

计算机网络

李峰 主讲
计算机学院

第4章

数据链路层

本章学习要求

- ◆ 了解：数据传输过程中差错产生的原因与性质
- ◆ 掌握：误码率的定义与差错控制方法
- ◆ 掌握：数据链路层的基本概念
- ◆ 了解：面向字符型数据链路层协议实例—BSC
- ◆ 掌握：面向比特型数据链路层协议实例— HDLC
- ◆ 掌握：Internet中的数据链路层协议

4.1 差错产生与差错控制方法

- ◆为什么要设计数据链路层
- ◆差错产生的原因和差错类型
- ◆误码率的定义
- ◆检错码和纠错码
- ◆循环冗余编码工作原理
- ◆差错控制机制

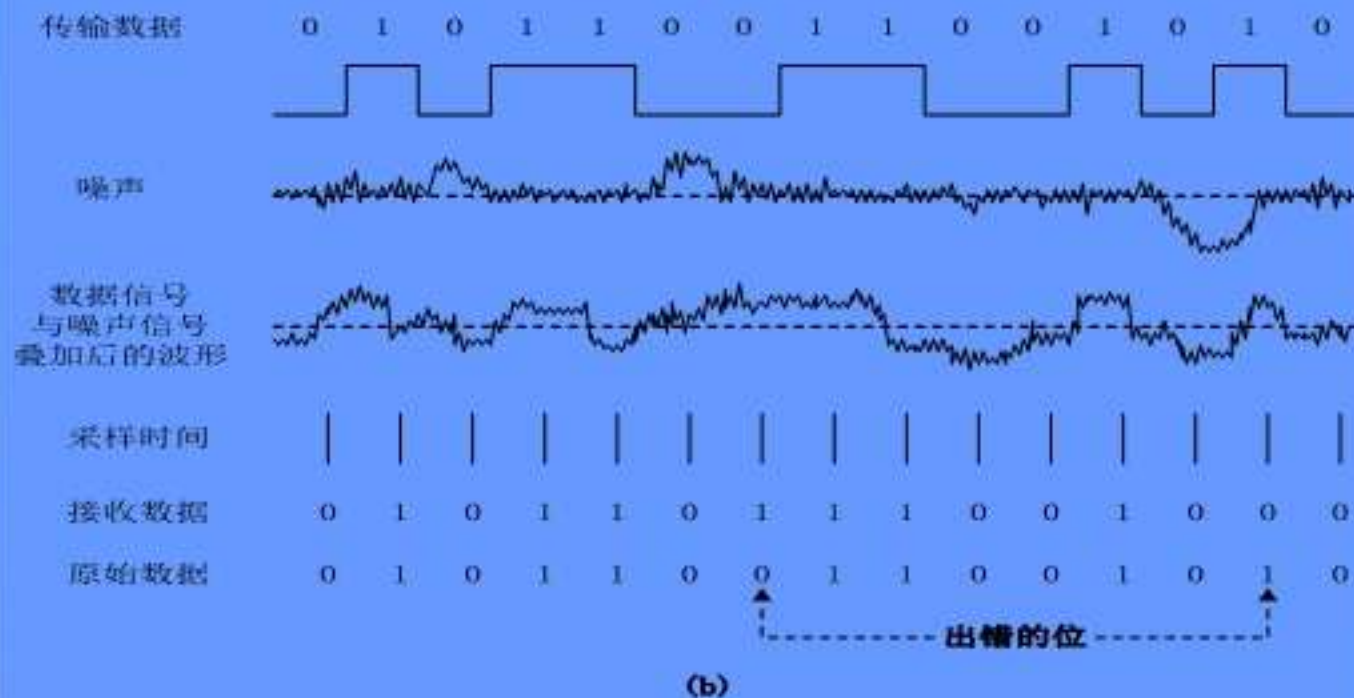
为什么要设计数据链路层

- ◆在原始物理传输线路上传输数据信号是有差错的；
- ◆设计数据链路层的主要目的：
 - 将有差错的物理线路改进成无差错的数据链路；
- ◆方法：差错检测、差错控制、流量控制
- ◆作用：改善数据传输质量，向网络层提供高质量的服务。

差错产生的原因和差错类型

- ◆ 传输差错：通过通信信道后接收的数据与发送数据不一致的现象；
- ◆ 差错控制：检查是否出现差错以及如何纠正差错；
- ◆ 通信信道的噪声分为两类：热噪声和冲击噪声；
 - 由热噪声引起的差错是随机差错，或随机错；
 - 冲击噪声引起的差错是突发差错，或突发错；
 - 引起突发差错的位长称为突发长度；
- ◆ 在通信过程中产生的传输差错，是由随机差错与突发差错共同构成的。

传输差错产生过程



误码率的定义

- ◆ 二进制比特在数据传输系统中被传错的概率，
- ◆ 它在数值上近似等于：

$$P_e = N_e/N$$

- ◆ 其中，**N**为传输的二进制比特总数，**N_e**为被传错的比特数

误码率的定义

◆说明:

- 误码率应该是衡量数据传输系统正常工作状态下传输可靠性的参数;
- 对于一个实际的数据传输系统, 不能笼统地说误码率越低越好, 要根据实际传输要求提出误码率要求;
- 对于实际数据传输系统, 如果传输的不是二进制比特, 要折合成二进制比特来计算;
- 差错的出现具有随机性, 在实际测量一个数据传输系统时, 只有被测量的传输二进制比特数越大, 才会越接近于真正的误码率值。

检错码与纠错码

◆ 纠错码:

每个传输的分组带上足够的冗余信息;
接收端能发现并自动纠正传输差错。

◆ 检错码:

