



西北師範大學
NORTHWEST NORMAL UNIVERSITY

计算机网络

第三章 数据链路层 (1)

陈旺虎

chenwh@nwnu.edu.cn



第 3 章 数据链路层

3.1 使用点对点信道的数据链路层

3.2 点对点协议 PPP

3.3 使用广播信道的数据链路层

3.4 使用广播信道的以太网

3.5 扩展的以太网

3.6 高速以太网

3.7 其他类型的高速局域网接口

Review



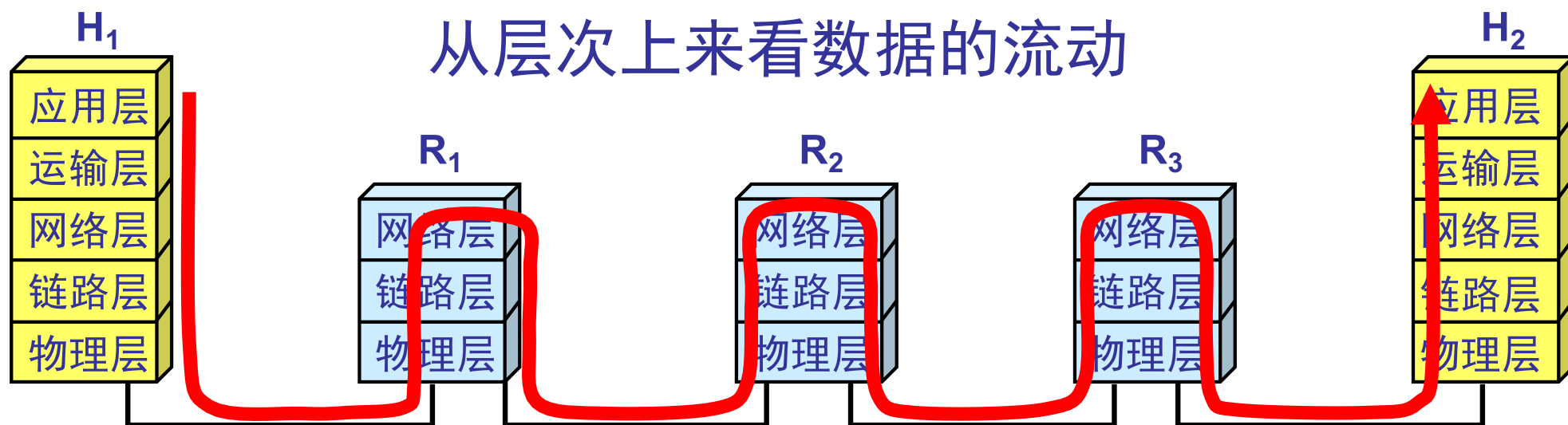
- 计算机网络的体系结构
- 物理层

数据链路层的简单模型

主机 H_1 向 H_2 发送数据



从层次上来看数据的流动

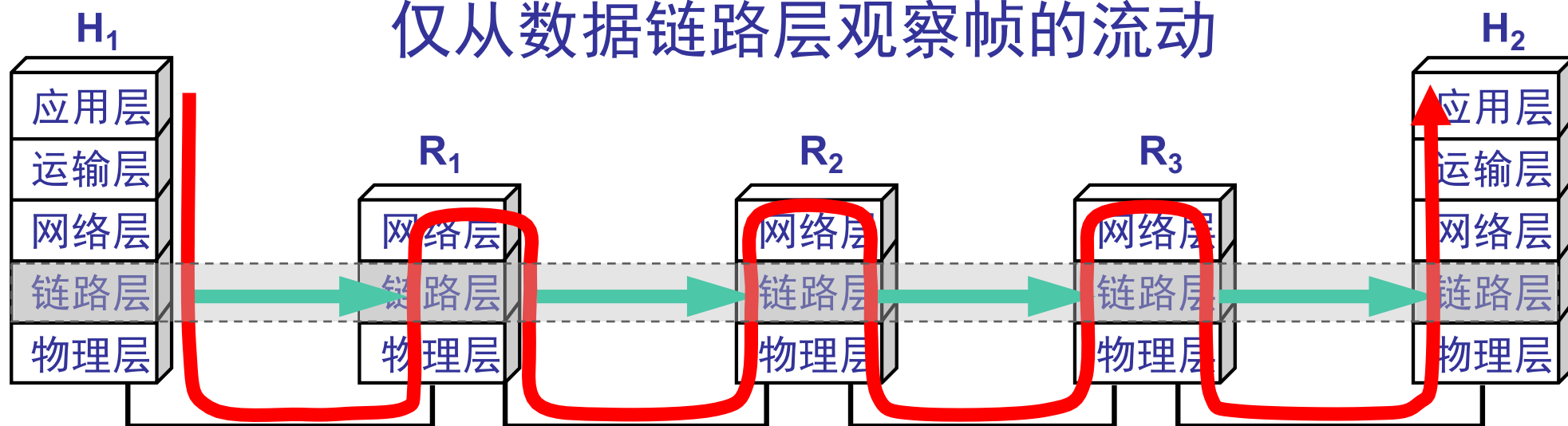


