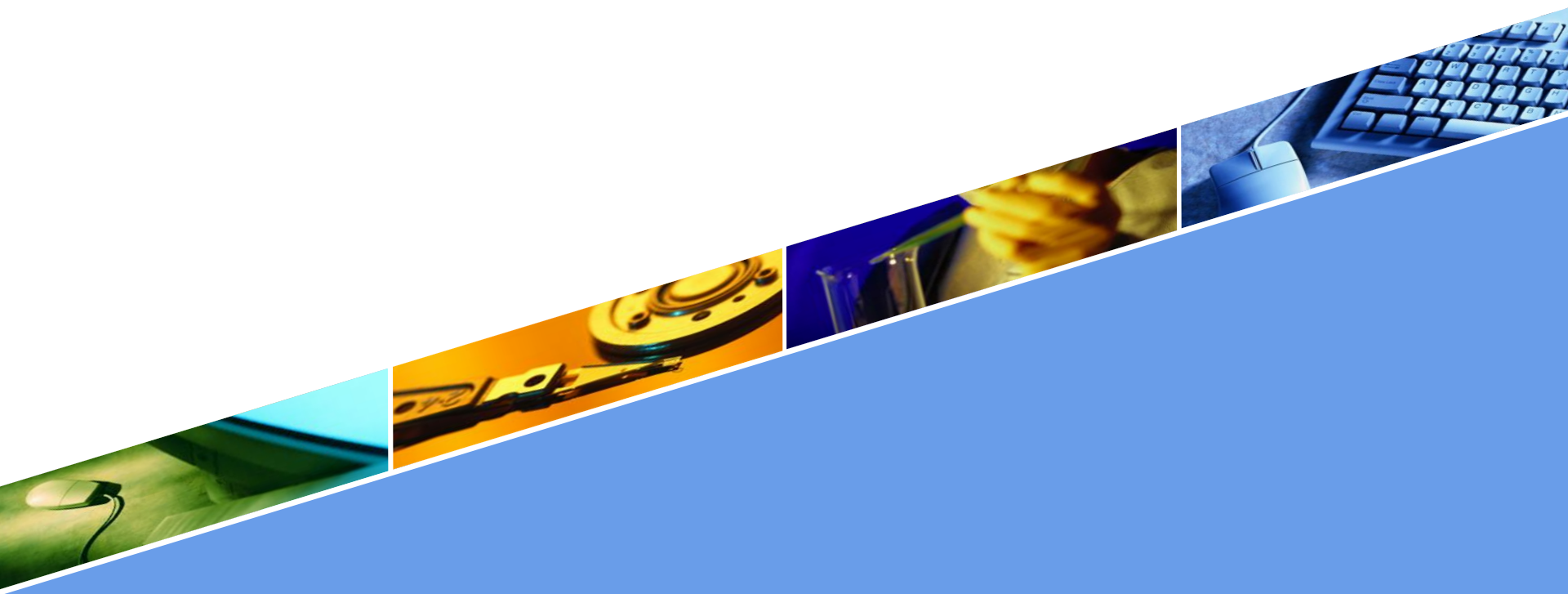


第三讲 数据链路层基础



主题 1



1 数据链路层概述

2 局域网的数据链路层

3 以太网技术

数据链路

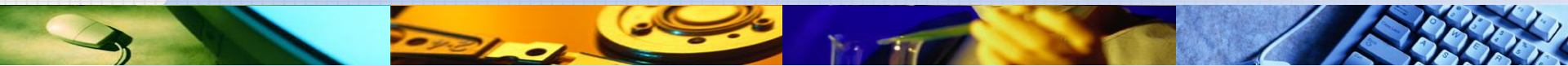
❖ **链路**(link)是一条无源的点到点的物理线路段，中间没有任何其他的交换结点。

- 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，还必须要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。

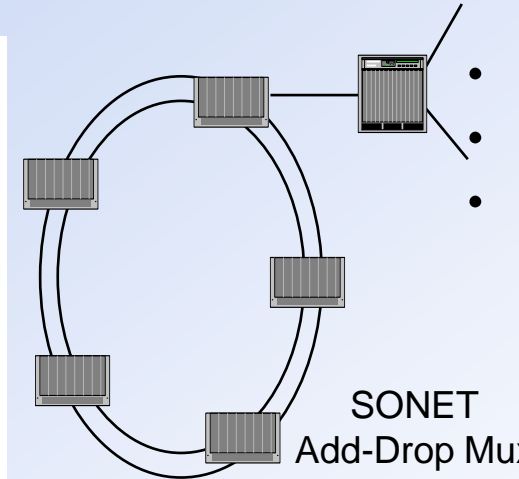
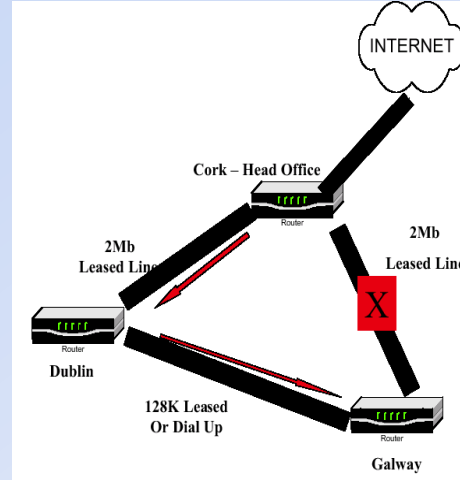
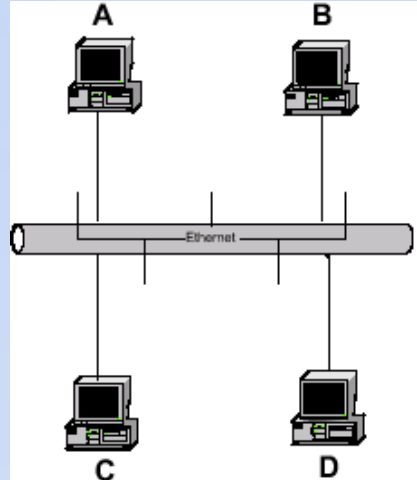
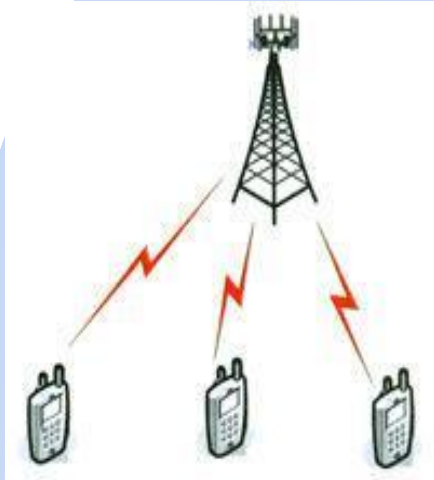
数据链路信道类型



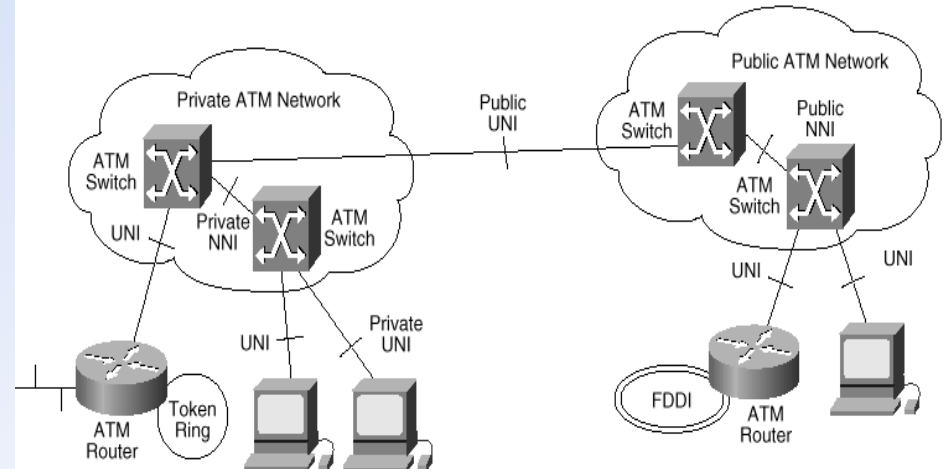
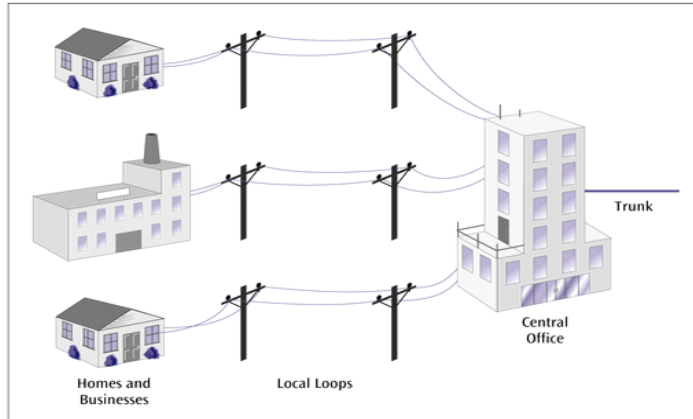
数据链路使用的信道主要有以下两种类型：

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

数据链路的例子

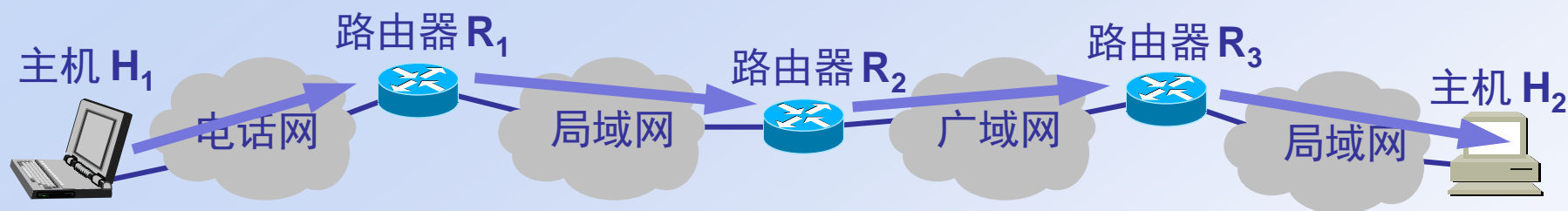


The local loop as it connects your house to the telephone company's central office

