计算机网络-复习

> 2024/12/18~12/21
> 重点是老师划的，例题和答案是书上找的。太长的答案用 GPT 总结了一下，总体还算看得过去。

提纲

题型

题型	题量	分值
单选	10	20
填空	5	10
判断	10	20
简答	3	15
计算	3	35

范围

简答

1-21、1-24、1-03、
3-4、CSMA/CD 原理、
4--3、4-04、4-07、TCP 与 UDP 区别、
分组交换与报文交换的区别、TCP 三报文握手、四报文握手、
熟悉常见网络设备的功能和所属的协议层。

计算

各种编码波形的绘画、CRC 计算、拥塞窗口 cwnd、ssthresh 计算（需要掌握 243 页例题和对应的课后题目）、236 页发送窗口、可用窗口计算例题和对应的课后题、路由器路由表的转发课后题【给定路由表，求转发端口】

其他

看题库

简答题

【1-03】 试从多个方面比较电路交换、报文交换和分组交换的主要优缺点。

1. 电路交换的特点与优缺点
   - 特点：通信必须经过三个阶段：建立连接、通信过程、释放连接。在通信期间，通信双方始终占用固定的物理信道。
   - 优点：
     - 静态分配传输带宽，一旦连接建立，传输带宽不会改变，通信质量稳定。
     - 已经占用的通信资源不受网络其他用户的影响，即使网络拥塞，已建立的连接仍可正常通信。
   - 缺点：
     - 由于计算机数据是突发性的，从通信线路的利用率来考虑，电路交换的效率比较低。
     - 若任何链路资源不足或出现故障，连接无法建立或会中断，重新通信需再次建立连接。
2. 报文交换的特点与优缺点
   - 特点：采用存储转发技术，将整个报文在节点存储后再转发，不进行分组划分。
   - 优点：
     - 省去了分组和重组的过程，降低了一些处理开销。
   - 缺点：
     - 灵活性不如分组交换。
     - 数据传输的时延较大。
     - 已较少使用，目前主要用于电报传输，应用场景有限。
3. 分组交换的特点与优缺点
   - 特点：以分组为传输单位，采用存储转发技术，无需建立和释放连接，传输过程中动态分配带宽，链路逐段占用。
   - 优点：
     - 动态分配带宽，能合理高效利用各链路的传输能力。
     - 采用分布式的路由选择协议，当某节点或链路故障时，可以动态调整传输路径，网络具有较好的生存性。
   - 缺点：
     - 分组存储转发时需要排队，可能导致一定的时延。
     - 端到端的传输带宽无法保证，通信量突增可能导致网络拥塞，甚至瘫痪。
     - 每个分组需携带控制信息，增加了一定的开销。
     - 路由器管理功能相对较弱，需专门的主机和管理软件进行网络管理。

【1-21】 协议与服务有何区别？有何关系？

1. 协议的定义与特点
   - 定义：协议是控制两个对等实体（或多个实体）进行通信的规则集合，包括语法（交换信息的格式）和语义（发送者或接收者需要完成的操作）。

- 特点：
  - 水平性：协议是水平的，用于对等实体间的通信控制。
  - 透明性：协议对使用它的上层实体是透明的，上层只关心服务，不关心协议的实现细节。

2. 服务的定义与特点
   - 定义：服务是下层向上层通过层间接口提供的功能集合。
   - 特点：
     - 垂直性：服务是垂直的，由下层向上层提供，体现为跨层接口的功能。
     - 可见性：只有那些能够被上层实体“看得见”的功能才称为服务。并非在一个层内完成的所有功能都算作服务。
3. 协议与服务的关系
   - 协议的实现保证了向上一层提供服务。协议定义了对等实体间通信的规则，而服务是协议的结果，体现为提供给上层的功能。

【1-24】 试述具有五层协议的网络体系结构的要点，包括各层的主要功能。

解答：我们知道，OSI 的体系结构是七层协议。TCP/IP 的体系结构是四层协议，而真正有具体内容的只是上面三层。在学习计算机网络的原理时往往采取折中的办法，即综合 OSI 和 TCP/IP 的优点，采用一种只有五层协议的体系结构。图 T-1-24 给出了五层协议的结构。

图 T-1-24　五层协议的结构

1. 物理层
   - 功能：
     - 负责在物理媒体上传输比特流，传输数据的基本单位是比特（bit）。
     - 提供透明的比特流传输，不关心比特所代表的意义。
     - 定义硬件特性，如连接器的插头、引脚数量及其连接方式等。
   - 关键点：
     - 传输介质（如双绞线、光纤）位于物理层协议的下层（可视为第 0 层）。
2. 数据链路层
   - 功能：
     - 在相邻结点之间提供可靠的数据帧传输，传输数据的单位是帧（frame）。
     - 负责组装和拆解帧，加入必要的控制信息，如同步信息、地址信息、差错控制等。
     - 提供差错检测功能，对检测到错误的帧直接丢弃，避免浪费网络资源。

- 关键点：
  - 提供点对点的通信服务。
  - 如果需要纠正错误，则由更高层的运输层完成。

3. 网络层
   - 功能：
     - 负责不同主机之间的数据传输，数据单位是分组或包（packet）。
     - 通过路由选择功能，确保分组能够通过网络找到目的主机。
     - 封装运输层的数据为分组，并将其交付到正确的目的地。
   - 关键点：
     - 使用 IP 协议时，分组称为 IP 数据报或数据报（datagram）。
     - 在广播信道中，路由选择的问题很简单，网络层也简单到可以不存在。
4. 运输层
   - 功能：
     - 为主机内两个进程间的通信提供端到端的服务，数据单位是报文段（segment）或用户数据报。
     - 负责复用和分用功能：
       - 复用：多个应用进程可以共享运输层服务。
       - 分用：运输层将收到的数据传递给对应的应用进程。
     - 提供两种主要协议：
       - TCP：面向连接，提供可靠传输。
       - UDP：无连接，提供尽最大努力交付的服务。
   - 关键点：
     - 可靠性由 TCP 提供，而 UDP 提供较快但不保证可靠的通信。
5. 应用层
   - 功能：
     - 直接为用户的应用进程提供服务，数据单位依具体协议而定。
     - 包含多种应用协议：
       - HTTP：支持万维网浏览。
       - SMTP：支持电子邮件服务。
       - FTP：支持文件传输服务。
   - 关键点：
     - 是用户与网络交互的接口。
     - 应用层协议种类多样，满足不同的应用需求。

