



计算机通信与网络

Computer Communications & Networks

第3章 数据链路层

南京邮电大学通信与信息工程学院

“计算机通信与网络” 国家精品课程组

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

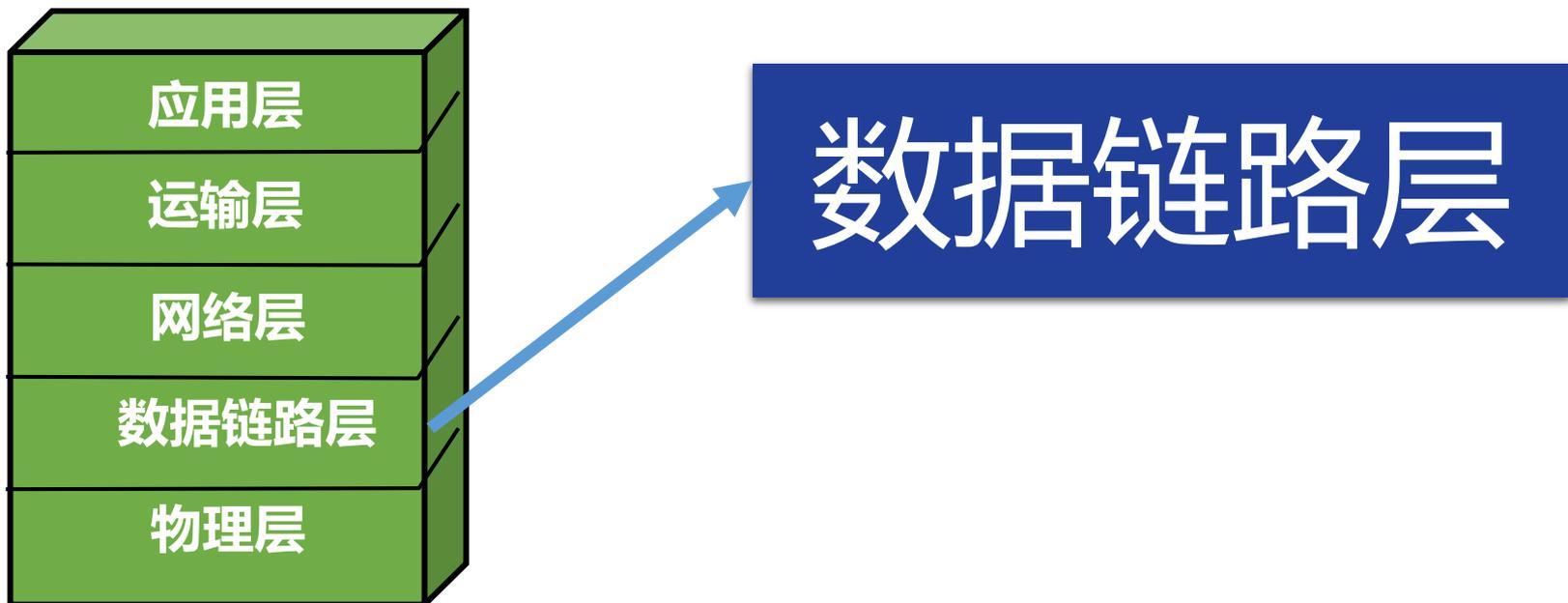
02 流量控制和差错控制

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



5层的体系结构





基本概念

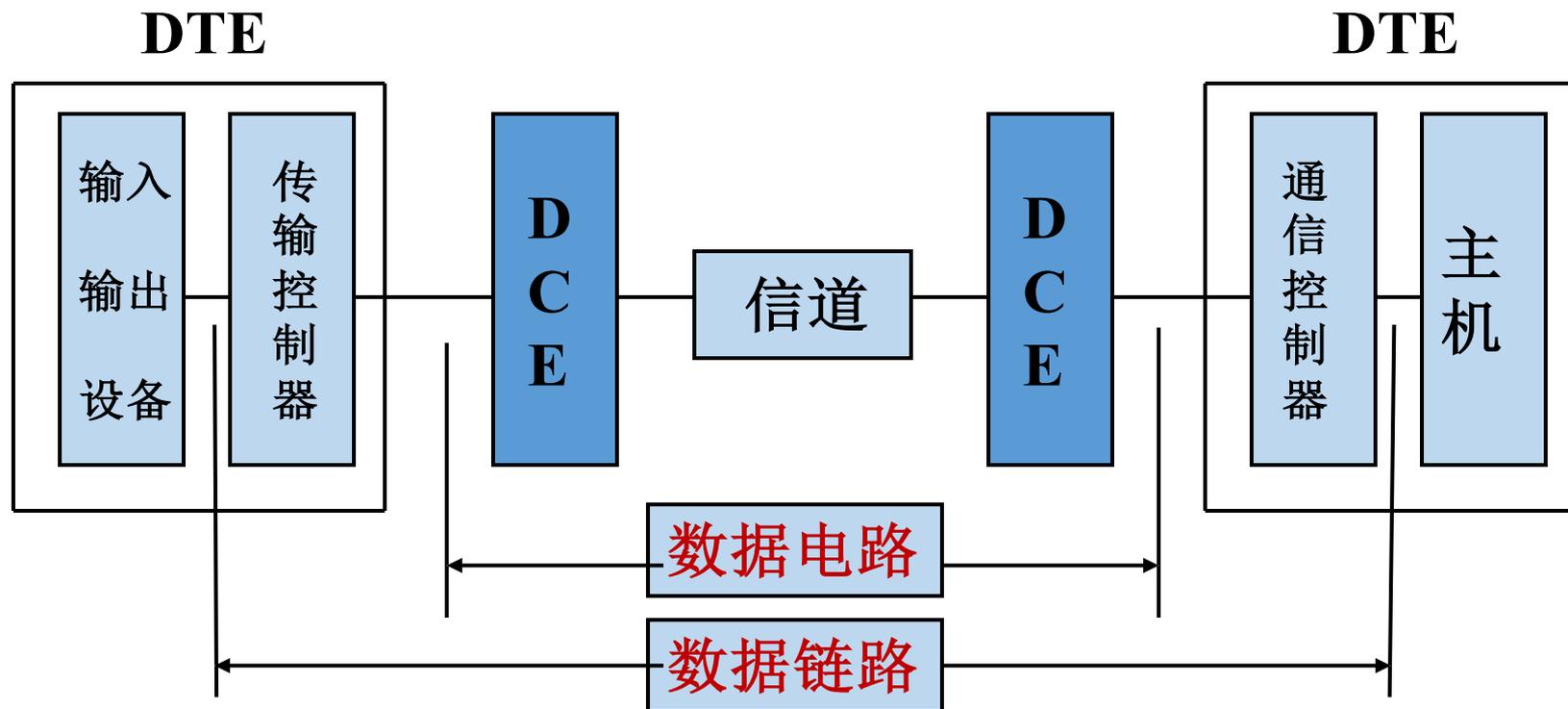
3.1.1 数据电路和数据链路

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。

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。

基本概念

3.1.1 数据电路和数据链路



DTE: 数据终端设备, 输入输出, 通信控制。

DCE: 数据电路设备, 传输信号的变换, 信号控制。



基本概念

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物。



基本概念

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**

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。

基本概念

3.1.2 数据链路的结构

- 链路上所连接的节点称为“**站**”。
- 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，能够发出命令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）

点到点链路（平衡型，不平衡型）

点到多点链路（平衡型，不平衡型）

数据链路层的传送方式：

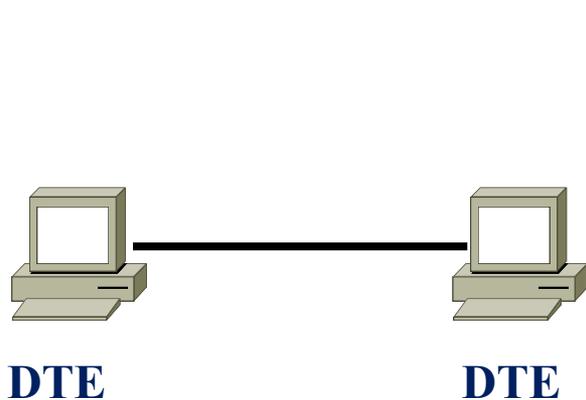
单工通信

半双工通信(Half Duplex)

