

# 第12章

## 多路访问

---

## 图12.1 数据链路层被分为两个功能定位的子层

---

**p** 上子层负责数据链路控制，下子层负责解决对共享介质的访问（如果通道是专用的，则不需要下子层）；

**p** 例如，IEEE对LAN做了分割：

- ∅ 上子层负责流量控制和差错控制，称为逻辑链路控制层（LLC）；

- ∅ 下子层主要负责多路访问的解决方案，称为介质访问控制层（MAC）

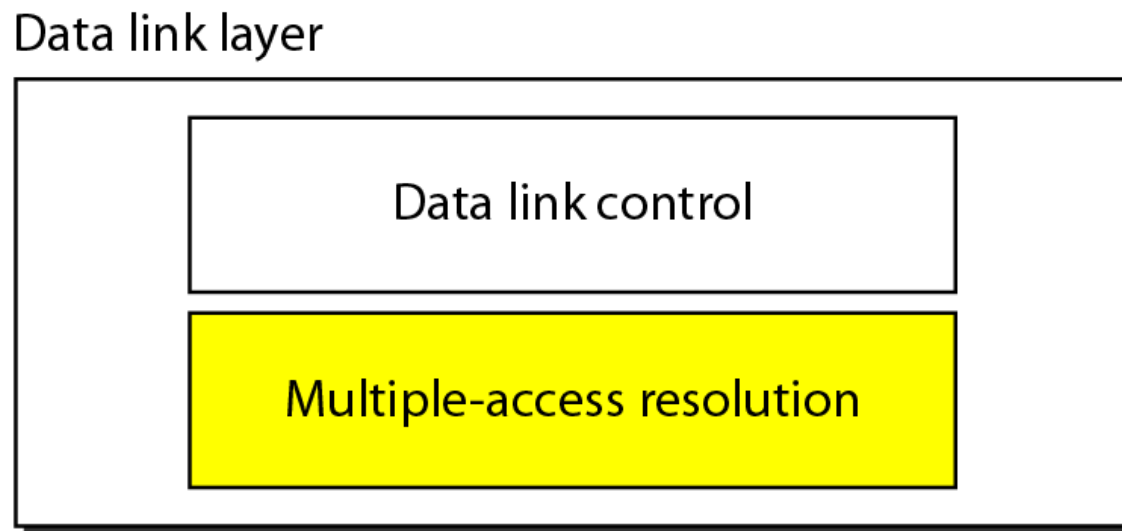
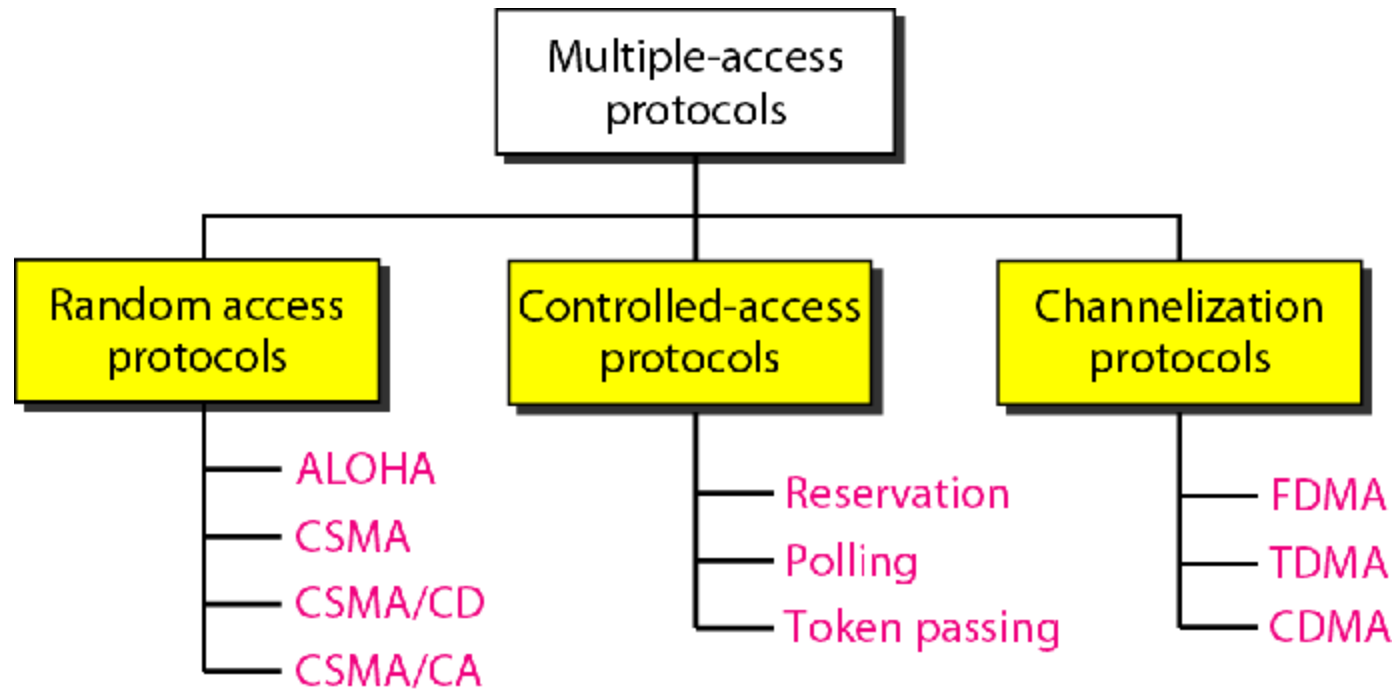


图12.2 多路访问协议的分类（随机访问、受控访问、通道化）



## 12-1 随机访问

**p**在随机访问或竞争访问方法中，没有站点优于其他站点，也没有任何站点能控制其他站点；

**p**一个站点没有权利允许或不允许另一个站点的发送，当一个站点要发送数据时，都要用协议定义的程序来决定是否要发送，而这个决定取决于介质的状态（空闲或忙碌）；

**p**换言之，只要遵循预定义的程序，包括介质状态的检测，满足条件的每一个站点都能传输数据；

**p**两个特性：（1）每一个站点的传输没有特定的时间表，站点的传输是随机的；（2）没有规则来规定下一个将要发送的站点是哪一个，为了访问介质，站点展开竞争

---

p 当一个以上的站点想要发送数据时，便会产生冲突（collision），就会有帧被损坏或修改；

p 为了避免访问冲突或在访问冲突发生时解决这个问题，每个站点遵循一个程序（即多路访问协议），解决如下问题：

- Ø 站点何时能访问介质？
- Ø 如果介质忙碌，站点能做什么？
- Ø 站点如何确定传输的成功或者失败？
- Ø 如果发生访问冲突，站点能做什么？

---

# ALOHA

---

- 最早的随机访问方法，由夏威夷大学在20世纪70年代初开发出来；
- 被设计用于无线电（无线）LAN，但也可以使用在任何共享介质上；
- 包含两种不同类型：
  - 纯ALOHA：简单但优秀，只要每个站点有帧要发送，它就发送帧（产生冲突）；
  - 时隙ALOHA：把时间分割成时隙，并强制站点只有在时隙开始之时才能发送（降低冲突概率）

**图12.3** 纯ALOHA网络中的帧（只有一条共享通道，帧只要交叠就会发生冲突）

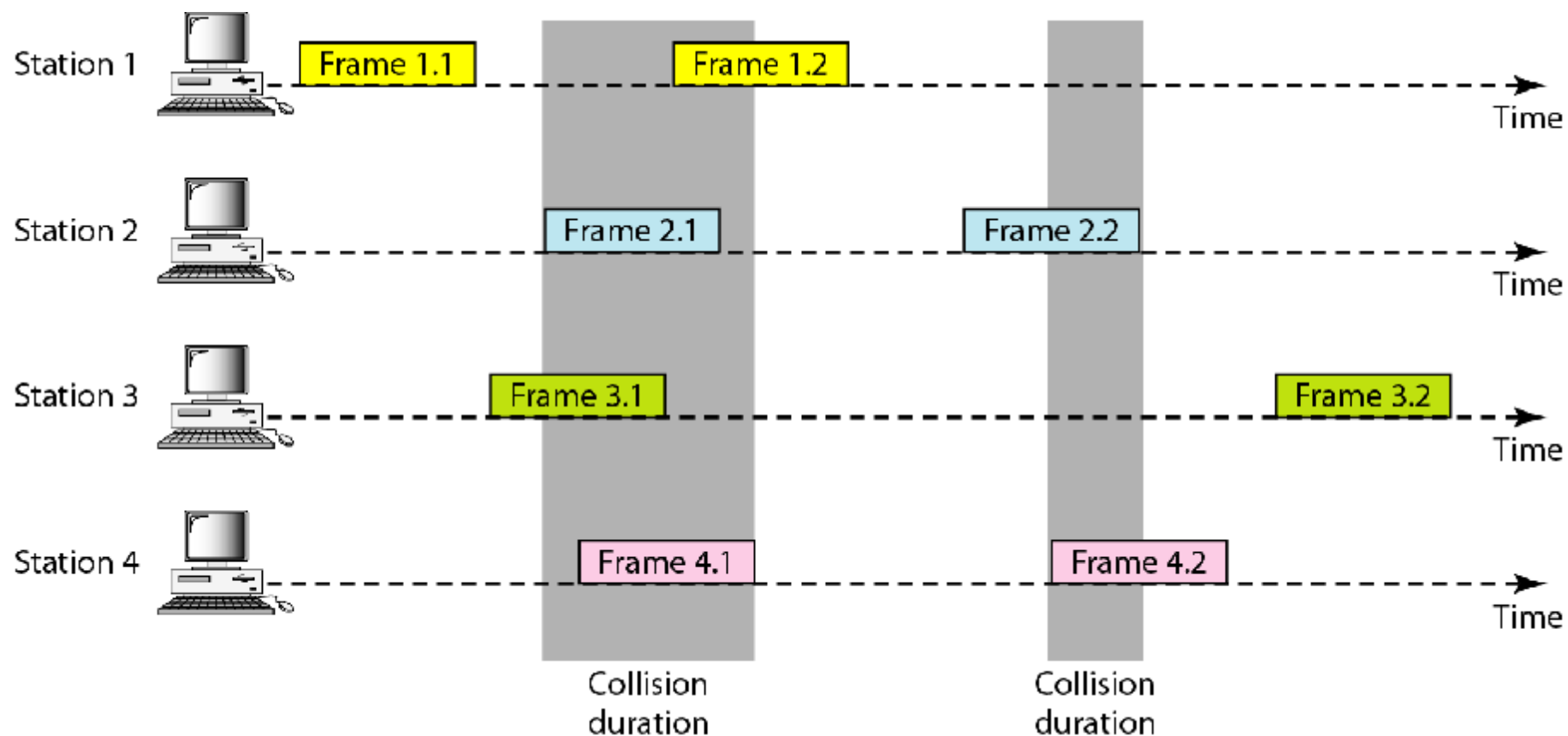


图12.4 纯ALOHA协议的流程图

**p**需要对方确认，若超时 ( $2T_p$ ) 后确认还未达到，重发帧；

**p**超时后每个站点重发帧之前都会随机等待一段时间（补偿时间  $T_B$ ）；

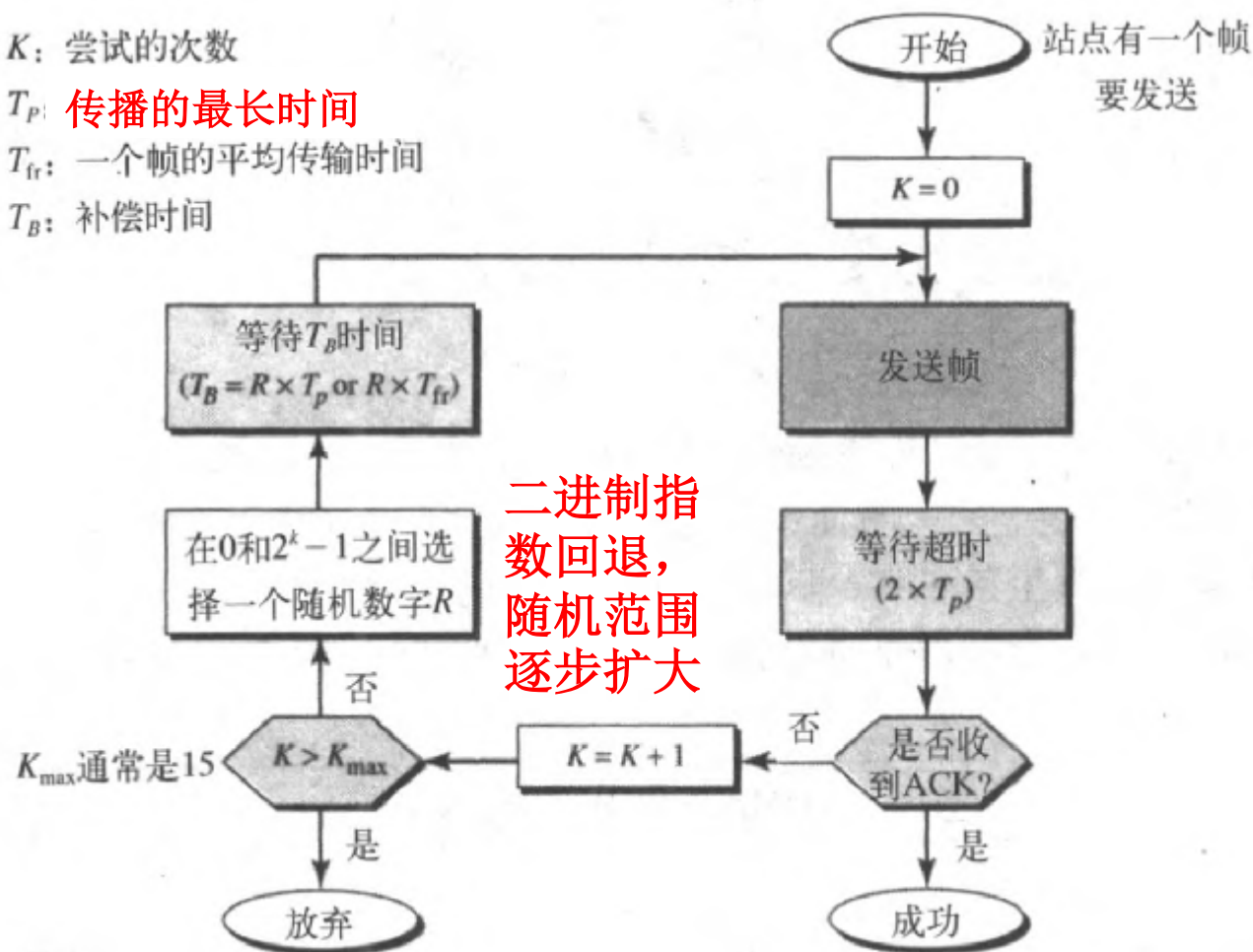
**p**一个站点在经过最大次数  $K_{\max}$  的重发尝试后，它必须放弃并以后不再试