分组仅包含足以使接收端发现差错的冗余信息;
接收端能发现出错,但不能确定哪一比特是错的,
并且自己不能纠正传输差错。

检错码与纠错码

◆常用的检错码

➤奇偶校验码

垂直奇（偶）校验

水平奇（偶）校验水平

垂直奇（偶）校验（方阵码）

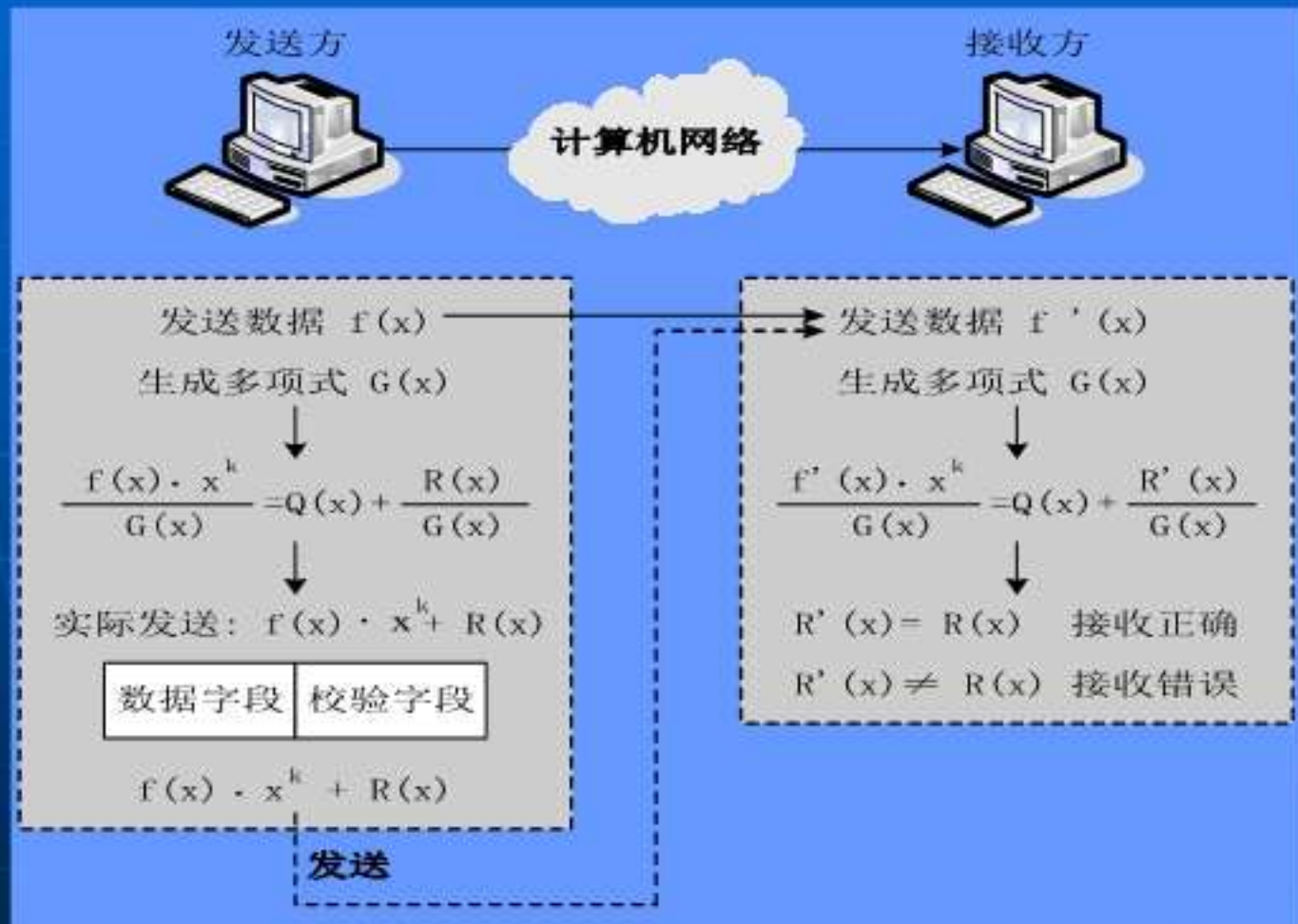
➤循环冗余编码**CRC**

目前应用最广的检错码编码方法之一

循环冗余编码工作原理

- ◆ 将要发送的数据比特序列当作一个多项式 $f(x)$ 的系数，在发送端用收发双方预先约定的生成多项式 $G(x)$ 去除（模2除），求得一个余数多项式，将余数多项式加到数据多项式后发送到接收端。
- ◆ 在接收端用同样的生成多项式 $G(x)$ 去除接收数据多项式 $f'(x)$ ，得到计算余数多项式。
- ◆ 如果计算余数多项式与接收余数多项式相同，则无出错；否则有差错。

循环冗余编码工作原理



举例：

发送端

$$\begin{array}{r}
 \begin{array}{l} G(x) \rightarrow 1 \ 1 \ 0 \ 0 \ 1 \end{array} \overline{) \begin{array}{l} 1 \ 0 \ 0 \ 0 \ 0 \end{array}} \leftarrow Q(x) \\
 \underline{1 \ 1 \ 0 \ 0 \ 1} \leftarrow f(x) \cdot x^k \\
 1 \ 0 \ 0 \ 0 \ 0 \\
 \underline{1 \ 1 \ 0 \ 0 \ 1} \\
 1 \ 0 \ 0 \ 0 \ 0 \\
 \underline{1 \ 1 \ 0 \ 0 \ 1} \\
 1 \ 0 \ 0 \ 1 \leftarrow R(x)
 \end{array}$$

$x^4 + x^3 + 1$

发送端发送的比特序列

1 1 0 0 1 1	1 0 0 1
<u>发送数据</u>	<u>CRC校验码</u>
<u>比特序列</u>	<u>比特序列</u>

带CRC校验码的
发送数据比特序列

接收端

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

循环冗余编码工作原理

◆ 标准CRC生成多项式 $G(x)$

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

➤ **CRC-16** $G(x) = x^{16} + x^{15} + x^2 + 1$

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

➤ **CRC-32** $G(x) = x^{32} + x^{26} + x^{23} + x^{22}$
 $+ x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x$
 $+ 1$

