接以 S 帧（F=1）进行响应，示意本次站无数据可发。

- **其它传输模式（ARM 和 ABM）：P=1：要求对方做出响应，（建链时）**
  - **对方需立即进行响应，并在响应**

中置 F=1。（同意建链）

● P/F 总是一一对应的，在接到 F=1 的帧之前，不允许再发 P=1 的帧。

②监督帧(S)

用于**流量控制、差错控制和响应帧**，其控制字段的第 1、2 位规定为“10”；第 3、4 位表示了四种类型的监督帧。

四种类型的监督帧。

Type=00，接收准备就绪(RR)，准备接收编号为 N(R)的帧；确认序号为 N(R)－1 及其以前的各帧。

Type=10，未准备就绪(RNR)，告诉对方已经收到 N(R)以前的所有帧，但希望对方暂停接收下一帧；

Type=01，拒绝接收(REJ)，告诉对方已经收到 N(R)以前的所有帧，但编号为 N(R)的帧有差错，拒绝接收 N(R)及其后的所有帧，
希望对方重发编号为 N(R)及其以后的所有帧；

Type=11，选择接收(SREJ)，告诉对方已经收到 N(R)以前的所有帧，但希望对方仅仅重发第 N(R)帧。

③无编号帧(定义 32 种工作模式，目前定义了 18 种)

确定链路控制命令（建立/拆链）和工作模式,用于命令的传输，**起控制作用，不含序号**，其控制字段的第 1、2 位规定为“11”。

M1，M2 表示帧类型：

M(M1M2)=11000(SARM)，主站置本次链路为**异步响应模式**；

M(M1M2)=00001(SNRM)，主站置本次链路为**正常响应模式**；

M(M1M2)=11100(SABM)，某一复合站置本次链路为**异步平衡模式**；

M(M1M2)=00010(DISC)，请求**释放**(拆除)本次链路；

M(M1M2)=00110(UA)，次站对主站命令的**确认**，类似 BSC 中的 ACK；

M(M1M2)=10001(CMDR)，次站对主站命令的**否认**，类似 BSC 中的 NAK；

HDLC 例子-流量控制

HDLC 例子-出错重传

HDLC 规程小结：

特点：

（1）使用**统一的帧格式**：实现数据、命令和响应的传输，实施起来方便；

（2）采用‘0’位插入法：使得规程可以**支持任意的位流传输**，保证了信息传输的**透明性**；

（3）采用**窗口机制**和**捎带应答**：支持**全双工**工作方式，允许在未收到确认的情况下，连续发送多个帧，**提高**了信息传输的**效率**；

（4）采用**帧校验序列**，并设置窗口序号：可以提高信息传输的**正确性和可靠性。**

二、Internet 的数据链路层

1. SLIP (Serial Line IP) P88-谢(4)

面向字符协议 SLIP 是早期的**串行** IP 协议。**主要完成数据报的传送**，没有**寻址、数据校验、没有差错检测、识别**和**数据压缩**等功能，通信双方需**事先知道对方的 IP 地址**，只能传送 IP 分组，**而不支持其他协议**，对一些高层应用不支持，实现起来较简单。

帧结构:IP 包的**头尾**加上字节 0xC0。IP 包中的 0xC0 换成两个字节 0xDB、0xDC。

SLIP 缺点：

①无差错检测功能

②只支持 IP 协议

③必须预先知道对方的 IP 地址

④无身份验证功能

2. **PPP 协议**（PPP，Point-to-Point Protocol）**P103－107(黄)**

- 使用得最多的数据链路层协议之一。
- 远程孤立计算机接入的主要协议。

用户接入因特网一般有**两种方法**：

①用户使用远程拨号接入因特网;

