



第三章 数据链路层



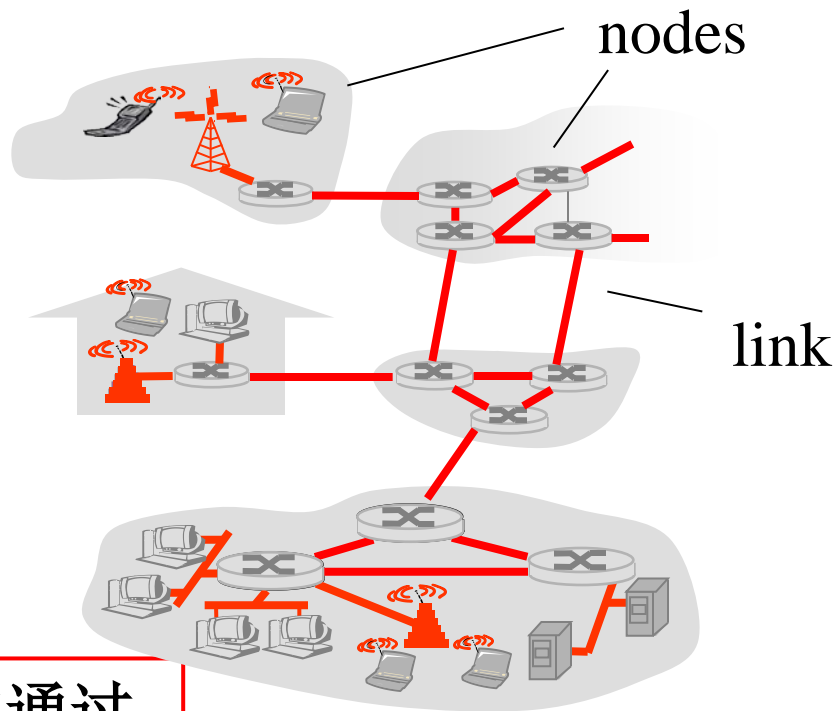
本章内容

- 概述
- 差错检测
- 可靠数据传输
- 停等协议
- 滑动窗口协议
- PPP协议

概述

一些术语:

- ❑ 主机和路由器是节点(nodes)
- ❑ 连接相邻节点的通道是链路(links)
 - ❖ 有线链路(wired links)
 - ❖ 无线链路(wireless links)
 - ❖ 局域网(LANs)
- ❑ 第2层的数据包(packet)是帧(frame)

