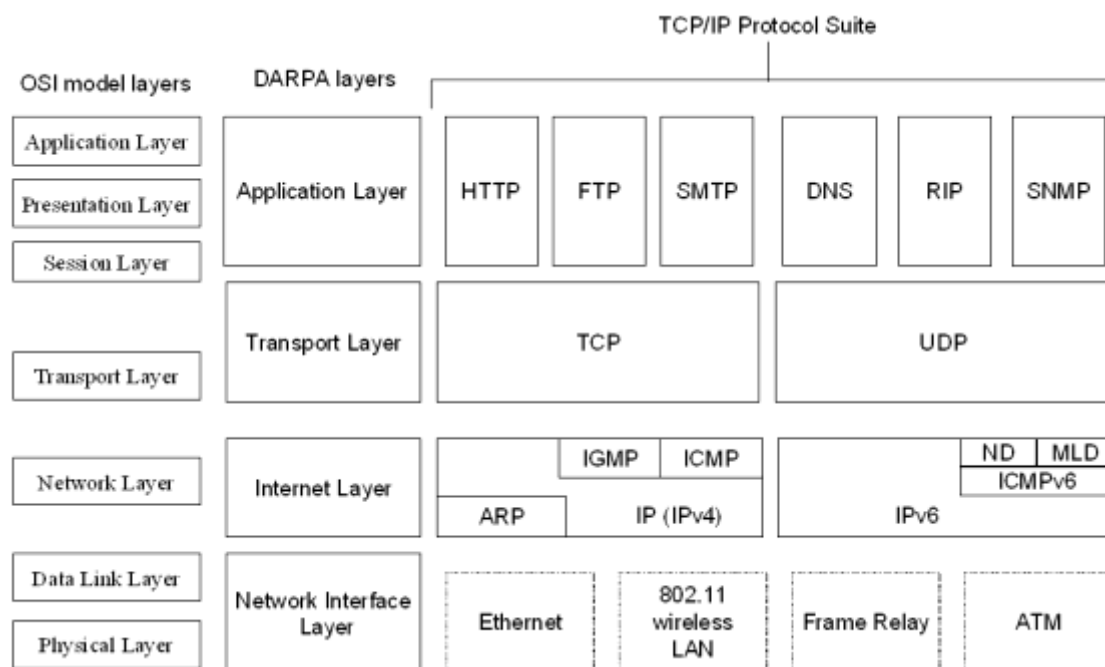


# 协议栈



## 物理层

定义了比特作为信号在信道上发送时的电气、时序和其他接口。

## 数据链路层

基本服务是将数据包通过单一通信链路从一个节点移动到相邻节点。两台相邻机器实现可靠有效的完整信息块（帧）通信的算法，不像物理层只关注单个比特传输。相邻指两台机器通过一个通信信道链接起来，通信信道概念上就像一条线路，这使得信道上传递的比特顺序和发送顺序一致。

**介质访问控制子层**：用来确定多路访问信道下一个使用者的协议。在 LAN 中，MAC 子层特别重要，特别是无线局域网，因为无线本质上是广播信道。

**可能**提供的服务包括：成帧；链路接入；可靠交付（许多有线链路层协议不提供可靠交付）；流量控制；差错检测；差错纠正；半双工和全双工

## 网络层

如何将源端数据包一路发送到接收方。这中间可能经历了许多跳 hop，这和数据链路层不同，他只关注将帧从线路一边传送到另一边。

网络层必须知道网络拓扑结构（即所有路由器和链路的集合），并从中选出适当的路径，即使是一个大型网络。

**转发和选路的区别**：选路算法

## 运输层

网络层使用数据报或虚拟电路技术为端到端通信提供了数据包交付服务。

**运输层**在网络层提供的服务之上，把数据传递服务从两台计算机之间扩展到了计算机上的进程之间，并且服务所需的可靠性程度独立于当前使用的网络。

解决“两个实体如何在一种会丢失或者损坏数据的介质上可靠的通信”；

“控制运输层实体的传输速率以避免网络中的拥塞，或从拥塞中恢复过来”。

## 应用层

应用程序所使用的协议。

以太网协议头

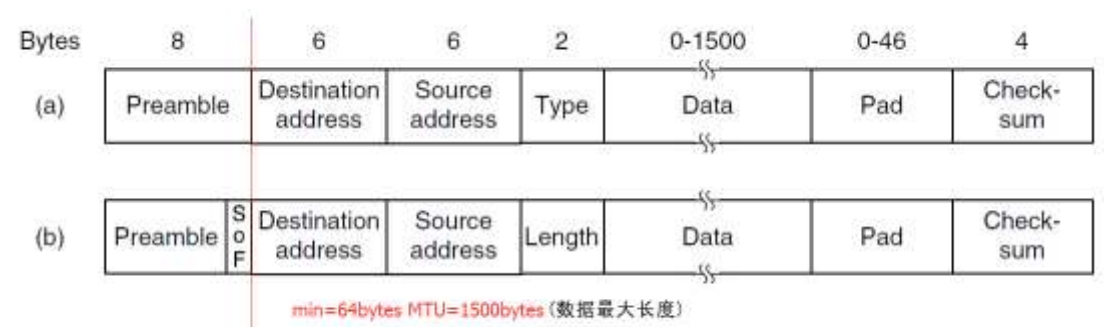


Figure 4-14. Frame formats. (a) Ethernet (DIX). (b) IEEE 802.3.

IP 协议头

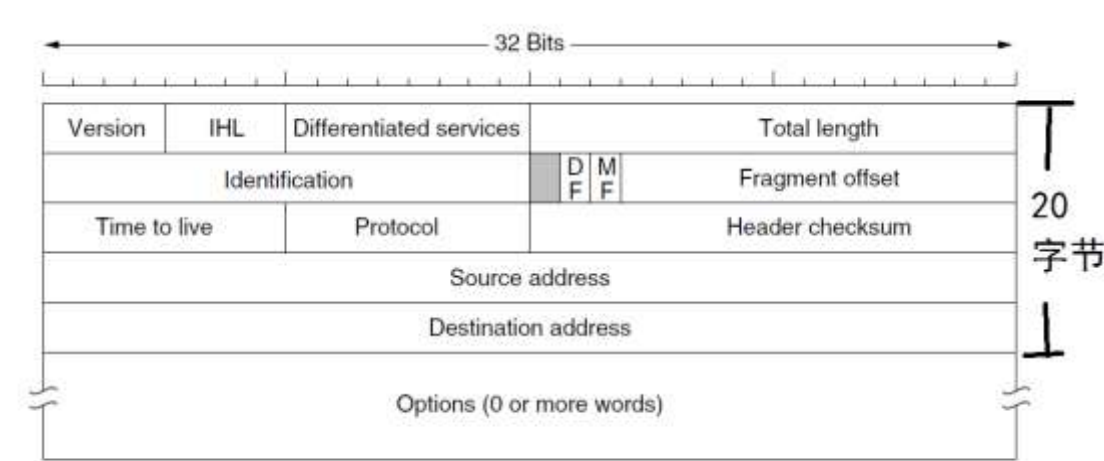


Figure 5-46. The IPv4 (Internet Protocol) header.