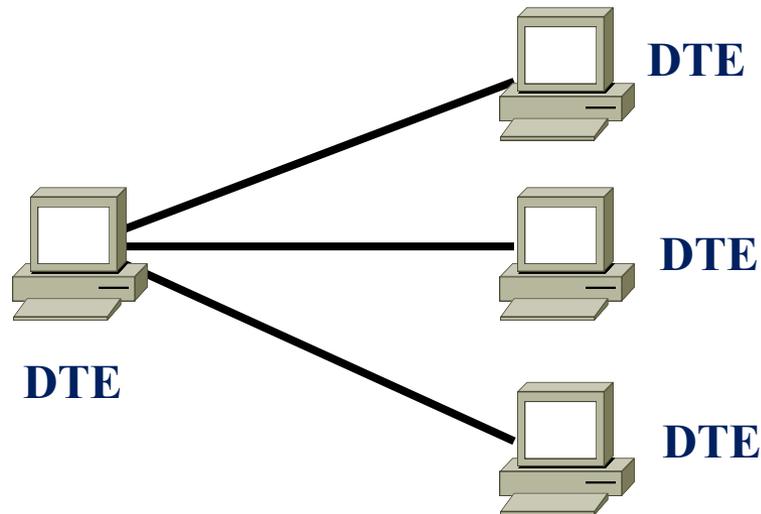
全双工通信(Full Duplex)

基本概念

3.1.2 数据链路的结构



点到点链路

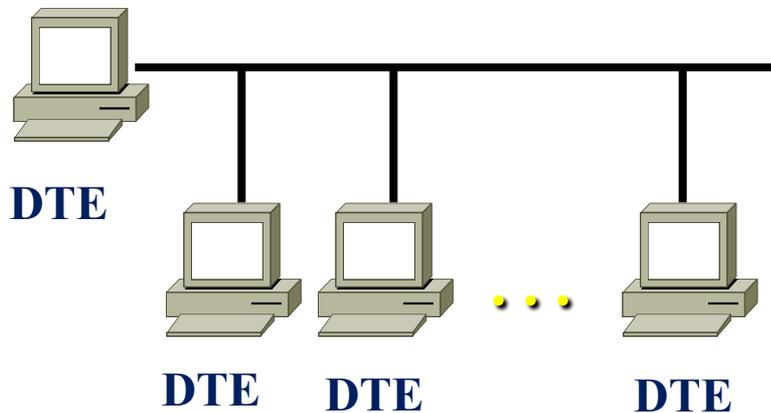


星型点到点链路

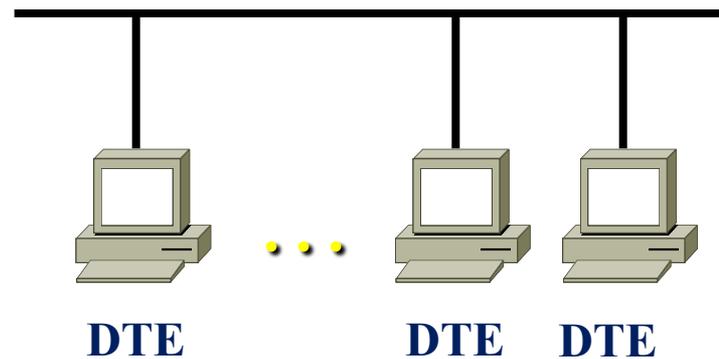
在点到点链路中，两端的站可能是主站、从站或复合站。链路可以是不平衡结构，或平衡结构。

基本概念

3.1.2 数据链路的结构



主从式点到多点链路



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。

基本概念

3.1.3 数据链路层的功能

- **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。**
- **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

基本概念

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。
- 为了便于实现流量控制和差错控制，数据链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 (protocol data unit)，即传输和处理的数据单位：**帧**。

基本概念

3.1.4 协议数据单元

- **数据链路层帧的结构，包括需要传输的数据、相应的控制信息、校验信息、帧之间的分隔标志等。必须有相应的方法表明各个部分的信息。**
- **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



基本概念

3.1.4 协议数据单元

- **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）。**
- **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



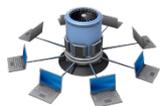
流量控制的基本概念



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



流量控制

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。

流量控制

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速率。
- **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。



