通信原理

第六章 差错控制编码

目 录

6.1 概述

➢信道编码的发展历史

- 香农公式:

$$C = B\log_2\left(1+\frac{S}{n_0 B}\right) \qquad (bit/s)$$

Shannon建立了对信息通信的**基本限制，开创了一个新的领域--信息论**，随后的几十年中，**尤其是在编码领域，**人们开始向**逼近**Shannon极限进行了不懈的努力。

- Hamming, 1950，提出了第一个实用的差错控制编码方案

6.1 概述

- Elias, 1955, 提出卷积码。与分组码的不同在于：它充分利用了各个信息块之间的相关性。但译码复杂！
- Viterbi，1967，提出了Viterbi译码算法。使卷积码被广泛应用，如GSM、 IS-95 CDMA、3G、商业卫星通信等。离香农极限2~3dB

Viterbi, CDMA之父

6.1 概述

- Berrou, 1993, Turbo码和迭代译码。Turbo码编码器是由两个卷积码并行级联构成，接收端在两个分量译码器之间进行迭代译码。离香农限0.01dB。

6.1 概述

- Fony, 1966, 级联码
- 1968, BCH码
- Viterbi等人， 卷积码及译码
- Ungerboeck, Fony, Wei, 1982~1987, 网格编码调制TCM
- Hamming,1950， 纠错和纠错编码的经典研究
- Muller, Reed, Solomen, 1960, 新的分组码。