TCP 协议头

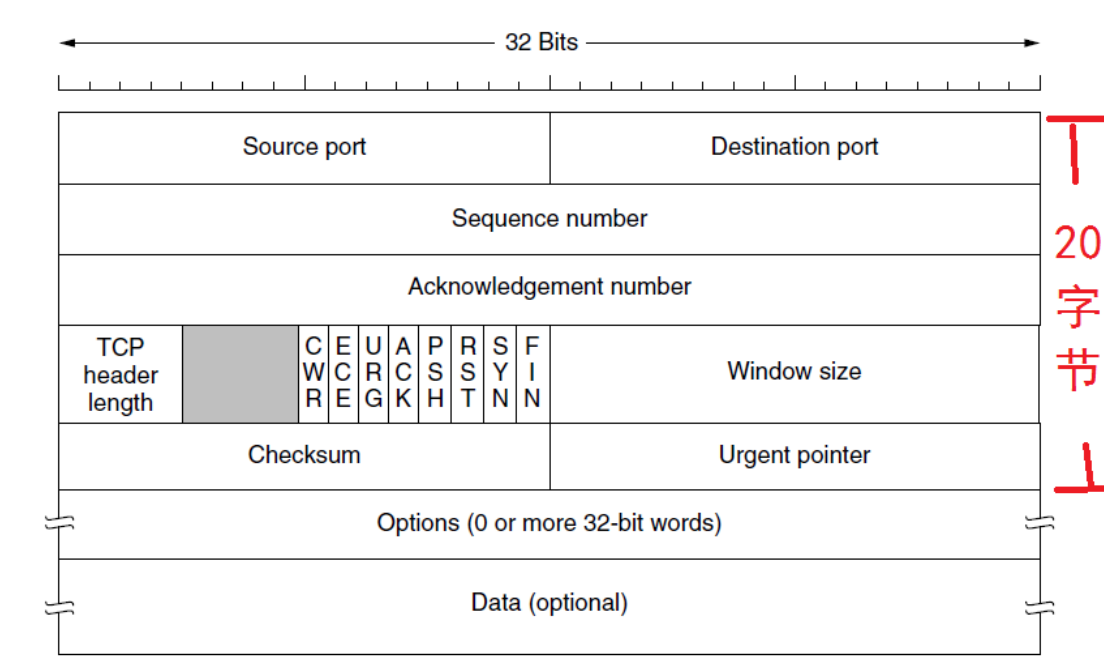
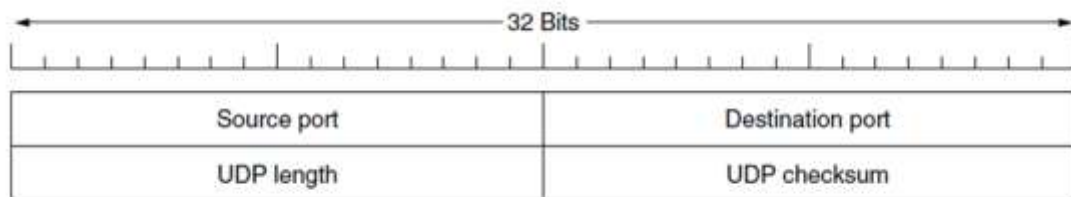


Figure 6-36. The TCP header.



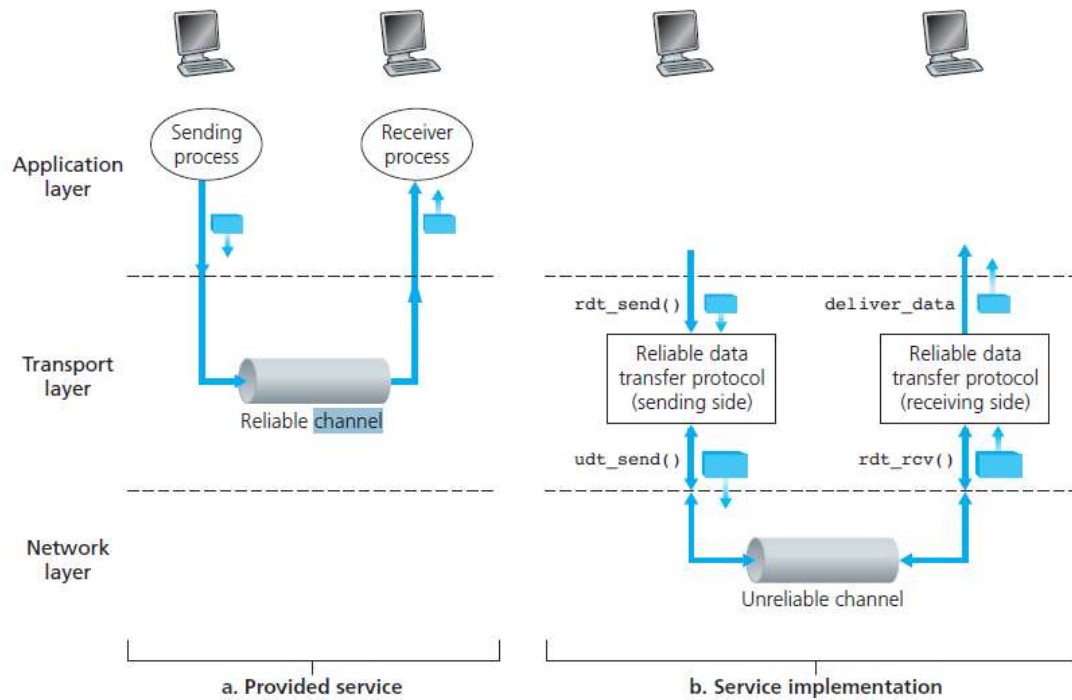
8字节头部

Figure 6-27. The UDP header.



# 可靠数据传输的原理

在一个不可靠的信道上实现可靠的数据传输，也就是没有比特受到损坏或者丢失，并且数据按照其发送数据进行传输

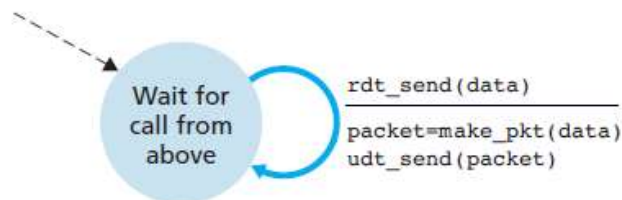


Key:

