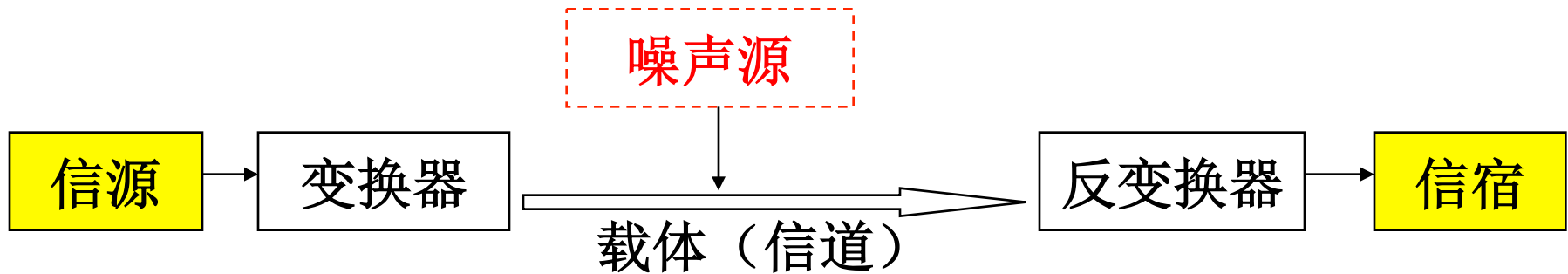


第2章 数据通信原理

1

通信系统的基本组成：



载体：信息的传送通道；

信源：信息的发出者（人，计算机设备等）；

信宿：信息的接收者（人，计算机设备等）；

变换器：将信息变换成载体上可传输的信号；

反变换器：将载体上传输的信号变换成信宿可识别的信息；

噪声：干扰信号。

2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

传输媒体类型划分—形式：

◆ 有线信道：双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；

◆ 无线信道：自由空间，红外、微波等，能量向空间发散。

双绞线/缆：以电磁波形式传输电气信号；

—无屏蔽双绞线（Unshielded Twisted Pair—UTP）

—屏蔽双绞线（Shielded Twisted Pair—STP）

2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道（二线），含传输媒体和中继通信设施；

传输媒体类型划分——形式：

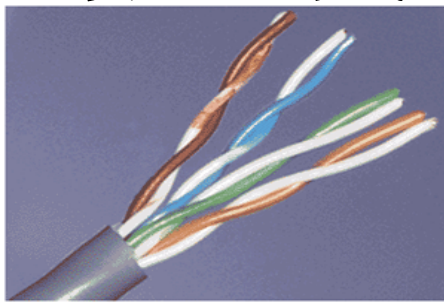
- ◆ 有线信道：双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；
- ◆ 无线信道：自由空间，红外、微波等，能量向空间发散。

双绞线/缆：以电磁波

——无屏蔽双绞线

——屏蔽双绞线

5类UTP双绞线



EIA/TIA相关UTP标准

类型	用途
Category 1	Voice Only (Telephone Wire)
Category 2	Data to 4 Mbps (LocalTalk)
Category 3	Data to 10 Mbps (Ethernet)
Category 4	Data to 20 Mbps (16 Mbps Token Ring)
Category 5	Data to 100 Mbps (Fast Ethernet)
Category 5e	Data to 1000 Mbps (Gigabit Ethernet)
Category 6	Data to 1000 Mbps (Gigabit Ethernet)
Category 7	Data to 10000Mbps (10Gbit Ethernet)

2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

传输媒体类型划分—形式：

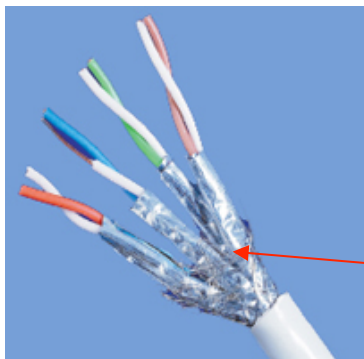
◆ 有线信道：双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；

◆ 无线信道：自由空间，红外、微波等，能量向空间发散。

双绞线/缆：以电磁波形式传输电气信号；

—无屏蔽双绞线（Unshielded Twisted Pair—UTP）

—屏蔽双绞线（Shielded Twisted Pair—STP）



STP—抗电气干扰能力强，传输距离100米；

金属屏蔽层

2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

传输媒体类型划分——形式：

◆ **有线信道：**双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；

◆ **无线信道：**自由空间，红外、微波等，能量向空间发散。

同轴电缆：以电磁波形式传输信号；

——粗缆：10Base-5， $\Phi 10\text{mm}$ ，AUI，单段长度 ≤ 500 米，最长5段达2.5公里。

——细缆：10Base-2， $\Phi 5\text{mm}$ ，BNC，单段长度 ≤ 185 米，最长5段达925米。

2.2 载体（信道）

载体：承载信息/信号的媒体；

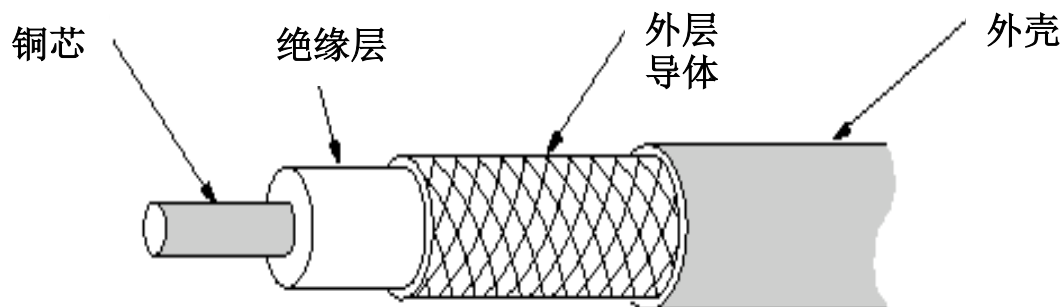
信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

传输媒体类型划分——形式：

◆ 有线信道：双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；

◆ 无线信道：自由空间，红外、微波等，能量向空间发散。

同轴电缆：以电磁波形式传输信号；



2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

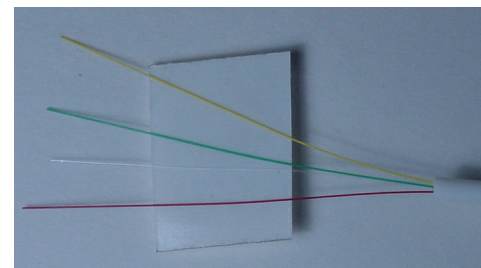
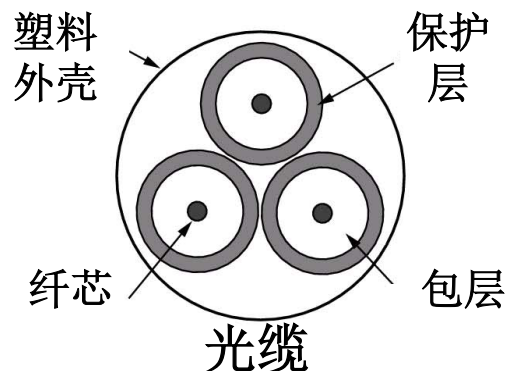
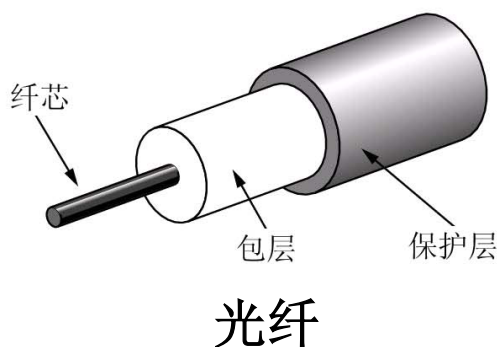
传输媒体类型划分—形式：

◆ 有线信道：双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；

◆ 无线信道：自由空间，红外、微波等，能量向空间发散。

光纤/缆：以光波形式传输信号；

由传导光波的高纯石英玻璃纤维和保护层构成，纤芯的折射率大于包层折射率，保证光信号在纤芯内折射传输。多根光纤封装于外壳中，形成多芯光缆。



实际光纤/缆

2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

传输媒体类型划分——形式：

◆ 有线信道：双绞线/缆、同轴电缆、光纤/缆等，
能量集中导线附近；

◆ 无线信道：自由空间，红外、微波等，能量向空间发散。

无线信道：以无线电频率（射频—RF）形式传输信号；

f= 1K	10K	1M	1G	100G	100T	10 ¹⁵	10 ¹⁶ (hz)
语音		无线电/广播	微波	红外	可见光	紫外线	

低频LF、中频MF波段电波沿地表传播；

高频HF和甚高频VHF波段电波通过电离层反射实现长距离传输；

红外线与毫米波传输：不能穿透障碍物，适用于室内

微波传输：能量集中，天线必须对准。

激光传输：不能穿透雨或浓雾。

2.2 载体（信道）

载体：承载信息/信号的媒体；

信道：信息单向传输通道(二线)，含传输媒体和中继通信设施；

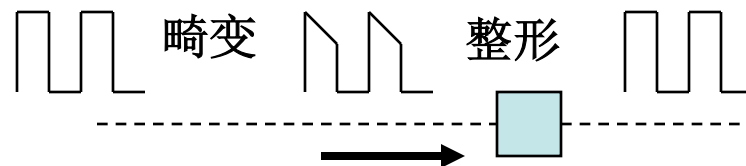
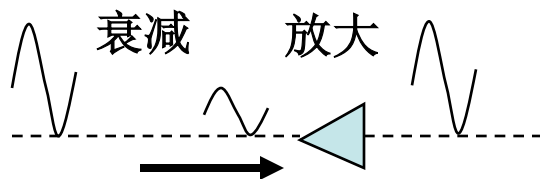
传输媒体类型划分—**方式：**

◆ 模拟信道：支持模拟信号传输，如双绞线、同轴电缆等；

◆ 数字信道：支持数字信号传输，如光纤、双绞线等；

注1：计算机仅能产生数字信号，处理信息；

2：阻抗导致信号衰减，线间电容导致信号畸变，长距离传输时必须放大（补充能量）和整形。



信道带宽与信道容量 (信道的物理特性)

信道带宽: 信道可以不失真地传输信号的频率范围;

信道带宽取决于信道的质量, 设计信道的指标。

信道容量: 信道在单位时间内可以传输的最大信号量

数据传输速率(bps):

信道在单位时间内可以传输的最大比特数;

信道容量和信道带宽成正比: **带宽越大, 容量越大**

数据传输速率和信道容量成正比, 比率取决于调制技术

局域网: 10Mbps, 100Mbps, 千兆bps, 10千兆

广域网: 64Kbps、2Mbps、155Mbps、622Mbps、2.5Gbps, ...

差错率/误码率

由于噪声的影响和信道带宽的限制，信号在传输过程中可能发生失真。

差错率/误码率：传输比特总数与其中出错比特数的比值；

$P_e = \text{出错比特数} / \text{传输比特总数}$

例： 传输10000比特，有2比特出错，

$$P_e = 2/10000$$

差错率越高表示信道的质量越差

信道的差错率与信号的传输速率和传输距离成正比

类似的有：误分组率等；

2.3 变换器/反变换器—调制/解调与编码/解码¹¹

模拟传输系统：模拟信道构成的传输系统，如电话网、

X.25分组交换网等；

数字传输系统：数字信道构成的传输系统，如宽带ISDN等

★ 调制/解调：利用模拟信道支持数据信息传输的技术

调制：将数据信息变换成适合于模拟信道上

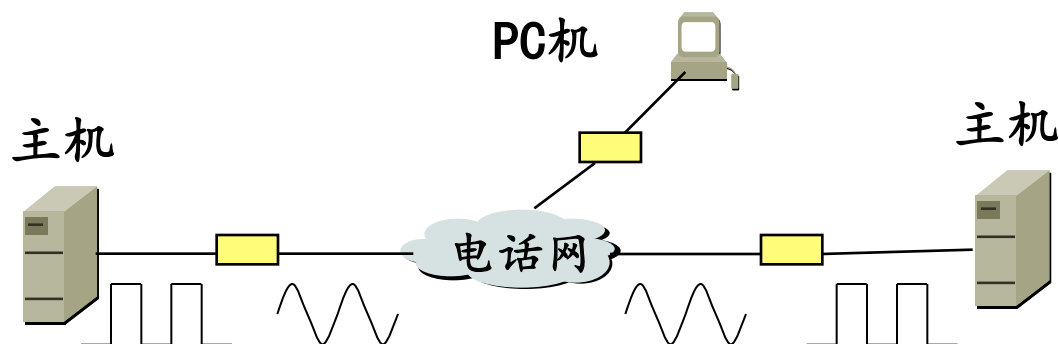
传输的电磁波（称为载波）信号，（数字→模拟）

解调：将从模拟信道上收取的载波信号还原成数据信息。

（模拟→数字）

调制解调器：

具有调制/解调
功能的通信设备。

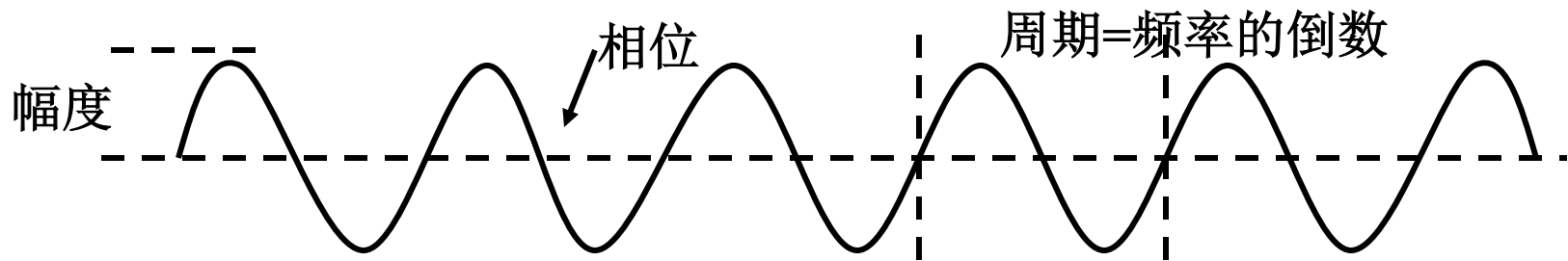


★调制方法—调制依据

- 1、任何周期为T的函数g(t)都可以展开为Fourier级数（n次谐波叠加），即模拟信号可由三角函数表示

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

- 2、模拟信号可由三个要素（幅度、频率和相位）予以定义



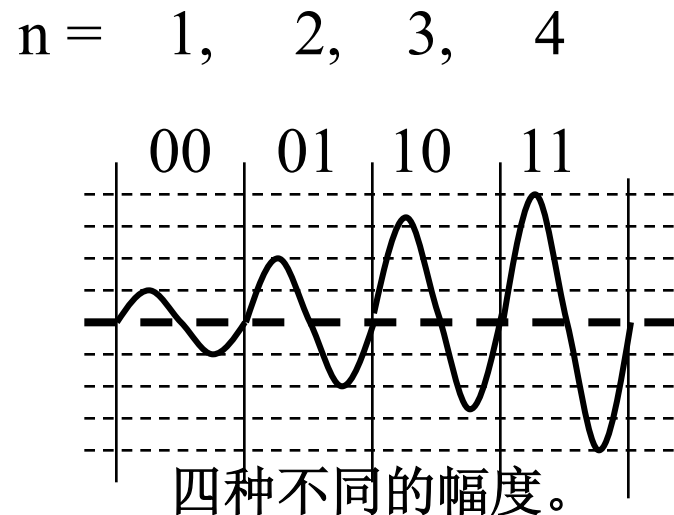
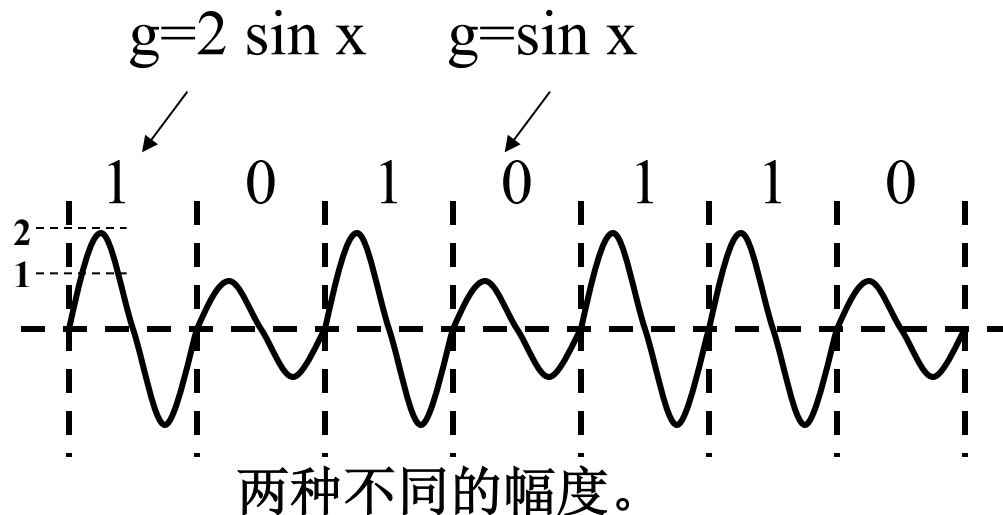
调制：三个要素的调制。

★调制方法：

调幅：（幅度调制或移幅键控法ASK）：

将不同的数据信息（0和1）调制成不同幅度
但相同频率的载波信号；

$g(x) = n * \sin(x)$ ，不同n产生不同幅度的载波信号。

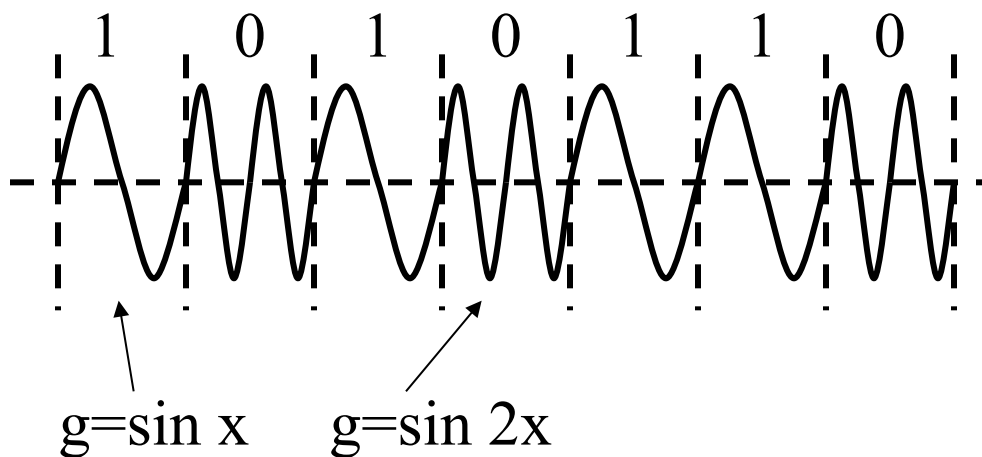


★调制方法：

调频：（频率调制或移频键控法FSK）：

将不同的数据信息（0和1）调制为相同幅度，但不同频率的载波信号；

$g(x) = A \sin(n * x)$ 不同 n 产生不同频率的载波信号



本例设 $A=1$ ，且仅采用两种不同的频率。

★调制方法：

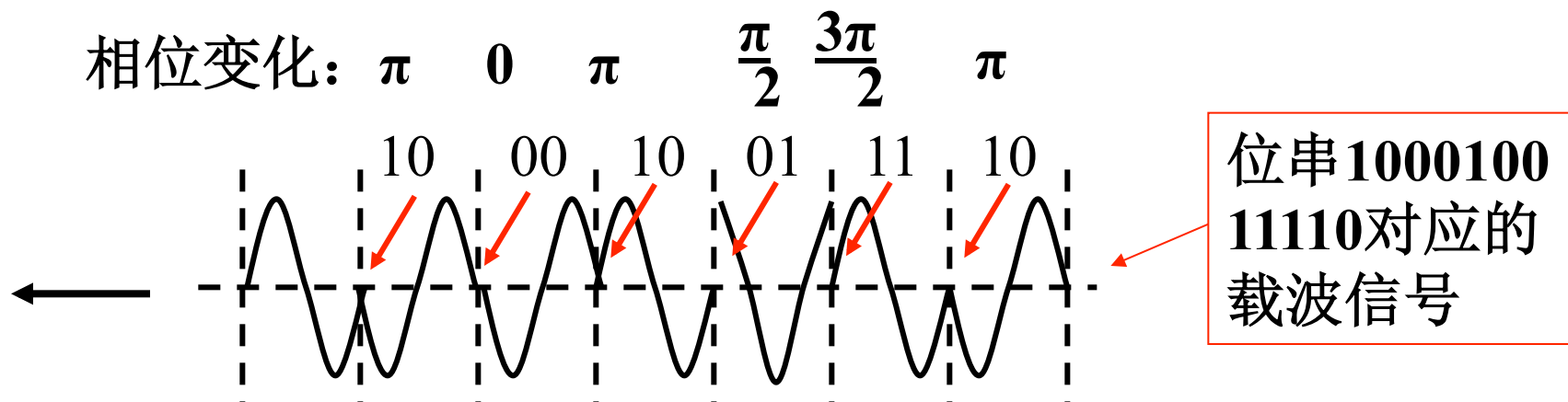
调相：（相位调制或移相键控法PSK）：

利用相邻载波信号的相位变化值来表示相邻信号是否具有相同的数据信息值，此时的幅度和频率均不发生变化；

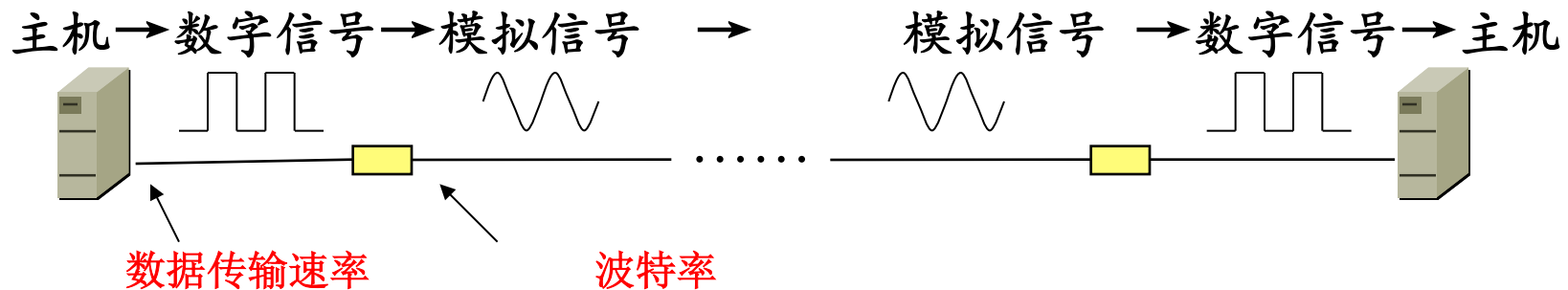
例如：0—相位变化180度，1—相位不变化；

或者：00—不变化，01—90度，10—180度，11—270度；

$g(x) = \sin(n+x)$ 不同 n 产生相位变化不同的载波信号



调制/解调的结果



调制设备负责将主机端的数字信号调制成可在模拟信道上传输的模拟信号（符号/码元）。

调制设备的性能影响信号调制的速率，影响数据传输速率。

调制速率（信道速率，或者**波特率**）：调制设备每秒可调制的符号/码元个数，即信道上每秒传输的符号个数。

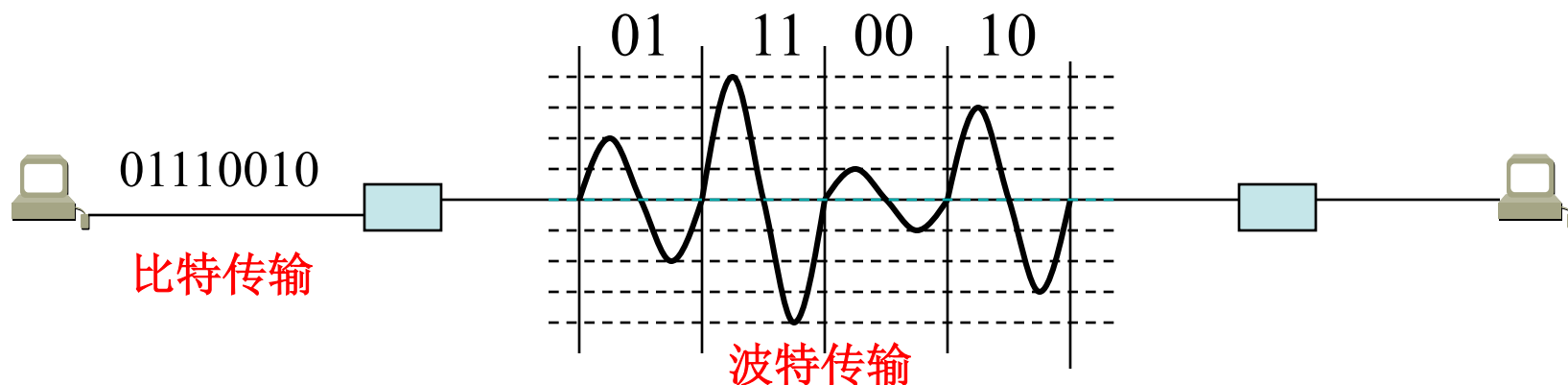
数据传输速率：信道在单位时间内可以传输的最大比特数，取决于调制速率和码元蕴含的**信息量**。

