



《计算机网络》

第5章 数据链路层与局域网

刘亚维



第5章 数据链路层与局域网



本章学习目标

- ❖ 理解数据链路层主要服务
- ❖ 理解链路层寻址及其特点
- ❖ 掌握差错编码原理及典型编码
- ❖ 掌握可靠数据传输基本原理及协议、信道利用率的计算
- ❖ 掌握多路访问控制协议特点
- ❖ 掌握以太网技术、交换机工作原理、VLAN技术等
- ❖ 理解PPP协议
- ❖ 了解链路虚拟化概念及原理

主要内容

- ❖ 5.1 数据链路层服务
- ❖ 5.2 差错检测与纠正
- ❖ 5.3 多路访问控制协议
- ❖ 5.4 ARP
- ❖ 5.5 局域网技术
- ❖ 5.6 PPP协议
- ❖ 5.7 链路虚拟化



5.1 数据链路层服务

刘亚维



数据链路层

5.1 数据链路层基本服务

- ❖ 向下：利用物理层提供的位流服务
- ❖ 向上：向网络层提供明确的 (**well-defined**) 服务接口



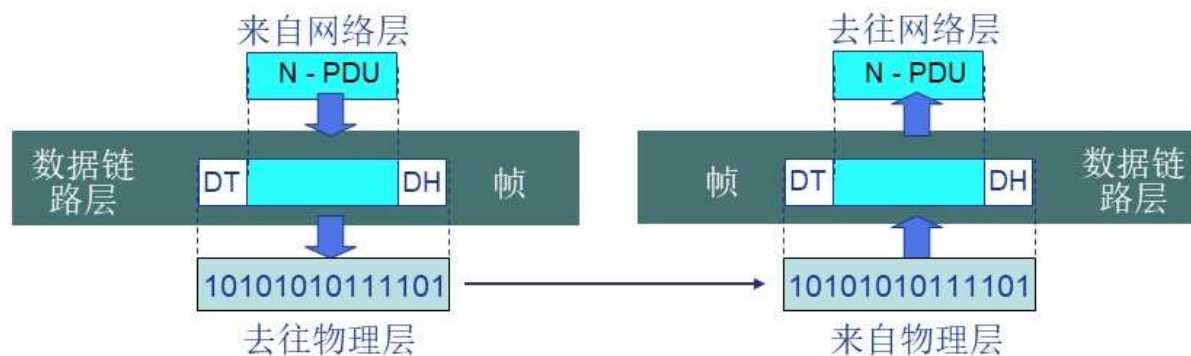
5层参考模型





数据链路层功能

5.1 数据链路层基本服务



- ❖ 负责结点-结点 (**node-to-node**) 数据传输
- ❖ 组帧 (**Framing**)
- ❖ 物理寻址 (**Physical addressing**)
- ❖ 流量控制 (**Flow control**)
- ❖ 差错控制 (**Error control**)
- ❖ 访问控制 (**Access control**)