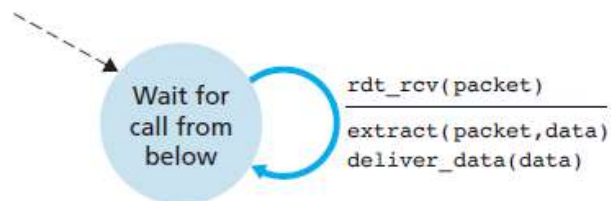
■ Data   ■ Packet

## 1. 在完全可靠信道上的可靠数据传输

假定接收方接受速率和发送方发送速率一样快



a. rdt1.0: sending side



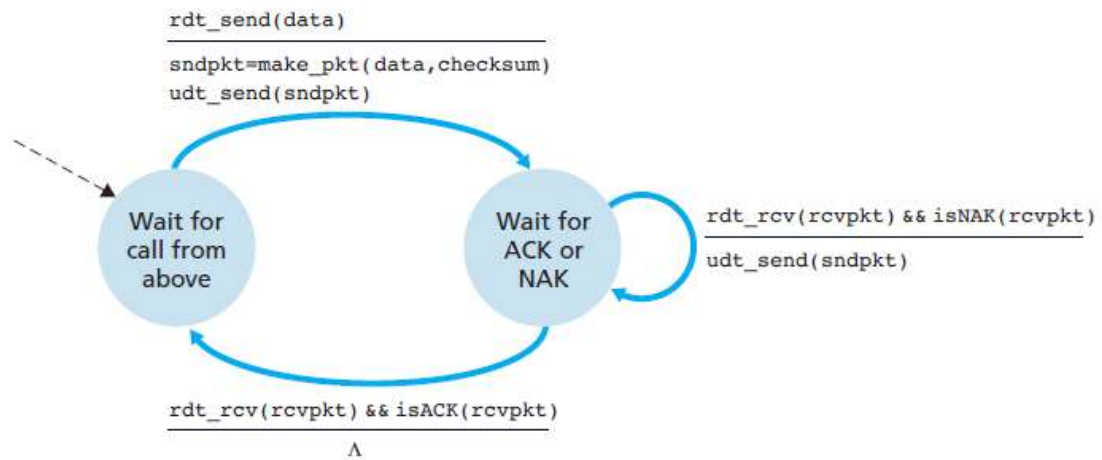
b. rdt1.0: receiving side

**Figure 3.9** ♦ rdt1.0 – A protocol for a completely reliable channel

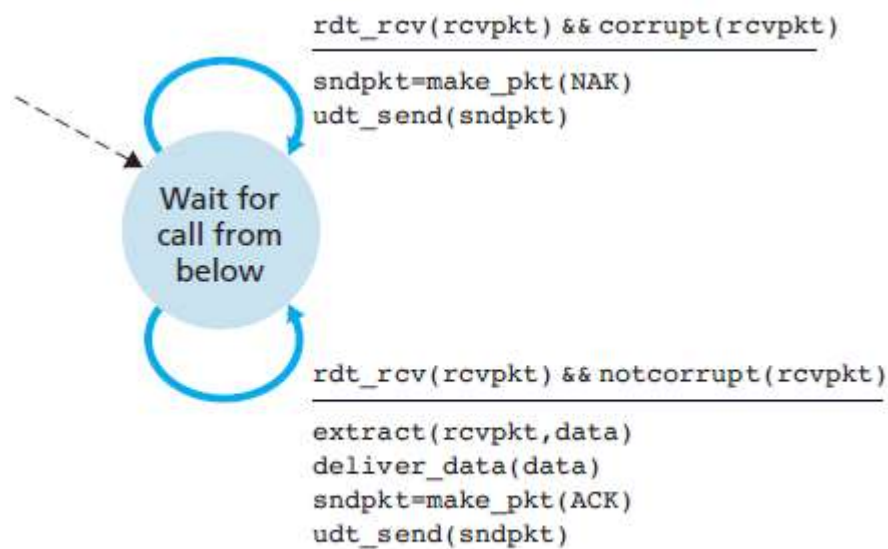
## 2. 具有比特差错的可靠数据传输

假定没有包丢失，所有传输的分组将按照其发送的顺序被接

Rdt2.0 使用了 差错检测，肯定确认与否定确认



a. rdt2.0: sending side



b. rdt2.0: receiving side

Rdt2.1 : rdt2.0 中 ACK 或 NCK 有可能受损  
使用了差错检测，肯定确认与否定确认，序列号

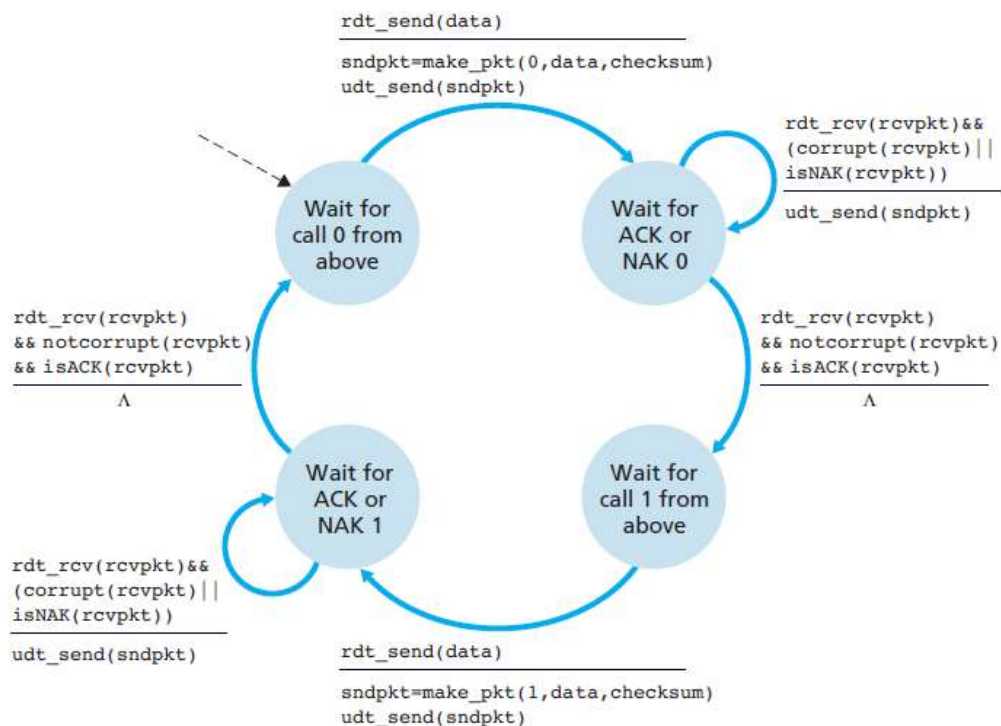


Figure 3.11 ♦ rdt2.1 sender

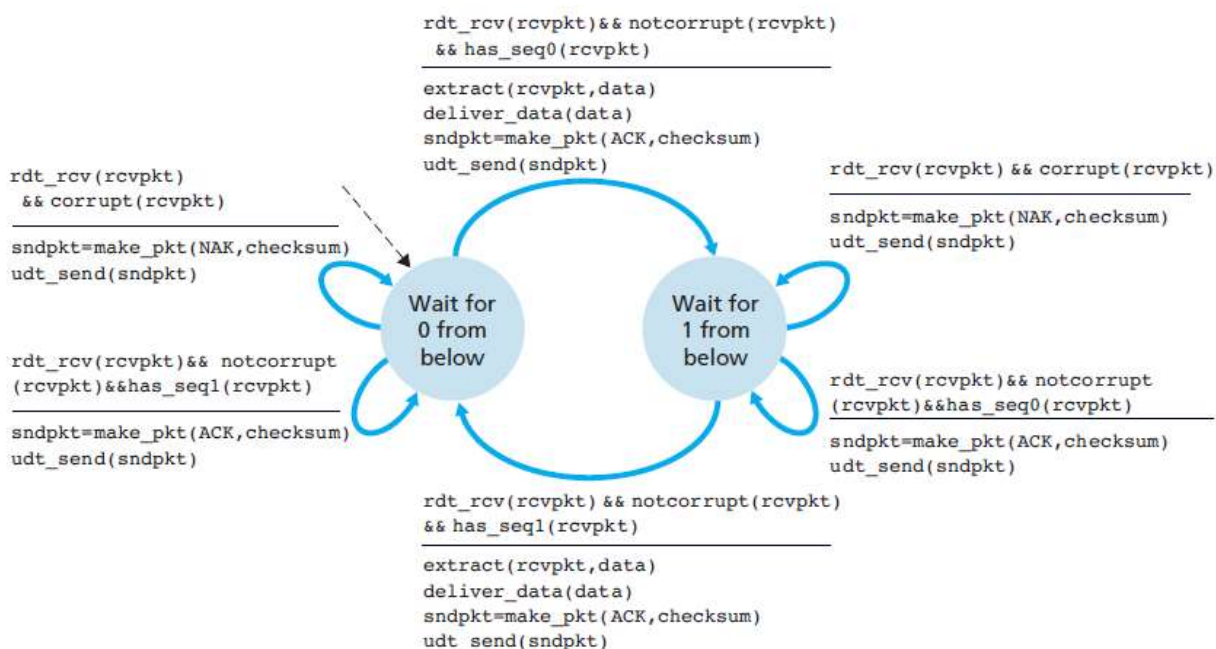


Figure 3.12 ♦ rdt2.1 receiver



Rdt2.2 基础上，不使用 NCK

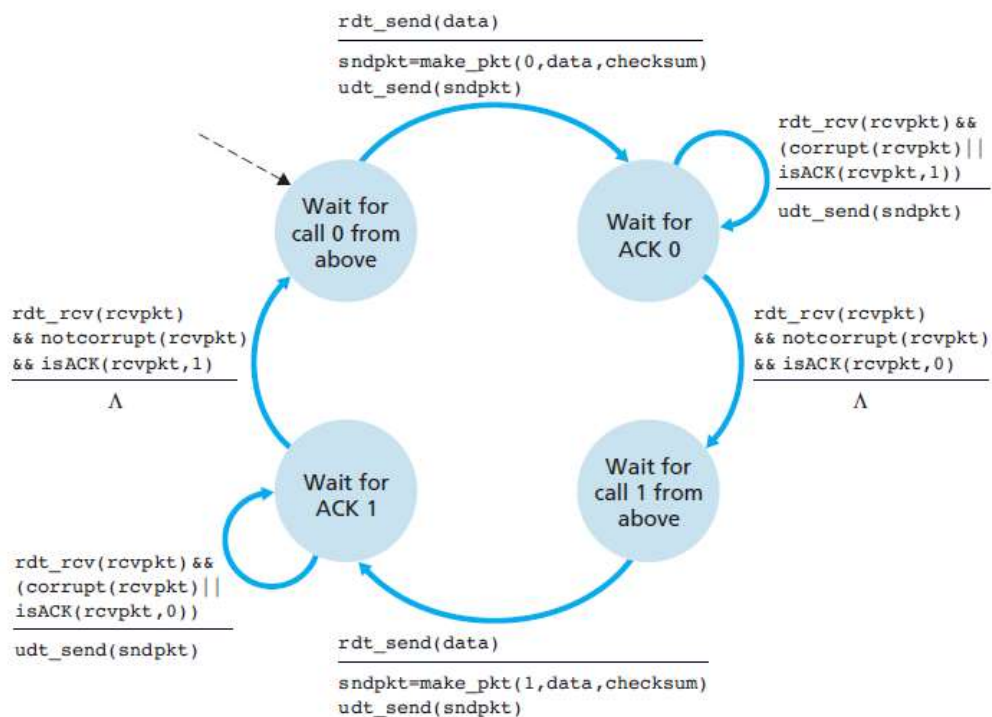