$K$ : 尝试的次数

$T_p$ : 传播的最长时间

$T_{fr}$ : 一个帧的平均传输时间

$T_B$ : 补偿时间







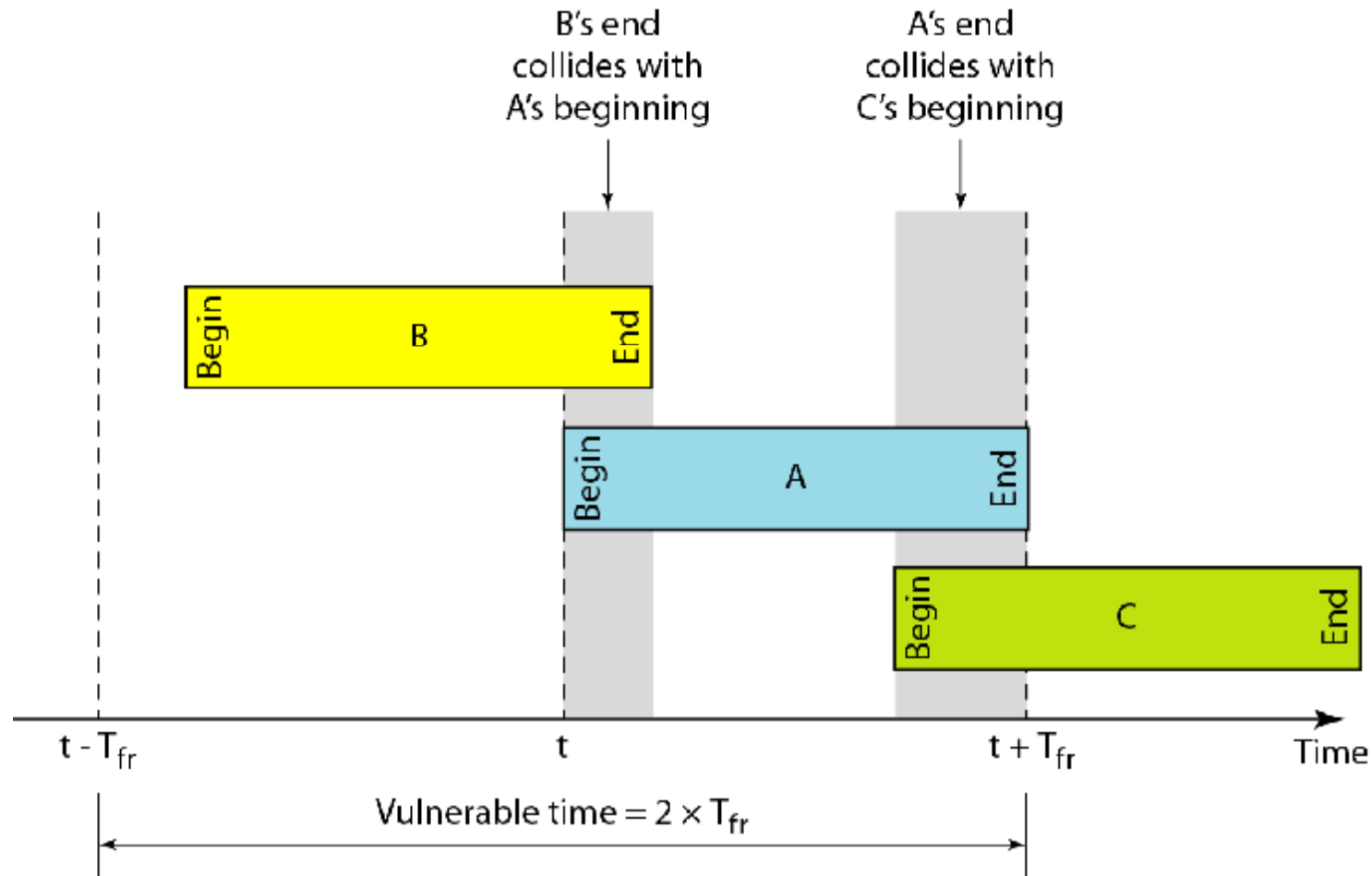
## 例12.1

一个无线ALOHA网络中的站点之间的最远距离是600公里。如果我们假设信号的传播速度是 $3 \times 10^8$ 米/秒，那么 $T_p = (6 \times 10^5) / (3 \times 10^8) = 2\text{ms}$ 。于是，对于不同的K值，得到不同的 $T_B$ 值。

- a. 如果 $K=1$ ，取值范围是 $\{0, 1\}$ ，站点会产生一个随机变量，变量值为0或1，这就意味着 $T_B$ 是0（ $0 \times 2$ ）毫秒或2（ $1 \times 2$ ）毫秒，取决于产生的随机变量；
- b. 如果 $K=2$ ，取值范围是 $\{0, 1, 2, 3\}$ ，这就意味着 $T_B$ 是0、2、4或6毫秒，取决于产生的随机变量；
- c. 如果 $K=3$ ，取值范围是 $\{0, 1, 2, 3, 4, 5, 6, 7\}$ ，这就意味着 $T_B$ 是0、2、4、.....、14毫秒，取决于产生的随机变量；
- d. 要提醒的一点是，如果 $K > 10$ ，它通常被设定为10。

图12.5 纯ALOHA协议的脆弱时间（可能发生冲突的时间，是帧传输时间 $T_{fr}$ 的2倍）

**p**B帧末端与A帧始端发生冲突，C帧始端与A帧末端发生冲突



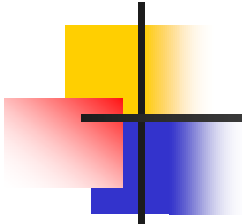


## 例12.2

一个纯ALOHA网络在一个传输速度为200kbps的共享通道上传输200位的帧，使这个帧在传输过程中避免冲突要达到什么样的条件？

**解：**

帧的平均传输时间 $T_{fr}$ 是200比特/200kbps，即1毫秒，脆弱时间是 $2 \times 1$ 毫秒=2毫秒，这就意味着在该站点开始传输前1毫秒，没有站点会开始传输，并且当该站点正在传输的1毫秒期间也没有站点会开始传输。



---

纯ALOHA的吞吐量（定义为成功传输帧的平均数量）是 $S=G \times e^{-2G}$ ，当 $G=1/2$ 时，最大吞吐量为 $S_{\max}=0.184$ 。

其中， $G$ 为一个帧传输时间内系统产生的帧的平均数量。

---



## 例12.3

一个纯ALOHA网络在一个传输速度为200kbps的共享通道上传输200位的帧，如果系统（包括所有站点）产生以下数量的帧，吞吐量将是多少？

- a. 每秒1000个帧
- b. 每秒500个帧
- c. 每秒250个帧

**解：**

帧的传输时间是200比特/200kbps，即1毫秒。

a. 如果系统每秒产生1000个帧，也就是每毫秒产生一个帧，负载为1，在这种情况下，即 $S=0.135$ （13.5%）。也就是说，吞吐量是 $1000 \times 0.135=135$ 帧，1000个帧中只有135个可能成功。

b. 如果系统每秒产生500个帧，也就是每毫秒产生1/2个帧，负载为1/2，在这种情况下，即 $S=0.184$ （18.4%）。也就是说，吞吐量是 $500 \times 0.184=92$ 帧，500个帧中只有92个可能成功（注意：这就是最大吞吐量百分比的例子）。

c. 如果系统每秒产生250个帧，也就是每毫秒产生1/4个帧，负载为1/4，在这种情况下，即 $S=0.152$ （18.4%）。也就是说，吞吐量是 $250 \times 0.152=38$ 帧，250个帧中只有38个可能成功。

图12.6 时隙ALOHA网络中的帧

**p**将时间分割成 $T_{fr}$ 秒的时隙，并强制站点只有在时隙开始之时才能发送

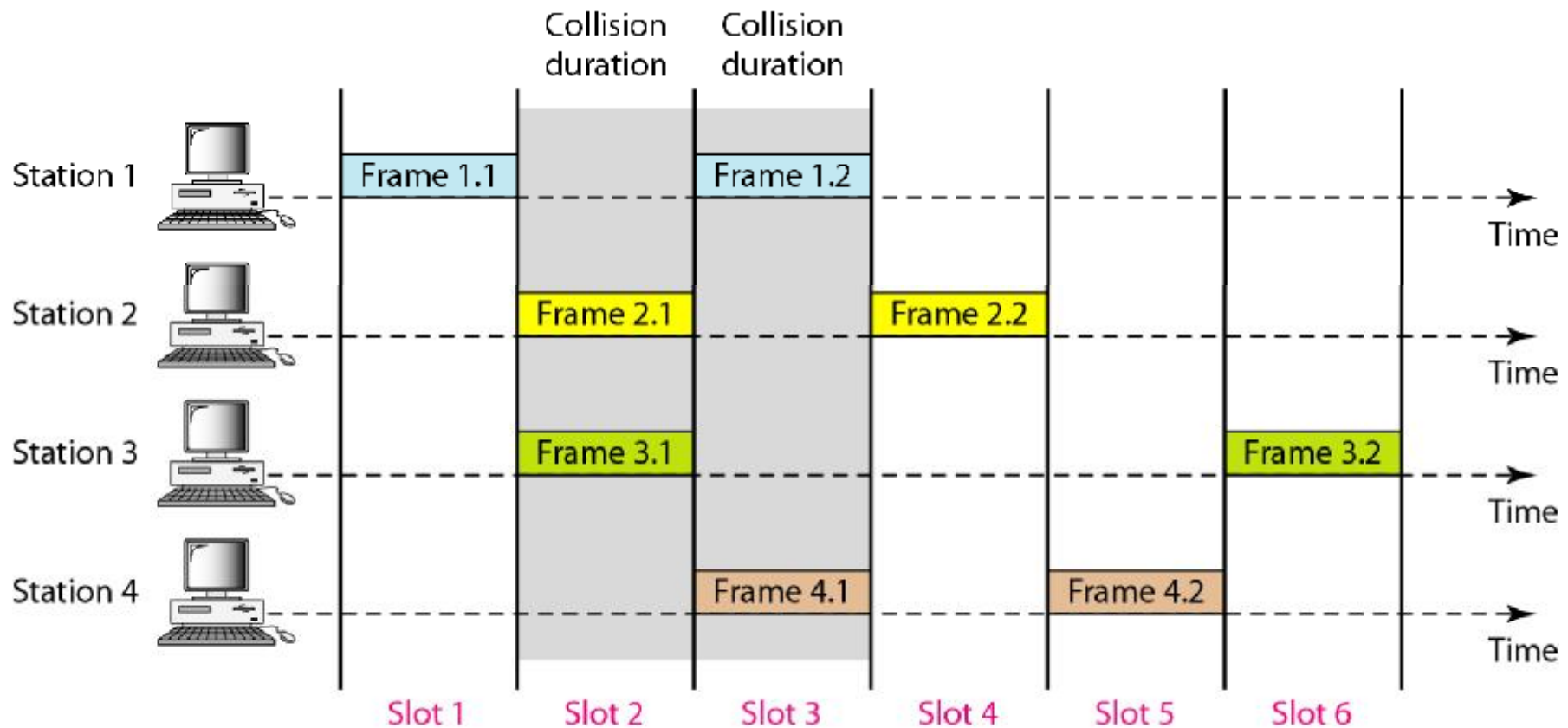
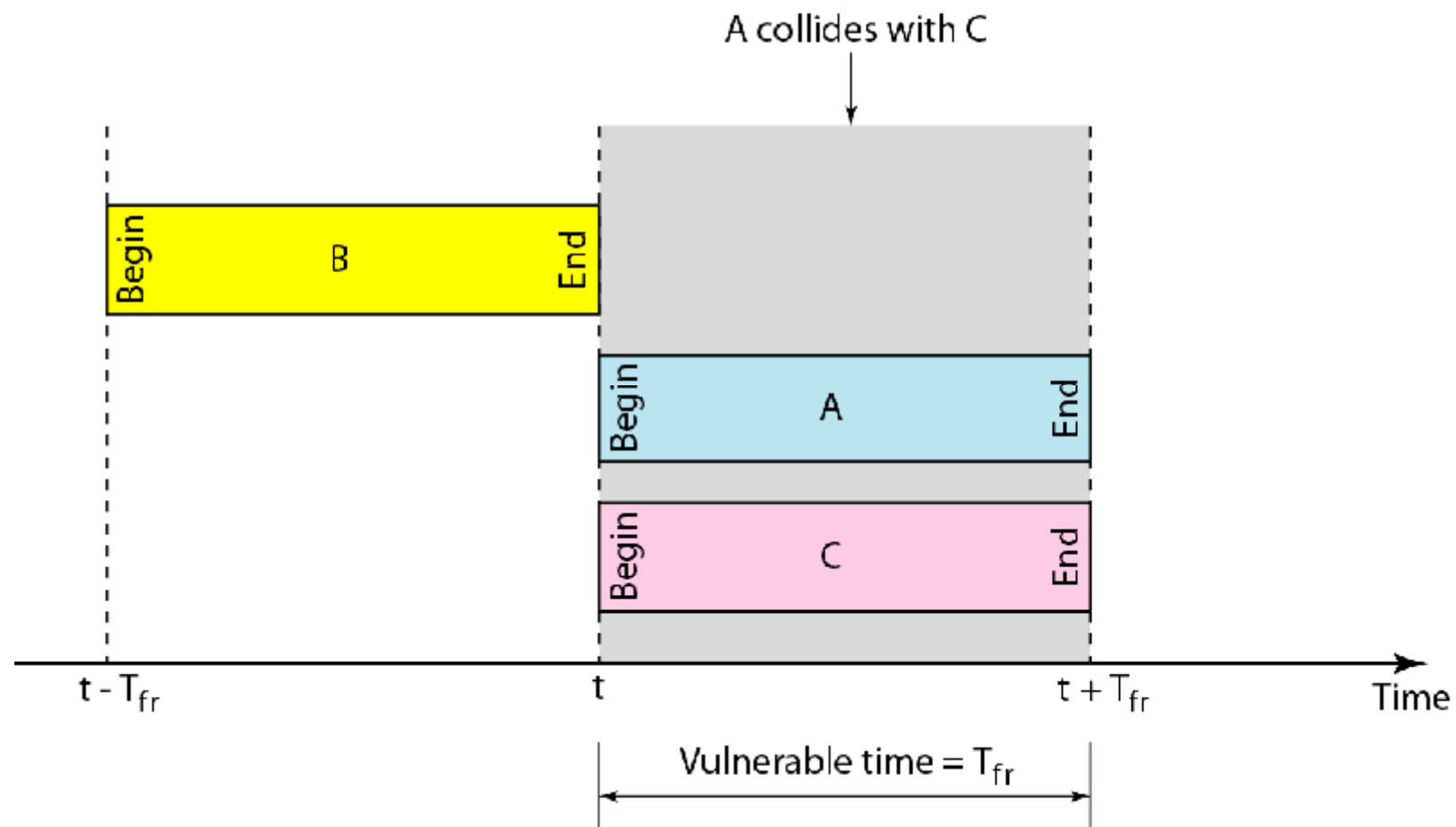
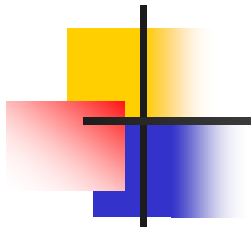