②多台计算机通过驻在网(局域网代理)接入因特网。

用户使用 PPP 示意图

用户到 ISP 的链路使用 PPP 协议

(1) PPP 协议应满足的需求 **P103-104(黄)**

简单——这是首要的要求

封装成帧

透明性

多种网络层协议

多种类型链路

差错检测

检测连接状态

最大传送单元

网络层地址协商

数据压缩协商

(2) PPP 协议不提供的功能

纠错
流量控制
序号
多点线路
半双工或单工链路

不提供使用序号和确认的可靠传输

PPP 协议之所以不使用序号和确认机制是出于以下的考虑：

在数据链路层出现差错的概率不大时，使用比较简单的 PPP 协议较为合理。

在因特网环境下，PPP 的信息字段放入的数据是 IP 数据报。数据链路层的可靠传输并不能够保证网络层的传输也是可靠的。

帧检验序列 FCS 字段可保证无差错接受。

(3) PPP 协议的组成

■ 1992 年制订了 PPP 协议。经过 1993 年和 1994 年的修订，成为 Internet 正式标准[RFC 1661]。

■ PPP 协议有三个组成部分

①**高级数据链路控制(HDLC)协议**：PPP 采用 HDLC 协议作为在线路上的**数据编码协议**。并将 IP 数据报封装到串行链路的方法。

②**链路控制协议(LCP)**：用来**建立、配置**和**测试**数据链路。

③**网络控制协议(NCP)**：PPP 允许同时**采用多种网络层协议**，每个不同的网络层协议要用一个相应的 NCP 来配置，为网络层协议建立和配置逻辑连接。

(4) PPP 协议的帧格式

了解 PPP 协议的帧格式与 HDLC 协议帧格式的相同点和不同点。

标志字段 F = 0x7E （十六进制的 7E 的二进制表示是 01111110）。

地址字段 A 只置为 0xFF。地址字段实际上并不起作用。

控制字段 C 通常置为 0x03，表示不使用编号。

PPP 是面向字节的，所有的 PPP 帧的长度都是整数字节。

(5)透明性问题

■ 当 PPP 用在**同步传输**链路时（在 SONET/SDH 链路），协议规定采用硬件来完成比特填充（和 HDLC 的做法一样）。

■ 当 PPP 用在**异步传输**时，就使用一种特殊的字符填充法。

◆ 将信息字段中出现的每一个 0x7E 字节转变成为 2 字节序列(0x7D, 0x5E)。

◆ 若信息字段中出现一个 0x7D 的字节， 则将其转变成为 2 字节序列(0x7D, 0x5D)。

◆ 若信息字段中出现 ASCII 码的控制字符（即数值小于 0x20 的字符），则在该字符前面要加入一个 0x7D 字节，同时将该字符的编码加以改变。

(6) PPP 协议的工作过程 P106－(黄)

① 用户拨号接入 ISP，ISP 的 MODEM 响应，建立物理连接。

② 用户机用 PPP 帧向 ISP 的路由器发一系列 LCP 包（封装成多个 PPP 帧），确定 PPP 参数，建立 LCP 连接。

③ NCP 为新接入的用户机分配一个临时 IP 地址。

④ 用户机成为 Internet 上的主机，开始通信。

⑤ 用户通信完毕，NCP 释放网络连接，收回 IP 地址。

⑥ LCP 释放数据链路层连接。

⑦ 释放物理连接。

由于 PPP 协议不提供使用序号和确认的可靠传输手段，因此，在信道质量较差的场合(如无线通信)，则应使用有编号的确认方式。

(6) PPP 协议的工作过程

■ **PPP 协议的新的应用**

◆ 宽带接入正在成为取代拨号上网的趋势，在宽带接入技术日新月异的今天，PPP 也衍生出新的应用。

◆ 典型的应用是在 ADSL（非对称数据用户环线，Asymmetrical Digital Subscriber Loop）接入方式当中，PPP 与其他的协议共同派生出了符合宽带接入要求的新的协议。

◆ 如 **PPPoE**（PPP over Ethernet）P193－(黄)，**PPPoA**（PPP over ATM）。

◆ PPPoE 即**保护**了**用户方**的**以太网资源**，又**完成了 ADSL 的接入要求。**

3. **动态主机配置协议(DHCP)**

DHCP （dynamic host configuration protocol）就是**动态主机配置协议**，它用于给某个网络段上的主机进行动态分配 IP 地址和相关网络环境的配置工作，比如我们使用 ADSL