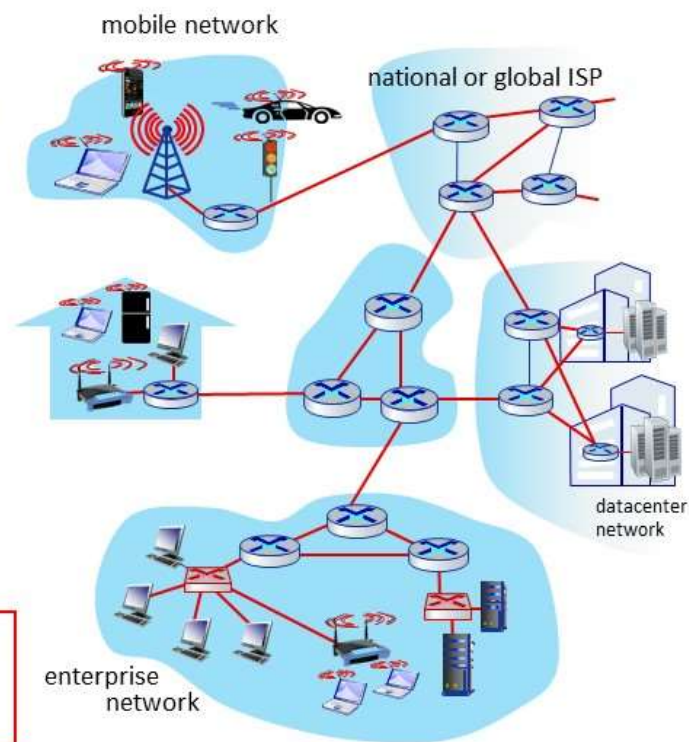




5.1 数据链路层基本服务

术语

- ❖ **结点(nodes)**：主机、路由器、交换机
- ❖ **链路(links)**：连接相邻结点的通信信道
 - 有线链路(wired links)
 - 无线链路(wireless links)
 - 局域网(LANs)
- ❖ **帧(frame)**：链路层(第2层)数据分组，封装网络层数据报



数据链路层负责通过一条链路从一个结点向另一个物理链路直接相连的相邻结点传送数据报。





链路层与网络层

5.1 数据链路层基本服务

- **数据报传输**通常需要经过若干不同的链路，这些链路上运行各自的链路协议：
 - 例如，第一个链路为WiFi，下一个链路是以太网
- 每个链路协议可提供**不同的服务**
 - 例如：可靠的数据传输，不是所有的链路协议都提供。

类比交通：

- 从普林斯顿到洛桑的旅行
 - 豪华轿车：普林斯顿到肯尼迪机场
 - 飞机：肯尼迪机场飞往日内瓦
 - 火车：日内瓦到洛桑
- 旅游者 = **数据报**
- 每段交通 = **通信链路**
- 交通方式 = **链路层协议**
- 旅行社 = **路由算法**

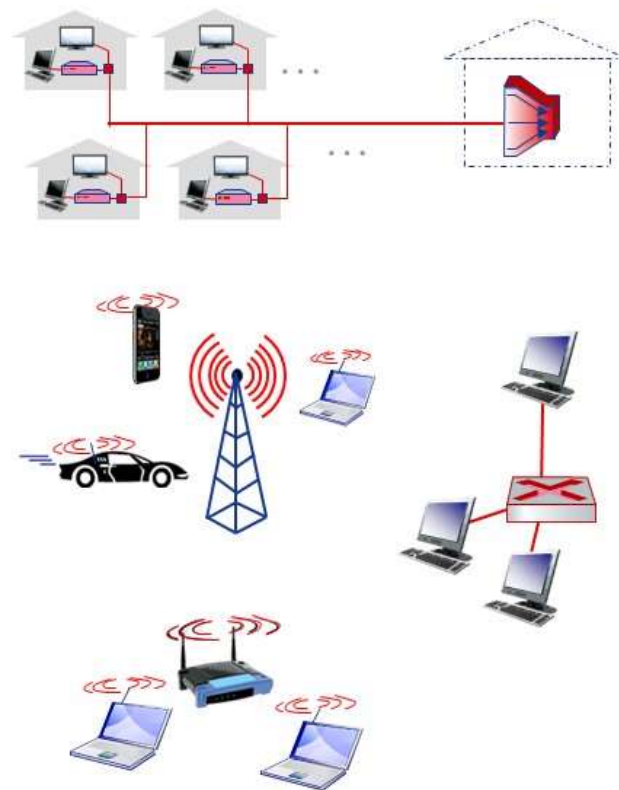




链路层服务

5.1 数据链路层基本服务

- **相邻节点之间的可靠交付**
 - 相关理论传输层已经学习!
 - **低误码率**链路很少使用
 - **无线链路**会使用：因其错误率高
 - **Q:**为什么同时具有链路级和端到端可靠性?
- **组帧, 链路访问 (Access) :**
 - 将数据报封装到帧中, 添加帧头, 帧尾
 - 如果是共享介质, 则进行信道访问 (控制)
 - 帧头中的“MAC”地址可标识源、目标 (与 IP 地址不同!)