【3-04】 数据链路层的三个基本问题（封装成帧、透明传输和差错检测）为什么都必须加以解决？

> 这个答案看起来更像是“如何解决”

1. 封装成帧

   封装成帧就是在一段数据的前后分别添加首部和尾部（在首部和尾部里面有许多必要的控

制信息），这样就构成了一个帧。接收端在收到物理层上交的比特流后，就能根据首部和尾部的标记，从收到的比特流中识别帧的开始和结束。

2. 透明传输

   透明传输就是上层交下来的数据，不管是什么形式的比特组合，都必须能够正确传送。由于帧的开始和结束的标记是使用专门指明的控制字符，因此，所传输的数据中的任何比特组合一定不允许和用作帧定界的控制字符的比特编码一样，否则就会出现帧定界的错误。数据链路层不应当对要传送的数据提出限制，即不应当规定某种形式的比特组合不能够传送。

3. 差错检测

   - 当数据链路层缺少差错检测时，即使数据在传输过程中出现错误，这些错误的数据帧仍会通过网络中的所有结点，直到到达目的主机。目的主机的高层软件需要对收到的数据进行差错检测。如果检测到错误，高层软件会请求源主机重传数据。
   - 如果数据链路层有差错检测，它可以在帧到达目的主机之前检测出错误，并立即丢弃错误的帧。

CSMA/CD 的工作原理

`CSMA/CD` 的意思是 `载波监听多点接入/碰撞检测` ，用于在以太网中协调各计算机在总线上的工作。

它的工作原理是不管在发送前，还是在发送中，每个站都必须不停地检测信道上的电压变化。如果适配器检测到信号电压变化幅度超过一定的门限值时，就认为总线上至少有两个站点在同时发送，需要立即停止发送，然后等待一段随机时间后再次发送。

【4-03】 作为中间设备，转发器、网桥、路由器和网关有何区别？

解答：将网络互相连接起来要使用一些中间设备。根据中间设备所在的层次，可以有以下四种不同的中间设备：

1. 物理层使用的中间设备叫做转发器。
2. 数据链路层使用的中间设备叫做网桥或桥接器。
3. 网络层使用的中间设备叫做路由器。
4. 在网络层以上使用的中间设备叫做网关。用网关连接两个不兼容的系统需要在高层进行协议的转换。

【4-04】 试简单说明下列协议的作用：IP，ARP，RARP 和 ICMP。

解答：

- 网际协议 IP（Internet Protocol）

  使用 IP 协议就可以把互连以后的计算机网络看成是一个虚拟互连网络。所谓虚拟互连网络也就是逻辑互连网络，或称为互联网。我们知道，各种物理网络的异构性本来是客观存在的，但是我们利用 IP 协议就可以使这些性能各异的网络在网络层上看起来好像是一个统一的网络。这种使用 IP 协议的虚拟互连网络可简称为 IP 网。使用 IP 网的好处是：

当 IP 网上的主机进行通信时，就好像在一个单个网络上通信一样，它们看不见互连的各网络的具体异构细节（如具体的编址方案、路由选择协议，等等）。

- 地址解析协议 ARP（Address Resolution Protocol）
  用来把一个机器（主机或路由器）的 IP 地址转换为相应的物理地址（或硬件地址）。
- 逆地址解析协议 RARP（Reverse Address Resolution Protocol）
  和 ARP 相反，用来把一个机器（主机或路由器）的物理地址（或硬件地址）转换为相应的 IP 地址。
- 网际控制报文协议 ICMP（Internet Control Message Protocol）
  用来使主机或路由器报告差错情况和提供有关异常情况的报告，这样就可以更有效地转发 IP 数据报和提高交付成功的机会。

【4-07】 试说明 IP 地址与硬件地址的区别。为什么要使用这两种不同的地址？

从层次的角度看，物理地址是数据链路层和物理层使用的地址，而 IP 地址是网络层和以上各层使用的地址，是一种逻辑地址（称 IP 地址是逻辑地址是因为 IP 地址是用软件实现的）。

由于全世界存在着各式各样的网络，它们使用不同的硬件地址。要使这些异构网络能够互相通信就必须进行非常复杂的硬件地址转换工作，因此由用户或用户主机来完成这项工作几乎是不可能事。但统一的 IP 地址把这个复杂问题解决了。连接到互联网的主机只需拥有统一的 IP 地址，它们之间的通信就像连接在同一个网络上那样简单方便。当需要把 IP 地址转换为物理地址时，调用 ARP 的复杂过程都由计算机软件自动进行，而用户是看不见这种调用过程的。因此，在虚拟的 IP 网络上用 IP 地址进行通信给广大的计算机用户带来很大的方便。

TCP 与 UDP 的区别

1. 连接
   - TCP 是面向连接的运输层协议，使用前需要建立连接，传输完毕后需要释放连接
   - UDP 是无连接的，发送数据前不需要建立连接
2. 可靠
   - TCP 提供可靠交付的服务
   - UDP 尽最大努力交付，即不保证可靠交付
3. 面向
   - TCP 面向字节流。TCP 把应用程序交下来的数据仅仅看成是一连串的无结构的字节流，它不保证所传输的数据块具有大小对应关系，但是会保证字节流完全一样。
   - UDP 面向报文。UDP 对应用层交下来的报文会保留其边界，一次交付一个完整的报文。
4. 数量
   - 每一条 TCP 连接只能有两个端点，每一条 TCP 连接只能是一对一的。
   - UDP 支持一对一、一对多、多对一和多对多的交互通信。
5. 拥塞
   - TCP 提供全双工通信。TCP 允许通信双方的应用进程在任何时候都能发送数据。TCP 连接的两端都设有发送缓存和接收缓存，用来临时存放双向通信的数据。