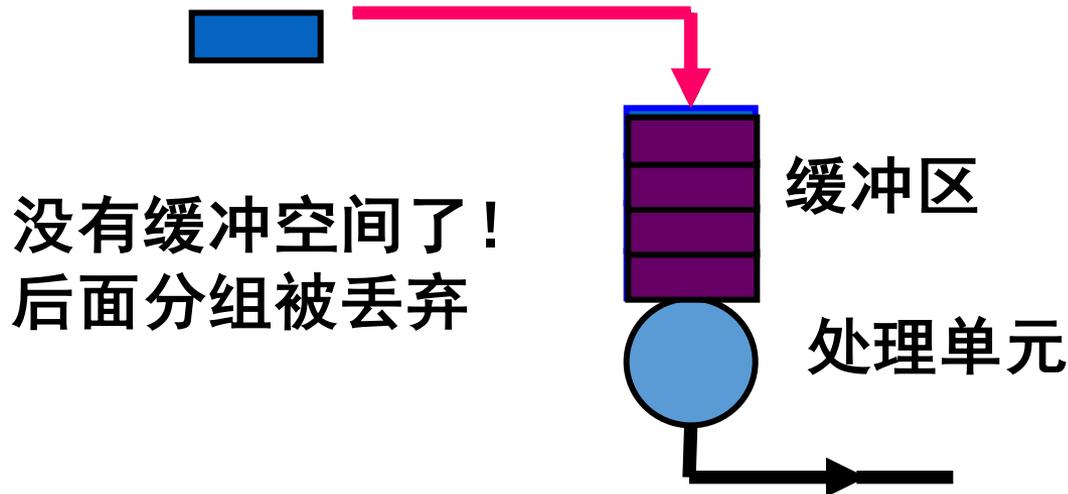
- **Logical Link Control (LLC) 逻辑链路子层**
 - **负责流量控制 (简单的流量控制)**
 - **如缓冲和排队, 防止乱序**
 - **负责差错控制**
 - **解析站 (节点) 地址**