Figure 3.13 ♦ rdt2.2 sender

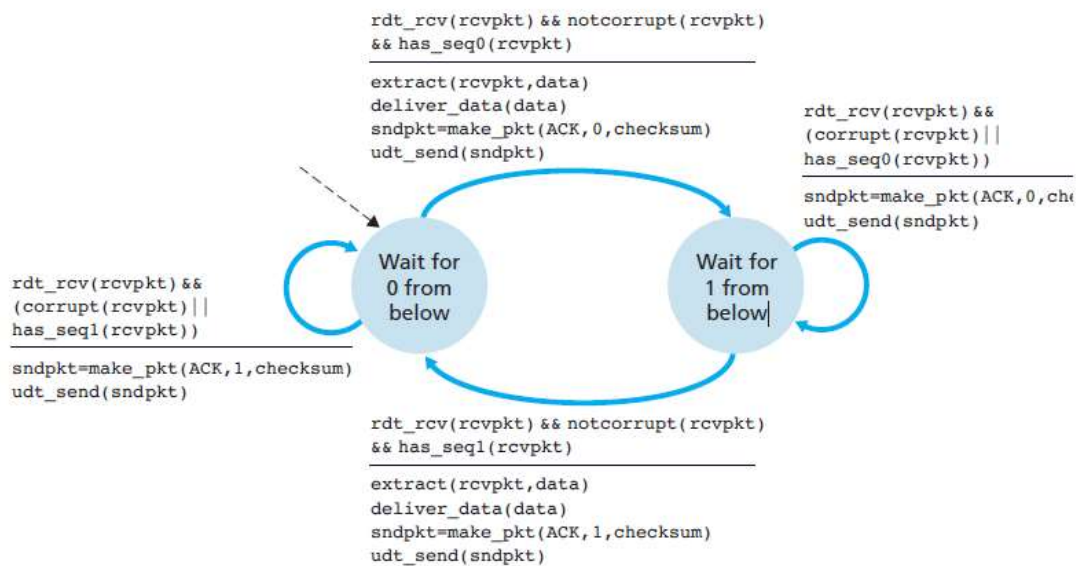


Figure 3.14 ♦ rdt2.2 receiver

3. 具有比特差错的丢包信道上的可靠数据传输  
停等协议 rdt2.2 基础上添加计时器处理丢包

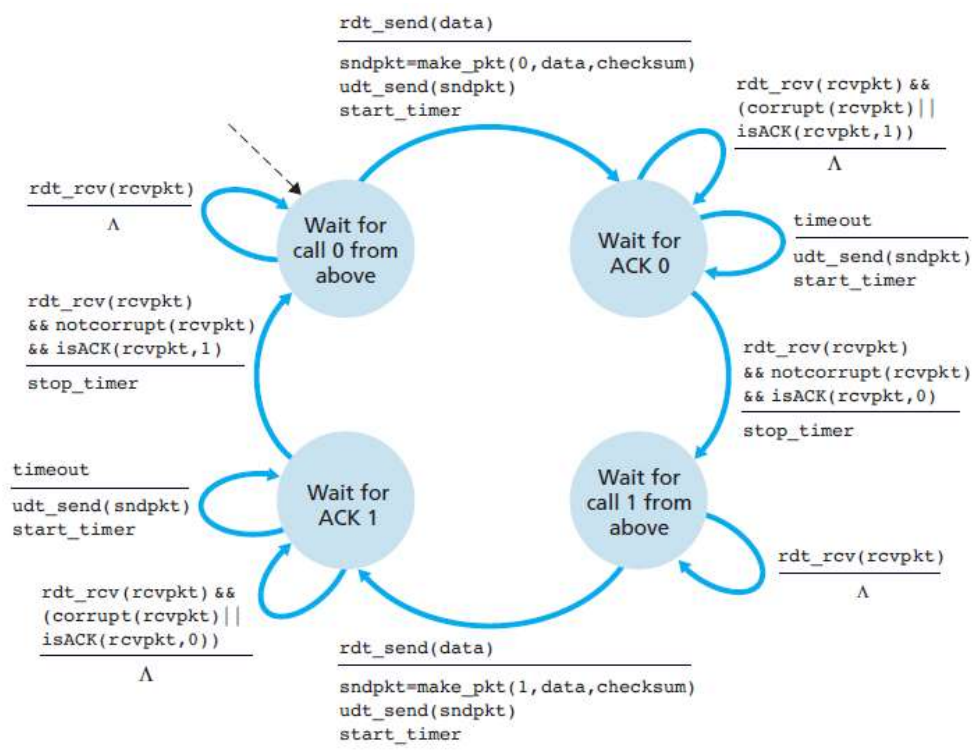


Figure 3.15 ♦ rdt3.0 sender

表3-1 可靠数据传输机制及其用途的总结

机 制	用途和说明
校验和	用于检测在一个传输分组中的比特错误
定时器	用于检测超时/重传一个分组，可能因为该分组（或其ACK）在信道中丢失了。由于当一个分组延时但未丢失（过早超时），或当一个分组已被接收方收到但从接收方到发送方的ACK丢失时，可能产生超时事件，所以接收方可能会收到一个分组的多个冗余拷贝
序号	用于为从发送方流向接收方的数据分组按顺序编号。所接收分组的序号间的空隙可使该接收方检测出丢失的分组。具有相同序号的分组可使接收方检测出一个分组的冗余拷贝
确认	接收方用于告知发送方一个分组或一组分组已被正确地接收到了。确认报文通常携带着被确认的分组或多个分组的序号。确认可以是逐个的或累积的，这取决于协议
否定确认	接收方用于告知发送方某个分组未被正确地接收。否定确认报文通常携带着未被正确接收的分组的序号
窗口、流水线	发送方也许被限制仅发送那些序号落在一个指定范围内的分组。通过允许一次发送多个分组但未被确认，发送方的利用率可在停等操作模式的基础上得到增加。我们很快将会看到，窗口长度可根据接收方接收和缓存报文的能力或网络中的拥塞程度这两种情况之一或全部来进行设置