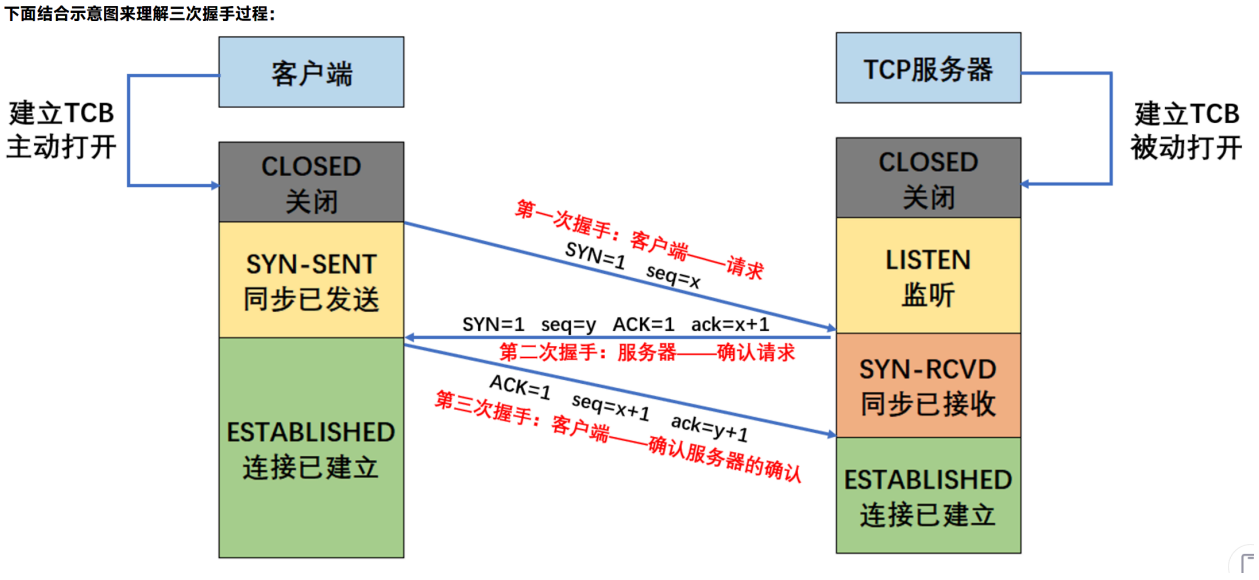
- UDP 没有拥塞控制。

6. 其他
   - UDP 的首部开销小，只有 8 个字节，比 TCP 的 20 个字节的首部要短。
   - TCP 适用于对可靠性要求高的场景，UDP 适用于对实时性要求高的场景。

分组交换与报文交换的区别

对比维度	报文交换	分组交换
单位	整个报文	分组
转发方式	存储整个报文后转发	分组逐段转发
路径选择	固定路径	动态选择路径
传输效率	整体传输，效率较低	动态分组，效率较高
应用现状	几乎淘汰	广泛使用

TCP 三报文握手



补充：为什么要使用三次握手机制？

1.为了阻止历史连接

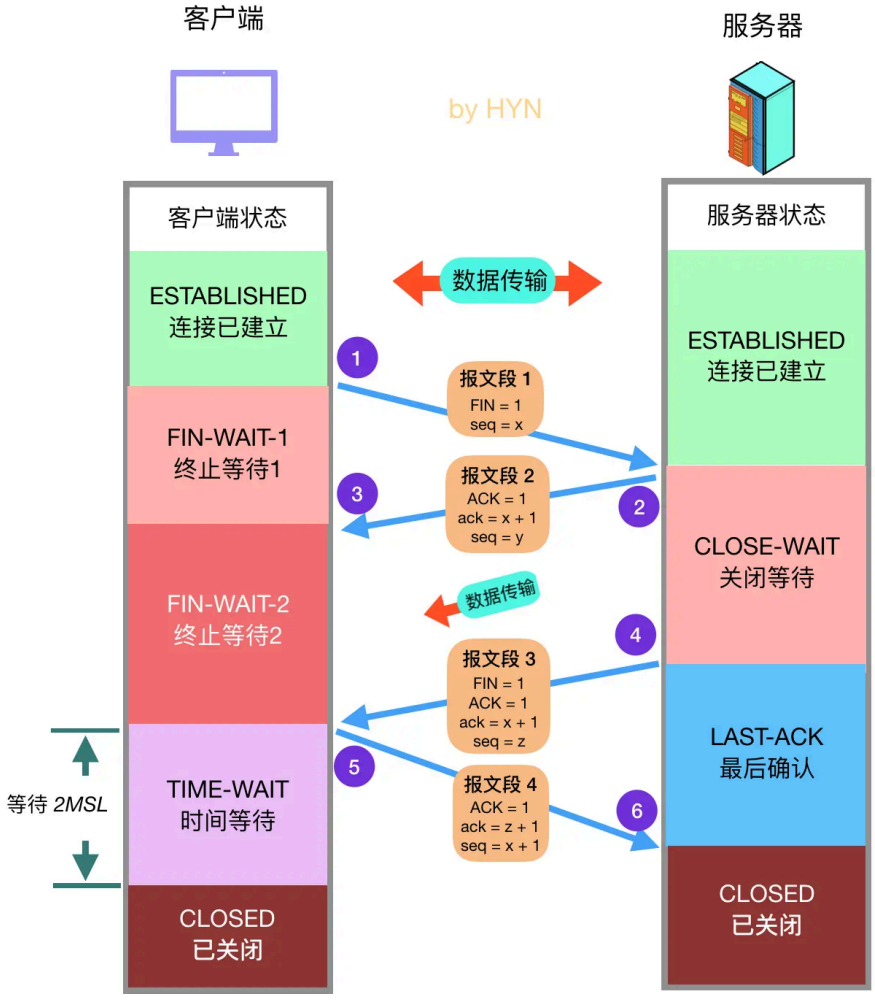
若网络状况不好的时候，客户端发送的SYN信号未能及时传输给服务器。当网络恢复时已经失效的SYN信号又到达服务器。如果只有两次握手便可建立连接，那么此时客户端就不知道这个连接是不是已经失效了的历史连接，从而导致错误的发生。三次握手时，客户端便可以根据上下文来判断此次连接是否为历史连接，避免错误的发生。

2. 为了避免服务器开启无用连接增加服务器开销

客户端设置了一个超时时间，超过了就重新发送一个TCP连接请求。如果没有第三次握手的话，服务端是不知道客户端是否收到服务返回的信息的，这样没有给服务器端一个创

建还是关闭连接端口的请求，服务器端的端口就一直开着，等到客户端因超时重新发出请求时，服务器就会重新开启一个端口连接。那么服务器端上没有接收到请求数据的上一个端口就一直开着，长此以往，这样的端口多了，就会造成服务器端开销的严重浪费。