图12.7 时隙ALOHA协议中的脆弱时间（与帧长度时间相等）





时隙ALOHA的吞吐量是 $S=G \times e^{-G}$ ，当 $G=1$ 时，最大吞吐量为 $S_{\max}=0.368$ 。





## 例12.4

---

一个时隙ALOHA网络使用一个带宽为200kbps的共享通道发送200比特的帧，如果系统（包括所有站点）产生以下数量的帧，吞吐量将是多少？

- a. 每秒1000个帧
- b. 每秒500个帧
- c. 每秒250个帧



## 例12.4 (cont.)

解:

帧传输时间是 $200/200\text{kbps}$ , 即1毫秒。

a. 此例中 $G$ 为1, 因此 $S=G \times e^{-G}$ , 即 $S=0.368$  (36.8%), 这意味着吞吐量是 $1000 \times 0.368=368$ 帧, 1000个帧中只有368个能成功 (注意这就是最大吞吐量百分比的例子)。

b. 此例中 $G$ 为 $1/2$ , 因此 $S=G \times e^{-G}$ , 即 $S=0.303$  (30.3%), 这意味着吞吐量是 $500 \times 0.303=151$ 帧, 500个帧中只有151个能成功。

c. 此例中 $G$ 为 $1/4$ , 因此 $S=G \times e^{-G}$ , 即 $S=0.195$  (19.5%), 这意味着吞吐量是 $250 \times 0.195=49$ 帧, 250个帧中只有49个能成功。

---

## 载波侦听多路访问CSMA

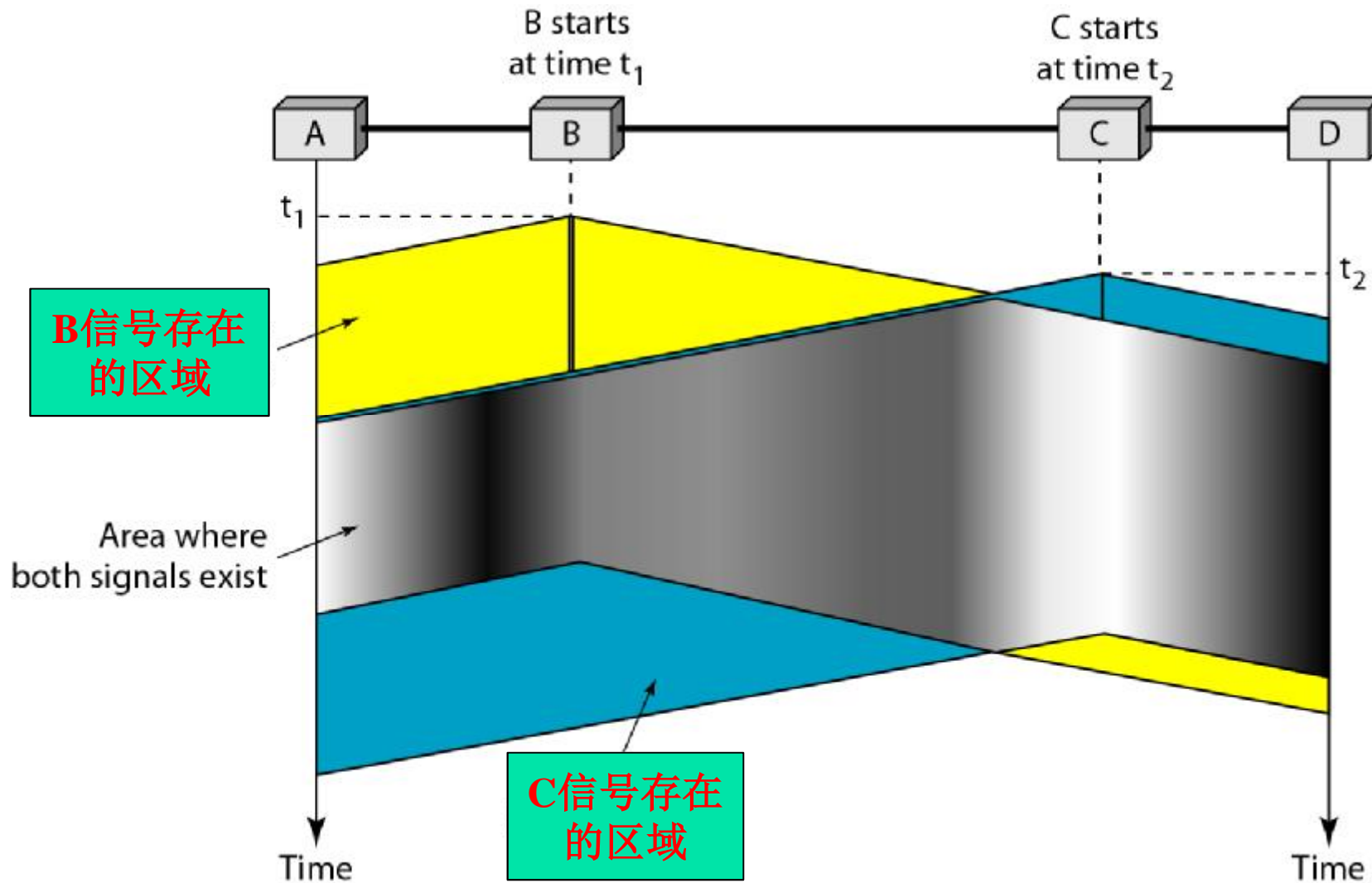
---

**pCSMA: Carrier Sense Multiple Access;**

**p**出发点：如果站点在发送前先侦听一下介质，其冲突的概率就会减少；

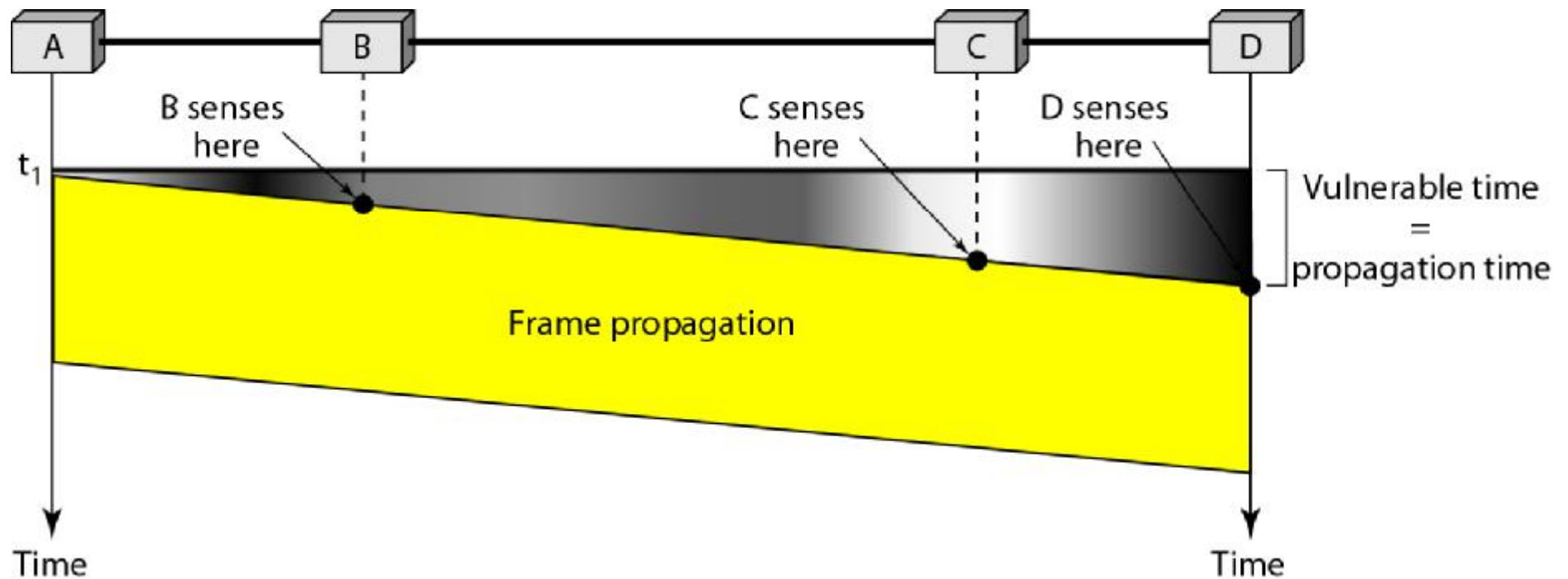
**pCSMA**要求每一个站点在发送之前首先要监听介质（或者检查介质的状态），换句话说，CSMA基于的原理就是“传输前先侦听”或“说话前先听”

**图12.8 CSMA中冲突的时空模型（减少冲突，但不能消除冲突，因为传播的延迟）**



**图12.9 CSMA的脆弱时间**（等于传播时间 $T_p$ ，即信号从介质一端到另一端的传播时间）

**p**灰色区域显示了时间上和空间上的脆弱区域



---

## 持续方法（监听算法）

---

**p**站点发送数据之前，先监听信道上是否有别的站发送的载波信号；若有，说明信道正忙，否则信道是空闲的；然后根据预定的策略决定：

（1）若信道空闲，是否立即发送？

（2）若信道忙，是否继续监听？

**p**即使信道空闲，若立即发送仍有可能发生冲突：一种情况是远端的站刚开始发送，载波信号尚未传到监听站，这时若监听站立即发送，就会和远端的站发生冲突；另一种情况是虽然暂时没有站发送，但碰巧两个站同时开始监听，如果它们都立即发送，也会发生冲突；