★ 波特率和数据传输速率之间的关系

如果调制设备的调制速率（波特率）为 M ，调制出的每个码元具有 N 种取值（线路的状态数，信息量），则有：

$$\text{数据传输速率} = M \times \log_2 N = \text{波特率} \times \log_2 N$$

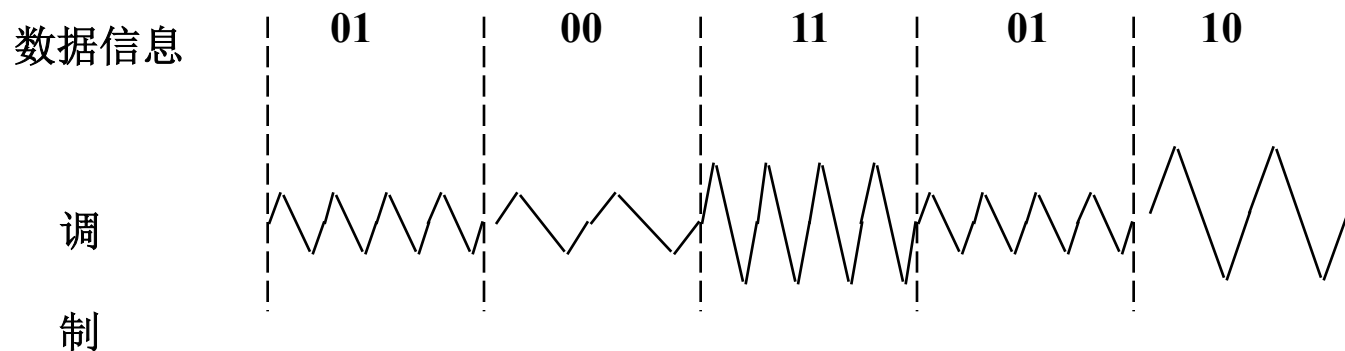
假设调制设备采用调幅调制技术，每秒调制出一个码元，该码元可取四种幅度之一：



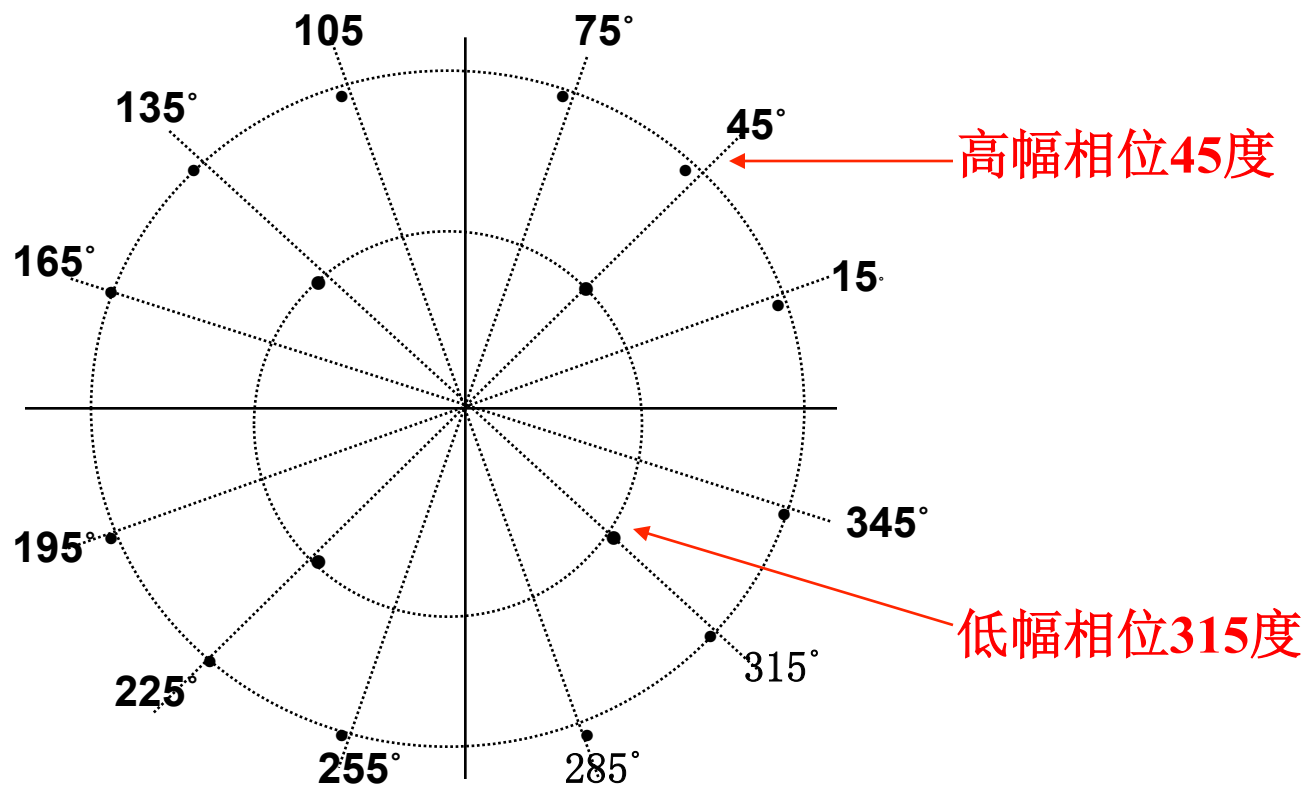
波特率：1码元/秒；

数据传输速率：2位/秒

幅度+频率
(2倍速)



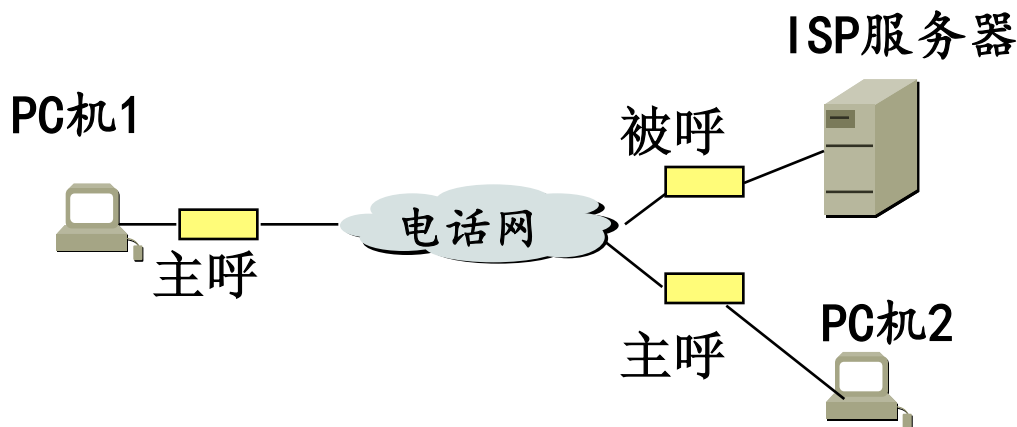
相位+幅度
(4倍速)



★调制/解调器的选择和应用

- 1) 性能：速率、功能等
- 2) 用途：使用的场合
- 3) 符合相关标准
- 4) 符合当地有关部门的入网规定

注意：Modem需成对使用，所有通信双方的Modem一定要匹配
一台主呼，另一台被呼 用户方一般为主呼

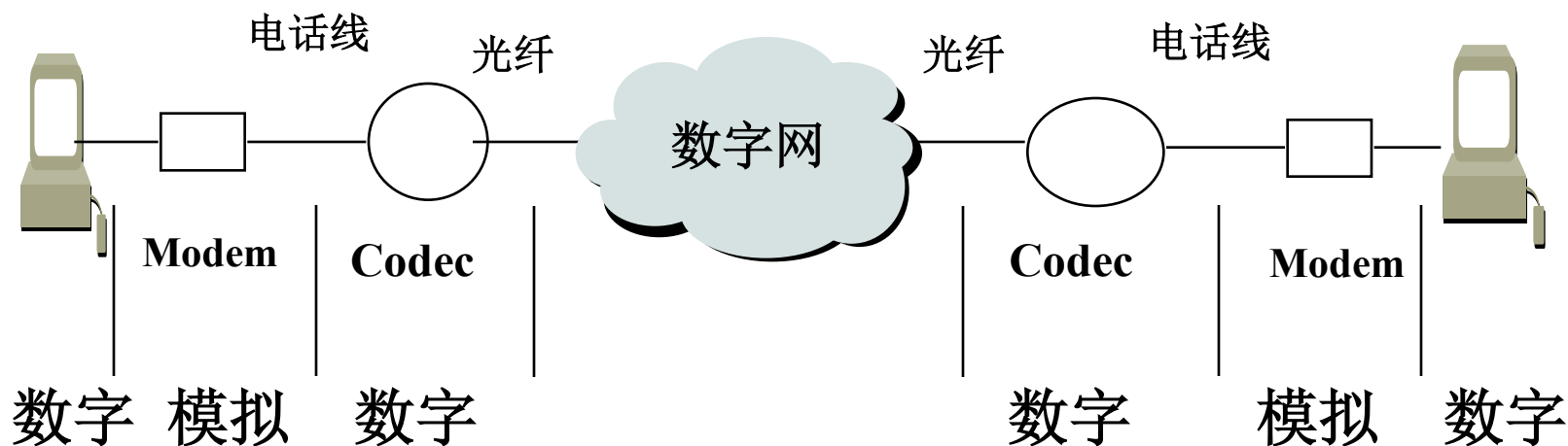


实现模拟信息与数字信号之间的转换

用于生产控制、数字信道传输模拟信息等

编码: 将模拟信息转换为数字信号的过程

解码: 将数字信号还原为模拟信息的逆过程



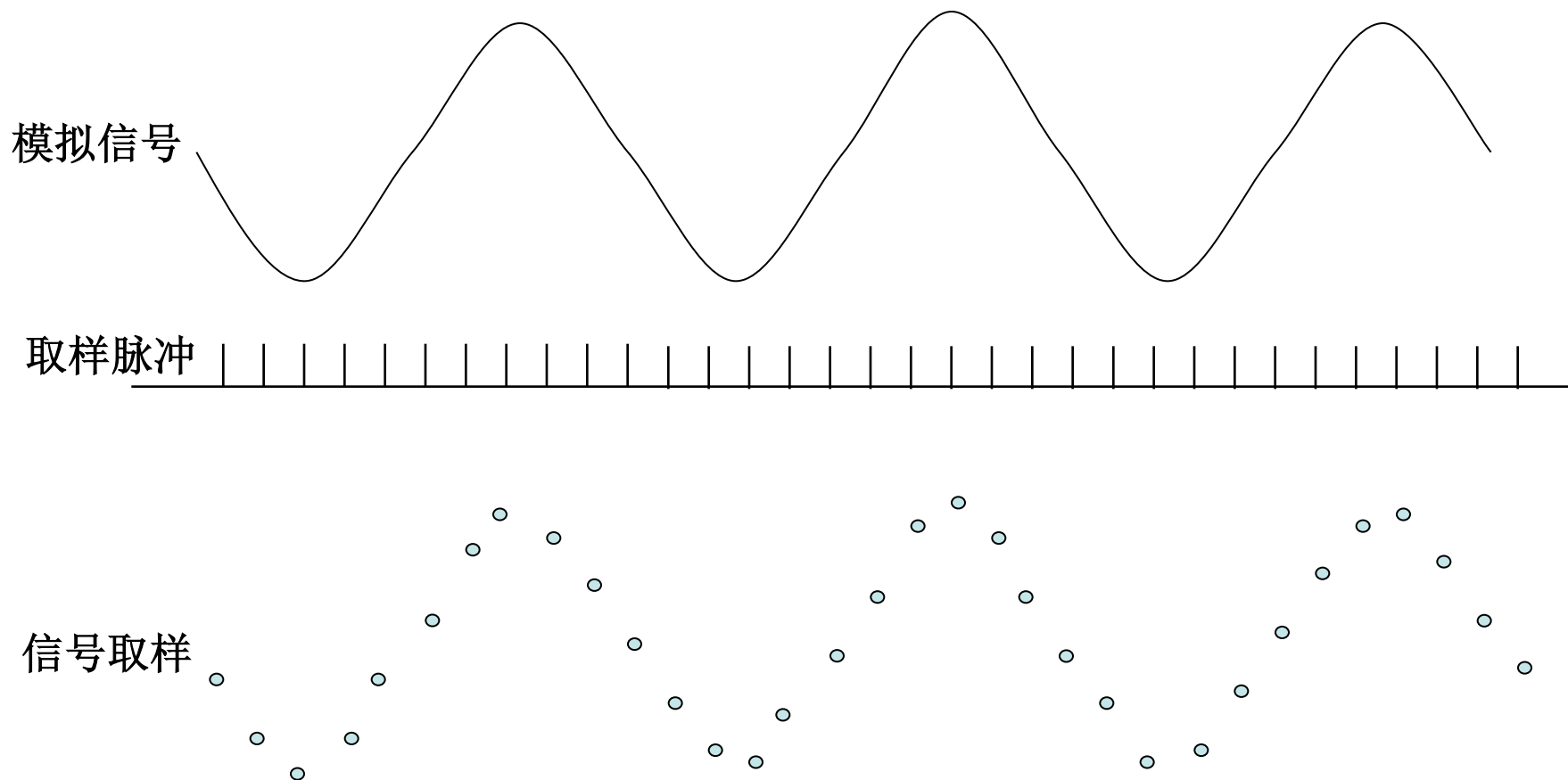
脉码调制技术 (PCM)

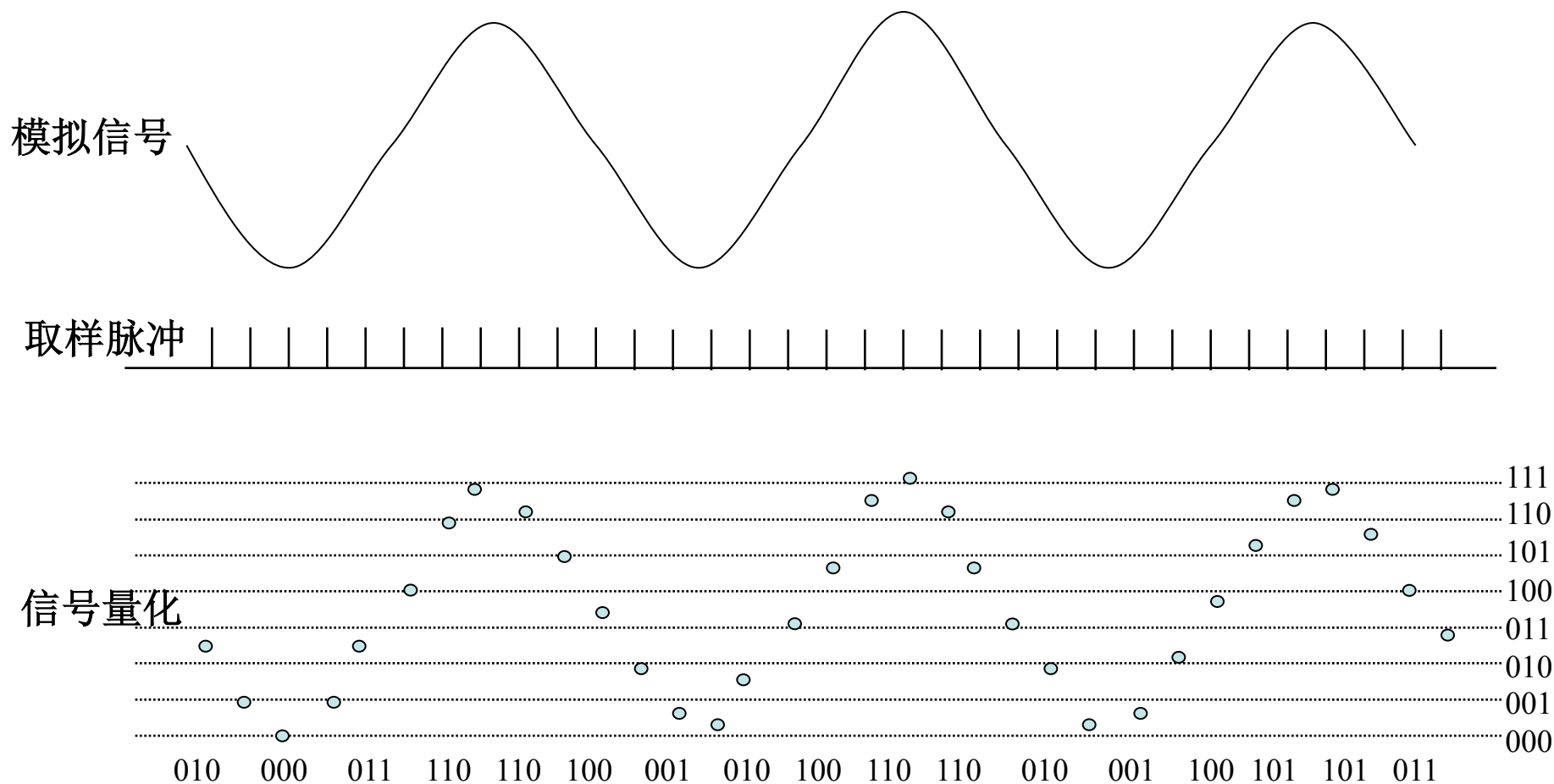
取样：通过某种频率的取样脉冲将模拟信息的值取出，
变连续的模拟信息为离散信号。

量化：量化的目的是确定取样出的模拟信号的数值。
通过规定一定的量化级，对取样的离散值进行
“取整”量化，得到离散信号的具体数值。

编码：将量化后的值编码成一定位数的二进制值。

依据—**奈奎斯特取样定理**：最大频率为 F 的模拟信号被不失真
还原的前提条件是取样频率不低于 $2F$ 。



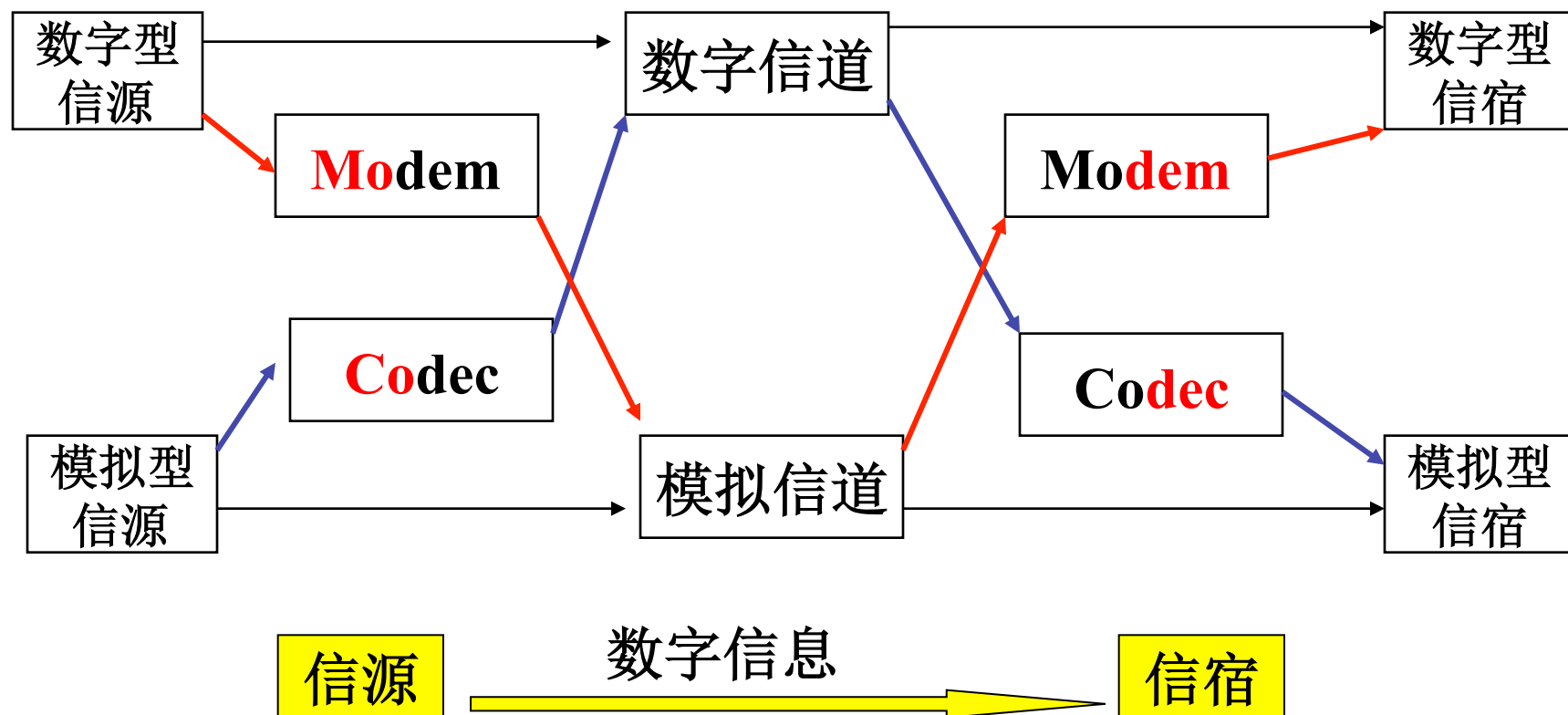


8级量化，可用3位表示。

语音频率小于4000Hz，采样频率8000Hz，量化256级，编码8位，传输速率：不小于64Kbps = 8000次/秒 * 8位/次。

变换器和反变换器的效果

通过调制/解调、编码/解码技术，可以保证计算机之间以数字信号的方式进行通信；



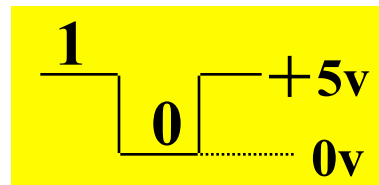
数字信号的发送和接收

数字信号的表示：

二进制数字 (0, 1) 对应两个电平 (或光脉冲)

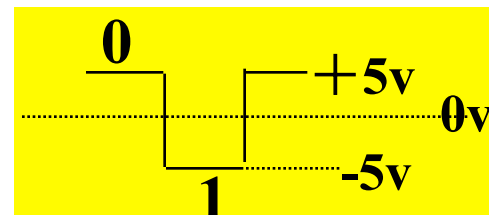
单极性脉冲： 无电压 (或者无电流) “0”

恒定正电压 (或者有电流) “1”



双极性脉冲： 数字信号 “0” 或者 “1”

相同幅度的正电压或者负电压

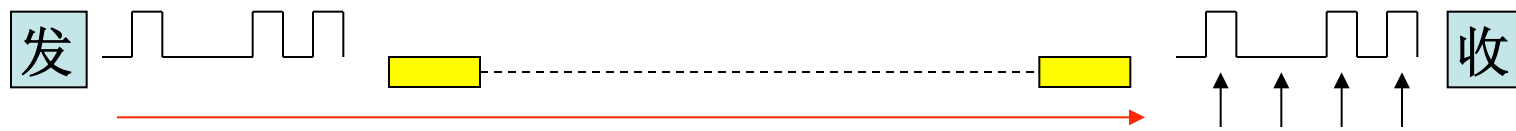


发送： 发送设备根据自身的时钟分频形成指定频率

(发送频率) 的数据波 (脉冲序列)，并发往线路；

接收： 接收端设备则根据自身的时钟形成指定频率

(接收频率) 的取样脉冲，对信道上的数据波进行取样
并通过设置**阈值**电平识别数据波对应的值。



★ **字符编码**： 利用0和1比特的特定组合来表示字符
BCD码、EBCDIC码、IA5码、ASCII码 (美国信息交换标准码, P23)

