

大纲要求

Requirements



- **01** 掌握数据链路层的基本概念; 了解数据链路的功能
- 02 掌握流量控制的概念,分析停止-等待,滑动窗口控制机制
- 03 理解点对点数据链路层协议的相关概念
- **04** 理解多路访问信道的常用介质访问控制方法原理

内容纲要 Contents Page



- 01 数据链路层的基本概念
- 02 流量控制和差错控制
- 03 点对点信道的数据链路层协议
- 04 多路访问信道的数据链路层

过渡页 Transition Page



- 01 数据链路层的基本概念
- 02 流量控制和差错控制
- 03 点对点信道的数据链路层协议
- 04 多路访问信道的数据链路层









数据链路的结构



数据链路层的功能



协议数据单元



3.1.1 数据电路和数据链路 >>



数据链路层在物理连接提供的比特流传输 服务基础上,通过一系列通信控制机制,构成 透明的、相对无差错的数据链路,实现可靠、 有效的数据传送。

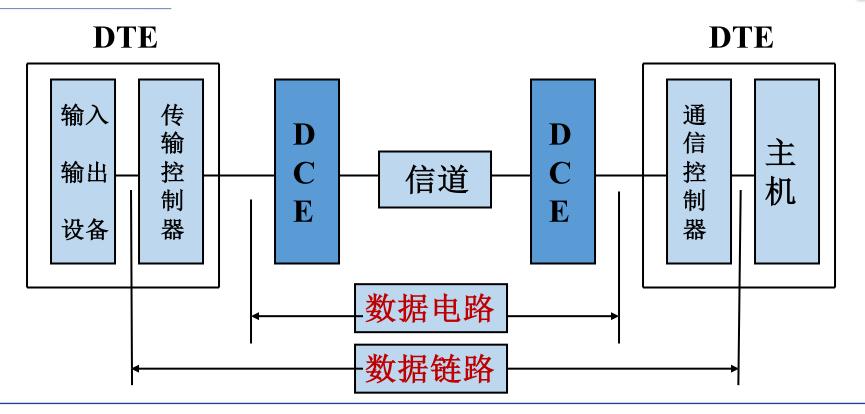
流量控制

数据链路层协议负责在物理网络与计算机 的协议组之间提供一个接口。









DTE: 数据终端设备,输入输出,通信控制。

流量控制

DCE: 数据电路设备, 传输信号的变换, 信号控制。



3.1.1 数据电路和数据链路 >>



信道: 信息传输的通路(模拟或数字信道)

数据电路:在传输信道两端加上信号变换设备 之后所形成的二进制比特流通路。即数据电路 由传输信道加上DCE组成,实现数据信号的 传输。

数据链路:在数据电路建立的基础上,在链路 协议控制下,使通信双方正确传输数据的终端 设备与传输线路的组合体。



3.1.1 数据电路和数据链路 >>



数据链路与数据电路

数据电路又常称为物理链路或链路

流量控制

数据链路又称为逻辑链路

数据链路是在数据电路上增加了传输控制功能 实现的。

只有建立数据链路,才能真正实现数据通信。







链路上所连接的节点称为"站"。

流量控制

- 发送命令或信息的站称为"主站",在通 信过程中起控制作用:接收数据或命令, 并做出响应的站称为"从站",在通信过 程中处于受控地位。
- 同时具有主站和从站功能的,能够发出命 今和响应信息的站称为"复合站"。







数据链路的基本结构(线路拓扑)

流量控制

点到点链路(平衡型,不平衡型)

多点链路(平衡型,不平衡型)

数据链路层的传送方式:

单工通信

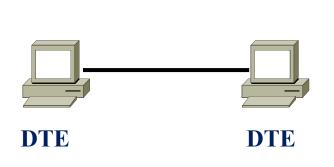
半双工通信(Half Duplex)

全双工通信(Full Duplex)

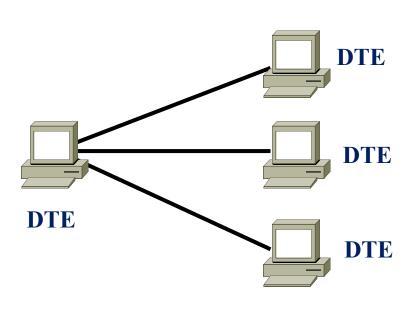


3.1.2 数据链路的结构 >>









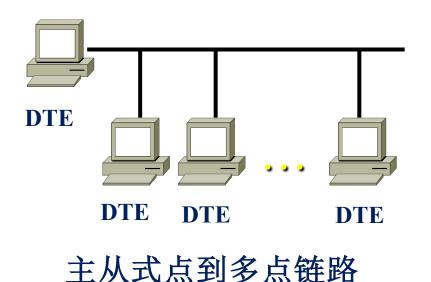
星型点到点链路

在点到点链路中,两端的站可能是 主站、从站 或 复 链路可以是不平衡结构,或平衡结构。

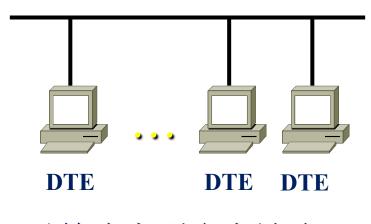


3.1.2 数据链路的结构 >>





流量控制



对等式点到多点链路

在主从式点到多点链路中,常常使用不平衡链路; 在对等式点到多点链路中,常使用平衡型链路。







数据链路层的目标:在数据链路上提供相对 可靠的信息传输。

流量控制

- 数据链路层的功能:
 - 链路管理
 - 帧同步(帧定界)
 - 流量控制
 - 数据和控制信息的识别

- 寻址
- 透明传输
- 差错控制



本概念

3.1.4 协议数据单元 >>

- 物理层的任务是尽可能实现比特流的可靠传 送,但不能保证没有错误,需要数据链路层 进行差错检测和纠正。

流量控制

- 为了便于实现流量控制和差错控制, 路层将较长比特流分解成多段离散的"段" 独立地发送、接收和处理。
- 这些"段"在数据链路层即称为"帧"
- 数据链路层的协议数据单元PDU,即传输 和处理的数据单位: 帧。







必须有相应的方法表明各个部分的信息。

流量控制

• 帧的构成方法:

- 字符计数法
- 含字节填充的分界符法
- 含位填充的分界标志法
- 物理层编码违例法





多路访问信道的数据链路层

数据链路层的帧形成后,可以通过相关手段 实现帧的定界(帧同步,字符同步)。

流量控制

- 帧同步方法:数据信号的识别,特定字符或 比特流的识别(分界符),外部信号状态的 识别等。
- 还需要识别传输中产生的差错、帧丢失,收 发双方的协调等。

过渡页 Transition Page



- 01 数据链路层的基本概念
- 02 流量控制和差错控制
- 03 点对点信道的数据链路层协议
- 04 多路访问信道的数据链路层





数据链路层基本概念







停止-等待流量控制方式



滑动窗口流量控制方式



连续ARQ方式



选择ARQ方式



3.2.1 流量控制的基本概念



- 在网状拓扑的各个网络节点之间的通信,数 据分组(数据包)采用存储—转发方式传输。
- 由于通信的随机性,网络中某个节点有可能 同时收到来自多个方面大流量数据传送,同 时节点处理能力的差异,可能导致接收处理 和存储能力不足。
- 设置一定的缓冲区可以调节收发双方速率不 匹配的情况,但不能从根本上解决问题。



3.2.1 流量控制的基本概念



- 由于接收方缓冲能力有限,如果不及时采取 控制措施,可能会出现接收方来不及接收和 处理数据的情况,会造成数据丢失。
- 当接收方来不及缓存和处理数据时,就必须 采取相应的流量控制措施来限制发送方发送 数据的谏率。
- 注意: 物理层的通信双方数据信号收发速率 必须相同,而链路层的收发速率不匹配主要 指数据处理能力方面。





3.2.1 流量控制的基本概念 >>

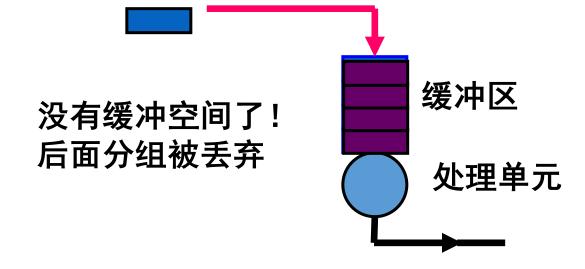
点对点信道数据链路层协议



流量控制的目的:

数据链路层基本概念

现代数据通信传输,大多数采用了存储转发的分组 交换技术,由于通信的随机性和突发性,当接收方 的处理能力小于发送方的发送量时,必须采用流量 控制。





数据链路层基本概念

3.2.1 流量控制的基本概念 >>

点对点信道数据链路层协议



- 在数据通信中,发送方将数据送入链路层发 送缓冲区,以一定数据速率发送,经过链路 传输到达接收方缓冲区,接收方链路层处理 后,将数据上交。
- 在通信过程中,要求发送方的发送数据速率 不能超过接收方接收和处理数据的速率。
- 理想情况下收发双方能够很好协调通信。

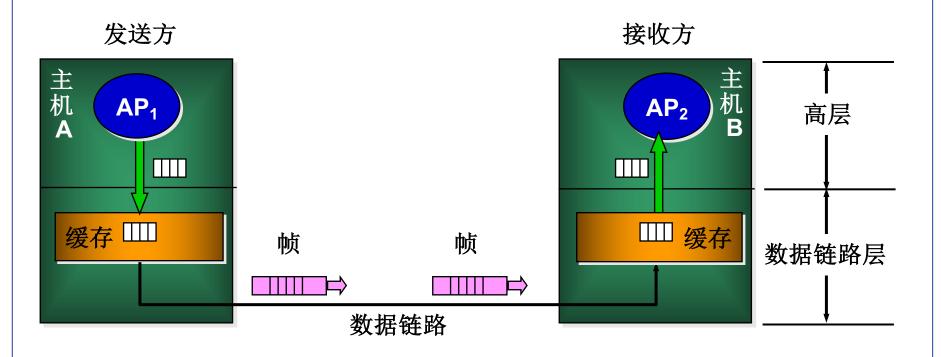






设主机A,B为数据链路层的高层用户完成数据收发, 物理层能够实现数据正确、可靠传送。

流量控制





数据链路层基本概念

3.2.1 流量控制的基本概念 >>

点对点信道数据链路层协议



完全理想化的数据传输不需要控制,其基 于两个假定:

假定1:链路是理想的传输信道,所传送的任何数 据既不会出差错也不会丢失。即数据能够按照发送 顺序正确到达接收方。

假定2:不管发送方以多快的速率发送数据。 方总是来得及收下,并及时上交主机。即接收方总 是处于接收准备好状态。



3.2.1 流量控制的基本概念 >>



- 实际应用中,上述的理想条件往往不能满足。
- 保留上述的第一个假定,即主机 A 向主机 B 传输数据的信道仍然是无差错的理想信道。 但不能保证接收端向主机交付数据的速率永 远不低于发送端发送数据的速率,则需要流 量控制,即控制发送的数据速率。
- 由收方控制发方的数据流,是计算机网络中 流量控制的一个基本方法。



