一 概述

1. 面向连接的服务和无连接的服务

•面向连接的服务(connection-oriented service)

– 类似电话系统。摘机，拨号，等待对方摘机，谈话，挂断

– 管道：发送者在一端放入物体，接收者在另一端按同样次序取出物体

• 无连接服务(connectionless service)

– 邮政系统：每个报文带有完整地址，由系统选择路线传递，可能会乱序

– 数据报服务，有确认的数据报服务

TCP—面向连接；UDP—无连接服务

2. ISO/OSI 参考模型（各层名称，作用）

a. 物理层 physical layer--与传输媒体的接口，完成传输媒体的信号与二进制数据间的转换

• 物理接口上发送或接收的是一串以某种规则表示的二进制的数据

• 物理层定义是接口的机械特性、电气特性、功能和过程特性等

• 例如：插头、插座的几何尺寸，每根引脚的功能定义，逻辑[0]和[1]的电平定义，信号宽带定义

b. 数据链路层 data link layer--提供点到点的可靠传输，通常把数据分帧，并且保证帧的正确接收

• 识别帧的标志 • 帧的接收要校验要确认 • 发送方在超时或收到否定性确认后，要重发

• 重复帧要丢弃 • 还要解决信道共享问题等等

c. 网络层 network layer--一台主机与另一台主机通过网络进行通信，其间可能存在很多条通路，网络层将选择路径

• 选择路由 • 拥塞控制 • 协议的转换 • 分段和重组 • 对用户的分组字符等计数等等

d. 传输层 transport layer

• 应用程序到应用程序的通路• 传输层将把高层要求传输的数据分成若干个报文 • 报文与帧不一样，帧只有帧标志(起始标志、结束标志)，而报文有信源和信宿的地址及端口、报文的顺序号、确认号等等

• 低三层的通信对象通常是路由器，传输层是端到端的，只要考虑该报文怎样能从源端正确地传输到目的端

e. 会话层 session layer

• 建立有关会话的机制。或双向对话，或双向对话要有切换等 • 如：说的一方应说一段就听一下对方的反应，因为可能线路已断

f. 表示层 presentation layer

– 表示层关心的是语法和语义 – 对相关的数据的描述采用抽象的定义,如浮点数都用科学表示法

• 相关数据的表示法转换 • 抽象数据结构的转换

g. 应用层 application layer--包括所有应用方面的协议

• 如全屏幕功能不同的终端,其控制字符不尽相同应作相应的转换,通常定义一个网络虚拟终端

– 不同系统之间的文件传输的方式不同,但表示的形式必须一致

3. TCP/IP 网络参考模型（各层名称，作用，与 OSI 的作用与对比）

a. TCP/IP 中的互联网层

• 这里的互联网是基于无连接的分组交换网络

• 互联网层定义了正式的分组格式和协议，即 IP 协议(internet protocol),每个 IP 包的路由问题是互联网层要解决的问题

• 互联网层与 OSI 中的网络层相对应

• 一个报文的各个不同的分组称为 IP 包可以通过不同的路径到达目的地其到达顺序可能与发送顺序不一致

b. TCP/IP 中的传输层

• 位于互联网层的上层与 OSI 中的传输层相对应

- 其功能是使源端和目的端主机的对等实体进行对话
- 定义了两个端到端的协议
  - 传输控制协议 TCP(transmission control protocol)
  - 用户数据报协议 UDP(user datagram protocol)

c. TCP/IP 中的应用层

• TCP/IP 模型的应用层包括所有的高层协议。实际上 OSI 模型中的会话层和表示层在很多应用中是没用的

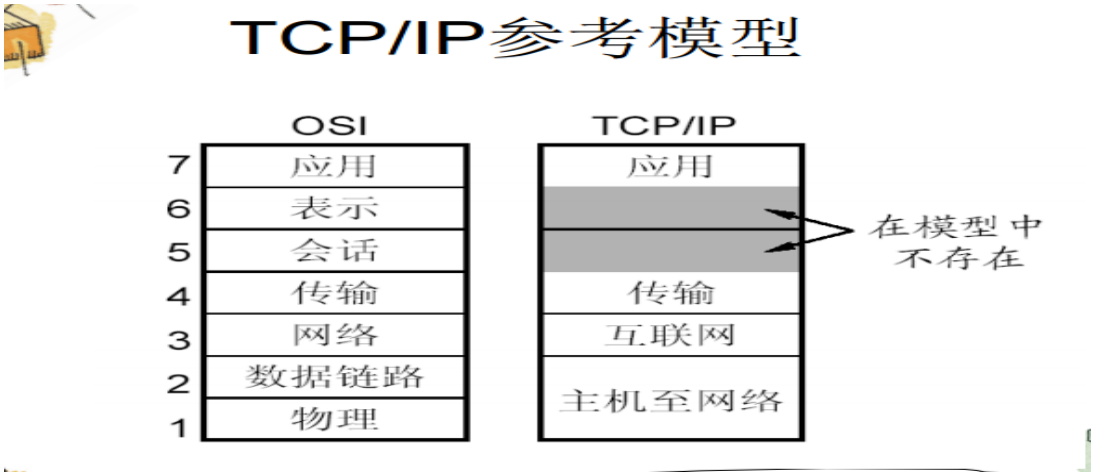
- 应用层常用协议
  - TELNET 标准终端仿真协议
  - FTP File Transfer Protocol 文件传输协议
  - SMTP Simple Mail Transfer Protocol 电子邮件协议
  - DNS Domain Name Service 域名系统服务

d. 主机至网络层

• 在互联网层以下 TCP/IP 参考模型没有定义认为是网络的连接，不认为是一个协议层

- TCP/IP 模型面向的是网络而不是主机

e. OSI 与 TCP/IP



• 传输层及以上层都提供端对端的、与网络无关的传输服务

• 服务、接口、协议概念是一致的

• 7 层 vs. 4 层，网络层，传输层，应用层

• TCP/IP 仅有一种无连接的通信模式，但在传输层上实现面向连接的服务

4. 组织机构名称（ITU,IAB—IRTF,IETF）

ITU: 国际电信联盟,

Internet 标准 IAB(Internet 活动/体系结构委员会)

- RFC, Request For Comments
- 分 IRTF 研究任务组，IETF 工程任务组

二 物理层--研究给定信道上传输数据所受的限制，讨论传输介质，介绍使用这些底层介质的例子

1. Nyquist 定理--在无噪声信道中，当**带宽为 H Hz 信号，电平为 V 级**，则

$$\text{数据传输速率} = 2H\log_2 V \text{ b/s}$$

当信号电平为 V 级,即在二进制中仅为 0,1 两级时,
则以每秒高于 2H 次的速率对线路采样是无意义的因为高频分量已被滤波器滤掉无法再恢复

2. Shannon 定理—高斯噪声干扰通道

Shannel公式： 高斯噪声干扰信道

$$C = W \log_2 (1+\frac{S}{N})$$

C = 传输率， 单位b/s
W = 带宽， 单位Hz

S/N_{dB} 信噪比（dB分贝）的定义

$$S/N_{dB} = 10 \log_{10} S/N$$

即： $\frac{S}{N} = 10^{(S/N_{dB})/10}$

例： 信道带宽W=3KHz， 信噪比为30dB， 则

$C=3000*\log_2(1+1000) \approx 30Kbps$

信道最大数据传输率

Nyquist 公式： 用于理想低通信道

$$C = 2W \log_2 M$$

C = 传输率， 单位b/s或bps
W = 带宽， 单位Hz
M = 信号电平级数

例如：
话音级线路(3000 Hz)
的信道容量计算， 如右
图所示。

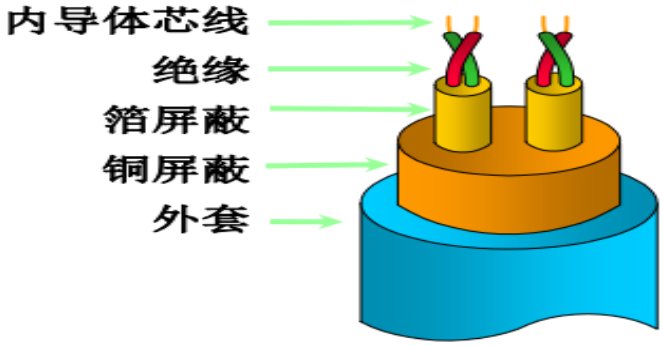
M	最大数据率 (C)
2	6000 bps
4	12000 bps
8	18000 bps
16	24000 bps
32	30000 bps
64	36000 bps

Nyquist公式为估算已知带宽信道的最高速率提供了依据。

Nyquist 公式和 Shannel 公式的比较
前者说明数据传输率 C 随信号编码级数增加而增加。无论采样频率多高，信号编码分多少级，
后者给出了信道能达到的最高传输速率。原因：噪声的存在将使编码级数不可能无限增加。

3. **调频，调幅，调相**
   a. 调幅 ASK(Amplitude Shift Keying)用载波的两个不同的振幅来表示两个二进制值，如用无信号表示 0，有信号表示 1
   b. 调频 FSK(Frequency Shift Keying) 用载波附近的两个不同的频率来表示两个二进制值。如用信号频率为 2f 表示 0 信号频率为 f 表示 1
   c. 调相 PSK(Phase Shift Keying)用载波的相位移动来表示二进制值如用信号相位角为 0 表示 0,相位角为 π 表示 1
4. **双绞线，光纤（多模/单模）速率与距离**
   a. 双绞线（速度 10~1000M/s，传输距离几十千米，模拟/数字传输）

双绞线



--螺旋绞合的双导线，Φ≈1mm
--每根4对、25对、1800对
--典型连接距离100m（LAN）
--RJ45插座、插头
--优缺点：
成本低
密度高、节省空间
安装容易（综合布线系统）
平衡传输（高速率）
抗干扰性一般
连接距离较短

屏蔽双绞线 STP：以铝箔屏蔽以减少干扰和噪音

非屏蔽双绞线 UTP：3 类、5 类、6 类（16M、155M、1200M）双绞线外没有任何附加屏蔽

• EMI（electromagnetic interference，电磁干扰）—电磁干扰是马达等电力设备引起的磁力场产生的信号干扰。

• RFI（radio frequency interference，无线电频率干扰)—因为电气设备发射的无线电波频率与网络信号传输使用的频率相同而引起的信号干扰为无线电频率干扰，也称射频干扰

EIA/TIA-568规范为水平电缆和主线电缆定义的双绞线	屏蔽否	最大传输速度(Mbps)
IBM Type 1A	屏蔽	4
IBM Type 2A	屏蔽	4
Category 3	非屏蔽	16
Category 4	非屏蔽	20
Category 5	非屏蔽	100