TCP 四报文握手



四次挥手过程

熟悉常见网络设备的功能和所属的协议层

> 没找到标准答案，以下内容来自 ChatGPT

设备	功能	所属协议层
集线器（Hub）	- 提供多个端口，用于连接局域网内的设备。 - 采用广播方式传输数据，无法分辨数据的目标地址，所有设备均收到同一数据包。	物理层（Physical Layer）
交换机（Switch）	- 提供多端口连接，建立交换表，实现基于MAC地址的帧转发。	数据链路层（Data Link Layer）

设备	功能	所属协议层
	- 提高局域网的效率，通过独立通信通道减少冲突。	
网桥（Bridge）	- 连接两个局域网或分隔同一局域网的不同部分。 - 根据MAC地址过滤和转发数据帧。	数据链路层（Data Link Layer）
路由器（Router）	- 根据IP地址选择路由，实现网络之间的互联。 - 使用路由协议动态选择最优路径，支持分组转发。	网络层（Network Layer）
网关（Gateway）	- 实现不同协议或不同网络体系结构之间的通信转换，如将IPX与IP协议互通。 - 处理上层协议的翻译或封装。	网络层以上
调制解调器（Modem）	- 将数字信号转换为模拟信号用于传输，或将模拟信号转换回数字信号供计算机处理。	物理层（Physical Layer）
无线接入点（Access Point，AP）	- 提供无线连接服务，充当无线设备与有线网络之间的桥梁。 - 支持Wi-Fi协议实现无线网络覆盖。	数据链路层（Data Link Layer）
防火墙（Firewall）	- 检查网络流量的包头，控制数据包的转发、丢弃或过滤。 - 实现安全策略管理，如基于IP地址、端口号或协议的访问控制。	网络层（Network Layer）及传输层（Transport Layer）
中继器（Repeater）	- 放大信号以延长网络传输距离，适用于物理层信号的转发。	物理层（Physical Layer）
网卡（NIC，Network Interface Card）	- 实现计算机与网络的物理连接。 - 负责帧的接收、发送及MAC地址识别。	数据链路层（Data Link Layer）

计算题

各种编码波形的绘画、CRC 计算、拥塞窗口 cwnd、ssthresh 计算（需要掌握 243 页例题和对应的课后题目）、236 页发送窗口、可用窗口计算例题和对应的课后题、路由器路由表的转发课后题【给定路由表，求转发端口】

常用的编码方式和波形图

P44

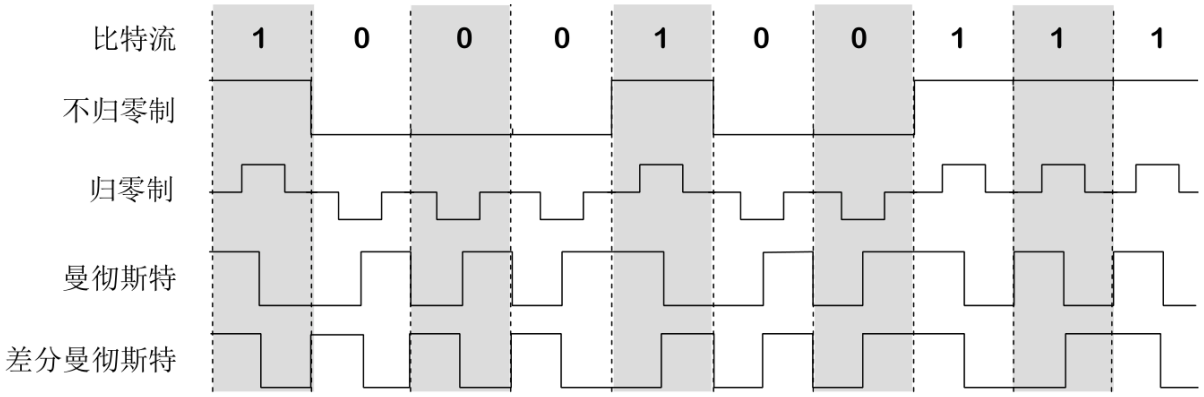


图 2-2　数字信号常用的编码方式

- 不归零制：正电平代表 1，负电平代表 0。
- 归零制：正脉冲代表 1，负脉冲代表 0。
- 曼彻斯特编码：位周期中心的向上跳变代表 0，位周期中心的向下跳变代表 1。
- 差分曼彻斯特编码：在每一位的中心处始终都有跳变。位开始边界有跳变代表0，而位开始边界没有跳变代表 1。

> **编码的诞生背景**
>
> 来自信源的信号常称为基带信号（即基本频带信号）。像计算机输出的代表各种文字或图像文件的数据信号都属于基带信号。基带信号往往包含有较多的低频成分，甚至有直流成分，而许多信道并不能传输这种低频分量或直流分量。为了解决这一问题，就必须对基带信号进行调制。
>
> 调制可分为两大类。一类是仅仅对基带信号的波形进行变换，使它能够与信道特性相适应。变换后的信号仍然是基带信号。这类调制称为基带调制。由于这种基带调制是把数字信号转换为另一种形式的数字信号，因此大家更愿意把这种过程称为编码（coding）。

CRC 计算

P74

前置知识

1. 在数据链路层广泛使用了 循环冗余检验 CRC （Cyclic Redundancy Check）来进行差错检测。
2. 在发送端，先把数据划分为组，假定每组 k 个比特。现假定待传送的数据 M = 101001(k = 6)。CRC 运算就是在数据 M 的后面添加供差错检测用的 n 位冗余码，然后构成一个帧发送出去，一共发送（k + n）位。
3. 接收端取出这个（k + n）位数，用 模 2 运算 除以收发双方事先商定的长度为（n + 1）位的除数 P，得出商是 Q 而余数是 R（n 位，比 P 少一位）。