数据链路层基本概念

3.2.1 流量控制的基本概念

点对点信道数据链路层协议



所谓流量控制方法,即接收方如何通知发 送方 "开始 or 停止"发送数据。

- 开关式流量控制: XON/XOFF, 硬件控制信号
- 协议式流量控制: ARQ自动重发请求
 - · 停止等待协议(Stop-and-Wait Protocol)
 - 滑动窗口协议(Slide Window Protocol)
 - (1) 连续ARQ协议
 - (2) 选择ARQ协议



数据链路层基本概念

3.2.2 停止-等待流量控制

点对点信道数据链路层协议



简单的停止-等待流量控制

发送结点

- 从主机高层取出一个数据帧:
- 将数据帧送到数据链路层的发送缓存:
- 将发送缓存中的数据帧发送出去: (3)
- (4)(等待什么?)
- 若收到由接收结点发过来的确认应答 (5)信息,则从主机取一个新的数据帧, 然后转到(2)。



数据链路层基本概念

3.2.2 停止-等待流量控制

点对点信道数据链路层协议



简单的停止-等待流量控制

接收结点

- 等待: (等待什么?)
- 若收到由发送结点发过来的数据帧, 则将其放入数据链路层的接收缓存:
- 将接收缓存中的数据帧上交主机: (3)
- 向发送结点发送一个确认信息,表示 (4)数据帧已经上交给主机高层:
- 转到(1)。 **(5)**

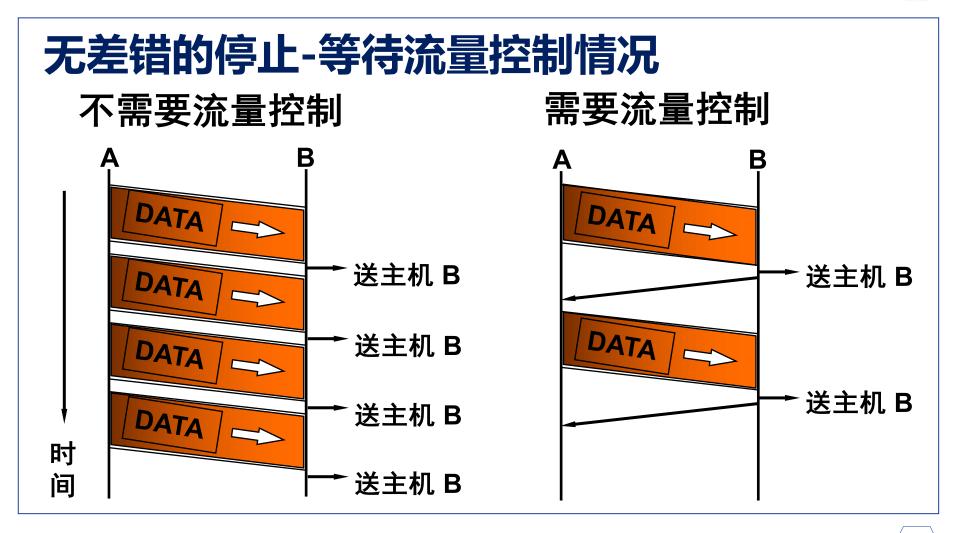


数据链路层基本概念

3.2.2 停止-等待流量控制 >>>

点对点信道数据链路层协议







3.2.2 停止-等待流量控制

点对点信道数据链路层协议



实用的停止-等待协议

- 在实际的数据传输过程中,由于传输信道 特性设计和实现的不理想和外界干扰的存 在, 出现传输差错是不可避免的。
- 传输差错导致接收的数据帧错误,接收方 要求发送方重发数据帧。
- 严重的传输差错还导致数据帧或应答帧丢 失, 使发送操作不能继续进行, 或接收方 重复接收数据。

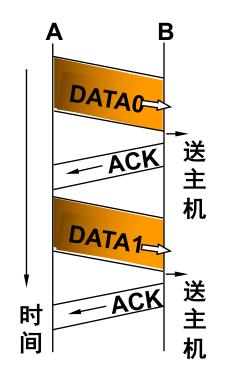


3.2.2 停止-等待流量控制

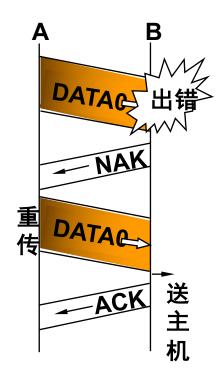


实用的停止-等待协议

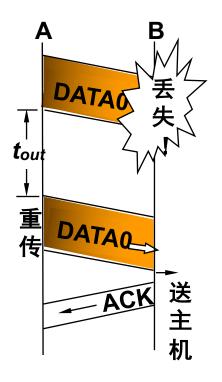
数据链路层基本概念





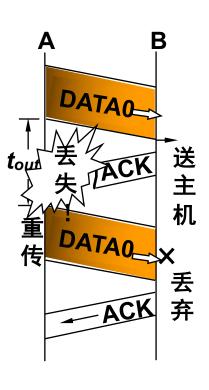


(b) 数据帧出错



点对点信道数据链路层协议

(c) 数据帧丢失



(d) 确认帧丢失



3.2.2 停止-等待流量控制

点对点信道数据链路层协议



实用的停止-等待协议

- 在一些数据传输应用中,为了简化处理, 对于出错的帧采用不发送确认应答的方式 处理这样就只需要处理超时重发。
- 停止-等待传输方式中,如果发送方未收到 应答,则一直等待。
- 必须有相应的控制机制,使发送方操作能 够继续进行,不会出现无尽的等待,造成 发送的死锁。



3.2.2 停止-等待流量控制

点对点信道数据链路层协议



实用的停止-等待协议

超时重发技术

- 就启动一个超时计时器, • 结点A发送完一个数据帧, 此计时器又称为定时器。
- 若到了超时计时器所设置的重传时间 tout而仍未 收到结点 B 的任何确认帧,则结点 A 就重传前面 所发送的这一数据帧。
- 一般可将重传时间选为略大于"从发完数据帧到收 到确认帧所需的平均时间"
- 重传若干次后仍不能成功,则主动报告差错。





点对点信道数据链路层协议



实用的停止-等待协议

重复帧问题

- 如果是接收方的应答帧丢失,会导致发送方重复 发送,出现重复帧。
- 结点 A 给每个数据帧带上不同的发送序号。每发 送一个新的数据帧就把它的发送序号加 1。
- 若结点 B 收到发送序号相同的数据帧,就表明出 现了重复帧。这时应丢弃重复帧,因为已经收到 过同样的数据帧。
- 但此时结点 B 还必须向 A 发送确认帧 ACK,以 保证协议正常执行。



3.2.2 停止-等待流量控制

点对点信道数据链路层协议



实用的停止-等待协议

帧的编号问题

- 帧的序号需要使用若干比特表示。
- 任何一个编号系统的序号所占用的比特数一定是 有限的。因此,经过一段时间后,发送序号就会 重复。
- 序号占用的比特数越少,数据传输的额外开销就 越小,但必须保证在合理的时间内不出现重复。
- 对于停止等待协议,由于每发送一个数据帧就停 止等待应答,因此用一个比特来编号就够了。



3.2.2 停止-等待流量控制

点对点信道数据链路层协议



实用的停止-等待协议

帧的发送序号

- 在停止-等待方式中,任何时候只会有一个帧未被 确认而需要标识。
- 一个比特可表示0和1两种不同的序号。数据帧中 的发送序号 N(S) 以 0 和 1 交替的方式出现在 数据帧中。
- 每发一个新的数据帧,发送序号就和上次发送的 不一样。用这样的方法就可以使收方能够区分开 新的数据帧和重传的数据帧了。



数据链路层基本概念

3.2.2 停止-等待流量控制

点对点信道数据链路层协议



停止-等待协议的性能分析

- 由于发送端口或传输信道的速率限制, 送一帧需要一定的时间: "发送时延"
- 接收一帧的时间和发送一帧的时间相同。 发送应答帧,也有"发送时延"
- 由于电磁波传输速率的存在,帧在信道中 的传送,具有"传播时延"
- 接收方收到帧后,差错检验、转交处理, 发送方收到应答后,准备发送下一帧, 需要"处理时延"

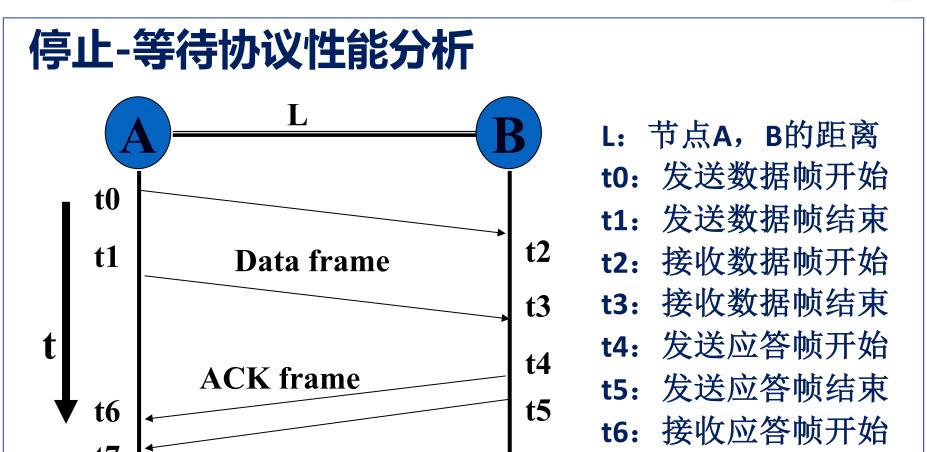


数据链路层基本概念



点对点信道数据链路层协议





接收应答帧结束



3.2.2 停止-等待流量控制



停止-等待协议性能分析

$$t_P = t_2 - t_0 = t_3 - t_1 = L/v$$
 传播时延

$$t_F = t_1 - t_0 = t_3 - t_2 = F/C$$
 发送时延

$$t_A = t_5 - t_4 = A/C$$
 应答帧发送时延

- L 节点A与B之间的距离
- v 信号传播速率 (3×108 m/s)
- F 数据帧长度 = H + D (帧头十数据)
- A 应答帧长度
- C 数据发送速率 bits/s



数据链路层基本概念

3.2.2 停止-等待流量控制

点对点信道数据链路层协议



停止-等待协议性能分析

正常情况信道利用率 (无差错)

$$U = t_D / (t_F + t_A + 2t_P + 2t_{proc})$$

假设收发双方处理时间相同:

不考虑处理时延和传播时延,及应答帧的开销, 则

U 仅与帧结构相关:

$$U=t_D/t_F=D/F=D/(H+D)$$

当传播时延较大时,发送时延与传播时延相比较小 时,信道利用率很低。



数据链路层基本概念

3.2.2 停止-等待流量控制

点对点信道数据链路层协议



停止-等待协议性能分析

非正常情况(可能出现差错, 需要重发若干次) 忽略应答时延和处理时延, 则重传时间为

$$t_{out} > = 2t_{P}$$

则成功发送一帧的间隔为 $t_T = t_F + 2t_P$

考虑传输可能出现差错的情况,则正确传送一个 数据帧的平均时间为

 $t_{av} = t_T (1 + 1$ 个帧的平均传送次数)



3.2.2 停止-等待流量控制 >>

点对点信道数据链路层协议



停止-等待协议性能分析

利用率 U= t_n/ t_{av}

- 有传输差错影响时,差错导致多次重传,信道利 用率更低。
- 除了传播时延、应答开销以外,信道利用率与控 制信息长度、信道误码率、控制开销等因素有关。
- 由于大量时间用于等待,因此系统利用率很低, 造成资源的浪费。



数据链路层基本概念

3.2.2 停止-等待流量控制 >>

点对点信道数据链路层协议



停止-等待协议性能分析

例1. 信道速率为8kb/s,采用停止等待协议,传播时 延 tp为20ms,确认帧长度和处理时间均可忽略,问 帧长为多少才能使信道利用率达到至少50%?