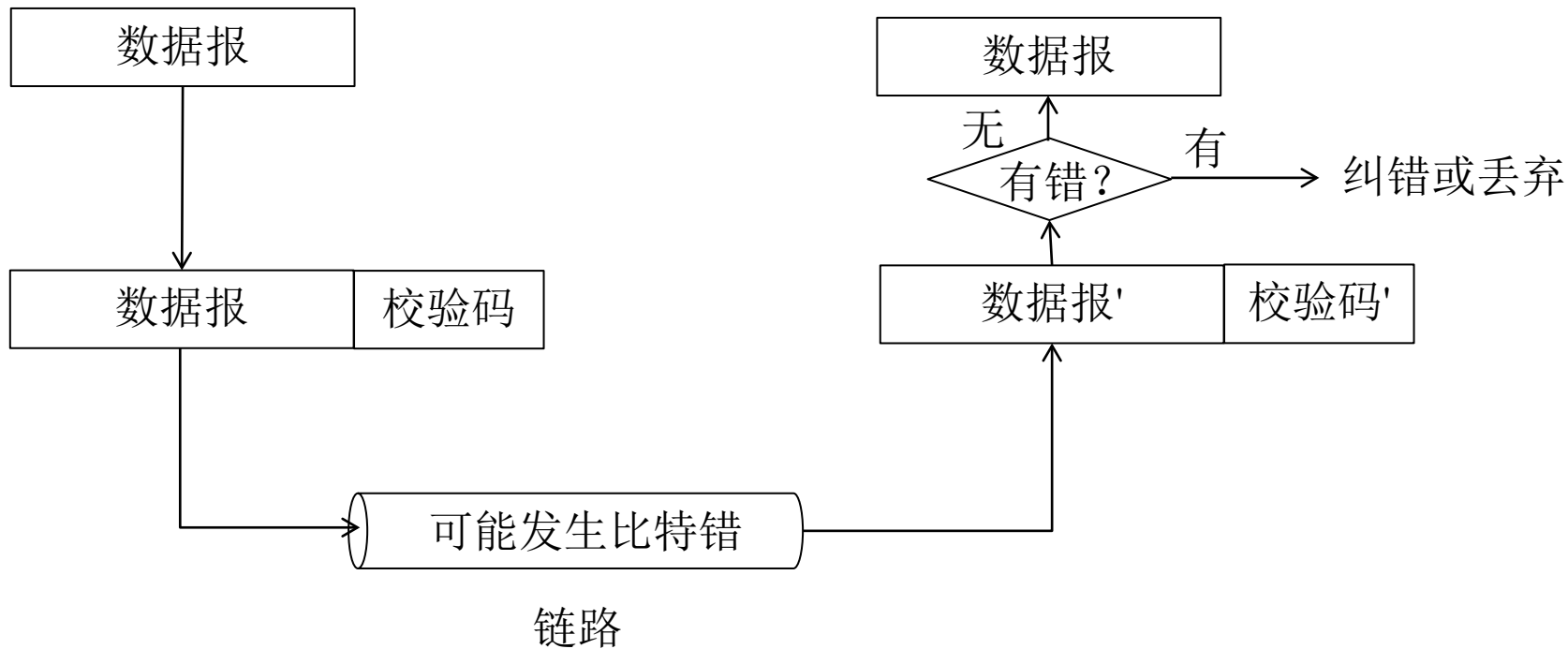


数据链路层负责把数据从一个节点通过链路传给相邻的另一个节点。

功能

- ❑ 形成帧 (framing)
- ❑ 差错检测(error detect): 比特错, 纠错
- ❑ 差错控制(error control): 丢包、重复、错序。
流控制(flow control)
- ❑ 介质访问控制(media access control): 多路访问, 碰撞(collision)

差错检测



检验和

(Checksum)

$$\begin{array}{r} 10000110 \ 10000111 \\ + \ 00000100 \ 01000100 \\ \hline \end{array}$$
$$\begin{array}{r} 10001010 \ 11001011 \\ + \ 11000000 \ 00000000 \\ \hline \end{array}$$
$$\begin{array}{r} 1 \ 01001010 \ 11001011 \\ + \hspace{15em} 1 \\ \hline \end{array}$$

01001010 11001100

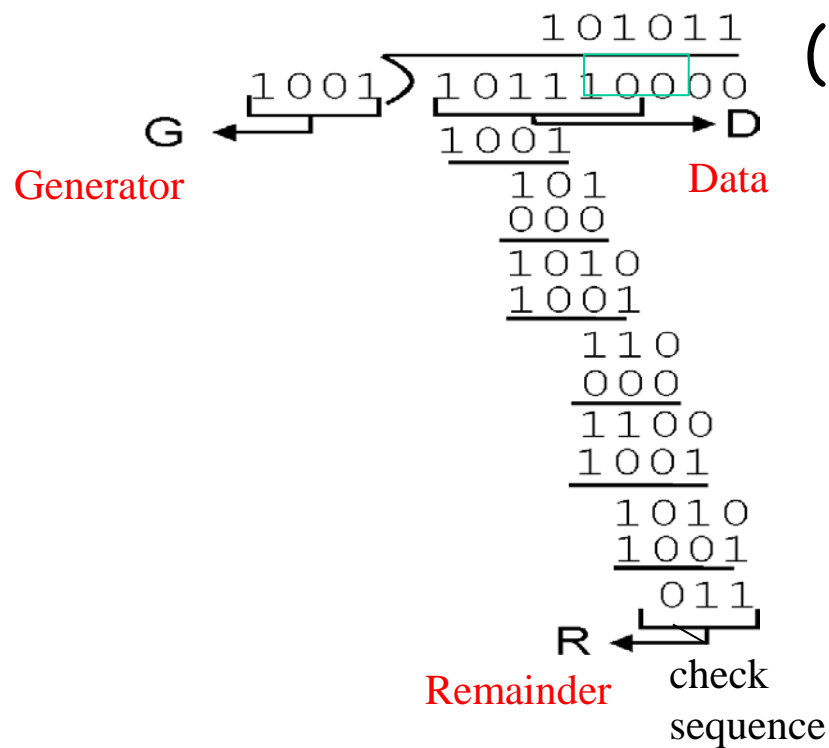
反码: 1011 010100110011

10000110 10000111 00000100 01000100 11000000 00000000 10110101 00110011

数据

校验和

循环冗余校验码



(Cyclic Redundancy Check, CRC)