Category 5 双绞线是新的电缆安装的非常好的选择，因为其高速网络互连能力可达１００Ｍｂｐｓ。

b. 光纤（速度几十 Gps，传输距离 30 千米以上，远距离传输）

光纤电缆具有进行高速网络传输的能力，它所支持的传输速度可以从１００Ｍｂｐｓ到1Ｇｂｐｓ，甚至更多。

- **多模光缆：通过光的反射在光纤中无损传输距离 2 km。**
- **单模光缆：直线传输距离>30 km，大功率激光驱动可达 100km**

光纤：纤芯-折射率高、玻璃包层-折射率低亮度调制，有脉冲-1，无脉冲-0

单向传输，双向需两根光纤

表2-1　单模光纤与多模光纤的比较

项　目	单模光纤	多模光纤
距离	长	短
数据传输率	高	低
光源	激光	发光二极管
信号衰减	小	大
端接	较难	较易
造价	高	低

常用传输媒体的比较

传输媒体	速率	传输距离	性能(抗干扰性)	价格	应用
双绞线	10-1000Mb/s	几十 kM	可以	低	模拟/数字传输
50Ω同轴电缆	10Mb/s	3kM 内	较好	略高于双绞线	基带数字信号
75Ω同轴电缆	300-450MHz	100kM	较好	较高	模拟传输电视、数据及音频
光纤	几十 Gbps	30kM up	很好	较高	远距离传输
短波	<50MHz	全球	较差	较低	远程低速通信
地面微波接力	4-6GHz	几百 kM	好	中等	远程通信
卫星	500MHz	18000kM	很好	与距离无关	远程通信

5. ADSL 工作原理

- xDSL(Digital Subscriber Line, 数字用户线路) • ADSL(Asymmetric DSL 非对称数字用户线)
- 本地回路的承载能力依赖于几方的因素，包括它的长度、粗细和综合质量
- 普通电话线在端局被滤波，因而带宽被限制在 4K。
- 而 ADSL 线路在端局没有滤波器，因而能承载更高的带宽，大约 1.1MHz

- ADSL 需要在用户家中安装 NID(Network Interface Device)
  - 一个分离器将普通电话频段 0-4000Hz 与数据信号分离开，接原电话或传真机
  - 数据信号接 ADSL 调制解调器处理
  - ADSL 调制解调器相当于 250 个 QAM Modem
- 通常 ADSL Modem 是外置，与计算机连接需要一个快速的方式，一般选择---网线
- 端局使用 DSLAM(Digital Subscriber Line Access Multiplexer, 数字用户线路访问复用器)对信号进行分离、调制解调
- 语音信号被过滤出来，传送给传统的语音交换机
- 数字信号被恢复成 bit 流，送给 ISP

6. FDM, WDM, TDM

主干线常采用时分多路复用技术以提高线路的利用率

FDM (频分复用) WDM (波分复用) TDM (时分复用)

a. 频分复用 Frequency Division Multiplexing
   原理：整个传输频带被划分为若干个频率通道，每个用户占用一个频率通道。频率通道之间留有防护频带。

b. 波分复用 Wave Division Multiplexing
   原理：整个波长频带被划分为若干个波长范围，每个用户占用一个波长范围来进行传输。

c. 时分复用 Time Division Multiplexing
   原理：把时间分割成小的时间片，每个时间片分为若干个通道（时隙），每个用户占用一个通道传输数据。
   TDM 的缺点：某用户无数据发送，其他用户也不能占用该通道，将会造成带宽浪费。
   改进：统计时分多路复用（STDM），用户不固定占用某个通道，有空槽就将数据放入。

7. T1, E1, OC1 线路带宽

T1 线路，1.544Mb/s，T1 线路由 24 个多路复用信道组成

E1 线路，2.048Mb/s，E1 线路由 32 个多路复用信道组成

OC1 线路，51.84Mb/s

三 数据链路层--实现相邻机器间的的可靠、有效通信

1. 成帧，字符填充，位填充

a. 成帧--比特流分成离散的帧，每一帧计算校验和

b. 带字符填充的首尾界符法--用特殊的字符作为帧头和帧尾 如 DLE STXMy name is John DLE ETX 接收方一旦丢失了帧信息只要查找 DLE STX 就可重新确定帧边界

面向字符的帧格式

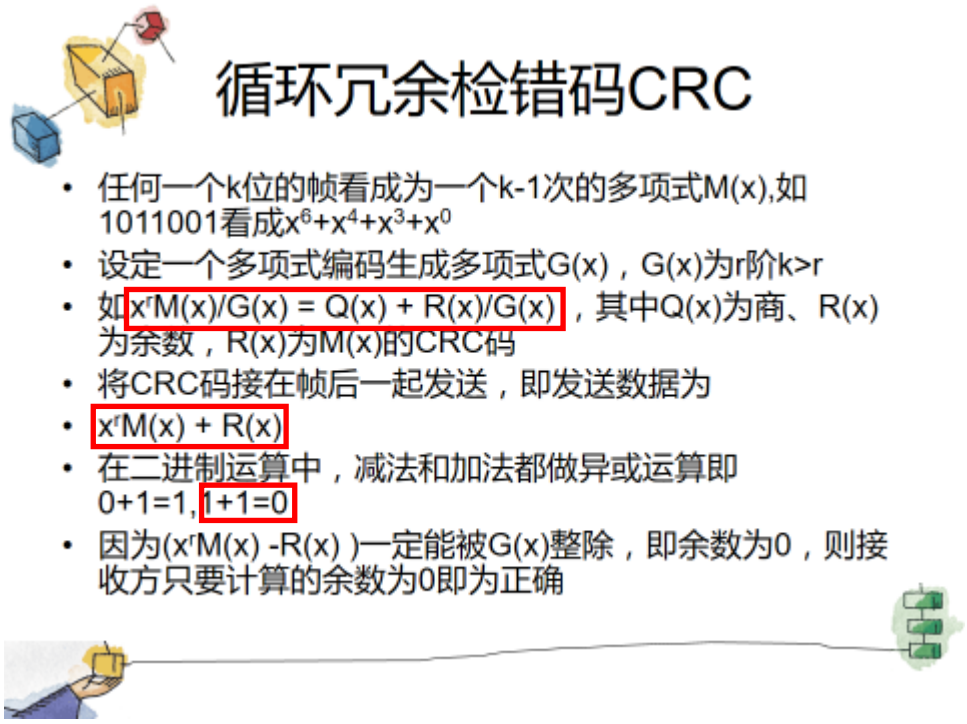
- 这是一种面向字符的帧格式，所传输的数据都是字符(ASCII 字符或 EBCDIC 字符)，在帧内容中不允许出现帧同步字符，在面向字符的异步串型通信中常使用这种格式 。
- 面向字符的帧格式不适宜传输数据中包含二进制数的帧，因为在包含二进制数的帧中很可能出现 DLE STX 等字符 。
- 一种方法是在二进制数中偶然出现的 DLE 前再插入一个 DLE。这就称为**字符填充**

c. **位填充**

带位填充的首尾标志法

- 在面向二进制位的同步串型通信中常使用带位填充的首尾标志格式，如 HDLC
  - 这是一种面向二进制位的帧格式，把所有需传输的数据(不论是 ASCII 字符还是二进制位串)一字排开，并以特殊的位模式 01111110 作为帧标志，即一个帧的开始(同时标志前一个帧的结束)
  - 当帧内容中出现一个与帧标志相同的位串 01111110，则在 5 个 1 后插入一个 0,即变成 01111101，接收方将自动删除第 5 个 1 后的 0。称为位插入法，或透明传输
  - 如果由于干扰，一个帧没有正确接收，则可扫描接收串一旦扫描到 01111110，即新的一帧从此开始。即可以再同步

2. CRC 校验--循环冗余检错码(Cyclic Redundancy Check)



CRC码计算举例

如一帧为1101011011

即M(x)= $x^9+x^8+x^6+x^4+x^3+x+1$

G(x)=x^4+x+1

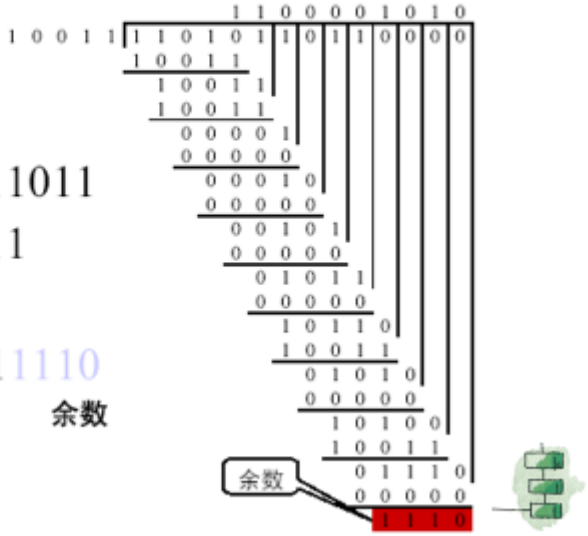
T(x)=x^4M(x)

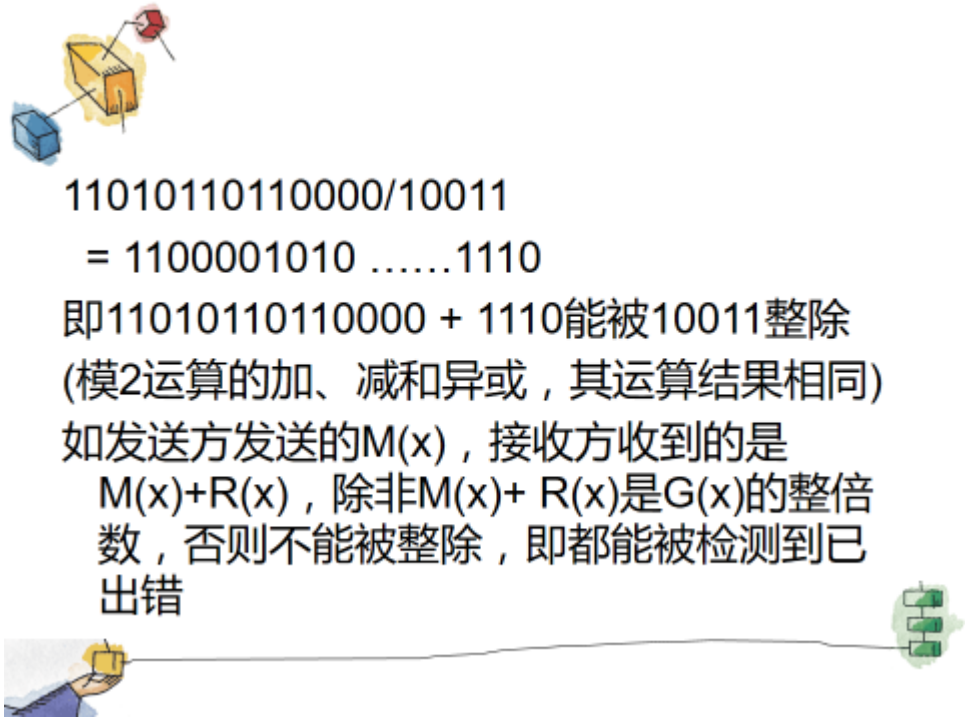
$=x^4(x^9+x^8+x^6+x^4+x^3+x+1)$

$=x^{13}+x^{12}+x^{10}+x^8+x^7+x^5+x^4$

CRC码计算举例

- 帧：1101011011
- 除数：10011
- 传输帧：
  11010110111110
  帧数据　　余数





3. 协议 1，协议 2，协议 3，协议 4

a. 协议 1-- 一种无限制的单工协议

一种理想的环境，理想的协议。假定链路是理想的传输通道，所传输的任何数据既不会出错也不会丢失。即不需校验，也不可能重发，不管发送方以怎样的概率发送数据，接收方都能及时接收。即处理器的处理速度无限高，处理时间可忽略不计，缓冲区空间无限大，毋需流量控制

b. 协议 2--一个单工的停-等协议

- 接收方不可能具有足够高的 CPU 处理能力来及时处理所有的接收帧，也不可能具有足够大的缓冲区。但仍假定 ：
  - 链路是理想的传输通道，所传输的任何数据既不会出错也不会丢失
  - 即不需校验，也不可能重发
- 所以当它来不及处理时，应通知发送方暂缓发送，一旦可以继续接收，则通知发送方继续发送

c. 协议 3 有噪声信道的单工协议

- 在噪声信道中应考虑传输有差错的情况
- 所谓差错
  - 帧的损坏；如帧中若干位出错，通常 CRC 都能检测到
  - 帧完全丢失；一旦帧头出错，接收方将检测下一个帧头以同步，但该帧已丢失
- 所以发送方应启动一个计时器超时即再发送