解:设帧长为L bit,则 发送时延 ts=L bit / 8Kbps, tp=20ms。 信道利用率= ts / (ts +2 tp) ≥ 50% 当 ts≥40ms 不等式成立,故帧长 L 应 大于等于320 bit。



数据链路层基本概念

3.2.2 停止-等待流量控制 >>

点对点信道数据链路层协议



停止-等待协议性能分析

例2. 在卫星通信系统中,两个地面卫星通信站之间 利用卫星的转发技术进行通信,信号从一个地面站 经卫星传到另一个地面站,若设其传播时延为250ms, 发送一个数据帧的时间为20ms (相当于帧长1000比特, 速率为50Kb/s), 试分析此系统的信道利用率。

注:由于卫星距离地面较远,电磁波需要较长时间才能从一 个地面站传播到另一个。在卫星通信中,从一个地面站经过 卫星到另一个地面站, 称为一"跳"。



数据链路层基本概念



点对点信道数据链路层协议



停止-等待协议性能分析

解:信号从一个地面站经卫星传到另一个地面站,其传播时 延为250ms,发送一个数据帧的时间为20ms,则从发送站开 始发送到数据帧被目的站接收,一共需要时间

20ms + 250ms = 270ms

不考虑目的站对接收到的数据帧的处理时间和应答帧的 发送时间(可以认为应答帧非常短),则应答帧也需要经过 250ms才能被发送站接收到。

从发送一帧开始,到收到应答所需要的时间为:

270ms + 250ms = 520ms

则此系统的信道利用率为:

20ms / 520ms=1 / 26 \approx 4%



数据链路层基本概念

3.2.2 停止-等待流量控制 >>

点对点信道数据链路层协议



停止-等待协议的优缺点

优点: 控制过程比较简单, 易于实现。

缺点:通信信道的利用率不高,信道还远远没有被 所传输的数据比特填满。

虽然物理层在传输比特时会出现差错,但由于 数据链路层的停止等待协议采用了有效的检错重传 机制,数据链路层对上面的网络层可以提供可靠传 输服务。

尽管停止-等待流量控制方式效率较低,但由于实现 简单, 因此仍然应用广泛。



数据链路层基本概念

3.2.3 滑动窗口流量控制

点对点信道数据链路层协议



滑动窗口流量控制方法

- 为了解决停止等待协议信道利用率低的问题,可 以采用"滑动窗口"控制方法。
- 在"滑动窗口"流量控制中,当发送完一个数据 帧后,发送方不是停下来等待确认帧,而是一边 等待,一边继续发送若干数据帧。由于在等待确 认时可以继续发送数据,减少了信道空闲时间, 因而提高了整个通信过程的吞吐量。
- 继续发送的数据帧数量,受到滑动窗口的限制。



数据链路层基本概念

3.2.3 滑动窗口流量控制

点对点信道数据链路层协议



滑动窗口流量控制方法

- "滑动窗口"机制是实现数据帧传输控制的逻辑过 程,它允许未经确认而继续发送多个数据帧。
- "滑动窗口"机制要求发送节点设置相应发送存储 单元,用于保存已发送但尚未被确认的帧,这些 帧号对应着一张连续序号列表,即发送窗口。
- 接收节点则有一张接收序号列表及相应的缓冲区, 即接收窗口。帧序号对应接收窗口中的序号才可 接收,否则丢弃。

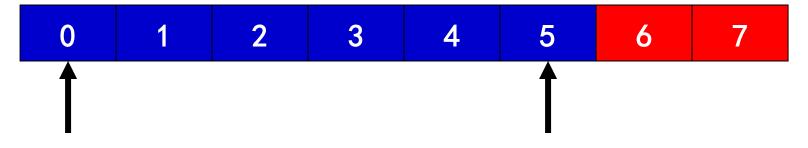


数据链路层基本概念





滑动窗口图形表示方法



后沿 L (W)

前沿H(W)

点对点信道数据链路层协议

窗口尺寸 W=6

 $H(W) = L(W) + W - 1 \mod 2^n$

发送窗口: 发送端允许连续发送的帧的序号表

接收窗口:接收端允许连续接收的帧的序号表





点对点信道数据链路层协议



发送窗口

- 发送窗口尺寸W_T: 发送端可以不等待应答到来而 允许继续发送的帧数:
- 发送窗口后沿L(W): 发送端最先发出而尚未收到 应答的帧序号:
- 发送窗口前沿H(W):发送端最后发出而尚未收到 应答的帧序号:
- 发送指针: 指向正在发送的帧序号:
- W_T的设置既要考虑传输效率,也要考虑流量控制 功能和可能出现差错的处理。



3.2.3 滑动窗口流量控制

点对点信道数据链路层协议



接收窗口

- 接收窗口尺寸:接收端允许接收的帧数:
- 在接收端只有当所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。
- 若接收到的数据帧序号是落在接收窗口之 外,则一律将其丢弃。
- 接收窗口也具有流量控制作用。





点对点信道数据链路层协议



滑动窗口控制规则

- 发送窗口:
 - 每发送一帧数据,发送指针顺序向前移 动一格,到达窗口前沿序号后停止;
 - 每接收一帧应答,窗口向前移动一格:
- 接收窗口:
 - 每接收一帧数据,检验正确后上交高层:
 - 发送一帧应答,窗口向前移动一格;
- 窗口自左向右滑动,实现流量控制。





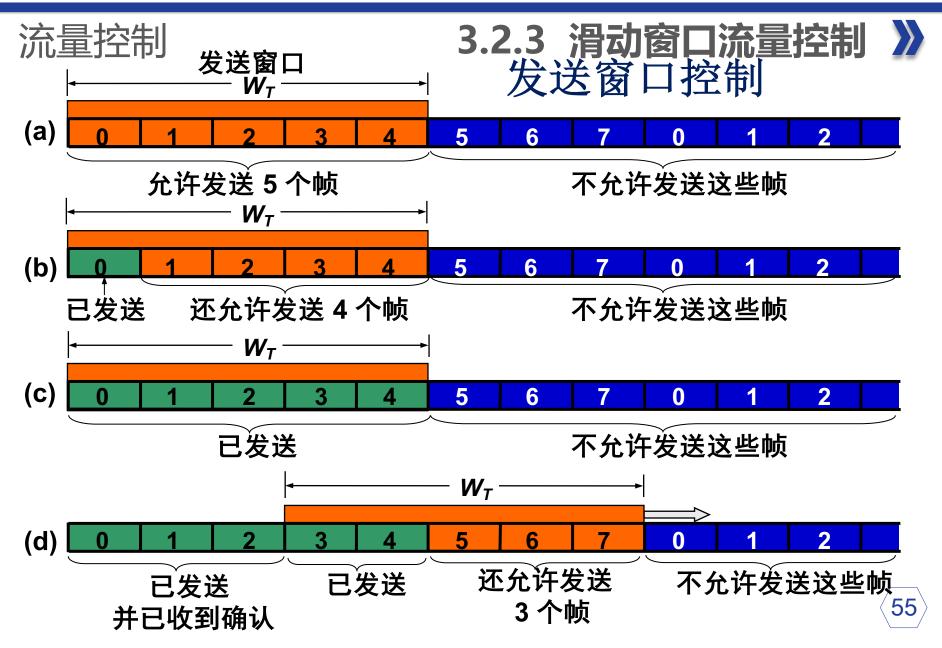
点对点信道数据链路层协议



滑动窗口

- 只有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。 即由接收方主导流量控制。
- 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, 就等效于停止等待协议。

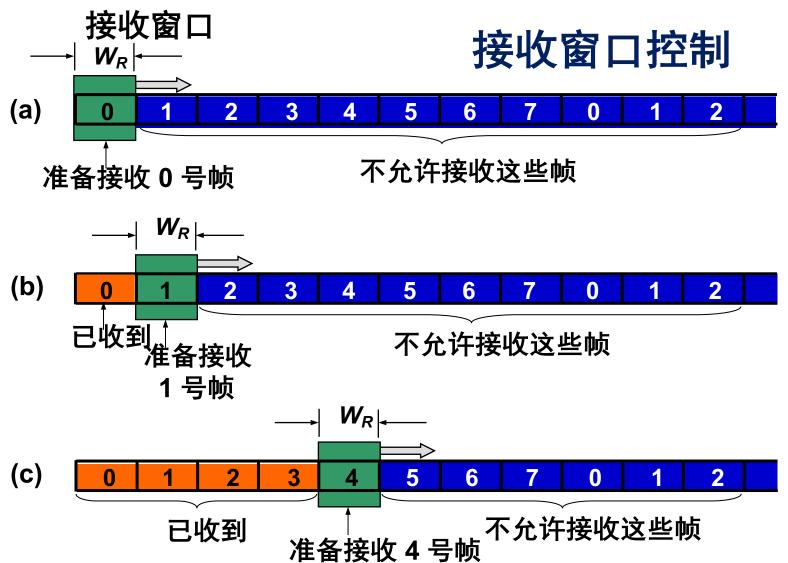






3.2.3 滑动窗口流量控制









滑动窗口控制方式

数据链路层基本概念

在滑动窗口流量控制方式中,根据通信双 方对于出现差错和需要流量控制过程时所 采用的处理方法不同,分为:

点对点信道数据链路层协议

连续 ARO 和选择 ARO。

ARQ, Automatic Repeat-reQuest, 自动重 发请求。自动请求重发



3.2.4 连续ARQ方式 **>>**



连续ARO方式

数据链路层基本概念

当发送完一个数据帧后,不是停下来等待 确认帧,而是继续发送若干数据帧:

点对点信道数据链路层协议

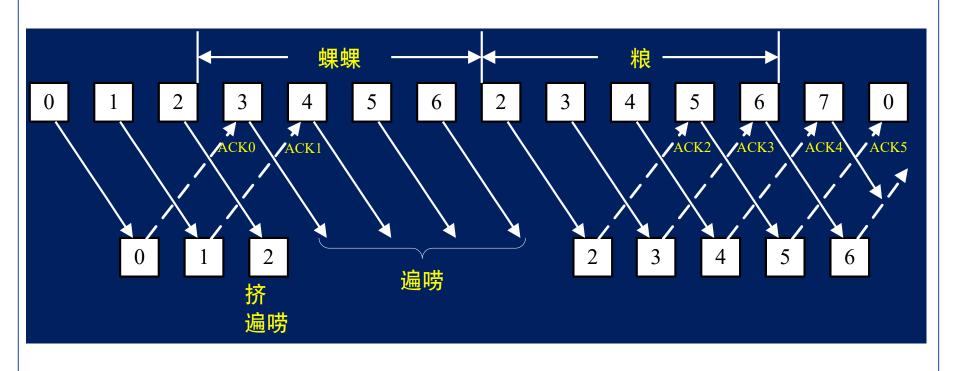
- 如果收到了接收方发来的确认帧,则发送 方可以继续发送数据帧:
- 如果出现差错,则从出现差错的数据帧开 始全部重发。
- 重发的帧序号是连续的。







连续ARQ方式







点对点信道数据链路层协议

连续ARQ方式

数据链路层基本概念

1. 接收端只按序接收数据帧。虽然在有差错 的 2号帧之后接着又收到了正确的 4 个数据 帧, 但接收端都必须丢弃这些帧, 因为这 些帧前面有一个2号帧还没有收到。虽然 丢弃了这些不按序的无差错帧,但可能重 复发送已发送过的最后一个确认帧(防止 确认帧丢失)。





连续ARQ方式

数据链路层基本概念

2. ACKO 表示确认 DATAO 帧,并期望下次收到 DATA1 帧; ACK1 表示确认 DATA1帧 ,期望 下次收到 DATA2 帧;依次类推。

点对点信道数据链路层协议

在协议式流量控制方式中,确认序号 N(R) 一般表示接收方希望接收的下一帧的序号,实 际上也表示对 N(R) -1 帧及其以前各帧的确 认。

· 在实际应用中常常使用ACK1 确认 DATAO 帧 %1





连续ARQ方式

3. 结点 A 在每发送完一个数据帧时都要设置 该帧的超时计时器。如果在超时时间内收 到确认帧,就立即将超时计时器清零,继 续发送后续的数据帧。但若在所设置的超 时时间到了而未收到确认帧,就要重传相 应的数据帧(仍需重新设置超时计时器)。





点对点信道数据链路层协议

连续ARQ方式

数据链路层基本概念

4. 在重传 2 号数据帧时,虽然结点 A 已经发 完了6号帧,但仍必须将2号帧及其以后的 各帧全部进行重传。连续 ARQ 又称为Goback-N ARQ,即"返回N帧的ARQ",意思 是当出现差错必须重传时,要向回退N个 帧, 然后再开始重传。







连续ARQ方式

- 只有接收窗口向前滑动时(正确接收并发 送了确认),发送窗口才可能向前滑动, 流量控制主要由接收方实施。
- 发送窗口尺寸一般大于1,需要多位编码来 表示已发送但未被确认的帧序号。
- 发送窗口大小的理想值: 即将发送完窗口 中最后一帧时, 收到了窗口中第一帧的确 认。
- 接收窗口尺寸等于1。







连续ARQ方式

- 当用n个比特进行编号时,则只有在发送窗口的 大小 $W_T \leq 2^n - 1$ 时,连续 ARQ 协议才能正确运行 (模-1)。
- 当采用 3 bit 编码时,发送窗口的最大值是 7 ,而 不是8。
- 在实际应用中,有模8和模128两种编码方式。 模8采用3位编码,一般用于地面链路通信; 模 128 采用 7 位编码,一般用于卫星链路通信 (卫星通信中往返传播时延较大)。





选择ARQ方式

- 连续ARQ方式实现相对比较简单,但如果出现传输差错会使效率降低。
- 在连续ARQ协议中,如果某个数据帧发生差错, 后续的数据帧即使被正确地接收到,也要被丢弃, 造成网络资源浪费。
- 为进一步提高信道的利用率,可设法只重传出错的数据帧或计时器超时的数据帧。







选择ARQ方式

数据链路层基本概念

由于差错或帧丢失,发送方按照顺序发送的帧到 达接收方时,可能会缺少某些序号的帧。使得按 照顺序接收帧的方法不能实现。

点对点信道数据链路层协议

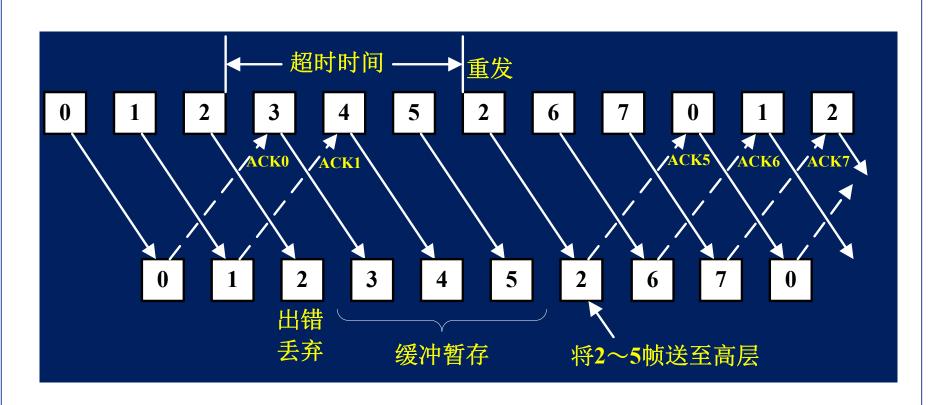
- 扩大接收窗口,先暂且收下发送序号不连续但仍 处在接收窗口中的那些数据帧,等到所缺序号的 数据帧收到后,再一并送交主机,发送应答。
- 选择 ARQ 协议可避免重复传送那些本来已经正确 到达接收端的数据帧。
- 需要付出的代价是在接收端要设置具有相当容量 的缓存空间且控制更加复杂。







选择ARQ方式







选择ARQ方式

数据链路层基本概念

选择 ARQ 方式,接收方允许接收多个不严格按照 顺序的帧。若用 n 比特进行编号,为避免编号混 乱,接收窗口的最大值受下式的约束

点对点信道数据链路层协议

$$W_R \leq 2^n/2$$

当接收窗口为最大值 $W_R = 2^n/2$ 时,选择ARQ方式 发送窗口

$$W_T = 2^n/2$$

数据传送



• 所谓差错,就是在通信接收端收到的数据和 发送端发送的数据不一致的情况。

多路复用

- 由于数据通信系统传输特性的不理想和外部 干扰的存在,传输中出现差错是不可避免的。
- 差错控制的目的: 用来提高数据传输的可靠 件与传输效率







差错控制原理



差错控制的方式



差错控制编码



差错产生的原因

3.6.1 差错控制原理 >>



•随机差错

• 原因: 信道热噪声

• 特点: 随机的、单个的

•突发差错

• 原因: 脉冲噪声(如闪电)

• 特点: 成片的、连续的



3.6.1 差错控制原理 >>



在发送的数据码元序列中加入监督位,并 进行某种变换,使它们和原来相互独立的数 据码元之间具有某种约束关系。在接收端检 测接收的数据码元和监督码元的约束关系, 如果这种约束关系被破坏,则接收端就可以 发现传输中的错误, 甚至纠正错误。



3.6.1 差错控制原理 >>>



- 变换的方法不同,就构成了不同的编码,因而 产生不同的差错控制方法。
- 检错码可以发现传输错误,但不能自动纠正
- 纠错码可以自动纠正传输错误。
- 差错控制是以降低效率为代价的。



3.6.2 差错控制的方式 >>



- 自动请求重发:接收端在收到的信码中检测出 错码时,即设法通知发送端重发。
- 前向纠错:接收端不仅能在收到的信码中发现 有错码,而且能够纠正错码。
- 混合方式: 对少量差错予以自动纠正, 而超过 其纠正能力的差错则通过重发的方法加以纠正。
- 信息反馈:接收端将收到的信码原封不动地转 发会发送端,并与原发送信码相比较。





- 在二进制编码中,设:
 - •消息长度 k 比特, 冗余信息 r 比特
 - •实际传输长度为 n 比特, n=k+r
 - 则 2^k < 2ⁿ
- 在 n 位二进制编码的 2ⁿ 种组合中,能表示 信息的 2k 种码组称为许用码组, 其余的称 为禁用码组。
- 在数据中出现禁用码组,则表示差错。



码距与汉明距离

3.6.3 差错控制编码 >>



■ 码间距离 (d) : 两个码字的对应位取值

不同的个数。例:

10001001

10110001

■ <mark>汉明距离(d₀):一个有效编码集中,</mark>任意两个码字的码间距离的最小值。即一 组编码中的最小码距。



汉明距离与纠检错能力



- •如果要能检测 e 个差错,则编码集 的汉明距离至少为 e+1;
- •如果要能纠正 t 个差错,则编码集 的 汉明距离至少为 2 t + 1;
- •如果要能检测 e 个差错,同时能纠 正 t 个差错(e > t) ,则编码集 的汉明距离至少为 e + t + 1;



汉明距离与纠检错能力



例1 数据 0 1 汉明距离	编码 0 1 d=1	00 11 d=2		000 111 d=3
例2 000000 000111 111000 111111	l 收到0	10111	4 1 5 2	恢复000111



3.6.3 差错控制编码 >>>



• 检错码

- 恒比码
- 正反码
- 奇偶校验码
- ·循环冗余码 (CRC)

•纠错码

• 汉明码 (Hamming)





- •码长为n,信息位为k, 则监督位数 r = n - k
- •r个监督位构造出r个监督关系式来指示一位 错码的n 种可能位置:

$$2^{r}$$
 1 ≥ n 或 2^{r} ≥ k + r + 1



3.6.3 差错控制编码 》



校正子与错码位置关系

S ₁ S ₂ S ₃	出错码	S ₁ S ₂ S ₃	出错码
0 0 0	无差错	0 1 1	C ₃
0 0 1	C ₀	1 0 1	C ₄
0 1 0	C ₁	1 1 0	C ₅
1 0 0	c ₂	1 1 1	C ₆



3.6.3 差错控制编码 >>>



汉明码偶校验关系

- $S_1 = C_6 \oplus C_5 \oplus C_4 \oplus C_7$
- $S_2 = C_6 \oplus C_5 \oplus C_3 \oplus C_1$
- $S_3 = C_6 \oplus C_4 \oplus C_3 \oplus C_0$

数据传送



汉明码

数据通信基本概念

3.6.3 差错控制编码



汉明码校验位生成式

数据传输方式

•
$$c_2 = c_6 \oplus c_5 \oplus c_4$$

•
$$c_1 = c_6 \oplus c_5 \oplus c_3$$

•
$$c_0 = c_6 \oplus c_4 \oplus c_3$$



3.6.3 差错控制编码 》



(7,4) 汉明码的许用码组

信息位	监督位	信息位	监督位
a ₆ a ₅ a ₄ a ₃	a ₂ a ₁ a ₀	a ₆ a ₅ a ₄ a ₃	a ₂ a ₁ a ₀
0 0 0 0	0 0 0	1 0 0 0	1 1 1
0 0 0 1	0 1 1	1 0 0 1	1 0 0
0 0 1 0	1 0 1	1 0 1 0	0 1 0
0 0 1 1	1 1 0	1 0 1 1	0 0 1
0 1 0 0	1 1 0	1 1 0 0	0 0 1
0 1 0 1	1 0 1	1 1 0 1	0 1 0
0 1 1 0	0 1 1	1 1 1 0	1 0 0
0 1 1 1	0 0 0	1111	1 1 1



3.6.3 差错控制编码 >>



[例2-8] 假如在接收端收到码字0000011, 请 判断是否有错?如何纠正?