101110 011

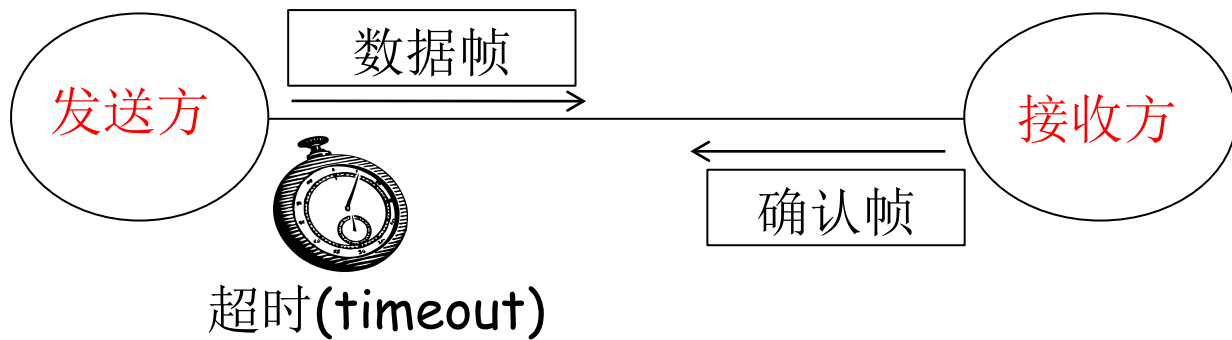
数据 校验码

如果传输过程中没有出现比特错，接收方用相同的除数去除数据加CRC校验码，余数应该为0。

采用模2除法：做减法时没有借位，类似于按位异或。

可靠数据传输

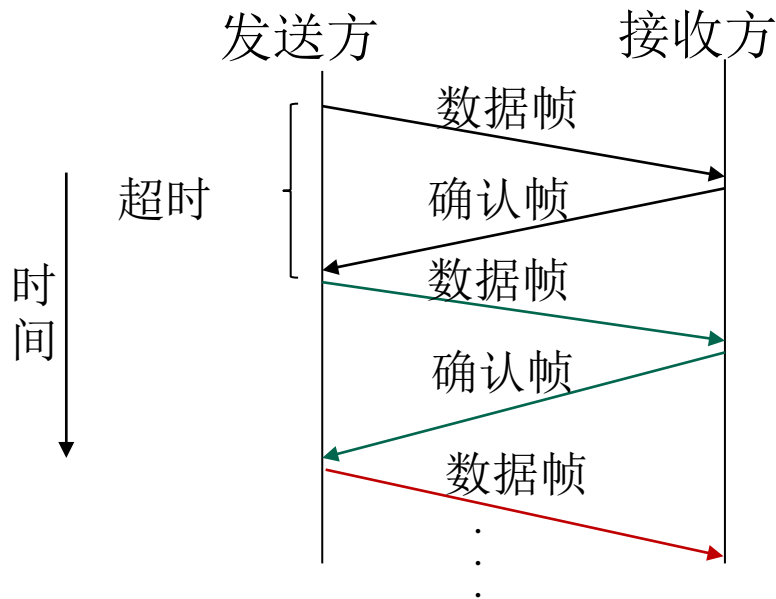
(Reliable Data Transfer)



如果不纠错，出现比特错的帧将被丢弃，即出现丢包错误(loss)。

自动重发请求: Automatic Repeat reQuest(**ARQ**)

停等协议



序号问题

例子: **1000Mbps** 链路, **15 ms**传播延迟,
1KB的帧, 停等协议, 该链路的吞
吐量是多少?