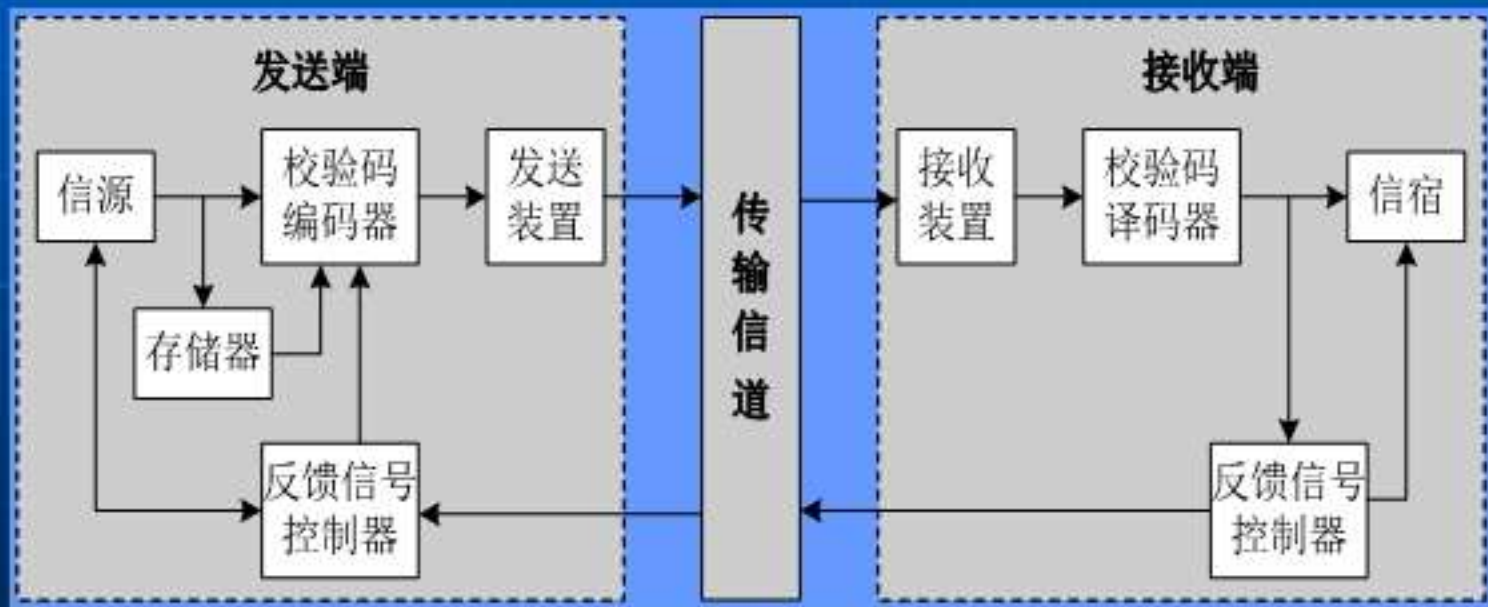
循环冗余编码工作原理

◆ CRC校验码的检错能力

- CRC校验码能检查出全部单个错;
- CRC校验码能检查出全部离散的二位错;
- CRC校验码能检查出全部奇数个错;
- CRC校验码能检查出全部长度小于或等于K位的突发错;
- CRC校验码能以 $[1 - (1/2)^{K-1}]$ 的概率检查出长度为 $(K+1)$ 位的突发错;
- 如果**K=16**, 则该CRC校验码能全部检查出小于或等于**16** 位的所有的突发差错, 并能以 $1 - (1/2)^{16-1} = 99.997\%$ 的概率检查出长度为**17**位的突发错, 漏检概率为**0.003%**;

差错控制机制

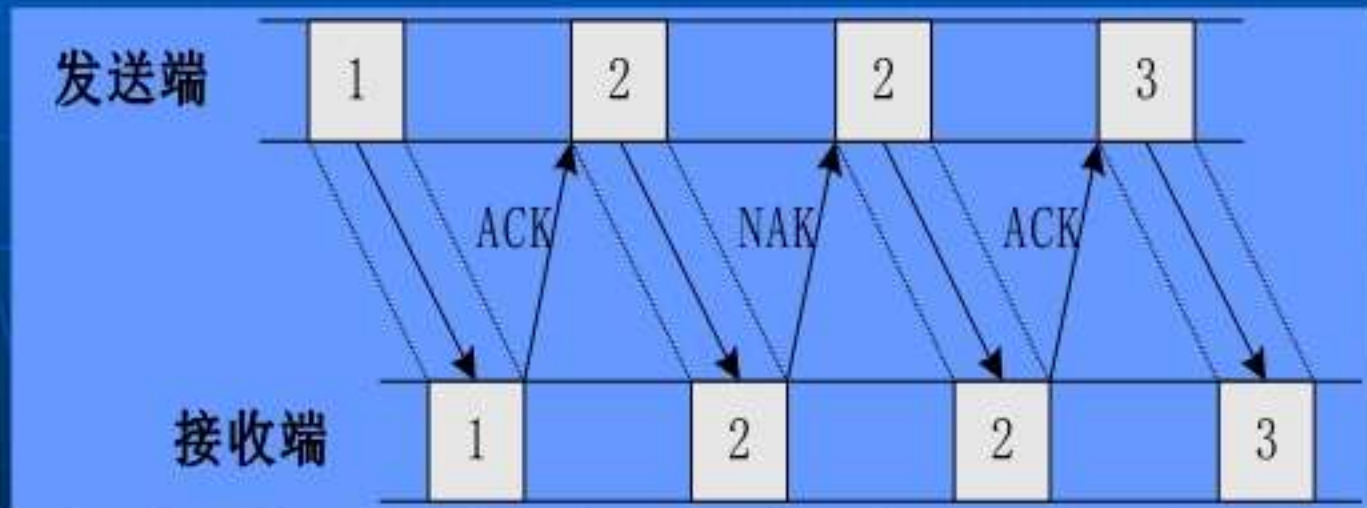
- ◆接收端通过检错码检查数据帧是否有错？一旦发现错误则采用反馈重发(**ARQ**)方法来纠正。