解:按校正因子的计算式 可得:

 $S_1 = C_6 \oplus C_5 \oplus C_4 \oplus C_2 = 0$

 $S_2 = c_6 \oplus c_5 \oplus c_3 \oplus c_1 = 1$

 $S_3 = c_6 \oplus c_4 \oplus c_3 \oplus c_0 = 1$,

因为三个校正因子不全为0,说明码字有错,

错误位置为 $S=S_1S_2S_3=011$,即信息位 C_3 有错,

将c3上的0变为1,即可纠正错误。最后去掉校

验位,得到正确信息位为0001。





- ·循环冗余码(CRC)是一种特殊的线性分组 码。
- 循环冗余码各码组中的码元循环左移(或右 移), 所形成的码组仍然是一个许用码组(全 零码组除外), 称为循环性。
- 循环冗余码具有较高的检错能力。



3.6.3 差错控制编码 >>>



(n, k)循环码中,为了便于描述与计算,经常使 用 n-1 次 码多项式来表示码字,码字 $A = [a_{n-1}]$ $a_{n-2} ... a_1 a_0$],它对应的码多项式为:

$$A(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \dots + a_2x^2 + a_1x + a_0$$

例如 $A_4=0111001$, 对应的码多项式为:

$$A_4(x) = 0 \cdot x^6 + 1 \cdot x^5 + 1 \cdot x^4 + 1 \cdot x^3 + 0 \cdot x^2 + 0 \cdot x + 1$$
$$= x^5 + x^4 + x^3 + 1$$



3.6.3 差错控制编码 >>



生成多项式g(x)

在(n, k)循环码中,存在性一的最高幂次 为(n-k),最多 k-1 个连续的 0 项,且常数项 必须为1的码多项式g(x),此码多项式的幂次 最低(0元除外);其它所有的码多项式都 能被g(x)整除;并且g(x)是 xn+1的一个因式。 g(x)称为该编码集的生成多项式。



3.6.3 差错控制编码 >>



标准的生成多项式g(x)

CRC-16

$$g(x) = x^{16} + x^{15} + x^2 + 1$$

CRC-CCITT

$$g(x) = x^{16} + x^{12} + x^5 + 1$$

CRC-32

$$g(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10}$$
$$+ x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$



3.6.3 差错控制编码 >>



循环冗余码CRC的编码步骤(发送端)

- 1) 求M(x)所对应的码字,可先求M(x),并乘 以x n - k;
- 2) 然后被G(x)除,求其余式;
- 3) 得x 「M (x)



3.6.3 差错控制编码 >>>



循环冗余码CRC的检错步骤

在接收端,校验的方法是用生成多项式G(x) 除接收下来的 $x^r M(x) R(x)$,如能整除, 则表明传输无差错。



3.6.3 差错控制编码 >>



[例2-9]一个报文的比特序列为1101011011 通过数据链路传输,采用CRC进行差错检测, 如所用的生成多项式为 $g(x) = x^4 + x + 1$,试说 朗:

- CRC码的产生过程及所产生的发送序列;
- 2) CRC码的检测过程(有差错及无差错)。



3.6.3 差错控制编码 >>

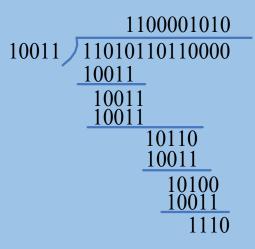


解: 生成多项式为 $g(x) = x^4 + x + 1$, 则其编码为10011, r=4。 因为r=4,所以CRC校验码是4位的。 对于报文1101011011,将其左移4位, 即在报文末尾加4个"0",这等于报文乘以24, 然后被生成多项式模2除。



3.6.3 差错控制编码

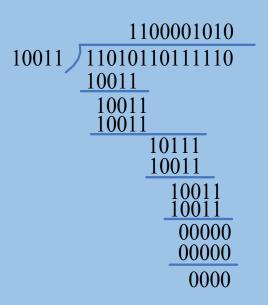




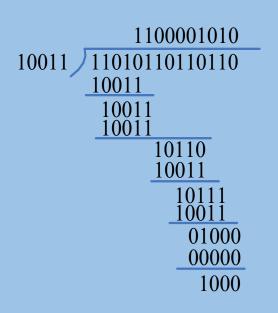
生成发送序列

 $\frac{11010110110000}{+1110} \\ \frac{110101101111110}{110101101111110}$

(a) 编码



无差错



有差错

(b) 译码





- 在数据后面添加上的冗余码称为帧检验序列 FCS (Frame Check Sequence).
- · 循环冗余检验 CRC 和帧检验序列 FCS并不 等同。
 - · CRC 是一种常用的检错方法,而 FCS 是 添加在数据后面的冗余码。
 - FCS 可以用 CRC 这种方法得出,但 CRC 并非用来获得 FCS 的惟一方法。





- 得出的余数 R 不为 0, 就表示检测到差错。 但这种检测方法并不能确定究竟是哪一个或 哪几个比特出现了差错。
- •一旦检测出差错,就丢弃这个出现差错的帧。
- 只要经过严格的挑选,并使用位数足够多的 除数 P, 那么出现检测不到的差错的概率就 很小很小。





- 仅用循环冗余检验 CRC 差错检测技术只能 做到无差错接受(accept)。
- "无差错接受"是指: "凡是接受的帧(即 不包括丢弃的帧),我们都能以非常接近于 1 的概率认为这些帧在传输过程中没有产生 差错"
- •要做到"可靠传输"(即发送什么就收到什 么)就必须再加上确认和重传机制。

过渡页 Transition Page



- 01 数据链路层的基本概念
- 02 流量控制和差错控制
- 03 点对点信道的数据链路层协议
- 04 多路访问信道的数据链路层





数据链路层协议概述

点对点信道数据链路层协议



面向比特的数据链路层协议



面向字符的数据链路层协议



3.3.1 数据链路层协议概述 >>

数据链路控制协议的作用:

在数据通信技术中,数据链路层协议往往又称为 数据链路控制规程。数据链路层控制功能的具体体现, 是使通信变得比较可靠。

数据链路控制规程的作用:

- 帧控制
- 透明传送
- 流量控制

- 链路管理
- 差错控制
- 异常状态的恢复



根据协议所处理的对象不同,分为:

- 面向字符的数据链路控制规程
 - · 基本型(如BSC,二进制同步通信)
 - · 高级(DDCMP,数字数据通信报文规程)
 - SLIP 和 PPP
- 面向比特的数据链路控制规程
 - IBM SDLC,同步数据链路控制规程
 - ISO HDLC, 高级数据链路控制规程
 - CCITT(ITU-T) LAPB,平衡型链路接入规程







- HDLC(高级数据链路控制规程)是ISO参考 相关厂家标准,制定的一种面向比特流的 数据链路规程,即它所识别和处理的是比 特流及其组合。
- HDLC是一个内容规模较大的协议集,适用 于多种同步通信应用。
- LAPB 是 HDLC 的一个子集。



3.3.2 面向比特的链路控制规程 🗼



HDLC的相关概念

站的定义(通信站):

主站: 发送控制报文的站

从站: 在主站控制下操作, 仅回复响应的站

复合站: 具有主、从站双重功能的站

链路配置:

不平衡型:用于点一点或点一多点链路

平衡型:用于点一点链路



3.3.2 面向比特的链路控制规程



HDLC的工作模式

· NRM 正常响应模式

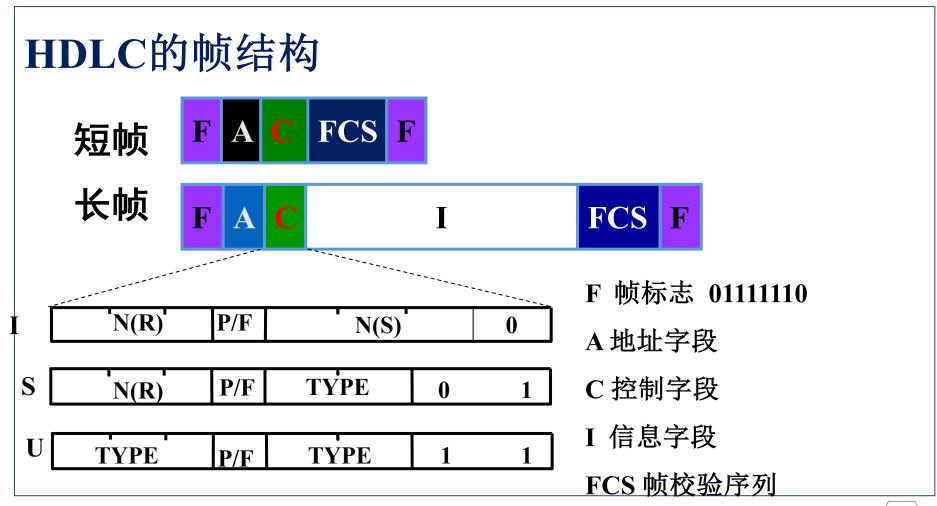
只有主站才能发起向从站的数据传输,从站只能 响应主站的询问。

- ARM 异步响应模式 允许从站发起向主站的数据传输,但主站仍然负 责全过程的初始化。
- ABM 异步平衡模式 任一复合站均可发送、接收命令/响应。



3.3.2 面向比特的链路控制规程

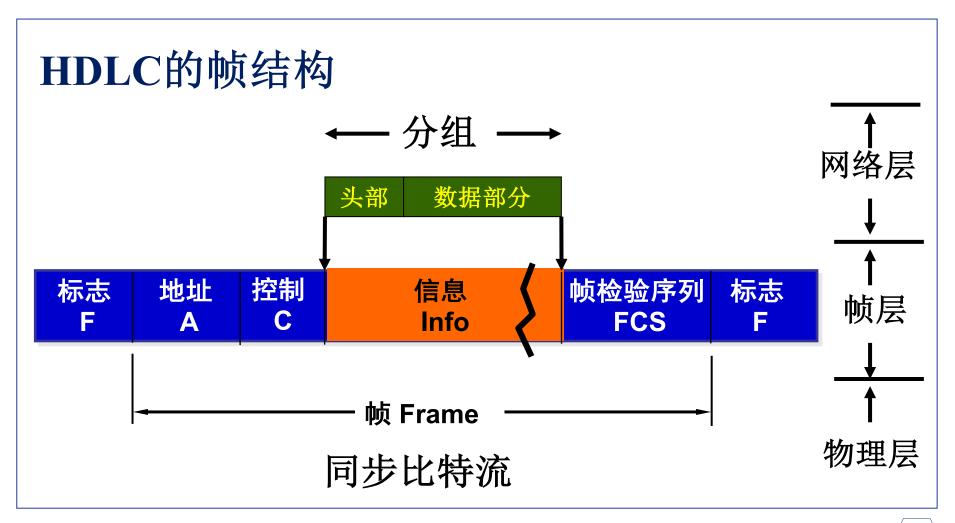






3.3.2 面向比特的链路控制规程

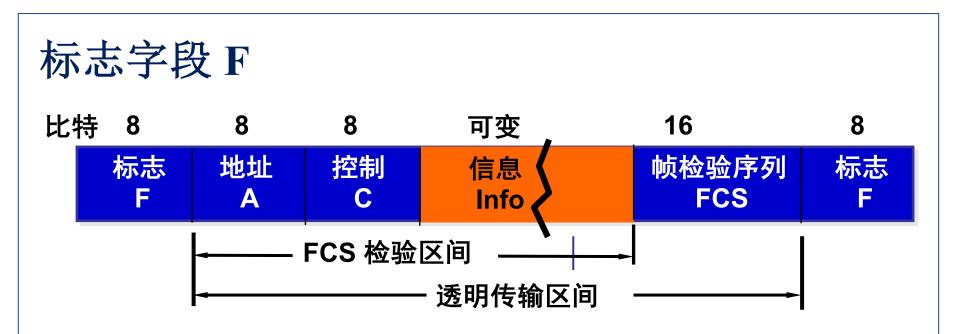






3.3.2 面向比特的链路控制规程 >>>





标志字段 F (Flag) 为 6 个连续 "1" 加上 两边各一个 "0", 共 8 bit (01111110),表示帧开始和结束,具有同步作用。接收 端根据标志字段就可确定一个帧的范围。







HDLC的帧结构

- HDLC帧中存在多种 0/1 比特组合。
 - 5个或5个以内连续1,或其他组合:正常数据。
 - 01111110: 帧标志 F, 帧开始和结束标志, 帧 间填充,帧同步。
 - 7~15个连续"1": 异常终止, 丢弃此帧。
 - 多于15个连续"1":链路空闲,停止工作。
- 必须采取措施,实现 HDLC 帧中任意比特组合的 传输,而不会造成误判。



3.3.2 面向比特的链路控制规程



HDLC的透明传输

- 采用"零比特自动插入/删除法"使一帧的两个 F 字段之间不会出现6个或6个以上连续"1"。
- 在发送端,当一串比特流数据中有5个连续1时, 就立即自动插入一个"0"。
- 在接收帧时,先找到F字段以确定帧的边界。接 着再对比特流进行扫描。每当发现5个连续1时, 就将其后的一个0删除,以还原成原来的比特流。





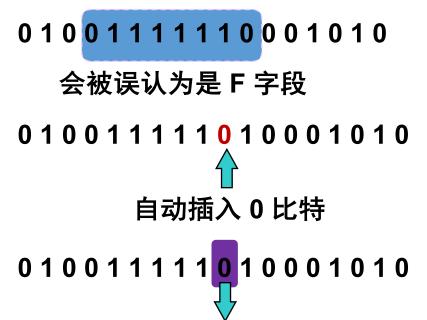


标志字段 F

数据中某一段比特组合恰好 出现和F字段一样的情况

发送端在 5 个连 1 之后 插入 0 比特再发送出去

在接收端将 5 个连 1 之后 的 0 比特删除,恢复原样



在此位置删除填入的 0 比特







HDLC的透明传输

- 采用零比特插入/删除法可传送任意组合的比特流, 即实现数据链路层的透明传输。
- 当连续传输两个帧时,前一个帧的结束标志字段 F可以兼作后一帧的起始标志字段。
- 当暂时没有信息传送时,可以连续发送标志字段, 使收端可以一直和发端保持同步, 同时表示链路 处于工作状态。





HDLC的帧结构

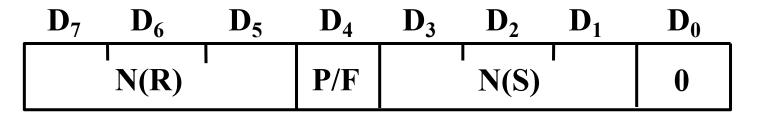
- 信息字段 I
 - 高层信息,如 IP 报文或其它信息
 - 长度可变,且无字段长度说明
 - 某些控制类帧可能无此字段
- 帧校验序列 FCS
 - 差错控制,使用CRC校验
 - · 长度16bit, 也属于透明传输范围



3.3.2 面向比特的链路控制规程 》



信息帧(I帧)



- LSB 为 0 表明该帧是信息帧。
- N(S) 号表明该帧的发送一个帧号。
- N(R)号表明该帧的发送者期望接收的下一个帧号。 (对N(R)-1及以前各帧的确认)
- 3比特编码表示为模 8 方式。
- 利用信息帧携带确认信息可提高传送效率。



3.3.2 面向比特的链路控制规程



监视帧(S帧)

- 监视帧,又称监督帧,监控帧,用于数据传送阶 段。实现流量控制和差错控制,保证数据传输的 正常进行。
- 根据传输控制的需要,S帧有以下类型:

Receiver Ready RR

接收端准备好

RNR **Receiver Not Ready** 接收端未准备好

REJ Reject 帧拒绝(连续ARQ)

SREJ Select Reject 选择拒绝(选择

ARQ)







监视帧(S帧)

数据链路层基本概念

RR	N(R)		P/F	0	0	0	1
RNR	N(R)	I	P/F	0	1	0	1
REJ	N(R)		P/F	1	0	0	1
SREJ	N(R)		P/F	1	1	0	1

N(R) 用于向发送方确认接收端所收到帧的序号。 N(R)表示该 帧的发送者期望接收的下一个帧的序号,同时对 N(R) -1及 以前各帧确认。





3.3.2 面向比特的链路控制规程



HDLC的监视帧的作用

- RR帧: 一种确认应答,确认已收到的帧,并表示 接收端已作好准备,发送方可继续发送信息帧。
- RNR 帧: 也是确认应答,确认前面已接收的帧。 同时进行流量控制,通知对方停止发送任何帧, 直到再发送RR帧,才可继续发送。
- REJ 帧: 在连续ARQ中用于拒绝收到出错的帧。
- SREJ帧: 在选择ARO中要求选择重发出错的帧。









无编号帧(U帧)

	\mathbf{D}_7	\mathbf{D}_6	$\mathbf{D_5}$	$\mathbf{D_4}$	\mathbf{D}_3	$\mathbf{D_2}$	\mathbf{D}_1	$\mathbf{D_0}$
SABM	0	0	1	P	1	1	1	1
DM	0	0	0	F	1	1	1	1
FRMR	1	0	0	F	0	1	1	1
UA	0	1	1	F	0	0	1	1
DISC	0	1	0	P	0	0	1	1

LSB 为 11 表明该帧是无编号帧(U帧)







无编号帧

- 无编号帧的控制字段中没有N(S)、N(R)序号,因 此不能提供确认、流量控制等功能。
- 无编号帧的控制字段中有5位,可以实现32种编码, 用来表示帧的功能,实际只使用了十几种。
- 无编号帧一般用于链路建立、拆除控制和异常情 况处理。
- 无编号帧没有编号,可以不受接收顺序影响, 先处理。







- 在 TCP/IP 协议族中,串行线路网际协议 (SLIP) 和点到点协议(PPP) 是专门用 于调制解调器或其它设备直接专线连接的, 它不需要进行介质访问控制,能够提供完 整的数据链路层功能。
- 用户使用拨号电话线接入因特网时,一般 都是使用 SLIP或 PPP 协议。
- SLIP 和 PPP 协议都是面向字符的协议。



- 由于SLIP在协议识别和差错控制等方面的 不足,已逐渐退出应用。现在使用得最多 的是点到点协议 PPP (Point-to-Point Protocol).
- · 1992年制订了 PPP 协议。经过 1993年和 1994年的修订,现在的 PPP协议已成为因 特网的正式标准 RFC 1661。
- · PPP目前不仅用于拨号或专线接入,也广泛 用于路由器连接、以太网应用 PPPoE。



3.3.3 面向字符的链路控制规程



点到点协议是为在两个对等实体间传输 数据包建立简单连接而设计的。这种连接提供 了全双工操作,并且假定数据包是按顺序投递 的。PPP连接提供了一种广泛的解决办法。

PPP包含三个组成部分

- 将IP数据报封装到串行链路的方法。
- 链路控制协议 LCP (Link Control Protocol)。
- 网络控制协议 NCP (Network Control Protocol)。



数据链路层基本概念

3.3.3 面向字符的链路控制规程



- 为了在点到点连接中建立通信,PPP连接的每 一端都必须首先发送LCP数据包来配置和测试 数据连接。在连接建立后,对等实体还有可能需 要认证。
- 建立数据连接后, PPP必须发送NCP数据包 来选择一种或多种网络层协议来配置。一旦被选 中的网络层协议被配置好后,该网络层的数据报 就可以在链路上传送了。
- 链路将保持可配置的状态直到有LCP数据包和 NCP数据包终止连接,或有其他外部事件发生。



3.3.3 面向字符的链路控制规程



PPP 协议的帧格式

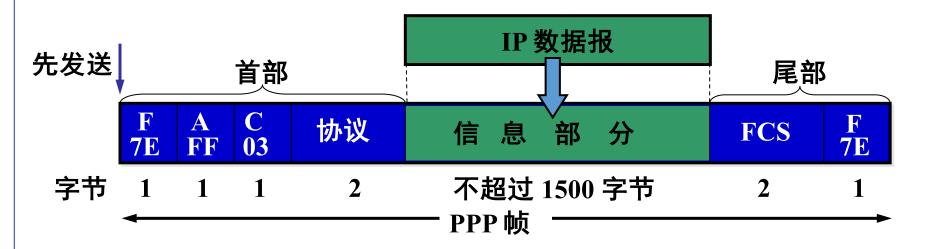
- PPP 的帧格式和 HDLC 的相似。
- 标志字段 F 仍为 0x7E (符号"0x"表示后 面的字符是用十六进制表示。十六进制的 7E的二进制表示是 01111110)。
- 地址字段A只置为0xFF。地址字段实际上 并不起作用。
- 控制字段 C 通常置为 0x03。







PPP的帧格式



PPP 是面向字符的,所有的 PPP 帧的长度都是整 数字节。









PPP 协议的帧格式

- PPP 有一个 2 个字节的协议字段,用于指示 帧的信息字段中所携带的内容。
 - · 当协议字段为 0x0021 时, PPP 帧的信息字段就 是IP数据报。
 - · 若为 0xC021,则信息字段是 PPP 链路控制数据。
 - · 若为 0x8021,则表示是网络控制数据。
- 信息字段无特别要求,其内容由协议字段说明,长 度不超过1500字节。
- 使用16位帧校验。









PPP 的透明传输

PPP帧在传输过程中,同样需要考虑透明 传输问题。即帧标志之间的比特组合。

- 当 PPP 用在同步传输链路时,协议规定采 用硬件来完成比特填充(和 HDLC 的做法一 样,零比特自动插入/删除)。
- 当 PPP 用在异步传输时,就使用一种特殊 的字符填充法。





PPP 的透明传输

- 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。









PPP 的透明传输

PPP字符填充应用

一个PPP帧的数据部分(十六进制)是

7D 5E FE 27 7D 5D 7D 5D 65 7D 5E, 问真正

的数据是什么?