数据链路层的简单模型

主机 H_1 向 H_2 发送数据



仅从数据链路层观察帧的流动



数据链路层

数据链路层的两种信道：



- **点对点信道：** 使用一对一的点对点通信方式。
- **广播信道：** 使用一对多的广播通信方式
 - ▣ 过程比较复杂。广播信道上连接的主机很多，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

3.1 使用点对点信道的数据链路层

3.1.1 数据链路和帧

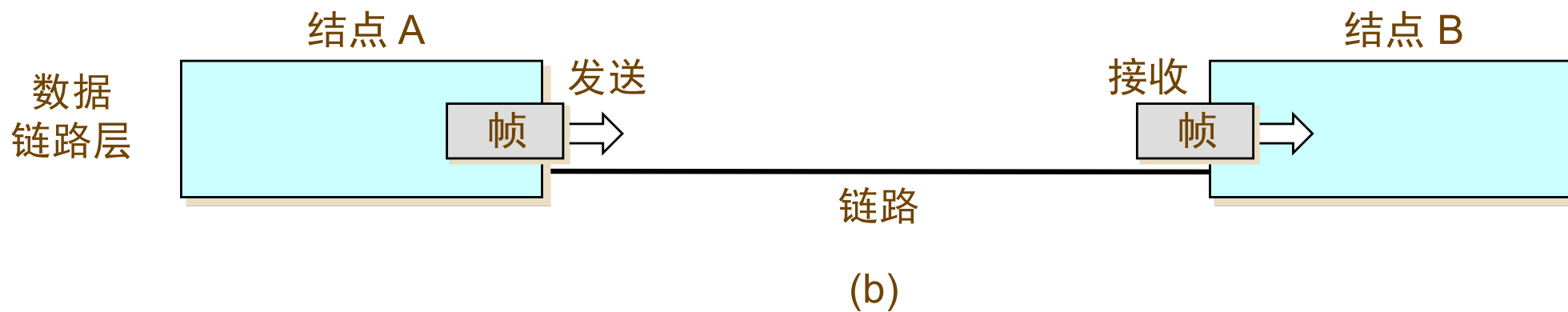
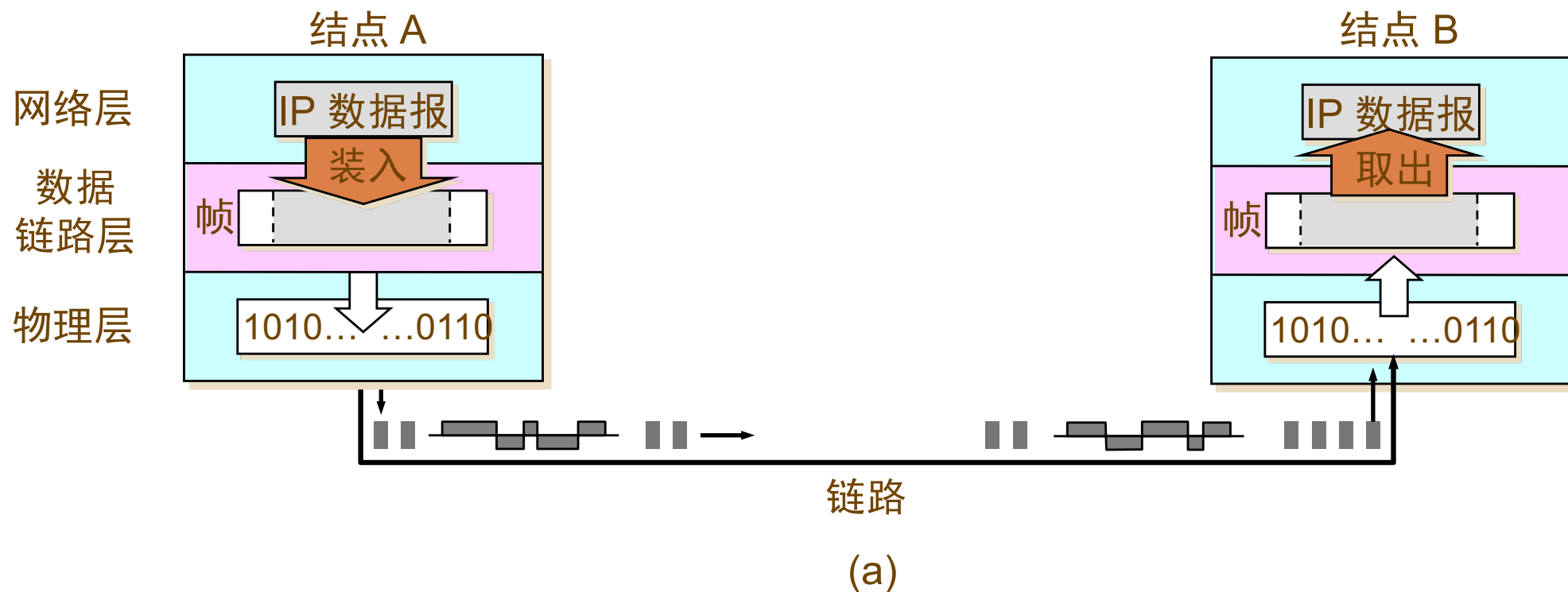
□ 链路(link)