拨号就是用的 DHCP 协议。

三、使用广播信道的数据链路层

局域网的数据链路层

局域网最主要的特点是：网络为一个单位所拥有，且地理范围和站点数目均有限。

局域网具有如下的一些主要优点：

具有广播功能，从一个站点可很方便地访问全网。局域网上的主机可共享连接在局域网上的各种硬件和软件资源。

便于系统的扩展和逐渐地演变，各设备的位置可灵活调整和改变。

提高了系统的可靠性、可用性和生存性。

1. 局域网的拓扑

2. 媒体共享技术

计算机网络中使用的信道共享技术可分为两种：

静态划分信道：使用集中器或复用器

频分复用

时分复用

波分复用

码分复用

动态媒体接入控制（多点接入）

随机接入，（包括不分时隙或分时隙）：ALOHA、CSMA 和 CSMA/CD。

受控接入 ，如多点线路探询(polling)，或轮询。

(1) 多点接入技术的分类：P94—谢(4)

A. 受控接入

受控接入的特点是用户**不能任意接入信道**而必须服从一定的控制。

集中式控制的有多点线路**轮询**(polling)，主机按一定顺序逐个询问各用户有无信息发送，如有，则被询问的用户就立即将信息发给主机；如无，则再询问下一站。

分散式控制的有令牌环网。只有获得令牌的站才有权发送信息，当信息发送完毕后，即将令牌传递给下一站。

B.随机接入

随机接入的特点是所有的用户都可以根据自己的意愿**随机**的发送信息。随机接入实际上就是**争用接入**，争用胜利者可获得总线(即信道)，从而获得信息的发送权。

3. 随机接入技术

(1) 纯 ALOHA：P420－423 图 B-2 谢(4)

■ **纯 ALOHA(**Additive Link On-line Hawaii system**)的工作原理：**

所有的用户都可以**随机的发送信息**。当两个或更多的用户同时发送信息时会产生了**冲突**（collision）。冲突的结果是冲突的双方(也可能是多方)所发送的数据都出现差错，必须**重传**。ALOHA 采用的重传策略是**并不能马上进行重传，让各站等待一段随机时间，然后再进行重传**。如果再发生冲突，则需再等待一段随机时间，直到重传成功为止。

■ **纯 ALOHA 一帧发送成功的条件：**

帧的长度不是用比特而是用发送该帧所需的时间来表示。用 T_0 来表示这段时间。

一帧欲发送成功，必须在该帧发送时刻之前和之后各一段时间 T_0 内（一共有 $2T_0$ 的时间间隔)，没有其它帧发送。否则就必产生冲突而导致发送失败。参见 P420--图 B-2 谢(4)

■ **纯 ALOHA 的主要性能**

吞吐量和平均时延,下面我们就来分析两者的计算。

A .吞吐量 S：等于在帧的发送时间 T_0 内成功发送的平均帧数。$0 \leqslant S \leqslant 1$，S=1 是极限值。

B. 网络负载(offered load)G ：等于在 T_0 内总共发送的平均帧数。这里包括发送成功的帧和因冲突未发送成功而重传的帧。

显然 $G \geqslant S$，而只有在**不发生冲突**时，**G 才等于 S**。

C.吞吐量和网络负载的关系

在**稳定状态**下，吞吐量 S 与网络负载 G 的关系为:

$$S=G\times P[\text{发送成功}] \quad 3\text{-}1$$

P[发送成功]是**发送成功**的帧在所发送的帧的总数中所占的**比例**。

$$P[\text{发送成功}]=P[\text{连续两个到达间隔}>T_0]=(P[\text{到达间隔}>T_0])\times(P[\text{到达间隔}>T_0])=(P[\text{到达间隔}>T_0])^2 \quad 3\text{-}2$$