**p**监听算法（持续方法）并不能完全避免发送冲突，但若对以上两种控制策略进行精心设计，则可以把冲突概率减到最小；

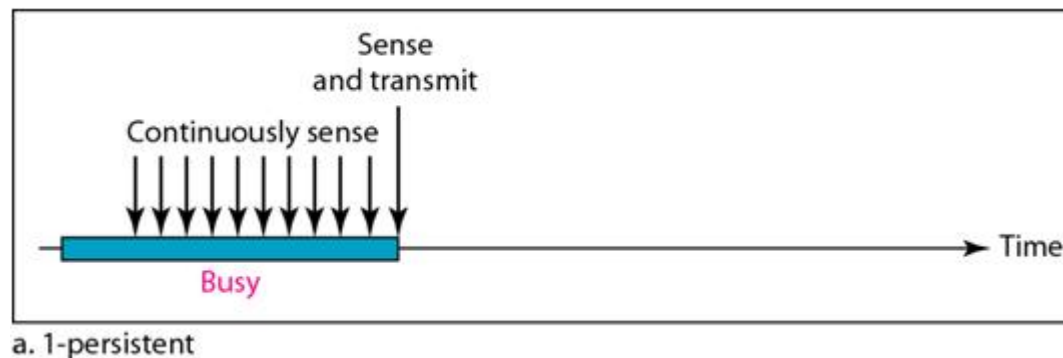
**p**三种持续方法：1-持续方法，非持续方法，p-持续方法

## 1-持续方法

**p** 站点发送帧之前先侦听，如果发现线路空闲，那么它就立即发送帧；否则就持续侦听，直到线路空闲为止；

**p** 优点：逻辑简单，直接，利用率高；

**p** 缺点：发生冲突的几率高，因为在发现了线路空闲后，两个或更多的站点都可以发送它们的帧。

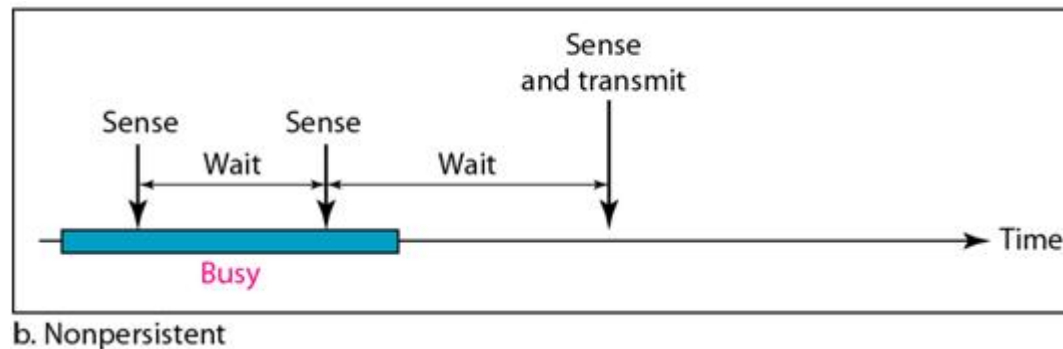


## 非持续方法

**p**要发送帧的站点先侦听线路，如果线路空闲，它就立即发送；如果线路不空闲，站点就等待一个随机时间，然后再次侦听线路；

**p**优点：减少了冲突的概率，因为不太可能存在两个或更多个站点等待相同的时间并且再次立即重新尝试；

**p**缺点：线路利用率低，因为有可能站点要发送帧而介质却空闲





## p-持续方法

**p**如果通道有时隙且时隙周期等于或大于最大传播时间，使用p-持续方法；

**p**站点发送帧之前先侦听，若线路繁忙则持续侦听直到线路空闲；若线路空闲：

(1) 以概率 $p$ 发送；

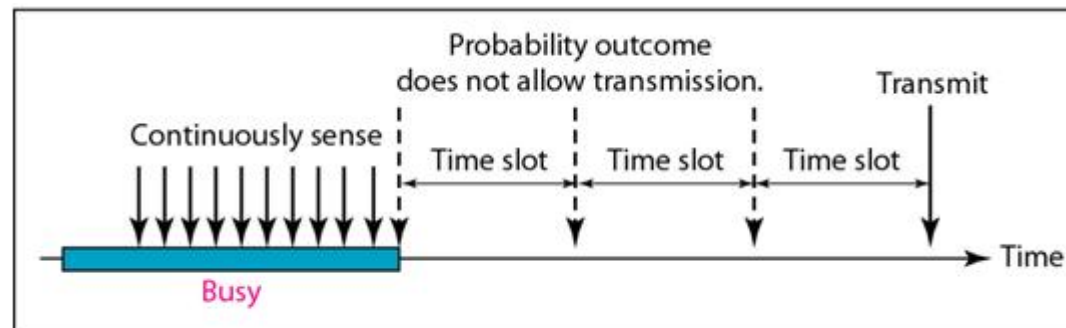
(2) 以概率 $(1-p)$ 等待下一个时隙的开始并再次检测线路；

(a) 若线路空闲，转(1)；

(b) 若线路繁忙，它会当做一个冲突已经发生了并使用补偿程序

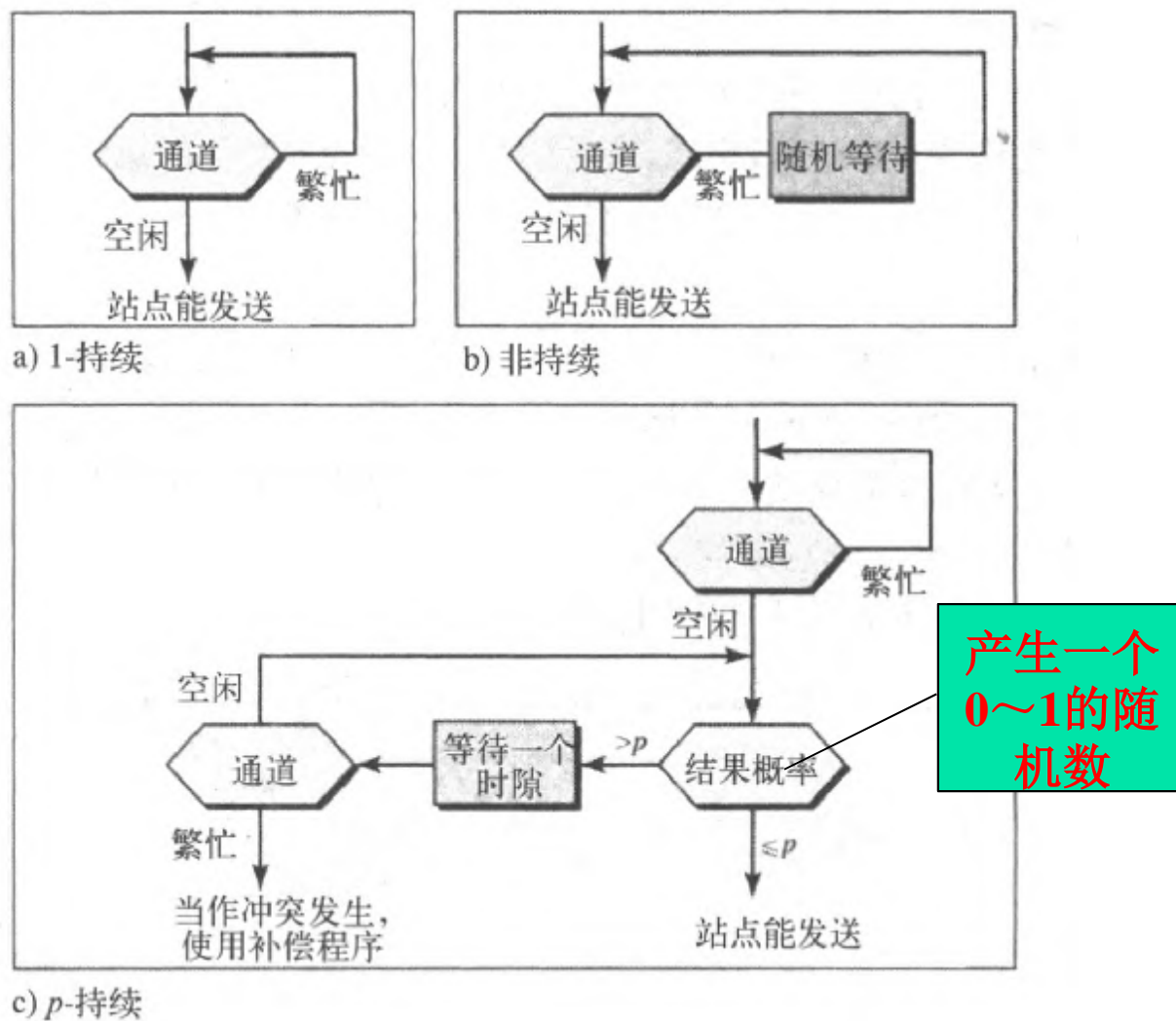
**p**优点：综合了前两个策略的优点，既减少了冲突机会，又提高了网络效率；

**p**缺点：较复杂（ $p$ 的取值）。



c. p-persistent

图12.11 三种坚持型方法的流程图



---

## 带冲突检测的载波侦听多路访问CSMA/CD

---

**p**CSMA的问题：载波侦听只能减小冲突的概率，不能完全避免冲突；当两个帧发生冲突后，若继续发送，将会浪费网络带宽，若帧比较长，问题尤其严重；

**p**为了进一步改进带宽的利用率，发送站应采取**边发边听**的冲突检测方法，如果在传输过程中检测到冲突，传输一个简单的干扰信号（jamming signal）以保证所有站点都知道发生了冲突，并停止传输；

**p**CSMA/CD: Carrier Sense Multiple Access with Collision Detection;

图12.12 在CMSA/CD中第一位冲突的情况

在时刻 $t_3$ ，当站点C收到A的第一个位时，它检测到冲突，C马上放弃了传输；

在时刻 $t_4$ ，当站点A收到C的第一个位时，它检测到冲突，它也同样马上放弃了传输；

**Q:** 要保证CD起作用，帧的长度有什么限制？

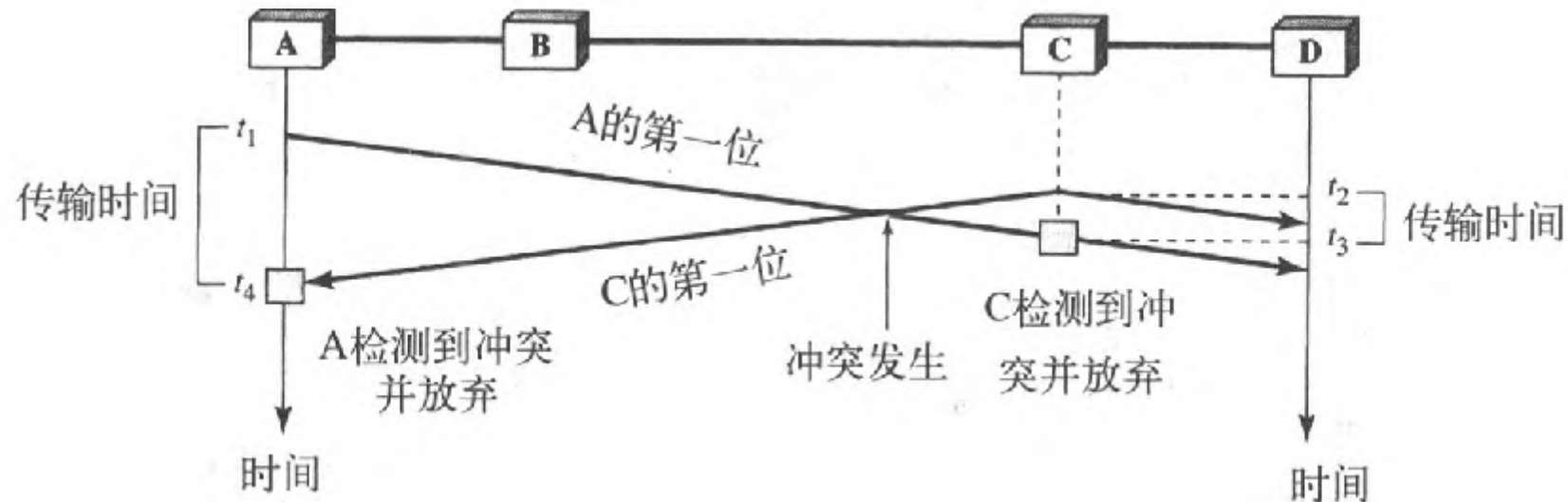
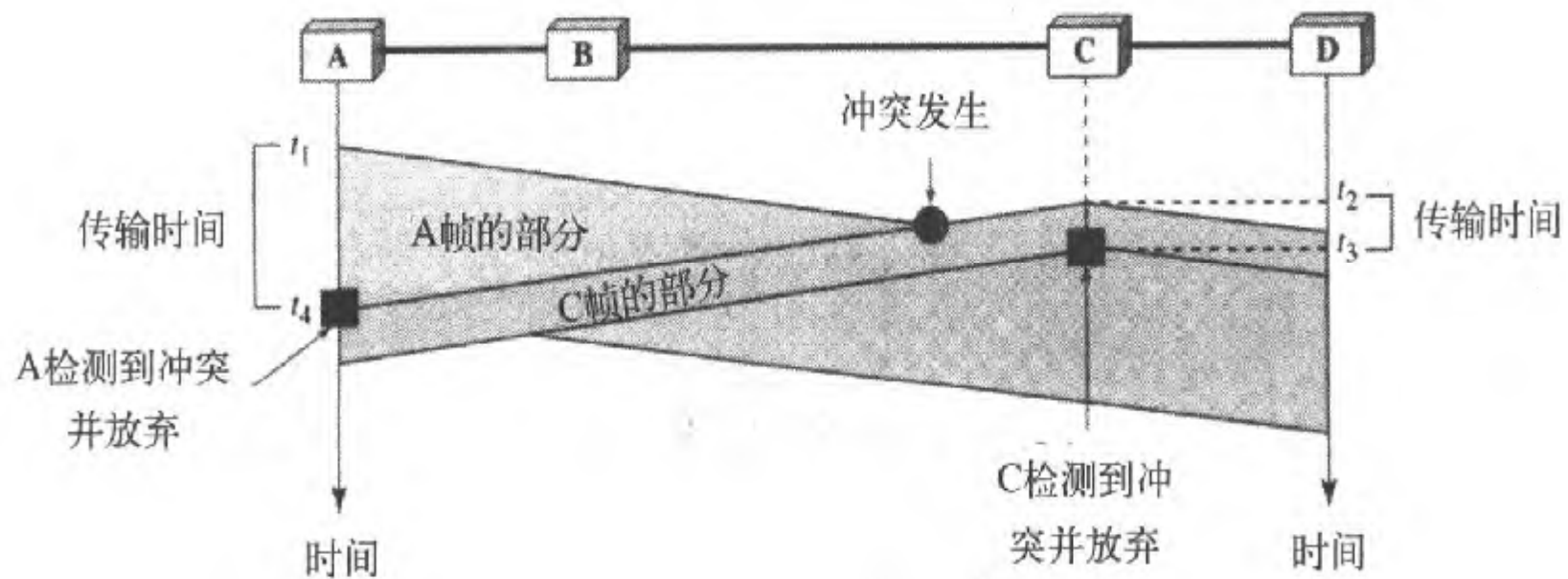


图12.13 在CSMA/CD中的冲突和放弃传输



## 帧的最小长度

为了实现CSMA/CD，需要限定帧的长度，使得在发送帧最后一位前，发送站点必须检测到冲突，如有冲突要放弃传输；

因为一旦整个帧被发送了，站点就无法保留帧的副本并无法掌控线路的冲突检测，因此，帧传输时间 $T_{fr}$ 必须至少是最大传播时间 $T_p$ 的两倍（这个时间 $2T_p$ 叫做冲突窗口）

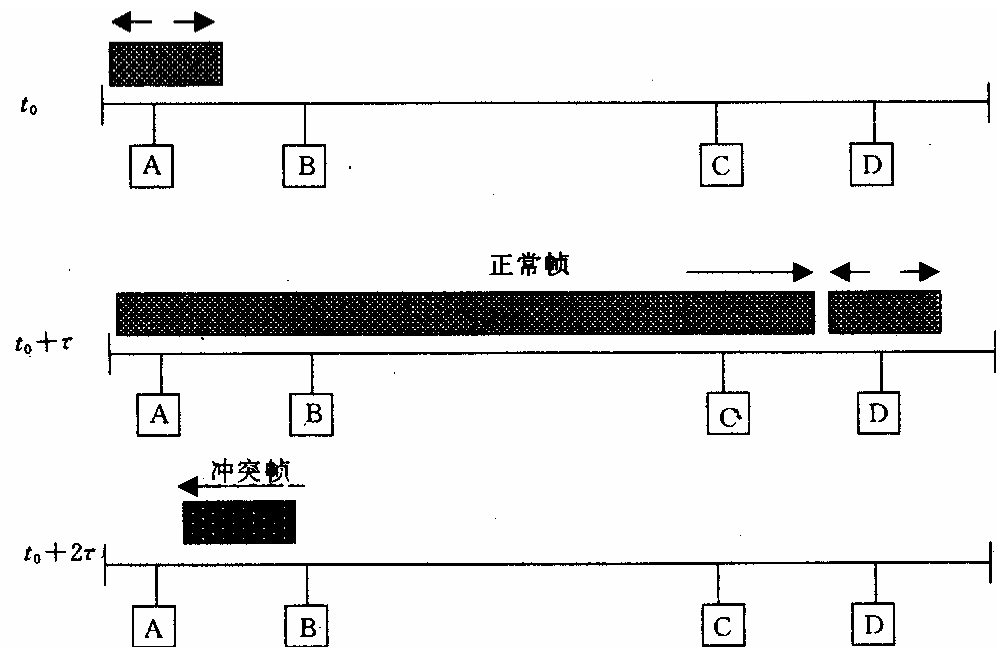
在 $t_0$ 时刻A开始发送；

假设经过一段时间 $\tau$ （即网络最大传播时延 $T_p$ ），D开始发送；

D立即就会检测到冲突，并能很快停止；

但A仍然感觉不到冲突，并继续发送；

再经过一段时间 $\tau$ ，A才会收到冲突信号，从而停止发送



---

## 帧的最小长度 (cont.)

---

**p** 网络标准中根据设计的数据速率和最大网段长度规定了最小帧长:

$$L_{\min} = 2 * R * d / v = 2 * R * T_p$$



## 例12.5

使用CSMA/CD的一个网络，带宽是10Mbps，如果最大传播时间（包括设备的延迟并忽略发送一个干扰信号所需的时间）是 $25.6\mu\text{s}$ ，那么帧的最小长度是多少？

**解：**

帧的传输时间 $T_{fr}=2\times T_p=51.2\mu\text{s}$ ，这就是说，最差情况下，一个站点需要 $51.2\mu\text{s}$ 后才能检测到冲突。帧的最小长度是 $10\text{Mbps}\times 51.2\mu\text{s}=512$ 比特或64字节，这也是标准以太网中帧的最小尺寸（见第13章）。