模 2 运算

模 2 运算不考虑进位和借位。

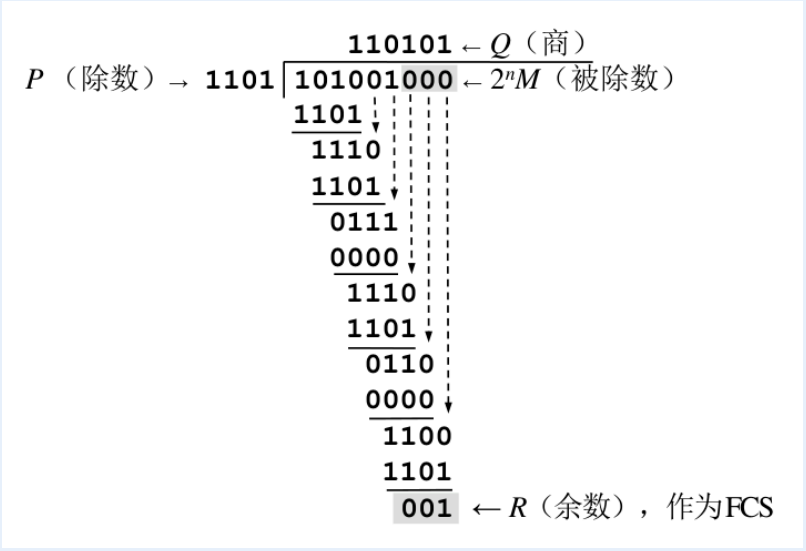
加减乘：

0+0=0；1+1=0；0+1=1；1+0=1；

0-0=0；1-1=0；0-1=1；1-0=1；

0 * 0 = 0； 0 * 1 = 0； 1 * 0 = 0； 1 * 1 = 1；

除法：



当除数和被除数位数相同，商为 1，否则为 0，余数的计算遵循减法规则。

4. 用余数 R 来判断有无差错
   - 如果在传输过程中无差错，那么经过 CRC 检验后得出的余数 R 肯定是 0。
   - 如果出现误码，那么余数 R 仍可能等于零，但概率非常非常小。
5. 一般用生成多项式 P（X）来表示除数 P，比如 `P(X)=X^3+1` 表示 P=1001。

   展开多项式 $P(X)$：

   $$P(X) = X^3 + 0 \cdot X^2 + 0 \cdot X^1 + 1 \cdot X^0$$

   - X^3：系数为 1，对应第 3 位为 1。
   - X^2：系数为 0，对应第 2 位为 0。
   - X^1：系数为 0，对应第 1 位为 0。
   - X^0：系数为 1，对应第 0 位为 1。

   转换为二进制：将系数从高次幂到低次幂排列：

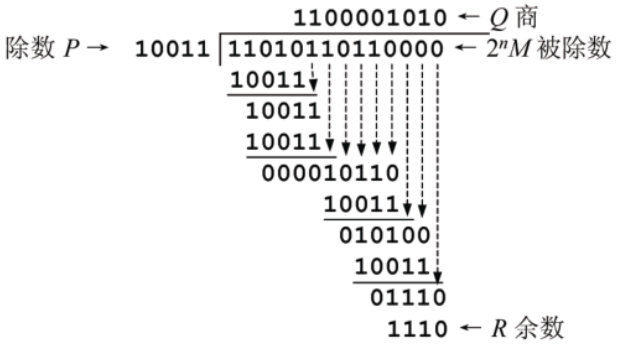
   $$P = 1001$$

例题【3-07】

上面的知识是关于如何使用冗余码进行差错检测，但在计算题中，往往是给出 M 和 P，求解冗余码。解题步骤是先求出 P，得到 n，然后在 M 后面加上 n 个 0，再用模 2 运算除以 P，得到的余数就是冗余码。如果余数的位数不足 n，就在前方补 0。

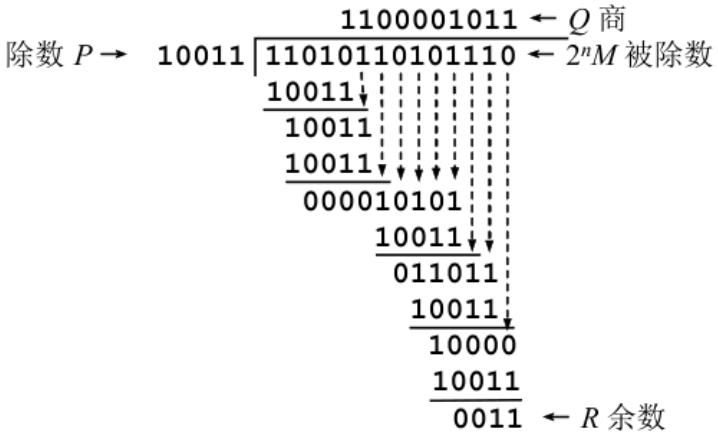
1. 要发送的数据为 1101011011。采用 CRC 的生成多项式是 P（X）= X^4 + X + 1。试求应添加在数据后面的余数。

- 采用 CRC 的生成多项式是 P(X) = X 4 + X + 1，用二进制表示就是 P = 10011
- 现在除数是 5 位，n=4，因此在数据后面添加 4 个 0 就得出被除数：11010110110000



- 除法运算得出的余数 R 就是应当添加在数据后面的检验序列：1110。

2. 数据在传输过程中最后一个 1 变成了 0，问接收端能否发现？
   - 现在数据在传输过程中最后一个 1 变成了 0，即 1101011010。
   - 然后把检验序列 1110 接在数据 1101011010 的后面，再次相除。



   - 余数 R 不为零，因此判定所接收的数据有差错。

3. 采用 CRC 检验后，数据链路层的传输是否就变成了可靠的传输？
   - 采用 CRC 检验后，数据链路层的传输并非变成了可靠的传输。当接收方进行 CRC 检验时，如果发现有差错，就简单地丢弃这个帧。数据链路层并不能保证接收方接收到的和发送方发送的完全一样。

拥塞窗口 cwnd、ssthresh 计算

P232

前置知识

1. 慢开始
   - 当主机开始发送数据时，如果立即把大量数据字节注入到网络，那么就有可能引起网络拥塞，因为现在并不清楚网络的负荷情况。
   - 经验证明，较好的方法是先探测一下，即由小到大逐渐增大发送窗口，也就是说，由小到大逐渐增大拥塞窗口数值。

- 通常在刚刚开始发送报文段时，先把拥塞窗口 cwnd 设置为一个最大报文段 MSS的数值。而在每收到一个对新的报文段的确认后，把拥塞窗口增加至多一个 MSS 的数值。使用慢开始算法后，每经过一个传输轮次，拥塞窗口 cwnd 就加倍。
- 更精确的说法见课本 P233