差错控制机制

◆ 反馈重发机制的分类

➤ 停止等待方式

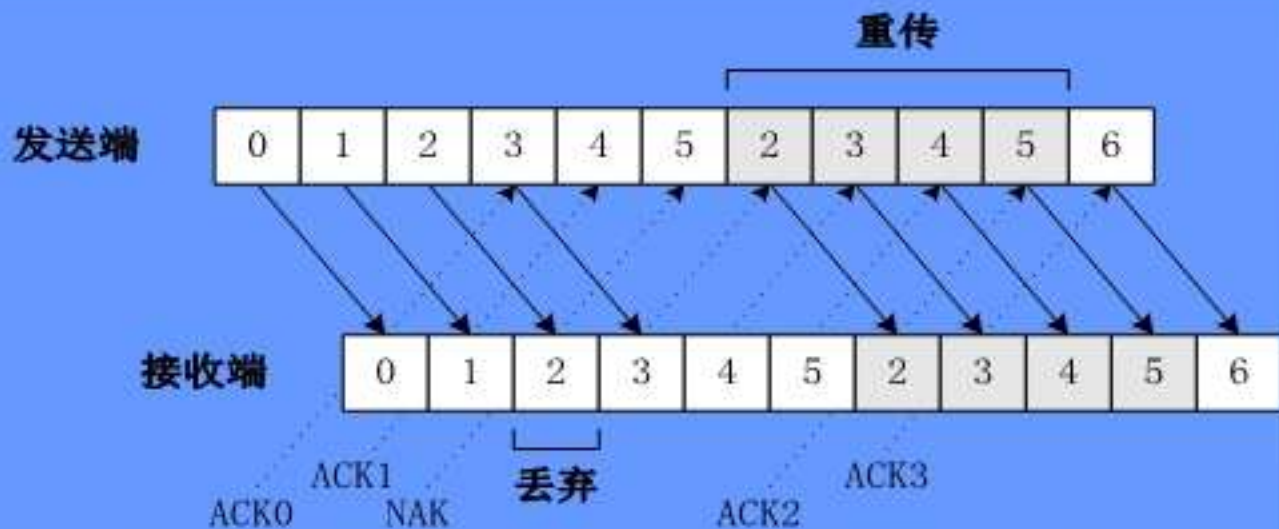


差错控制机制

◆ 反馈重发机制的分类

➢ 连续工作方式

■ 拉回方式

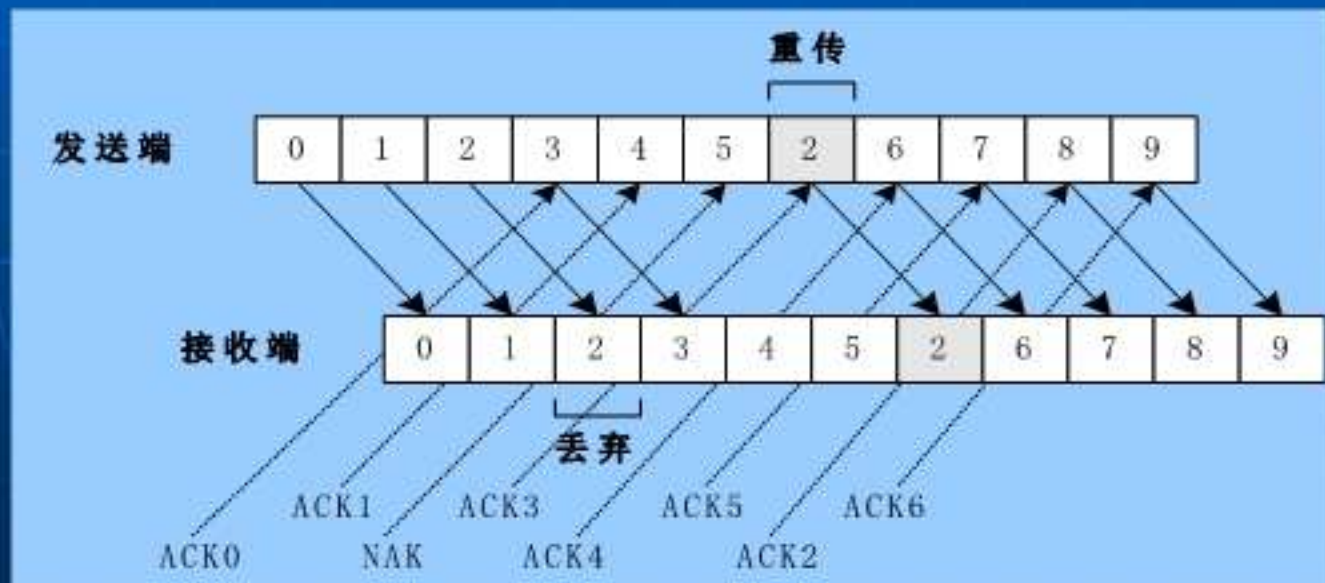


差错控制机制

◆ 反馈重发机制的分类

➤ 连续工作方式

■ 选择重发方式



4.2 数据链路层的基本概念

- ◆物理线路与数据链路
- ◆数据链路层主要功能
- ◆数据链路层向网络层提供的服务

物理线路与数据链路

- ◆ **链路**：一条点对点的线路，中间没有任何交换结点。因此，链路一般指物理线路。
- ◆ 在物理线路上要传输数据还需要相关协议来控制这些数据传输，以确保正确性，因此，实现这些协议或规程的软硬件和物理线路构成了数据链路。



数据链路层的主要功能

- ◆ 链路管理
- ◆ 帧同步
- ◆ 流量控制
- ◆ 差错控制
- ◆ 帧的透明传输
- ◆ 寻址
- ◆ 数据链路层协议 — 为实现数据链路控制功能而 制定的规程或协议。

数据链路层向网络层提供的服务

- ◆数据链路层是**OSI**参考模型的第**2**层;
- ◆设立数据链路层的主要目的是将原始的、有差错的物理线路变为对网络层无差错的数据链路;
- ◆为了实现这个目的,数据链路层必须实现链路管理、帧传输、流量控制、差错控制等功能;
- ◆数据链路层为网络层提供的服务主要表现在:
正确传输网络层的用户数据,为网络层屏蔽物理层采用的传输技术的差异性。

4.3 面向字符型协议实例

- ◆ 数据链路层协议的分类
- ◆ 面向字符型协议实例-**BSC**

数据链路层协议的分类

◆ 面向字符型

- 早期协议，以字符为控制传输信息的基本单元，即建立、维护与释放数据链路的控制有一些规定的控制字符完成。
- 缺陷：
 - 不同字符集的两台计算机很难通信；
 - 控制字符的编码不能在用户数据中出现。

◆ 面向比特型

- 为克服面向字符型协议而提出。

面向字符型协议实例-BSC

- ◆ 使用 **ASCII** 码中的 **10** 个控制字符完成通信控制，并规定了数据和控制报文的格式。

格式字符: **SOH (start of heading)**

STX (start of text)

ETB (end of transmission block)

ETX (end of text)

控制字符: **ACK (acknowledge)**

NAK (negative acknowledge)

ENQ (enquire)