协议 3 的要点

- 发送方要记录下一个准备发送的顺序号 • 接收方要记录下一个期望接收的顺序号
- 发送过程和接收过程是严格交替的 • 也称为 ARQ 协议 automatic repeat request 自动重复请求
- 即使接收到的不是期望的帧，只要接收到一个正确的帧，都将发送一个空的确认

协议 3 的重发机制存在的问题

- 效率较低 - 如接收方收到的帧出错，或者整个数据帧丢失，则不发确认帧，发送方在超时后重发，直至正确，效率极低
- 接收方会收到重复帧 - 如接收方收到了正确的收到了数据帧并发送了确认帧，但此确认帧丢失，发

送方在超时后重发此帧，这样，接收方的数据链路层收到两个完全相同的帧

- 对重复帧仍发确认止成发送方误解

协议 3 的问题

- 不能双向传输 • 由于应答无序号，导致系统不可靠 • 效率低

d. 滑动窗口协议

- 双向传输
- 数据帧编号，发送方和接收方维持一组序列号，对应发送窗口和接收窗口
- 发送窗口中的序列号代表已经发送但尚未确认的帧
- 接收窗口对应于允许接收的帧，任何落在窗口之外的帧被丢弃

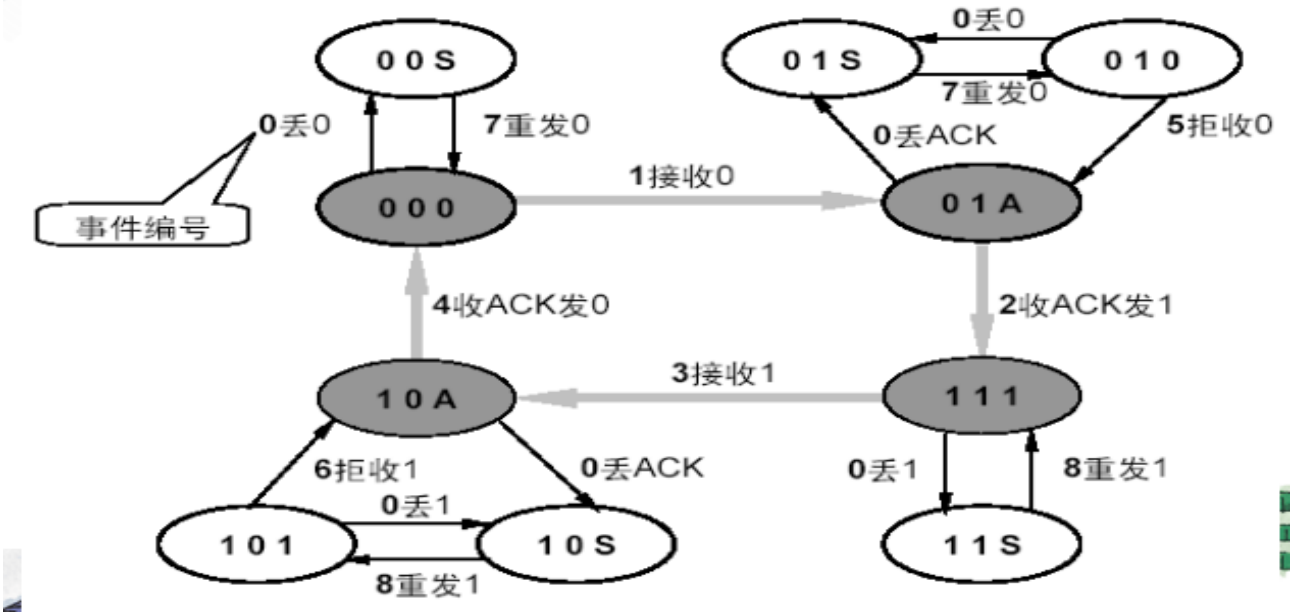
4. 协议 3 的有限状态机模型

- 协议机(protocol machine)：包括发送方和接收方
- 有限状态机把协议形式化为一个四元组(S,M,I,T)其中:
  - S:进程和信道可能进入的状态集合
  - M:能在信道上进行交换的帧的集合
  - I:进程初始状态的集合
  - T:两两状态之间转换的集合
- 每个系统状态都可分解为:发送方状态、接收方状态、信道状态
- 状态在发生某个事件时，可能会转换到另一个状态，把状态作为节点，转换用有向线段表示，则协议的状态图是一个有向图
- 有向图中的节点(状态)是一个三元组(发送方状态,接收方状态,信道状态)

协议3的有限状态机模型

发送方状态	0	发送了0号帧
	1	发送了1号帧
接收方状态	0	期望接收0号帧
	1	期望接收1号帧
信道状态	0	信道上有0号帧
	1	信道上有1号帧
	A	信道上有ACK帧
	S	信道上没有任何

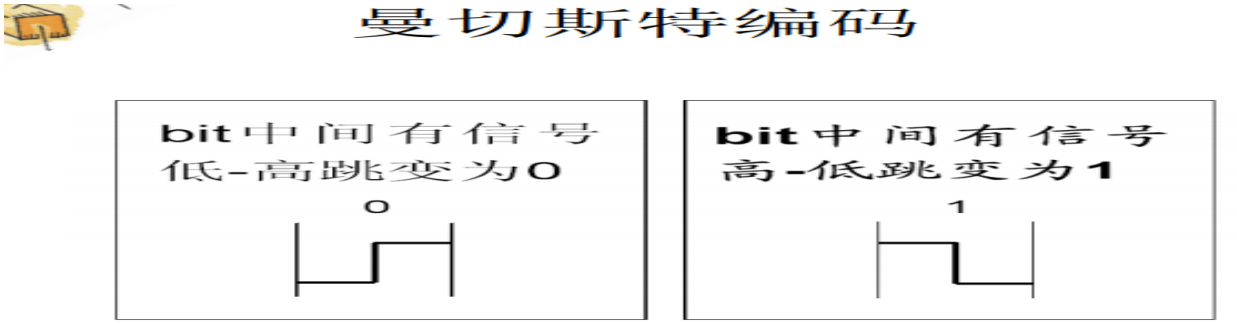
协议3有限状态机状态变迁图

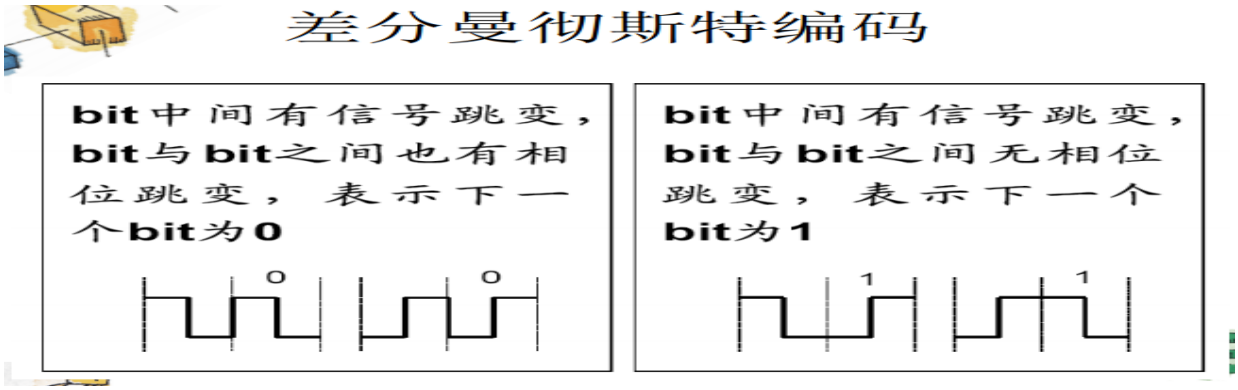


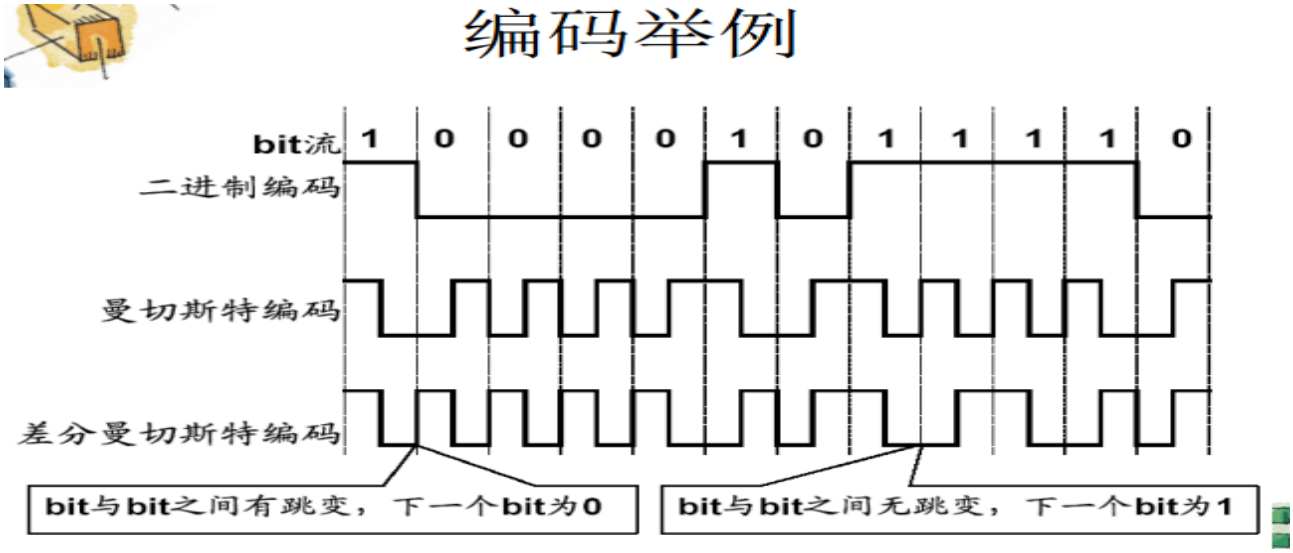
事件和状态变迁表

编号	事件简称	事件	动作	状态变迁
0	数据丢失	信道出错，造成帧丢失	无	(XXX)→(XXS)
1	接收0号帧	0号帧到达正期待0号帧的接收方	接收0号帧，改期待帧号为1，并向信道发ACK	(000)→(01A)
2	发送1号帧	ACK帧到达发送0号帧的发送方	接收ACK帧，发送帧号改为1，并从网络层取分组发送1号帧	(01A)→(111)
3	接收1号帧	1号帧到达正期待1号帧的接收方	接收1号帧，改期待帧号为0，并向信道发ACK	(111)→(10A)
4	发送0号帧	ACK帧到达发送1号帧的发送方	接收ACK帧，发送帧号改为0，并从网络层取分组发送0号帧	(10A)→(000)
5	拒收0号帧	0号帧到达正期待1号帧的接收方	从信道取下0号帧，拒收，即不交网络层，并向信道发ACK	(010)→(01A)
6	拒收1号帧	1号帧到达正期待0号帧的接收方	从信道取下1号帧，拒收，即不交网络层，并向信道发ACK	(101)→(10A)
7	重发0号帧	发送0号帧的发送方等待ACK计时器超时	重发0号帧	(0XS)→(0X0)
8	重发1号帧	发送1号帧的发送方等待ACK计时器超时	重发1号帧	(1XS)→(1X1)