6.1 概述

➢内容提要

本章介绍差错控制编码的基本概念。首先介绍差错控制的基本方式以及几种常见的检错码，然后介绍纠错码的基本原理。重点掌握纠错原理，为后续课程打下基础。

6.1 概述

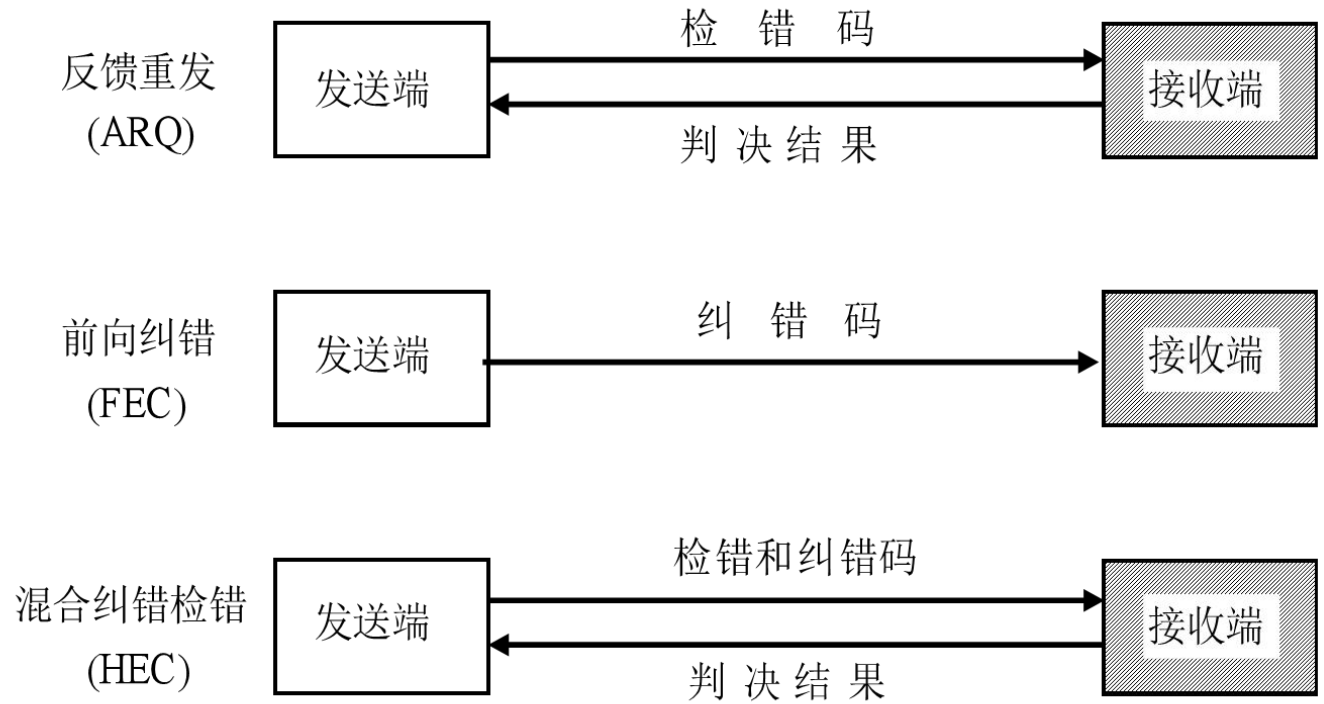
➢6.1.1差错控制的基本方式

◆ **反馈重发(ARQ)：** 用于**检测**的纠错码在译码器输出端只给出当前码字传输是否可能出错的指示，当有错时按某种协议通过一个反向信道请求发送端重传已发送的码字全部或部分。特点是需要反馈信道，译码设备简单，对突发错误很有效。但在误码率高的信道中，由于重发多，导致传输效率低，**实时性差**。

◆ **前向纠错(FEC)：** 发送端发送能够**纠正**错误的码，通过信道传输到接收端，接收端译码器根据相应的译码算法纠正传输中的出现的错误。其特点是只需要单向传输，**实时性好**，传输效率高，但译码设备较复杂。

6.1 概述

- **混合纠错检错(HEC)：** 发端发送同时具有自动**纠错和检测**能力的码组，接收端收到码字后，译码器首先检验错误情况，如果在码的纠错能力以内，则自动纠错；如果超过码的纠错能力，接收端通过反馈信道命令发送端重发来纠正错误。



➢6.1.2几种简单的检错码

□ 奇偶校验码：

- **编码规则：**在每一个原信息码字后增加一位校验位，使得码字中“1”的个数为偶数/奇数。

只有一位校验位

$$c_{n-1} \oplus c_{n-2} \oplus \cdots \oplus c_2 \oplus c_1 \oplus c_0 = 0$$ 偶校验码

$$c_{n-1} \oplus c_{n-2} \oplus \cdots \oplus c_2 \oplus c_1 \oplus c_0 = 1$$ 奇校验码

- **纠检能力：**只能检测奇数个错误，不具备纠错能力

二维奇偶校验码

- **编码规则**：

按行
按列
实施
奇偶
校验

$$
\begin{array}{cccc|c}
c_{n-1}^{m-1} & c_{n-2}^{m-1} & \cdots & c_1^{m-1} & c_0^{m-1} \\
c_{n-1}^{m-2} & c_{n-2}^{m-2} & \cdots & c_1^{m-2} & c_0^{m-2} \\
\vdots & \vdots & \cdots & \vdots & \vdots \\
c_{n-1}^{1} & c_{n-2}^{1} & \cdots & c_1^{1} & c_0^{1} \\
\hline
c_{n-1}^{0} & c_{n-2}^{0} & \cdots & c_1^{0} & c_0^{0}
\end{array}
$$

行校验码

列校验码

- **纠检能力**：较强，并具有一定的检错能力，但无法检测任意构成方阵的 4 个错误
- **适用**：适用检测长度不大于行数（或列数）的突发错误

6.1 概述

恒比码:

- **编码规则**：码字中 1 的数目与 0 的数目保持恒定比例
- **检测方法**：只要计算接收码元中 1 的个数是否与规定的相同，就可判断有无错误
- **适用**：用于电报传输系统或者其他键盘产生的字母和符号

例

目前国际上通用的 ARQ 电报通信系统中，采用 3:4 恒比码，即“7 中取 3”恒比码，每个码字中有3个“1”，因而共有 35 个码字，用于代表 26 个英文字母及其它符号上