假定帧的达到是泊松分布过程，所以到达时间间隔的概率密度为

$$(t)=\lambda e^{\lambda t} \quad 3\text{-}3$$

其中 λ 是帧的平均到达率,在这里 $\lambda=G/T_0$，于是可得到

$$P[\text{到达时间间隔}>T_0]= \quad 3\text{-}4$$

将 4-4 代入 4-2 中得出 **P[发送成功]= e^{2G}**

3-5

再将 4-5 代入 4-1 中得出

$S=G\times e^{2G}$

3-6

当 G=0.5 时，S=0.5e^{1}≈0.184 。这时吞吐量 S 可能达到的极大值。

P422--谢（4）图 B-3 该图说明 G=0.5 时，S≈0.184 是吞吐量 S 的极大值。G>0.5 时，吞吐量下降。**在纯 ALOHA 系统中，网络负载 G 一定不能超过 0.5** 。

一个理想随机接入系统的吞吐量 S 的极限值是 1。但纯 ALOHA 的吞吐量的极大值只能达到理想值的 18.4% 。但在实际情况下，为安全起见，纯 ALOHA 的吞吐量 S 不应超过 10 % 。

(2) 时隙 ALOHA(S-ALOHA)

■ **工作原理**：为了提高吞吐量，可将所有各站在时间上同步起来，并将时间划分为一段段等长的时隙（slot），记为 T_0，同时规定，不论帧在何时产生（即到达一个站），它只能在每个时隙开始时才能发送出去。这样的 ALOHA 系统叫做**时隙 ALOHA** 或 S-ALOHA。P423-424 谢(4)

(3)**性能分析**：在时隙 ALOHA 系统中**冲突危险区为原来的一半**。(P90--谢(2))

P[发送成功]= P[到达时间间隔>T_0－T_X]×P[到达时间间隔>T_X]

3-9

或者 P[在 T_0的时间内有 0 个帧到达]

对上式进行积分并代入 4-3 式得：

P[发送成功]=e^{-G} 3-10

将它代入 4-1 最后得：

$S=Ge^{-G}$ 3-11

参见 P424--谢（4）图 B-6

当 G=1 时，S≈0.368 。

P424--425 谢（4）图 B-7

(4) CSMA（Carrier Sense Multiple Access 载波监听多点接入）

CSMA 是在 ALOHA 的基础上提出来的。

工作原理：每个站在**发送数据前**都要**监听**信道上是否有数据正在发送，如果有数据在信道上发送，则该站不发送；否则该站就发送数据。这样做减少了冲突次数。提高了整个系统的吞吐量。

注意：由于信号在信道上以有限速度传输，所以采用载波监听并不能完全消除冲突。

CSMA 与 ALOHA 的主要区别：多了一个载波监听装置，这种装置提供的功能通常称为**发送前监听**。

(5) 几种类型监听方式

● **非坚持型 CSMA**

● **坚持型 CSMA**

● **P 坚持型 CSMA**

■ **非坚持型 CSMA**

工作原理：

站点在发送数据前，先监听信道上是否有站点在发送数据，如果没有，则立即发送数据；**否则延迟一段时间再监听信道**的状态看能否发送？重复上述过程，直到数据发送出去。

缺点：

一旦监听到信道忙就马上延迟一个随机时间再重新监听，但很有可能在再次监听之前信道就已经空闲了。**也就是说，非坚持 CSMA 不能把信道刚一变成空闲的时刻找出。**这就影响了信道利用率的提高。

为了克服这一缺点，可采用坚持型 CSMA。

■ **坚持型 CSMA（1 坚持）**

工作原理：

站点在发送数据前，先监听信道上是否有站点在发送数据，如果没有，则**立即发送**；在监听到信道忙时，**仍坚持听下去**，一直坚持听到信道空闲为止。

缺点：

如果有**两个或更多的站同时**在监听信道，则一旦信道空闲就必然使这些同时发送的帧相互冲突。反而不利于吞吐量的提高。

■ ***P* 坚持型 CSMA 的工作原理**

站点在发送数据前，先监听信道上是否有站点在发送数据，当听到**信道空闲**时，**就以概率 *P* 发送数据**，而以概率（1－*P*）延迟一段时间 τ（端到端的传播延迟），重新监听信道。