5. 曼彻斯特编码







物理层编码违例法

- 在曼切斯特编码中，连续高电平或连续低电平可用作帧边界
- 采用冗余编码技术，如曼切斯特编码，即两个脉冲宽带表示一个二进制位
- 数据 0：低-高电平对• 数据 1：高-低电平对
- 高-高电平对和低-低电平对没有使用，可用作帧边界，在令牌环网中使用编辑违例格式

四 介质访问子层（MAC 子层）--几乎所有局域网都采用多路复用信道。 在多站点共享信道时，如何分配信道使用权？在数据链路层的下层。

1. ALOHA,分隙 ALOHA

a. 纯 Aloha 系统中，用户只要有数据待发，就让他们发。任何时间，只要两帧试图同时使用信道就

会产生冲突

b. 分隙 ALOHA

把时间分为离散的时间段，每段时间对应一帧。计算机并不是在按下回车键后立即传送信息帧，而是等到下一时隙开始时才传送。由于冲突危险区减少为原来的一半，把 ALOHA 系统利用率提高一倍。这种方法要求用户时间同步。

c. 纯 ALOHA 和分隙 ALOHA 的比较

• 纯 ALOHA 中一旦产生新帧就立即发送，全然不顾是否有用户正在发送，所以发生冲突的可能伴随着发送的整个过程

• 分隙 ALOHA 中规定发送行为必须在时隙的开始，一旦在发送开始时没有冲突，则该帧将成功发送

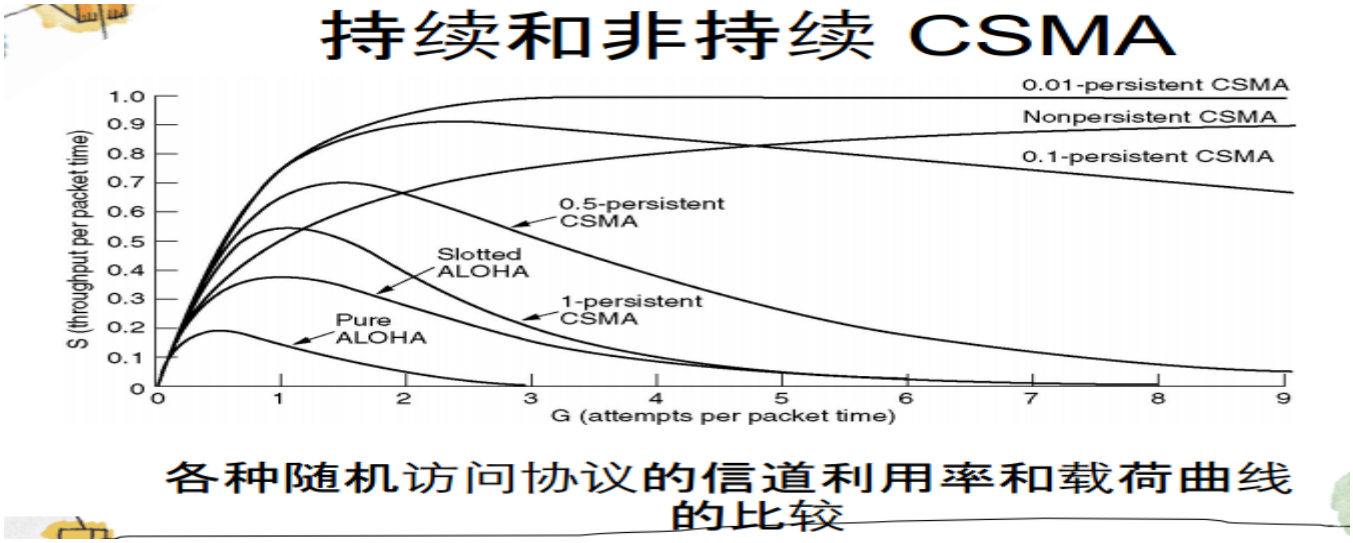
2. CSMA,1-持续 CSMA,非持续 CSMA,p-持续 CSMA

载波侦听多路访问协议(CSMA)

• 局域网中站点可以检测到其他站在干什么，相应地调整自己的动作，网络可以获得更高的利用率——载波侦听协议

• 持续和非持续 CSMA

- 1-持续 CSMA：持续侦听，一旦信道空闲立刻发送，如冲突则延时一随机时隙数后
- 非持续 CSMA：信道忙时，等待一个随机时间再来侦听，信道利用率高，**时延长些**
- p-持续 CSMA：用于分隙信道。信道空时，以概率 p 传送，以概率 1-p 将发送推迟到下一时隙；信道忙则等下一时隙。



各种随机访问协议的信道利用率和载荷曲线的比较

但 CSMA 并不能完全解决冲突问题，如两个或多个准备发送的站都检测到信道空闲而同时发送，将发生冲突

3. 无冲突法：位图协议，二进制倒计数协议

a. 基本位图协议

--竞争周期由 N 个时隙组成。如果站点 0 想发送一帧，就在第 0 个竞争时隙内发送 1 比特。这种发送前首先声明的协议称为预订协议

--位图协议的效率分析

在低负荷条件下，如每帧的数据量的 d bit，额外比特数为 N，则效率为 d/(d+N) ;在高负荷条件下，即所有的站总是有东西要发送，位图按平均分配给每一帧，一帧只占一位，则效率为 d/(d+1)

b. 二进制倒计数法

• 基本位图法的问题是每站点需要 1 比特的额外开销

• 二进制计数法：

- 每个想用信道的站点，具有一个相同宽度的地址
- 首先将其地址以二进制位串的形式，按照由高到低的顺序进行广播，
- 从最高位开始逐位比较是否成功，否则退出竞争 - 赢得信道后发送帧

• 地址编号高的站点总是赢得竞争：

- 虚拟站号，不固定 - 每二进制倒计数法效率分析

• N 个站的二进制编码所需位数是 log2 N 位 • 信道的效率为 d/(d+log2 N)

• 如果规定每个帧的帧头为发送地址，即竞争的同时也在发送，则效率为 100%

4. 有限竞争协议

竞争法 vs. 无冲突法

• 竞争法：

-轻载荷下时延短性能优，重载荷时仲裁开销大

• 无冲突法：

-轻载荷下时延较长，重载荷下，用信道利用率高

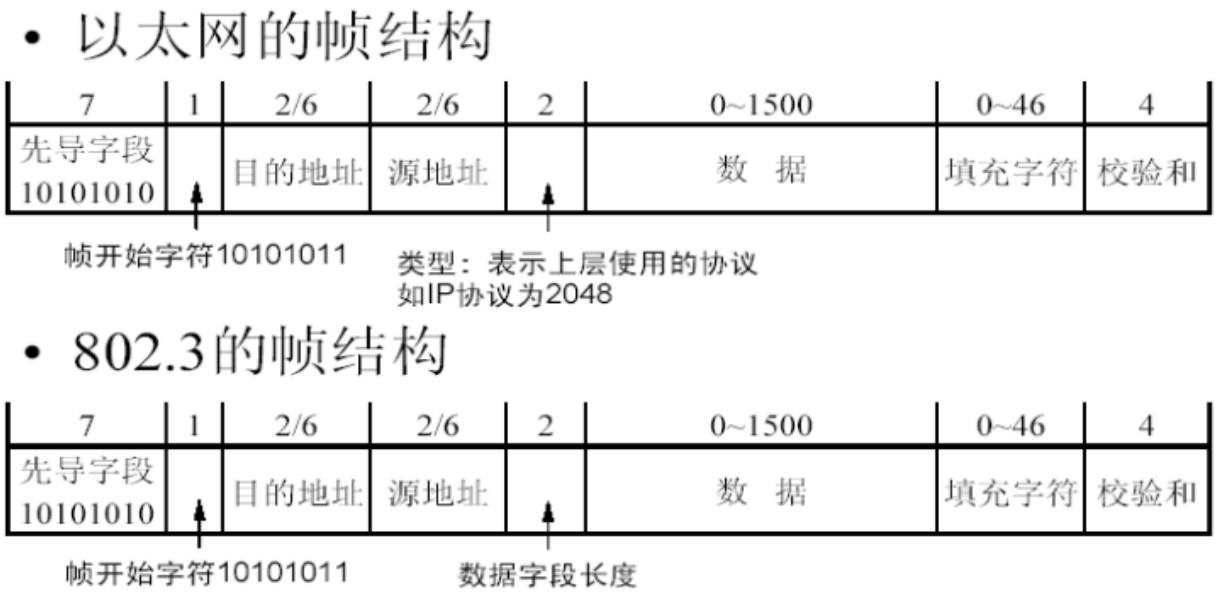
假如能有一种协议在低载荷时采用竞争法时延较短，在重载荷时采用无冲突法，使信道利用率较高

有限竞争协议：只要减少参与竞争的站点数，就可以增加站点获取信道的概率

• 将站点分组，信道分时隙 • 0 组成员只允许在 0 号时隙内竞争 • 1 组成员只允许在 1 号时隙内竞争 • ……• 问题是分几个组？每个组几个成员？

5. 以太网 MAC 子层，二元指数后退算法

a.