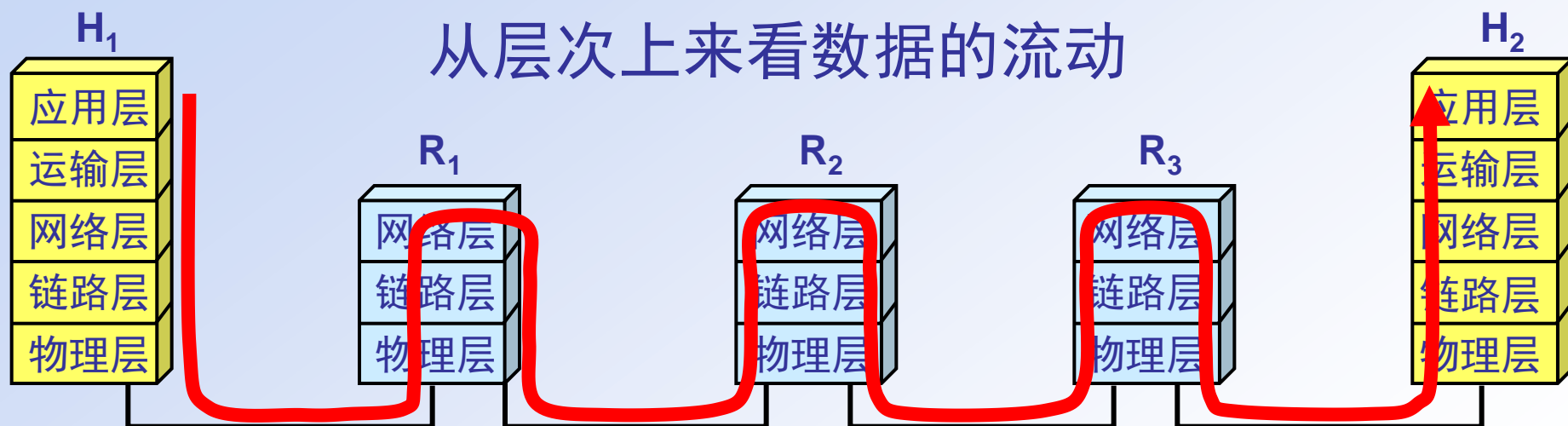


数据链路层的简单模型

主机 H_1 向 H_2 发送数据

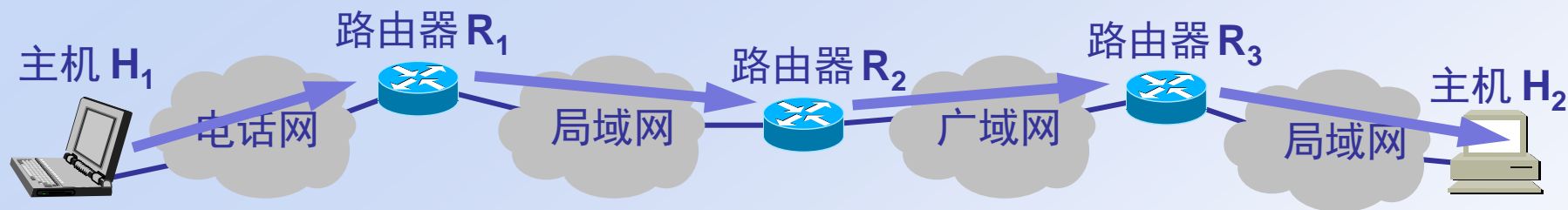


从层次上来看数据的流动

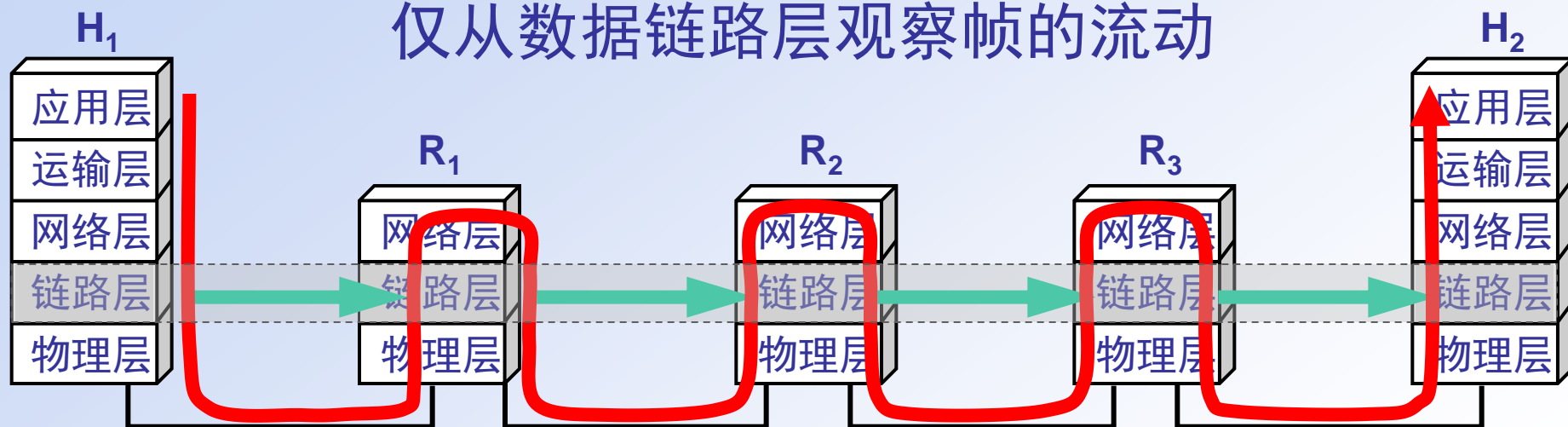


数据链路层的简单模型（续）

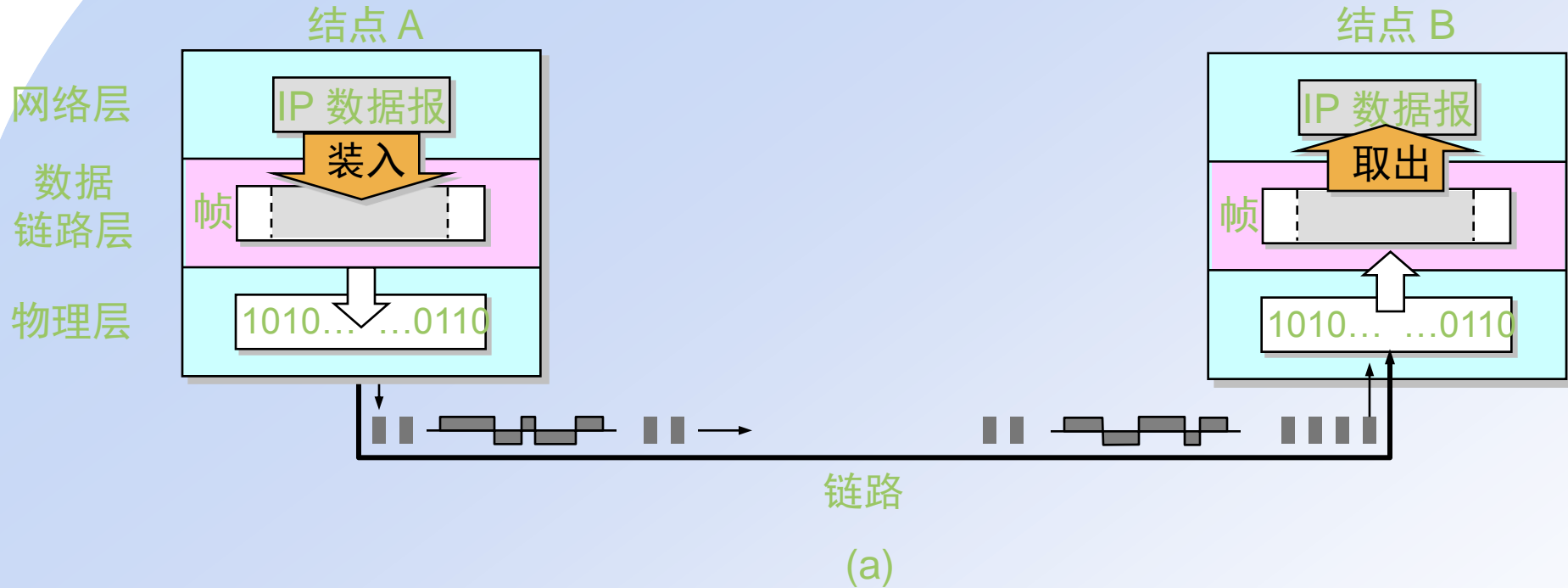
主机 H_1 向 H_2 发送数据



仅从数据链路层观察帧的流动



数据链路层传送的是帧



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。

链路层三个基本问题



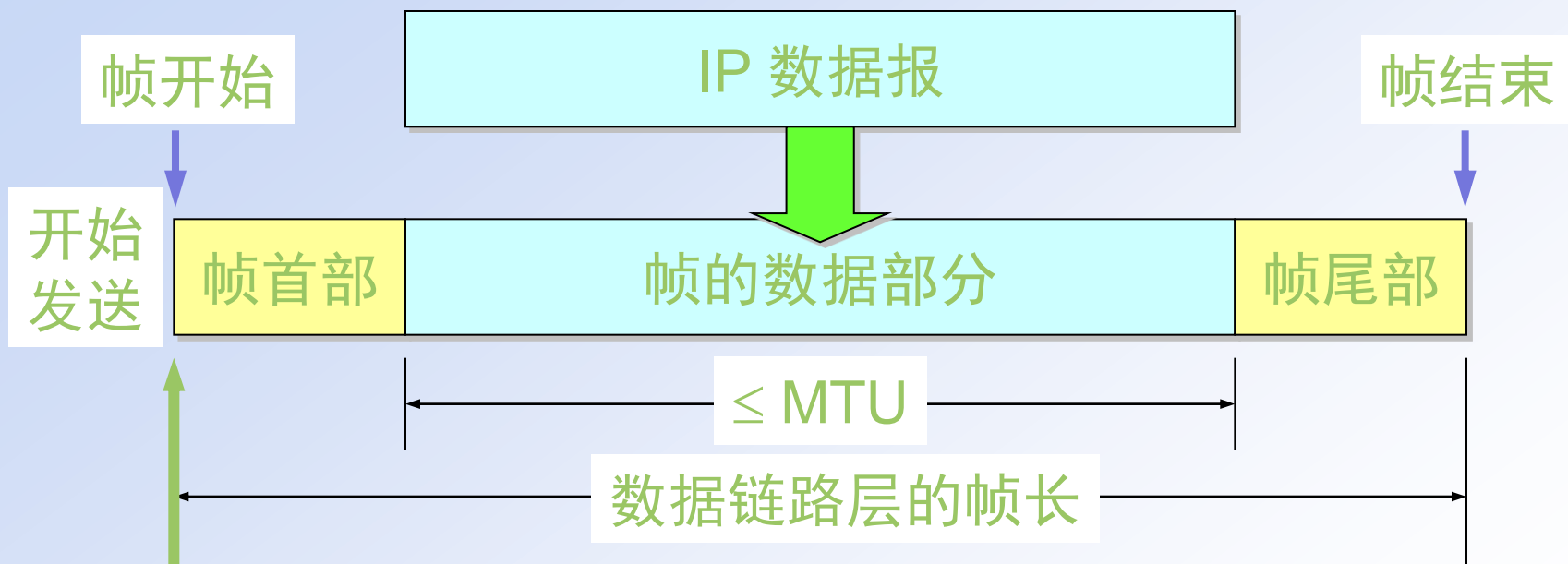
(1) 封装成帧

(2) 透明传输

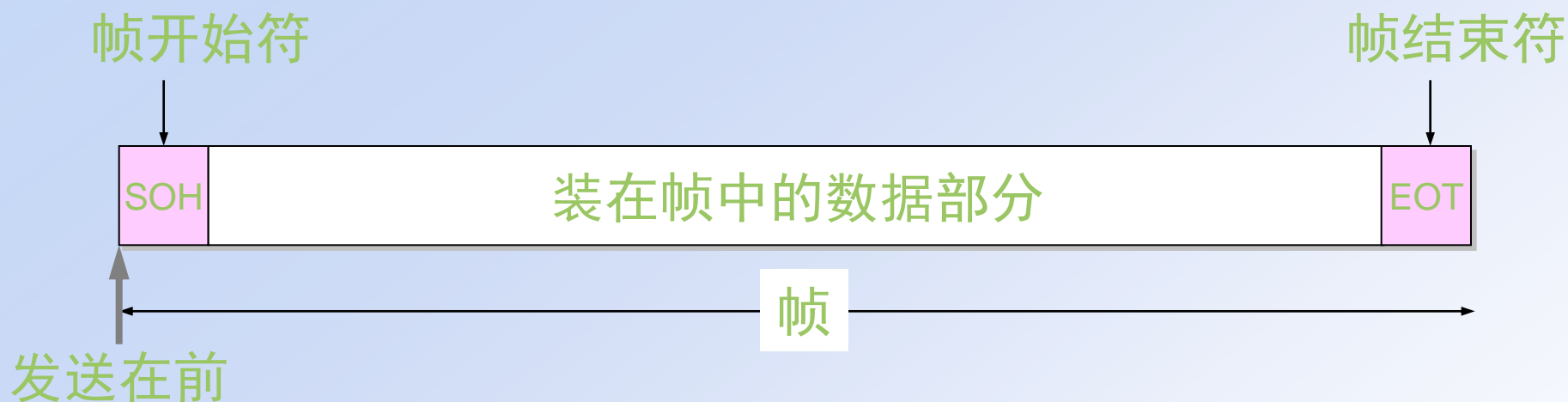
(3) 差错控制

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，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。