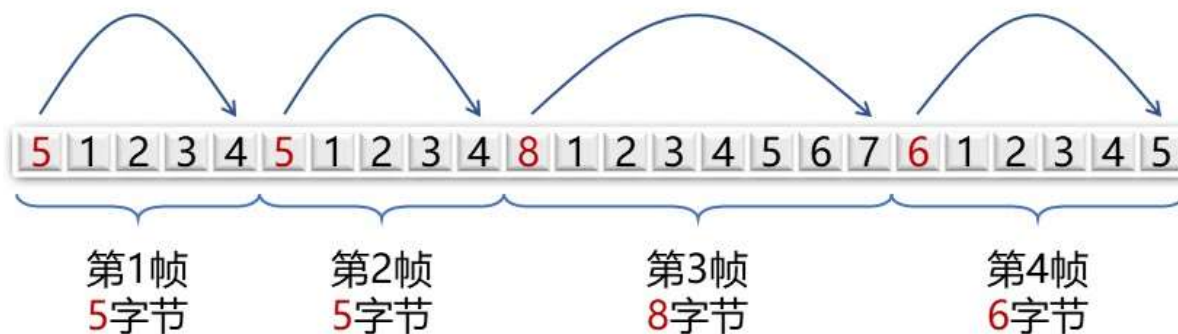


5.1 数据链路层基本服务

链路层服务

❖ 组帧(framing)

- 封装网络层数据报构成数据帧
- 帧同步：从物理层接收的比特流中准确区分出一帧的开始和结束
 - 字节计数法 (Byte count)



问题：如果某个计数字节出错会发生什么情况？





5.1 数据链路层基本服务

链路层服务

❖ 组帧(framing)

- 封装网络层数据报构成数据帧
- 帧同步：从物理层接收的比特流中准确区分出一帧的开始和结束
 - 字节计数法 (Byte count)
 - 带字节填充的定界符法 (Flag bytes with byte stuffing)
 - 定界符 (FLAG)：一个特殊的字节，比如 01111110，即 0x7E，用于区分前后两个不同的帧



问题：如果有效载荷部分包含与“定界符”相同的字节会有什么问题？





5.1 数据链路层基本服务

链路层服务

❖ 组帧(framing)

- 封装网络层数据报构成数据帧
- 帧同步：从物理层接收的比特流中准确区分出一帧的开始和结束
 - 字节计数法 (Byte count)
 - 带字节填充的定界符法 (Flag bytes with byte stuffing)





5.1 数据链路层基本服务

链路层服务

❖ 组帧(framing)

- 封装网络层数据报构成数据帧
- 帧同步：从物理层接收的比特流中准确区分出一帧的开始和结束
 - 字节计数法 (Byte count)
 - 带字节填充的定界符法 (Flag bytes with byte stuffing)
 - 带比特填充的定界符法 (Flag bits with bit stuffing)
 - 定界符：如两个0比特之间，连续6个1比特，即01111110，0x7E

0 1 0 1 1 0 1 1 1 1 1 1 0 0 1 1 1 0 0 0 1 0 1 1

上一帧

定界符

下一帧

问题：如果有效载荷部分包含与“定界符”相同的位组合如何解决？





5.1 数据链路层基本服务

链路层服务

❖ 组帧(framing)

- 封装网络层数据报构成数据帧
- 帧同步：从物理层接收的比特流中准确区分出一帧的开始和结束
 - 字节计数法 (Byte count)
 - 带字节填充的定界符法 (Flag bytes with byte stuffing)
 - 带比特填充的定界符法 (Flag bits with bit stuffing)
 - 定界符：如两个0比特之间，连续6个1比特，即01111110，0x7E
 - 若在有效载荷中出现连续5个1比特，则直接插入1个0比特

原始载荷

0 1 1 1 1 1 0 1 1 1 1 1 1 0 0 0 1 1 1 0 0 0 1 0

载荷填充

0 1 1 1 1 1 0 0 1 1 1 1 1 0 1 0 0 0 1 1 1 0 0 0 1 0





5.1 数据链路层基本服务

链路层服务

❖ 组帧(framing)