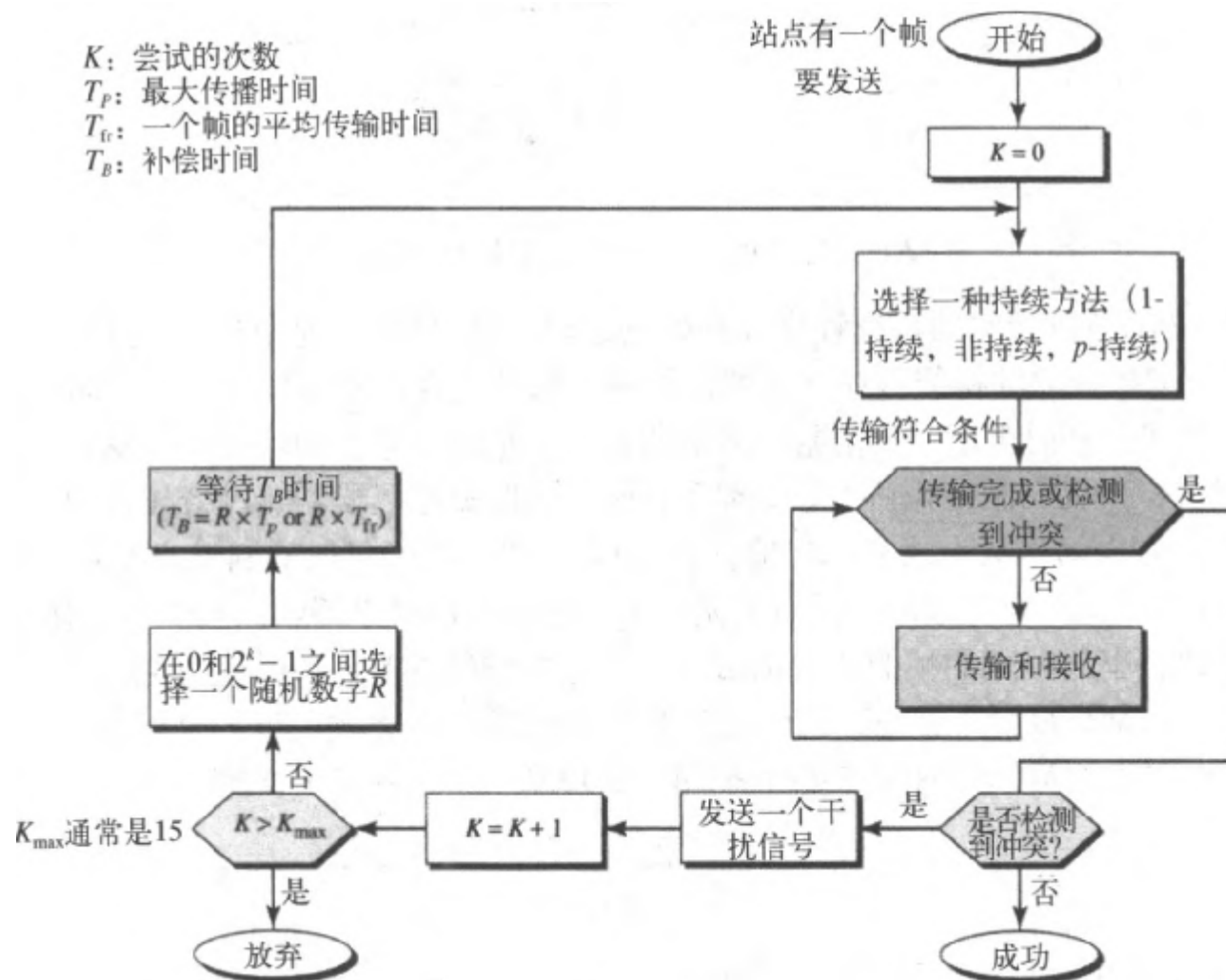
---

## p 二进制指数退避技术

- 在CSMA/CD协议中，一旦检测到冲突，为降低再冲突的概率，需要等待一个随机时间，然后再使用CSMA方法试图传输；
- 为了保证这种退避维持稳定，采用了二进制指数退避算法的技术，其算法过程如下：

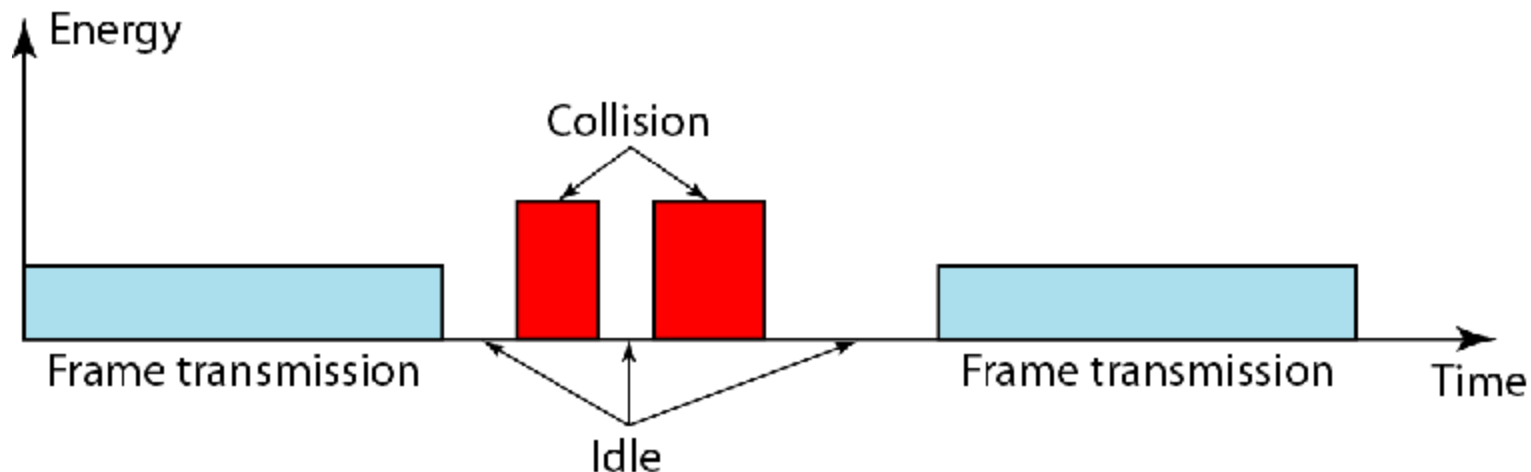
1. 将冲突发生后的时间划分为长度为 $2T_p$ 的时隙；
2. 发生第一次冲突后，各个站点等待0或1个时隙再开始重传；
3. 发生第二次冲突后，各个站点随机地选择等待0，1，2或3个时隙再开始重传；
4. 第 $i$ 次冲突后，在0至 $2^i-1$ 间随机地选择一个等待的时隙数，再开始重传；
5. 10次冲突后，选择等待的时隙数固定在0至1023（2的10次方减1）间；
6. 16次冲突后，发送失败，报告上层。

图12.14 CSMA/CD的流程图



**图12.15** 传输，空闲或冲突时的能量级别（依此可以判别 CD）

- 通道中的能量级别有三个值：0，正常和不正常；
- 在级别0，通道是空闲的；在正常级别，一个站点成功地占用了通道并正在发送一个帧；在非正常级别，发生了冲突且能量的级别是正常级别的两倍；
- 将要发送或者正在发送帧的站点必须监控能量级别来测定通道是处于空闲，繁忙还是冲突状态



---

## CSMA/CD吞吐量

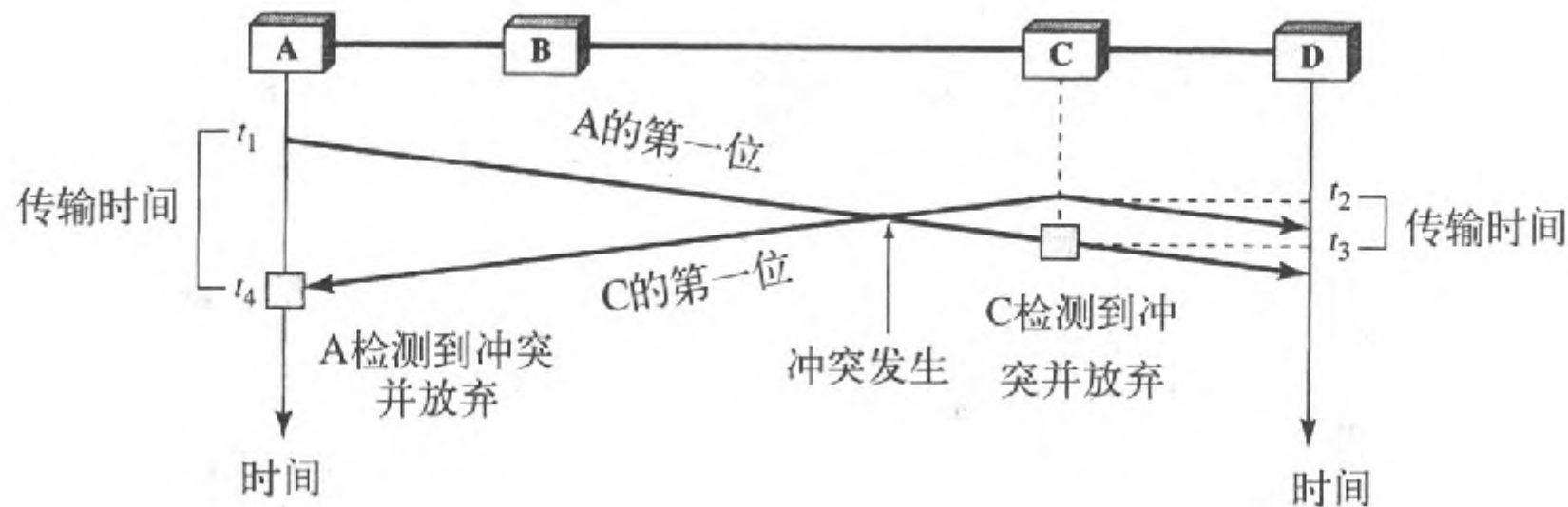
---

- p** CSMA/CD的吞吐量大于纯ALOHA或时隙ALOHA；
- p** 基于不同的持续方法及在p-持续方法中不同的p值，不同的G 值将有不同的最大吞吐量；
- p** 1-持续方法中，当 $G=1$ 时，得到最大吞吐量约为50%；非持续方法中，当G值在3到8之间时，能得到高达90%的最大吞吐量。

# 练习题

## p 课后练习题17 (P262) :

- a. 站点C听到冲突的时刻 $t_3$ =?
- b. 站点A听到冲突的时刻 $t_4$ =?
- c. 站点A在检测到冲突前发送的比特数是多少?
- d. 站点C在检测到冲突前发送的比特数是多少?
- e. 若A-D之间距离为10km, 按照CSMA/CD协议, 该局域网最短帧长应该是多少?



---

## 练习题答案

---

**p** 课后练习题17（P262）：

- a. 站点C听到冲突的时刻 $t_3 = 2000/2 \times 10^8 = 10\mu\text{s}$ ;
- b. 站点A听到冲突的时刻 $t_4 = 2000/2 \times 10^8 + t_2 = 13\mu\text{s}$ ;
- c. 站点A在检测到冲突前发送的比特数是：  
 $10\text{Mbps} \times 13\mu\text{s} = 130$ 比特;
- d. 站点C在检测到冲突前发送的比特数是：  
 $10\text{Mbps} \times (10-3)\mu\text{s} = 70$ 比特;
- e. 若A-D之间距离为10km，按照CSMA/CD协议，该局域网最短帧长应该是：  
 $2 \times 10 \times 10^6 \times (10 \times 10^3 / 2 \times 10^8) = 1000$ 比特。

---

## 带冲突避免的载波侦听多路访问CSMA/CA

---

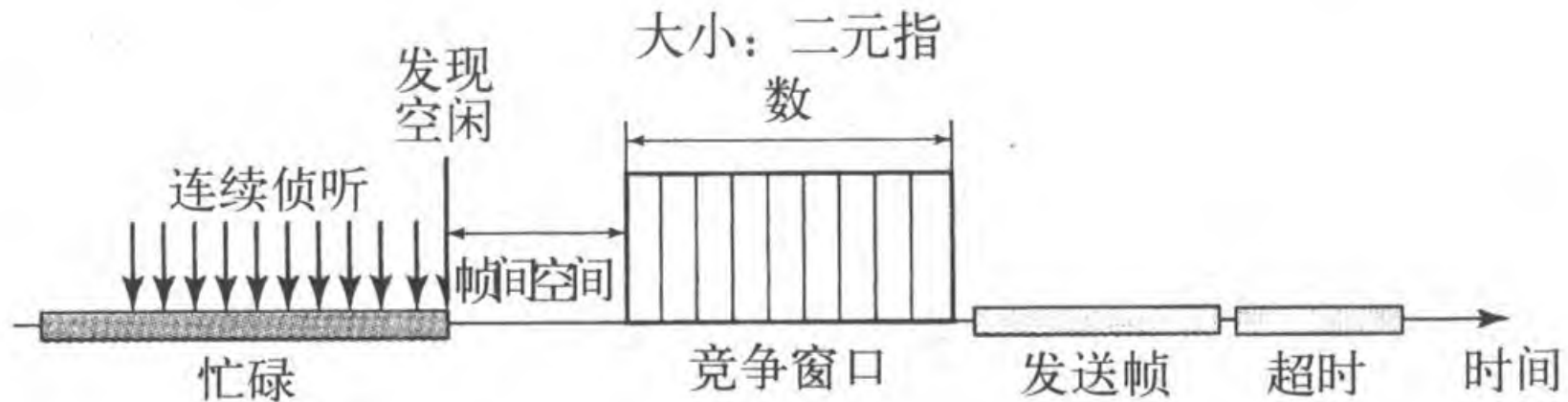
**p** 在有线网络中，接收信号的能量与发送信号的能量基本一样，或者因为线路长度较短，或者因为存在中继器放大了发送方和接收方之间的能量，即在冲突中被检测到的能量几乎翻倍；