## GBN(后退 N)

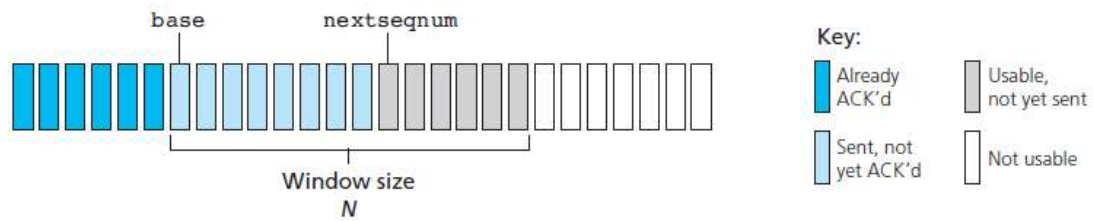


Figure 3.19 ♦ Sender's view of sequence numbers in Go-Back-N

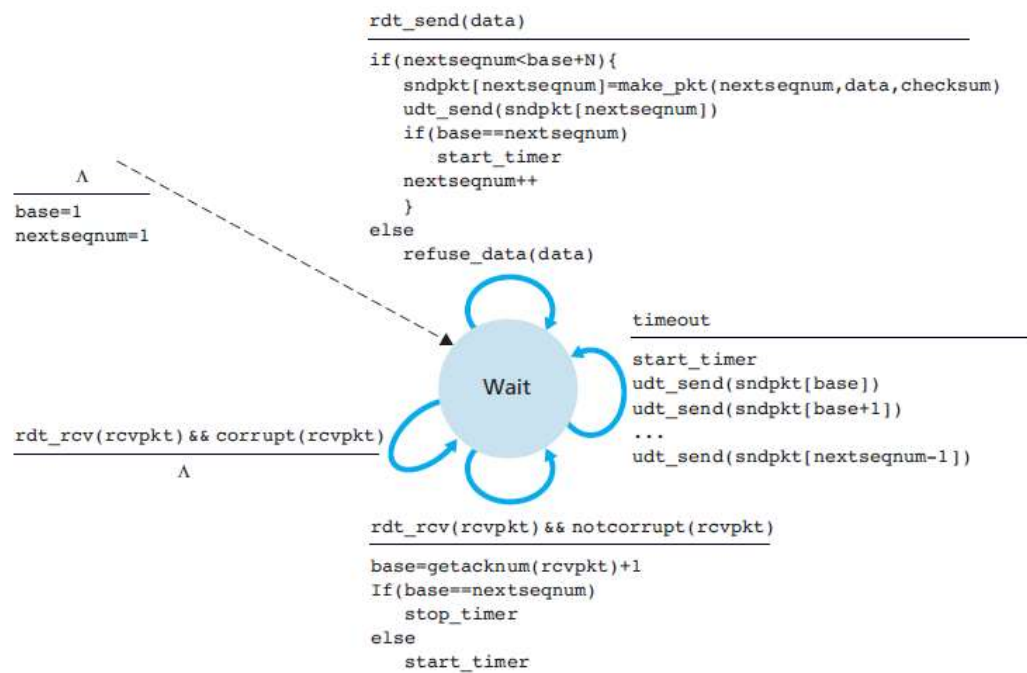


Figure 3.20 ♦ Extended FSM description of GBN sender

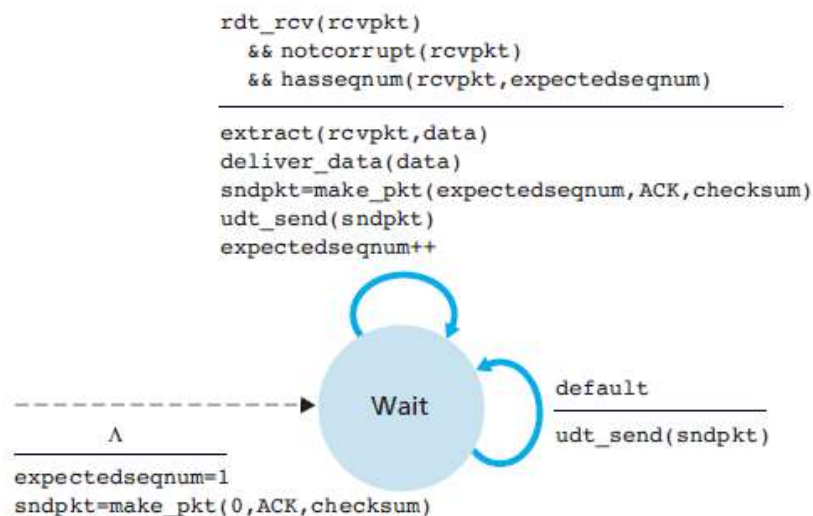


Figure 3.21 ♦ Extended FSM description of GBN receiver

## SR (选择性重复)

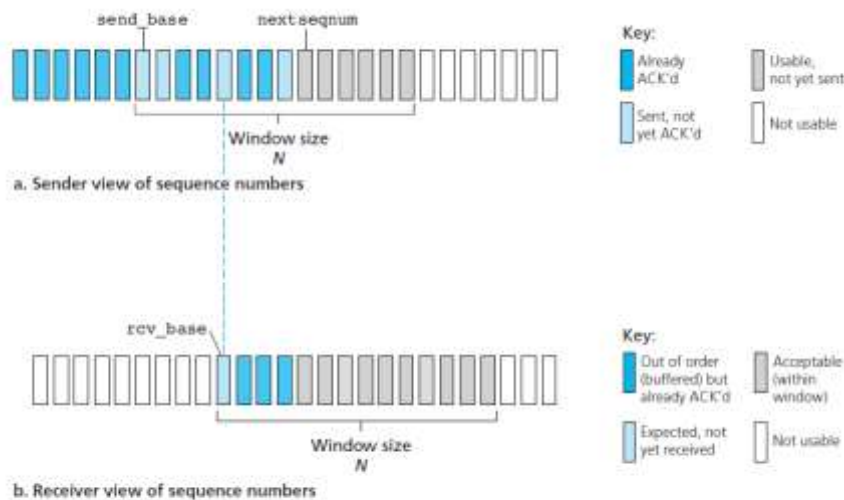


Figure 3.23 ♦ Selective-repeat (SR) sender and receiver views of sequence-number space

1. *Data received from above.* When data is received from above, the SR sender checks the next available sequence number for the packet. If the sequence number is within the sender's window, the data is packetized and sent; otherwise it is either buffered or returned to the upper layer for later transmission, as in GBN.
2. *Timeout.* Timers are again used to protect against lost packets. However, each packet must now have its own logical timer, since only a single packet will be transmitted on timeout. A single hardware timer can be used to mimic the operation of multiple logical timers [Varghese 1997].
3. *ACK received.* If an ACK is received, the SR sender marks that packet as having been received, provided it is in the window. If the packet's sequence number is equal to `send_base`, the window base is moved forward to the unacknowledged packet with the smallest sequence number. If the window moves and there are untransmitted packets with sequence numbers that now fall within the window, these packets are transmitted.