先导字段

• 7 个字节的 10101010，实际上下一个字符也是先导字段，只是最后的两位为 1，表示紧接着的是真正的 MAC 帧

• 7 个字节的 10101010 的曼切斯特编码将产生 10MHz 持续 5.6μs 的方波，周期为 0.1μs，可用于时钟同步两个 MAC 地址

• 目的地址和源地址都允许为 2 字节或 6 字节，但在 10M b/s 的基带以太网中是 6 字节

• 目的地址最高位为 0：普通地址 • 最高位为 1：多点发送 Multicast

• 目的地址全 1：广播发送 Broadcast • 次高位(第 46 位)区分局部或全局地址

• 在 6 个字节(共 48 位)的地址中，有 46 位用于地址的指定，即全局地址有 2 的 46 次方=7.03687x10 的 13 次方个

• 网卡地址是一个全局地址 ，如 44-45-53-54-00-00

b. 二进制指数后退算法

• 第一次冲突产生后，每个站点等待 0 或 1 个时隙后重新尝试发送

• 如果每个站点等待时隙数相同，它们将再次冲突。这一次他们会从 0,1,2,3 中随机挑选一个时隙数等待重发

• i 次冲突后，等待时隙数就从 0 到 2 的 i-1 次方中随机选出。

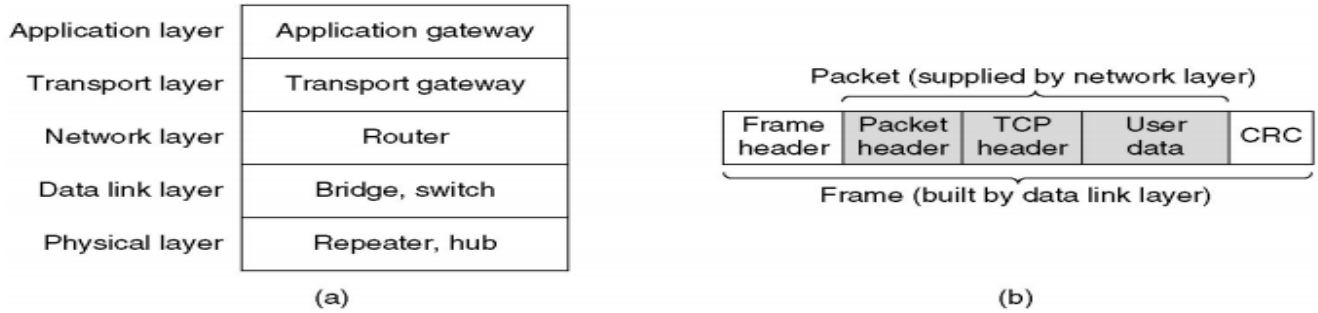
• 10 次冲突后，最大时隙数就固定在 1023 • 16 次冲突后，向计算机报告失败

为确保通信可靠，接收方必须计算校验和，如正确就向发送方送回一个确认帧。

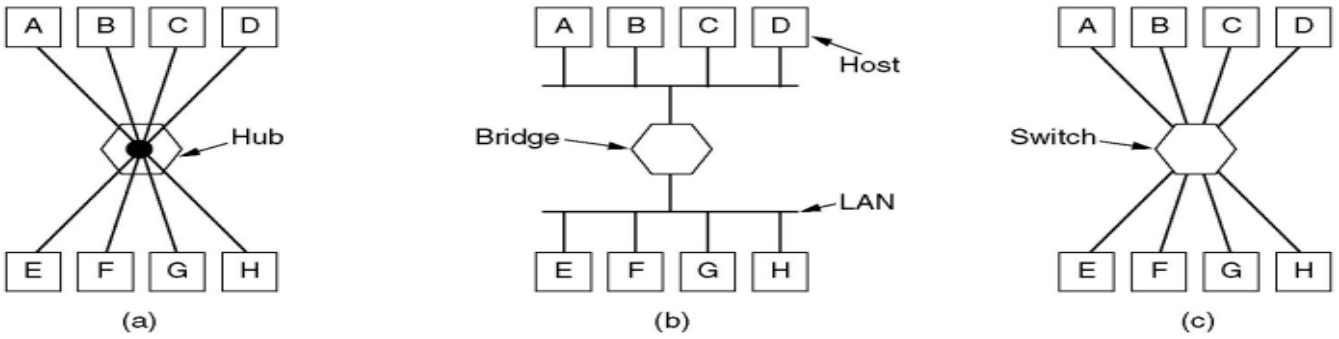
成功发送之后将第一个竞争时隙留给目的站，以便发送确认帧

6. 中继器，集线器，网桥，交换机，路由器，网关

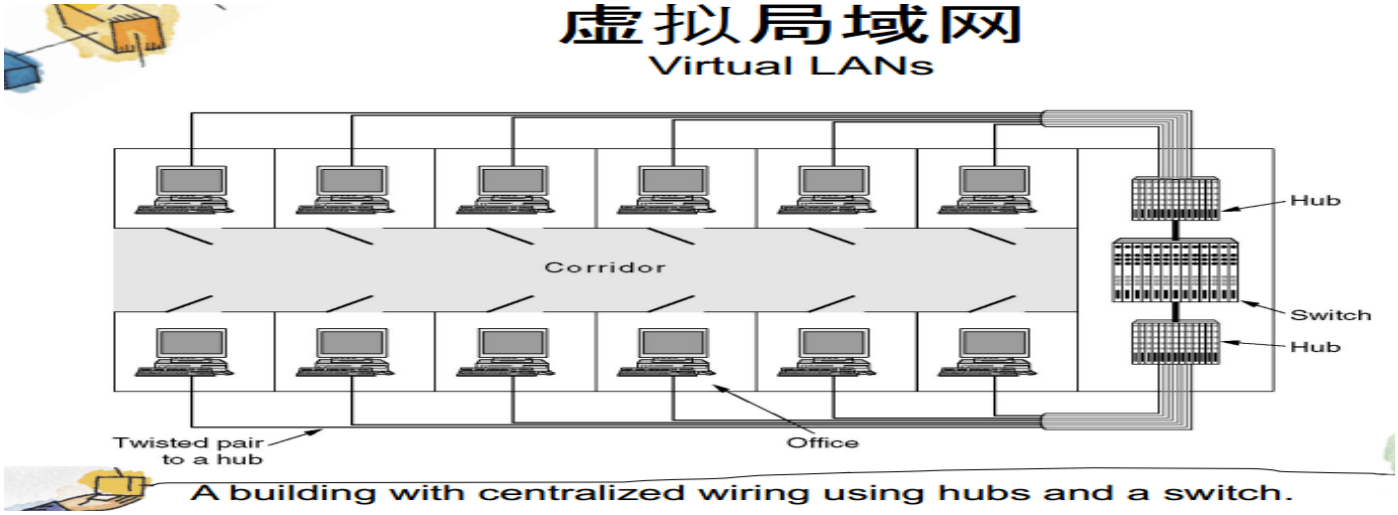
中继器 Repeaters,集线器 Hubs, 网桥 Bridges, 交换机 Switches, 路由器 Routers, 网关 Gateways



(a) Which device is in which layer.
(b) Frames, packets, and headers.



(a) A hub. (b) A bridge. (c) a switch.



五 网络层--数据链路层仅将数据帧从导线的一端到另一端，网络层则处理端到端数据的传输

1. 面向连接的服务和面向无连接的服务

复杂的纠错、重复、丢失等数据校验功能放在何处的问题。网络层(通讯子网)还是传输层(主机)

a. 面向连接的服务

- 以电信公司为代表 - 传送数据前需建立连接 - X.25 ATM

虚电路

- 虚电路的想法是避免对发送的每一个分组都必须进行路由选择
- 连接建立时选择一条路径 - 每个分组包含一个连接号 - 通信结束后链路撤消
- 虚电路号不可重复，中间路由器将记录此号 - 须提供有效手段清理被未正常释放的虚电路

b. 无连接的服务

- 以 Internet 委员会为代表 - IP 网

数据报子网

- 每个数据报包含全部的目的地址，自行寻找路径
- 发出的每个分组所选择的路由独立于其前面发出的路由 - 更健壮，更容易处理传送失败和拥塞

虚电路子网与数据报子网的比较

- 虚电路子网

- 通过路径选择后建立连接，通信后撤销连接 - 创建虚电路需要时间，不适合频繁连接场合，最好一旦建立后就用它几个月那种的。 - 数据到终点后毋需重新排序 - 每个分组不需带目的地址但带虚电路号 - 建立连接时可以提示所需的带宽和路由器容量，易于避免拥塞 - 一个路由器崩溃，则所有虚电路都丢弃，数据也丢失

- 数据报子网

- 每个分组分别选择最佳路径，健壮性较好 - 数据报到终点后需重新排序 - 差错控制和排序工作由协议高层(主机)完成 - 每个分组必须带目的地 - 一个路由器崩溃，仅此路由器上数据丢失

方式	数据报子网	虚电路子网
无连接的	IP之上的UDP	ATM上的IP
面向连接的	IP之上的TCP	ATM上的AAL1

AAL: ATM Adaptation Layer

2. Dijkstra 算法，扩散法

a. 最短路由计算

• 用来计算路由表项的软件把网络看成一张图，使用一种称为 Dijkstra 算法的方法来计算（从 PDF17 开始）

• 该算法从一个源点出发，计算沿最短路径到图中其他各点的距离，在计算最短路径的过程中构造下一站路由表。对每个路由表都必须用算法计算一次。为包交换机 P 计算路由表，就以对应于 P 的站点作为源点。

• Dijkstra 算法由于能用来计算各种意义的最短路径 shortest path）而得到广泛应用。特别是它不要求图中的边代表地理距离，它甚至允许每条边可被赋予一个非负值，称之为权（weight），并将两站点之间的距离定义为沿该两点间路径的权值之和。

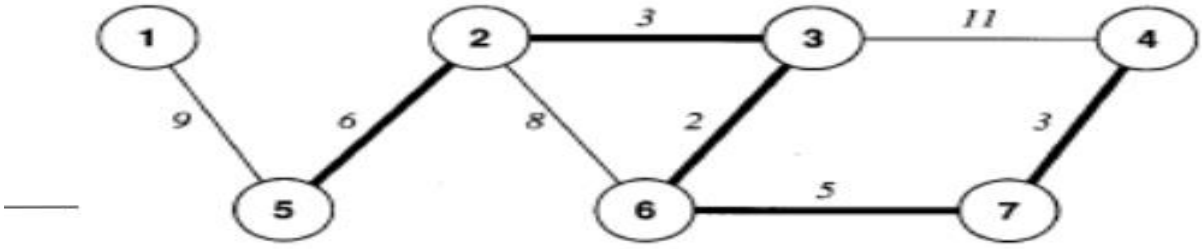
b. 扩散法:• 不计算路径，有路就走

• 如从 5 出发到 4

• 数据包从 5→1,2 2→3,6 3→6,4 6→3,7

7→4

• 要解决的问题：数据包重复到达某一节点，如 36



• 抑制重复分组的办法

- 在数据包头设一计数器,每经过一个站点自动减 1，当计数值为 0 时，丢弃数据包
- 在每个节点上建立登记表，则数据包再经过时丢弃

• 缺点:重复数据包多浪费带宽• 优点:可靠性高路径最短常用于军事 a

3. 距离矢量路由法

• 动态、分布式算法

• 每个路由器维护一张向量表，表中给出了到每个目的地已知的最佳距离和路线。通过与相邻路由器交换信息来更新表。

• 实现分布式算法的三要素

- The measurement process(测量) - The update protocol(更新邻接点距离矢量)
- The calculation(计算)