帧长 $L=1\text{KB}$

数据传输率 $R=1000\text{Mbps}$

往返时间 $(RTT)=2*15\text{ms}=30\text{ms}$

发送延迟 $(L/R)=1\text{KB}/1\text{Gbps} = 0.008\text{ms}$

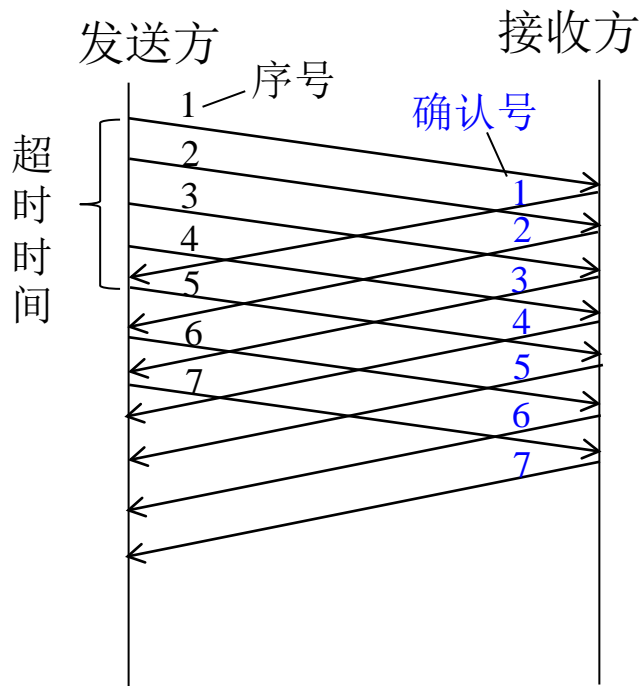
吞吐量(throughput) $=L/(RTT+L/R)$

$=1\text{KB}/30.008\text{ms}$

$=0.266\text{Mbps}$

滑动窗口协议

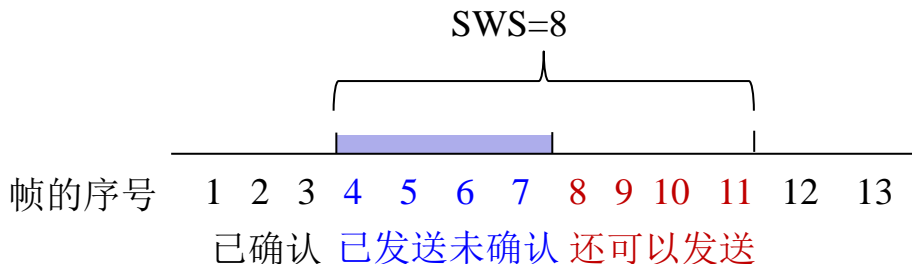
(Sliding Window)



确认帧(Acknowledgement Frame)是接收方发给发送方的一个控制帧，表示这一帧(确认号为ACK)及之前的数据帧全部收到并已交给上层协议。

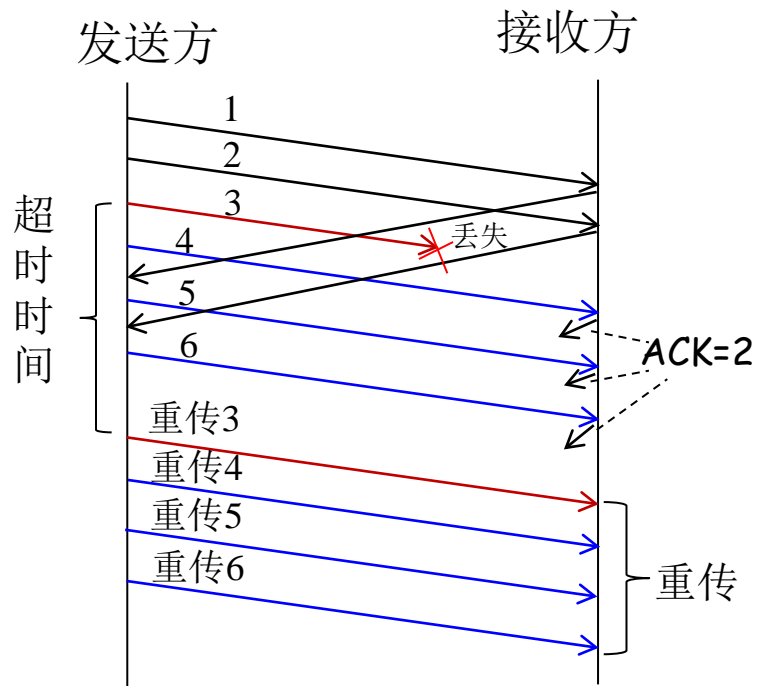
发送窗口表示在已确认的帧之后还能连续发送多少数据帧，用序号范围表示，主要用于流控制(flow control)。