**p** 然而，在一个无线网络中，大量发送信号的能量在传输中丢失，接收信号的能量也很小，因此，一个冲突可能只增加5%~10%的额外能量，这对有效的冲突检测起不了什么作用；

**p** 因为无线网络的冲突无法检测到，故而要避免冲突，带冲突避免的载波侦听多路访问（CSMA/CA）就是为这种网络发明的

图12.16 CSMA/CA中的时间安排

pCSMA/CA通过三种方法来避免冲突：帧间间隔，竞争窗口和确认





---

## 帧间间隔（IFS）

---

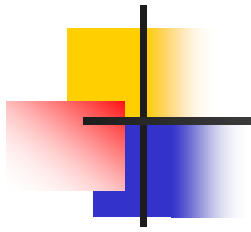
- 首先，通过推迟传输来避免冲突，即使通道是空闲的；
  - 当发现通道空闲时，站点并不马上发送，它会等待一段时间——帧间间隔（interframe space, IFS）；
  - 虽然通道看上去空闲，但一个远端站点可能早已开始传输了，只是远端站点信号还未到达而已；
  - IFS时间使得远端站点传输的信号能到达该站点，如果在IFS时间后，通道仍然空闲，那么站点就能发送了，但是它仍然要等待一段时间，这段时间等于竞争时间；
  - IFS变量也能用来区分站点或帧的优先次序，一个站点被赋值的IFS越短，它的优先权越高。
-

---

## 竞争窗口

---

- p 竞争窗口将时间分割成时隙，预备发送的站点选择一个随机时隙作为等待的时间；
- p 窗口中时隙的数量是根据补偿策略中的二元指数而变化的，即第一次设定一个时隙，以后每次站点在IFS时间后无法检测到空闲通道时，时隙的数量都要翻番；
- p 有趣的一点：每次时隙后站点都要侦听通道，然而如果站点发现通道繁忙，它并不重启程序，而是停止定时器并当侦听到通道空闲时再重启（恢复）定时器（这给了站点最长等待时间的优先权？）。



在CSMA/CA 中，如果站点发现通道繁忙，它并不重启竞争窗口的时钟，而是停止定时器并当通道空闲时再重启（恢复）定时器

---

## 确认

---

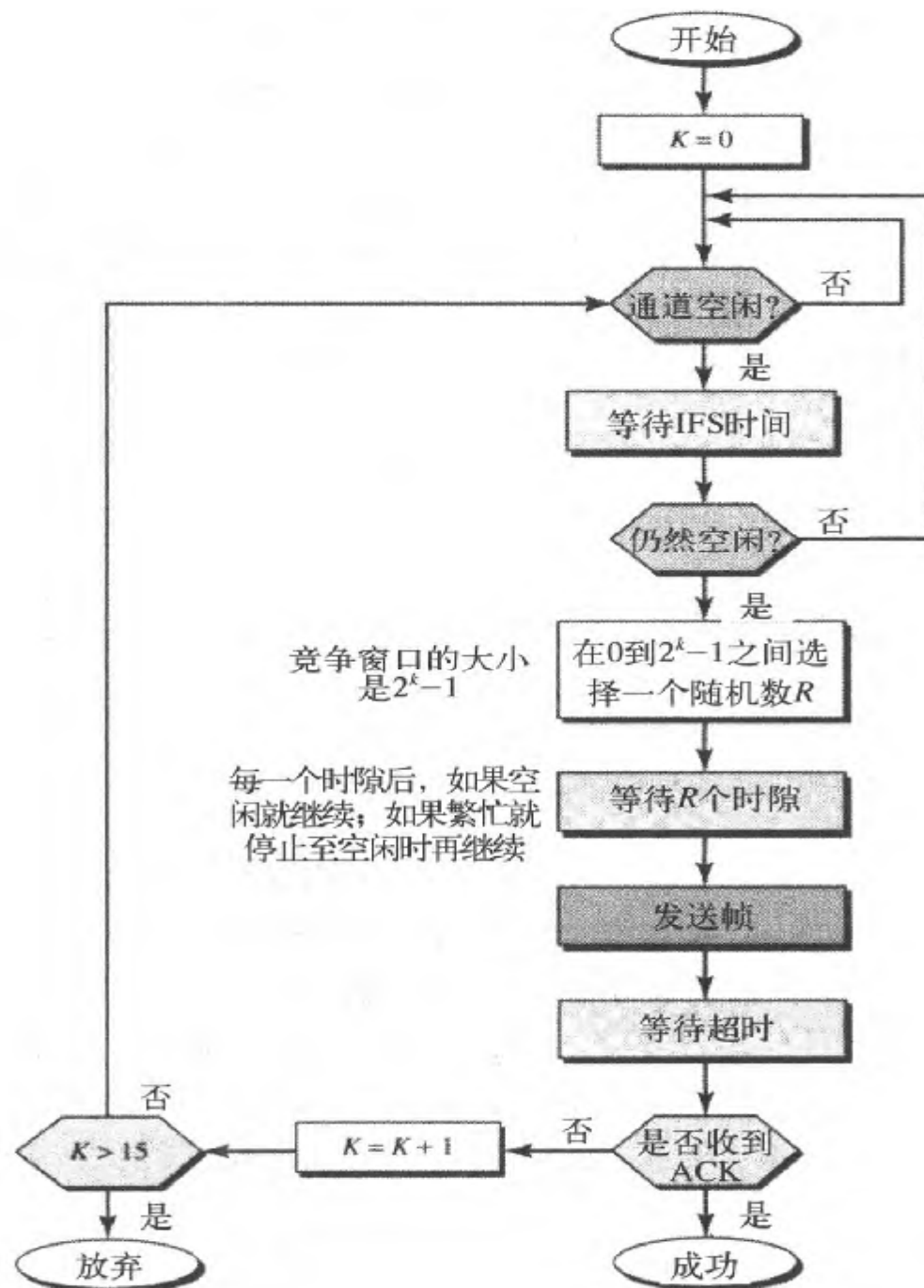
- p 仍然有可能产生冲突而导致帧的损坏;
- p 另外, 数据也可能在传输中被损坏;
- p 肯定的确认和定时器的超时有助于确保接收方收到帧。

图12.17 CSMA/CA流程图

**p**通道在IFS之前和之后都要被侦听；

**p**在竞争时间内，通道也要被侦听；

**p**在竞争窗口中的每一个时隙内，通道都要被侦听；如果发现通道空闲，定时器继续，如果通道繁忙，定时器停止并在通道空闲后再继续



## 12-2 受控访问协议

- 在受控访问协议中，站点之间相互协商以确定哪一个站有权发送；
- 没有得到授权的站点无权发送数据；
- 主要有三类受控访问协议：预约、轮询和令牌传递

图12.18 预约访问协议

- 站点在发送数据之前需要预约；
- 将时间划分为时隙，在每一个时隙内，在数据帧之前先发送一个预约帧；
- 如果系统中有 $N$ 个站点，在预约帧中就恰好会有 $N$ 个预约子时隙，每个子时隙均属于一个站点；
- 当一个站点想要发送一个数据帧时，它在自己的子时隙中做一个预约，已经预约完的站点可以在预约帧之后发送数据帧。

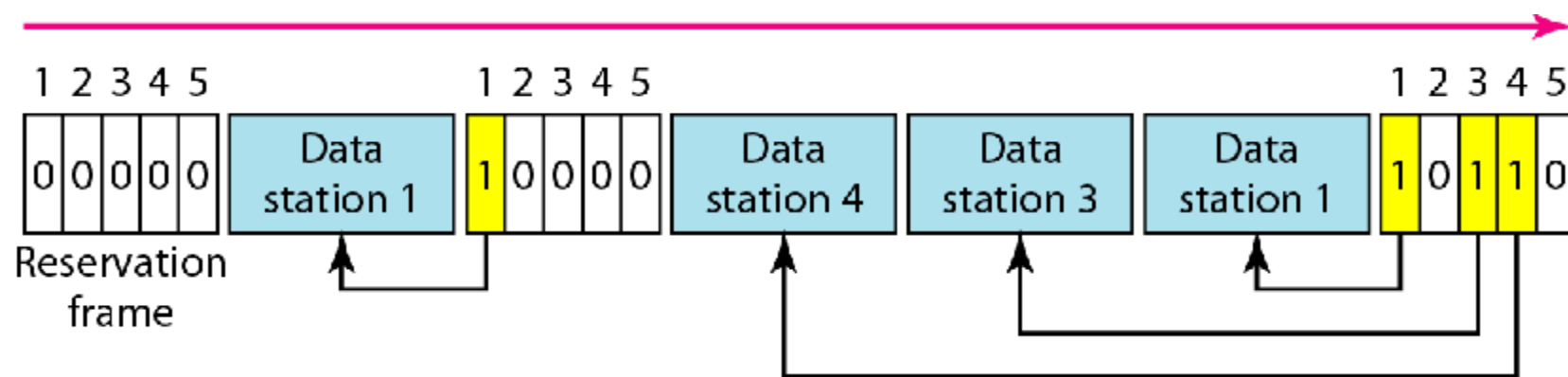
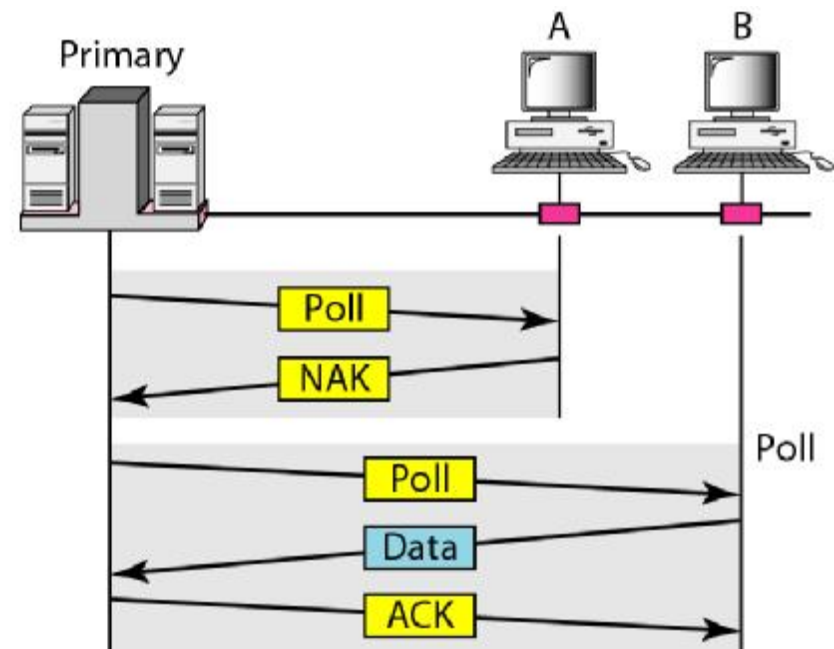
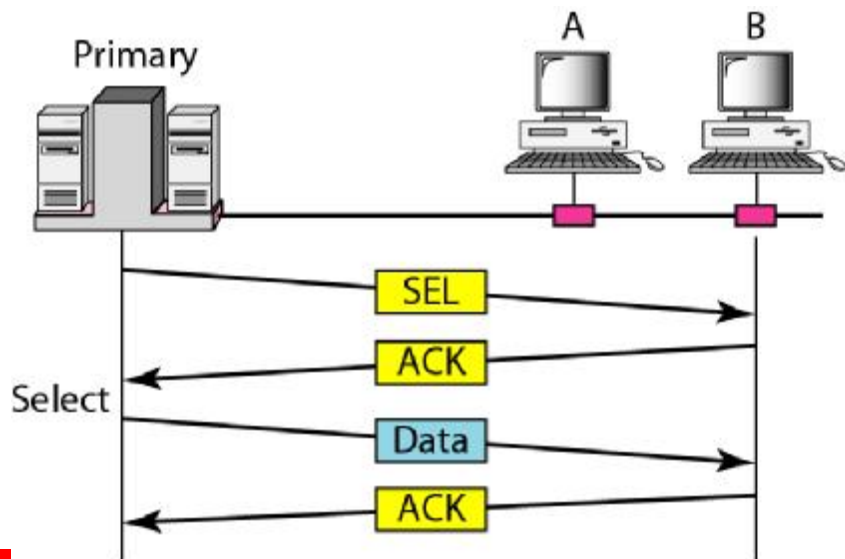


图12.19 在轮询访问方法中的选择和轮询功能

**p**一个设备作为主站，而另一些设备作为从站，所有的数据交换都要通过主站点进行；

**p**主设备控制链路，而从设备只是跟随它的指令；主设备决定在给定的时间内哪个设备被容许使用通道，始终是会话发起者；

**p**如果主设备希望接收数据，它就询问从设备是否有数据要发送（称做轮询）；如果主设备希望发送数据，它就通知从设备准备接收（称做选择）。





---

## p选择

- Ø主设备控制链路，如果主设备既不发送数据也不接收数据，那么链路可用，这时，如果它有数据要发送就可以发送；
- Ø但它不清楚目标设备是否准备接收，所以，主设备必须通知从设备并等待从设备处于准备好状态的确认；
- Ø在发送数据之前，主设备要创建并发送一个选择（SEL）帧，其中的一个字段包含目标从设备的地址

## p轮询

- Ø当主设备准备接收数据时，它必须逐个询问每个设备是否要发送数据；
  - Ø当询问到第一个设备时，如果它没有数据发送，就用NAK帧来响应；如果它要发送数据，就用数据（用数据帧格式）响应；
  - Ø如果响应的是NAK帧，主设备就用同样方法轮询下一个从站点，直到它找到一个要发送数据的站点为止；如果响应是数据帧，主设备就读取该帧并返回一个ACK帧
-