用控制字符进行帧定界的方法举例

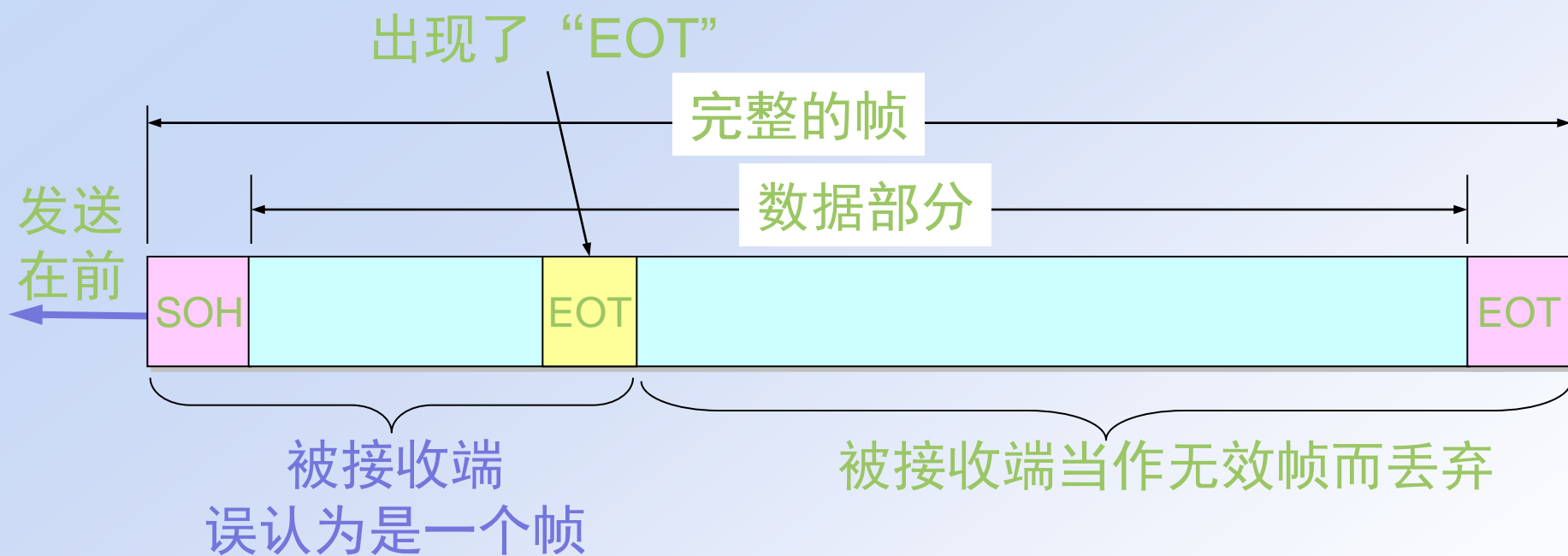


SOH和EOT是两个控制字符的名称：

SOH : 0X01

EOT : 0X04

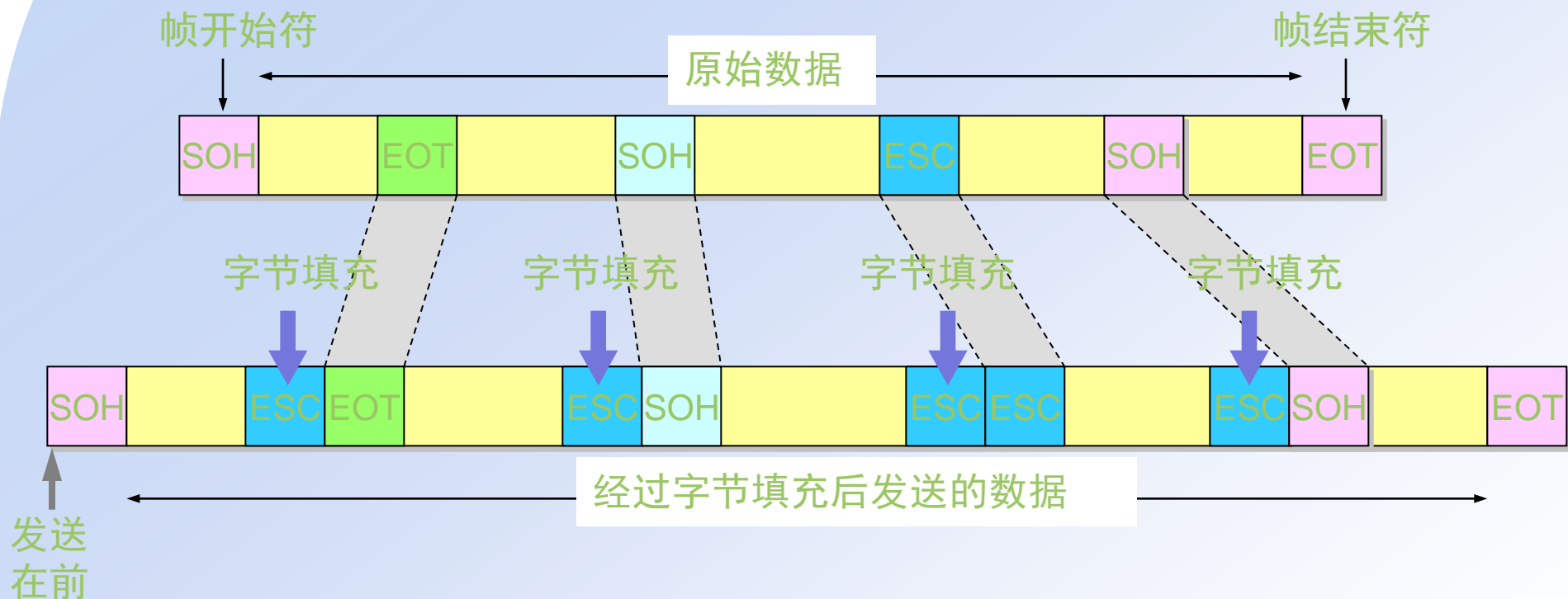
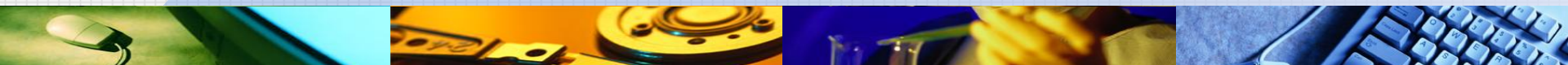
2. 透明传输



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. 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。
- ❖ 在一段时间内，传输错误的比特占所传输比特总数的比率称为误码率 BER (Bit Error Rate)。
- ❖ 误码率与信噪比有很大的关系。
- ❖ 为了保证数据传输的可靠性，在计算机网络传输数据时，必须采用各种差错检测措施。

循环冗余检验的原理

- ❖ 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。
- ❖ 在发送端，先把数据划分为组。假定每组 k 个比特。
- ❖ 假设待传送的一组数据 $M = 101001$ （现在 $k = 6$ ）。我们在 M 的后面再添加供差错检测用的 n 位冗余码一起发送。

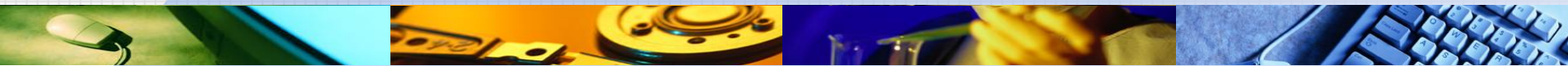
冗余码的计算

- ❖ 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。
- ❖ 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 ，得出商是 Q 而余数是 R ，余数 R 比除数 P 少 1 位，即 R 是 n 位。

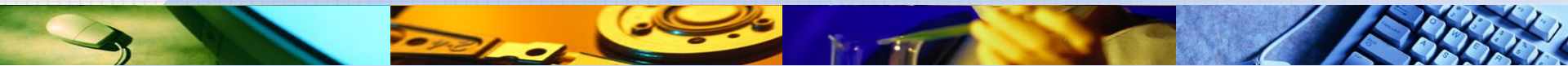
冗余码的计算举例

- ❖ 现在 $k = 6$, $M = 101001$ 。
- ❖ 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。

循环冗余检验的原理说明

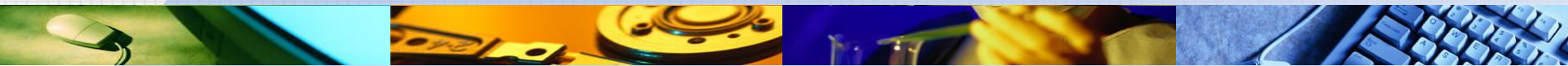

$$\begin{array}{r} 110101 \leftarrow Q \text{ (商)} \\ P \text{ (除数)} \rightarrow 1101 \overline{) 101001000} \leftarrow 2^n M \text{ (被除数)} \\ \underline{1101} \\ 1110 \\ \underline{1101} \\ 0111 \\ \underline{0000} \\ 1110 \\ \underline{1101} \\ 0110 \\ \underline{0000} \\ 1100 \\ \underline{1101} \\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

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



- ❖ 把接收到的帧除以同样的除数 P
- ❖ (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。
- ❖ 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。
- ❖ 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。

生成多项式

❖ 用生成多项式 $P(X)$ 表示除数 P

❖ 在CRC中广泛使用的生成多项式 $P(X)$ 有：

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

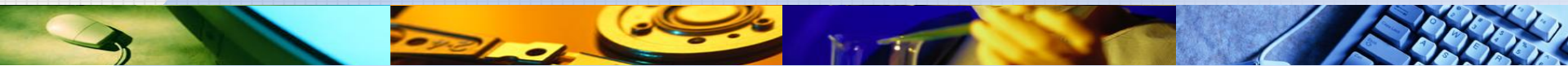
- $\text{CRC-CCITT} = X^{16} + X^{12} + X^5 + 1$

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

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”是指：“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。

数据链路控制协议分类



- ❖ 点到点链路控制协议——用于广域网中点到点链路控制
 - **PPP协议**
- ❖ 广播链路控制协议——用于广播信道的局域网链路控制
 - **以太网协议**
 - **无线局域网协议**
 - **令牌环网协议等**

本课程重点：常用的以太网协议

主题 2

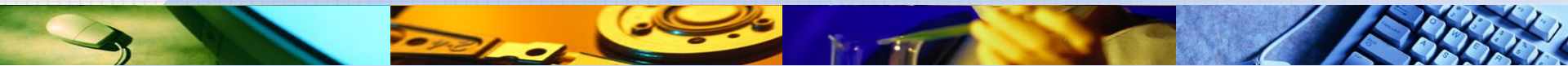


1 数据链路层概述

2 局域网的数据链路层

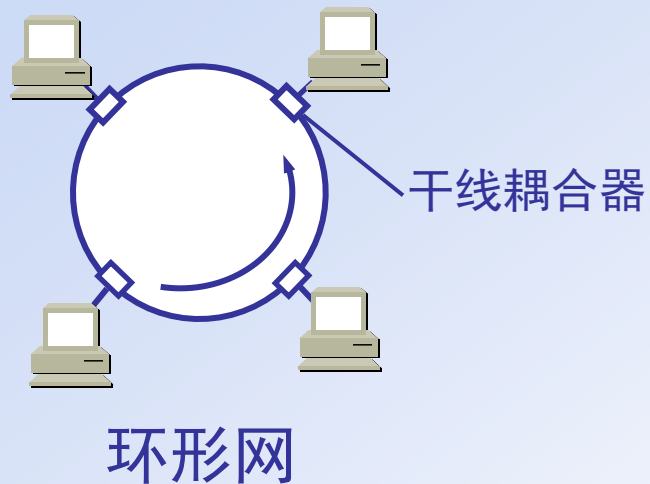
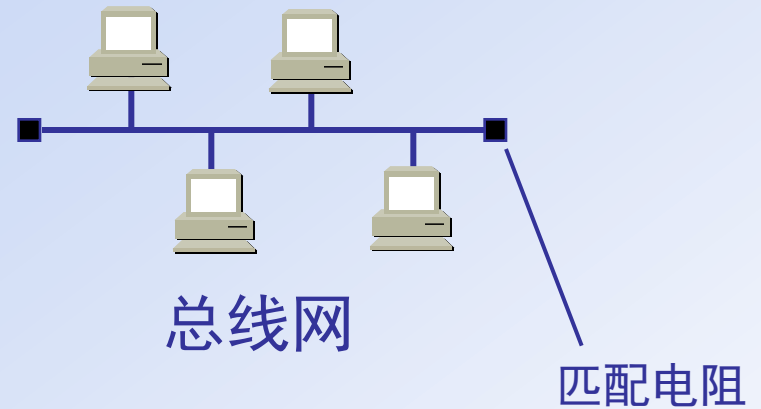
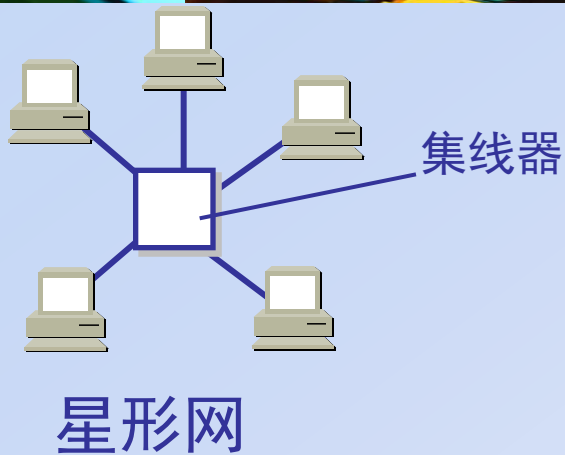
3 以太网技术