流量控制

3.2.1 流量控制的基本概念

流量控制的目的：

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。





流量控制

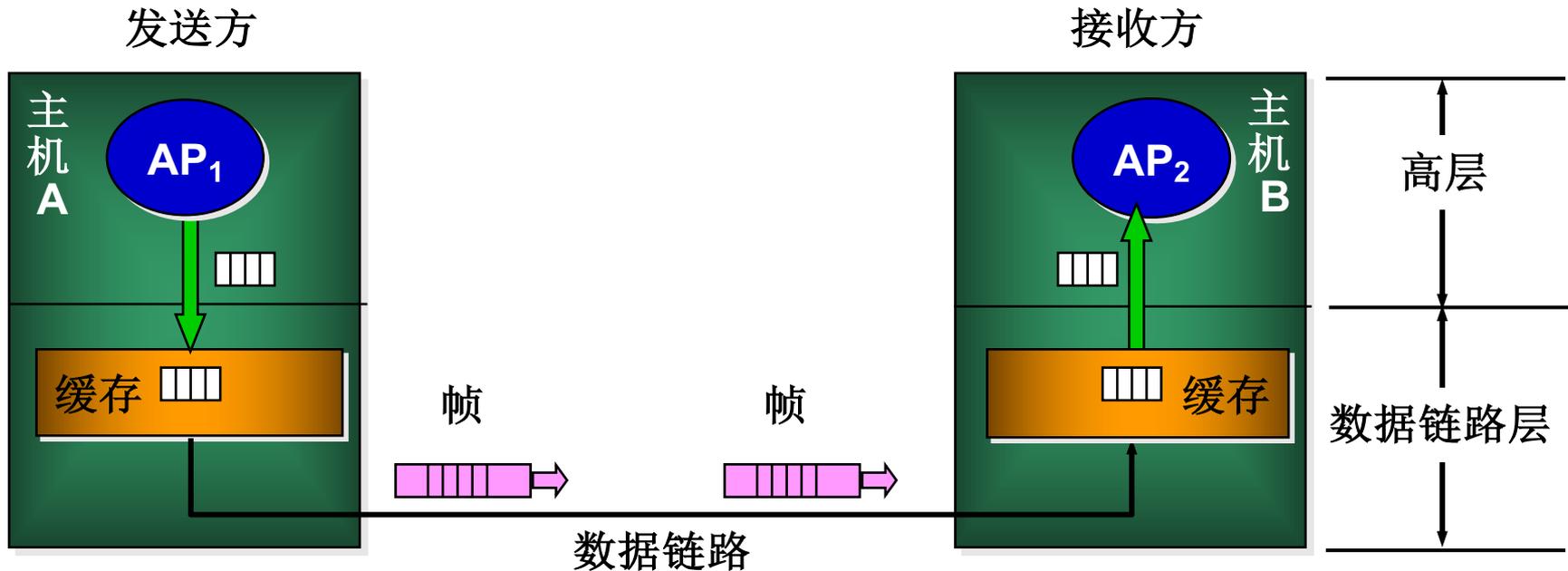
3.2.1 流量控制的基本概念

- 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。

流量控制

3.2.1 流量控制的基本概念

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

3.2.1 流量控制的基本概念

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，接收方总是来得及收下，并及时上交主机。即接收方总是处于接收准备好状态。

流量控制

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（传输层负责）。
- 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。

过渡页

Transition Page



差错控制技术



差错控制原理



差错控制的方式



差错控制编码



- 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。
- 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性与传输效率。



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



差错产生的原因

3.6.1 差错控制原理

- 差错控制
 - 发现差错
 - 如何处理差错
 - 纠错
 - 重传
 - . . .



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回发送端，并与原发送信码相比较。

3.6.3 差错控制编码

- 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 $2^k < 2^n$
- 在 n 位二进制编码的 2^n 种组合中，能表示信息的 2^k 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
- **汉明距离 (d_0)** : 一个有效编码集中, 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

3.6.3 差错控制编码

- 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 $2t + 1$;
- 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

3.6.3 差错控制编码

例1 数据 编码

0 0 00 000

1 1 11 111

汉明距离 $d=1$ $d=2$ $d=3$

例2 000000

000111

111000

111111

收到010111

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)



循环冗余码

3.6.3 差错控制编码

- 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**。

循环冗余码

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 $A = [a_{n-1}, a_{n-2}, \dots, a_1, a_0]$ ，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：

$$\begin{aligned} A_4(x) &= 0 \cdot x^6 + 1 \cdot x^5 + 1 \cdot x^4 + 1 \cdot x^3 + 0 \cdot x^2 + 0 \cdot x + 1 \\ &= x^5 + x^4 + x^3 + 1 \end{aligned}$$



循环冗余码

3.6.3 差错控制编码

生成多项式 $g(x)$

$$n = k + r$$

在 (n, k) 循环码中，存在**惟一**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 $g(x)$ ，最高位和最低位必须得是1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 $x^n + 1$ 的一个因式。 $g(x)$ 称为该编码集的**生成多项式**。

循环冗余码

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

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



循环冗余码

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^r （在最后添加 r 个0）；
- 2) 然后被 $G(x)$ 除，求其余式；
- 3) 减去（或加）余数（模2运算），得到 $x^r M(x)$

模2运算（异或运算）不借位，无论加减



循环冗余码

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 $R(x)$ ，如能整除，则表明传输无差错。



循环冗余码

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 $g(x) = x^4 + x + 1$ ，试说明：

- 1) 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）。



循环冗余码

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 2^4 ，
然后被生成多项式模2除。



循环冗余码

3.6.3 差错控制编码

$$\begin{array}{r}
 1100001010 \\
 10011 \overline{) 11010110110000} \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 10110 \\
 \underline{10011} \\
 10100 \\
 \underline{10011} \\
 1110
 \end{array}$$

生成发送序列

$$\begin{array}{r}
 11010110110000 \\
 + 1110 \\
 \hline
 11010110111110
 \end{array}$$

(a) 编码

$$\begin{array}{r}
 1100001010 \\
 10011 \overline{) 11010110111110} \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 10111 \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 00000 \\
 \underline{00000} \\
 0000
 \end{array}$$

无差错

$$\begin{array}{r}
 1100001010 \\
 10011 \overline{) 11010110110110} \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 10110 \\
 \underline{10011} \\
 10111 \\
 \underline{10011} \\
 01000 \\
 \underline{00000} \\
 1000
 \end{array}$$

有差错

(b) 译码

循环冗余码

3.6.3 差错控制编码

- 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。

循环冗余码

3.6.3 差错控制编码

- 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。

循环冗余码

3.6.3 差错控制编码

- 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。

Thank You

Have A Nice Day

南京邮电大学通信与信息工程学院

“**计算机通信与网络**” 国家精品课程组