- 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结点。
- 一条链路只是一条通路的一个组成部分。

□ 数据链路(data link)

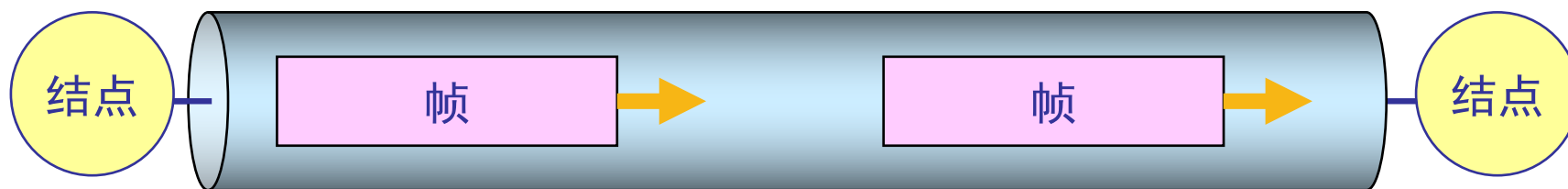
- 除了物理线路外，还必须有通信协议来控制数据的传输。若把实现这些协议的硬件和软件加到链路上，就构成了数据链路。
- 现在使用适配器（网卡）实现这些协议的硬件和软件。
- 一般的适配器都包括了数据链路层和物理层这两层的功能。

数据链路层传送的是帧



数据链路层像个数字管道

- 常常在两个对等的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。



- 在数据链路层，规程和协议也是同义语。

3.1.2 三个基本问题

- 数据链路层协议有许多种，但有三个问题是共同的
 - 帧定界问题
 - 发送方应让接收方知道发送的帧从什么地方开始和结束。
 - 透明传输问题
 - 数据链路层传送的数据比特组合必须是不受限制的。即数据链路层协议不能禁止某种特殊的比特组合。如：帧分隔符
 - 差错检测功能。