什么是局域网



- ❖ 局域网最主要的特点是：共享介质，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：
 - 使用广播信道，具有广播功能，从一个站点可很方便地访问全网。
 - 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。

局域网的拓扑—共享媒体（介质）

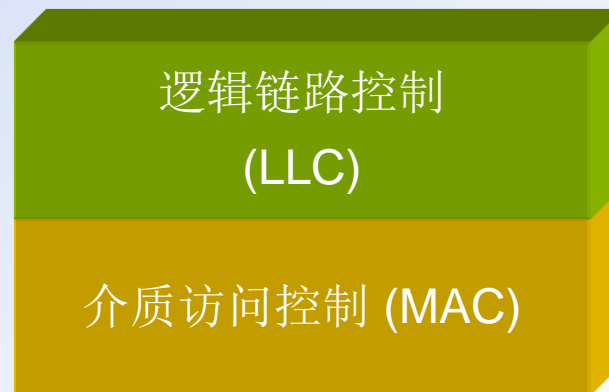


局域网模型



逻辑链路控制 (LLC) 提供了传统的 HDLC 类型的协议

介质访问控制层 (MAC) 按照预先设定的规则对物理信道进行访问控制



IEEE 802 局域网标准



OSI 分层

IEEE 802 局域网标准

高层

高层

数据链路层

802.2 逻辑链路控制

802.3

802.4

802.5

802.11

介质访问控制

物理层

以太网
总线

令牌传递
总线

令牌传递
环网

无线局域网

数据链路层的两个子层

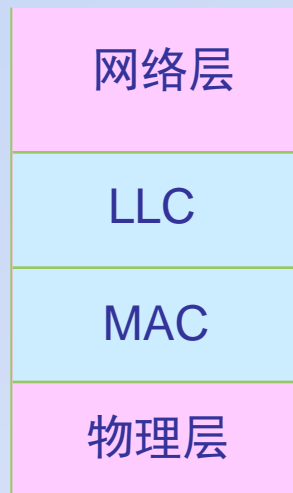
- ❖ 为了使数据链路层能更好地适应多种局域网标准，802 委员会就将局域网的数据链路层拆成两个子层：
 - 逻辑链路控制 LLC (Logical Link Control)子层
 - 媒体接入控制 MAC (Medium Access Control)子层
- ❖ 与接入到传输媒体有关的内容都放在 MAC子层，而 LLC 子层则与传输媒体无关，不管采用何种协议的局域网对 LLC 子层来说都是透明的

局域网对 LLC 子层是透明的

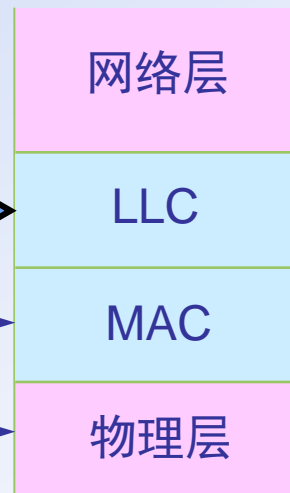
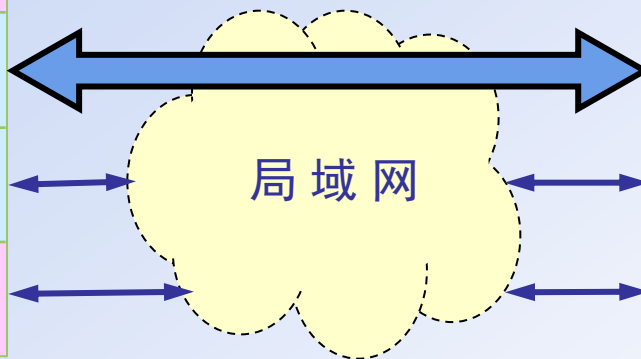
LLC 子层看不见
下面的局域网

逻辑链路控制

媒体接入控制



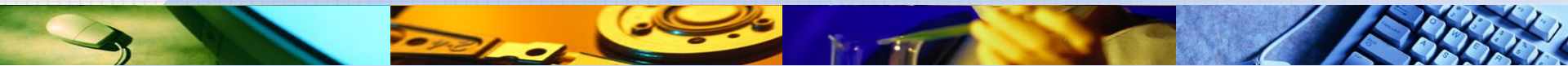
站点 1



站点 2

数据
链路层

以太网的标准



- ❖ DIX Ethernet V2 是世界上第一个局域网产品（以太网）的规约。
- ❖ IEEE 的 802.3 标准。
- ❖ DIX Ethernet V2 标准与 IEEE 的 802.3 标准只有很小的差别，因此可以将 802.3 局域网简称为“以太网”。
- ❖ 严格说来，“以太网”应当是指符合 DIX Ethernet V2 标准的局域网

以后一般不考虑 LLC 子层

- ❖ 由于 TCP/IP 体系经常使用的局域网是 DIX Ethernet V2 而不是 802 标准中的几种局域网，因此现在 802 委员会制定的逻辑链路控制子层 LLC（即 802.2 标准）的作用已经不大了。
- ❖ 很多厂商生产的适配器上就仅装有 MAC 协议而没有 LLC 协议。

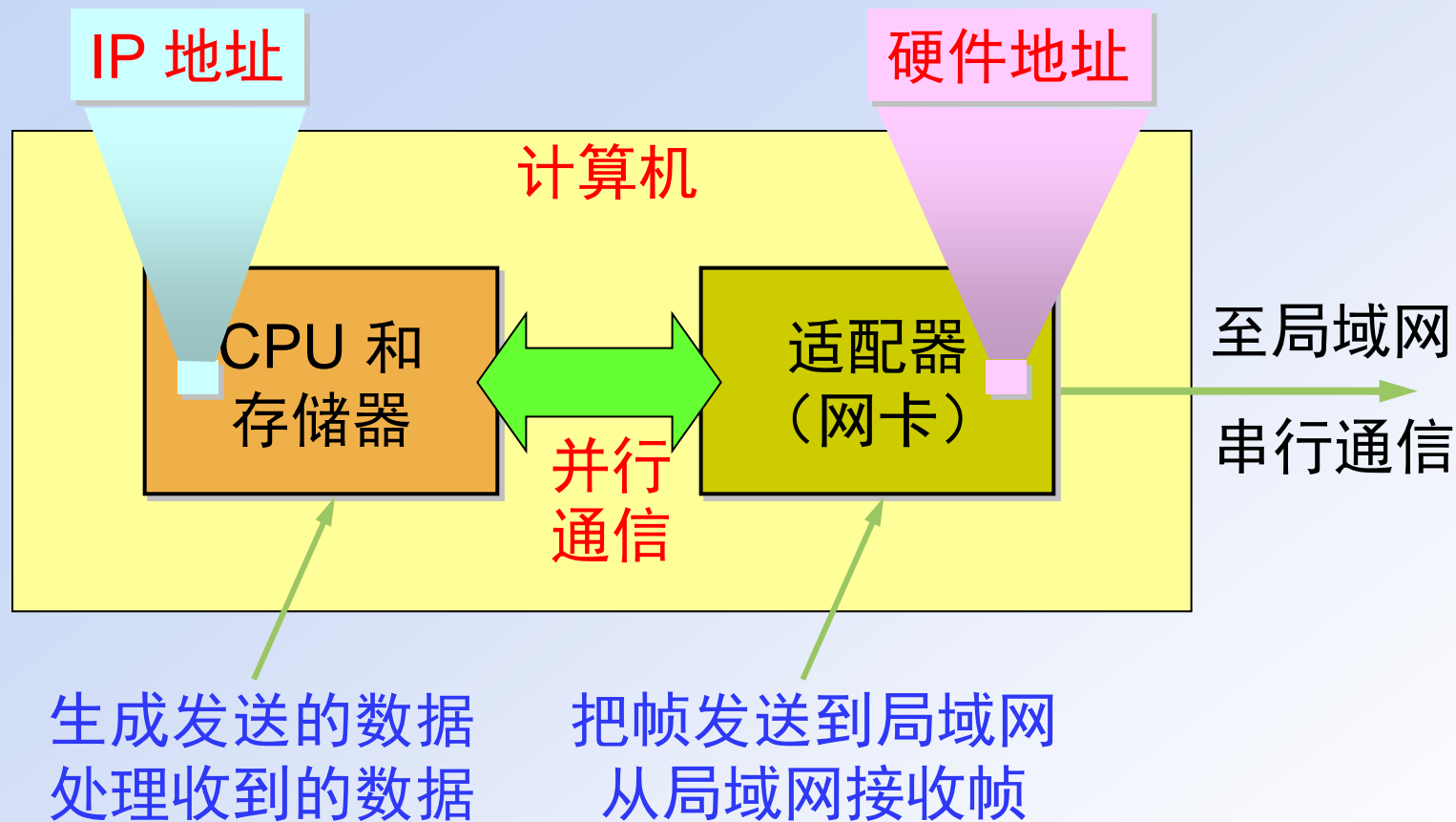
适配器的作用

❖ 网络接口板又称为**通信适配器**(adapter)或**网络接口卡** NIC (Network Interface Card)，或“**网卡**”。

❖ 适配器的主要功能：

- 进行串行/并行转换。
- 对数据进行缓存。
- 在计算机的操作系统安装设备驱动程序。
- 实现以太网协议。

计算机通过适配器和局域网进行通信



主题 3

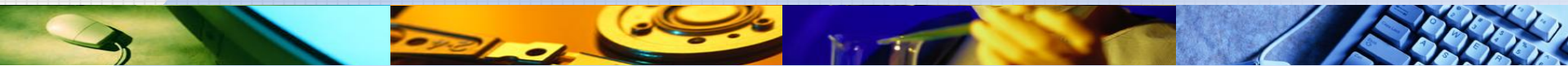


1 数据链路层概述

2 局域网的数据链路层

3 以太网技术

以太网MAC 层的硬件地址



- ❖ 在局域网中，硬件地址又称为物理地址，或MAC地址。
- ❖ 802 标准所说的“地址”严格地讲应当是每一个站的“名字”或标识符。

48 位的 MAC 地址

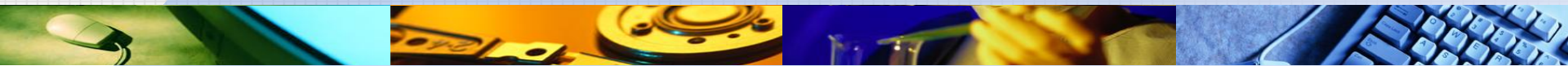
- ❖ IEEE 的注册管理机构 RA 负责向厂家分配地址字段的前三个字节(即高位 24 位)。
- ❖ 地址字段中的后三个字节(即低位 24 位)由厂家自行指派, 称为扩展标识符, 必须保证生产出的适配器没有重复地址。
- ❖ 一个地址块可以生成 2^{24} 个不同的地址。这种 48 位地址称为 MAC-48, 它的通用名称是 EUI-48。
- ❖ “MAC 地址” 实际上就是适配器地址或适配器标识符 EUI-48。