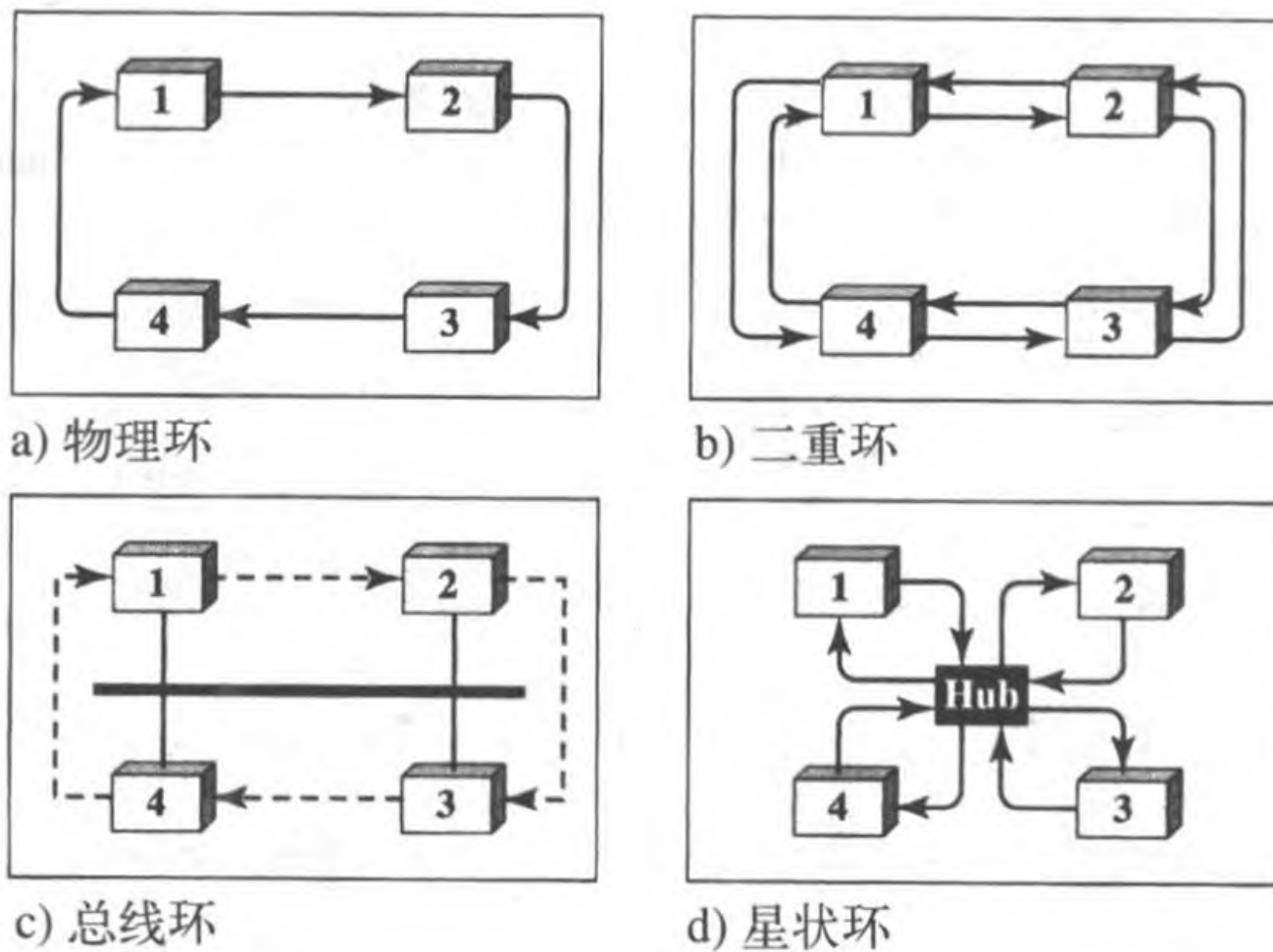
---

## 令牌传递

---

- p** 站点被组织在一个逻辑环中，每一个站点都有前驱和后继，访问权限由前驱站点传递给它，当该站点没有数据再要发送时，便将权限传递给后继站点；
- p** 一个特殊的信息分组被称为令牌（token）在环中循环，令牌的持有赋予站点访问通道并发送数据的权限；
- p** 令牌管理是必需的：站点占有令牌的时间必须得到限制，令牌必须得到监控以确定它没有丢失或损坏；令牌管理的另一个功能是给站点和被传输的数据类型分配优先权；最后，令牌管理能够让低优先权站点将令牌释放给高优先权的站点。

图12.20 在令牌传递访问方法中的逻辑环和物理拓扑结构



## 12-3 通道化

**p**通道化（**channelization**）也是一种多路访问方法，不同的站点间在时间上、频率上或通过编码来共享链路的可用带宽；

**p**三种信道化协议：**FDMA**（频分多路访问）、**TDMA**（时分多路访问）和**CDMA**（码分多路访问）

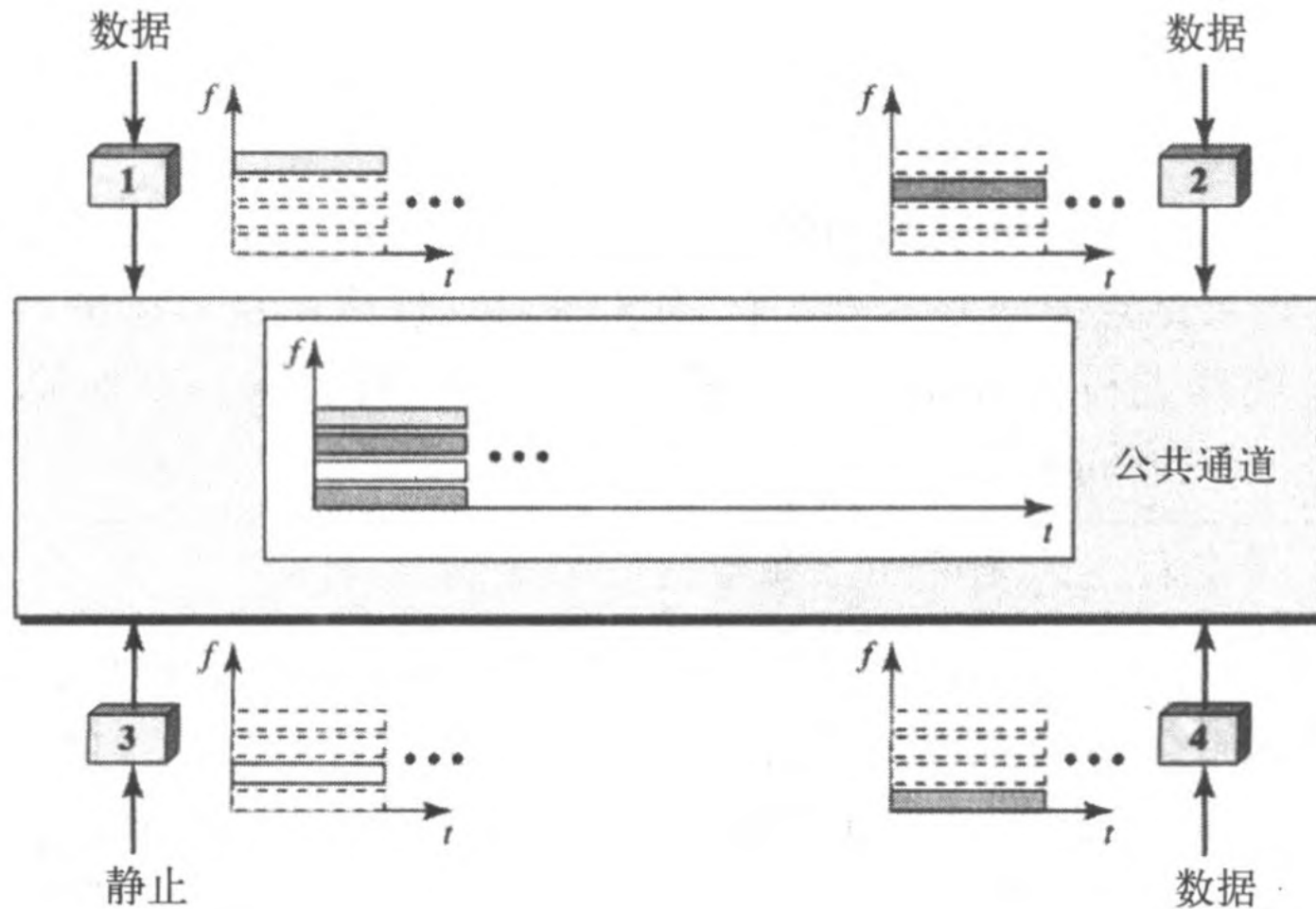
---

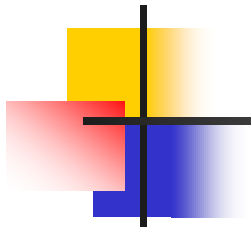
## 频分多路访问（FDMA）

---

- p 可用的带宽被划分成频带，每一个站点都使用分配给它的频带发送数据；
- p 每一个频带都预留给特定的站点使用，该频带永远属于一个站点；
- p 每个站点也使用带通滤波器来限定传输器的频率；
- p 为了防止站点间的冲突，通过小的防护频带将指定的波段彼此分开。

图12.21 频分多路访问（FDMA）





在FDMA中，公共通道的可用带宽被分割成频带并由防护频带将其分离。

---

## FDMA与FDM区别

---

**p**FDM是一种物理层技术，能组合低带宽通道的负载并使用一个高带宽通道来传输它们，这些被组合的通道是低通通道；多路复用器调制信号，将它们组合并产生一个带通信号，每个通道的带宽都被多路复用器转换了；

**p**而FDMA是数据链路层的一种访问方法，每个站点的数据链路层都告知它的物理层把传递给它的数据转换成带通信号；信号必须在被指定的频带中产生，在物理层并没有实际的多路复用器存在；每个站点产生的信号是自动带通过滤的，当它们被发送到公共通道时，它们是混合在一起的。



---

## 时分多路访问（TDMA）

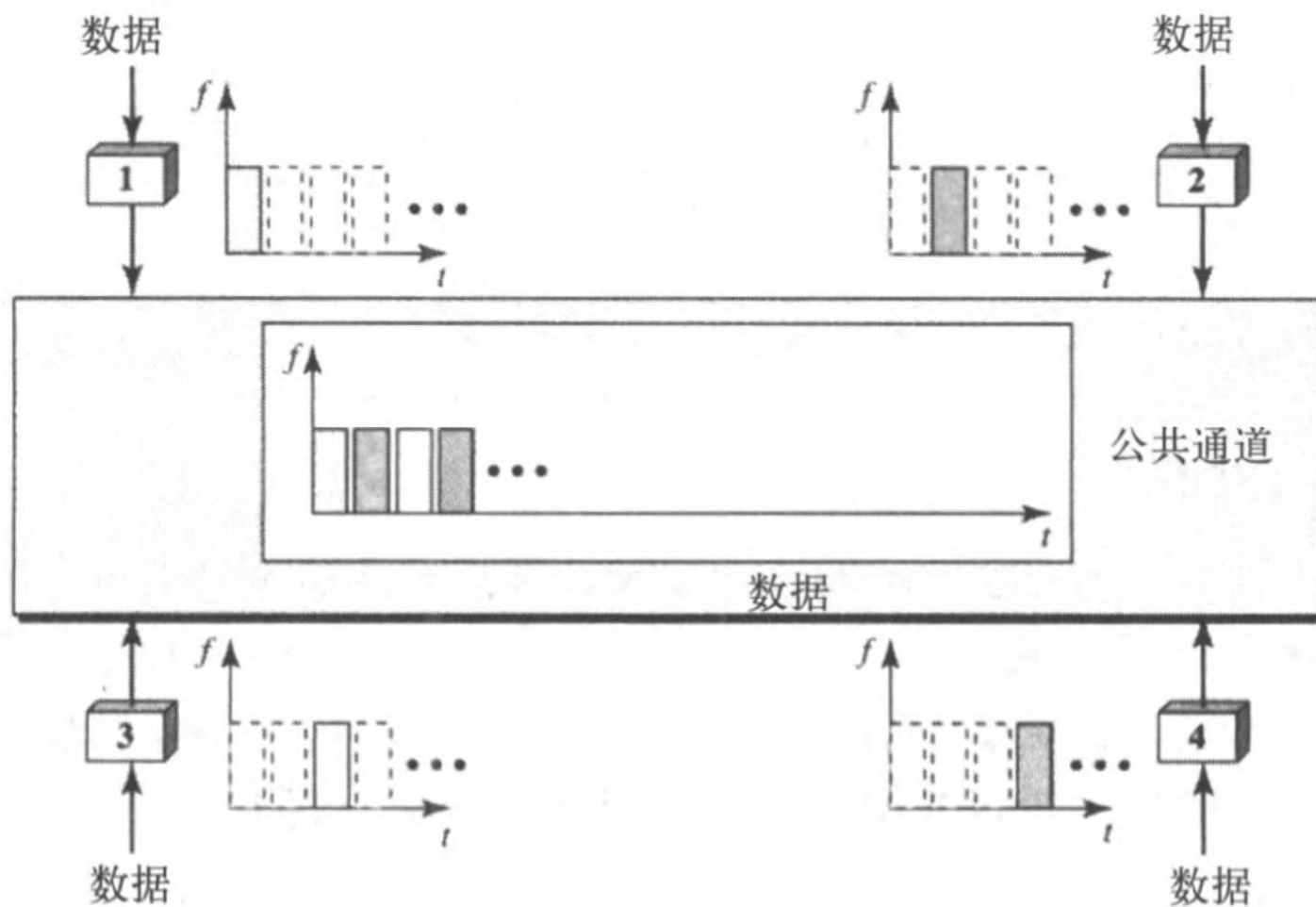
---

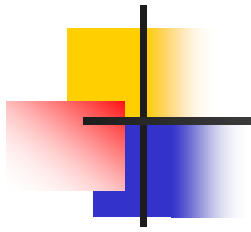
**p**所有的站点在时间上共享通道的带宽，每一个站点被分配一个时隙，只有在这一时隙内它能够发送数据，每个站点在指定的时隙中传输数据；