2. Ssthresh
   - 为了防止拥塞窗口 cwnd 增长过大引起网络拥塞，还需要设置一个 慢开始门限 ssthresh 状态变量。
   - 当 cwnd > ssthresh 时，停止使用慢开始算法而改用拥塞避免算法。
3. 拥塞避免算法
   - 拥塞避免算法的思路是让拥塞窗口 cwnd 缓慢地增大，即每经过一个往返时间 RTT 就把发送方的拥塞窗口 cwnd 加 1，而不是加倍。这样，拥塞窗口 cwnd 按线性规律缓慢增长，比慢开始算法的拥塞窗口增长速率缓慢得多。
4. 快重传算法
   - 快重传算法首先要求接收方立即发送 ACK，即使收到了失序的报文段也要立即发出对已收到的报文段的重复确认。
   - 如果发送方连续收到三次重复的 ACK，就说明这个报文段大概率丢失了，立即重传它，并进入快恢复。
   - 三次重复 ACK 常发生在部分 segment 丢失的情况下，且网络状况总体良好。
5. 超时重传
   - 一般情况下，超时计时器需要的 ACK 是接收方等待自己发送数据时才进行捎带的，比快重传慢一些。
   - 超时常发生在全部 segment 丢失或丢包率较高时，网络状况较差。
   - 超时重传后，ssthresh = cwnd / 2，进入慢启动（cwnd 回到初始值）。
6. 快恢复算法
   - 执行“乘法减小”算法，把慢开始门限 ssthresh 减半，ssthresh = cwnd / 2
   - 把 cwnd 值设置为慢开始门限 ssthresh 减半后的数值，然后开始执行拥塞避免算法（“加法增大”），使拥塞窗口缓慢地线性增大。

[为什么收到三个重复的ACK意味着发生拥塞?]

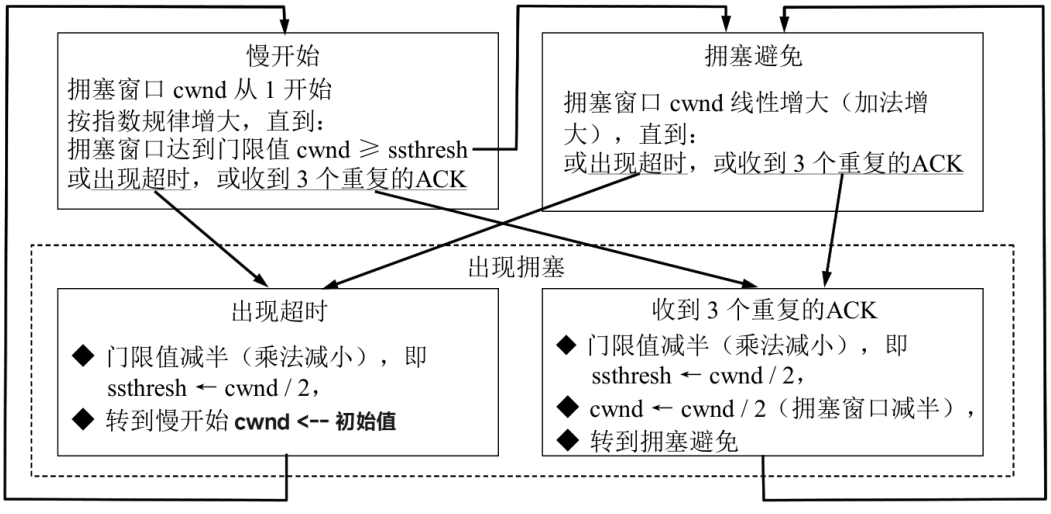


图 T-5-80　TCP 的拥塞控制流程图

例题【5-39】

表 T-5-39　拥塞窗口 cwnd 大小与传输轮次 n 的关系

n	1	2	3	4	5	6	7	8	9	10	11	12	13
cwnd	1	2	4	8	16	32	33	34	35	36	37	38	39
n	14	15	16	17	18	19	20	21	22	23	24	25	26
cwnd	40	41	42	21	22	23	24	25	26	1	2	4	8

(1) 试画出如教材的图 5-25 所示的拥塞窗口与传输轮次的关系曲线。
(2) 指明 TCP 工作在慢开始阶段的时间间隔。
(3) 指明 TCP 工作在拥塞避免阶段的时间间隔。
(4) 在第 16 轮次和第 22 轮次之后发送方是通过收到三个重复的确认，还是通过超时检测到丢失了报文段?
(5) 在第 1 轮次、第 18 轮次和第 24 轮次发送时，门限 ssthresh 分别被设置为多大?
(6) 在第几轮次发送出第 70 个报文段?
(7) 假定在第 26 轮次之后收到了三个重复的确认，因而检测出了报文段的丢失，那么拥塞窗口 cwnd 和门限 ssthresh 应设置为多大?
解答:

1. 图

   (1) 拥塞窗口与传输轮次的关系曲线如图 T-5-39 所示。

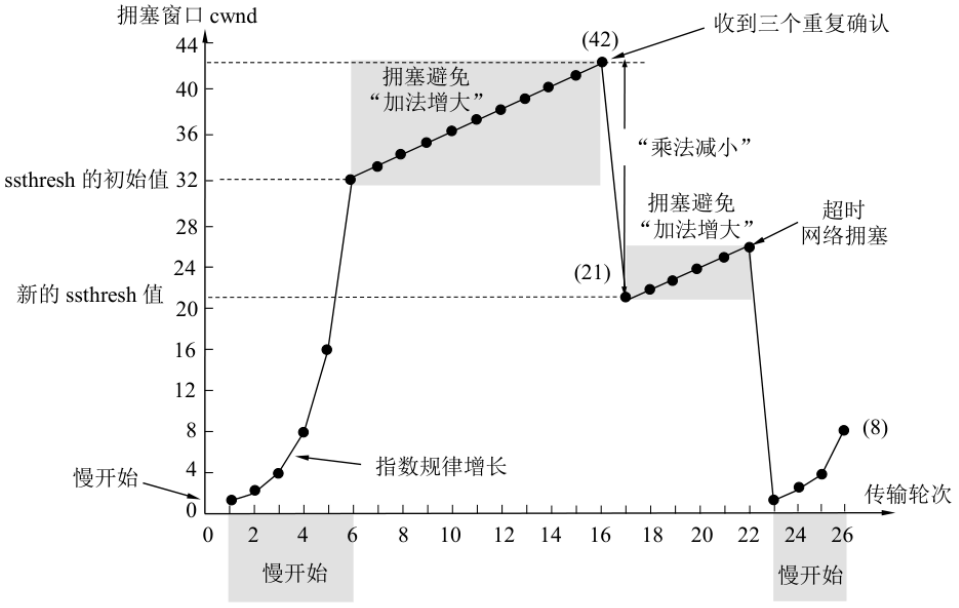


图 T-5-39　拥塞窗口与传输轮次的关系曲线

2. 慢开始时间间隔：[1，6]和[23，26]。
3. 拥塞避免时间间隔：[6，16]和[17，22]。
4.
   - 在第 16 轮次之后发送方通过收到三个重复的确认，检测到丢失了报文段，因为题目给出，下一个轮次的拥塞窗口减半了。
   - 在第 22 轮次之后发送方是通过超时检测到丢失了报文段，因为题目给出，下一个轮次的拥塞窗口下降到 1 了。
5.
   - 在第 1 轮次发送时，门限 ssthresh 被设置为 32，因为从第 6 轮次起就进入了拥塞避免状态，拥塞窗口每个轮次加 1。