发送窗口大小(Sending Window Size, SWS)表示发送窗口的大小，也是发送缓冲区的大小。

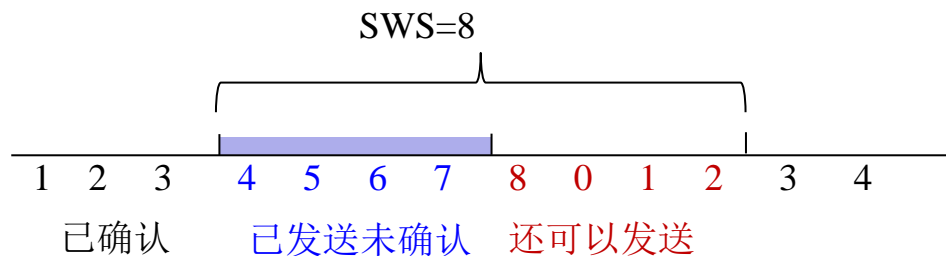


回退N协议

(go back N)

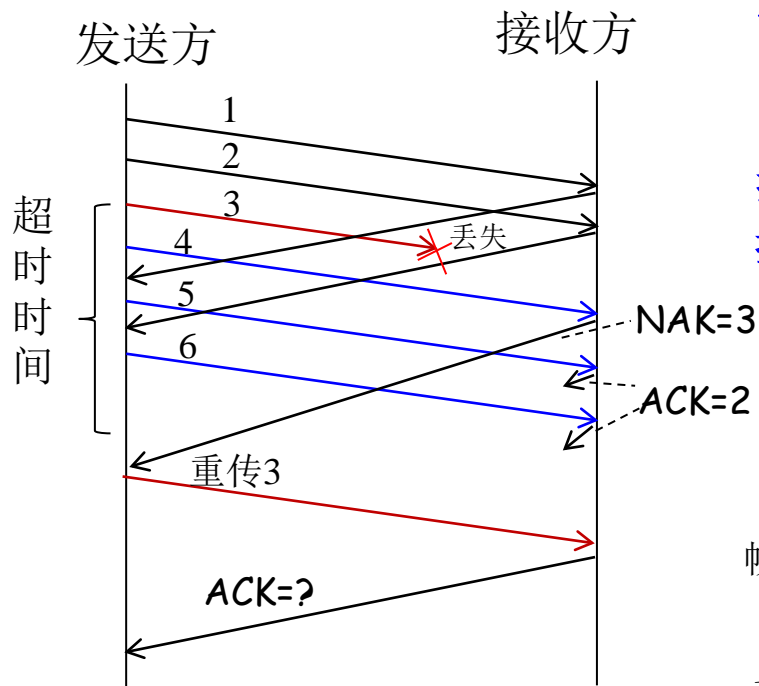


序号可以循环使用



选择性重传

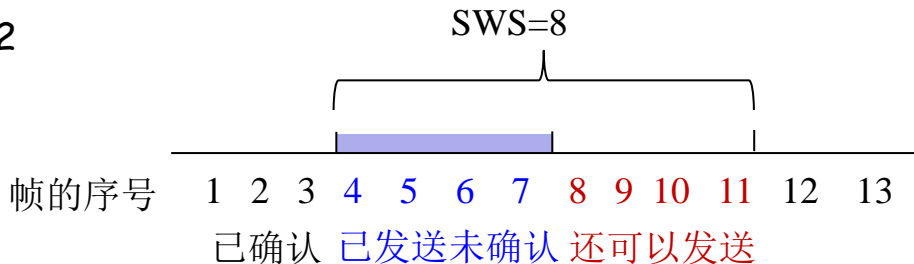
(Selective Repeat)



否定性确认帧(Negative Acknowledgement, NAK)用于表示这一帧之前的数据帧全部收到并已交给上层协议，要求发送方重传这一帧。

接收窗口用于确定应该保存哪些帧，用序号范围表示。

接收窗口大小(Receiving Window Size, RWS)表示接收窗口的大小，也是接收缓冲区的大小。



丢失NAK是否会出错？

提高滑动窗口协议的效率

选择性确认(Selective Acknowledgement)

接收方把已收到的帧的序号告诉发送方。

捎带确认(Piggybacking)

通信双方其实是全双工方式工作。接收方在发数据给对方时顺便把确认号也告诉对方。

延迟确认(Delayed Acknowledgement)