适配器检查 MAC 地址



- ❖ 适配器从网络上每收到一个 MAC 帧就首先用硬件检查 MAC 帧中的 MAC 地址。
 - 如果是发往本站的帧则收下，然后再进行其他的处理。
 - 否则就将此帧丢弃，不再进行其他的处理。
- ❖ “发往本站的帧” 包括以下三种帧：
 - 单播(unicast)帧（一对一）
 - 广播(broadcast)帧（一对全体）
 - 多播(multicast)帧（一对多）

MAC 帧的格式

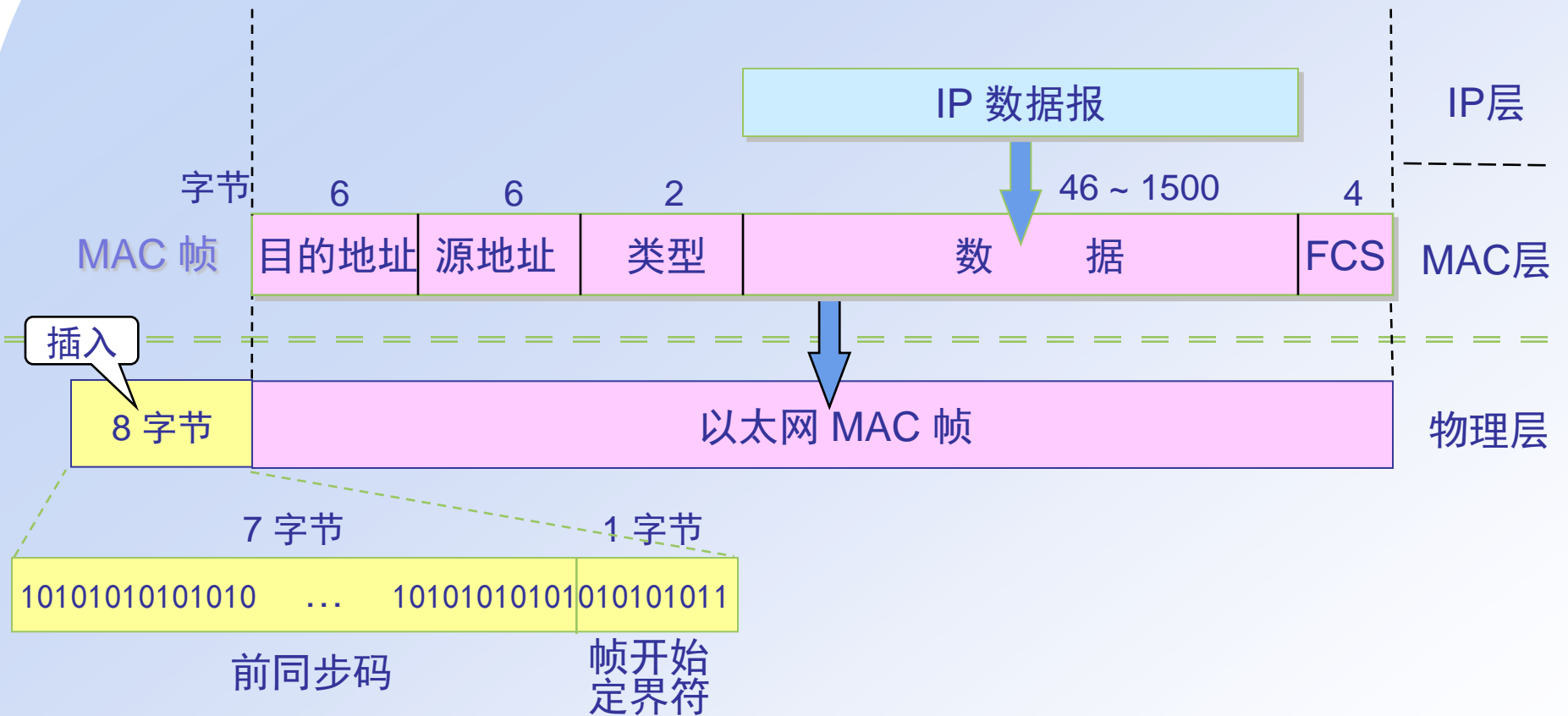


❖ 常用的以太网MAC帧格式有两种标准：

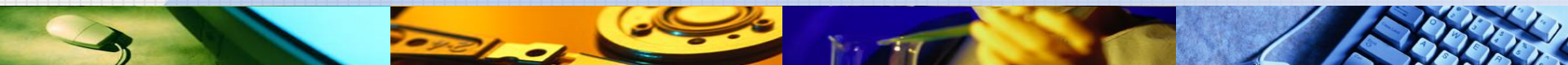
- DIX Ethernet V2 标准
- IEEE 的 802.3 标准

❖ 最常用的 MAC 帧是以太网 V2 的格式。

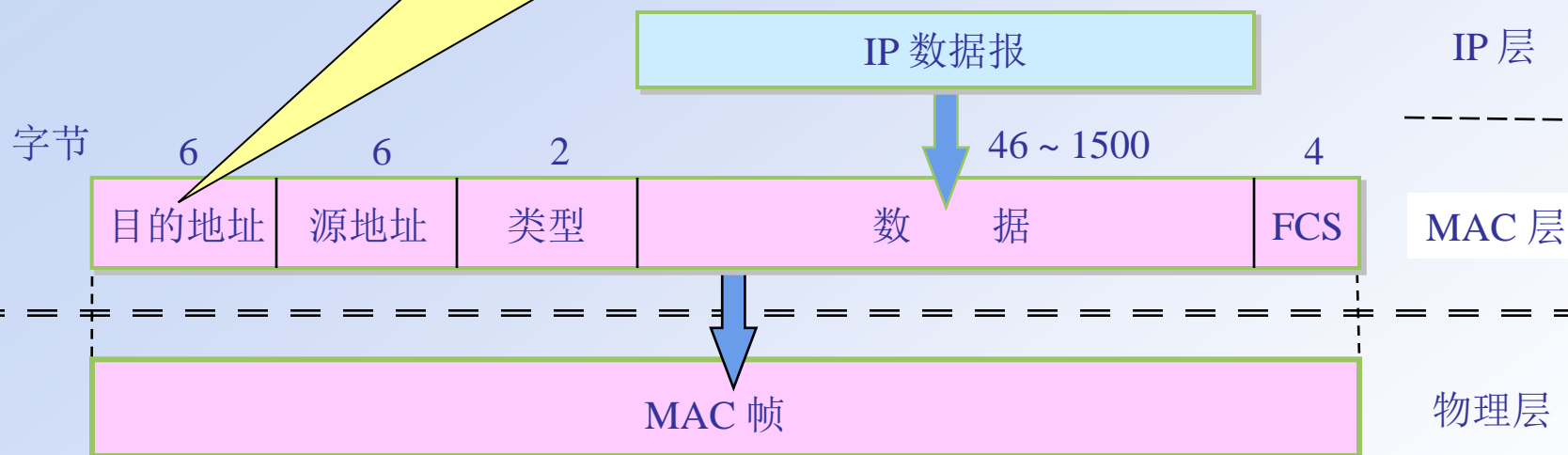
以太网的 MAC 帧格式



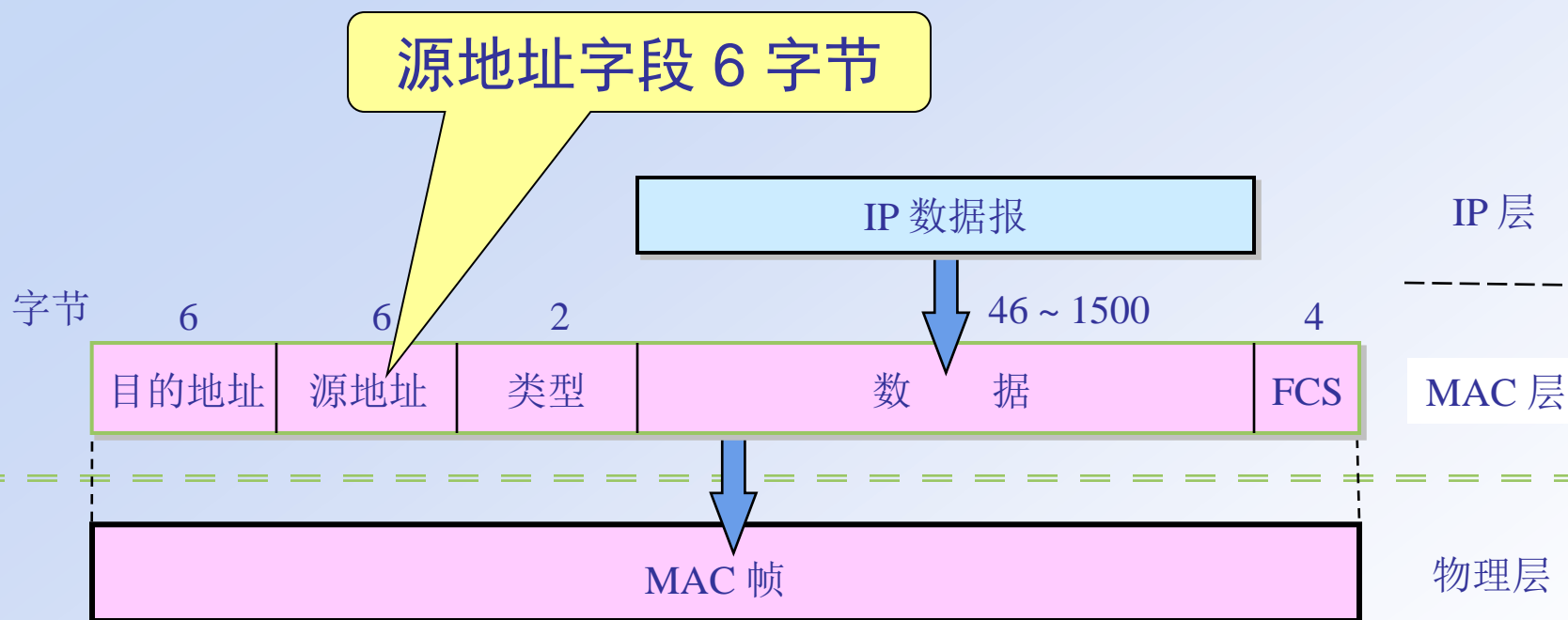
以太网 V2 的 MAC 帧格式



目的地址字段 6 字节



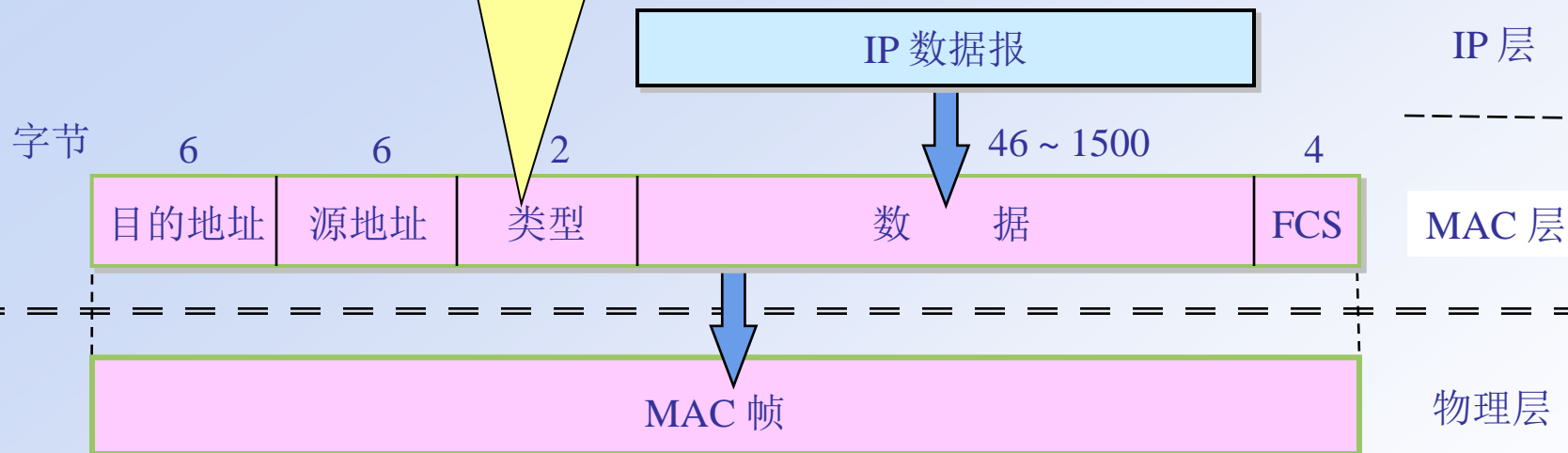
以太网 V2 的 MAC 帧格式



以太网 V2 的 MAC 帧格式

类型字段用来标志上一层使用的是什麼协议，以便把收到的 MAC 帧的数据上交给上一层的这个协议。

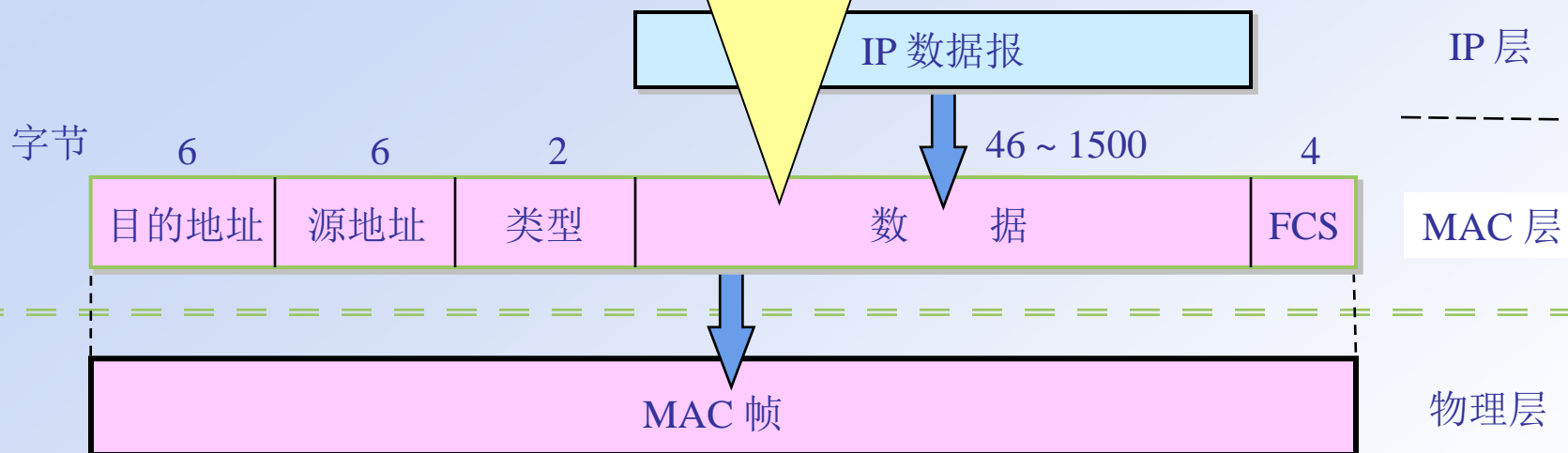