图形字符：数字、字母、运算符号、语句符号等

控制字符：传输控制、格式控制、信息分隔字符等

ASCII码的表示： $b_7b_6b_5b_4b_3b_2b_1$ (表2-1)

(简记X/Y 或XY: $X=b_7b_6b_5$ $Y=b_4b_3b_2b_1$)

例：A: 1000001, 记为4/1 或41

(用ASCII码 “41”表示字符 ‘A’)

传输控制字符：用于控制信息的传输

SOH(标题开始, 0000001, 0/1),	
STX(正文开始, 0000010, 0/2)	ETX(正文结束, 0/3)
EOT(传输结束, 0000100, 0/4),	ENQ(询问, 0/5),
ACK(确认, 0000110, 0/6),	NAK(否认, 1/5),
DLE(数据转义, 0010000, 1/0),	SYN(同步, 1/6);

格式控制字符：控制打印和显示设备的信息格式和定位

BS(退格, 0001000), LF(换行, 0001010, 0/A)
CR(回车, 0001101, 0/D)

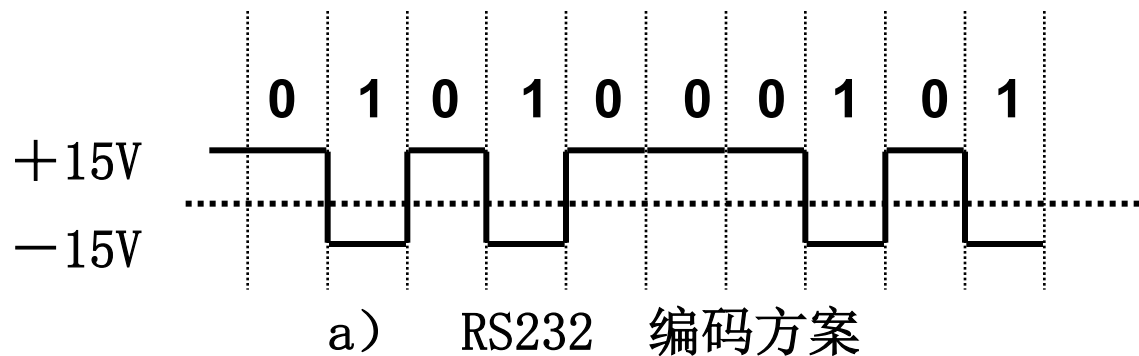
信息分隔字符：用于分隔信息

US(单元分隔, 1/F), RS(记录分隔, 1/E), GS(组分隔, 1/D),
FS(文卷分隔, 1/C)。

通信编码：利用特定的电平信号来表示0、1比特值，
并通过计算机或者其它通信设备的输入输出端口传输

- (1) **RS-232编码**：利用不同的电平表示不同的二进制值，
正电平（+15v）表示数字信号 “0”
负电平（-15v）表示数字信号 “1”

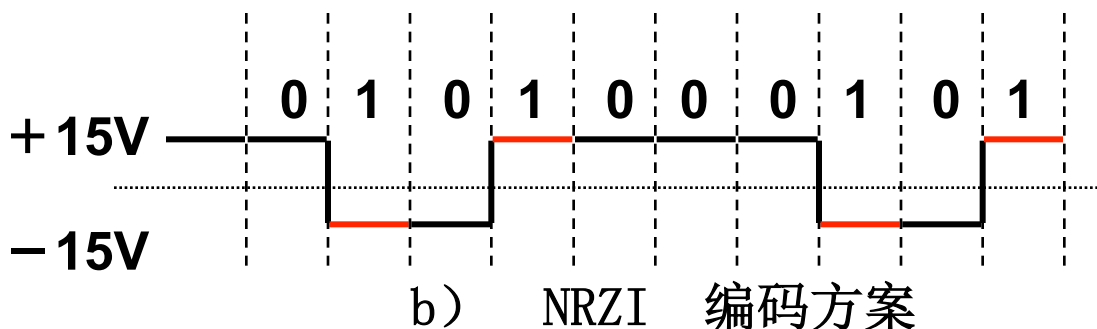
PC机的RS232串行通信端口



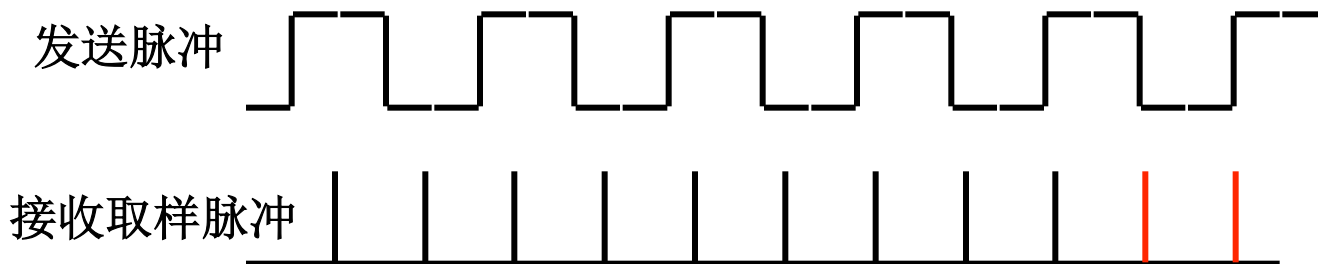
(2) **不归0交替编码 (NRZI)**: 根据相邻比特的电平变化确定值

比特间隔发生电平变化表示 “1”

比特间隔不发生电平变化表示 “0”



特点: 编码中不含同步信息 发送/接收设备的时钟略有差异时, 可能造成误差积累, 造成取样脉冲的偏移, 出现差错

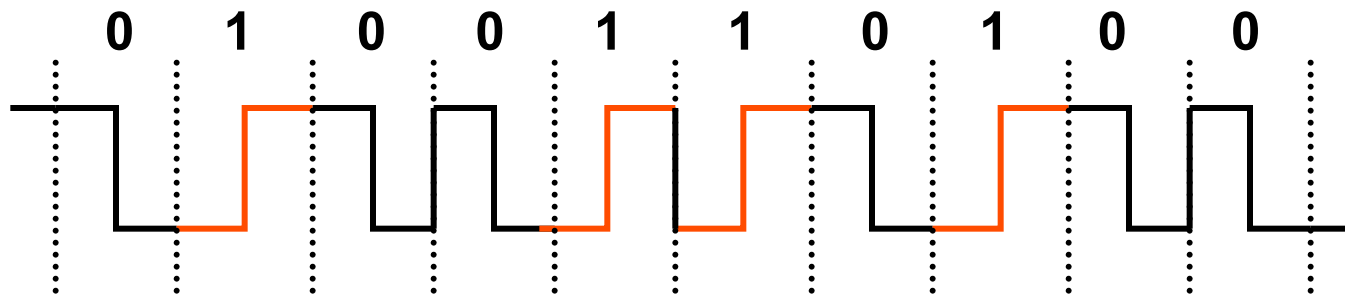


收发设备的误差积累

(3) 曼彻斯特编码

一个比特时间一分为二，在每个比特间隔的中间引入跳变来同时代表不同比特和同步信息。

比特时间内 发生低电平到高电平的变化表示“1”，
高电平到低电平的变化表示“0”



(4) 差分曼彻斯特编码

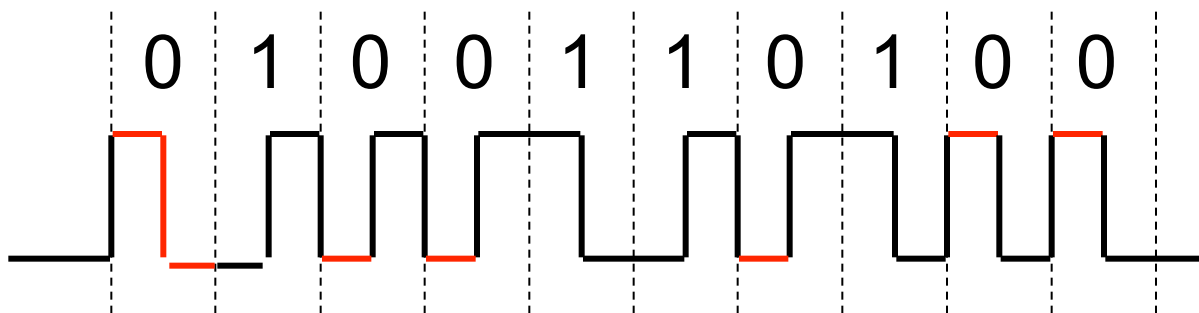
编码特征：一个比特时间一分为二，

比特时间的中部发生电平变化，

表示的值依赖于前一比特的最终电平状态

当前比特的前半部分电平不同于前一比特的最终电平状态
(即位间电平发生变化)，表示“0”

当前比特的前半部分电平相同于前一比特的最终电平状态
(即位间电平不发生变化)，表示“1”



特点:

- (3) (4) 编码中含有同步信息(每个比特中部的电平跃变信号)。
- 接收方可以根据该同步信息及时调整接收脉冲的产生, 消除误差积累, 可以支持较大数据块的传输。
- 要求发送/接收设备能够产生较高频率的发/收脉冲。
- 编码效率较低(50%)。

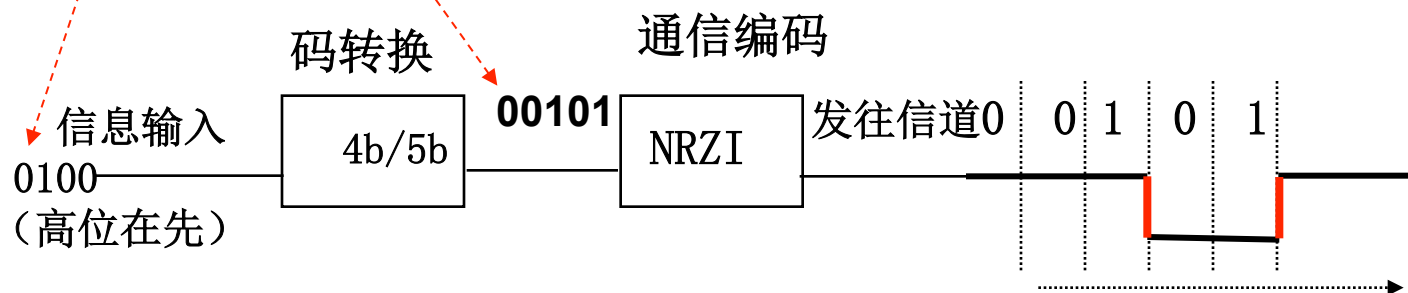
(5) 4b/5b码——光纤应用，降低成本

用5位(32个)的符号表示4位(16个)的信息(数据)

采用不归0交替编码(NRZI)表示这5位符号。

(32选16) 使得每个符号中至少有2个“1”比特
(跃变) 出现,

例: 0010→10100, 0110→01110, 1100→11010
1000→10010, 0000→11110



特点: 内含同步信号, 支持批量数据传输;
编码效率较高, 80%。

类似有5b/6b, 8b/10b, 64b/66b,。

4B/5B编码规则

- 每个5比特码组中不含多于3个“0”；
- 或者5比特码组中包含不少于2个“1”；
- 逢“1”跳变，逢“0”不跳变。为了让4B/5B编码后的码流中有足够多的跳变就需要编码后的码流中有尽量多的“1”和尽量少的“0”。

十进制数	4位二进制数	4B/5B码
0	0000	11110
1	0001	01001
2	0010	10100
3	0011	10101
4	0100	01010
5	0101	01011
6	0110	01110
7	0111	01111
8	1000	10010
9	1001	10011
10	1010	10110
11	1011	10111
12	1100	11010
13	1101	11011
14	1110	11100
15	1111	11101

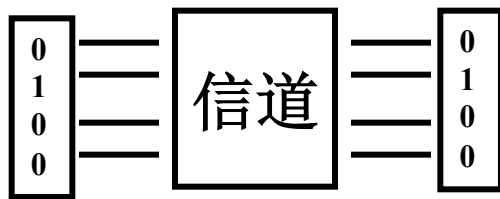
2.5 传输方式

➤ 并行传输: 字符编码的各个比特同时传输

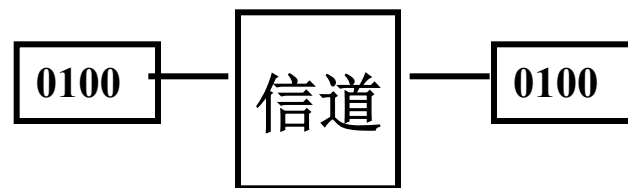
特点: 一个比特时间内可传输一个字符, 传输速度快,
每个比特传输要求一个单独的信道支持, 通信成本高,
远距离传输时, 线间干扰导致可靠性下降。

➤ 串行传输 将组成字符的各个比特串行地发往线路

特点: 传输速度低, 一次一个比特;
通信成本较低, 只需一个信道



并行传输



串行传输

★传输方式

☆ 同步传输：以多个字符或者多个比特组合成的数据块为单位进行传输，

利用独特的同步模式来限定数据块，达到同步接收的目的。

发送：同步符号(起始字符) + 数据块 + 同步符号(结束字符)

接收：遇到同步符号，开始接收数据，直到结束符号为止。

同步符号：标识数据块的开始和结束

SYN, G, H, ..., B, A, D, E, SYN → 传输方向

可能问题：假同步现象—数据块中含有与同步符号相同的内容

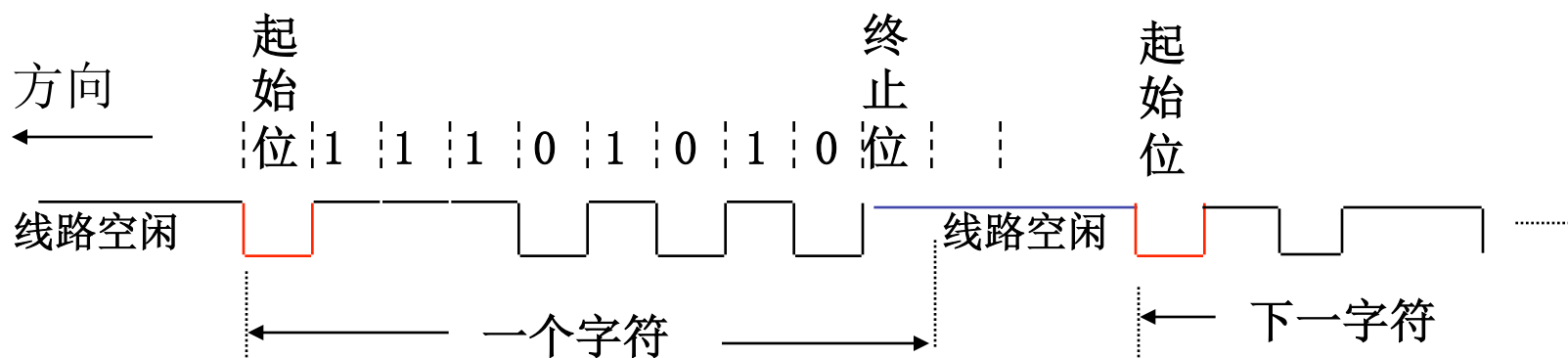
解决方法：增加匹配同步符号的难度

SYN, SYN, G, H, ..., B, A, SYN, D, E, SYN, SYN → 传输方向

★传输方式

☆异步传输

特点： 字符内部的各个比特采用固定的时间模式，
每个字符独立传输， 字符之间间隔任意；
用独特的起始位和终止位来限定每个字符。
传输效率较低。



目的： 保证接收方在时间上与发送方取得同步，
以便能够正确地识别和接收发送方发来的数据。

位同步： 使接收方可以正确地接收各个比特

自同步法：接收方直接从数据波中获取同步信号（曼码）。

外同步法：发送方在发数据前，先向接收方发一串同步时钟序列，接收方根据这一同步时序锁定接收频率。

（异步传输的起始/终止位）

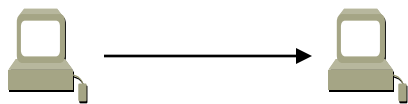
字符同步： 使接收方可以正确地识别数据群
利用同步传输时的同步字符（SYN）

接收方在识别到独特的同步字符或同步模式后，
才开始真正的数据接收。

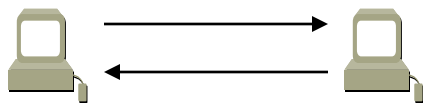
接入X.25分组交换网：同步端口

接入电话网：异步端口

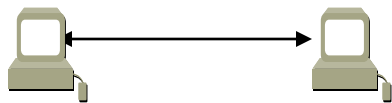
单工传输：任意时刻只允许向一个方向进行信息传输；



半双工传输：可以交替改变方向的信息传输，
但在任一特定时刻，信息只能向一个方向传输；



全双工传输：任意时刻信息都可进行双向的信息传输。



目的：保证信息传输的正确性；

噪声导致差错，无差错处理能力的系统是不可用系统。

方法1：自动重传请求 (ARQ)

- (1) 发送方发送具有检测错误能力的代码（检错码）；
- (2) 接收方根据代码的编码规则，验证接收到的数据代码，
并将结果反馈给发送方（正确接收/接收有错）；
- (3) 发送方根据反馈的结果决定是否执行重传动作，
如果接收方未正确接收，则重传；
- (4) 在规定的时间内，若未能收到反馈结果（称为超时），
则发送方可以认为传输出现差错，进而执行重传动作。

发送一块数据，计时。

等待接收方的反馈结果，

如果接到否定确认，

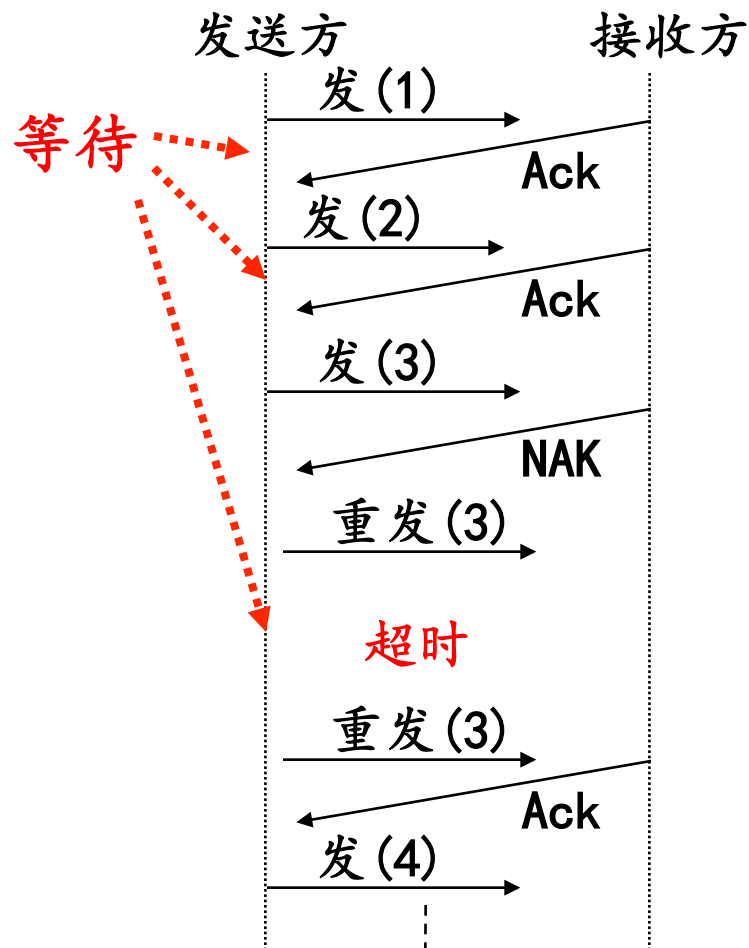
重新传输本数据块；

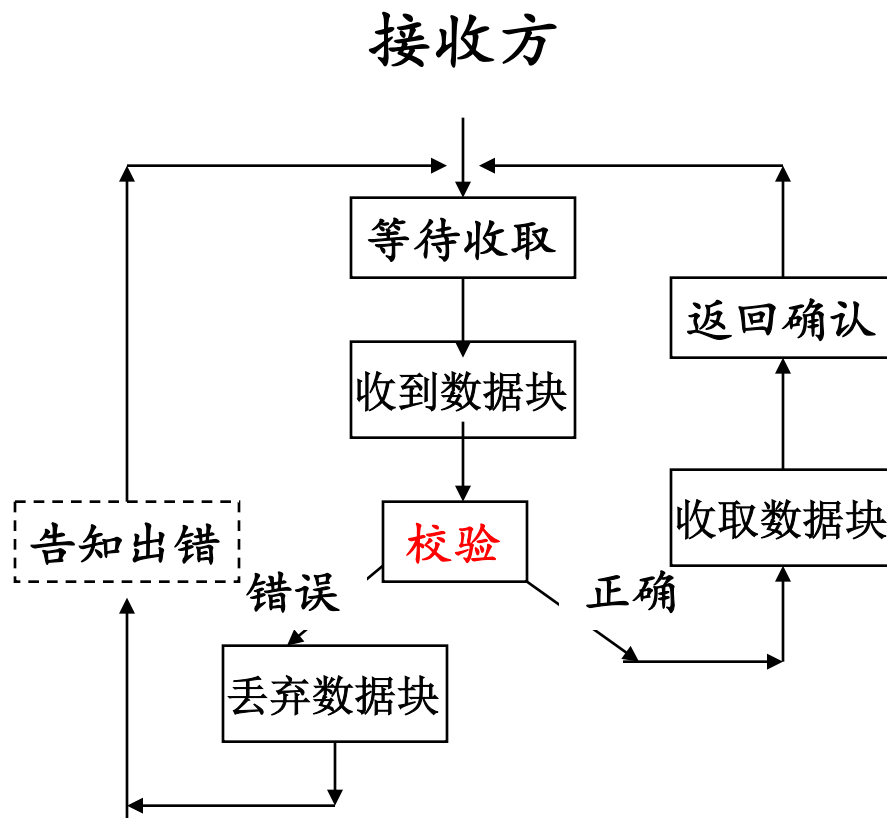
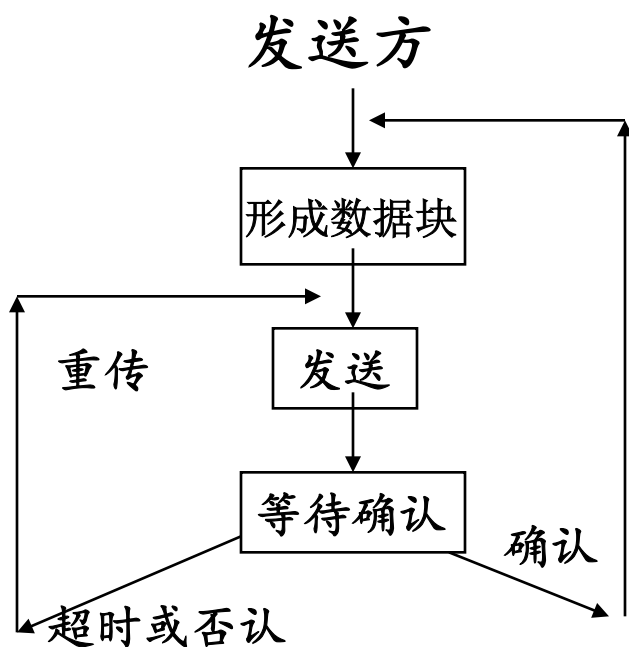
如果收到接收确认，