- 在第 18 轮次发送时，门限 ssthresh 被设置为发生拥塞时拥塞窗口 42 的一半，即 21。
- 在第 24 轮次发送时，门限 ssthresh 被设置为发生拥塞时拥塞窗口 26 的一半，即 13。

6. 累加 cwnd 得到，第 7 轮次发送报文段是 64 ~ 94，包含第 70 个。
7. 检测出了报文段的丢失时拥塞窗口 cwnd 是 8，因此拥塞窗口 cwnd 的数值应当减半，等于 4，而门限 ssthresh 应设置为检测出报文段丢失时拥塞窗口 8 的一半，即 4。

发送窗口、可用窗口

> 没看懂这一部分要出什么类型的题...

例题【5-24】

题目

一个 TCP 连接下面使用 256 kbit/s 的链路，其端到端时延为 128 ms。经测试，发现吞吐量只有 120 kbit/s。试问发送窗口 W 是多少？（提示：可以有两种答案，取决于接收端发出确认的机。）

解答

设发送窗口 = W (bit)，再设发送端连续发送完窗口内的数据所需的时间 = T。有两种情况：

1．接收端在完整地收完一批数据后才发出确认，因此发送端经过 (256 ms + T) 后才能发送下一个窗口的数据。

2．接收端每收到一个很小的报文段后就发回确认，因此发送端经过比 256 ms 略多一些的时间即可再发送数据。可以看做每经过 256 ms 就能发送一个窗口的数据。

对于(a)：

$$\text{吞吐量} = \frac{W}{\frac{W}{256\ \text{kbit/s}} + 256\ \text{ms}} = 120\ \text{kbit/s}$$

发送窗口 $W / 120 = W / 256 + 256$

$256W = 120W + 256 \times 256 \times 120$

发送窗口 $W = 256 \times 256 \times 120 / 136 = 57825.88$ bit，约为 7228 字节。

对于(b)：

$$\text{吞吐量} = \frac{W}{256\ \text{ms}} = 120\ \text{kbit/s}$$

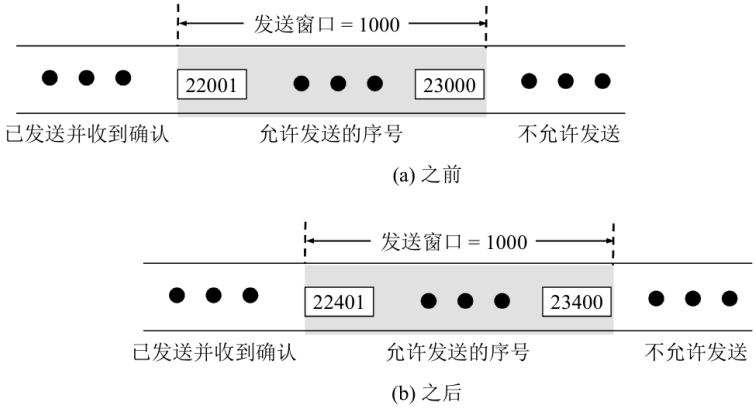
$W = 256 \times 120 = 30720\ \text{bit} = 3840\ \text{B}$

例题【5-49】

题目

TCP 连接使用 1000 字节的窗口值，而上一次的确认号是 22001。它收到了一个报文段，确认了字节 22401。试用图来说明在这之前与之后的窗口情况。

解答



第二次确认了 22401，可能是 22400 丢失了吧。

例题【5-61】

题目

在本题中列出的 8 种情况下，画出发送窗口的变化，并标明可用窗口的位置。已知主机 A 要向主机 B 发送 3 KB 的数据。在 TCP 连接建立后，A 的发送窗口大小是 2 KB。A 的初始序号是 0。

(1) 一开始 A 发送 1 KB 的数据。
(2) 接着 A 就一直发送数据，直到把发送窗口用完。
(3) 发送方 A 收到对第 1000 号字节的确认报文段。
(4) 发送方 A 再发送 850 B 的数据。
(5) 发送方 A 收到 ack = 900 的确认报文段。
(6) 发送方 A 收到对第 2047 号字节的确认报文段。
(7) 发送方 A 把剩下的数据全部都发送完。
(8) 发送方 A 收到 ack = 3072 的确认报文段。