类型字段 2 字节??



以太网 V2 的 MAC 帧格式

数据字段的正式名称是 **MAC 客户数据字段**
最小长度 64 字节 – 18 字节的首部和尾部 = 数据字段的最小长度

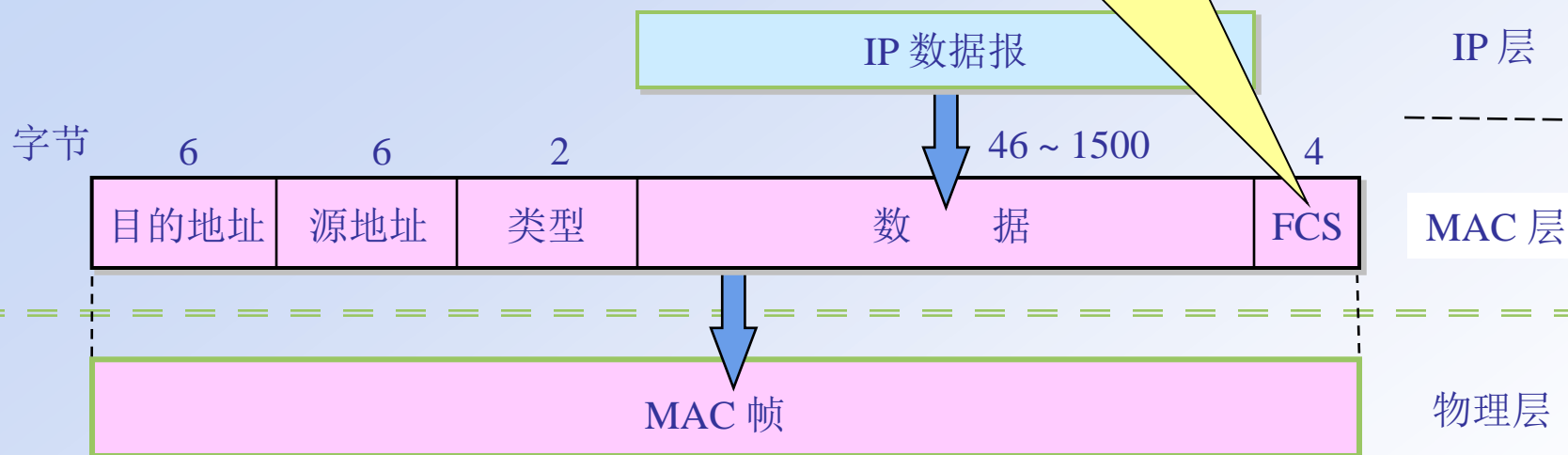
数据字段 46 ~ 1500 字节



以太网 V2 的 MAC 帧格式

当传输媒体的误码率为 1×10^{-8} 时，
MAC 子层可使未检测到的差错小于 1×10^{-14} 。

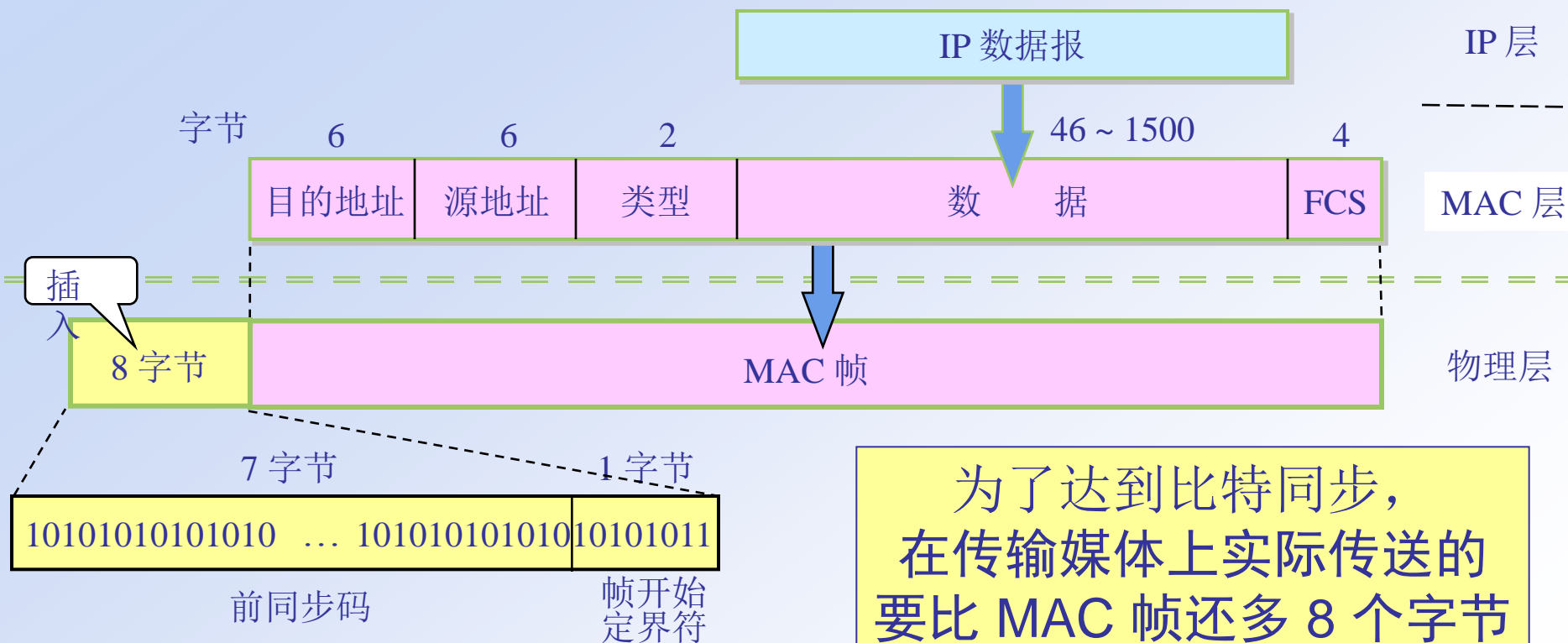
FCS 字段 4 字节



当数据字段的长度小于 46 字节时，
应在数据字段的后面加入整数字节的填充字段，
以保证以太网的 MAC 帧长不小于 64 字节。

以太网 V2 的 MAC 帧格式

在帧的前面插入的 8 字节中的第一个字段共 7 个字节，是前同步码，用来迅速实现 MAC 帧的比特同步。第二个字段是帧开始定界符，表示后面的信息就是 MAC 帧。



无效的 MAC 帧

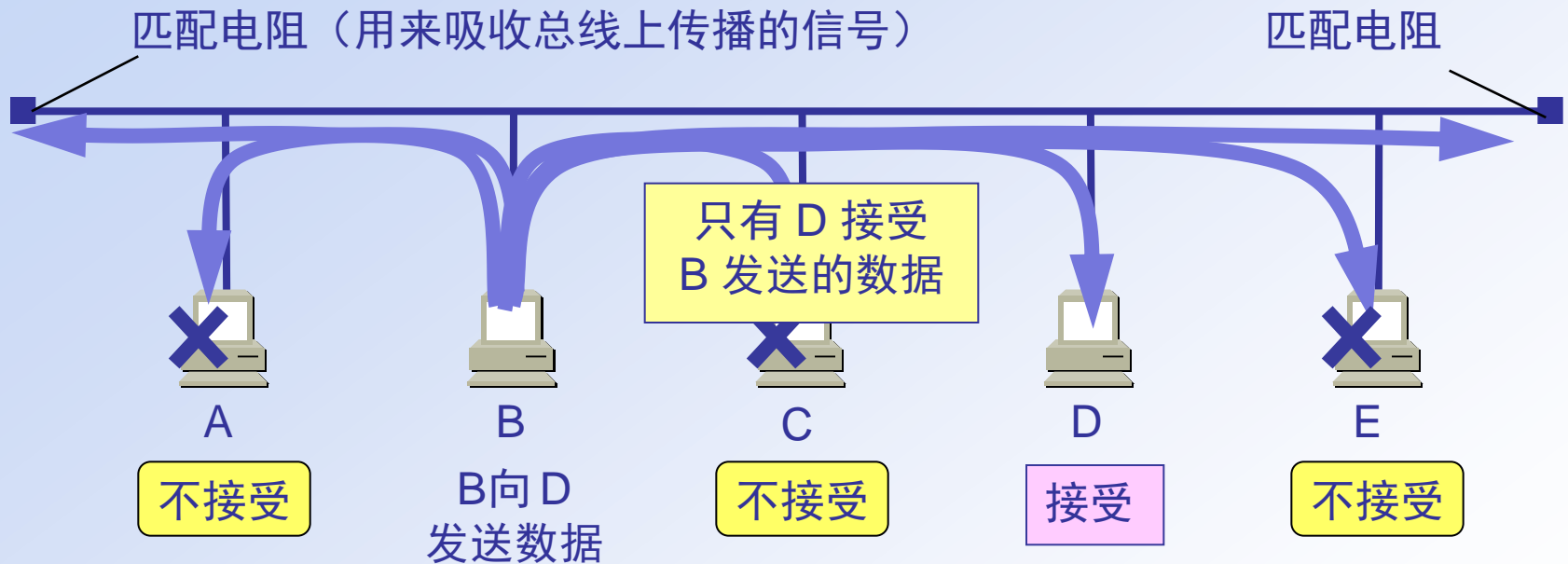
- ❖ 帧的长度不是整数个字节；
- ❖ 用收到的帧检验序列 FCS 查出有差错；
- ❖ 数据字段的长度不在 46 ~ 1500 字节之间。
- ❖ 有效的 MAC 帧长度为 64 ~ 1518 字节之间。
- ❖ 对于检查出的无效 MAC 帧就简单地丢弃。以太网不负责重传丢弃的帧。