Figure 3.24 ♦ SR sender events and actions

1. *Packet with sequence number in  $[\text{rcv\_base}, \text{rcv\_base}+N-1]$  is correctly received.* In this case, the received packet falls within the receiver's window and a selective ACK packet is returned to the sender. If the packet was not previously received, it is buffered. If this packet has a sequence number equal to the base of the receive window (`rcv_base` in Figure 3.22), then this packet, and any previously buffered and consecutively numbered (beginning with `rcv_base`) packets are delivered to the upper layer. The receive window is then moved forward by the number of packets delivered to the upper layer. As an example, consider Figure 3.26. When a packet with a sequence number of `rcv_base=2` is received, it and packets 3, 4, and 5 can be delivered to the upper layer.
2. *Packet with sequence number in  $[\text{rcv\_base}-N, \text{rcv\_base}-1]$  is correctly received.* In this case, an ACK must be generated, even though this is a packet that the receiver has previously acknowledged.
3. *Otherwise.* Ignore the packet.

Figure 3.25 ♦ SR receiver events and actions

# 拥塞控制

拥塞控制和流量控制

流量控制：为了解决发送方使得接收方缓存溢出的可能性而遏制发送方

拥塞控制：因为 IP 网络的拥塞而遏制发送方

## 拥塞的原因与开销

情况 1：两个发送方和一个无穷大缓存的路由器

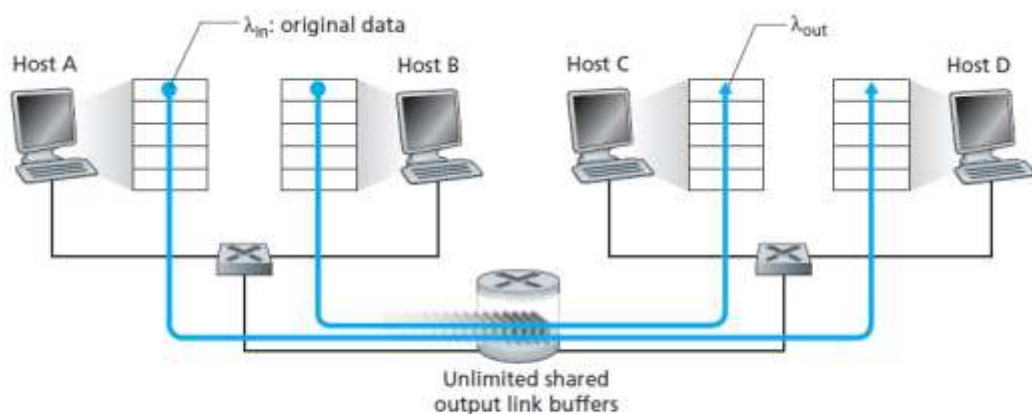


Figure 3.43 ♦ Congestion scenario 1: Two connections sharing a single hop with infinite buffers

拥塞开销：当分组到达速率接近链路容量时，分组经历的巨大排队时延

情况 2：两个发送方和一个有限缓存的路由器

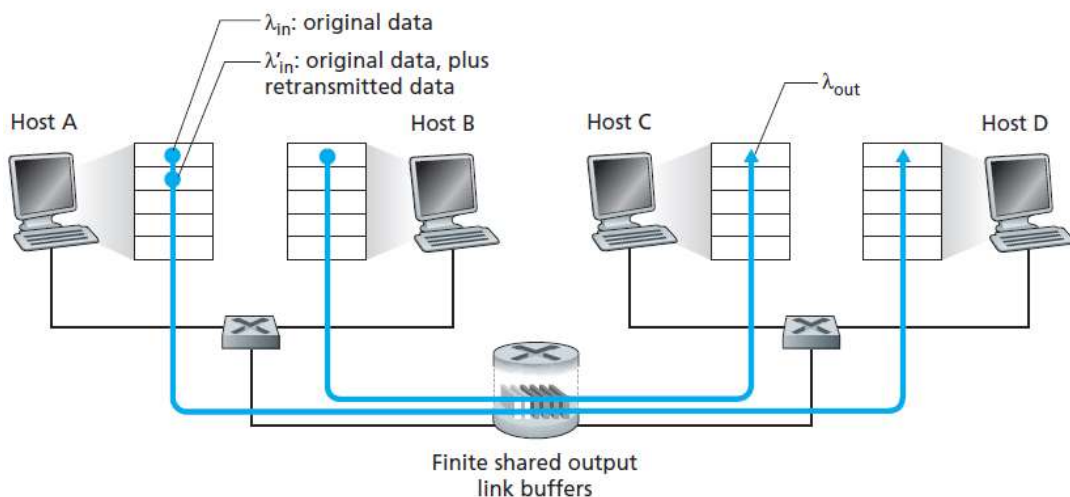


Figure 3.45 ♦ Scenario 2: Two hosts (with retransmissions) and a router with finite buffers