D-V 算法的工作原理

• 每个路由器用两个向量 Di 和 Si 来表示该点到网上所有节点的路径距离及其下一个节点

• 相邻路由器之间交换路径信息• 各节点根据路径信息更新路由表

4. 令牌桶

• 数据的输出必须持有令牌

• 令牌的产生是定时的，每隔 t 产生一个令牌，如令牌没有被及时使用，可以存在令牌桶内，令牌桶满则将被丢弃

• 当突发数据到达时，如令牌桶内有多个令牌，则突发数据可及时输出

令牌桶突发时间长度的计算

如： C为令牌桶的容量
ρ为令牌到达速率
M为最大的输出速率(数据到达速率> M)

则最大的突发时间S为

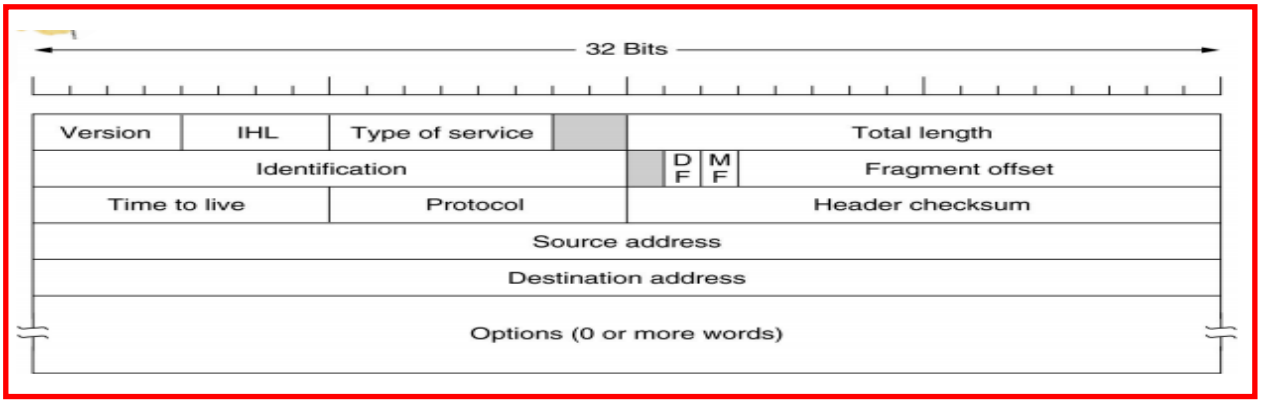
$C + \rho S = MS$　　即$S = C/(M - \rho)$

当： 令牌桶的容量$C = 250K$ byte
令牌到达速率$\rho = 2M$ byte/s
最大的输出速率$M = 25M$ byte/s时

如令牌桶已满则$S = 250K/(25M - 2M)$ (ms)
$= 250/23$ (ms) 约为11 ms

5. IPv4 头部，特殊 IP 地址

a. 头部：



• Version: IPV4　• IHL:头部长以 32 位字长为单位

• 服务类型• 总长:包括报头和数据报，最长 2 的 16 方 - 1

• 标识：分组号，用于数据报的分段重组• DF=1：不要分段; MF=1:段没结束

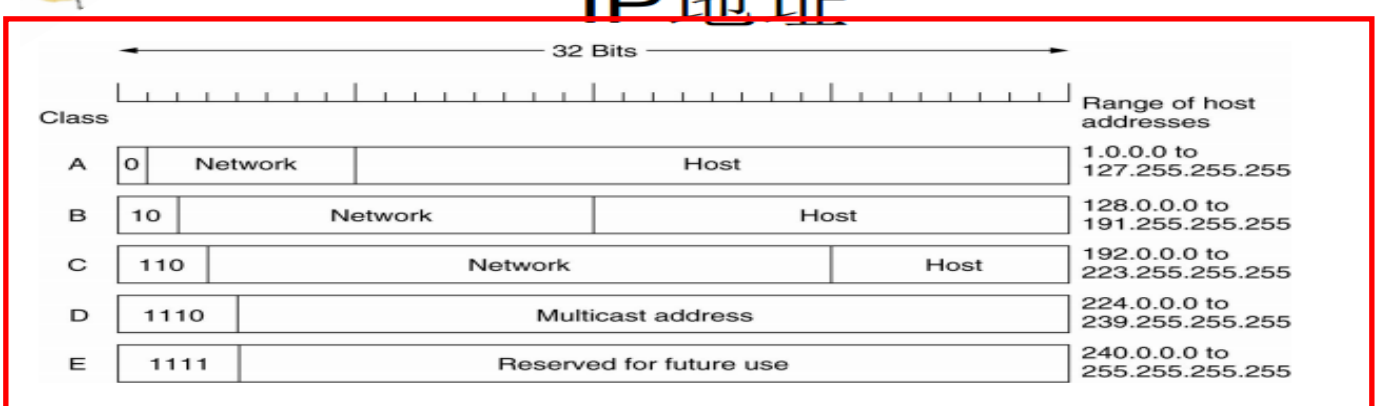
• 段偏移：分段在原数据报的位置，8 个字节为单位，13 位->8192

• 生存时间：经过路由器的个数为单位

• 协议:• 头部校验：头部 word 累加求补• 源地址• 目的地址

• 选项：允许后续版本引入更多信息

IP地址



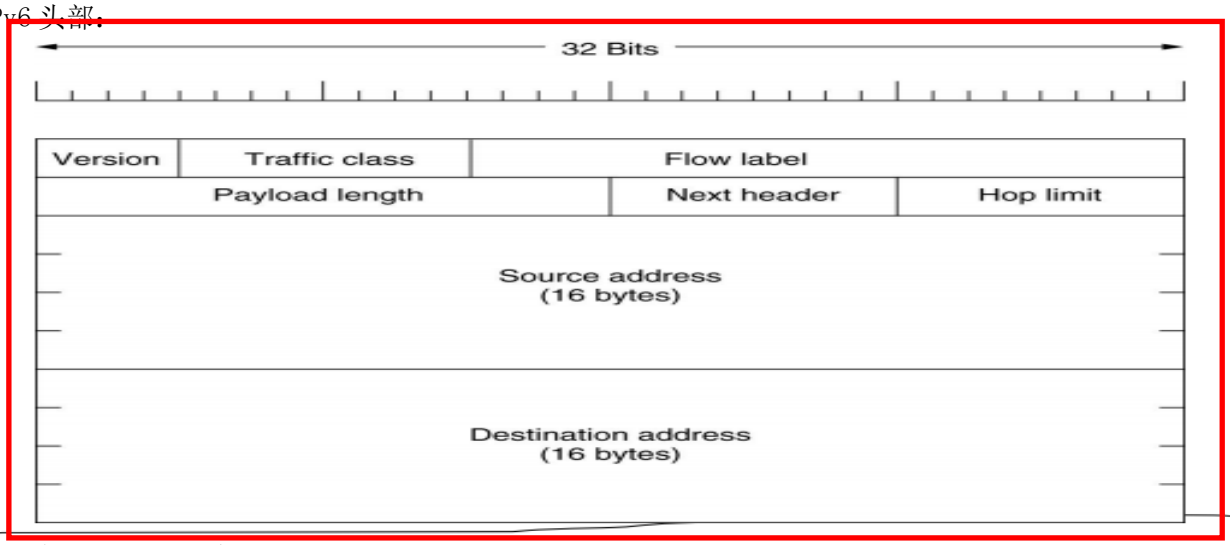
私有网络地址:

地址类别	地　　址
A类	10.0.0.0 - 10.255.255.255
B类	172.16.0.0 - 172.31.255.255
C类	192.168.0.0 - 192.168.255.255

6. IPv6 头部，与 v4 的对比（32 位与 128 位）

- IPv4 的不足
  - 地址基本耗尽，这是当前最棘手的问题 - 功能不足，缺少对多媒体信息传输的支持
  - 缺少对高速传输的支持 - 缺少对安全的支持 - 寻找路径的功能不强
- IPv6 的主要改进
  - 更大的地址空间 128 位
  - 灵活的首部格式，用一系列固定格式的扩展部取代了 IPv4 中可变长度的选项字段
  - 简化了协议，如取消了首部的校验和字段，分段只能在源端进行
  - 允许对网络资源的预分配，支持实时图象等要求保证一定的带宽和时延的应用
  - 允许协议继续演变增加新的功能

IPv6 头部:



- 版本 (0-3 位)• 流量类别
- 流标签 ：把在时间上敏感的一串报文，打上同一个标记路由，可优先通过
- 负载长度 16 位：除基本报头以外的长度• 下一个头部：8 位，指出下一扩展头部是什么类型
- 驿站限制： 8 位，跳数限制，类似 IPv4 的生存时间• 源、目的地址：128 位，
  - 如按 IPv6 的 128 位地址均匀分配意味着地球上每平方米平均分配的地址数有 7 x 10 的 25 方个扩展头部
- 6 种扩展头

– 逐跳选项 – 目标选项 – 路由 – 分段 – 认证 – 加密的安全净荷

• IP 协议只负责传送 IP 数据包，无法监视和控制网络中出现的一些问题，这些工作由 Internet 的控制协议来完成 – ICMP – ARP

7. ARP 协议

地址解析协议 ARP

• 协议地址 – 软件提供的抽象地址，如 IP 地址。它使整个互联网看成一个网络，但真正的物理网络并不能通过 IP 地址来定位机器

• 物理地址 – 硬件地址，如 MAC 地址协议

• 地址解析 – 协议地址和物理地址之间的转换，如 IP 地址和 MAC 地址之间的转换

ARP 工作原理

• 一个 ARP 请求消息是一个数据帧，其中包含发送站的硬件地址和协议地址，以及目的地址的 IP 地址，并把此数据帧在本物理网络内广播

• 暂存 ARP 应答于 Cache 或内存中，以后即可查表不必再发询问报文，以减少网络的通信量

• 从消息中取出发送方的协议地址和硬件地址，更新 cache 中已有的信息

• 检查消息是请求还是应答，若是应答则接收，若是请求，检查是否为发送给本站的，如是则发应答消息

8. ICMP,ping,tracert

a. ICMP—Internet 消息控制协议

类型	类型域	ICMP报文类型
差错报文	3	目的站点不可达
	11	数据报超时
	12	数据报参数错
控制报文	4	源抑制
	5	重定向
请求/应答报文	8	回应请求
	0	回应应答
	13	时间戳请求
	14	时间戳应答
	17	地址掩码请求
	18	地址掩码应答