帧间最小间隔

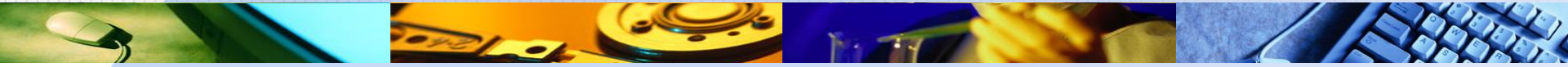
- ❖ 帧间最小间隔为 $9.6\ \mu\text{s}$ ，相当于 96 bit 的发送时间。
- ❖ 一个站在检测到总线开始空闲后，还要等待 $9.6\ \mu\text{s}$ 才能再次发送数据。
- ❖ 这样做是为了使刚刚收到数据帧的站的接收缓存来得及清理，做好接收下一帧的准备。

最初的以太网——总线型以太网

❖ 最初的以太网是将许多计算机都连接到一根总线上。当初认为这样的连接方法既简单又可靠，因为总线上没有有源器件。

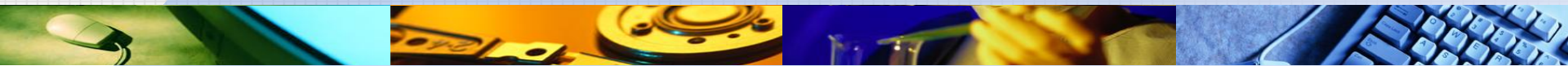


以太网的广播方式发送



- ❖ 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。

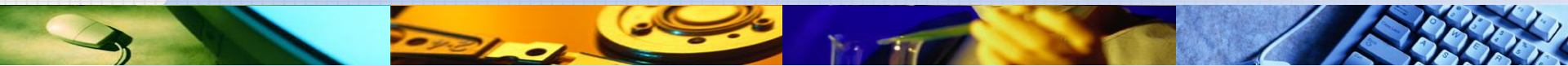
以太网通信



为了通信的简便，以太网采取了两种重要的措施：

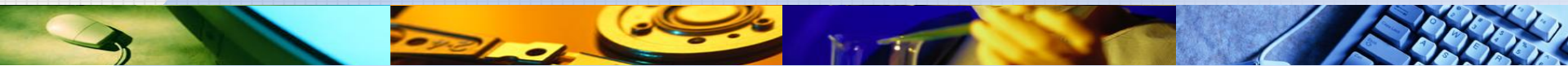
- ❖ 采用较为灵活的无连接的工作方式，即不必先建立连接就可以直接发送数据。
- ❖ 以太网对发送的数据帧不进行编号，也不要求对方发回确认。
 - 这样做的理由是局域网信道的质量很好，因信道质量产生差错的概率是很小的。
- ❖ 以太网发送的数据都使用曼彻斯特(Manchester)编码

以太网提供的服务



- ❖ 以太网提供的服务是不可靠的交付，即尽最大努力的交付。
- ❖ 当目的站收到有差错的数据帧时就丢弃此帧，其他什么也不做。差错的纠正由高层来决定。
- ❖ 如果高层发现丢失了一些数据而进行重传，但以太网并不知道这是一个重传的帧，而是当作一个新的数据帧来发送。

以太网的MAC协议——CSMA/CD

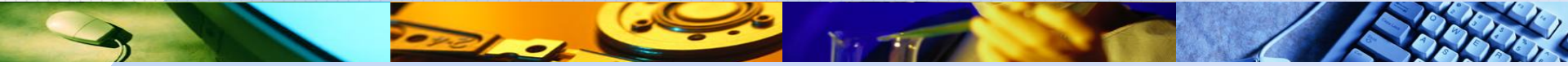


- ❖ 载波监听多点接入/碰撞（冲突）检测
CSMA/CD
- ❖ CSMA/CD 表示 Carrier Sense Multiple Access with Collision Detection。
- ❖ “多点接入”表示许多计算机以多点接入的方式连接在一根总线上。
- ❖ “载波监听”是指每一个站在发送数据之前先要检测一下总线上是否有其他计算机在发送数据，如果有，则暂时不要发送数据，以免发生碰撞。

碰撞检测

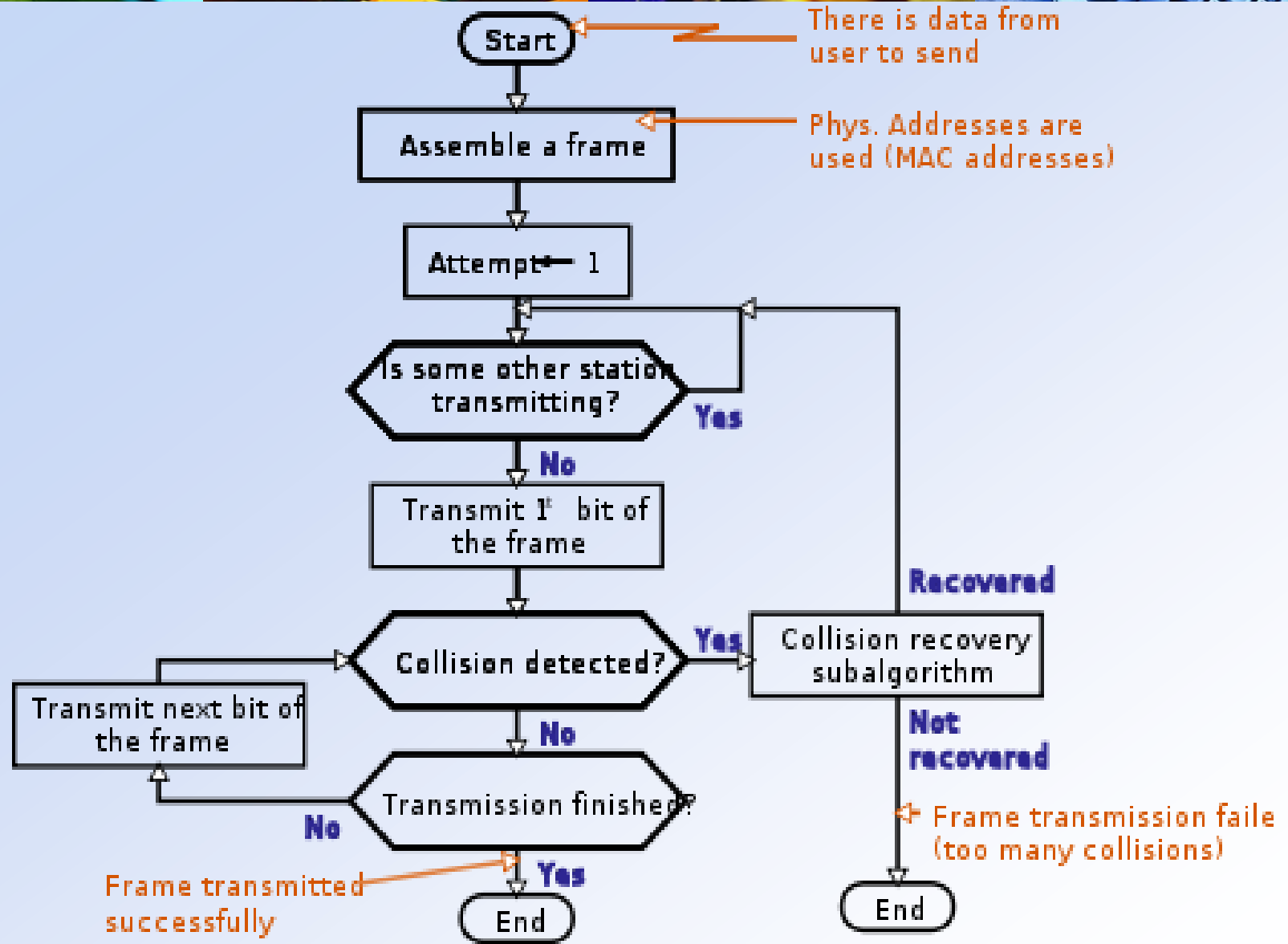
- ❖ “碰撞检测”就是计算机边发送数据边检测信道上的信号电压大小。
- ❖ 当几个站同时在总线上发送数据时，总线上的信号电压摆动值将会增大（互相叠加）。
- ❖ 当一个站检测到的信号电压摆动值超过一定的门限值时，就认为总线上至少有两个站同时在发送数据，表明产生了碰撞。
- ❖ 所谓“碰撞”就是发生了冲突。因此“碰撞检测”也称为“冲突检测”。

检测到碰撞后

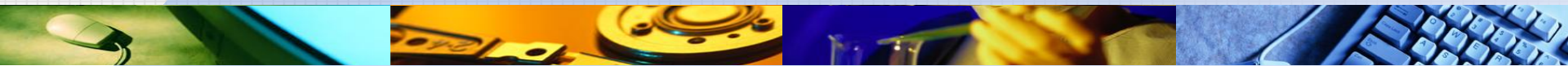


- ❖ 在发生碰撞时，总线上传输的信号产生了严重的失真，无法从中恢复出有用的信息来。
- ❖ 每一个正在发送数据的站，一旦发现总线上出现了碰撞，就要立即停止发送，免得继续浪费网络资源，然后等待一段随机时间后再次发送。