拥塞开销：发送方必须执行重传以补偿因为缓存溢出丢弃（丢失）的分组；

发送方在遇到巨大时延时所进行的不必要重传引起路由器利用链路带宽转发不必要的分组拷贝。

情况 3：四个发送方和一个无穷大缓存的多台路由器和多跳路径

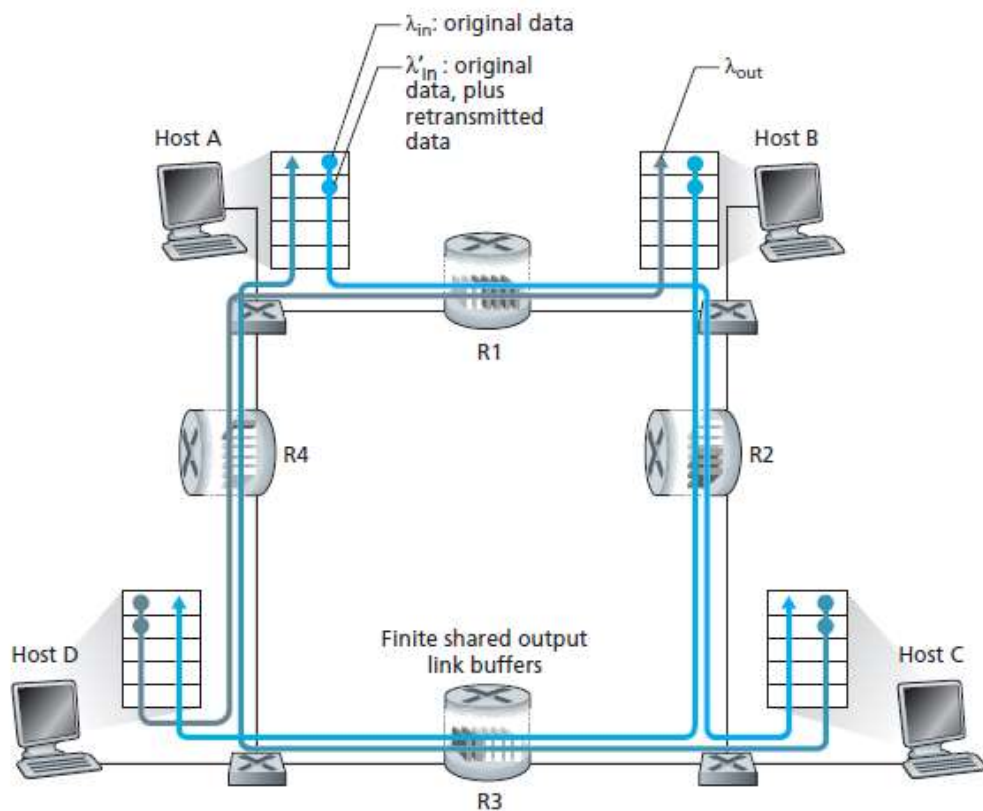


Figure 3.47 ♦ Four senders, routers with finite buffers, and multihop paths

情况 3：当一个分组沿着一条路径被丢弃时，每个上游路由器用于转发该分组到丢弃该分组而使用的传输容量最终被浪费掉了

### 拥塞控制方法

端到端拥塞控制：TCP 的慢启动算法

网络辅助的拥塞控制：ATM 的 ABR 拥塞控制，传输的信元包括数据信元和资源管理信源两种

## TCP 连接的建立和释放

建立连接

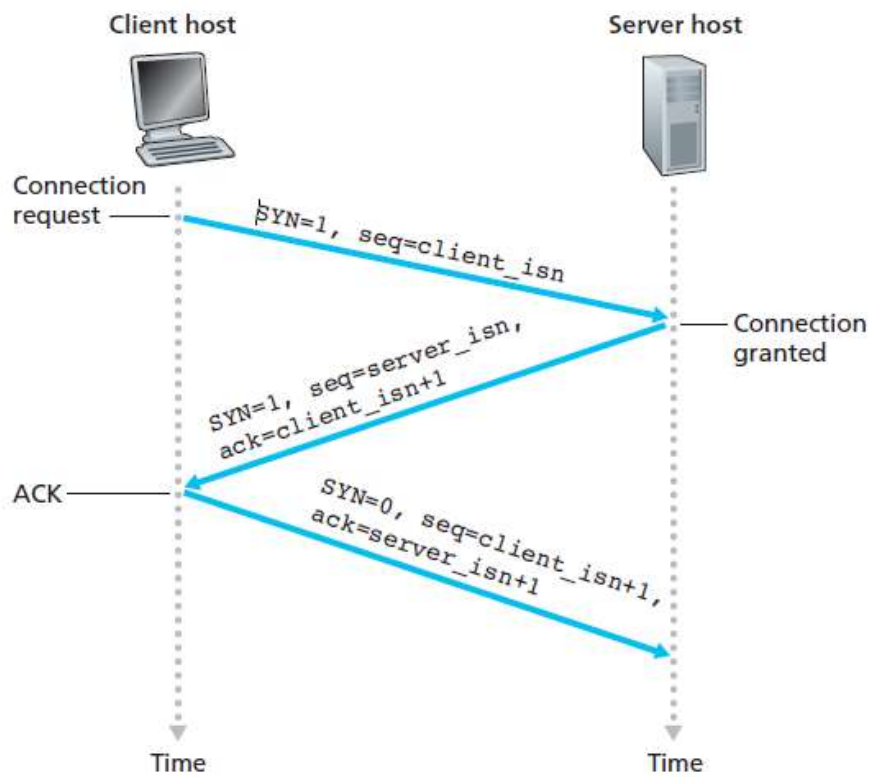


Figure 3.39 ♦ TCP three-way handshake: segment exchange

释放链接

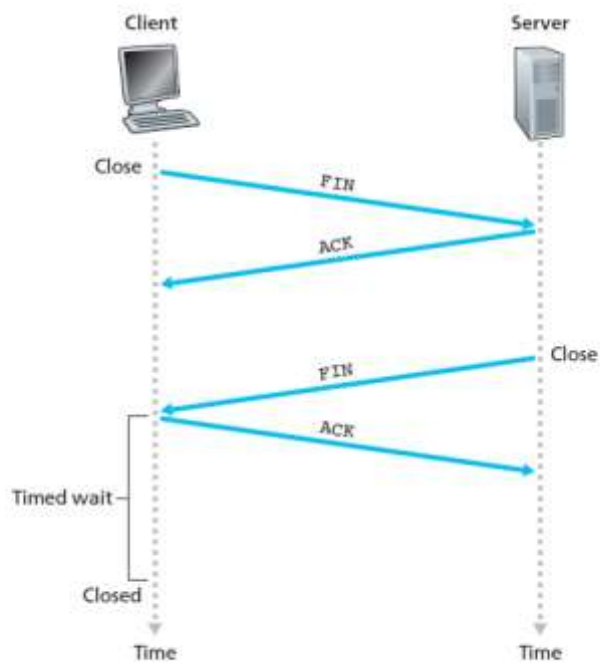
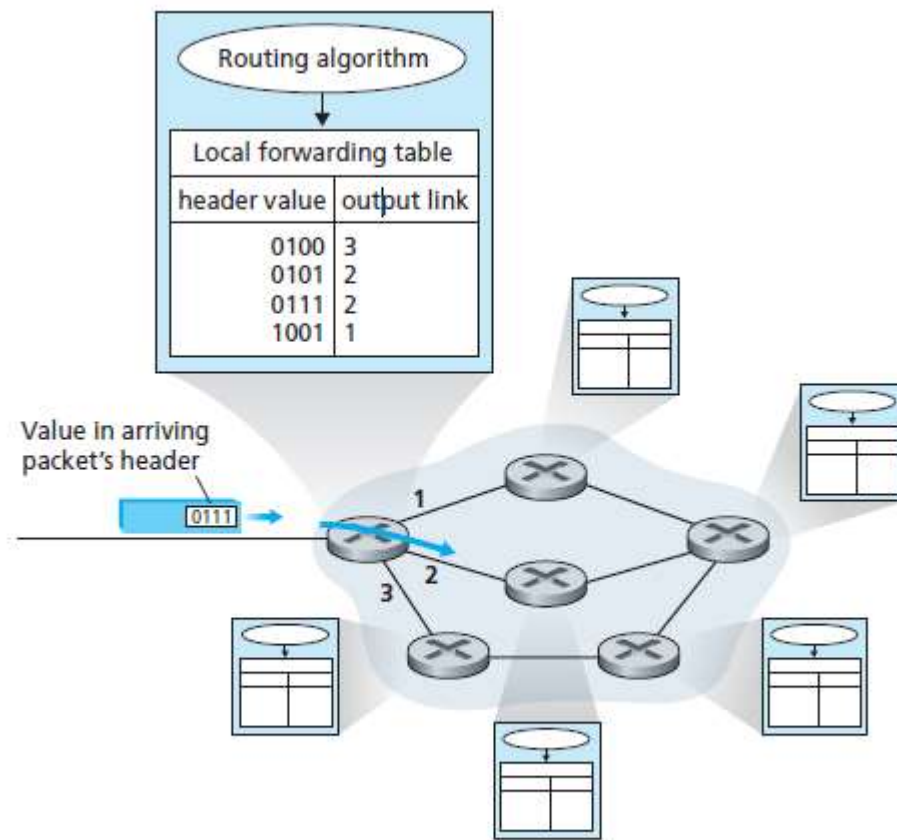