- 封装网络层数据报构成数据帧
- 帧同步：从物理层接收的比特流中准确区分出一帧的开始和结束
 - 字节计数法 (Byte count)
 - 带字节填充的定界符法 (Flag bytes with byte stuffing)
 - 带比特填充的定界符法 (Flag bits with bit stuffing)
 - 物理层编码违例 (Physical layer coding violations)
 - 核心思想：定界符为不会在数据部分出现码字，例如：
 - 4B/5B编码方案：未使用的码字
 - 前导码：存在很长的前导码 (preamble)
 - » 例如：传统以太网、802.11
 - 曼切斯特编码 / 差分曼切斯特编码：比特周期中间信号无跳变
 - » 例如：802.5令牌环网





5.1 数据链路层基本服务

链路层服务

❖ 差错检测(error detection)

- 信号衰减和噪声会引起差错
- 差错编码

❖ 差错纠正(error correction)

- 前向纠错：接收端直接纠正比特差错-纠错码
- 通知发送端重传

❖ 链路接入(link access)

- 若物理信道是共享介质，需要解决信道接入(channel access)
- 帧首部中的“MAC”地址，用于标识帧的源和目的

❖ 相邻结点间可靠交付

- 在低误码率的有线链路上很少采用(如光纤，某些双绞线等)
- 无线链路：误码率高，需要可靠数据传输协议
 - 停-等协议
 - 滑动窗口协议





链路层服务

5.1 数据链路层基本服务

❖ 流量控制(flow control)

- 协调(pacing)相邻的发送结点和接收结点
- 避免接收端被数据“淹没”
- 解决方案:
 - 基于**反馈** (feedback-based) 的流量控制
 - 接收方反馈, 发送方调整发送速率
 - 基于**速率** (rate-based) 的流量控制
 - 发送方根据内建机制, 自行限速

❖ 全双工和半双工通信控制

- 全双工: 链路两端结点**同时**双向传输
- 半双工: 链路两端结点**交替**双向传输

❖ 服务类型

- 无连接服务
- 面向连接服务

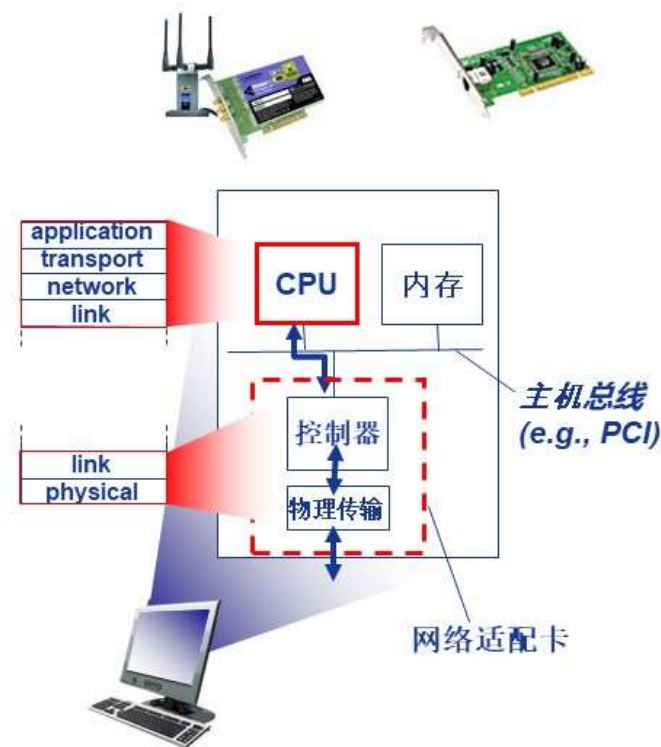




5.1 数据链路层基本服务

链路层的具体实现？

- ❖ 每个主机或路由器接口
- ❖ 链路层在“适配器”
(即网络接口卡-NIC)中实现或者在一个芯片上实现
 - 以太网网卡、802.11网卡；以太网芯片组
 - 实现链路层和物理层
- ❖ 连接主机的系统总线
- ❖ 由硬件、软件与固件组成