继续发送后继块数据；

如果计时器超时，

重新传输本数据块。





校验含差错分析和序号检测

收发双方以半双工方式进行工作，

特点：控制简单，易于实现；等待验证，线路空闲，效率较低。

➤ 滑动窗口协议——停等协议的改进

允许发方连续发送多块数据；

允许收方对多块数据统一确认。

(1) 发方一次连续发送多块数据

(块数限于窗口尺寸)；

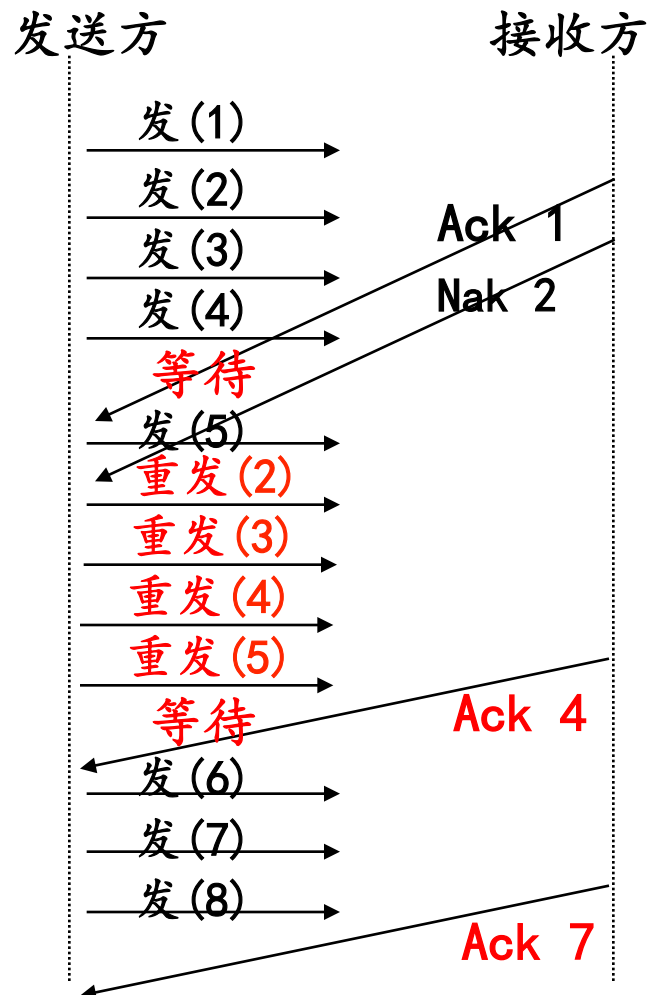
(2) 收方对每块数据进行差错分析，

如果发现错误，立即反馈发送方；

(3) 收方可对收到的多个正确的数据块进行一次性确认；

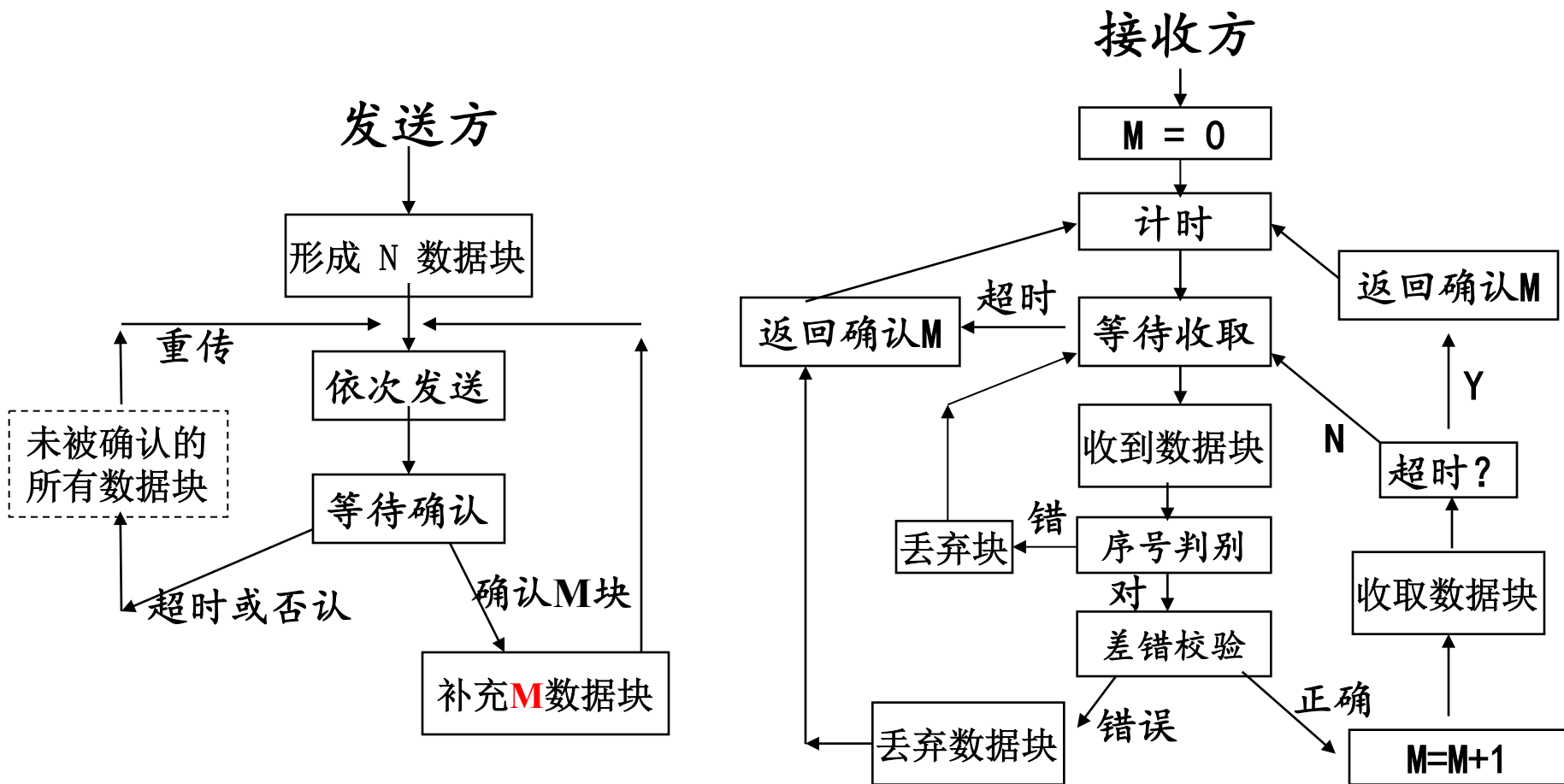
(4) 发方根据反馈的结果，重发指定数据块(SB)，或重发指定数据块及其后所有数据块(GBN)，或者直接发送后续数据块。

假设窗口=4



SB: 选择后退重传1帧; **GBN**: 后退重传N块; (例为GBN)

允许连续发送多块数据；允许对多块数据统一确认。



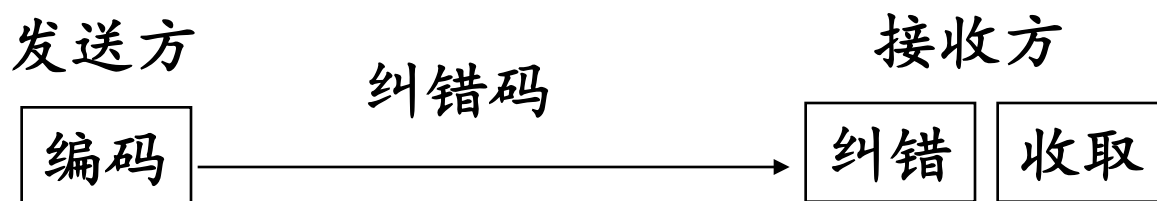
特点：提高信道利用率，全双工方式，控制相对复杂；

方法2：前向纠错法（FEC）

6

发送方发送具有纠错能力的编码；

接收方根据编码规则纠正传输中的差错。



特点：无需反馈信道；

编码复杂（纠错能力有限）。

差错处理的核心：**检错码/纠错码**的构造。

检错码是差错检测的**核心**。

☆ 检错码=信息字段+校验字段（冗余字段）

校验字段和信息字段之间存在相关性、**联动**性；

校验字段越长，编码检错能力越强，编/解码设施越复杂；

附加的冗余信息在整个编码中所占的比例越大，传输的有效成分越低。

好的检错码：检错能力**强**，编码效率**高**。

传输**顺序**：信息字段在前，校验字段在后。

差错**原因**：噪声；

差错**现象**：突发性和连续性（影响若干位。）

奇/偶校验码的校验字段仅占一个比特（一位，校验位）。

(1) 水平奇/偶校验码（可发现奇位错）

校验位的取值应使整个码字(包括校验位)中为“1”的比特个数为奇(偶)数。

传输时，形成的校验位附加在字符之后传输。

例：←	信息字段	奇校验码	偶校验码
	0110001	01100010	01100011

编码效率： $Q/(Q+1)$ （信息字段占Q个比特）

异步传输方式中采用水平偶校验，

同步传输方式中采用水平奇校验。

(2) 垂直奇/偶校验码 (可发现有限位错)

将被传输的信息进行分组,

组中每个分组 (字符) 的相同位进行奇/偶校验

例如: 4行7列信息组的垂直奇/偶校验码。

信息组:	0111001
	0010101
	0101011
	1010101

垂直奇校验字符	0101101
---------	---------

垂直偶校验字符	1010010
---------	---------

0111001001010101010111010101 0101101 (奇校验)

就差错的连续性而言, 强于水平奇/偶效验码

编码效率为: $PQ/P(Q+1)$ (假设信息分组占Q行P列)

(3) 水平垂直奇/偶校验码 (可用于纠一位错)

同时实施水平、垂直校验，只能使用偶校验。

例：4行7列信息组的水平垂直偶校验码。

信息组	校验位	
0111001	0	1
0010101	1	0
0101011	0	1
1010101	0	1
垂直偶校验字符	1010010	1
奇	0101101	0/1
0111001000101011010101101010101010100101		

若被传的信息分组占Q行P列， 编码效率为 $QP / (P+1)(Q+1)$

若收到的比特串为：

0111001010101011010101101010101010100101

能否判断其正确性？

分组：

0	1110010	偶1
1	0101011	奇1
0	1010110	偶1
1	0101010	偶1
1	0100101	偶1

正确比特串：

01110010**0**0101011010101101010101010100101

问题：出错 2 位，情况如何？

应用奇偶校验码的说明：

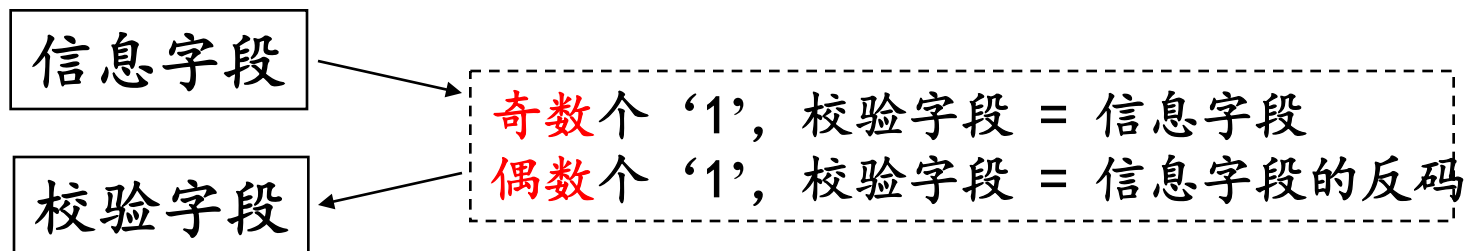
奇/偶校验码是最常用的校验码，尤其适用于大数据块的校验；应用时应注意信息字段（字符）的传输方向（依赖通信协议）；校验字段附接信息字段之后传输。

例：ASCII码字符‘S’（ $b_7b_6b_5b_4b_3b_2b_1=1010011$, 53H），奇校验，

低位先传时形成的校验码：11010011 →

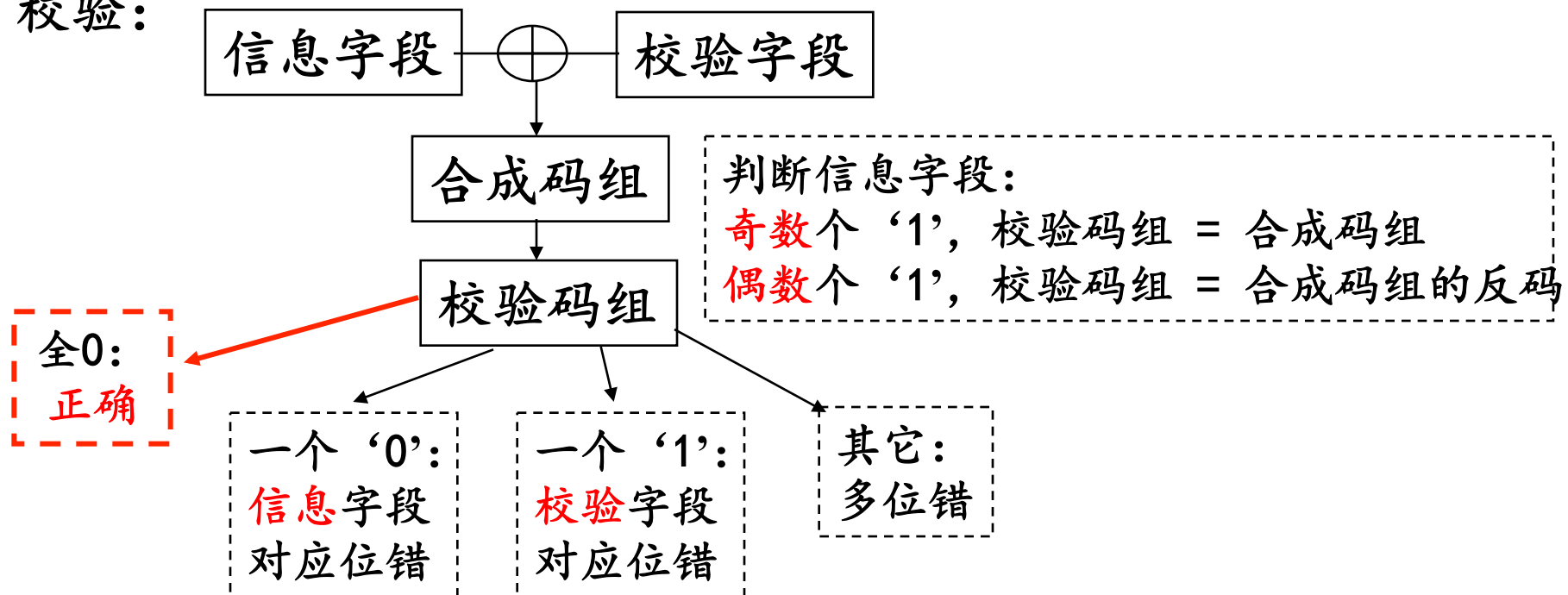
高位先传时形成的校验码：11100101 →

校验字段与信息字段占有相同的位数



传输: 信息字段 + 校验字段

校验:



接收码字	合成码组	信息字段奇/偶	校验码组	结果	原信息字段
0101101011	00000	奇（不变）	00000	正确	01011
1001001101	11111	偶（取反）	00000	正确	10010
0111101011	00100	偶（取反）	11011	信息位3错	01011
0101101001	00010	奇（不变）	00010	校验位4错	01011

正反码具有纠一位错的能力，其编码效率为50%。

正反码适用于小数据块（仅几位）的校验。

原理：当确定字段长度后，任意一个二进制位串（字段）都可以和一个系数仅为0和1取值的多项式一一对应。

例： 1010111: $x^6+x^4+x^2+x+1$

$x^5+x^3+x^2+x+1$: 101111

若信息字段为K位，校验字段为R位，则码字长度为 $N=K+R$ ；
任一合法码字都可由一个R阶多项式 $g(x)$ 产生。

合法码字 $V(x) = x^R m(x) + r(x) = A(x) g(x)$

$m(x)$ — K-1阶信息多项式，信息字段，

$r(x)$ — R-1阶校验多项式，校验字段，

$g(x)$ — R阶生成多项式

$$g(x) = g_0 + g_1x + g_2x^2 + \dots + g_{(R-1)}x^{(R-1)} + g_Rx^R。$$

其中： $g_i = 0$ 或 1 , $0 < i < R$; $g_0 = g_R = 1$,

校验和计算算法

- 设 $g(x)$ 为 R 阶，在帧的末尾加 R 个0，使帧为 $K+R$ 位，相应多项式为 $x^R m(x)$ ；
- 按模2除法用对应于 $g(x)$ 的位串去除对应于 $x^R m(x)$ 的位串，余式 $r(x)$ ；
- 按模2减法从对应于 $x^R m(x)$ 的位串中减去余数（等于或小于 R 位），结果就是要传送的带校验和的多项式 $V(x) \rightarrow$ 编出的码组 $V(x) = x^R m(x) + r(x)$ 。
- 在任何一种除法中，如果将被除数减掉余数，则剩下的差值一定可以被除数除尽

☆ 校验字段 $r(x)$ 生成方法之一：软件方法——多项式除法取余数 ¹⁶

例如：信息字段代码为： 1011001

对应 $m(x) = x^6 + x^4 + x^3 + 1$

生成多项式： $g(x) = x^4 + x^3 + 1$

对应代码： 11001

则 $x^4 m(x) = x^{10} + x^8 + x^7 + x^4$

对应代码： 10110010000

除法： $\begin{array}{r} 1101010 \\ 11001 \overline{) 10110010000} \end{array}$

$\begin{array}{r} 11001 \\ \underline{11001} \end{array}$

$\begin{array}{r} 11110 \end{array}$

$\begin{array}{r} 11001 \end{array}$

$\begin{array}{r} 11110 \end{array}$

$\begin{array}{r} 11001 \end{array}$

$\begin{array}{r} 11100 \end{array}$

$\begin{array}{r} 11001 \end{array}$

$\begin{array}{r} 1010 \end{array}$

校验字段 \longrightarrow 1010 (余数)

4阶项 $g(x)$ 产生4位校验字段
多项式除法使用模2加

接收方使用相同的 $g(x)$ 和除法
进行校验：

接收字段/生成码
如果除尽，则正确，
否则错。

形成码字： 10110011010

☆ 校验字段 $r(x)$ 生成方法之一：软件方法—多项式除法取余数¹⁶

注意事项: 1、根据 $g(x)$ 确定校验字段的位数, R阶R位;

2、余数不足R位，前面补0;

例如：信息字段代码为： 1011 对应多项式： $m(x)=x^3+x+1$

生成多项式: $g(x) = x^4 + x^3 + 1$ 对应代码: 11001

则 $x^4 \mid m(x) = x^7 + x^5 + x^4$ 对应代码: 10110000

除法:

$$\begin{array}{r} 1101 \\ 11001 \overline{) 10110000} \\ \underline{11001} \\ 11110 \\ \underline{11001} \\ 11100 \\ \underline{11001} \\ 11001 \end{array}$$

4阶项 $g(x)$ 产生4位校验字段
多项式除法使用模2加

接收方使用相同的 $g(x)$ 和除法
进行校验：

校验字段 0101 (余数)

接收字段/生成码

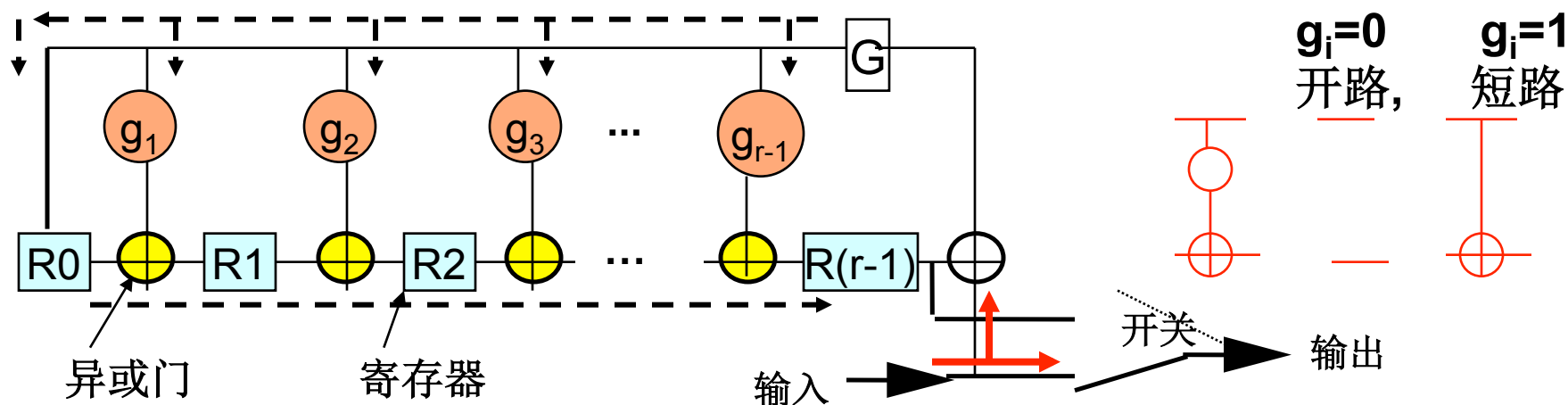
如果除尽，则正确，
形成码字：**10110101**

否则错。

☆ 校验字段 $r(x)$ 生成方法之二：硬件实现

根据 $g(x)$ 构建编码电路，信息字段作为输入，校验字段存寄存器。

$$g(x) = g_0 + g_1x + g_2x^2 + \dots + g_{(R-1)}x^{(R-1)} + g_Rx^R$$



注：为区分寄存器标识，图中表示的次项的‘ R ’暂记为‘ r ’；
移位寄存器（ R ）的个数： $g(x)$ 的次方；

输入：**发送**时为信息字段，或者**校验**时为接收到的码字。

移位寄存器 R ：发送时逐步形成校验字段，

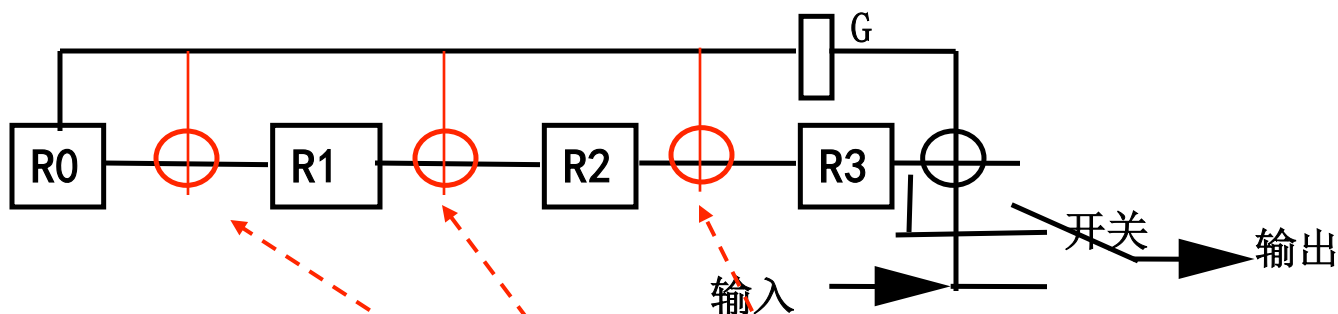
发送完后开关上拨，逐位输出 R 的内容；

校验时逐步形成校验结果（全0正确）。

硬件编码使用举例：

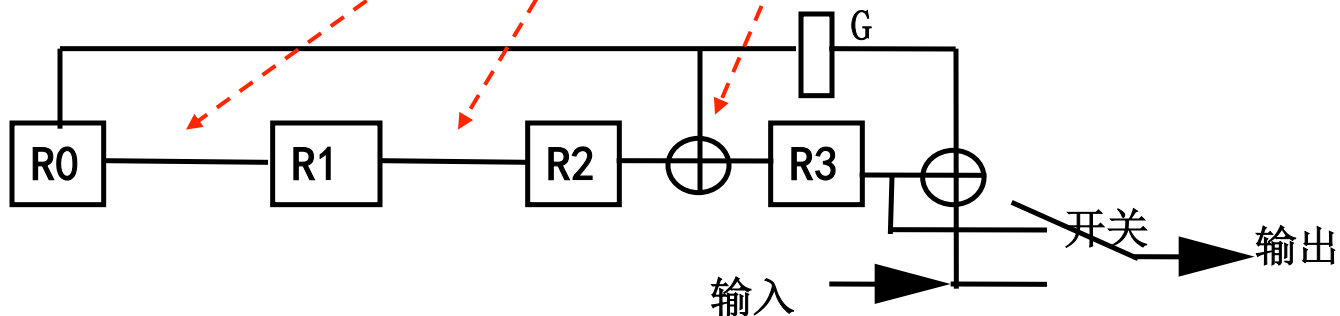
$g(x) = x^4 + x^3 + 1$ 的编码电路：

4阶多项式的通用编码电路：



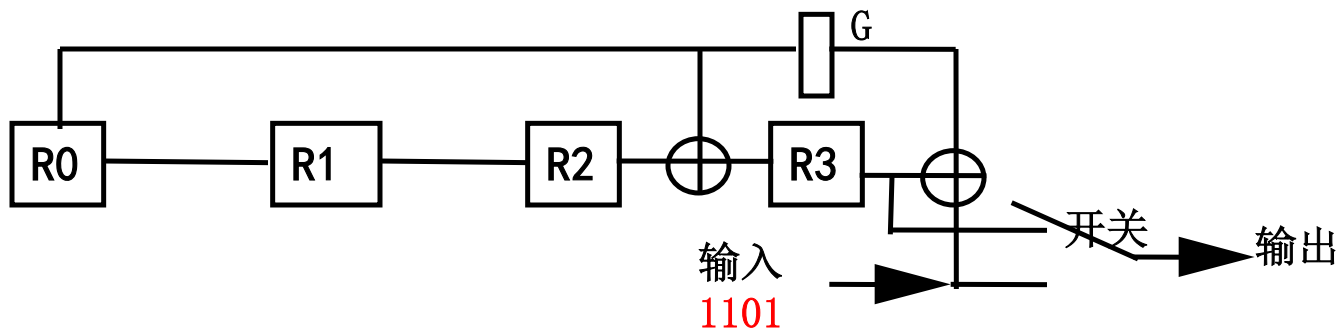
此例： g_0, g_4 恒为1， g_1, g_2 为0， g_3 为1；

$g(x) = x^4 + x^3 + 1$ 的编码电路：



硬件编码使用举例：

$g(x) = x^4 + x^3 + 1$ 的编码电路：



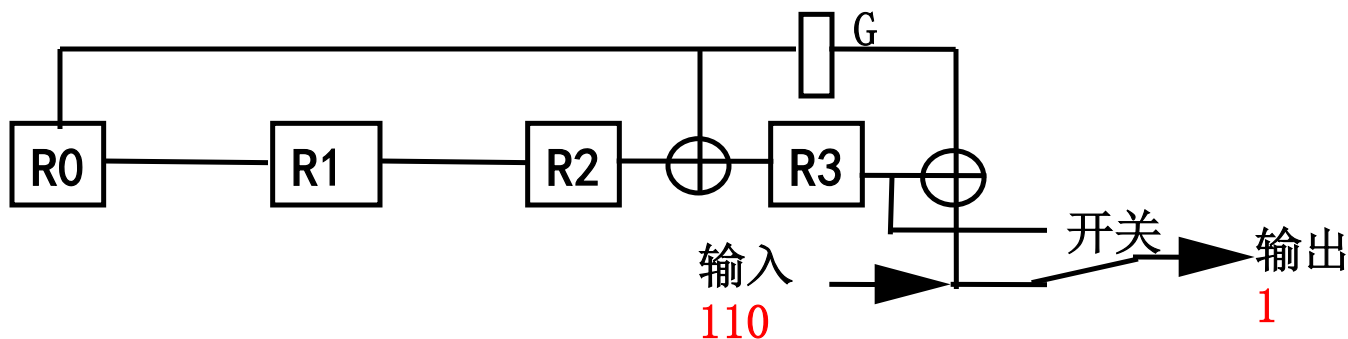
输入	R0	R1	R2	R3	输出
	0	0	0	0	

$R0 = G = \text{原}R3 \oplus I$
 $R1 = \text{原}R0$
 $R2 = \text{原}R1$
 $R3 = \text{原}R2 \oplus G$ (新R0)