第 3 种是对上述两种协议的折衷。

这种协议在具体执行时，**选择一个 0~1 之间的随机数 *I***。若 *I*≤*P*（*P* 的大小事先给定），则发送数据，否则延迟时间 τ 后再重新监听信道。

***P*值是根据信道上的通信量的多少设定的**。

CSMA 有时隙 CSMA，它和时隙 ALOHA 相似，

需要全网同步。

(6) CSMA/CD 协议(Carrier Sense Multiple Access with Collision Detection 载波监听多点接入/碰撞检测) P97－99 谢(4)， P138(黄)

总线的特点是： 通信方式是**广播通信**。在发送数据帧时，**在帧的首部写明接收站的地址**。仅当数据帧中的目的地址与计算机的地址一致时，该计算机才能接收这个数据帧。**计算机对不是发送给自己的数据帧，则一律不接收。以太网的广播方式发送**

总线上的每一个工作的计算机都能检测到 B 发送的数据信号。

由于只有计算机 D 的地址与数据帧首部写入的地址一致，因此只有 D 才接收这个数据帧。其他所有的计算机（A, C 和 E）都检测到不是发送给它们的数据帧，因此就丢弃这个数据帧而不能够收下来。

具有广播特性的总线上实现了一对一的通信。

以太网采取了**两种方法**。

第一，采用无连接的工作方式，即**不必先建立连接**就可以**直接发送数据**。

第二，不要求收到数据的**目的站发回确认**。理由是局域网信道的质量很好，差错的概率很小。因此，**以太网提供的服务是不可靠的交付，即尽最大努力的交付。**

如何协调总线上各计算机的工作，**使在同一时间只能允许一台计算机发送信息**，以太网采用的协调方法是使用一种特殊的协议，**CSMA/CD 协议**。

CSMA/CD 比 CSMA 又增加了一个功能，即**边发送边监听，只要监听到发生冲突，则冲突的双方就必须停止发送。**

■ **工作原理：**

站点在发送数据**前**，先**监听**信道上是否有站点在发送数据，当**听到信道空闲**时，就**发送数据，并继续监听信道**。如果监听到发生了冲突，则立即放弃本次数据帧的发送。

协议**增加了边发送边监听的功能**，只要监听到发生了冲突，则冲突双方就必须停止发送，这样，信道就很快空闲下来。然后**等待一段**随机时间后**再次**发送，从而提高了信道的利用率。“载波监听”是指每一个站在发送数据之前先要检测一下总线上是否有其他计算机在发送数据，如果有，则暂时不要发送数据，以免发生碰撞。

总线上并没有什么“载波”。因此，“载波监听”就是用电子技术检测总线上有没有其他计算机发送的数据信号。

■ **冲突检测的方法：** P97－谢(4)

- **比较接收到的信号的电压大小：接收到的信号的电压摆动值超过某一门限值**，就可认为是**发生了冲突**。
- 采用**曼彻斯特编码的电压过零点位置是否在正中央**来判断是否发生了冲突。不在正中央证明发生了冲突，否则,没有冲突发生。
- 在发送帧的同时也进行接收，将**收到的信号逐比特地与发送的比特进**行**比较**。若有不符的，就说明发生了冲突。

使每个站都能及早正确地判断是否发生了冲突，往往采用一种叫做**强化冲突的措施**。以便让所有用户都知道现在已发生了冲突。P98－谢(4)

那么为什么还会出现数据在信道上的碰撞(冲突)？

当某个站监听到总线是空闲时，也可能总线并非真正是空闲的。

A 向 B 发出的信息，要经过一定的时间后才能传送到 B。

B 若在 A 发送的信息到达 B 之前发送自己的帧(因为这时 B 的载波监听检测不到 A 所发送的信息)，则必然要在某个时间和 A 发送的帧发生碰撞。

碰撞的结果是两个帧都变得无用。

在使用 CSMA/CD 协议时，一个站**不可能同时进行发送和接收**。因此使用 CSMA/CD 协议的以太网**不可能进行全双工通信**而只能进行**半双工通信**。