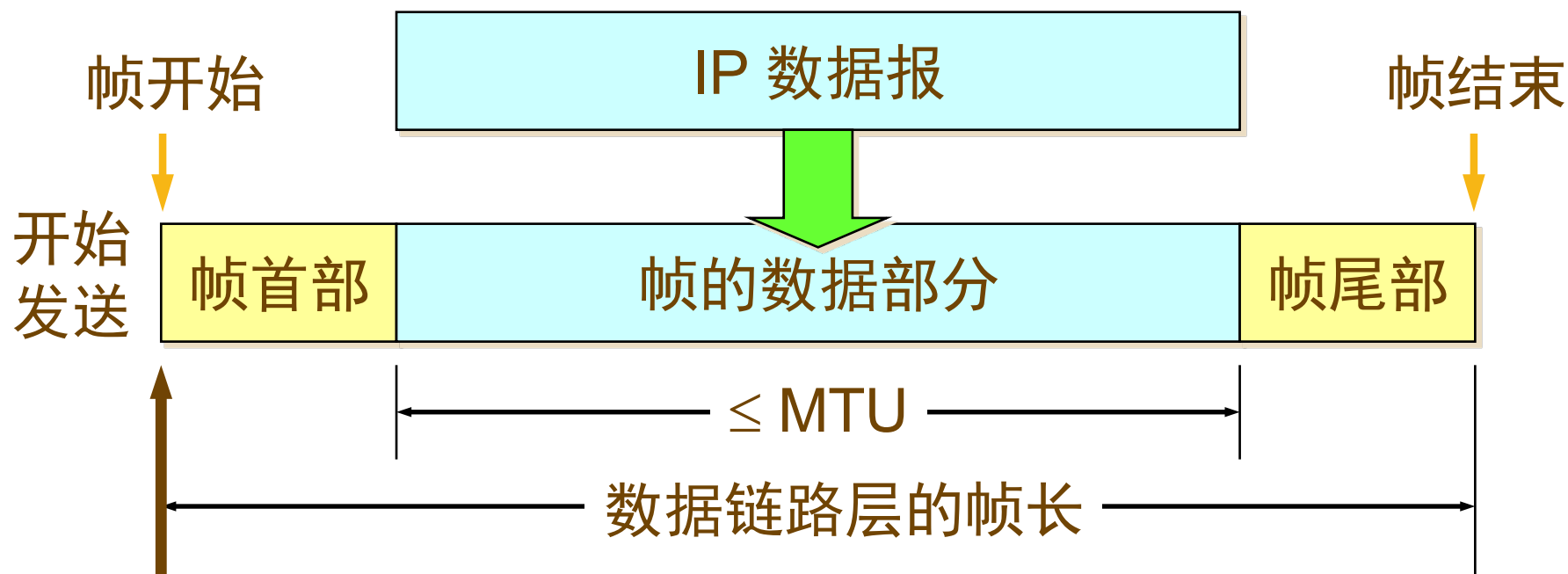
(1) 封装成帧；(2) 透明传输；(3) 差错控制

1. 封装成帧

□ 封装成帧(framing)

□ 就是在一段数据的前后分别添加首部和尾部，使其构成一个帧。

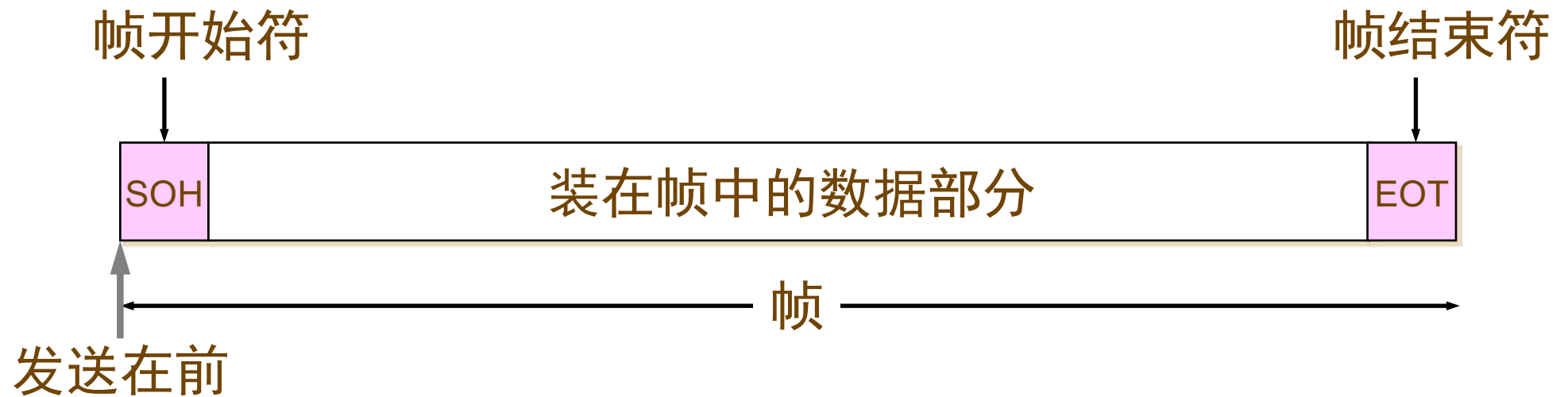
□ 首部和尾部的一个重要作用就是进行帧定界。



帧同步

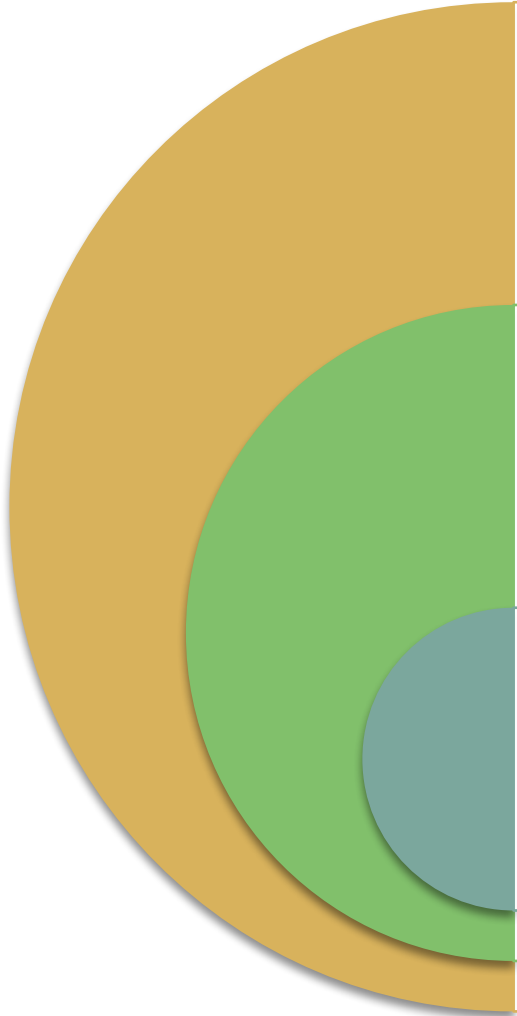
- 发送帧时，发送方根据双方协商好的标记加在帧首和帧尾作为定界符。
- 为什么在数据链路层要把比特组合成帧为单位传送？
 - ▣ 一个主要的原因：在出错时，只重发有错的帧，而不必重发全部数据，从而提高效率
 - ▣ 每一帧利用冗余位进行校验。而要校验，必须能从物理层收到的比特流中明确区分出一帧的开始和结束——帧同步