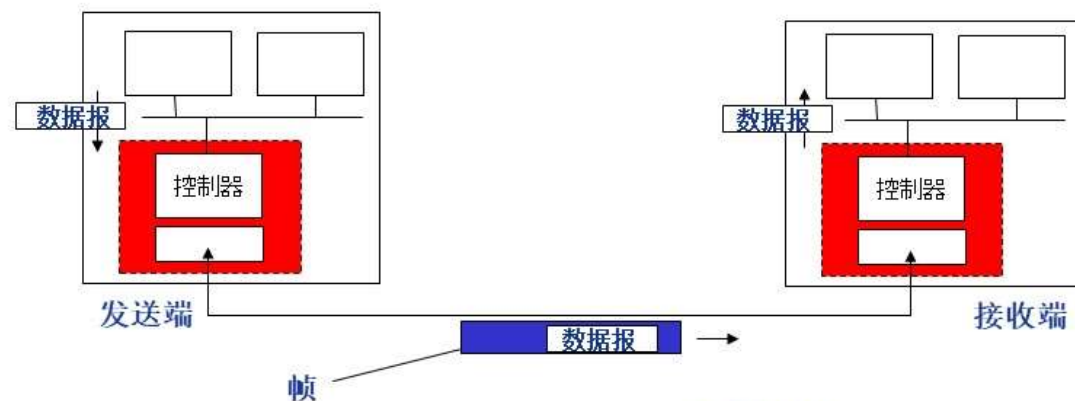


网卡间通信

5.1 数据链路层基本服务

❖ 发送端:

- 将数据报封装成帧
- 增加差错检测比特, 实现可靠数据传输和流量控制等.



❖ 接收端:

- 检测差错, 实现可靠数据传输和流量控制等
- 提取数据报, 交付上层协议实体





5.2 差错检测与纠正

刘亚维

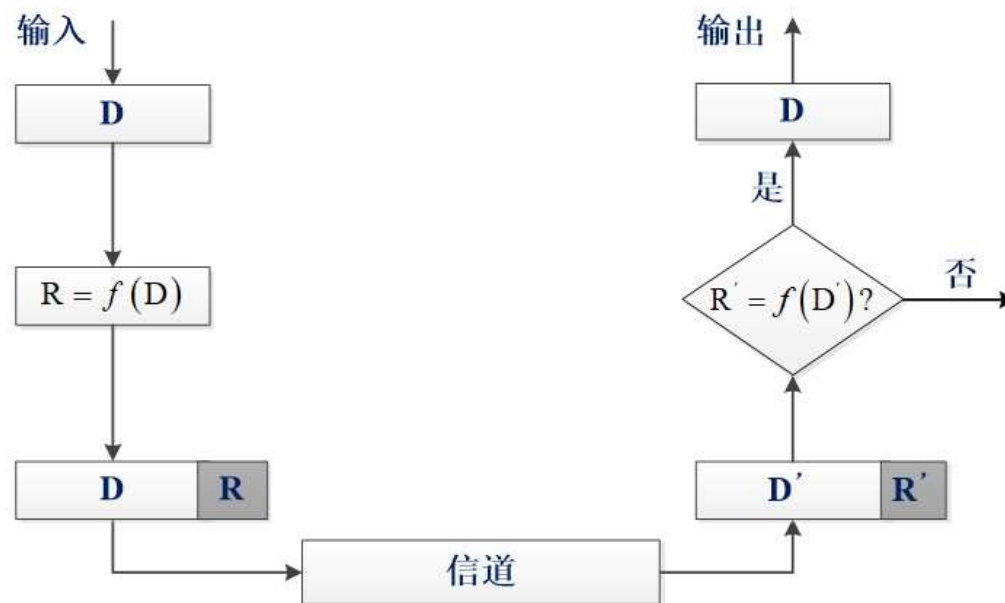
24



差错检测：差错编码

差错编码基本原理：

$D \rightarrow DR$ ，其中 R 为差错检测与纠正比特（冗余比特）



差错编码不能保证**100%**可靠！

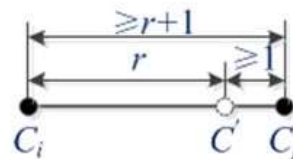


差错编码的检错能力

❖ 差错编码可分为**检错码**与**纠错码**

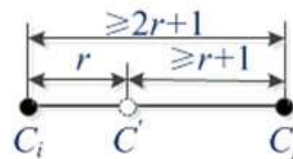
❖ 汉明距离 (Hamming distance): 两个码字之间对应位不同的数目

❖ 对于检错码, 如果编码集的汉明距离 $d_s = r + 1$, 则该差错编码可以检测 r 位差错



- 例如, 编码集 {0000, 0101, 1010, 1111} 的汉明距离 $d_s = 2$, 可以100%检测1比特差错

❖ 对于纠错码, 如果编码集的汉明距离 $d_s = 2r + 1$, 则该差错编码可以纠正 r 位差错



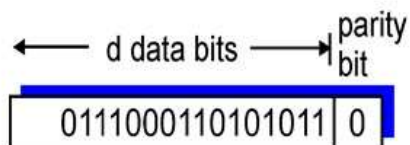
- 例如, 编码集 {000000, 010101, 101010, 111111} 的汉明距离 $d_s = 3$, 可以纠正1比特差错, 如100010纠正为101010。



奇偶校验码

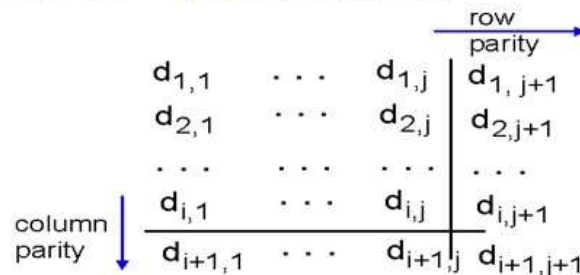
1比特校验位:

- ❖ 检测奇数位差错



二维奇偶校验:

- ❖ 检测奇数位差错、部分偶数位差错
- ❖ 纠正同一行/列的奇数位错



1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

no errors

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

parity error

correctable single bit error



Internet校验和(Checksum)

发送端:

- ❖ 将“数据”(校验内容)划分为16位的二进制“整数”序列
- ❖ 求和(sum): 补码求和(最高位进位的“1”, 返回最低位继续加)
- ❖ 校验和(Checksum): sum的反码
- ❖ 放入分组(UDP、TCP、IP)的校验和字段

接收端:

- ❖ 与发送端相同算法计算
- ❖ 计算得到的“checksum”:
 - 为16位全0(或sum为16位全1): 无错
 - 否则: 有错



循环冗余校验码(CRC)

- ❖ 检错能力更强大的差错编码
- ❖ 将数据比特, **D**, 视为一个二进制数
- ❖ 选择一个**r+1**位的比特模式 (生成比特模式), **G**
- ❖ 目标: 选择**r**位的CRC比特, **R**, 满足
 - $\langle D, R \rangle$ 刚好可以被G整除(模2)
 - 接收端检错: 利用G除 $\langle D, R \rangle$, 余式全0, 无错; 否则, 有错!
 - 可以检测所有突发长度小于r+1位差错。
- ❖ 广泛应用于实际网络 (以太网, 802.11 WiFi, ATM)

← d bits → ← r bits →



bit
pattern

$$D * 2^r \text{ XOR } R$$

mathematical
formula

单选题 1分

5.1 数据链路层基本服务

5.2 差错检测与纠正

若数据传输时采用<D,EDC>差错编码，其中D为数据， $EDC=DD$ （即复制两份数据），则该差错编码可以

- A 检测3个比特差错，纠正3个比特差错
- B 检测3个比特差错，纠正2个比特差错
- C 检测2个比特差错，纠正2个比特差错
- D 检测2个比特差错，纠正1个比特差错**