每一个站在自己发送数据之后的一小段时间内，存在着遭遇碰撞的可能性。以太网的这一特点称为发送的**不确定性**。这段时间**最多是两倍的总线端到端的传播时延，即** 2τ 。

(6) 什么是争用期？

在每个站发送数据刚刚开始的一个**很短的时间**由于电磁波在网上传输需要时间，因此冲突仍有可能发生。我们将这段**可能发生冲突的时间间隔称为争用期**(contention period)，争用期又称为**碰撞窗口**(collision window)。

将以太网的端到端往返时延 2τ ($\delta \rightarrow 0$)称为**争用期**。**经过争用期这段时间还没有检测到**

碰撞，才能肯定这次发送不会发生碰撞。

为了保证系统的稳定性，可以采用以太网所用的**截断二进制指数类型**(truncated binary exponential type)的**退避算法**来**决定重传帧的延迟时间**。在发生碰撞后，**停止的站不是马上发送数据**，而是**推迟**(退避)一个随机的时间。这种**退避算法是：**

- 确定基本退避时间，一般是取为争用期 2τ。
- 定义重传次数 k ，$k \le 10$，即

  k = Min[重传次数, 10]
- 从整数集合[0,1,…, (2^k-1)]中随机地取出一个数，记为 r。

重传所需的时延就是 r 倍的基本退避时间。

- 当重传达 16 次仍不能成功时即丢弃该帧，并向高层报告。

P99－谢(4)

帧的最小长度的要求

整个帧的**发送时间**应当**不小于**信号在网中**“传播距离最大”**的两个节点之间**传播时间**的两倍。

最小帧长目的： 保证发送节点可以对发送的冲突进行有效的冲突检测。

帧实际传输时间的估算：

帧从节点到媒体的时间 + 媒体上传输的时间 + 转发器的处理时间

以 IEEE802.3 定义的标准 CSMA/CD 网络 10BASE5 为例。基带传输、速率 10Mbps、采用粗同轴电缆、单段缆线最长 500 米、最多允许 5 段。

1、信号发送到节点的时间约 20 比特；

2、从信号到媒体的时间（50 米）约 2.5 比特；

3、节点 MAU(Media Attachment Unit 媒体连接单元)转发时间约 20 比特；

4、2500 米传输所需时间约 125 比特；

5、4 个转发器转发处理时间约 80 比特（20 比特/转发器）。

发送节点从发送信号到检测出冲突的总时间为上述各时间值之和的 2 倍，即：

(20+2.5+20+125+80) ×2 = 512 (比特时间)

即为了保证发送节点在完成发送任务之前可以发现冲突，帧的最小长度规定为 512 位（64 字节），或者可填充的 PAD(填充字段) 字段的最大长度为 46 字节。

[例题]**假定 1km 长的 CSMA/CD 网络的数据率为 1Gbit/s 。设信号在网络上的传播速率为 200000km/s。求能够使用此协议的最短帧长。**

答：对于 1km 电缆，单程端到端传播时延为：

$\tau = 1 \div 200000 = 5 \times 10^{-6} s = 5 \mu s$，

端到端往返时延为：$2\tau = 10 \mu s$

为了能按照 CSMA/CD 工作，最小帧的发送时延不能小于 10μs,以 1Gb/s 速率工作,10μs 可发送的比特数等于：$10 \times 10^{-6} \times 1 \times 10^9 = 10000 bit = 1250$ 字节。

CSMA/CD 工作原理小结 ：

① **发送前监听**。监听线路使用情况，如线路为空，则发送数据。

② **边发边听**。在发送数据的同时，检测是否有冲突发生。

③ **发生冲突立即停止发送**。在发送过程中如果检测到冲突，立即停止发送数据。

④ **发送增强冲突信号**。停止发送数据后，向线路上发送增强冲突信号。

⑤ **退避算法**。执行退避算法，等待一定时间后，再尝试发送。

二、协议

1. RS(EIA)-232-C/ CCITT V.24 接口标准 P109(黄)

(1) 用途：DTE－DCE(MODEM)