7D 5E FE 27 7D 5D 7D 5D 65 7D 5E

FE 27 7D 7D 65





PPP 的传输

- PPP是面向连接的协议
- PPP不使用序号和确认机制
 - 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。



3.3.3 面向字符的链路控制规程



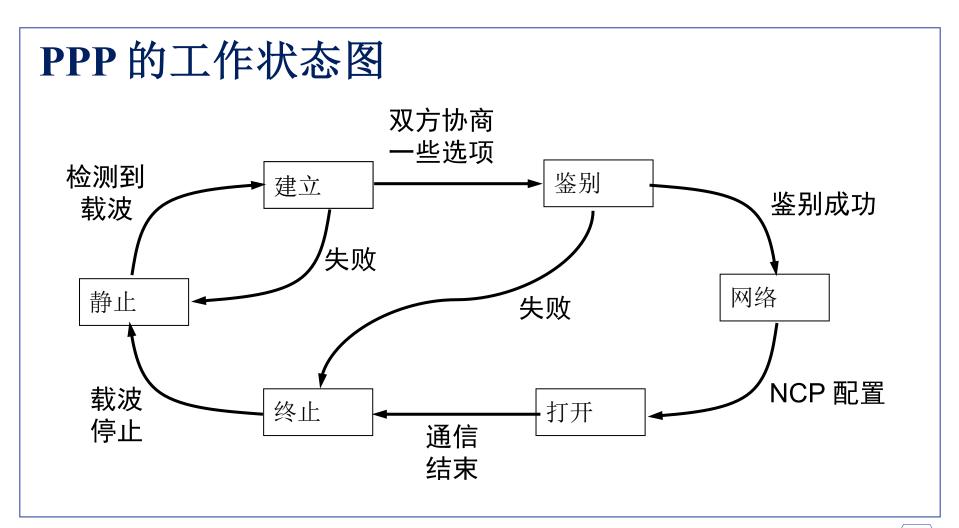
PPP 的工作状态

- 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。
- 用户向路由器发送一系列 LCP 分组, 建立链路。
- 用户发送分组及其响应选择一些 PPP 参数,进行 网络层配置, NCP 给新接入的 PC机分配一个临 时的 IP 地址, 使 PC 机成为因特网上的一个主机。
- 通信完毕时, 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。

数据链路层基本概念

3.3.3 面向字符的链路控制规程





过渡页 Transition Page



- 01 数据链路层的基本概念
- 02 流量控制和差错控制
- 03 点对点信道的数据链路层协议
- 04 多路访问信道的数据链路层







信道共享技术



竞争系统介质访问控制

点对点信道数据链路层协议



环型网络介质访问控制



令牌总线介质访问控制



3.4.1 信道共享技术 >>>

- 在网络通信应用中,为了提高通信资源的 利用率, 广泛采用了信道共享技术。即多 个通信实体共享公共的传输信道实现多对 实体之间的通信和资源共享. 又称"多点接 **\(\)** " \(\)
- 所谓"访问"指的是在两个实体之间建立 联系并交换数据信息。
- 介质访问控制方式一般指分配公共传输信 道使用权限的机理、策略和算法。

数据链路层基本概念

3.4.1 信道共享技术 >>

信道共享技术,即如何分配使用公共信道传输资源的方法和策略,有静态和动态的信道分配方案。

- 静态分配方案:预先申请,一旦分配资源,则保持本次通信过程内资源拥有情况不变。控制简单,但不够灵活,可能利用率不高。
- 动态分配方案:能够根据数据源对传输资源的 随机需求而动态分配、调整。资源利用率高, 但控制机制比较复杂。



3.4.1 信道共享技术 >>>

多个计算机系统共享使用公共信道资源的方法 又称为多路访问技术(介质访问控制方式), 分为受控访问和随机访问。

- 受控访问:各个用户不能任意接入到信 道而必须服从一定的控制。又分为集中 式控制和分散式(分布式)控制。
- 随机访问: 所有的用户都可以根据自己的意愿随机地发送信息,又称为争用接入。



常见的信道共享技术:

- 基于信道划分的信道共享技术,一般是静态共享。 如频分复用,时分复用,码分复用,波分复用等, 属于受控访问,一般是集中控制。
- 基于轮询的信道共享技术,可采用集中或分布控制机制,各个站点轮流获得信道使用权限。
- 基于随机访问的信道共享技术,是一种动态分配资源方法,由于所有的用户是随机地使用公共资源,需要通过竞争获得。



- 动态信道分配的主题是如何在多个随机竞争的用户之间分配单个共享的公共信道, 并避免和解决可能出现的竞争冲突。
- 所有站点都连接到一个共享信道上,所用的接入和共享信道的技术称为多点接入控制方法(多路访问技术),又称为介质访问控制方法,媒体接入控制,媒体访问控制。
- Medium Access Control, MAC



竞争系统的介质访问控制方法

- ALOHA
 - 纯ALOHA、时隙ALOHA
- CSMA
 - 非坚持,1一坚持,P一坚持
 - 改进的CSMA
 - CSMA/CD, CSMA/CA

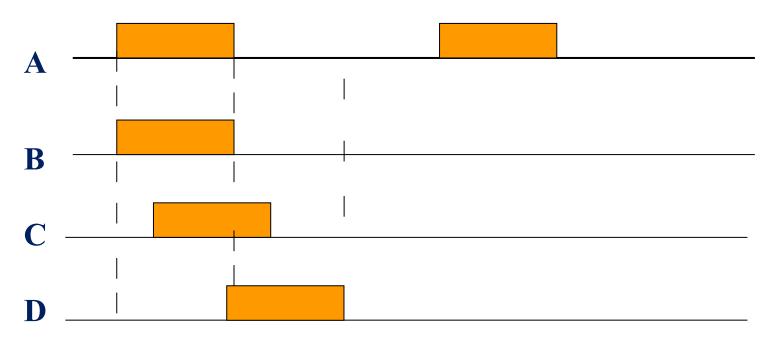


ALOHA是一种完全随机发送数据的无线 局域网,最初应用于夏威夷大学。

- 纯ALOHA: 用于局域网无线公用信道上,集中控 制,仅使用两个频率,上行传输(争用)为407.35 MHz, 下行传输(广播)为413.475 MHz, 信道利 用率最高为18.4%。
- 时隙ALOHA:将时间分为等长的时隙,只能在每 个时隙的开始才发送分组,目的是减少冲突。信 道利用率最高为36.8%。

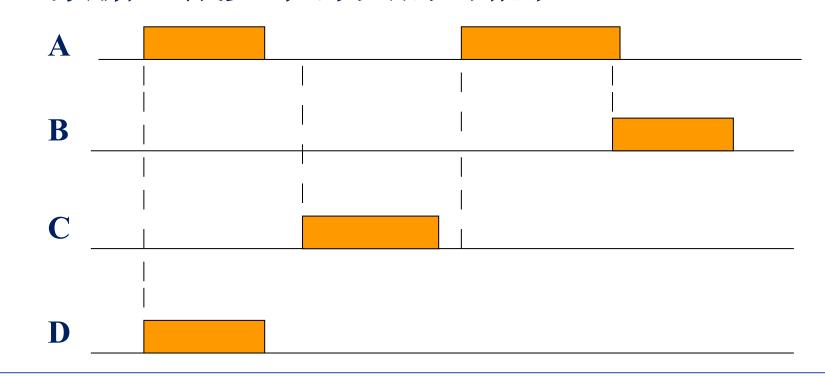


ALOHA: 网络中多个站点随机发送数据,在 公共信道中可能产生相互干扰,称为"冲突"





时隙ALOHA: 网络中站点只在时隙开始时发送数据,减少了冲突的可能性。





- 在ALOHA方式中,每个站点不关心网络忙闲状态,随机发送数据,因此容易产生信号冲突,降低了通信效率。
- 如果发送前检测网络状态,则可减少冲突。
- CSMA, Carrier Sense Multiple Access 即是 所作出的改进。
 - 每个站点在发送分组前,监听公共信道上其它 站点是否在发送分组。如果信道忙,就暂不发 送。如果信道空闲,则进入发送处理(先听后

说)。



3.4.2 竞争系统的介质访问控制 》

- · 非坚持CSMA
 - 发现信道忙则不再持续侦听,等待一个随机长的时间后,再重新开始侦听/发送过程。
- 1一坚持CSMA
 - 发现信道忙则持续侦听,直至信道空闲;
 - 等到信道空闲后发送数据(概率为1)。
- P一坚持CSMA
 - 发现信道忙则持续侦听,直至信道空闲;
 - 等到信道空闲后,以概率 P 发送数据,或以 (1-P) 概率推迟发送。

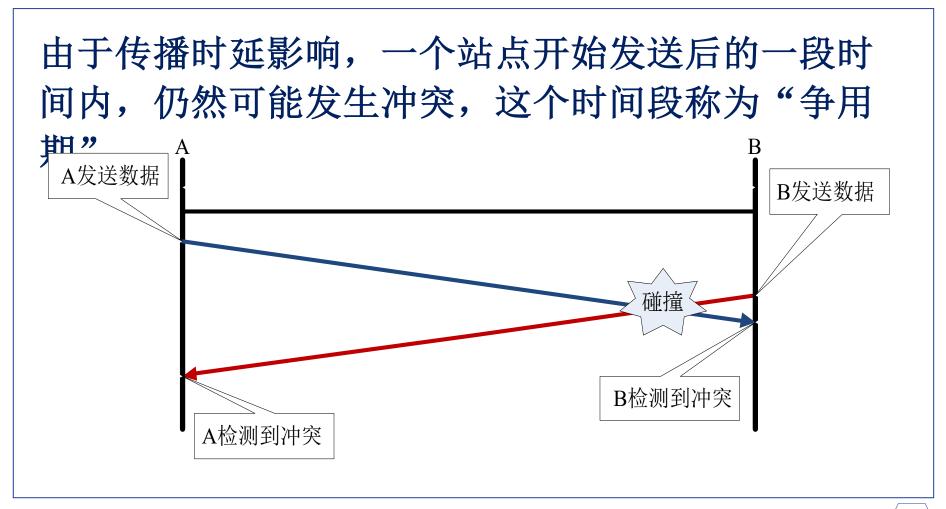


3.4.2 竞争系统的介质访问控制 》

尽管有发送前载波监听,但由于通信的随机性和信 道忙造成多个站点发送"同步",仍然可能发生冲 突。 忙 信道状态 闲 A站发送 B站发送 监听 监听 A、B站同时监 听到信道空 闲,进入发 送状态。