**p**问题是如何使不同的站点达到同步？每个站点都要知晓自身时隙的开始点和所处的位置；如果站点分布在一个广泛的区域内，那么系统产生的传播延迟会给前述问题带来困难；

**p**为使延迟得到补偿，插入保护时间；通过在时隙开始点插入同步位（通常称为前导位）来完成同步化。

图12.22 时分多路访问（TDMA）





在TDMA中，带宽被视为是时间上共享的一个通道。

---

## TDMA与TDM区别

---

**pTDM**是一种物理层的技术，它将来自低速通道的数据组合起来并使用一个高速通道来传输它们，这个过程使用一个物理多路复用器从每个通道交叉存取数据单元；

**pTDMA**是数据链路层的一种访问方法，每个站点的数据链路层都告知它的物理层使用指定的时隙，在物理层并没有实际的多路复用器存在。

---

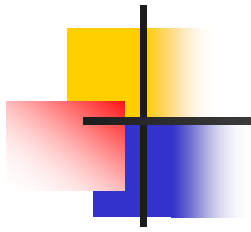
## 码分多路访问（CDMA）

---

**p**码分多路访问（code-division multiple access, CDMA）概念在几十年前就已经提出，电子技术的新发展使得它的实现成为可能；

**p**与FDMA不同，CDMA 仅有一个通道占据链路整个的带宽；

**p**而与TDMA不同的是，所有的站点都可以同时发送数据，没有时间的共享问题。



**CDMA中，一个通道同时承载所有的传输。**

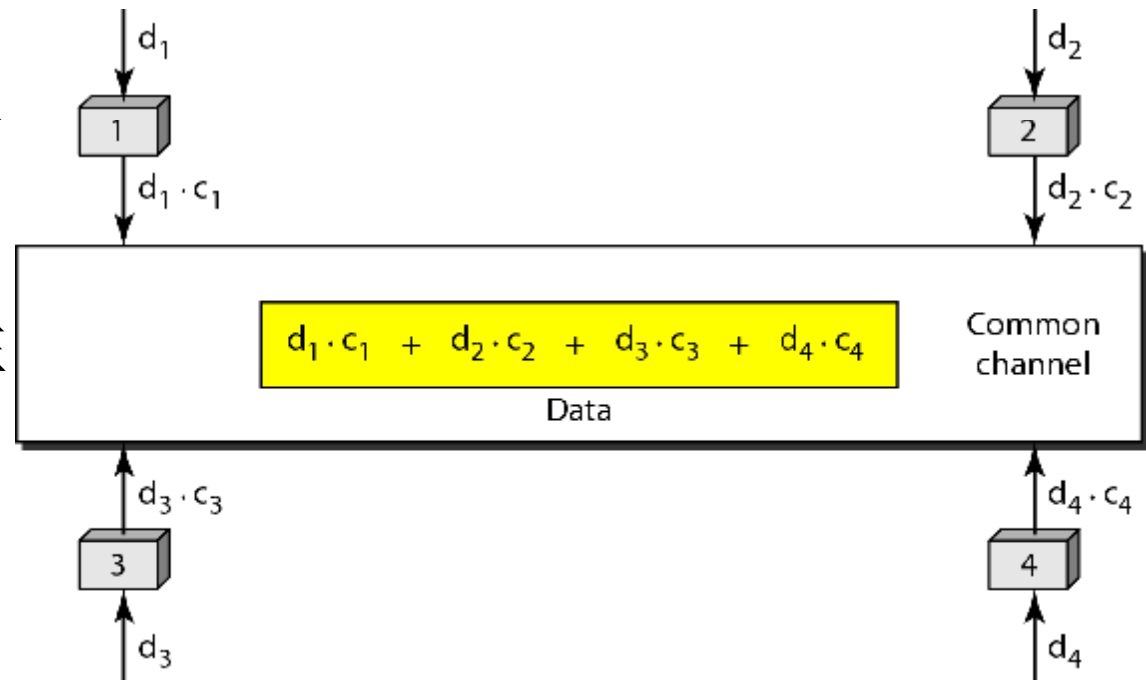
图12.23 用不同编码通信的简单思想

假定被分配的编码有两个特性：

1. 两个（不同的）编码相乘得0；
2. 编码自身相乘得4（站点的数量）

每个站点将它的数据与编码相乘，通道上的数据是所有这些结果的总和，任何站点想要接收的数据是发送方的编码和数据的乘积；

假定站点1和站点2正在通话，站点2想知道站点1在说什么，它在通道中将数据和站点1的编码 $c_1$ 相乘再除以4（P257）



## 图12.24 正交化的码片序列

**p** 每个站点被指定一个编码，编码是一个被称之为芯片（码片）的数字序列；

**p** 序列的选择并不是随机的，它们被称为正交序列到（orthogonal sequences），有以下的特性：

Ø1. 每个序列有N个元素，而N正是站点的数量；

Ø2. 如果将两个相同序列的元素逐个相乘，并将结果相加，结果为N；

Ø3. 如果将两个不同序列的元素逐个相乘，并将结果相加，结果为0





---

## 图12.25 CDMA中数据的表示方法（数据编码）

---

- 如果一个站点发送0位，编码为-1；
- 如果一个站点发送1位，编码为+1；
- 当站点空闲时，它不发送信号，编码为0



---

## 编码和解码

---

- p 假定站点1和站点2正在发送0位，站点4正发送1位，而站点3是静止的，发送方的数据被依次编码为-1， -1， 0和+1；
- p 每个站点将相应的数与它的芯片序列（正交序列）相乘，这对每个站点来说都是唯一的，结果是产生一个新的序列并被发送到通道上；
- p 假定所有的站点在同一时点发送结果序列，通道上的序列是所有四个序列的总和；
- p 假定静止的站点3正在“听”站点2，站点3将通道上的数据乘以站点2的编码[+1 -1 +1 -1]，得到：

$$[-1 -1 -3 +1] \times [+1 -1 +1 -1] = -4/4 = -1 \Rightarrow \text{位0 (P258)}$$

图12.26 CDMA中的共享通道

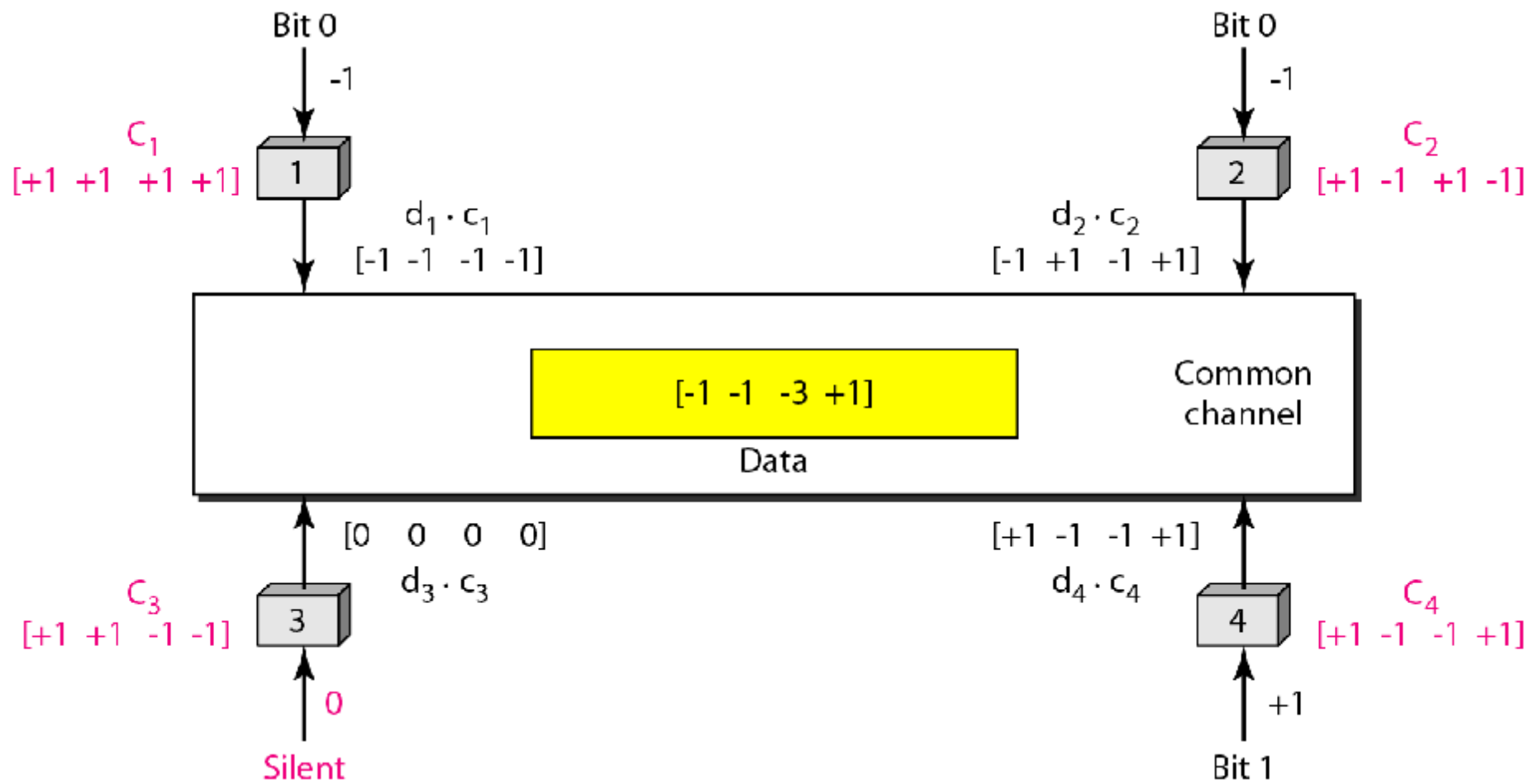
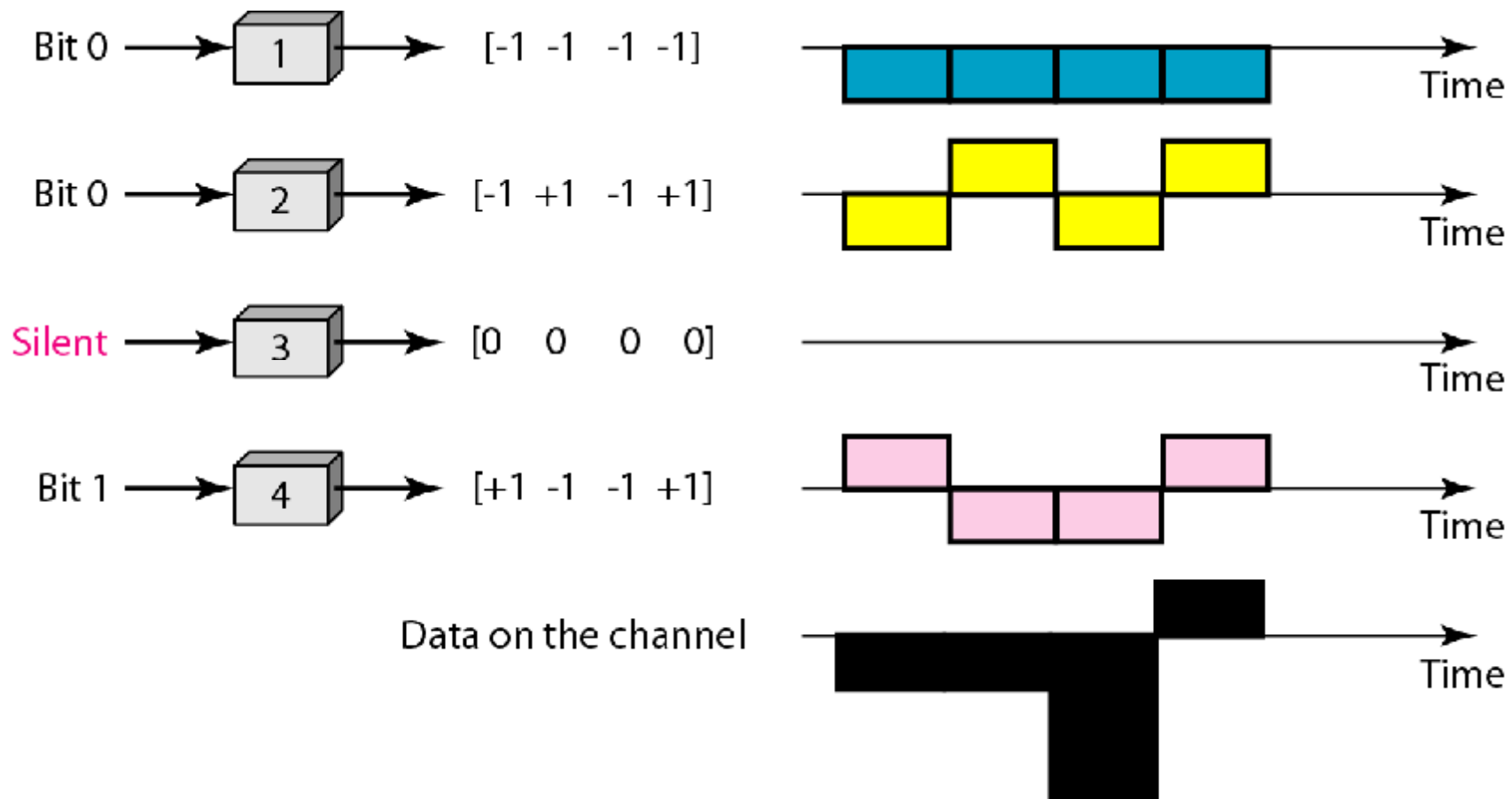
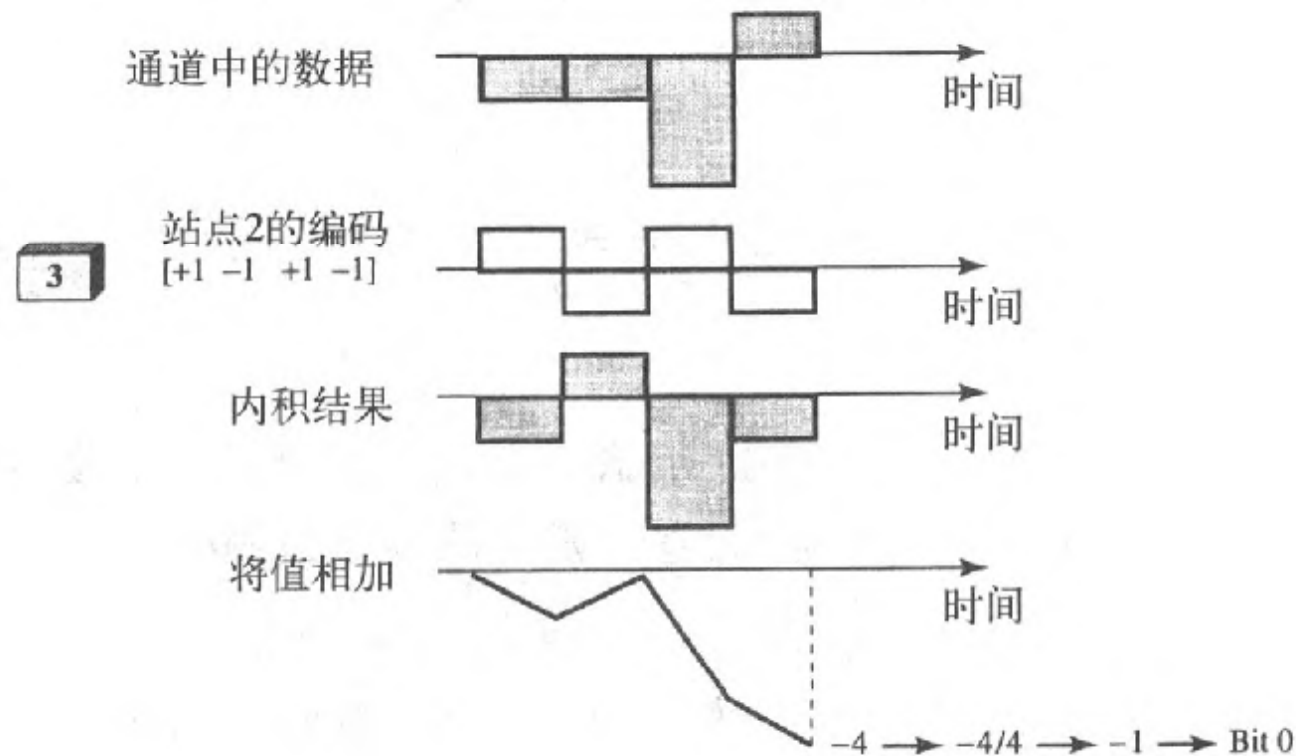


图12.27 在CDMA中四个站点产生的数字信号（NRZ-L）



## 图12.28 CDMA中合成信号的解码

**p** 通过使用站点2的编码，站点3可以检测到来自站点2的数据



---

## 序列的产生

---

**p**通过使用例如Walsh表的正交编码方法来产生序列。