(2) 概况：25 针，

数据信号：+12V/+8V＝0, -12V/-8V＝1

控制信号：0=ON, 1=OFF

数据率：0~20Kbps

电缆长度：15m

分为主信道和辅信道

(3) 主信道的连接

EIA-232-E 标准规定的常用的 10 根引脚

(4)简化接口

9 针

(5)空 MODEM：3 对针

(6)工作过程

DTE 就绪→MODEM 就绪→请求发送→允许发送→发数据

→清请求发送→清允许发送

→清 DTE 就绪→清 MODEM 就绪

计算机－MODEM 的连接
两个 DTE 通过 DCE
进行通信的例子
完整过程
利用虚调制解调器
与两台计算机相连
2. RS-449/CCITT V.36 接口标准　　P65－66
谢（4）

EIA-232 接口的弱点：

- 数据的传输速率最高为 20kb/s；
- 连接电缆的最大长度不超过 15m。

(1)由 RS－449、RS－423A、RS－422A 组成

(2)RS-449：规定机械、功能、规程特性。37 针

(3)RS-423A：非平衡模式（共用公共地）的电气特性。10m 时，300Kbps

(4)RS-422A：平衡模式（没有公共地）的电气特性。60m 时，2Mbps；10m 时，10Mbps

- RS-449 的机械特性规定使用 37 芯引脚。
- RS-449 的电气特性涉及 RS-423-A 和 RS-422-A 两个标准

RS-423-A 规定了接口采用差动接收的非平衡电气连接方式的电气特性。信号电平采用±6V 的负逻辑，－4V~+4V 的过渡区，使传输距离和速度比 RS-232 有较大的提高，当传输距离为 100m 时，速率为 10kb/s；而距离为 10m 时，速率为 300kb/s。

RS-422-A 规定了接口采用平衡电气连接方式的电气特性，因采用双线平衡传输，大大地提高了抗干扰性能。又由于信号电平采用±6V 的负逻辑，－2V~+2V 的过渡区。传输距离为 1000m 时，速率为 100kb/s；而在距离为 10m 时，速率可达到 10Mb/s。

- RS-449 的功能特性对 30 条信号线作了功能性的定义。与 RS-232 相比，新增的信号线主要是为了解决环回测试和其他功能的问题。信号线包括：发送公共回线（SC）、接收公共回线（RC）、本地还路返回（LL）、远程环路返回（RL）和测试模式（TM）等。
- RS-449 的规程特性沿用了 RS-232 的规程特性。

3. PS/2 接口

最早在 IBM 的 PS/2 型的机子上，因而得名。这是一种鼠标和键盘的专用接口，6 针的圆型接口，但键盘只使用其中的 4 针传输数据和供电，其余 2 个为空脚。PS/2 接口的传输速率比 COM 接口稍快一些，而且是 ATX 主板的标准接口。

5. USB (Universal Serial Bus)通用串行总线

由康柏、DEC、IBM、NEC、Intel、微软和加拿大北方电讯公司等七家厂商研制的一种总线规范。用来简化输入装置的安装程序，并且能让电话与计算机相互连通。1995 年 11 月，USB 0.9 规格正式提出， 1996 年，可以支持 USB 标准的 PC 机问世，该总线目前的带宽为 1.5Mbps。2000 年发布的 USB2.0 支持的总线速度比 USB1.x 快 40 倍。

PS/2 接口和 USB 接口的键盘在使用方面差别不大，USB 接口支持热插拔， USB 接口键盘在使用中略方便一些。但是计算机底层硬件对 PS/2 接口支持的更完善一些，因此如果电脑遇到某些故障，使用 PS/2 接口的键盘兼容性更好一些。

P112　5，7，12，14，15