举例：用控制字符进行帧定界



有哪些可行的方法？

控制信息的形成方法（成帧）



字节计数法

- 先用一个特殊字符来表示一帧的开始
- 再在帧头中使用一个字段来表示帧内的字节数

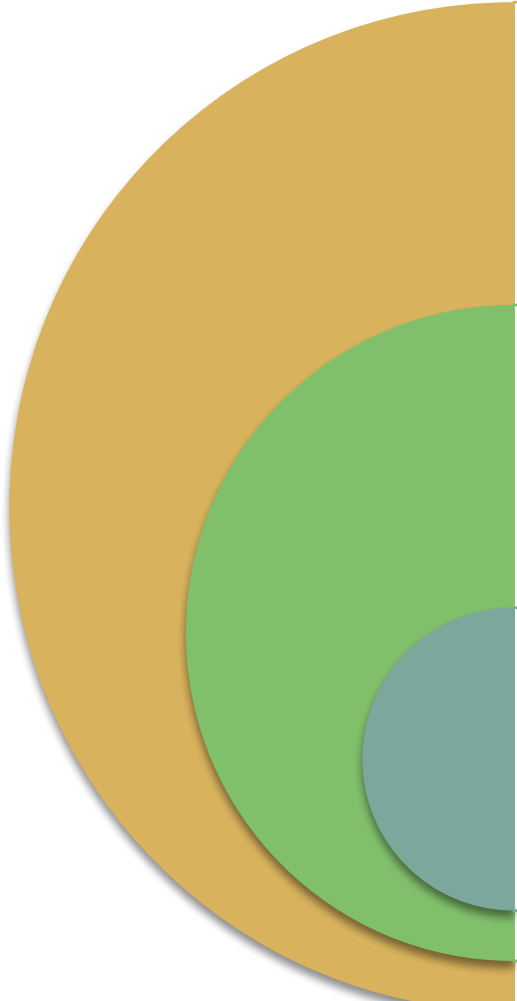
带填充字符的首尾界符法

- 以特定的**字符序列**为控制字段
- 帧定界符：SOH、EOT（十六进制编码为01和04）

带填充位的首尾标志法

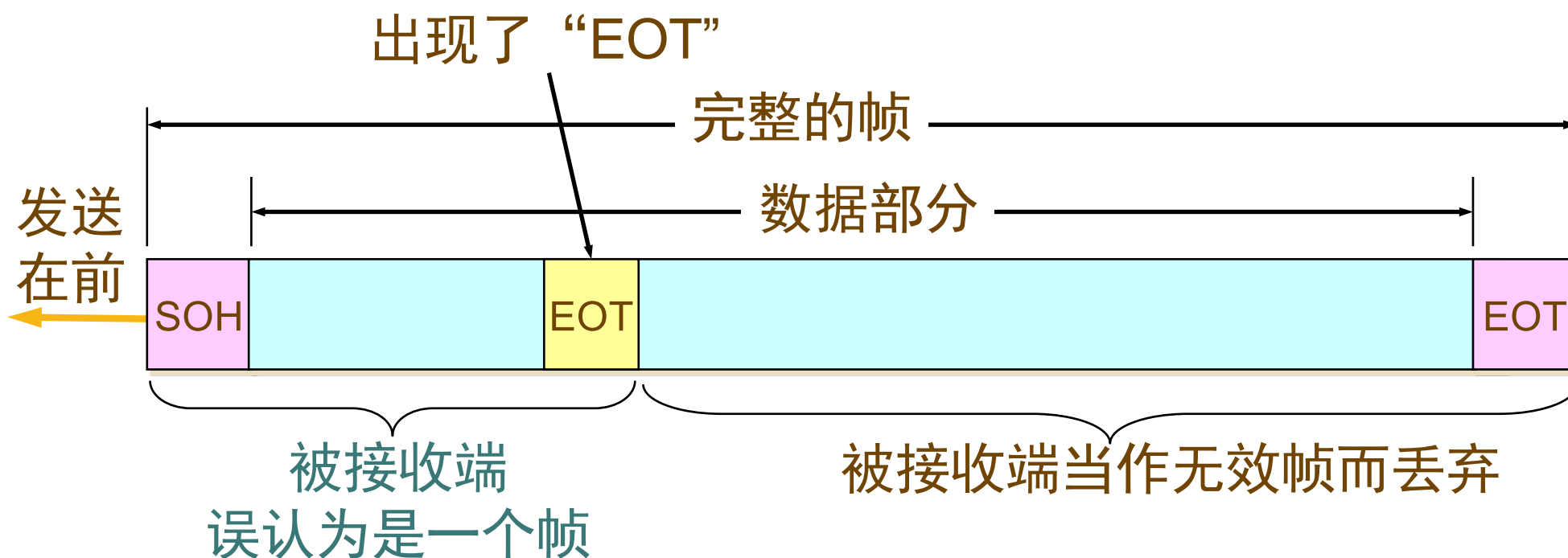
- 采用统一的帧格式，以**特定**的位序列进行帧同步和定界。

成帧方法比较



字节计数法	• 计数字段一旦出错，将无法再同步。引起灾难性后果。 • 如：DDCMP协议（DEC公司的数字数据通信报文协议DDCMP）
带填充字符的首尾界符法	• 依赖于字符集，不通用，也无法扩展 • 如：ISO1745、IBM的BSC协议。 带填充位的首尾标志法