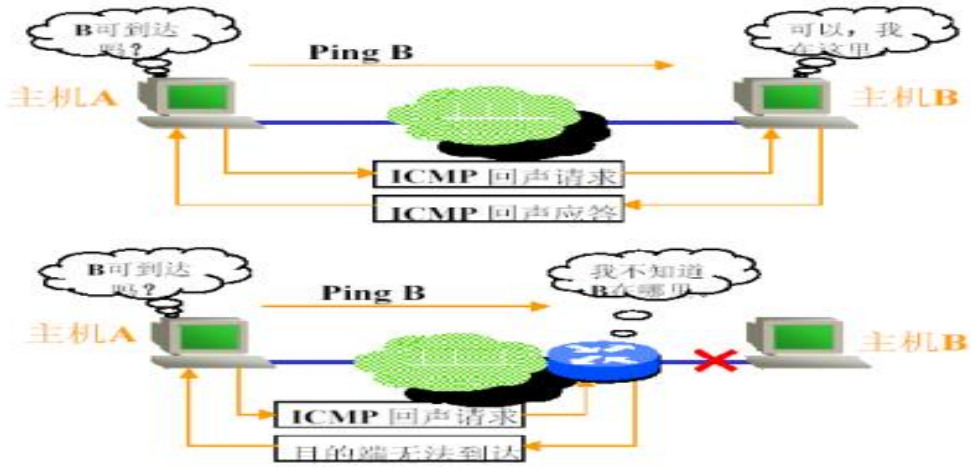
• IP 协议提供的是尽力而为的通信服务

• ICMP 提供了一种把通信服务中的差错向源站点报告的机制

可以用于：

b. 测试报文的可达性 ping

利用回显和回显应答消息可以用来判断一个指定的目标是否可达，是否还活着。目标主机接收到回显消息以后，应该立即送回一个回显应答消息，这些消息可以被 ping 工具探测 Internet 上是否存在某一个特定的主机

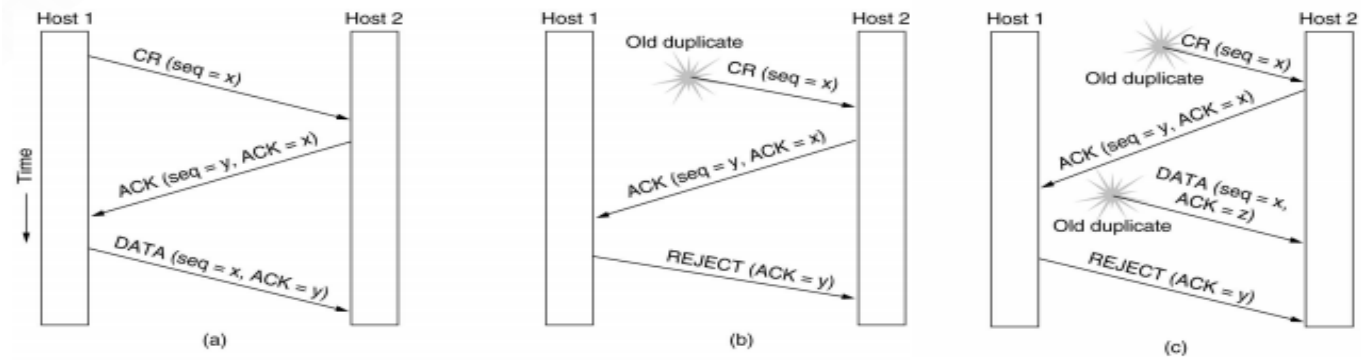


c. 路由跟踪命令 tracert

- tracert 过程是通过 ICMP 数据报超时报文来得到一张途经路由器列表的
- 源主机向目的主机发一个 IP 报文并置 ttl(生存期)为 1，到达第一个路由器时 ttl 减 1 为 0，则该路由器回发一个 ICMP 数据报超时报文，源主机取出路由器的 IP 地址即为途经的第一个路由端口地址
- 接着源主机再向目的主机发第二个 IP 报文，并置 ttl 为 2，然后再发第三个、第四个 IP 数据报……直至到达目的主机

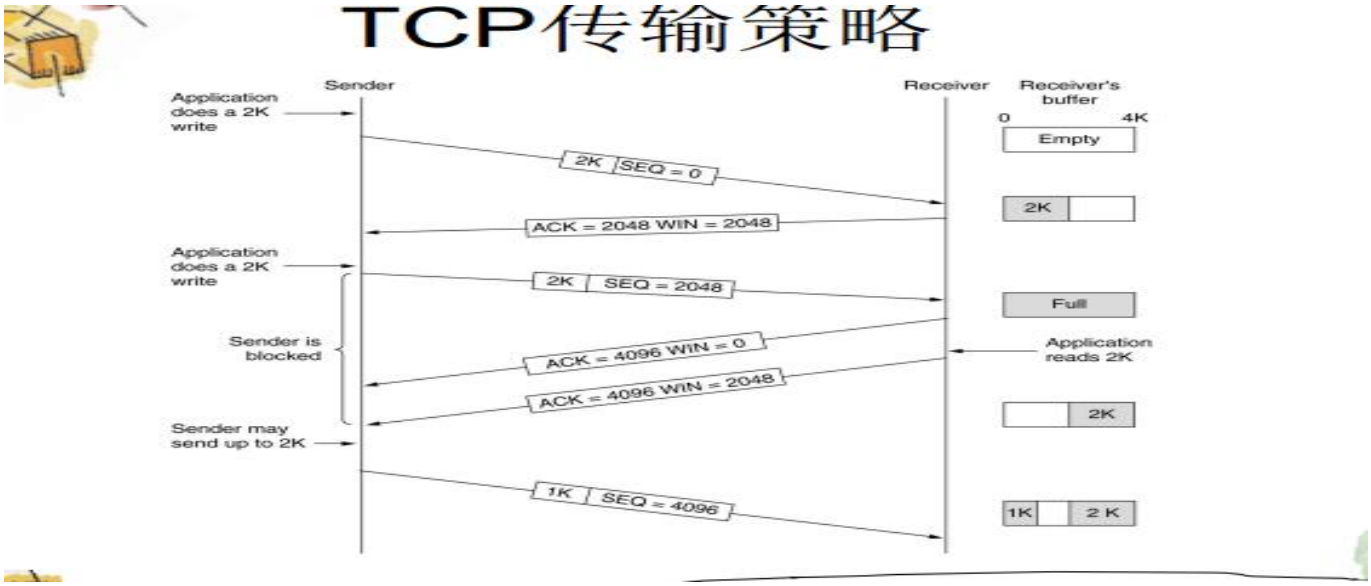
六 传输层--传输层是整个协议层次的核心，它提供端到端的可靠数据传输

1. 三次握手



CR=CONNECTION REQUEST. (a) 正常的连接建立过程(b) 过期的 CONNECTION REQUEST 突然出现. (c) 过期的 CONNECTION REQUEST 和过期的 ACK 都出现

2. 滑动窗口，TCP 的零窗口通告及处理



TCP 滑动窗口策略：在确认帧中通告窗口大小

零窗口通告

• 发送方收到一个零窗口通告时必须停止发送，直到接收方重新通告一个正的窗口，

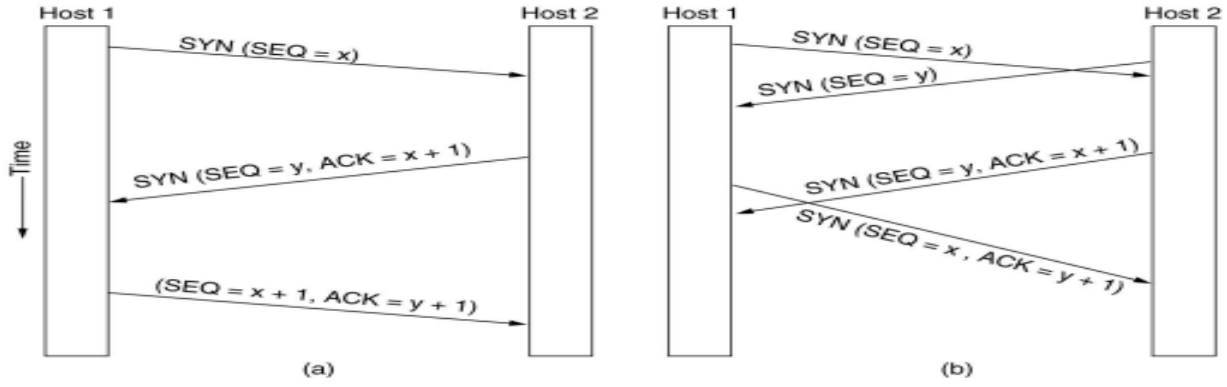
但有两种情况可以除外：

- 发送紧急数据，如允许用户终止 kill 远端机上运行进程
- 发送方可以发送一个字节的数据段通知对方重新声明它希望接收的下一字节及窗口大小，以防止窗口声明丢失而导致的死锁

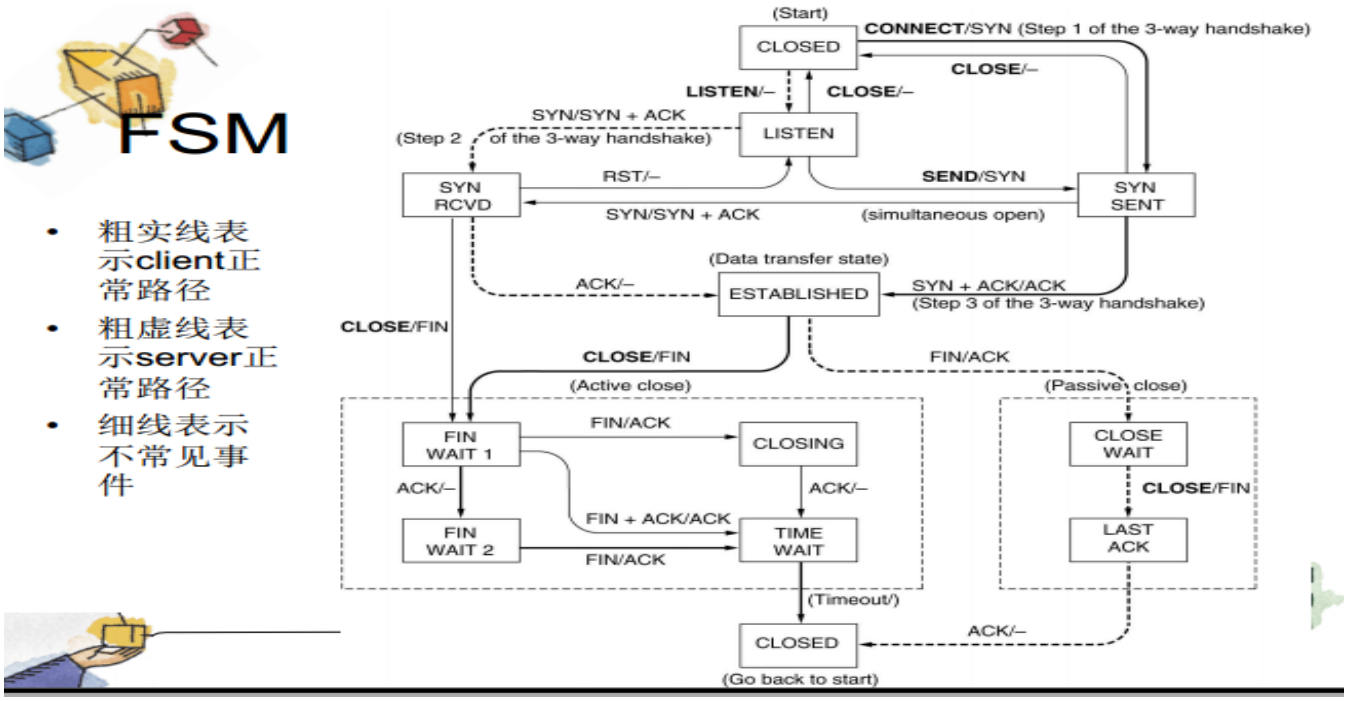
TCP 提供的服务

• 面向连接(Connection Orientation)• 端到端的服务(End-to-End Communication)
• 完全可靠性服务(Complete Reliability)• 全双工服务(Full Duplex Communication)
• 流接口(Stream Interface)• 可靠的连接建立(Reliable Connection Startup)
• 完美的连接终止(Graceful Connection Shutdown)