校验时：整个码字输入完后，移位寄存器 R_i 应为全0，否则接收到的码字出错。

硬件编码使用举例：

$g(x) = x^4 + x^3 + 1$ 的编码电路：



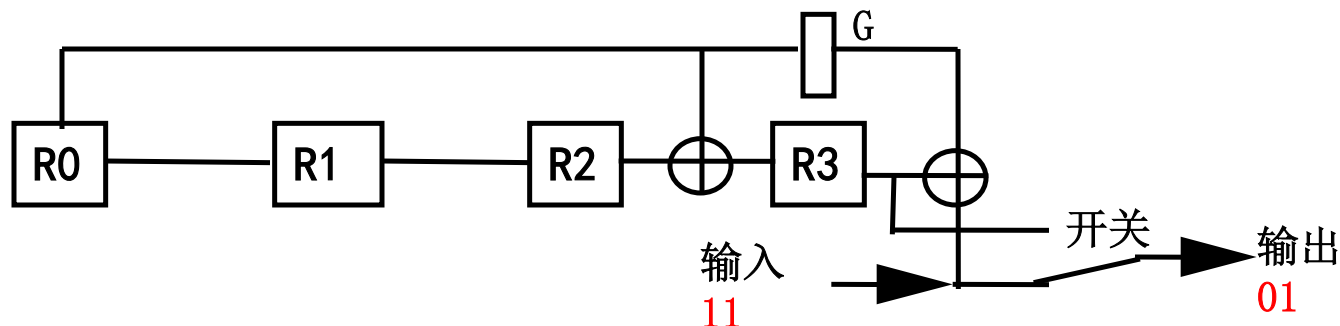
输入	R0	R1	R2	R3	输出
1	0	0	0	0	
	1	0	0	1	1

$$\begin{aligned}
 R0 &= G = \text{原}R3 \oplus I \\
 R1 &= \text{原}R0 \\
 R2 &= \text{原}R1 \\
 R3 &= \text{原}R2 \oplus G \quad (\text{新}R0)
 \end{aligned}$$

校验时：整个码字输入完后，移位寄存器 R_i 应为全0，否则接收到的码字出错。

硬件编码使用举例：

$g(x)=x^4+x^3+1$ 的编码电路：



输入	R0	R1	R2	R3	输出
1	0	0	0	0	
0	1	0	0	1	1
	1	1	0	1	01

$$R0=G=\text{原}R3 \oplus I$$

$$R1=\text{原}R0$$

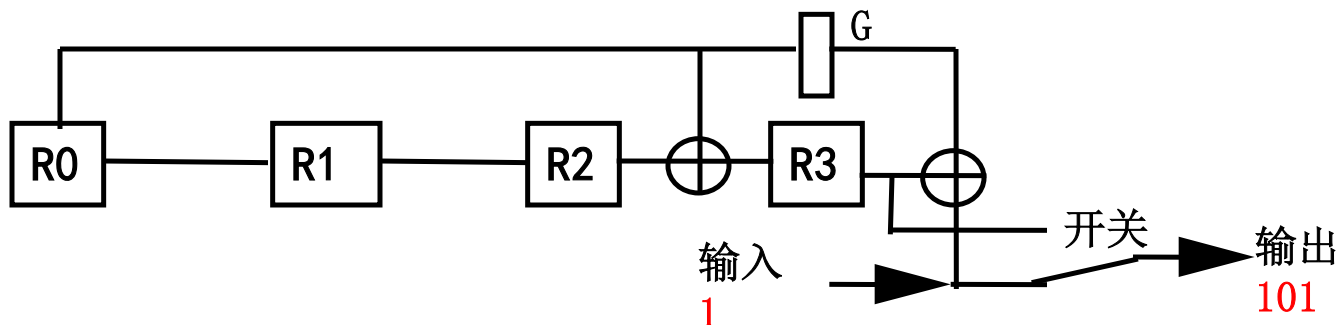
$$R2=\text{原}R1$$

$$R3=\text{原}R2 \oplus G \text{ (新}R0\text{)}$$

校验时：整个码字输入完后，移位寄存器 R_i 应为全0，
否则接收到的码字出错。

硬件编码使用举例：

$g(x) = x^4 + x^3 + 1$ 的编码电路：



输入	R0	R1	R2	R3	输出
1	0	0	0	0	
0	1	0	0	1	1
1	1	1	0	1	01
	0	1	1	0	101

$$R0 = G = \text{原}R3 \oplus I$$

$$R1 = \text{原}R0$$

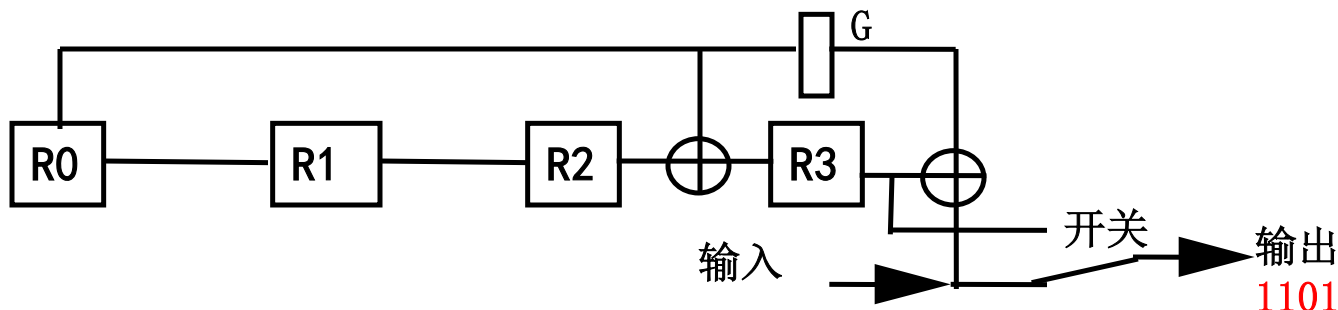
$$R2 = \text{原}R1$$

$$R3 = \text{原}R2 \oplus G \quad (\text{新}R0)$$

校验时：整个码字输入完后，移位寄存器 R_i 应为全0，
否则接收到的码字出错。

硬件编码使用举例：

$g(x) = x^4 + x^3 + 1$ 的编码电路：



输入	R0	R1	R2	R3	输出
1	0	0	0	0	
0	1	0	0	1	1
1	1	1	0	1	01
1	0	1	1	0	101
	1	0	1	0	1101

$$R0 = G = \text{原}R3 \oplus I$$

$$R1 = \text{原}R0$$

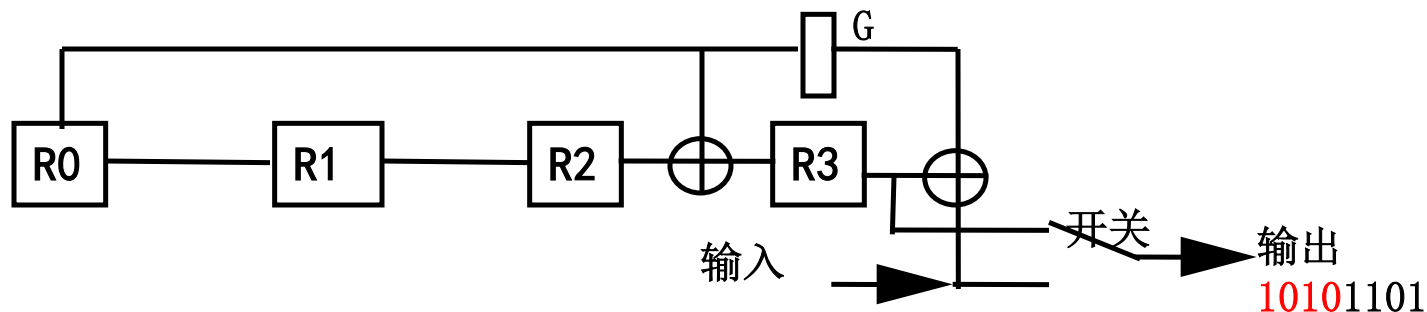
$$R2 = \text{原}R1$$

$$R3 = \text{原}R2 \oplus G \quad (\text{新}R0)$$

校验时：整个码字输入完后，移位寄存器 R_i 应为全0，
否则接收到的码字出错。

硬件编码使用举例：

$g(x) = x^4 + x^3 + 1$ 的编码电路：



输入	R0	R1	R2	R3	输出
1	0	0	0	0	
0	1	0	0	1	1
1	1	1	0	1	01
1	0	1	1	0	101
	1	0	1	0	1101

$$R0 = G = \text{原}R3 \oplus I$$

$$R1 = \text{原}R0$$

$$R2 = \text{原}R1$$

$$R3 = \text{原}R2 \oplus G \quad (\text{新}R0)$$

校验时：整个码字输入完后，移位寄存器 R_i 应为全0，
否则接收到的码字出错。

CRC校验码使用广泛。

硬件电路主要用于自动生成码字，或者进行校验；
多项式除法主要用于软件编程。

常用的CRC生成多项式 $g(x)$ 为：

$$\text{CRC16} = x^{16} + x^{15} + x^2 + 1$$

R=16, IBM专用

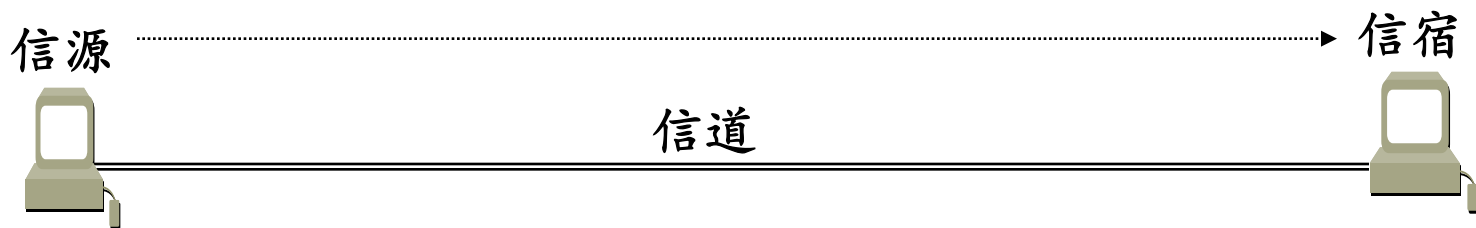
$$\text{CRC16} = x^{16} + x^{12} + x^5 + 1$$

R=16, CCITT专用

$$\text{CRC32} = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

R=32, LAN中常用

通信原理小结：

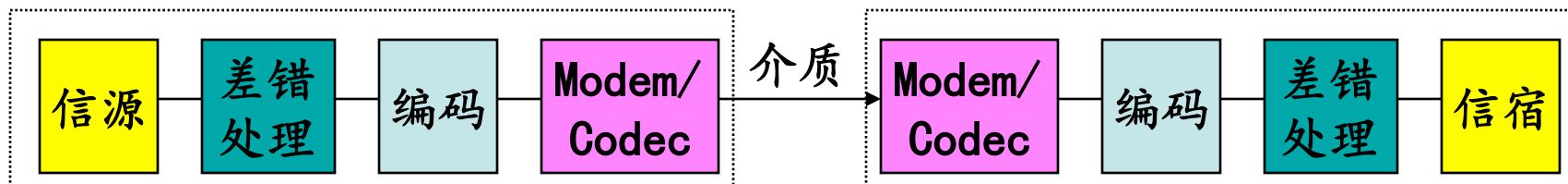


编码：字符到比特到线路（数字脉冲）；

Modem和Codec：适应模拟和数字信道的特性；

差错处理：解除线路噪声问题，提高正确性；

结论：现有知识的合理应用应可支持计算机间的信息可靠传输。



2.9 传输控制规程

目的：协调通信双方的动作，保证数据信息传输的正确性。

异步传输控制规程：实现以字符为单位的传输；

同步传输控制规程：实现以多个字符或比特组合成的数据块为单位的传输；

2.9.1 面向字符型的传输控制规程 （支持字符数据传输）

规程实例：二进制**同步**通信规程(BSC)

(1) 目标

支持任意字符型数据(文本) 在计算机之间的传输。

当用于支持位序列传输时，位序列将以7位或8位（有校验位）一组形成字符，不足位补‘0’，最终形成字符串。用户以其它方式（如约定**前缀n位**）说明位序列的实际长度。

例： **010**1011 1010101 1100010 1011101 10100**00** (**补0**)

ASCII + **U** **b** **]** **P** +Ub]P

★ 数据块的组成:

为了体现信息的逻辑性,借用了字符编码(如ASCII、EBCDIC码)的10个特殊字符(控制字符):

SOH (0/1, 标题开始)、STX (0/2, 正文开始)、
ETX (0/3, 正文结束)、EOT (0/4, 传输结束)、
ETB (1/7, 组结束)、DLE (1/0, 数据转义)、
ENQ (0/5, 询问)、SYN (1/6, 同步)、
ACK (0/6, 确认)、NAK (1/5, 否认)等。

例如: SOH xxxxxx STX xxxxxxxxxxxxxxxxxxxx ETX

潜在问题: 数据本身可能蕴含这些控制字符:

例: SOH xxx ETB xx STX x DLE ETX xxx ETX

(2) 数据块的组成和歧义问题的解决

★ 歧义问题的解决：字符转义和字符填充；

目的：支持任意字符数据传输。

字符转义：区分控制字符和数据块中的‘假’控制字符，在控制字符之前增加一个**前缀**转义字符DLE形成特定语义的**控制字符组**，增加匹配控制字符组的难度。

DLE SOH、DLE STX、DLE ACK、DLE NAK等

字符填充：在数据DLE前**再增加**一个DLE，使其转义为一般字符，避免数据中同时出现DLE和控制字符时可能的歧义。

原意：SOH xxxETBxx STX xDLEETXxxx ETX

转义：DLESOH xxxETBxx DLESTX xDLEETXxxx DLEETX

填充：DLESOH xxxETBxx DLESTX xDLEDLEETXxxx DLEETX

动作：若收到两个连续的DLE，则丢弃一个，并认为保留的是数据DLE字符。（此处的控制字符仅仅是一个字符！）

歧义解决方案的补充说明

★ 因特网中歧义问题的解决方案

IP数据报用字符END（0Xc0）进行数据块（报文）定界；

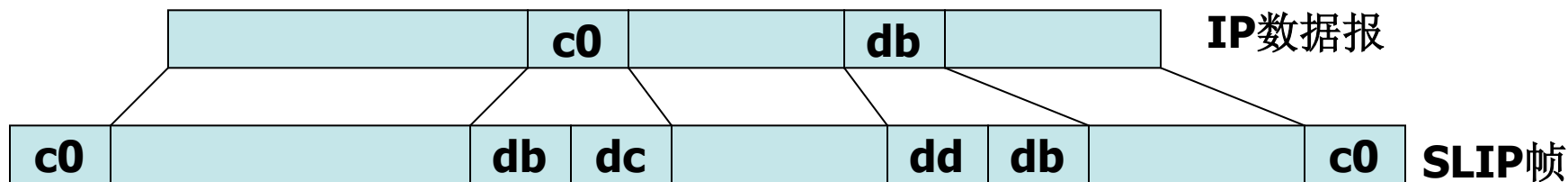
转义字符： 0Xdb；

方案：如果在IP数据中出现END字符（0Xc0），就用2字节字符0Xdb， 0Xdc代替；

如果IP数据报中有字节为0Xdb，就用0Xdd，0Xdb代替；

接收方执行相反的还原过程。

举例：SLIP（Serial Line IP）协议



(3) 通信双方交换的信息序列类型:

正文信息: 通信双方正常交换的数据 (双方均可发送) ;

控制序列: 控制通信双方交换信息的过程。

★ 控制序列

确认 (ACK) : SYN SYN 0/1 DLE ACK

增加0或1的目的在于区分应答的对象;

否认 (NAK) : SYN SYN DLE NAK

询问 (ENQ) : SYN SYN 站地址 DLE ENQ

拆链 (EOT) : SYN SYN DLE EOT

基本格式，含标题和正文两部分：

SYN SYN DLESOH 标题 DLESTX 正文 DLEETX 校验码

无标题格式，省略标题时的数据块格式：

SYN SYN DLESTX 正文 DLEETX 校验码

成组格式(多段正文)：

SYN SYN DLESOH 标题 DLESTX 正文组1 DLEETB 校验码

SYN SYN DLESTX 正文组2 DLEETB 校验码

... ..

SYN SYN DLESTX 正文组n DLEETX 校验码

校验码：（1）水平垂直奇偶校验

（2）当采用循环校验时， $g(x)=x^{16}+x^{12}+x^5+1$ ，

校验对象为SYN SYN之后的所有内容。

其中：SOH：标题开始，STX：正文开始，ETX：正文结束；ETB：组结束。

建立链路：建立链路指建立通信双方的收发关系

数据传输：在链路建立的基础上，传输数据

拆除链路：释放通信双方已建的关系

★ 建立链路：

主叫站

被叫站

ENQ. 被叫地址 →
← NAK (未准备好接收)

ENQ. 被叫地址 →
← 0. ACK (收发关系建立)

★ 数据传输:

正文1 →
← 1. ACK
正文2 →
← 0. ACK

正文3 →
超时(传输错误, 丢弃正文3)

正文3 → (重发)
← 1. ACK

正文3 →
超时(ACK丢失) ← 1. ACK

正文3 → (重发)
(重收, 收方自动丢弃正文3)
← 1. ACK

← ENQ (被叫站要发数据)
ACK. 0 →
← 正文1
ACK. 1 →
.....

★ 释放链路:

任一方:

EOT →
← EOT

面向字符型的传输控制规程的特点：

半双工的停-等协议、超时重发，传输效率较低。

数据块和控制序列格式不统一，易引起二义性。

控制序列的差错校验能力仅依赖于控制字符本身的字符奇偶校验，可靠性较低。

以字符传输为目标，适用性较弱。

仅需要很少的缓存容量，规程简单，易于实现。

支持任意二进制数据的传输。

常用的标准：ISO的高级数据链路控制规程 (ISO HDLC)

CCITT 的平衡型链路访问规程 (CCITT X25 LAP-B)

HDLC介绍

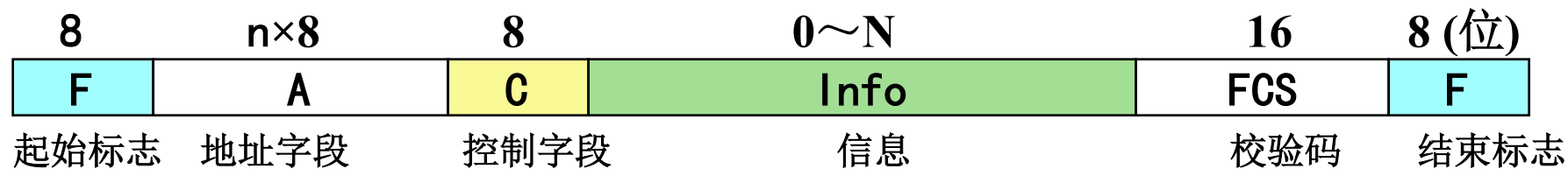
(1) 交换对象

帧： 通信双方交换的最小单位——比特序列。

帧的组成： 帧间隔符 比特序列 终止标志

011111100011011000010110110011011101111110

(2) 一般帧结构



★ F（间隔符模式）：“01111110”——同步符号、帧之间的填充字符

01111110111110000111100010101011010101010101001010100111111001111110

☆ “0”比特插入法：避免帧内出现间隔符 ‘01111110’模式。

发送方：逢5个 ‘1’，自动插入一个 ‘0’；

接收方：若5个 ‘1’后为 ‘0’，移去 ‘0’；否则帧结束；

01111110111110001110001100111111101101111110111010111111001111110

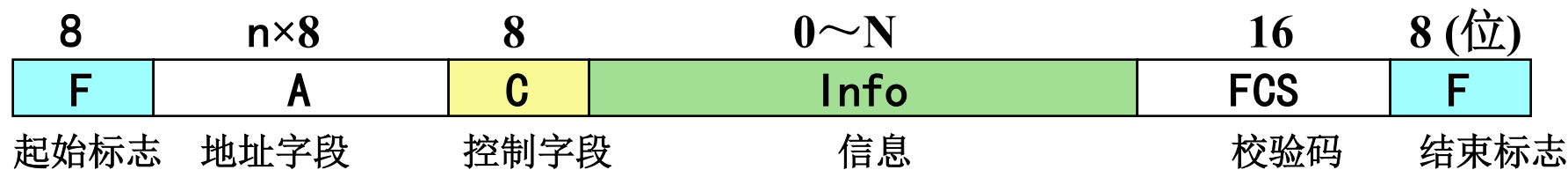
011111101111100011100011001111111010011011111010111010111111001111110

★ A(地址字段)— 对方的地址

对应地址字段所属的字节首位为 ‘0’，表示后继字节仍为地址字段，

字节首位为 ‘1’，表示本字节为地址字段的最后字节。

(2) 一般帧结构



★ C (控制字段) : 区分帧的类型

	0	1	2	3	4	5	6	7 (位)
信息帧 (I)	0	Ns			P/F	Nr		
监控帧 (S)	1	0	Type		P/F	Nr		
无编号帧 (U)	1	1	M1		P/F	M2		

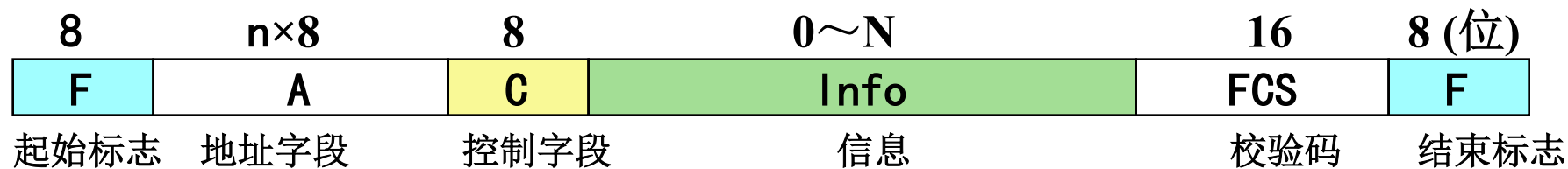
☆ 信息帧 (I帧, $C_0 = 0$) , 用于传输用户数据

Ns (发送帧序号) 说明本帧对应的帧序号 (采用模8计数) ,
每发一帧, Ns模8计数一次;