解答

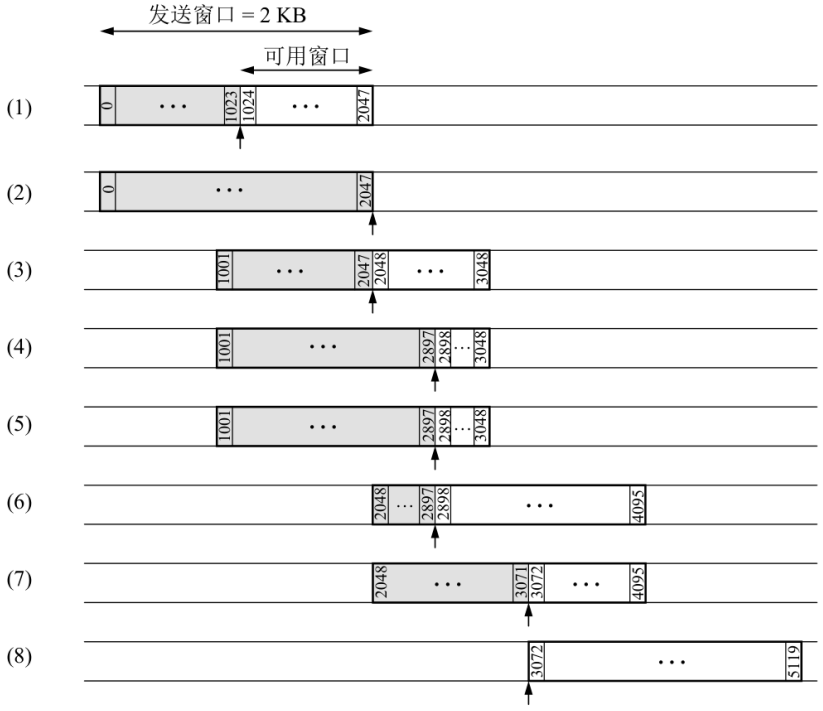


图 T-5-61　发送窗口和可用窗口的变化情况

(1) 我们应当注意到，发送窗口 = 2 KB 就是 2 × 1024 = 2048 字节。因此，发送窗口应当是从 0 到第 2047 字节为止，长度是 2048 字节。A 开始就发送了 1024 字节，因此发送窗口中左边的 1024 个字节已经用掉了（窗口的这部分为灰色），而可用窗口是白色的，从第 1024 字节到第 2047 字节为止。请注意，不是到第 2048 字节为止，因为第一个字节的编号是 0 而不是 1。
(2) 发送方 A 一直发送数据，直到把发送窗口用完。
(3) 发送方 A 收到对第 1000 号字节的确认报文段，表明 A 收到确认号 ack = 1001 的确认报文段。 这时，发送窗口的后沿向前移动，发送窗口从第 1001 字节（不是从第 1000 字节）到第 3048 字节（不是第 3047 字节）为止。可用窗口从第 2048 字节到第 3048 字节。
(4) 发送方 A 再发送 850 字节，使得可用窗口的后沿向前移动 850 字节，即移动到 2898字节。现在的可用窗口从第 2898 字节到第 3048 字节。
(5) 发送方 A 收到 ack = 900 的确认报文段，不会对其窗口状态有任何影响。这是个迟到的确认。
(6) 发送方 A 收到对第 2047 号字节的确认报文段。A 的发送窗口再向前移动。现在的发送窗口从第 2048 字节开始到第 4095 字节。可用窗口增大了，从第 2898 字节到第 4095 字节。
(7) 发送方 A 把剩下的数据全部都发送完。发送方 A 共有 3 KB（即 3072 字节）的数据，其编号从 0 到 3071。因此现在的可用窗口变小了，从第 3072 字节到第 4095 字节。
(8) 发送方 A 收到 ack = 3072 的确认报文段，表明序号在 3071 和这以前的报文段都收到了，后面期望收到的报文段的序号从 3072 开始。因此新的发送窗口的位置又向前移动，从第 3072 号到第 5119 号。整个发送窗口也就是可用窗口。

路由器路由表的转发

预备知识

1. 分组转发算法
    1. 从数据报的首部提取目的主机的 IP 地址 D，得出目的网络地址为 N。
    2. 若 N 就是与此路由器直接相连的某个网络地址，则进行直接交付，不需要再经过其他的路由器，直接把数据报交付目的主机（这里包括把目的主机地址 D 转换为具体的硬件地址，把数据报封装为 MAC 帧，再发送此帧）；否则就是间接交付，执行 (3)。
    3. 若路由表中有目的地址为 D 的特定主机路由，则把数据报传送给路由表中所指明的下一跳路由器；否则，执行 (4)。
    4. 若路由表中有到达网络 N 的路由，则把数据报传送给路由表中所指明的下一跳路由器；否则，执行 (5)。
    5. 若路由表中有一个默认路由，则把数据报传送给路由表中所指明的默认路由器；否则，执行 (6)。
    6. 报告转发分组出错。
2. 目的主机地址 是一个完整的 32 位（IPv4）或 128 位（IPv6）的地址，用于标识网络中的唯一主机。
3. 网络地址 是根据目的主机地址和子网掩码计算出的地址，用于标识一个网络。

4. 将目的主机地址和子网掩码都转换成二进制形式,对两者逐位进行逻辑与操作（AND）,结果即为网络地址。在进行 AND 运算时，只要把掩码地址中非全 1（即非 255）的那一个字节换算成二进制即可。全 1 字节与任何一个数 X 相与时，结果一定是 X，非常简单。
5. 题目中会给出路由表、子网掩码、目的主机地址，求出网络地址并在路由表中查找下一站即可。

例题【4-20】

【4-20】 设某路由器建立了如下路由表：

目的网络	子网掩码	下一跳
128.96.39.0	255.255.255.128	接口 m0
128.96.39.128	255.255.255.128	接口 m1
128.96.40.0	255.255.255.128	R_2
192.4.153.0	255.255.255.192	R_3
*（默认）	–	R_4

现共收到 5 个分组，其目的地址分别为：

(1) 128.96.39.10
(2) 128.96.40.12
(3) 128.96.40.151
(4) 192.4.153.17
(5) 192.4.153.90

试分别计算其下一跳。

IP 地址的 4 个字节分别表示为 B1，B2，B3 和 B4。把路由表中的 4 个目的网络地址分别记为 N1，N2，…，N4。收到的 5 个分组的目的地址分别记为 D1，D2，…，D5。
以 D1 为例计算：

	B1	B2	B3	B4
网络 N_1 的子网掩码 M_1（点分十进制）	255	255	255	128
网络 N_1 的子网掩码 M_1（第 4 字节用二进制表示）	255	255	255	10000000
收到的分组的目的地址 D_1（第 4 字节用二进制表示）	128	96	39	00001010
（M_1）AND（D_1）（第 4 字节用二进制表示）	128	96	39	00000000
（M_1）AND（D_1）（点分十进制）	128	96	39	0

所得结果与 N1 匹配。故选“接口 m0”。后面就不计算了。
如果不匹配，就继续计算 N2、N3、N4、N5。

二进制与十进制的转换

二进制 --> 十进制
从右到左用二进制的每个数去乘以2的相应次方（次方从0开始），再将其每个数进行相加。

$$1\times2^0+0\times2^1+1\times2^2+1\times2^3=1+4+8=13$$

十进制 --> 二进制
采用"除2取余，逆序排列"。用十进制整数除2，可以得到一个商和余数；再用商去除2，又会得到一个商和余数，如此进行，直到商为零时为止，然后把先得到的余数作为二进

制数的低位有效位，后得到的余数作为二进制数的高位有效位，依次逆序排列起来组合成二进制数。

例：把(17)10转换为二进制数。

即(17)10=(10001)2