带填充位的首尾标志法	• 不依赖于字符集，硬件实现较方便，可用于双向同时通信，不必停等确认而可连续发送。 • 同步数据链路控制协议（SDLC）：ISO的HDLC协议、PPP协议

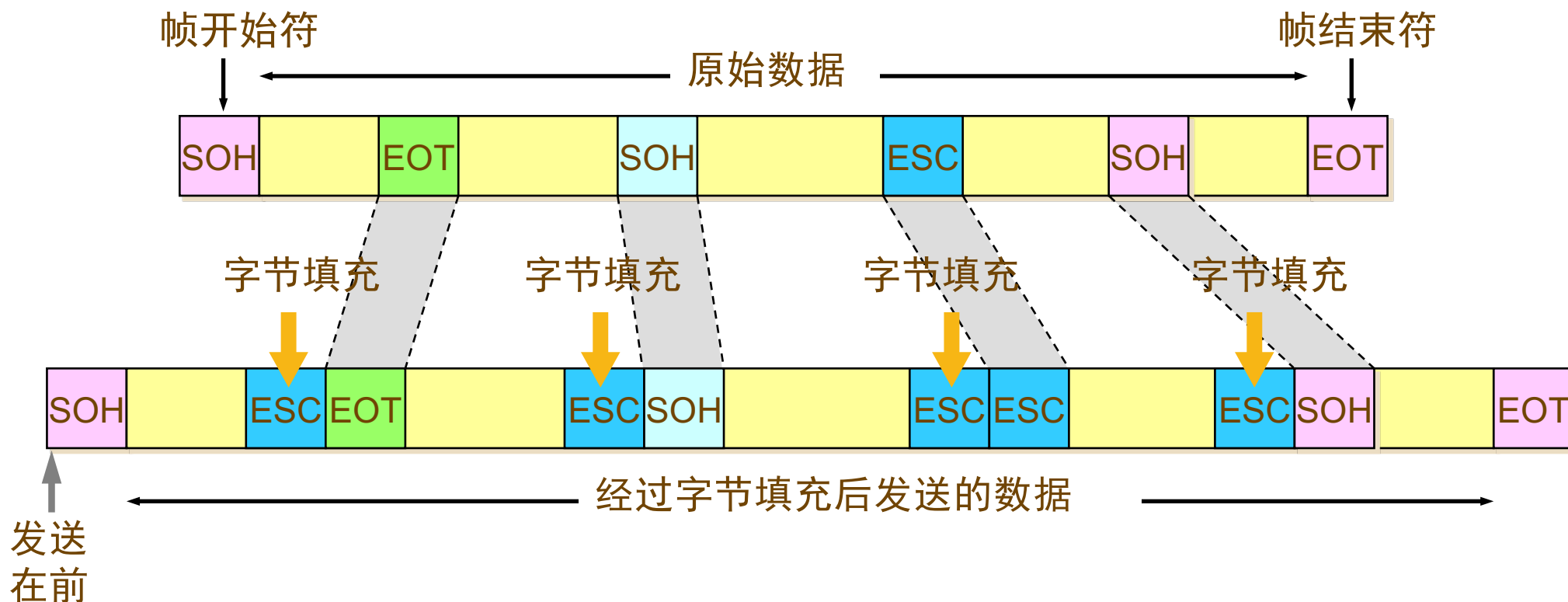
2. 透明传输



透明传输: 即在信息位中可以传送任何字符或比特组合，包括帧定界符。

解决透明传输问题

- 具体保证方法有：使用转义字符、比特插入。
- 例如：



示例说明：字节填充法解决透明传输

□ 如果采用字符/字节填充方法

- ▣ 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)。
- ▣ 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在数据当中，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。