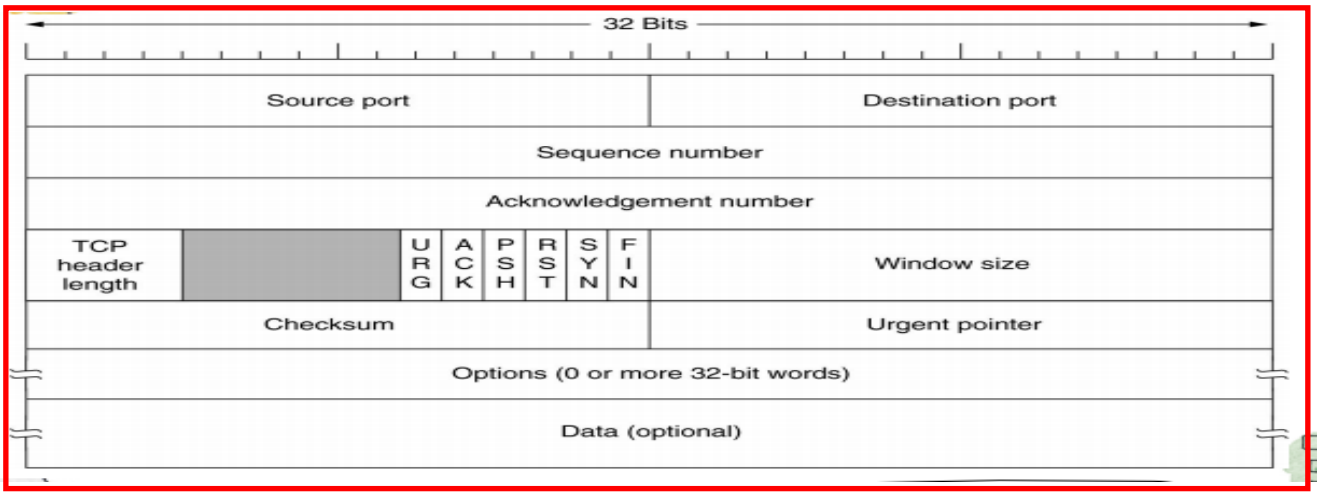
3. TCP 的连接建立过程和链接释放过程(FSM)



• A) 采用三次握手建立连接
• B) 呼叫碰撞的情况，由于是根据端口建立连接表格，因此只会建立一条连接



4. TCP 段头部



•端口：每个端口对应一个应用程序•序号：发送的字节序号
•确认号：接收到的字节序号•段头长度：段头中包含多少个 32 位字
•保留： 以备扩展之用•窗口：接收方窗口大小

URG	ACK	PSH	RST	SYN	FIN

• URG：紧急指针有效• ACK：确认号有效• PSH：接收方请求数据一到立即送往应用程序
• RST：复位由于主机崩溃或其他原因而出现的错误的连接• SYN：用于建立连接• FIN：用于端开连接

• 校验和校验包括头部、数据和伪段头
• 伪段头包括 IP 地址，06H，TCP 段长

5. TCP 的拥塞控制

• 数据的丢失有两种情况

– 接收方的容量太小(接收缓冲区) – 网络的容量太小(网络带宽)

• 主机如何知道拥塞

– 收到 ICMP 的源抑制报文 – 因报文丢失引起的超时

Internet 中两个潜在问题：网络的容量和接收方的容量

拥塞窗口

• 接收窗口表示接收缓冲区的容量

• 拥塞窗口表示网络的容量

• 接收窗口和拥塞窗口的大小两者取其小

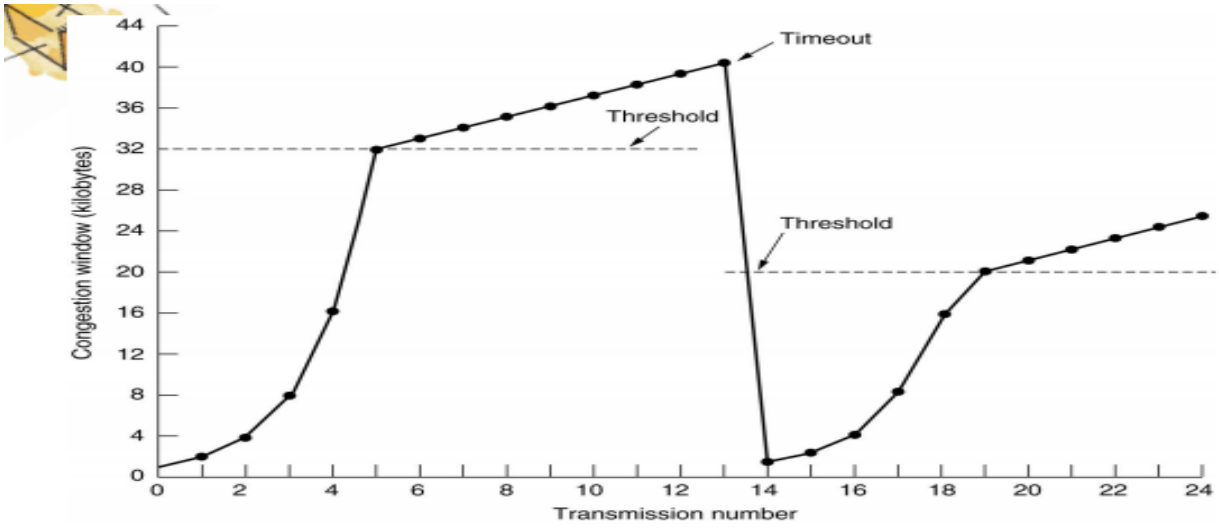
• 连接建立时，发送方将拥塞窗口的初始大小设置为最大的数据包长度，并随后发一个最大长度的数据包

• 如该数据包在定时器超时前得到了确认，发送方在原来的拥塞窗口的基础上再增加一倍长度发送两个数据包，如两个数据包都得到了确认则再增加一倍长度，直到数据传输超时或到达接收方的窗口大小为止

• 除接收窗口和拥塞窗口外，拥塞控制时还需指定一个临界值

• 临界值的初始值为 64K，如果发生数据传输超时将临界值设为当前拥塞窗口的 1/2，并使拥塞窗口恢复到最大的数据段长度，成功的传输使拥塞窗口按指数增加成倍直到到达临界值

• 以后按线性增加按最大的数据段长度



6. TCP 中的自适应超时定时器，持续定时器

a. 持续定时器

• 如接收方向发送方发出一个窗口为 0 的确认，当接收方的上层处理了一部分缓冲区数据后接收方更新窗口大小并向发送方再发一个确认分组，包括新的窗口公告，但该分组丢失，此时双方将相互等待出现死锁

• 为防止死锁，当持续定时器超时发送方将向接收方发一探测报文仅一字节，接收方的应答报文将避免相互等待

b. 自适应超时定时器

• TCP 在发送一个数据段的同时启动一个数据重发定时器，如果在定时器超时前该数据段被确认，则关闭该定时器

• 如果在确认到达之前定时器超时则需要重发该数据段，并且该定时器重新开始计时；

• 问题是超时间隔应该设为多长?

7. TCP 的自适应重发

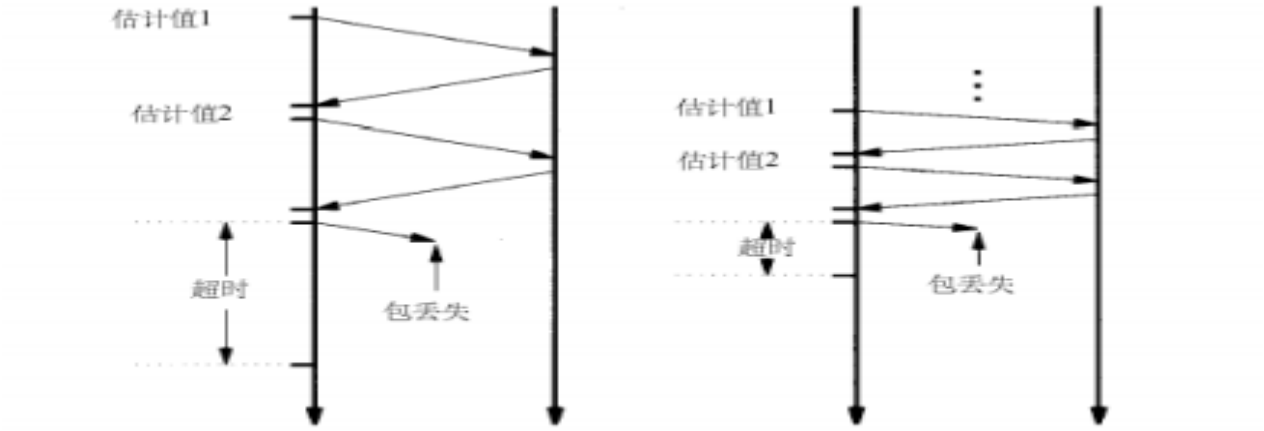


图22-3 两个有不同往返延迟的连接上的超时和重发。TCP通过使用往返估计来计算重发定时器以优化吞吐率

- 计算新的往返时间估计值

– TCP 不使用固定的重发定时器，而是根据对网络性能的不断测定，主要是对远程的确认报文作延迟分析，不断调整超时间隔的动态算法，使用一种称为自适应的重发定时器(RTT Round-Trip Timer)

– $RTT = \alpha RTT_0 + (1-\alpha)M_0$

– M 为最近一次成功的确认所需的时间
– RTTO 指上一次的 RTT 值
– α 修正因子，一般为 7/8

- 超时限制=ß RTT
– 偏差值方法在最初的程序实现中 ß 总为 2
– 但经验表明常量是很不灵活的，因为当发生变化时它便不能很好地适应偏差值

- 偏差值方法

– $D = \alpha D_0 + (1-\alpha)|RTT_0 - M_0|$
– 超时值= RTT + 4*D

UDP数据报格式

0	8	16	31
UDP源端口		UDP目的端口	
UDP长度		UDP校验	
数据（可选）			

- UDP源端口：UDP端口号，当不需要返回数据时源端口域置0
- UDP目的端口：UDP端口号
- UDP长度：整个数据段的长度包括头部和数据部分，以字节计，最小值为8，仅头部长度
- UDP校验和：可选域全0为未选，全1表示校验和为0