接收方收到一帧后并不立即发送确认帧，而是等待一段时间再发送。

PPP 协议

RFC 1557

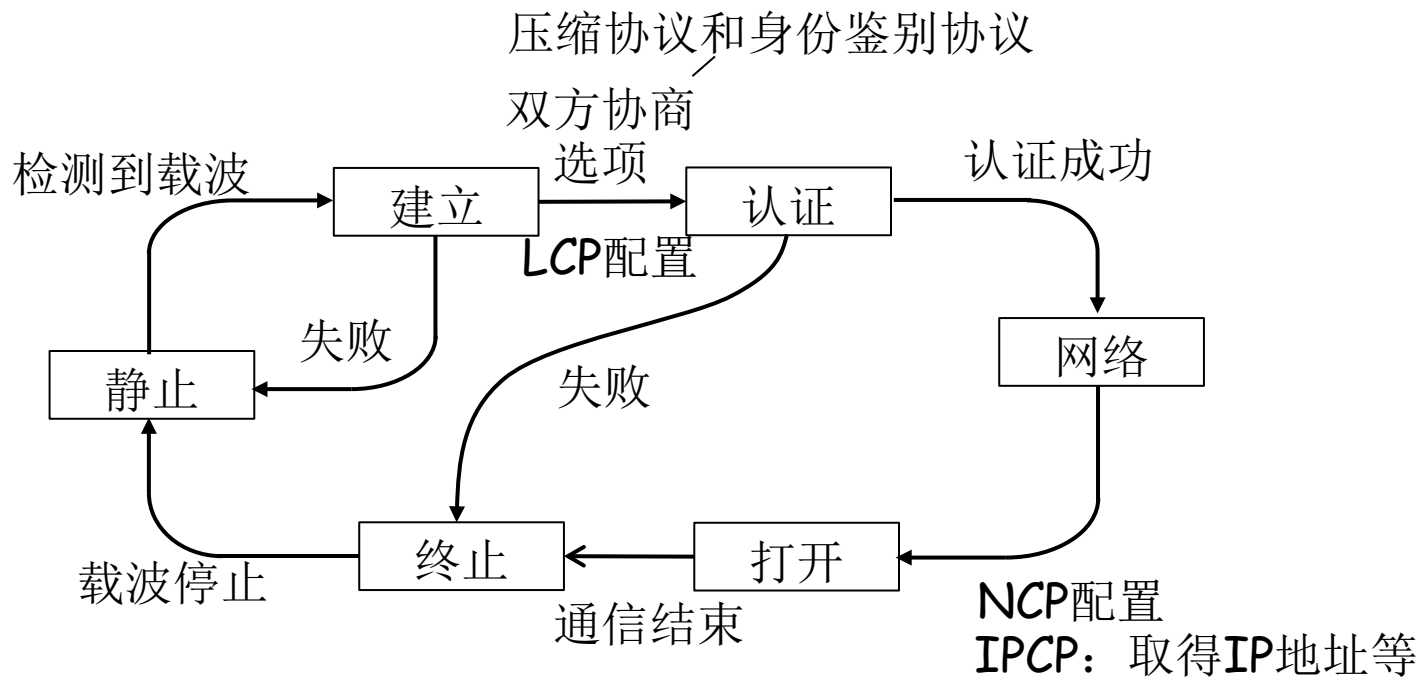
- ❑ PPP协议(Point-to-Point Protocol) 是点到点网络的数据链路层协议。
- ❑ PPP协议是根据HDLC(High-Level Data Link Control)协议进行设计的，主要用于串行电缆(V.35)、电话线(MODEM) 等各种串行链路。
- ❑ PPP协议可以提供连接认证、传输加密和压缩功能。
- ❑ PPP协议可以为各种网络层协议提供服务：因特网的IP协议、Novell公司的IPX协议、苹果公司的AppleTalk协议。
- ❑ PPP协议的多链路捆绑技术可以通过将通信两端之间的多条通信链路捆绑成一条虚拟的链路而达到扩充链路可用带宽的目的。
- ❑ ADSL的PPPoE协议和VPN中的PPTP协议等采用了PPP协议进行封装。

PPP数据帧

字节: 1	1	1	1 或 2	≤1500	2 or 4	1
标志	地址	控制	协议	信息	校验码	标志
01111110	0xFF	0x03			CRC-16/CRC-32	01111110