3. 差错检测

- 在传输过程中可能会产生比特差错
 - ▣ 1 可能会变成 0，而 0 也可能变成 1。
- 为了保证数据传输的可靠性，必须采用各种差错检测措施。

误码率和误帧率

□ 误码率BER (Bit Error Rate)

- ▣ P_b = 发生错误的比特数/接收的总比特数
- ▣ 误码率与信噪比有很大的关系。

□ 误帧率

- ▣ 数据帧的差错率，即一帧中至少有一个比特出错的概率
- ▣ $P = 1 - (1 - P_b)^{L_f}$ ， P_b 为误码率， L_f 为帧位数
- ▣ 当 P_b 很小时可近似为 $P = L_f * P_b \ll 1$

例.

某信道误码率为 10^{-5} ，每帧长度为 10000bits, 试问

(1) 若差错都是单个错，则在该信道上传送的帧的平均出错率是多少？

(2) 若差错大多为突发错，平均突发长度为 100bits，则在该信道上传送的帧的平均出错率是多少？

(1) $10^4 * 10^{-5} = 0.1$

(2) $(10^4 / 100) * 10^{-5} = 10^{-3}$

成帧&信道利用率

- 信道的利用率不可能达到100%
 - ▣ 每个数据帧都必须包括一定的控制信息（如帧的序号，地址，同步信息及其他）。
 - ▣ 出现差错时，数据帧的不断重传也将会进一步降低信道利用率。
- 为提高信道利用率，在信道质量变化时，应该如何选择帧的大小？

结论

- 当信道质量好时，采用较长的数据帧可提高信道利用率。
 - ▣ 因为，误码率低，相应重传次数少，从而可增长帧达到减少控制信息的效果。
- 当信道质量不好时，就应采用较短的帧以便减少出错的概率。

适当的帧长

- 1. 过长的帧，出错重传要引起更多信道的浪费
- 2. 帧过长会引发设备资源使用的不公平
- 3. 帧过长对于其他的设备缓存空间是个考验
- 所以适当的帧长对于网络传输来说是很有必要的

差错检测措施

- 为了保证数据传输的可靠性，在计算机网络传输数据时，都必须采用各种差错检测措施。
- 常用检错码或纠错码来进行检测
 - ▣ 检错码：发现错误后配合重传机制形成差错控制
 - ▣ 纠错码：可以发现错误的准确位置，并直接改正，无须重传

最简单的校验方法

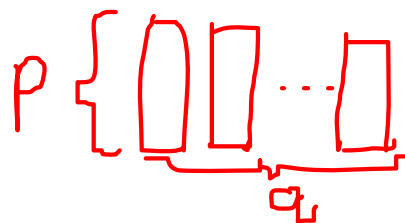
□ 奇偶校验码

- 一位冗余位作为校验位
- 码字中1的个数为奇数，则为奇校验。
- 码字中1的个数为偶数，则为偶校验。

□ 例如：原信息为1100100

- 奇校验则冗余位为0，码字为11001000；
- 偶校验则冗余位为1，码字为11001001
- 缺点？

改进



每段定长 P 位（每段为一个字符长度）， q 个字符组成一个信息块

- 垂直校验；水平校验；水平垂直校验

循环冗余检验

- 在数据链路层传送的帧中，广泛使用了循环冗余检验 CRC 的检错技术。
- 原理
 - ▣ 在发送端，先把数据划分为组。假定每组 k 个比特。
 - 例：假设待传送的一组数据 $M = 101001$ ，则 $k = 6$
 - ▣ 在 M 的后面再添加供差错检测用的 n 位冗余码一起发送。

冗余码的计算

- (1) 用二进制的模 2 运算进行 2^n 乘 M 的运算，这相当于在 M 后面添加 n 个 0。
- (2) 用得到的 $(k + n)$ 位的数，除以事先选定好的长度为 $(n + 1)$ 位的除数 P ，得出商 Q 和 余数 R
 - ▣ 余数 R 比除数 P 少 1 位，即 R 是 n 位
- 冗余码：余数 R