Nr (待收帧序号) : 希望接收对方帧的序号,
(采用模8计数) ;

Nr 隐含指示该序号之前的所有帧已被正确接收;

(2) 一般帧结构



★ C (控制字段)：区分帧的类型



☆ 信息帧中 P/F: (Probe/Final--查询/终止指示位)

—要求对方做出响应;

建立链接阶段: 发方置 P=1, 询问对方是否同意链接;

对方需立即响应, 并在响应中置 F=1。 (例: 同意建链)

数据传送阶段: 发送方置 P=1, 询问对方是否有数据待发;

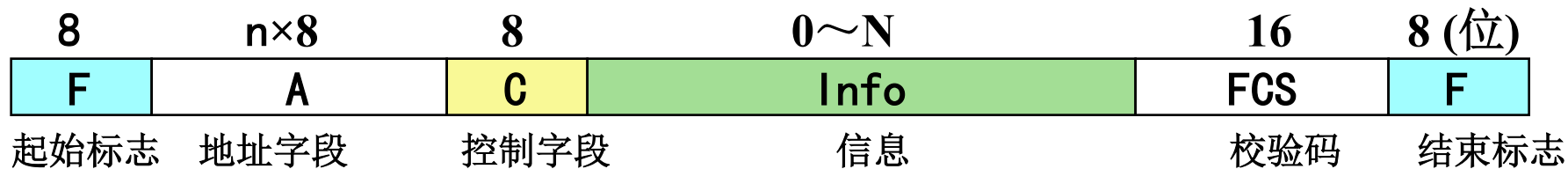
如果对方有数据待发, 开始发送信息帧 (I);

对方可连发多帧, 并在最后一个 I 帧置 F=1, 传输完毕;

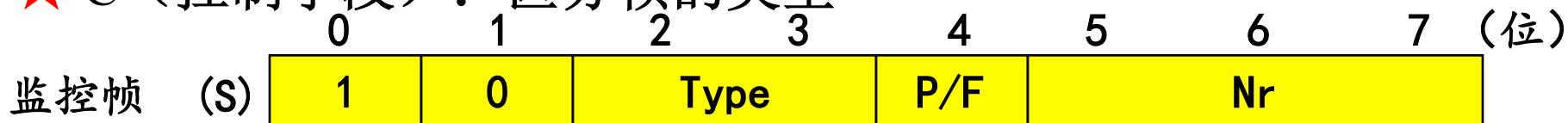
如果对方无数据待发, 直接以 S 帧 (F=1) 进行响应。

△ P/F 必须一一对应, 在接到 F=1 的帧之前, 不允许再发 P=1 的帧

(2) 一般帧结构



★ C (控制字段)：区分帧的类型



☆ 监控帧 (S帧, $C_0C_1 = 10$)，用于表示接收状态 (4种类型)。

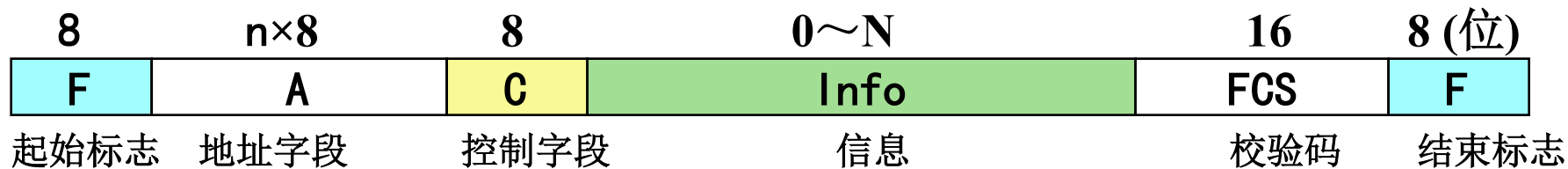
Type=00, 接收准备就绪(RR)，准备接收序号为Nr的帧；

Type=10, 未准备就绪(RNR)，告诉对方已经收妥Nr以前的所有帧，但希望对方暂缓发送Nr帧；

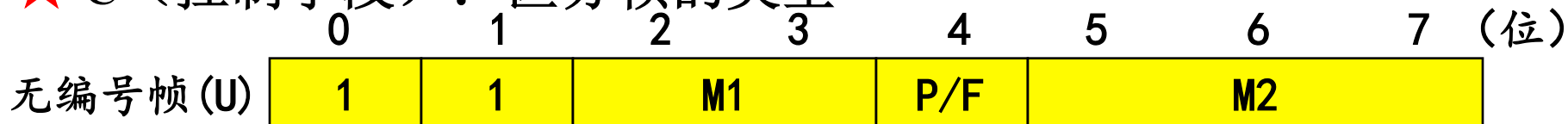
Type=01, 拒绝接收(REJ)，告诉对方已经收妥Nr以前的所有帧，但编号为Nr的帧有差错，希望对方重发序号为Nr及其以后的所有帧；

Type=11, 选择接收(SREJ)，类似REJ监控帧，但希望对方仅仅重发第Nr帧。

(2) 一般帧结构



★ C (控制字段) : 区分帧的类型



☆ 无编号帧 (U帧, $C_0C_1 = 11$) , 用于控制链路, 不含序号;
M1, M2表示帧类型:

$M(M1M2)=11000$ (SARM), 置本次链路为异步响应模式;

$M(M1M2)=00001$ (SNRM), 置本次链路为正常响应模式;

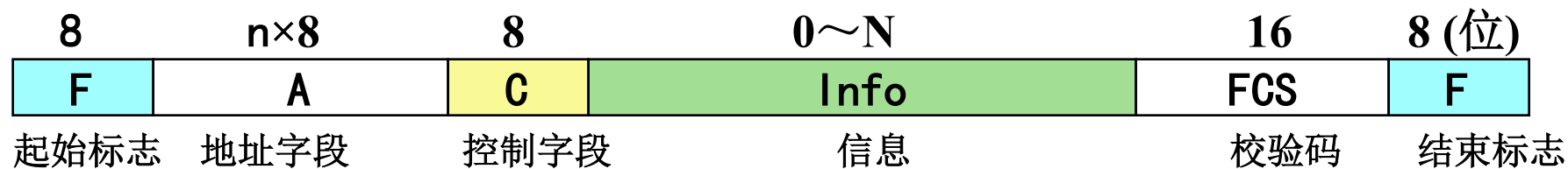
$M(M1M2)=11100$ (SABM), 置本次链路为异步平衡模式;

$M(M1M2)=00010$ (**DISC**), 请求释放(拆除)本次链路;

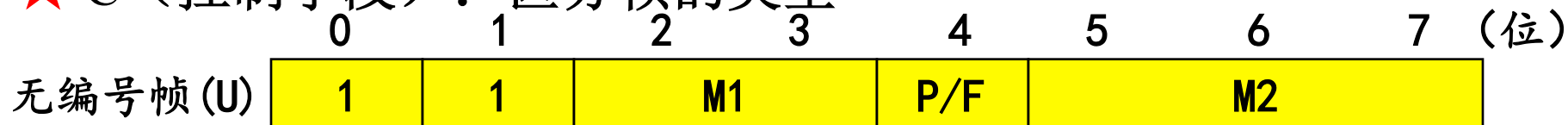
$M(M1M2)=00110$ (**UA**), 对对方命令进行确认, 类ACK;

$M(M1M2)=10001$ (**CMDR**), 对对方命令予以否定, 类NAK;

(2) 一般帧结构



★ C (控制字段)：区分帧的类型



☆ 无编号帧 (U帧, $C_0C_1 = 11$)，用于控制链路，不含序号；
M1, M2表示帧类型：

SABM请求建链

CMDR不同意建链，否认

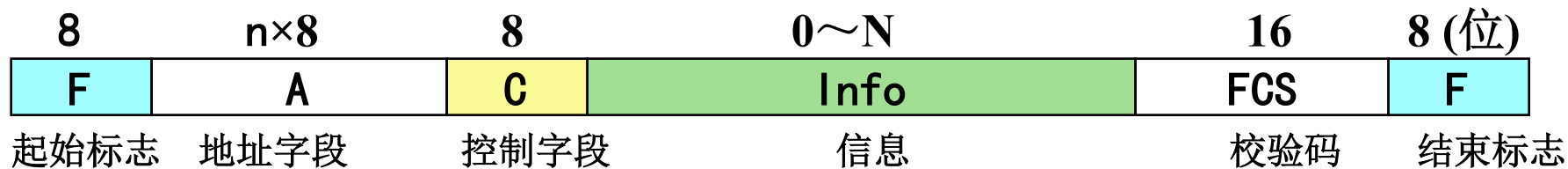
SABM请求建链

UA 同意建链，确认

DISC请求拆链

UA 同意拆链，确认

(2) 一般帧结构



★ Info(信息字段): 用于携带用户数据;
仅在I帧中出现, 任意位串 (已实施 ‘0’ 比特插入)。

★ FCS(校验字段): 对A、C和Info字段进行循环校验;

$$g(x) = x^{16} + x^{12} + x^5 + 1 \quad (\text{CCITT和ISO制定})$$

$$g(x) = x^{16} + x^{15} + x^2 + 1 \quad (\text{IBM公司制定—源于SDLC})$$

因为HDLC的帧中至少包含A、C、FCS字段,
因此帧长应大于等于32位。

(3) 窗口机制——提高效率的保障

传输窗口：通信双方同意在同一条链路上连续使用的信息帧序号

窗口尺寸：通信双方协商同意的在同一条链路上可连续发送且未被认可的信息帧个数；

HDLC 窗口尺寸确定为 $2^3-1=7$ ，即任一方可以最多连续发送7帧无需对方的确认，帧序号循环使用（模8）。

在信息帧中， 表示当前窗口的情况；

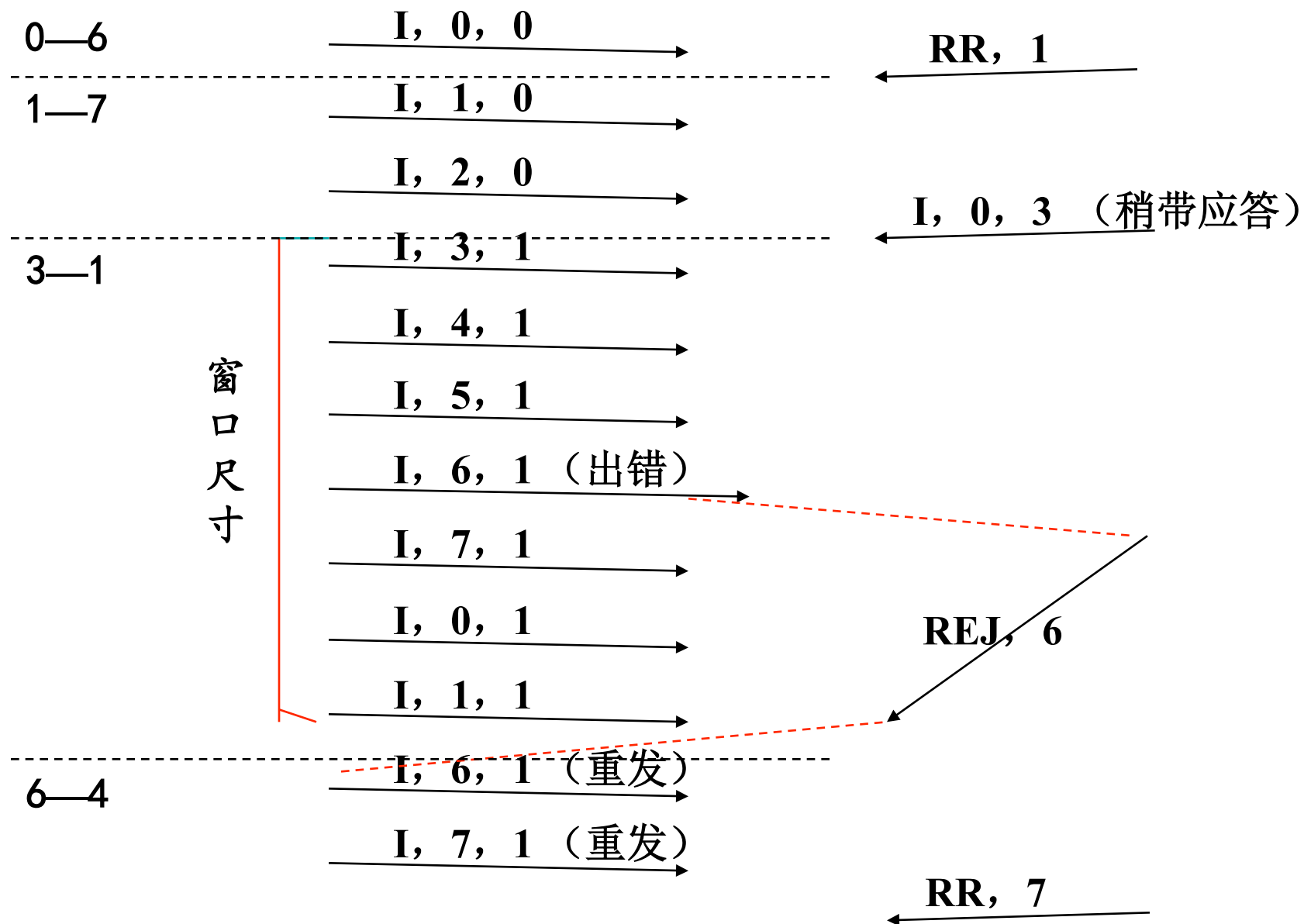
窗口尺寸为 $2^3-1=7$ 的原因：明确待收帧 N_r 。

窗口尺寸越大，传输/应答效率越高。

若窗口尺寸为8，则发方在收到对方发来的 $N_r=0$ 的帧后，连续发送 8 帧：0、1、2、...、7；

若再收到 $N_r=0$ 的帧，如何理解？ 8 帧都收妥？ 1 帧未收到？

送且携带应答；否则若在一定时间内未收到后继信息，则发RR帧，准备继续接收后续帧。



(4) 数据交换过程

- ★ **建立链路：** 置初始化模式和置响应模式；
通信双方确认可以通信，并协商通信的模式。
- ★ **传输信息帧：** 通信双方通过交换I帧和S帧，
完成双方的高层数据交换
全双工方式，双方均可发送信息帧和监控帧。
- ★ **释放(拆除)链路：** 任意一方在发送完数据之后，
均可用DISC命令要求拆除链路；
对方同意拆链，用UA命令响应。

☆ 建立链路

A站	方向	B站	说 明
B:RR, P=1, 0	→		; A询问B, P=1;
		← B:RIM, F=1	; B要求初始化;
B:SIM, P=1	→		; A置B于初始化模式
		← B:UA, F=1	; B肯定答复, 进入联机
B:SNRM, P=1	→		; A置B正常响应模式
		← B:UA, F=1	; B肯定答复

A站	方向	B站	说 明
B: RR, P=1, 0	→		; A要求B传输(全双工)
B: I, 0, P=0, 0	→		; A发0号帧, 可接B的0号帧
	←	B: I, 0, F=0, 1	; B发0号帧, 可接A的1号帧
B: I, 1, P=0, 1	→		; A发1号帧, 可接B的1号帧
B: I, 2, P=0, 1	→		; A发2号帧, 可接B的1号帧
	←	B: I, 1, F=0, 3	; B发1号帧, 可接A的3号帧
B: I, 3, P=0, 2	→		; A发3号帧, 可接B的2号帧
	←	B: I, 2, F=1, 4	; B发最后2号帧, 可接A的4号
B: I, 4, P=0, 3	→		; A发4号帧, 收妥B的2号帧
B: I, 5, P=1, 3	→		; A发5号帧, 要求B的确认
	←	B: RR, F=1, 6	; B确认A的5号帧
		; 双方继续交换信息

A站	方向	B站	说 明
	←—	B:RD	; B请求拆链
B:DISC	—→		; A命令拆链
	←—	B:UA	; B对拆链进行确认

- ★ HDLC 统一的帧格式：数据、命令和响应具有统一格式，易于实施；
- ★ 采用“0”比特插入法：支持任意的比特流传输，提高了信息传输的透明性；
- ★ 采用窗口机制和捎带应答，支持全双工工作方式，提高信息传输的效率；
- ★ 采用帧校验序列，以及窗口序号的设置，提高信息传输的正确性和可靠性。

通信在质和量两方面的要求：

“质”：保证传输信息的可靠性—差错处理；

“量”：充分利用线路资源，为较多用户传输较多信息；

“多路复用和集中传输”主要解决如何充分利用信道的问题。

2.10.1 多路复用

起因：用户操作具有间断性；

目的：使得多路信号可以共用一个信道，

将多路信号组合在一条物理信道上传输，充分利用信道容量。

原理：当物理信道的可用带宽超过单个原始信号的带宽时，

将物理信道的总带宽分割成若干个和被传输的单个信号带宽相同（或者略宽一点）的**子信道**，并利用每个子信道传输一路信号，达到多路信号共用一个信道的目的，节省线路资源。

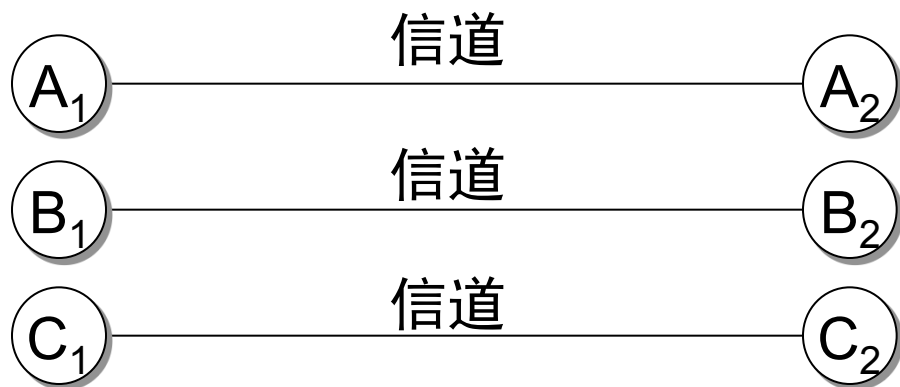
技术支持：组合和分离不同用户的信息。

复用 (Multiplexing) 是通信技术中的基本概念,

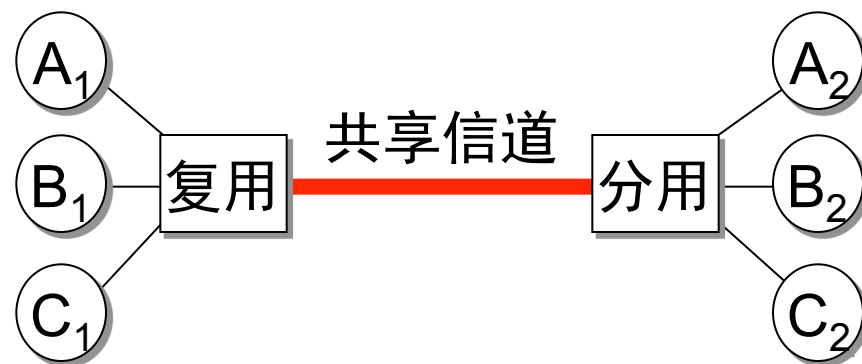
类同n车道的公路支持多车并行。

复用: 将多路信号汇集到一个信道 (共享信道);

分用 (解复用): 将共享信道的信息分解给不同的收方。



不使用复用技术



使用复用技术

(1) 频分多路复用 (FDM)

频分多路复用主要用于模拟信道的复用（铜线、微波线路）

原理：对整个物理信道的可用带宽进行分割，并利用载波调制技术，实现原始信号的频谱迁移，使得多路信号在整个物理信道带宽允许的范围内，在频谱上的不重叠，从而共用一个信道。



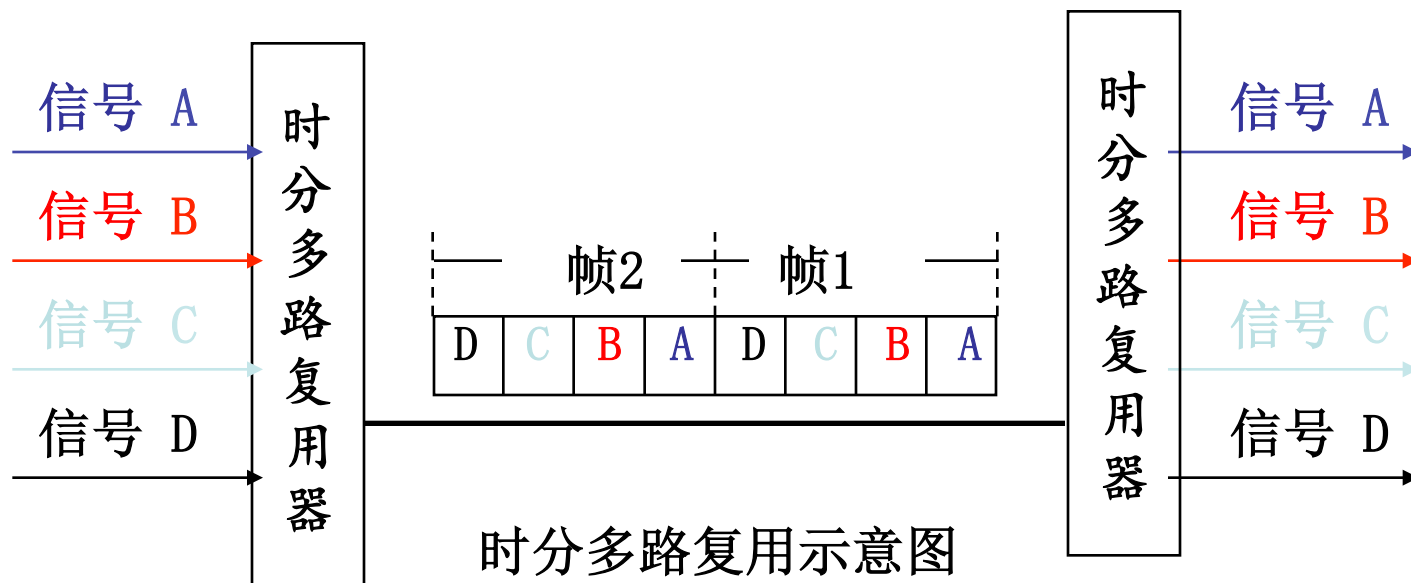
频分多路复用示意图

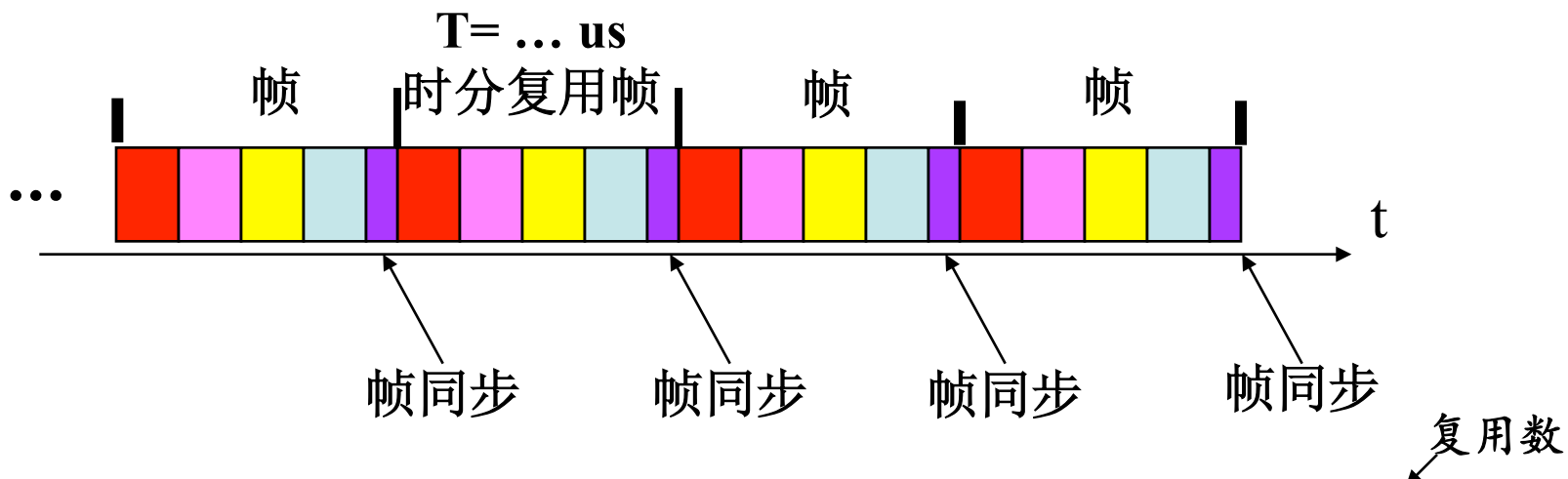
(2) 时分多路复用 (TDM)