- ❑ PPP协议采用HDLC的广播地址(0xFF)和无编号帧(0x03)。
- ❑ PPP协议采用字节填充法(Byte-stuffing): 信息字段出现的标志字节(0x7E)用0x7D-5E 替换, 0x7D用0x7D-5D替换。所有小于0x20的字节(控制字节)都加上值0x20, 并在前面加上转义字符0x7D, 例如, 0x01用0x7D-21替换。
- ❑ PPP协议的16位CRC校验码的除数为 $x^{16} + x^{12} + x^5 + 1$ 。
- ❑ 协议字段(Protocol)指明上层协议, 例如, 0x802B-IPX, 0x0021-IP, 0xC021-LCP, 0x8021-IPCP等。
- ❑ 可以省略地址和控制字节(头部压缩)。PPP协议还有TCP压缩和数据压缩协议。
- ❑ PPP协议没有纠错功能, 也没有流控制和确保有序的功能。

PPP协议的状态图

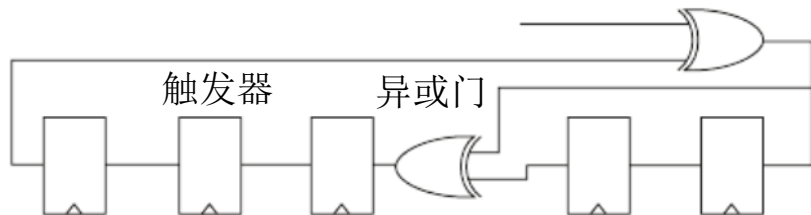


Link Control Protocol(LCP)

Network Control Protocol (NCP): IPCP(Internet), IPXCP(Novell)

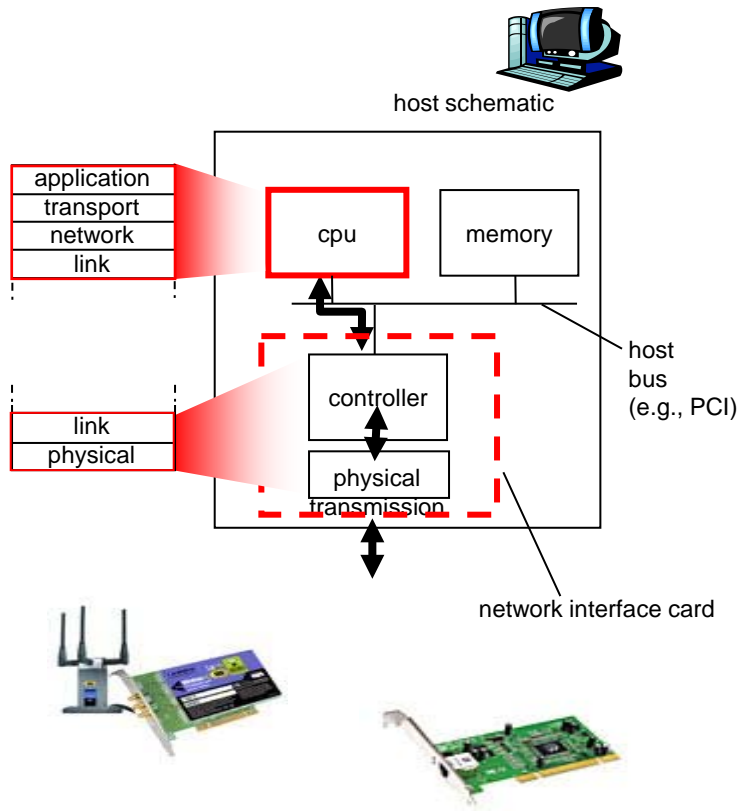
链路层的实现

- ❑ 链路层主要在网络接口卡(*network interface card, NIC*)及其驱动程序上实现。路由器是在接口模块上实现。
- ❑ 链路层校验一般采用**CRC**检验, 因为容易用硬件实现, 速度快, 检错率很高。



$$G(X)=X^5+X^2+1$$

CRC-CCITT($x^{16}+x^{12}+x^5+1$)的错误检测能力: 可以检出所有随机奇数位错误和双位错, 可以检出所有长度小于等于**16**位的突发错; 对于长度等于**17**位的突发错误, 检错率为**99.9969%**; 长度大于等于**18**位的突发错误, 检错率为**99.9985%**。



本章完