□ 编码思想

- ▣ 将位串看成系数为0或1的多项式

□ 检错思想

- ▣ 收发双方约定一个生成多项式 $G(x)$ ，其最高阶和最低阶系数必须为1
- ▣ 发送方在帧的末尾加上校验和，使带校验和的帧的多项式能被 $G(x)$ 整除
- ▣ 接收方收到后，用 $G(x)$ 除多项式，若有余数，则传输有错。

校验和的计算

- 设生成多项式 $G(x)$ 为 n 阶
- 在帧的末尾附加 n 个零，使帧为 $m+n$ 位，则相应的多项式是 $2^nM(x)$ 。
- 按模2除法用对应于 $G(x)$ 的位串除对应于 $2^nM(x)$ 的位串。
- 按模2减法从对应于 $2^nM(x)$ 的位串中减去余数。结果就是要传送带校验和的帧，记作多项式 $T(x)$ 。
 - ▣ 如果用被除数减去余数，则剩下的部分肯定能够被除数除尽。
 - ▣ 这种方法除了是 $G(x)$ 整数倍数据的多项式差错检测不到外，其他错误均能捕捉到，由此可看出它的检错率是非常高的。



□ 目前，常见的生成多项式**G(x)**国际标准有以下几种：

□ CRC-12 $G(x)=x^{12}+x^{11}+x^3+x^2+x+1$

□ CRC-16 $G(x)=x^{16}+x^{12}+x^2+1$

□ CRC-CCITT $G(x)=x^{16}+x^{12}+x^5+1$

$$\begin{array}{r}
 \overline{1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0} \\
 1\ 0\ 0\ 1\ 1\ \sqrt{1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0} \\
 \underline{1\ 0\ 0\ 1\ 1} \\
 1\ 0\ 0\ 1\ 1 \\
 \underline{1\ 0\ 0\ 1\ 1} \\
 0\ 0\ 0\ 0\ 1 \\
 \underline{0\ 0\ 0\ 0\ 0} \\
 0\ 0\ 0\ 1\ 0 \\
 \underline{0\ 0\ 0\ 0\ 0} \\
 0\ 0\ 1\ 0\ 1 \\
 \underline{0\ 0\ 0\ 0\ 0} \\
 0\ 1\ 0\ 1\ 1 \\
 \underline{0\ 0\ 0\ 0\ 0} \\
 1\ 0\ 1\ 1\ 0 \\
 \underline{1\ 0\ 0\ 1\ 1} \\
 0\ 1\ 0\ 1\ 0 \\
 \underline{0\ 0\ 0\ 0\ 0} \\
 1\ 0\ 1\ 0\ 0 \\
 \underline{1\ 0\ 0\ 1\ 1} \\
 0\ 1\ 1\ 1\ 0 \\
 \underline{0\ 0\ 0\ 0\ 0} \\
 1\ 1\ 1\ 0
 \end{array}$$

帧1101011011和 $G(x)=x^4+x+1$ 的算法。

帧：1101011011

除数：10011

附加4个零后形成的串：11010110000

传输的帧：11010110111110

举例：冗余码的计算

- 设 $k = 6$, $M = 101001$ 。
- 设 $n = 3$, 除数 $P = 1101$,
- 被除数是 $2^n M = 101001000$ 。
- 模 2 运算的结果是：商 $Q = 110101$,
余数 $R = 001$ 。
- 把余数 R 作为冗余码添加在数据 M 的后面发送出去。发送的数据是： $2^n M + R$
即：101001001，共 $(k + n)$ 位。

计算过程

$$\begin{array}{r} \text{110101} \leftarrow Q \text{ (商)} \\ P \text{ (除数)} \rightarrow 1101 \overline{) 101001000} \leftarrow 2^n M \text{ (被除数)} \\ \underline{1101} \\ 1110 \\ \underline{1101} \\ 0111 \\ \underline{0000} \\ 1110 \\ \underline{1101} \\ 0110 \\ \underline{0000} \\ 1100 \\ \underline{1101} \\ 001 \leftarrow R \text{ (余数)} \end{array}$$

帧检验序列 FCS

- 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 的唯一方法。

接收端对收到的每一帧进行 CRC 检验

- (1) 若得出的余数 $R = 0$ ，则判定这个帧没有差错，就接受(accept)。
- (2) 若余数 $R \neq 0$ ，则判定这个帧有差错，就丢弃。
- 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。
- 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。

应当注意

- 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”：
 - ▣ 凡是接受的帧（即**不包括丢弃的帧**），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
 - ▣ 也就是说：凡是接收端数据链路层接受的帧都没有传输差错（有差错的帧就丢弃而不接受）
- 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。

练习

- 1. 已知信息字段为 $I=1101$ ，生成多项式为 $G(x) = x^4 + x + 1$ ，求出CRC校验码。

作业

□ 7,8