6.1 概述

□ 循环冗余校验(CRC)码

- **编码规则：**根据网络数据包或计算机文件等数据产生简短固定位数校验码，它是利用除法及余数的原理来侦测错误的
- **适用：**主要用来检测或校验数据传输或者保存后可能出现的错误

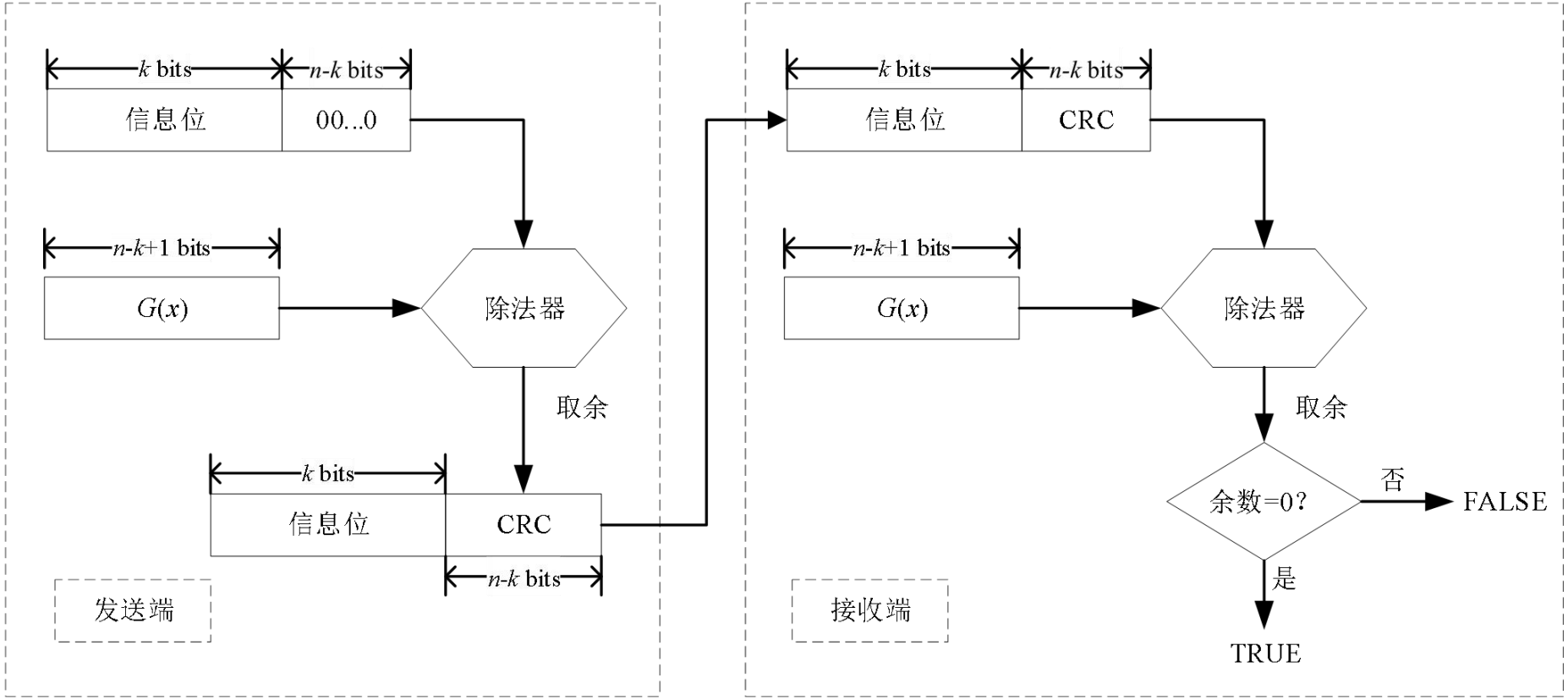
常用 CRC 生成多项式

名称	生成多项式	应用举例
CRC-4	x^4+x+1	ITU、G707
CRC-8	x^8+x^2+x+x	ATM header
CRC-10	$x^{10}+x^9+x^5+x^4+x+1$	ATM AAL
CRC-16	$x^{16}+x^{12}+x^5+1$	Bluetooth、Zigbee
CRC-32	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$	ZIP、RAR、LANs