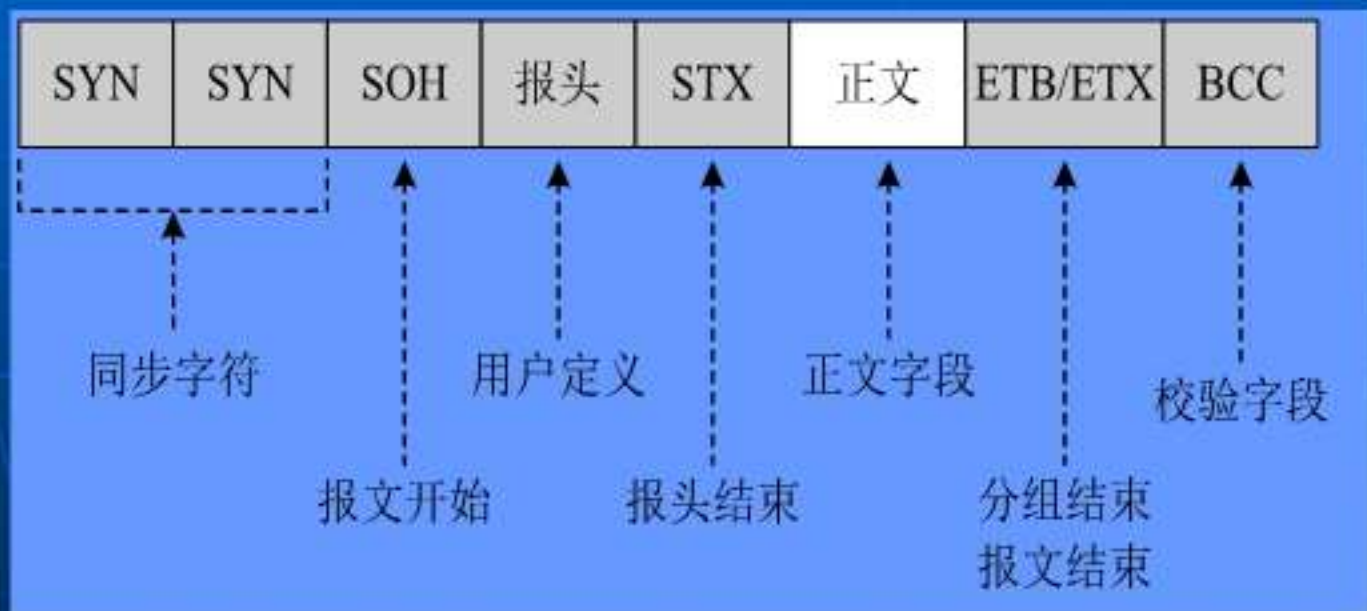
EOT (end of transmission)

SYN (synchronous)

DLE (data link escape)

面向字符型协议实例-BSC

◆面向字符型BSC协议的数据报文格式:

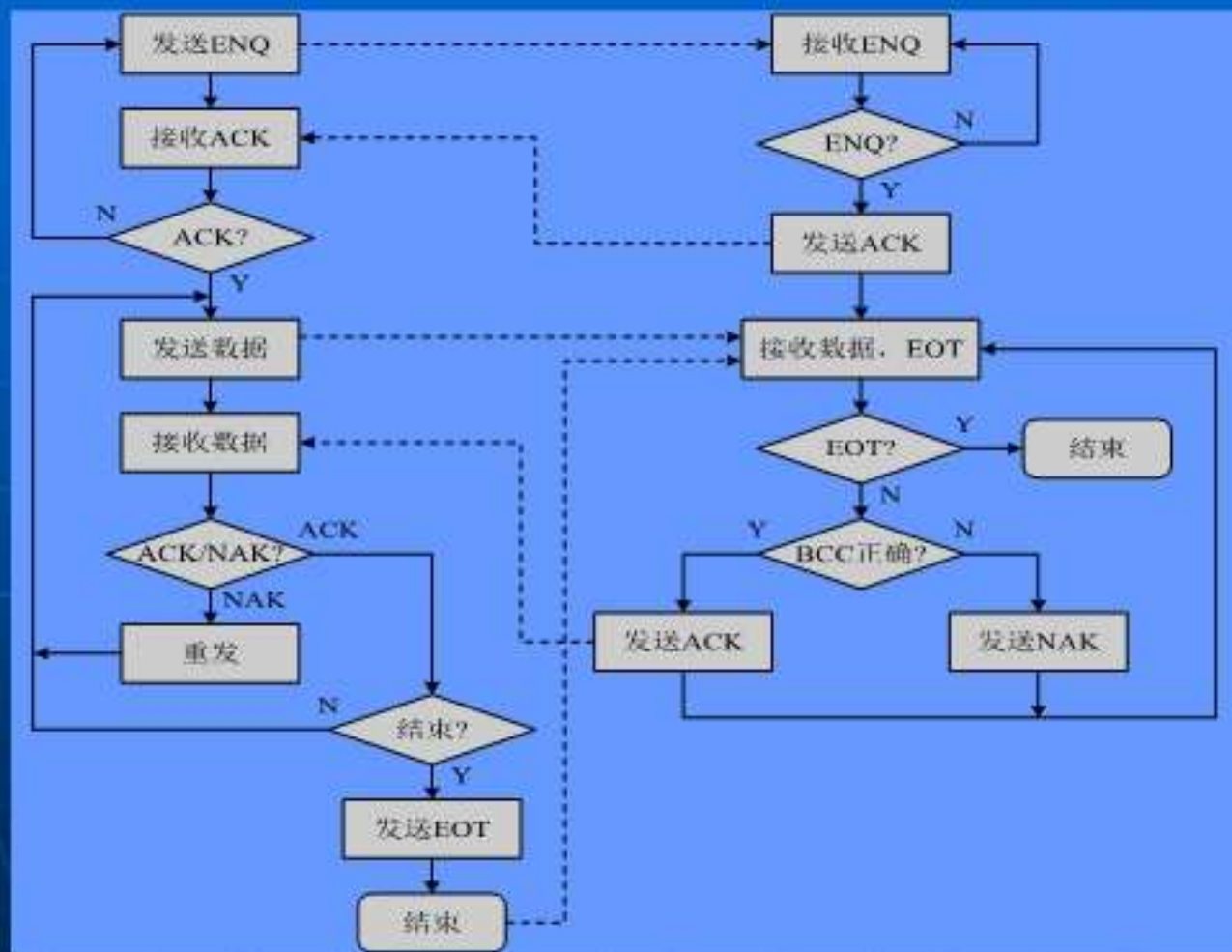


面向字符型协议实例-BSC

◆透明传输的实现

- 规定正文中出现与控制字符相同的编码时，发送硬件自动在其后插入一个**DLE**字符；当接收方收到**DLE**字符时，由接收硬件自动删除**DLE**字符。

建立、维护与释放数据链路流程图



4.4 面向比特型数据链路层协议

- ◆ **HDLC**产生的背景
- ◆ 数据链路的配置方式和数据传送方式
- ◆ **HDLC**的帧结构
- ◆ 数据链路层的工作过程

HDLC产生的背景

◆面向字符型数据链路层协议的缺点:

- 报文格式不一样;
- 传输透明性不好;
- 等待发送方式, 传输效率低。

◆面向比特型协议的设计目标:

- 以比特作为传输控制信息的基本单元;
- 数据帧与控制 帧格式相同;
- 传输透明性好;
- 连续发送, 传输效率高。

HDLC的帧结构

- ◆ 数据链路层的数据传输以帧为单位，即数据链路协议数据单元。**HDLC**的帧结构具有固定的格式。网络层提交给数据链路层传输的数据放在帧的信息字段中。

标志字段F (8位)	地址字段A (8/16位)	控制字段C (8/16位)	信息字段I (长度可变)	帧校验字段FCS (16/32位)	标志字段F (8位)
---------------	------------------	------------------	-----------------	----------------------	---------------

HDLC的帧结构

◆其中：

- **F (flag)**：固定格式— **01111110**，实现帧同步传输数据的透明性（零比特插入与删除）
- **A (address)**：地址
- **C (control)**：帧的类型、帧的编号、命令与控制信息
- **I (information)**：网络层数据， $N_{\max} = 256B$
- **CRC (checksum)**：校验A、C、I字段的数据 $G(X) = X^{16} + X^{12} + X^5 + 1$

◆透明传输的实现：0比特插入/删除法

帧类型及控制字段的意义

标志字段F (8位)	地址字段A (8/16位)	控制字段C (8位)	信息字段I (长度可变)	帧校验字段FCS (16/32位)	标志字段F (8位)
---------------	------------------	---------------	-----------------	----------------------	---------------

	b0	b1	b2	b3	b4	b5	b6	b7
信息帧I	0	N(S)		P/F	N(R)			
监控帧S	1	0	监控		P/F	N(R)		
无编号帧U	1	1	未分配		P/F	未分配		

HDLC的帧结构

◆ 帧类型:

- **I帧**：**N(S)** — 发送帧的序号
N(R) — 接收帧的序号
P/F = Poll / Final, **P=1** 询问, **F=1** 响应, **P**与**F**成对出现
- **S帧**：监控功能位
S = 00, **RR** (receive ready)
S = 01, **RNR** (receive not ready)
S = 10, **RJE** (reject)
S = 11, **SREJ** (select reject)
- **U帧**：用于实现数据链路控制功能

U帧的格式与链路控制功能

无编号帧

01111110

A

1 1

M

P/F

M

FCS

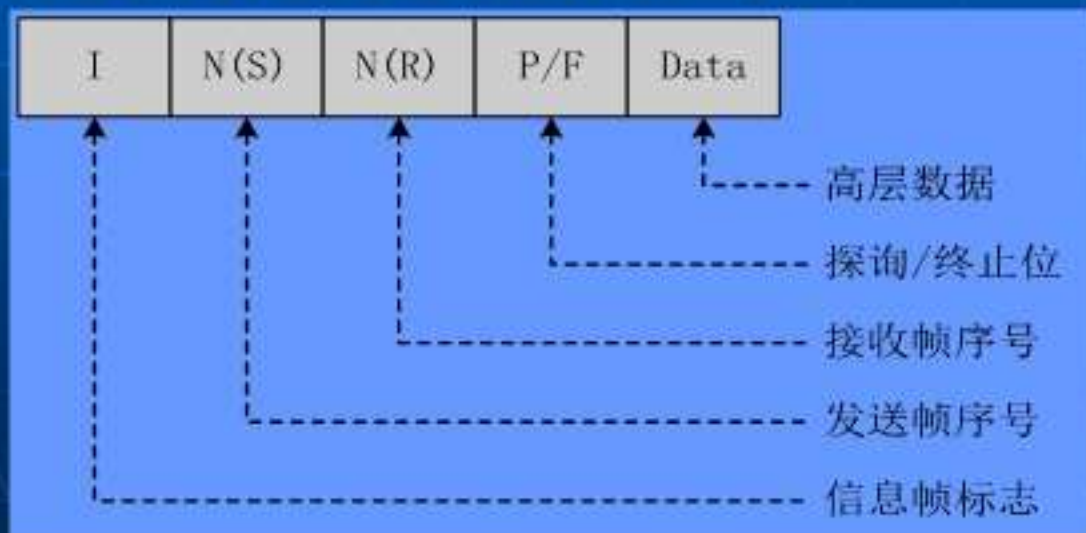
01111110

命令 响应

置异步响应	SARM		1	1		0	0	0	建立主从的点-点结构
置正常响应	SNRM		0	0		0	0	1	建立主从的多点结构
置异步平衡响应	SABM		1	1		1	0	0	建立复合站的平衡结构
拆链	DISC		0	0		0	1	0	结束已建立的数据链路
无编号确认		UA	0	0		1	1	0	从站响应主站的命令
命令拒绝		CMAD	1	0		0	0	1	从站报告帧传输异常

数据链路层的工作过程

- ◆在建立了物理链路之后，通信双方才可以进行数据链路的建立；数据链路的工作过程分为建立、帧传输和释放三个阶段。
- ◆简化的信息帧结构的表示方法



数据链路层的工作过程

◆ 一个信息帧的表示

I, N(S)=3, N(R)=4, P=1

◆ 无编号帧的表示方法

置异步响应模式:

U, SNRM, P=1

无编号确认:

U, UA, F=1

数据链路层的工作过程

◆ SNRM帧与UA帧结构的表示方法

置异步响应模式:

01111110

A

11001001

FCS

01111110

无编号确认:

01111110

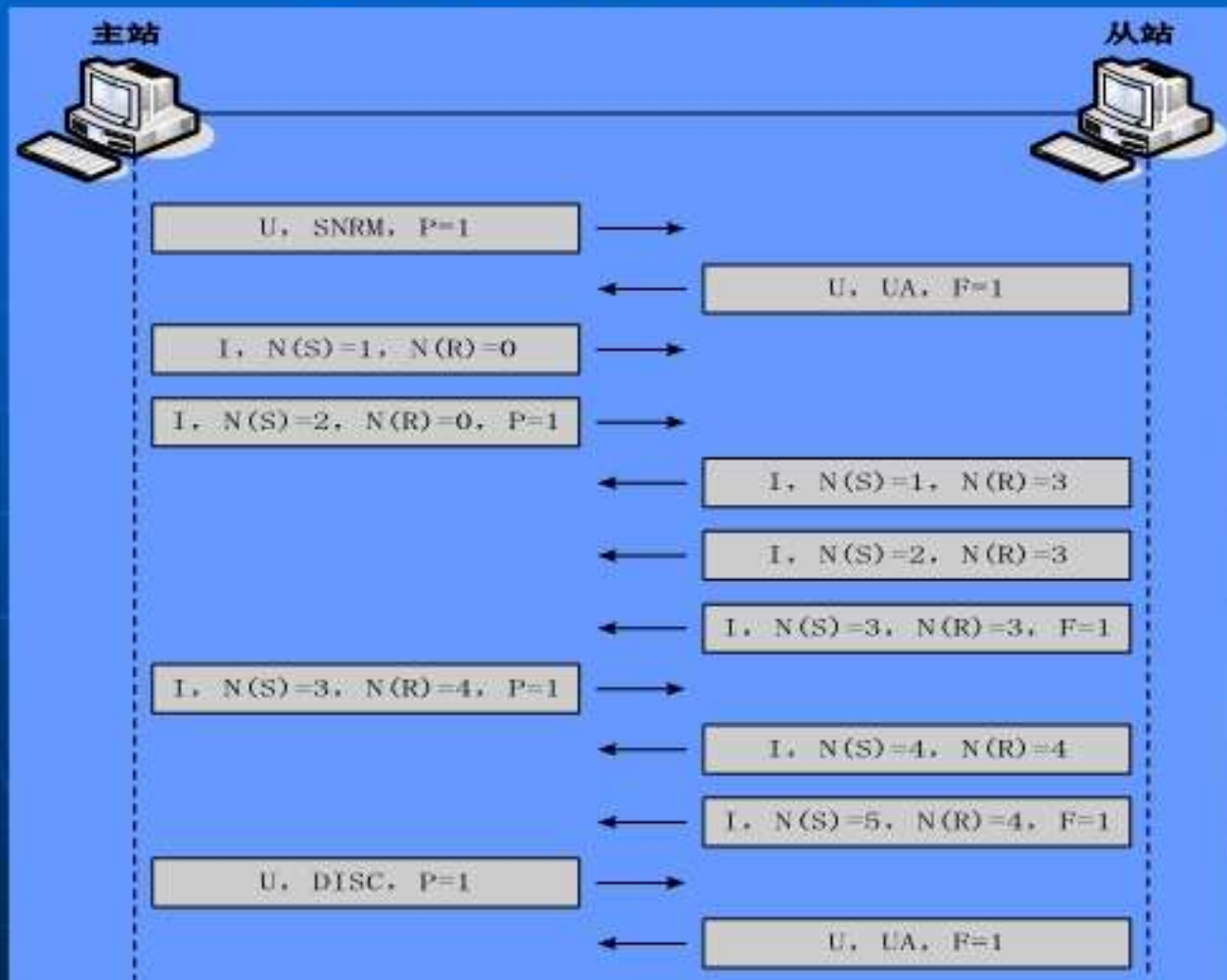
A

11001110

FCS

01111110

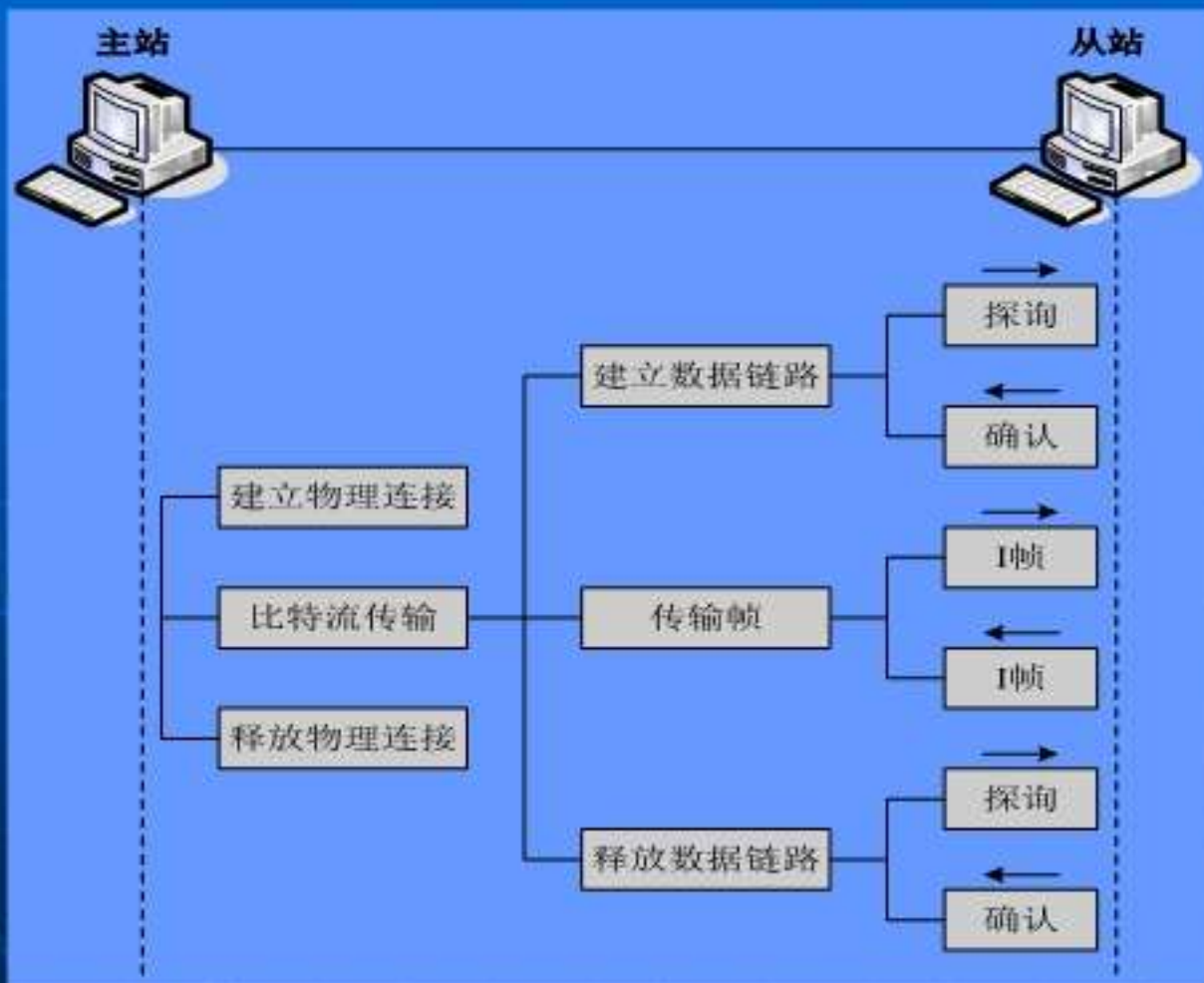
正常响应模式数据链路工作



数据链路层的工作过程

- ◆ 因此，通过数据链路层，收发双方可以有条不紊、正确地完成数据帧传输，使物理线路上可能出现的数据传输错误得到即时的发现与纠错，从而提高数据传输的可靠性。

数据链路层与物理层的关系



4.5 Internet中的数据链路层

- ◆ Internet中主要的的数据链路层协议
- ◆ PPP协议

Internet中主要的数据链路层协议

◆ **SLIP (Serial Line IP)**

➤ 串行线路的Internet数据链路层协议

◆ **PPP (Point-to-Point Protocol)**

➤ 点-点协议

◆ **SLIP与PPP**用于串行通信的拨号线路上，是目前家庭计算机或公司用户通过**ISP**接到**Internet**主要的协议。

PPP协议

◆PPP协议基本特点

- PPP协议是Internet标准，RFC 1660、RFC 1661定义了PPP协议与帧结构；
- PPP协议处理了差错检测，支持面向字符型协议与面向比特型协议，可以支持IP协议及其他一些网络层协议（例如IPX协议）；
- PPP协议不仅在拨号电话线，并且在路由器—路由器之间的专用线上广泛应用；
- PPP协议是在大多数家庭个人计算机和ISP之间使用的协议，它可以作为在高速广域网和社区宽带网协议族的一部分。

PPP协议

- ◆根据**PPP**协议的帧中包含的信息、格式和目的，**PPP**协议的帧可分为：**PPP**信息帧、**PPP**链路控制**LCP**帧和**PPP**网络控制**NCP**帧。
- ◆通信过程