3.4.2 竞争系统的介质访问控制 》





3.4.2 竞争系统的介质访问控制 >>

- CSMA机制仍然不能完全避免多个站点发送 数据的冲突。
- CSMA的缺点: 当两(多)个站发生冲突后, 各冲突站并不知道,仍继续发送已遭破坏 的数据帧。应答超时后才能确认冲突。若 帧很长,则信道的浪费相当大。
- CSMA的改进:增加了"冲突检测" 的功能, 即改为"CSMA/CD"(边说边听)。



CSMA/CD (Carrier Sense Multiple Access / Collision Detection)

CSMA/CD的工作原理

- 载波监听
- 冲突检测
- 多路访问(多次访问)

CSMA/CD 在发送过程中继续检测信道。如果发现冲突,立即停止发送,并通知全网。等待随机时间后重新监听和发送。有效提高了网络传输效率。



- 竞争型网络中,随着用户数量的增加,或者网络中各个站点通信数据量的增加,即网络整体负荷增加,将导致网络中各个站点之间传输冲突可能性增加。
- 网络冲突导致发送无效,造成资源浪费, 发送时延增加,而且无法预测。甚至通信 失败。
- 环型网采用无冲突的介质访问控制方法, 属于分布式轮询控制方式。



- 环型网中各个站点和传输链路依次相连接,构成一个闭合的环。
- 环型网通过逐站转发帧实现传输介质共享, 不是广播方式,但具有广播的效果。
- 主要的介质访问方法有令牌环、时隙环及寄存器插入环。
 - 时隙环以环上时间片分割来控制帧的发送和接收。寄存器插入环通过移位寄存器实现帧的接收和发送。



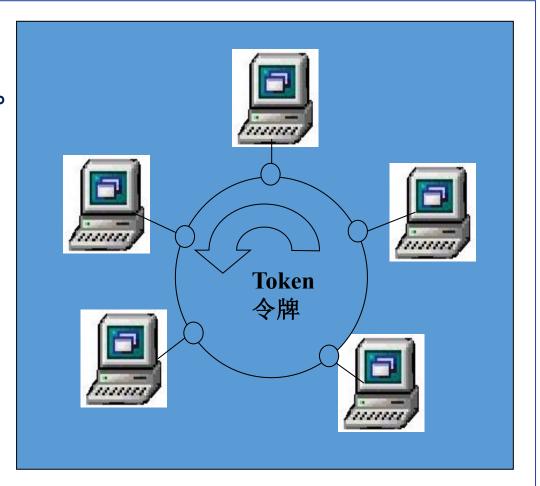
• IEEE 802.5 令牌环 (标记环)

Token Ring, IBM公司

- IEEE 802.5 令牌环介质访问方式的特点
 - 将各个站点、链路依次串成闭合环路
 - 令牌沿环循环,同一时刻环中只有一个令牌。
 - 环内令牌、数据单向传输,站点获得空闲令牌 才能发送数据帧。
 - 分散控制,无冲突
 - 传输介质:屏蔽双绞线,4Mbit/s,16Mbit/s



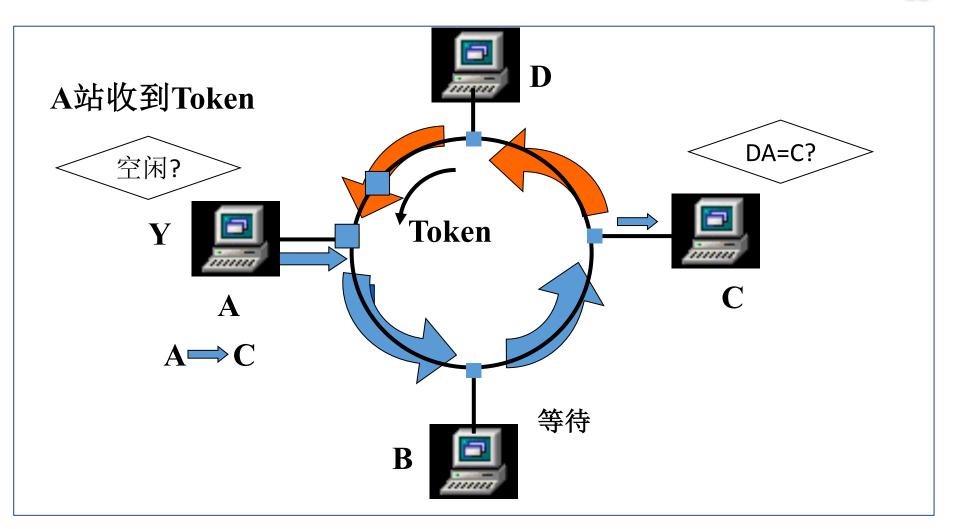
- 终端通过收发器接 入网络,构成环型。
- IEEE 802.5令牌环 介质访问控制使用 一个令牌沿着环单 向循环,且应确保 令牌在环中唯一。
- 获得空令牌才能发 送数据帧。





- 网上站点要求发送帧,必须等待空令牌。
- 当获取空令牌,则将它改为忙状态,后随数据帧; 环内其它站点不能发送数据。
- 环上站点接收、移位数据,并进行检测。如果与本站地址相同,则在转发同时接收数据,接收完成后,设置相应标记。
- 该帧在环上循环一周后,回到发送站,发送站检 测相应标记后,将此帧移去。
- 将忙令牌改成空令牌,继续向下一个站点传送, 供后续站发送帧。

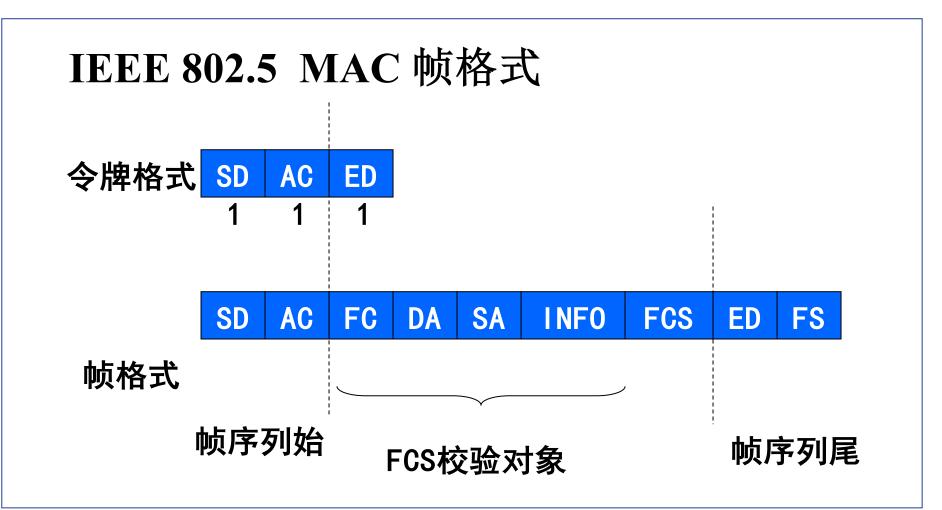






- 环型网中各个站点只有获得空闲令牌才能发送数据,因此不会出现竞争。
- 网络中只能有一个令牌。传输数据时令牌成为忙 状态,只能传送一个帧
- 环型网上站点数量一定,如果每个站持有令牌发送数据的时间确定,则任意站点最大等待时间是可以预计的。
- 随着网络负荷的增加,环型网总传输利用率提高。 而竞争类型的总线网络随着负荷增加,竞争加剧, 冲突增加,效率下降。







在令牌环网络中,令牌或数据帧在网络中单向传送, 必须识别帧的正确接收,保证令牌的唯一和正确。

- 如何防止数据帧在环上无休止循环?
 - 设置监控器
 - 在帧结构上留一个标识
- 如何监测令牌出错?
 - 无令牌
 - 多个令牌
 - 忙令牌死循环

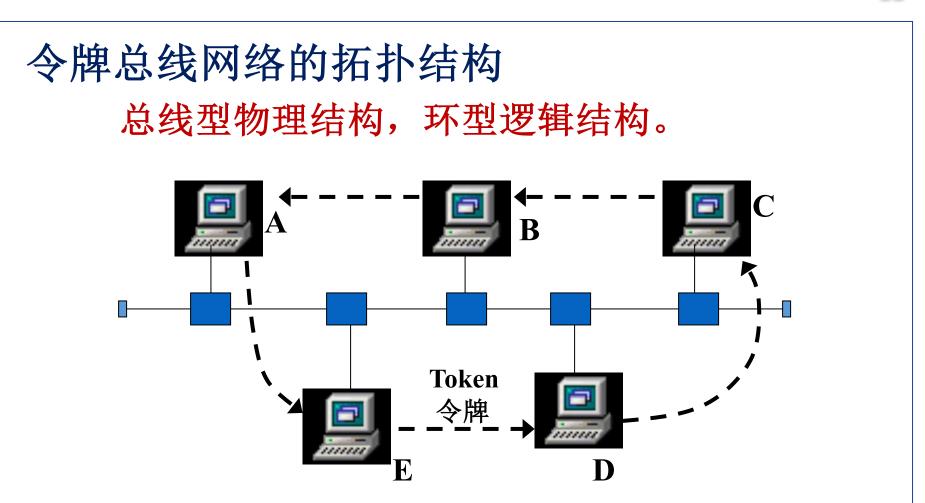


令牌和帧的检测方法

- 集中式检测
 - 设置监控站(超时计数器),检测令牌丢失。
 - 在帧结构上检测忙标记,发现死循环。
- 分布式检测
 - 每站设置定时器:当站有数据要发且等待令牌的时间超限,认为令牌丢失。



3.4.4 令牌总线介质访问控制 >>>





3.4.4 令牌总线介质访问控制 >>

令牌总线(Token Bus)网络工作原理

- 令牌总线网络在物理总线上建立逻辑环。
- 逻辑环上,令牌是站点可以发送数据的必要条件。
- 令牌和数据帧在逻辑环中按某种设定的顺序传送 到下一站点。
- 从物理上看,含DA的令牌或帧广播到BUS上,所 有站点按DA = 本站地址判断收否。
- 当站点加入或退出网络时,需要按照设定的规则 初始化,重新构成逻辑环。



3.4.4 令牌总线介质访问控制 >>

令牌总线网络的特点

- 局域网标准 IEEE 802.4。
- 无冲突,令牌环的信息帧长度可按需而定。
- 具有总线网络的可靠性。
- 具有令牌网络的优点,如公平性、站点等 待Token的时间确知。
- 算法复杂,维护难度较大。



本章小结

- 数据电路与数据链路
- 流量控制的基本概念
- 停止-等待方式和滑动窗口流量控制
- · 高级数据链路控制规程HDLC
- · 点到点协议PPP
- 信道共享与介质访问控制技术
- 竞争系统的介质访问控制
- 环型网络的介质访问控制

Inank You

Have A Nice Day

南京邮电大学计算机学院

"计算机通信与网络" 国家精品课程组