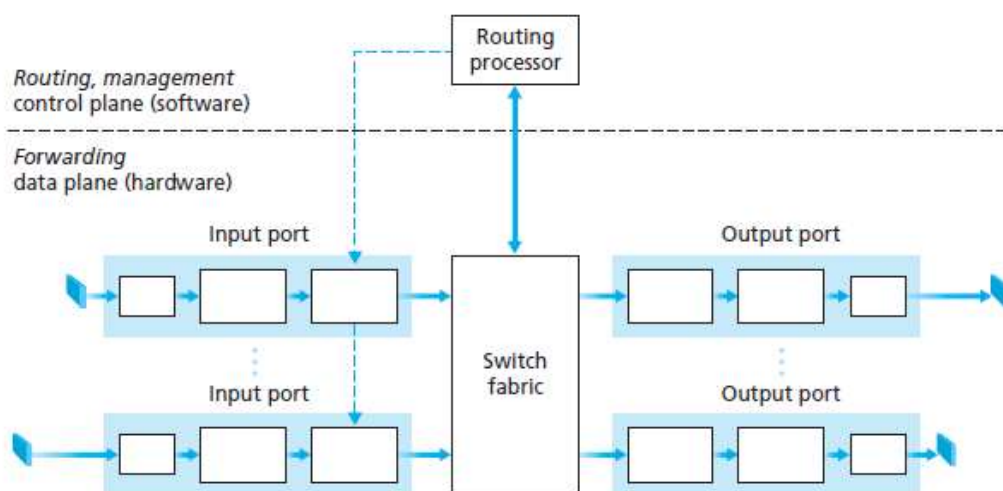
Figure 3.40 ♦ Closing a TCP connection



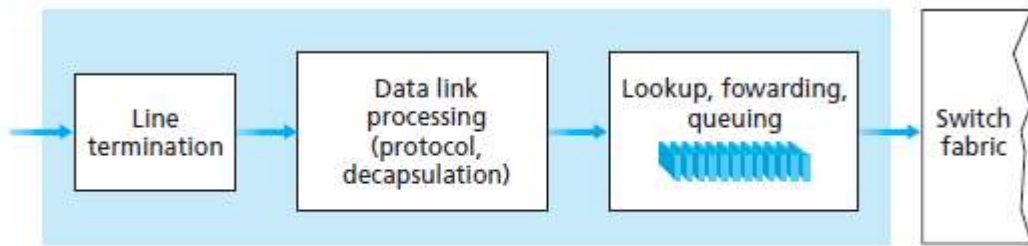
## 路由器结构



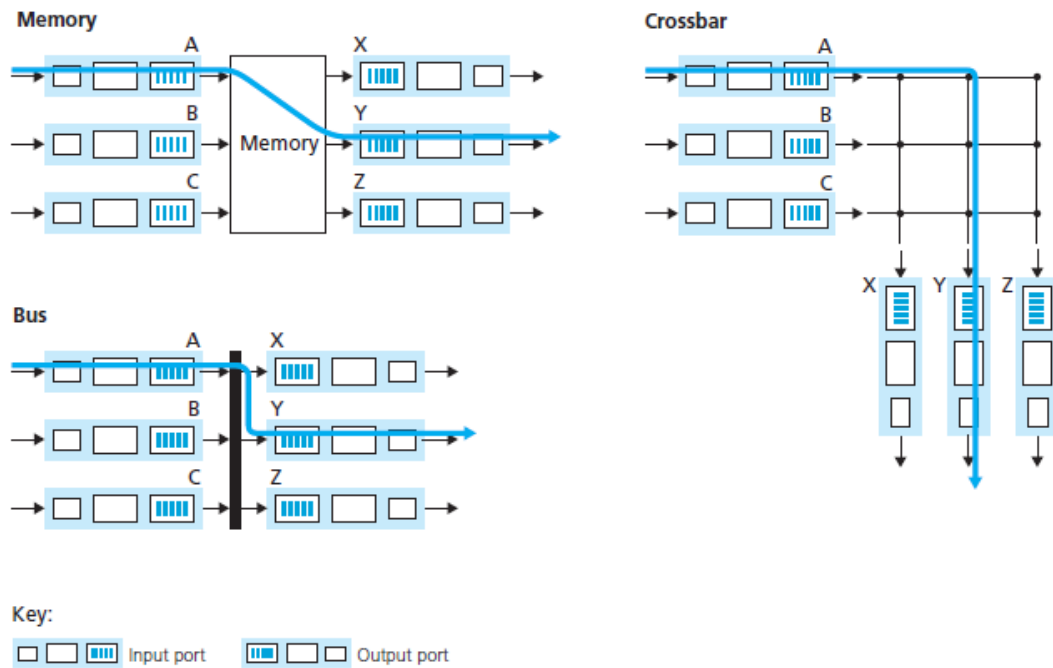
**Figure 4.2** ♦ Routing algorithms determine values in forwarding tables



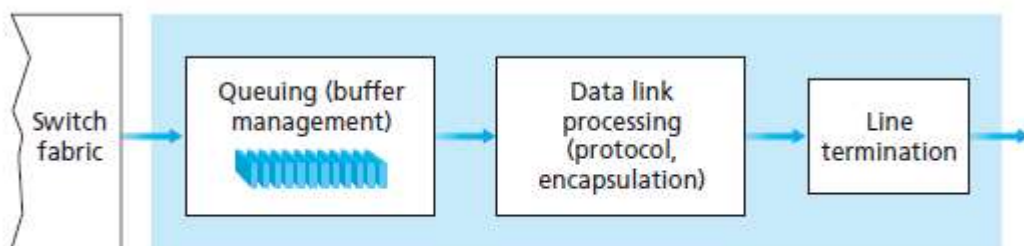
**Figure 4.6** ♦ Router architecture



**Figure 4.7** ♦ Input port processing



**Figure 4.8** ♦ Three switching techniques



**Figure 4.9** ♦ Output port processing

## 计算机网络复习提纲 2015

### 第一章 概述

#### 1.1 计算机网络的定义、P2P

#### 1.2 网络硬件（广播、多播、单播、局域网、广域网、无线网络）

#### 1.3 网络软件（协议、层次、错误控制、流量控制、面向连接与无连接的服务、可靠和不可靠的服务、OSI 参考模型、TCP/IP 参考模型）

#### 1.6 网络标准化

### 第二章 物理层

#### 2.1 数据通信的理论基础（带宽、信道的最大传输率）

#### 2.2 有导向的传输介质

#### 2.6 PSTN 电话系统（电路交换、分组交换）

### 第三章 数据链路层

#### 3.1.2 成帧（位填充）

#### 3.2 错误检测和纠正（CRC）

#### 3.3 基本数据链路协议（停等协议）

#### 3.4 滑动窗口协议（捎带确认、发送窗口、接收窗口、1 位滑动窗口协议、回退 N、选择性重发协议）

### 第四章 介质访问子层

#### 4.1 介质访问

#### 4.2 多路访问协议（CSMA/CD、最小帧长、MACAW）曼彻斯特编码、差分曼彻斯特编码

#### 4.3 以太网（二进制指数后退算法、交换式以太网）

#### 4.8 数据链路层交换（网桥工作原理、生成树网桥、中继器、集线器、交换机、路由器）

### 第五章 网络层

#### 5.1 网络层设计要点（虚电路子网、数据报子网）

#### 5.2 路由算法（优化原则、汇集树、距离矢量路由及无穷计算问题、链路状态路由、距离矢量路由和链路状态路由的比较、分级路由、广播路由、移动路由）

#### 5.3 拥塞控制（RED）

#### 5.4 服务质量（可靠性、延迟、抖动、带宽、漏桶算法、令牌桶算法、资源预留）

#### 5.5 网络互连（隧道技术）

#### 5.6 Internet 上的网络层（IP 协议、IP 地址、子网、子网掩码、CIDR、地址聚合技术、NAT、ICMP、ARP、DHCP、OSPF、BGP、~~和~~）

### 第六章 传输层

#### 6.2 传输协议的要素（编址、建立连接、释放连接、流控制和缓冲）

#### 6.4 UDP

#### 6.5 TCP（TCP 服务模型、TCP 协议、TCP 连接建立三次握手、TCP 连接释放、TCP 传输策略、Nagle 算法、愚笨窗口综合症、TCP 拥塞控制、慢启动算法、）

### 第七章 应用层

#### 7.1 DNS、HTTP (应用层常见的协议)