 - 每个端结点首先要发送**LCP**帧，已配置和测试**PPP**数据链路；
 - PPP**链路建立后，每个端点发送**NCP**帧，以选择和配置网络协议；
 - 当网络协议配置好后，网络层的数据包可以通过**PPP**信息帧传输。

PPP协议

◆ PPP信息帧格式



PPP协议

◆ PPP信息帧格式

- 标志 (flag) : 01111110
- 地址 (address) : 值为“FF”(11111111), 表示网中所有的站都接收该帧
- 控制 (control) : 值为“03”(00000011)
- 协议 (protocol) : 长度为2字节, 它标识出网络层协议数据域的类型。常用的网络层协议的类型主要有:
 - 0021H—TCP/IP
 - 0023H—OSI
 - 0027H—DEC
- 数据字段: 长度可变

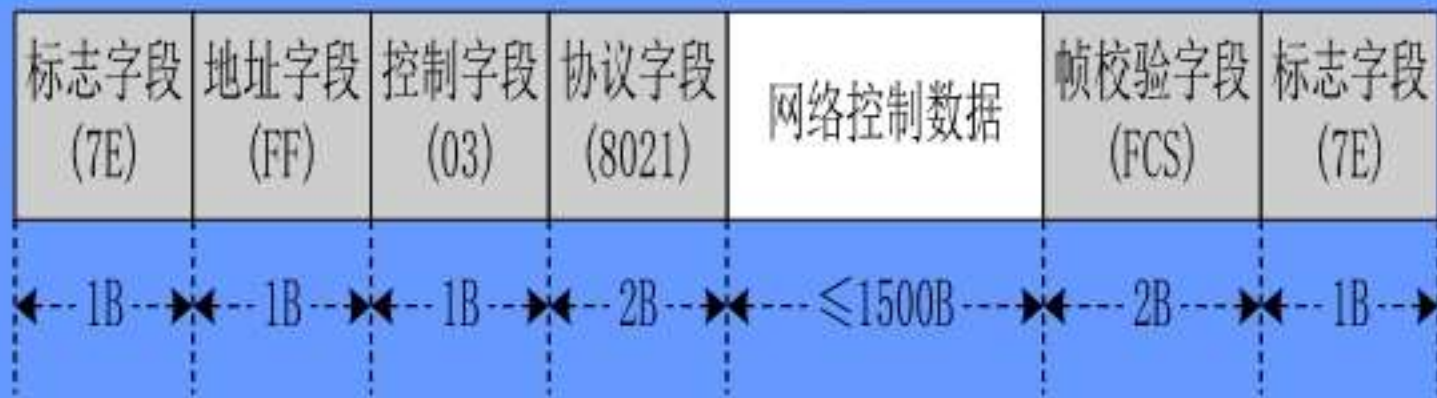
PPP协议

◆ 链路控制帧 (LCP)



PPP协议

◆ 网络控制帧 (NCP)



PPP协议

◆PPP协议功能总结

- 用于串行链路的基于HDLC数据帧封装机制;
- LCP用于建立、配置、管理和测试数据链路的连接;
- NCP用于建立和配置不同的网络层协议。

小结

- ◆ 物理传输线路上传输数据信号是有差错的;
- ◆ 误码率是指二进制比特在数据传输过程中被传错的概率;
- ◆ 数据链路层是将一条原始的、有差错的物理线路变为对网络层无差错的数据链路;
- ◆ 数据链路层完成链路管理、帧传输、流量控制、差错控制等功能;

- ◆ 数据链路层协议基本可以分为两类：面向字符型与面向比特型；
- ◆ 数据链路层的数据传送单位是帧，帧具有固定的结构；
- ◆ HDLC的帧分为信息帧（I帧）、无编号帧（U帧）与监控帧（S帧）；
- ◆ Internet数据链路层主要的协议是PPP协议；
- ◆ PPP协议不仅用于拨号电话线，并且可以用于路由器—路由器之间的专用线路上。