CSMA/CD算法

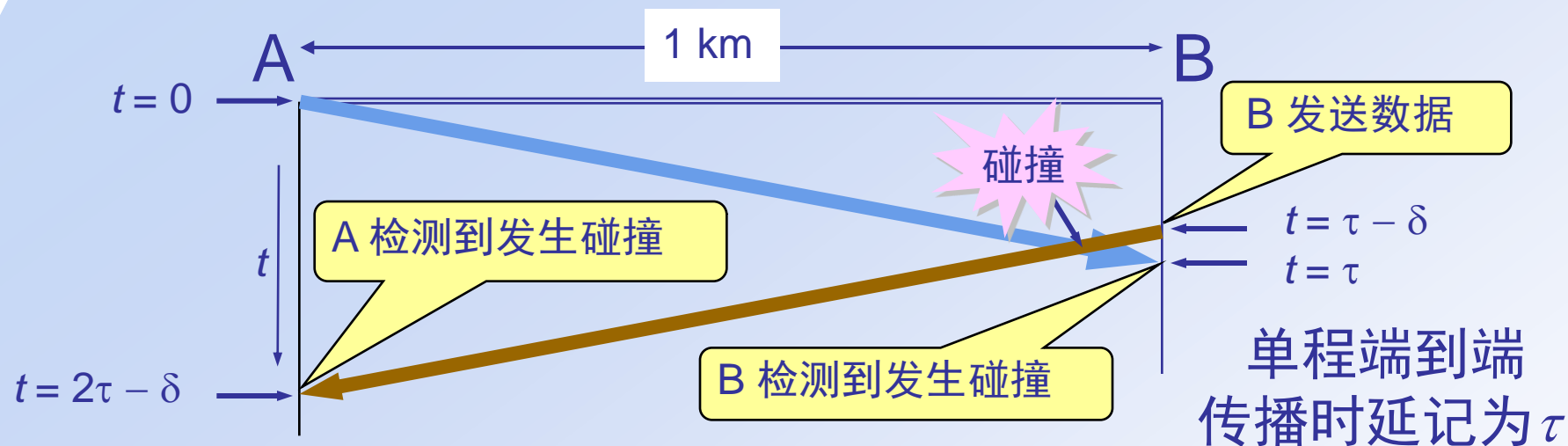


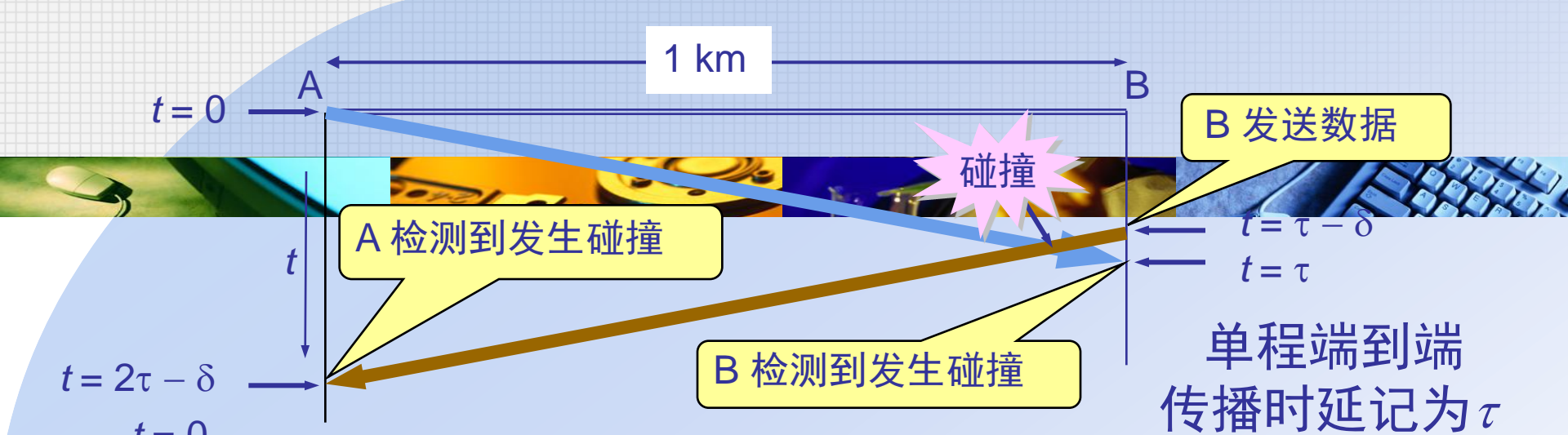
电磁波在总线上的有限传播速率的影响



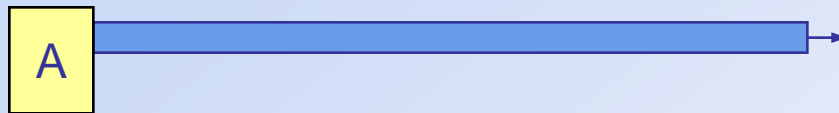
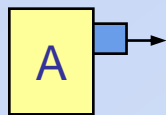
- ❖ 当某个站监听到总线是空闲时，也可能总线并非真正是空闲的。
- ❖ A 向 B 发出的信息，要经过一定的时间后才能传送到 B。
- ❖ B 若在 A 发送的信息到达 B 之前发送自己的帧（因为这时 B 的载波监听检测不到 A 所发送的信息），则必然要在某个时间和 A 发送的帧发生碰撞。
- ❖ 碰撞的结果是两个帧都变得无用。

传播时延对载波监听的影响

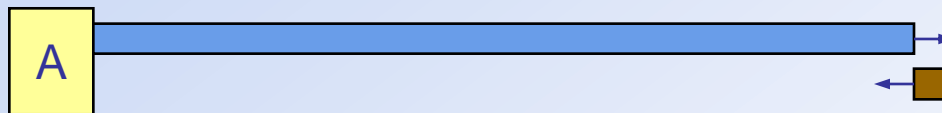
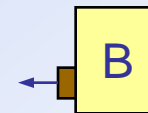




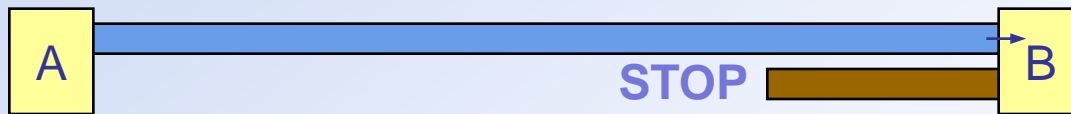
$t = 0$
A 检测到信道空闲
发送数据



$t = \tau - \delta$
B 检测到信道空闲
发送数据

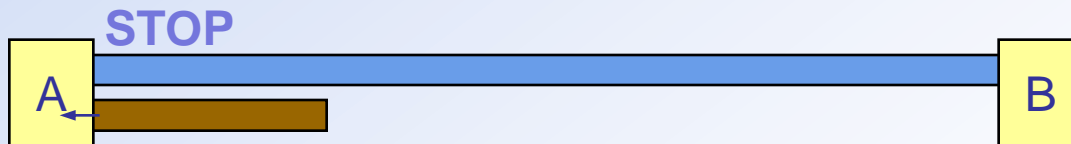


$t = \tau - \delta / 2$
发生碰撞



$t = \tau$
B 检测到发生碰撞
停止发送

$t = 2\tau - \delta$
A 检测到发生碰撞



重要特性

- ❖ 使用 CSMA/CD 协议的以太网不能进行全双工通信而只能进行双向交替通信（半双工通信）。
- ❖ 每个站在发送数据之后的一小段时间内，存在着遭遇碰撞的可能性。
- ❖ 这种发送的不确定性使整个以太网的平均通信量远小于以太网的最高数据率。

争用期

- ❖ 最先发送数据帧的站，在发送数据帧后至多经过时间 2τ （两倍的端到端往返时延）就可知道发送的数据帧是否遭受了碰撞。
- ❖ 以太网的端到端往返时延 2τ 称为争用期，或碰撞窗口。
- ❖ 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。

二进制指数类型退避算法

(truncated binary exponential type)

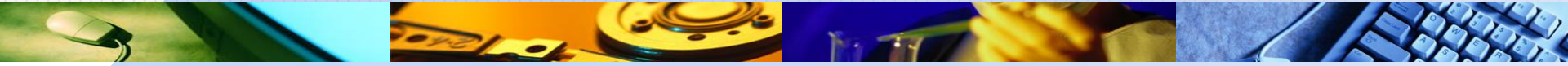
❖ 发生碰撞的站在停止发送数据后，要推迟（退避）一个随机时间才能再发送数据。

- 基本退避时间取为争用期 2τ 。
- 从整数集合 $[0, 1, \dots, (2^k - 1)]$ 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。
- 参数 k 按下面的公式计算：

$$k = \text{Min}[\text{重传次数}, 10]$$

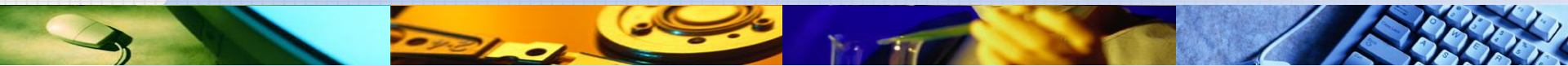
- 当 $k \leq 10$ 时，参数 k 等于重传次数。
- 当重传达 16 次仍不能成功时即丢弃该帧，并向高层报告。

争用期的长度



- ❖ 以太网取 $51.2 \mu\text{s}$ 为争用期的长度。
- ❖ 对于 10 Mb/s 以太网，在争用期内可发送 512 bit ，即 64 字节。
- ❖ 以太网在发送数据时，若前 64 字节没有发生冲突，则后续的数据就不会发生冲突。

最短有效帧长



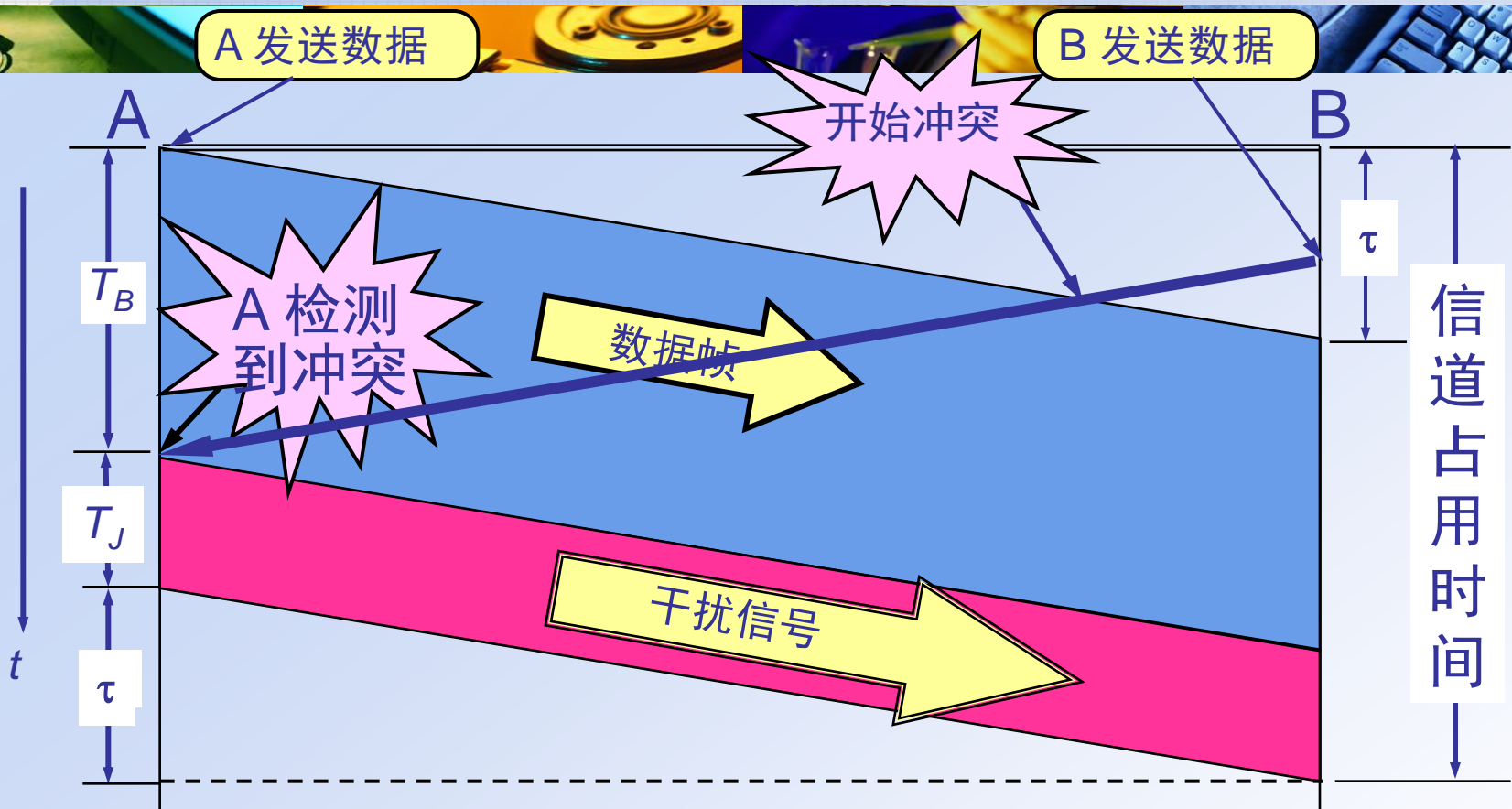
- ❖ 如果发生冲突，就一定是在发送的前 64 字节之内。
- ❖ 由于一检测到冲突就立即中止发送，这时已经发送出去的数据一定小于 64 字节。
- ❖ 以太网规定了最短有效帧长为 64 字节，凡长度小于 64 字节的帧都是由于冲突而异常中止的无效帧。

强化碰撞



- ❖ 当发送数据的站一旦发现发生了碰撞时：
 - 立即停止发送数据；
 - 再继续发送若干比特的人为**干扰信号**(jamming signal), 以便让所有用户都知道现在已经发生了碰撞。

人为干扰信号



B 也能够检测到冲突，并立即停止发送数据帧，接着就发送干扰信号。这里为了简单起见，只画出 A 发送干扰信号的情况。