时分多路复用主要用于数字信道的复用；

原理：当物理信道可支持的比特传输速率 (bps) 超过单个原始信号要求的数据传输速率时，可以将该物理信道划分成若干时间片，并将各个时间片轮流地分配给多路信号，使得它们在时间上不重叠。

习惯上，也将各子信道组合的结果称之为‘帧’。



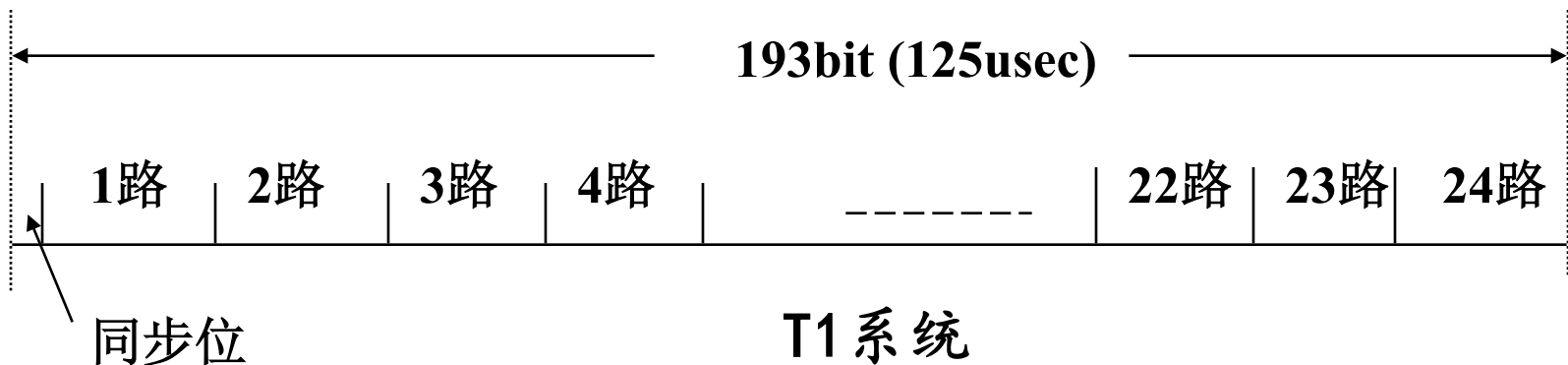


若：X位/帧，帧/Y微秒，则线路速率=X/Y，用户速率 $\approx X/YN$

★ **T1系统 (北美)** : 24路/帧 、 8bit/路 、 帧/125微秒,
帧 长 度: $24 \times 8 = 192\text{bit} + 1$ (同步位),
传输速率: $193/125\text{微秒} = 1.544\text{Mbps}$ (1.536Mbps)

借助T1系统的语音传输:

语音带宽: 4Khz; 采样频率: 8Khz (1次采样/125us);
量化级: 256 (8位) — 传输速率: 64Kbps;
24路语音信息的传输。

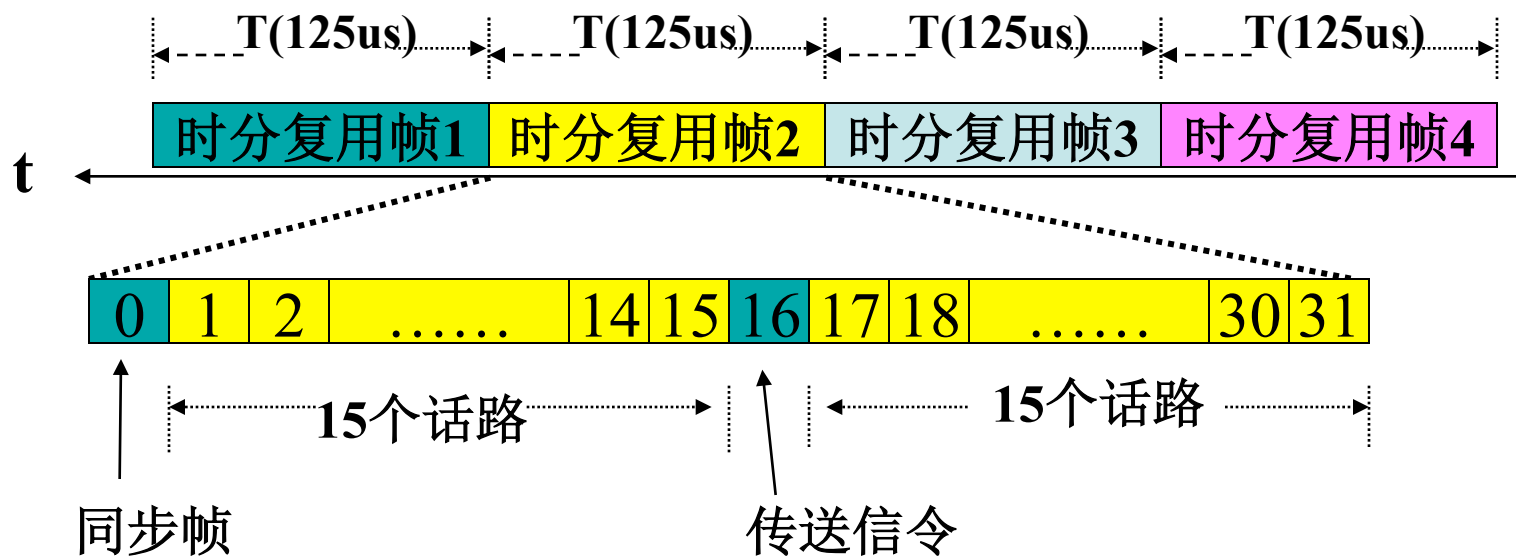


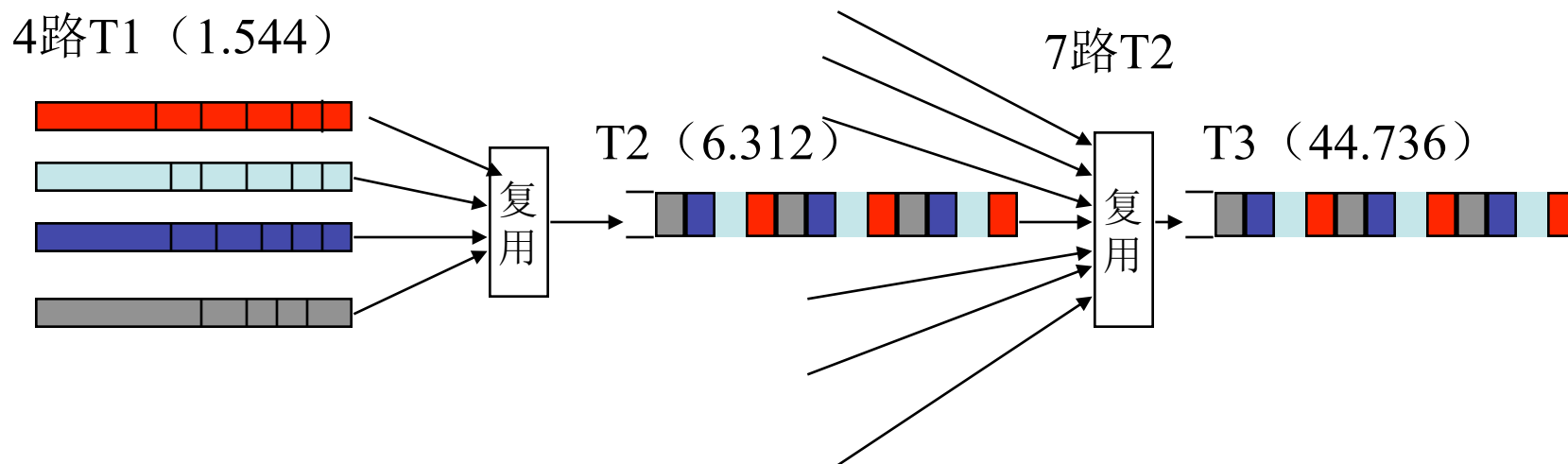
★ **E1系统（欧洲）**：32路/帧、8bit/路、125微秒/帧

传输速率： $32 \times 8 / 125 \text{微秒} = 2.048 \text{Mbps}$

0路和16路用于同步和控制信号。

E1系统可支持30路语音信息的传输。





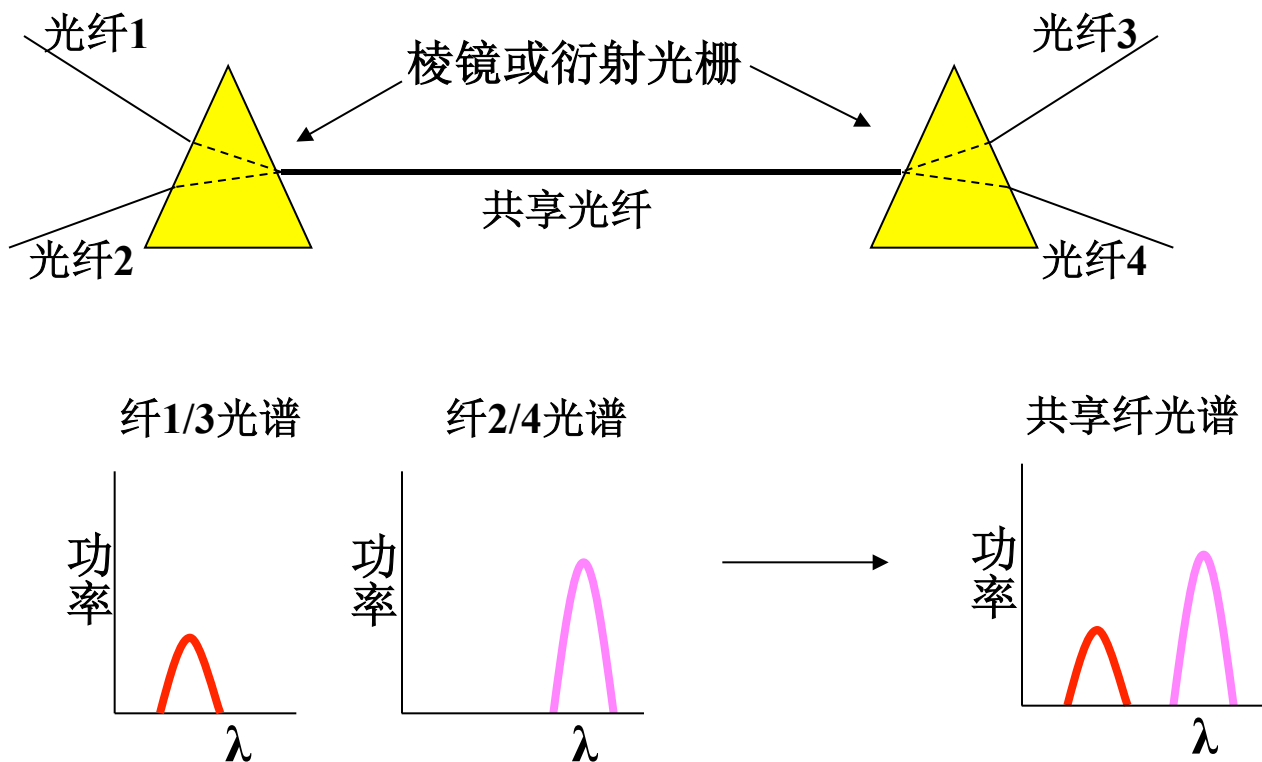
T1: 1.544Mbps, T2: 6.312Mbps, T3: 44.736Mbps, ...。
部分比特用于帧间间隔和同步。

(3) 波分多路复用 (WDM)

9

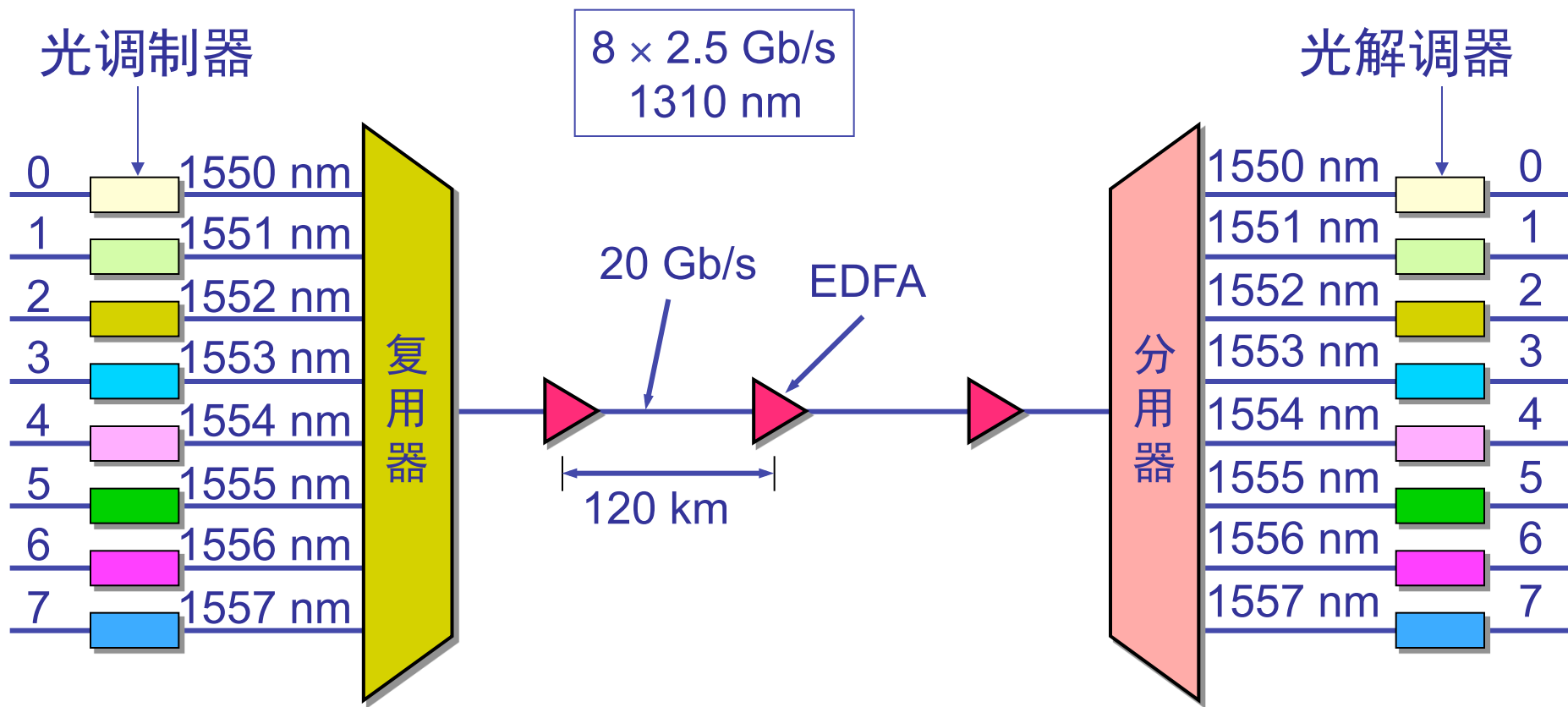
波分多路复用主要用于光纤信道；

原理：类似**频分**多路复用 (FDM)，将不同路信号调制成不同波长的光，并借助同一光纤信道传输；接收端进行光分离处理。



波分多路复用示意：

各路信息调制为不同波长的光信号经复用器复用到一起一根光纤，长距离传输时光信号强度发生衰减，需要增强能量。



(4) 集中传输 (对多路复用技术的改进)

多路复用的特点：**各个子信道**（频分多路复用中的子频段，时分多路复用中的时间片，波分多路复用中的波长）**被静态地分配给各路信号传输**，接收方可以直接通过识别固定子频段、时间片或者波长来完成信号分离。

不足之处：**信道利用率不够高**，信号的传输往往是间断的，在某个时刻，子信道会出现空闲状态（无数据）

解决办法：集中传输！

问题1：如何区分子信道的数据？

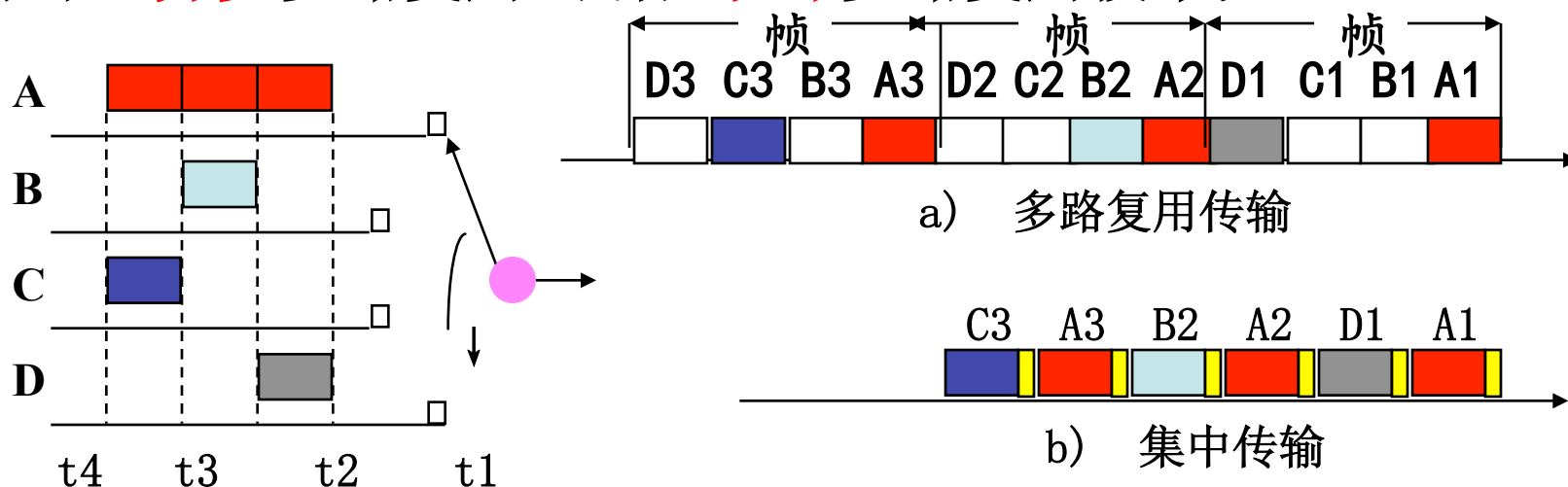
解决方法：数据块前增加地址标识；

结果：子信道可动态分配给不同用户，系统可以容纳更多用户；

问题2：瞬时输入可能大于总的输出，产生信息丢失；

改进：集中传输设备具有缓冲存储的能力，
临时保存输入的信息，并等待空闲的信道。

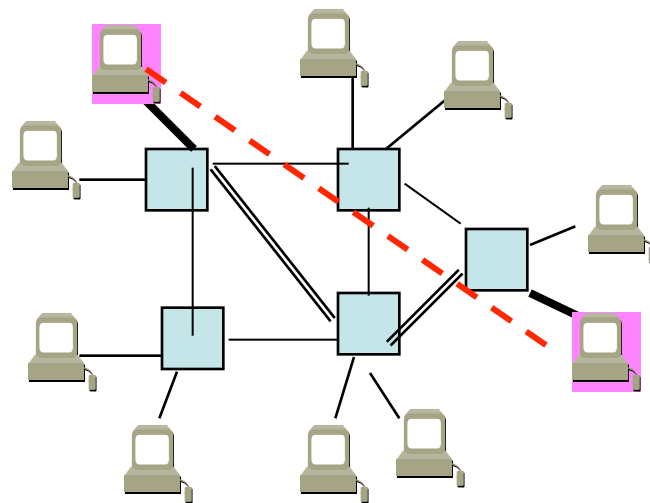
称呼：**异步**多路复用 或者 **统计**多路复用技术；



时分多路复用 和 集中传输 的 比较

2.11 数据交换

限于媒体的长度和成本限制，为容纳更多的用户，实际通信采用多点接续方式进行，中间结点执行数据交换的过程。



中间结点（交换设备）：不关心被传输的数据内容，仅执行数据交换的动作，将数据从一个端口**交换**到另一端口，继而传输到另一台中间结点，直至目的地。

数据传输的过程实质上是数据交换的过程。

结点：用于数据交换的中间设备，

站点：发送和接收数据的终端设备。

数据交换方式：线路交换、存储交换（报文交换和分组交换）

物理连接工作方式，无存储能力，原理类似于电话系统；
过程(三阶段)： 建立线路、占用线路并传输数据、释放线路。

建立线路：发起方站点向接收方站点发送请求，该请求将通过中间结点传输至终点；

如果中间结点有空闲的输出线路（端口），分配线路，**接受并下传请求**，直至终点。否则**拒绝请求**，并释放已建线路。

线路一旦分配，在未释放前，将不能被其它站点所使用，即使线路上并没有数据传输。

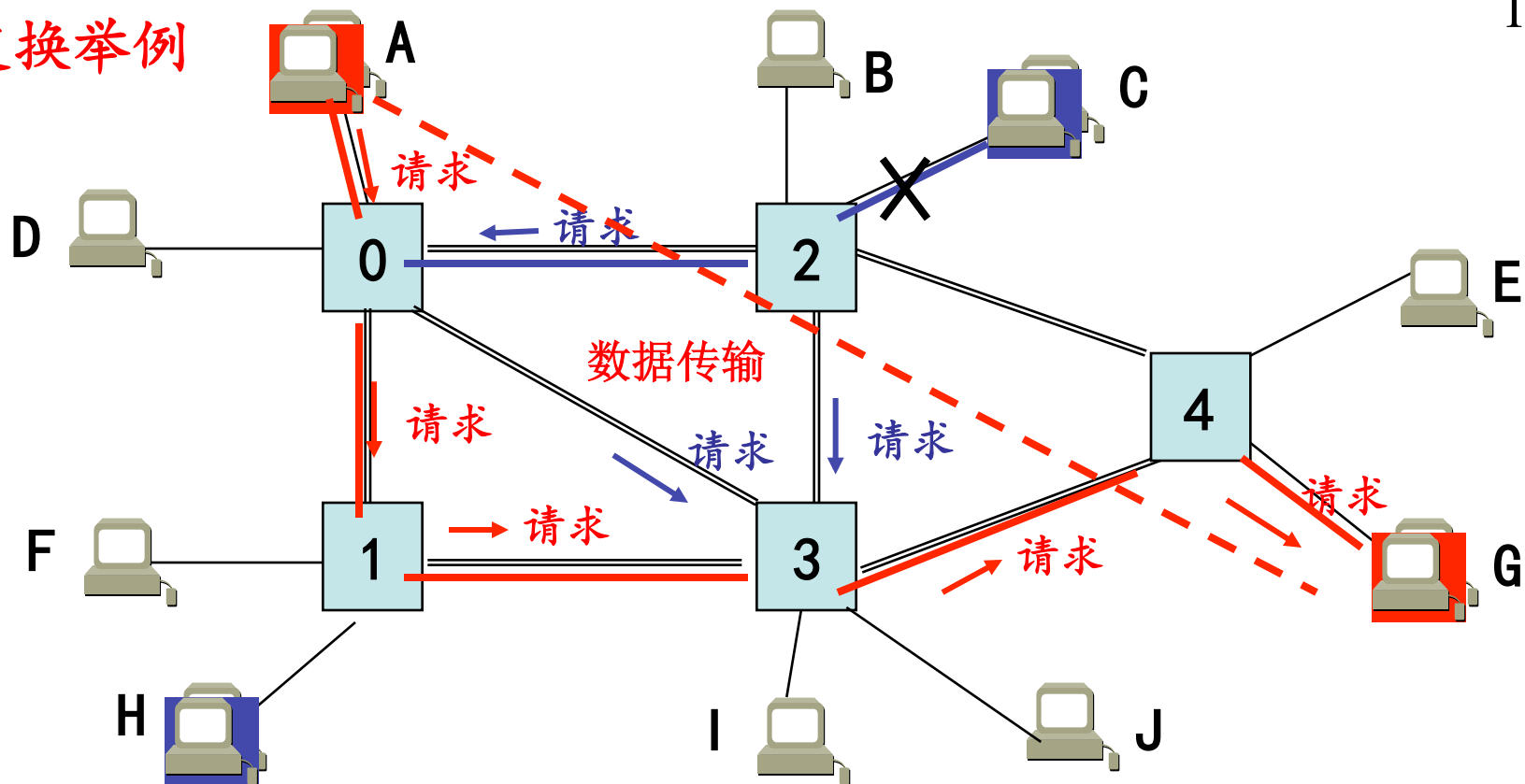
数据传输：物理线路建立后，站点之间进行数据传输。

释放线路：站点之间的数据传输完毕，执行释放线路的动作。

可以由任一站点发起，释放线路请求通过途径的中间结点送往对方，释放线路资源。

线路被释放之后，进入空闲状态，可由其它站点通信所用。

线路交换举例



线路交换的特点:

- ★ 独占性，线路的**利用率较低**，易于引起建链时的拥塞；
- ★ 实时性好，传输延迟小（近似线速, 使用整个线路资源）；
- ★ 线路交换不提供任何缓存装置，数据透明且按序传输；
- ★ 收发双方自动进行速率匹配（交换机设备比较简单）。

指导思想：利用结点的存储能力来提高线路利用率；

中间结点由具有存储能力的计算机承担；

用户信息（报文）附加目的地地址，并传递给中间结点；

中间结点**暂存报文**，根据地址确定输出端口，排队等待线路空闲时再转发给下一结点，直至终点。

特点：“存储—转发”。

★ 不独占线路，

★ 多个用户的数据可以通过**存储和排队**共享一条线路，

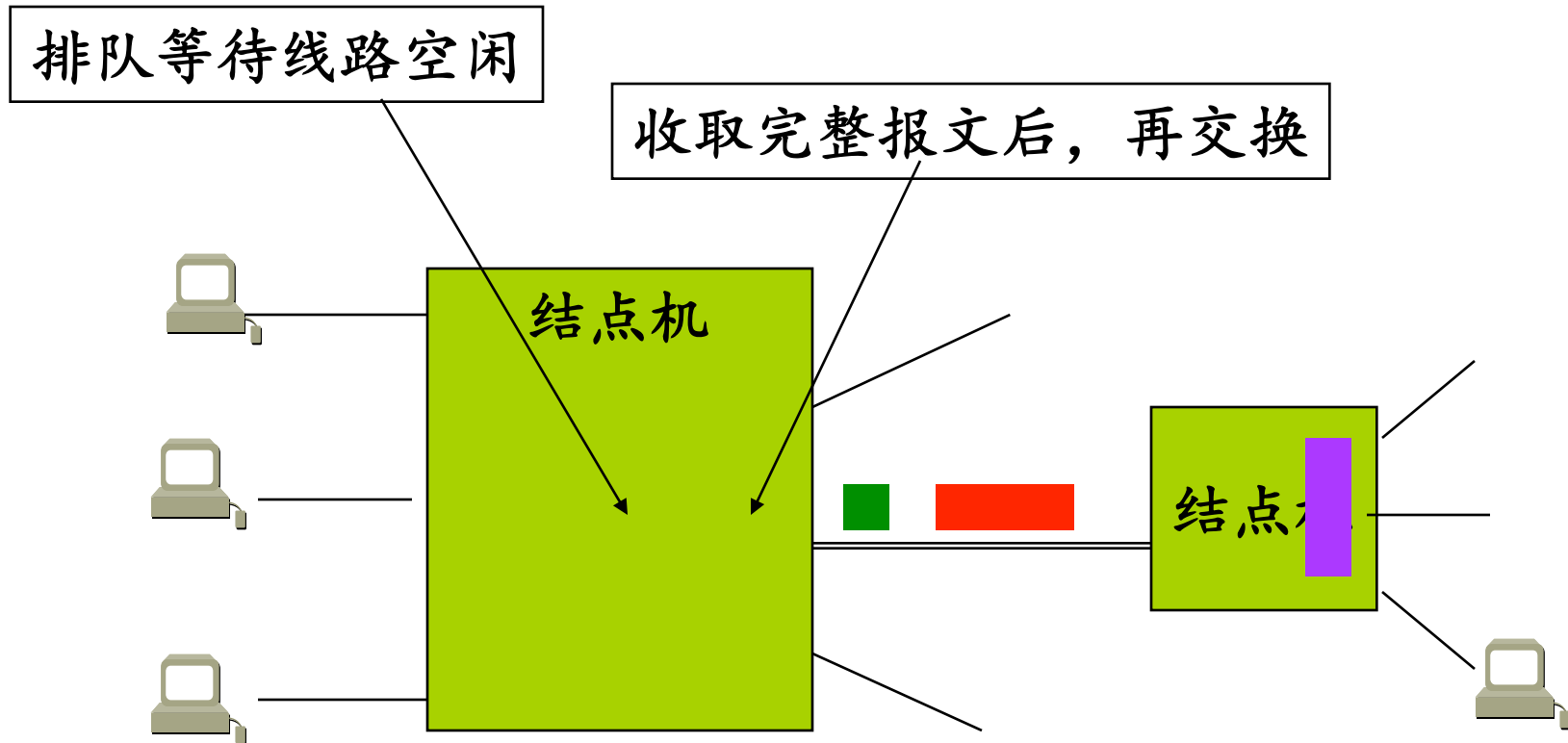
★ 无线路建立的过程，提高了线路的利用率；

★ 报文中增加地址字段，结点根据地址字段进行复制和转发；

★ 可以支持多点传输（一个报文传输给多个用户）；

★ 中间结点可进行数据格式的转换，方便接收站点的收取；增加差错检测功能，避免出错数据的无谓传输等。

报文传输示意图:



不足之处：

- 1 “**存储—转发**”可能浪费后续线路资源（等待完整报文的接收）；
- 2 报文长度未作规定，报文只能暂存在磁盘上，读取磁盘占用了额外的时间；
- 3 任何报文都必须排队等待（FIFO, 公平性不足）：不同长度的报文要求不同长度的处理和传输时间，即使非常短小的报文（如：交互式通信中的会话信息），也难以预测延迟；
- 4 报文交换难以支持实时通信和交互式通信要求。

(3) 存储交换—分组交换

结合线路交换和报文交换两者的优点，优化性能；

类似报文交换，只是它规定了交换设备处理和传输的数据长度（称之为分组），将长报文分成若干个小分组进行传输。不同站点的数据分组可以交织在同一线路上传输，提高了线路的利用率。

分组长度固定，中间结点可以采用高速缓存技术来暂存分组，提高了转发的速度。

分组长度有限，可以较早利用后继线路的资源。

分组交换实现的关键：分组长度的选择

分组越小，冗余量（分组中的控制信息等）在整个分组中所占的比例越大，影响用户数据传输的效率；

分组越大，数据传输出错的概率越大，增加重传的次数，影响用户数据传输的效率。

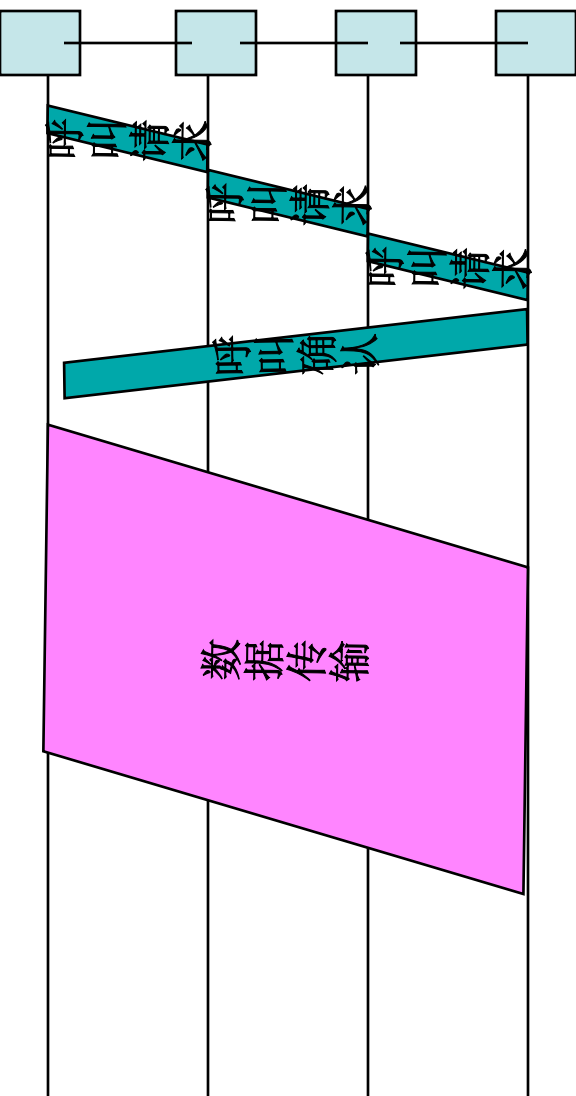
线路质量越好，可支持的分组长度越长。

X.25分组交换网： 分组长度定义为131字节，（包括128字节的用户数据和3字节的控制信息）；

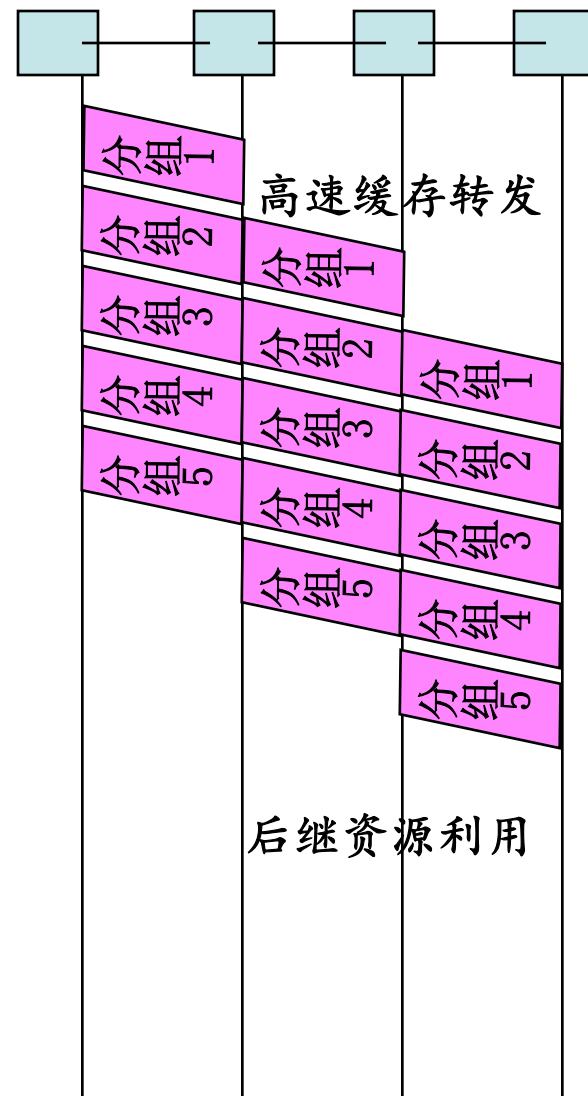
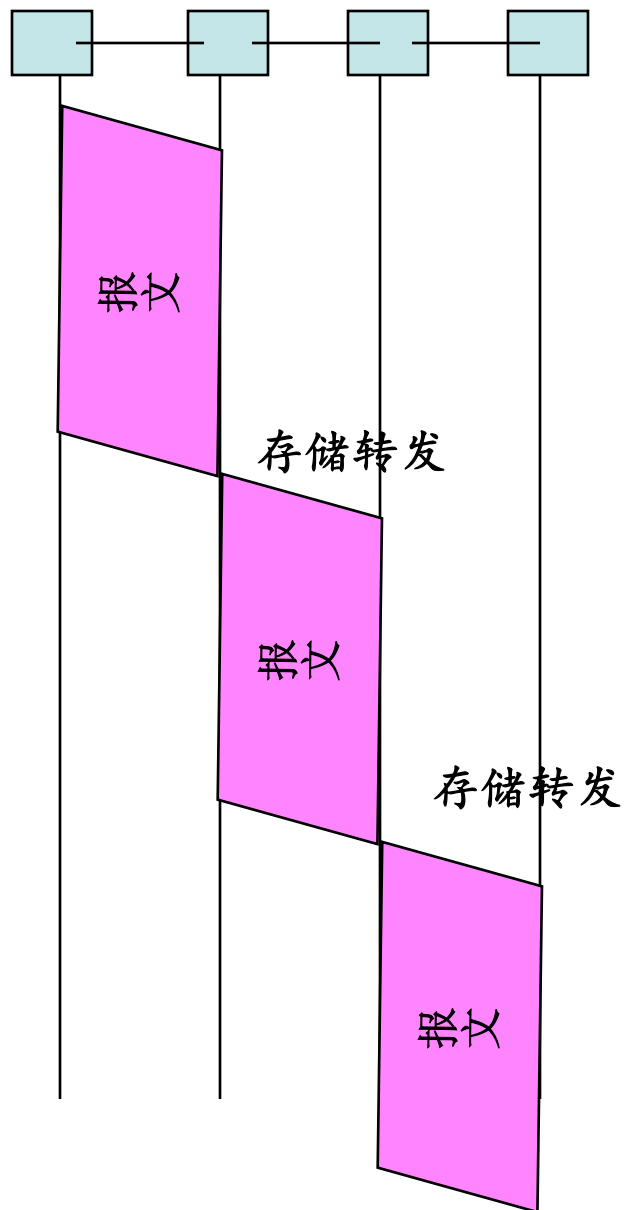
以太网： 分组长度定义为1500字节左右（较好的线路质量和较高的传输速率）；

FDDI网络： 约为4800字节（光纤）。

线路交换



报文交换



(4) 线路交换和分组交换的比较

1、分配通信资源（主要是线路）的方式：

线路交换：静态分配线路，线路资源浪费，接续困难；

分组交换：动态（按序）分配线路，提高线路利用率；使用缓存技术暂存分组；可能出现内存资源耗尽，而丢弃分组的现象。

2、用户的灵活性：

线路交换：信息传输透明，用户自行定义传输信息内容、速率、体积、格式等，因此可以同时传输语音、数据、图像等；

分组交换：半透明传输，按照分组设备的要求使用基本的参数。

3、数据传输实时性：

线路交换：接续难，传输快；

分组交换：基本满足要求。

4、资费情况：

线路交换网络：依赖通信的距离和使用的时间；

分组交换网络：传输的字节（或者分组）数和连接的时间。

2.12 数据报和虚电路

分组交换技术应用—用户分解报文为分组，网络应解决用户**分组的传输管理**问题：数据报和虚电路。

(1) 数据报—面向**无连接**的数据传输

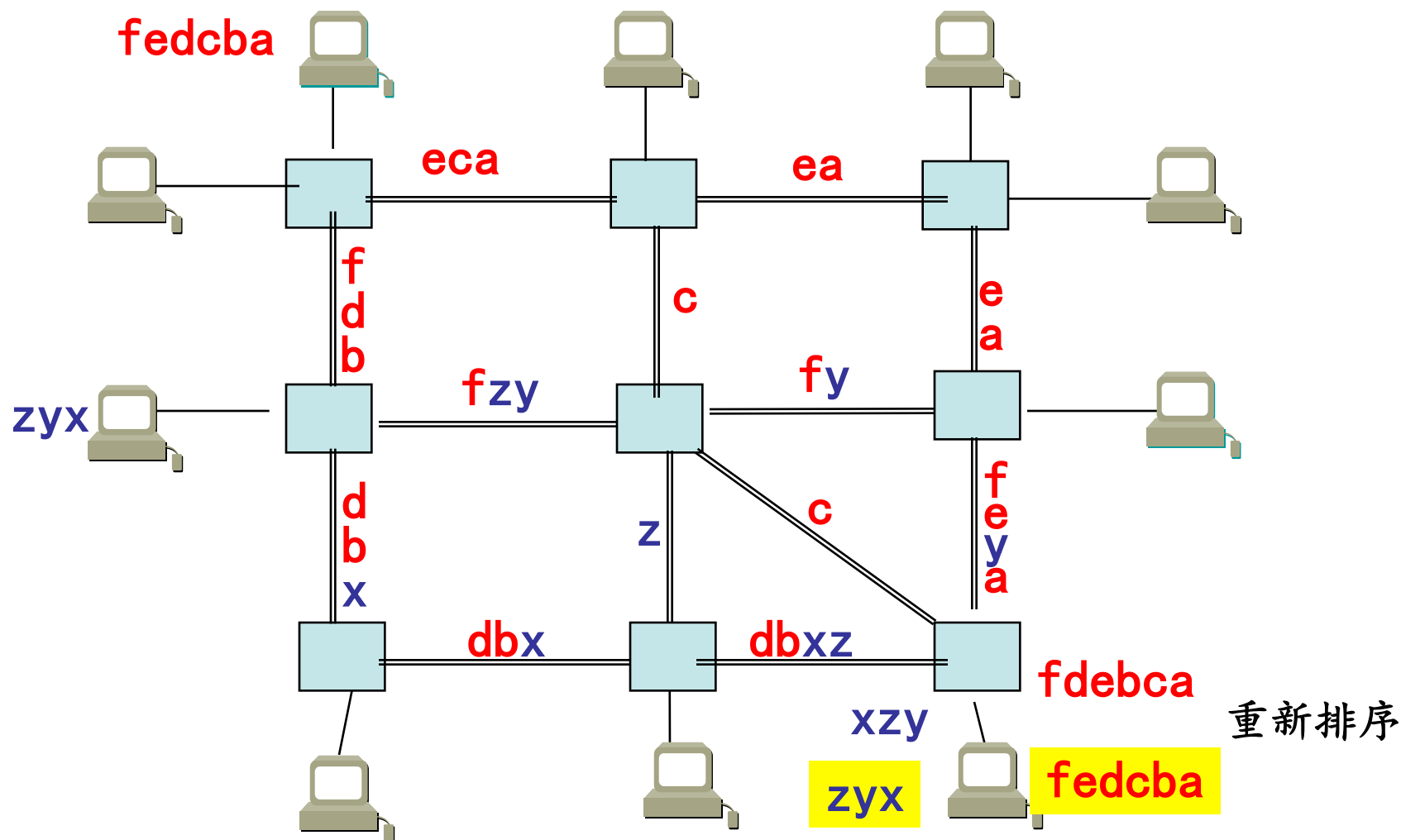
借鉴**报文交换**的思想。传输的分组称为数据报。

数据报的前部具有**地址信息字段**。网络中的结点根据地址信息和**路由**规则，独立的选择输出端口，暂存和排队数据报，并在传输媒体空闲时，发往相邻结点，直至最终站点。

当一对站点之间需要传输多个数据报时，由于每个**数据报均被独立地路由、排队和传输**，在网络中可能会走不同的路径，产生不同的时间延迟。**按序发送的多个数据报可能以不同的顺序达到终点**。站点必须具有**存储和重新排序**的能力。

数据报传输示意：

不同站点发出的分组汇聚到网络，分组**独立路由**导致同一站点发出的分组（**数据报**）选择**不同的路径**和不同的投递顺序。



(2) 虚电路——面向连接的数据传输

借鉴线路交换的思想，但电路是**虚拟**的。

采用多路复用技术，物理介质被理解为由多个子信道（**逻辑信道-LC**）组成，子信道的串接形成**虚电路**（VC），利用不同的虚电路来支持不同的用户数据的传输。

虚电路进行数据传输的过程：

➤ **虚电路建立**：发送方发送含有地址信息的特定的控制信息块（如：呼叫分组），该信息块途经的每个中间结点根据当前的逻辑信道（LC）使用状况，分配LC，并建立输入和输出LC映射表，所有中间结点分配的LC的**串接形成虚电路**（VC）。

虚电路进行数据传输的过程（续）：

➤ 数据传输：

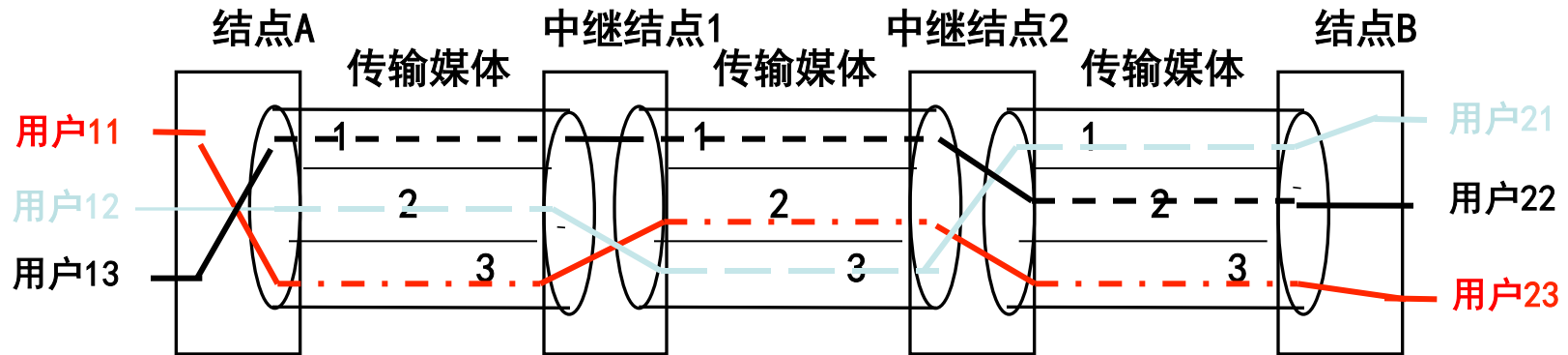
站点发送的所有分组均沿着相同的VC传输，分组的发收顺序完全相同；

分组中只带逻辑信道号，而不带地址。

➤ 虚电路释放：

数据传输完毕，采用特定的控制信息块（如：拆除分组），释放该虚电路。通信的双方都可发起释放虚电路的动作。

虚电路的形成



$$VC1 \text{ (用户11, 用户23)} = LC1 \text{ (3)} + LC2 \text{ (2)} + LC3 \text{ (3)}$$

$$VC2 \text{ (用户12, 用户21)} = LC1 \text{ (2)} + LC2 \text{ (3)} + LC3 \text{ (1)}$$

$$VC3 \text{ (用户13, 用户22)} = LC1 \text{ (1)} + LC2 \text{ (1)} + LC3 \text{ (2)}$$

虚电路的扩展：

由于虚电路的建立和释放需要占用一定的时间，因此虚电路方式不很适合站点之间具有频繁的、短小数据交换的场合；

永久虚电路PVC 和 呼叫虚电路SVC。

永久虚电路：在两个站点之间事先建立固定的链接，占用固定的逻辑信道，类似于存在一条专用电路，任何时候，站点之间都可以进行通信。

呼叫虚电路：根据需要，动态建立和释放虚电路。

(3) 数据报和虚电路的比较

★ 数据报:

- 传输无需连接建立和释放的过程;
- 每个数据报中需带地址信息（冗余信息），占用信道资源;
- 用户的连续数据块会无序地到达目的地，接收站点处理复杂。
- 当使用网状拓扑组建网络时，任一中间结点或者线路的故障不会影响数据报的传输（可以选择不同的路径），可靠性较高。
- 数据报较适合站点之间小批量数据的传输（缓存/排序占用资源）。

数据报和虚电路的比较（续）：

★ 虚电路：

- 传输需虚电路建立和释放的过程；
- 数据分组中仅含少量的地址信息（LC号），结点执行LC的映射工作，用户的连续数据块沿着相同的路径，按序到达目的地；接收站点处理方便。
- 如果虚电路中的某个结点或者线路出现故障，将导致虚电路传输失效。
- 虚电路方式较适合站点之间大块数据的传输（地址冗余量小）。

（第二章完）

- ★ 信源/信宿和信道类型匹配：调制/解调和编码/解码；
调制/解调：调幅/调频/调相/组合；
编码/解码：取样-量化-编码；
- ★ 信息在信道上的表示：通信编码
单极性、双极性，RS232码、不归0交替码、曼码、4b/5b码；
- ★ 信息在计算机内的显示：ASCII码等；
- ★ 收发双方的同步：同步/异步传输
只有实现位同步，才能完成块同步；
- ★ 差错处理—反馈重传法：
停-等协议、滑动窗口（多块传输、统一确认）。
常用的检错码：奇偶校验码、正反码、循环冗余码（CRC）；
具有差错处理能力的传输控制规程：
字符型传输控制规程（基于半双工的停-等协议）
比特型传输控制规程（基于全双工的滑动窗口协议）；

★ 差错处理—反馈重传法:

具有差错处理能力的传输控制规程:

字符型传输控制规程（基于半双工的停-等协议）

比特型传输控制规程（基于全双工的滑动窗口协议）；

注意控制字符（或者控制序列）在数据字段中出现时的歧义解决方法（填充转义）

★ 提高线路利用率的方法:

多路复用(频分/时分/波分)和集中传输;

★ 中间结点转发信息的方法:

电路交换、报文交换、分组交换;

★ 分组流传输管理方法:

面向无连接的数据报、面向连接的虚电路。

第二章作业

- 在因特网环境下，**HDLC**的地址字段是什么内容？
- 试述滑动窗口协议的工作原理。与等-停协议相比，滑动窗口协议有何特点？