6.1 概述

□ 循环冗余校验(CRC)码

发送端计算出CRC后随数据一同发送给接收端，接收端接收到后同样进行**除法运算**，得到结果为 0 则证明没有出错，否则说明出现错误



CRC校验过程

6.1 概述

例：待发送的信息为101001，生成多项式为$G(x) = x^3 + x^2 + 1$，计算余数，并求发送码字。

解：

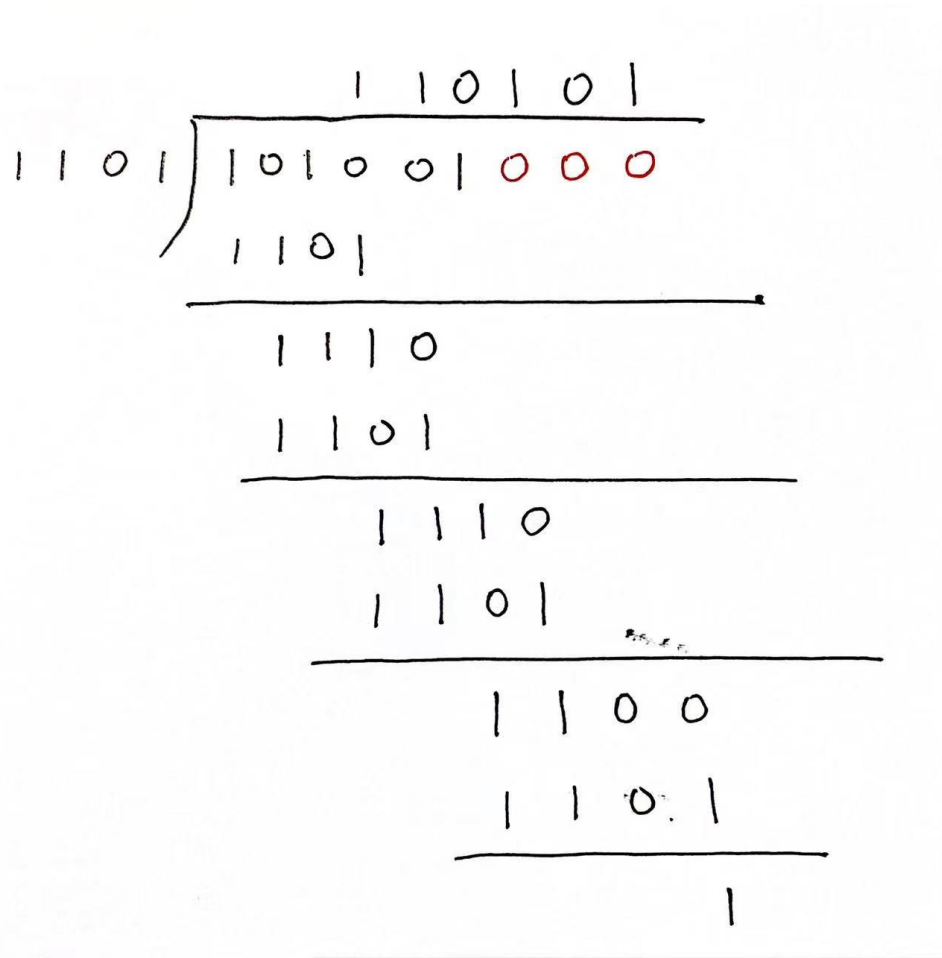
首先由信息位补0得被除数：多项式最高次数为3，所以补3个0。故被除数为101001000。

然后求除数：由生成多项式得到，除数为1101.

然后进行除法，作异或运算，求余数

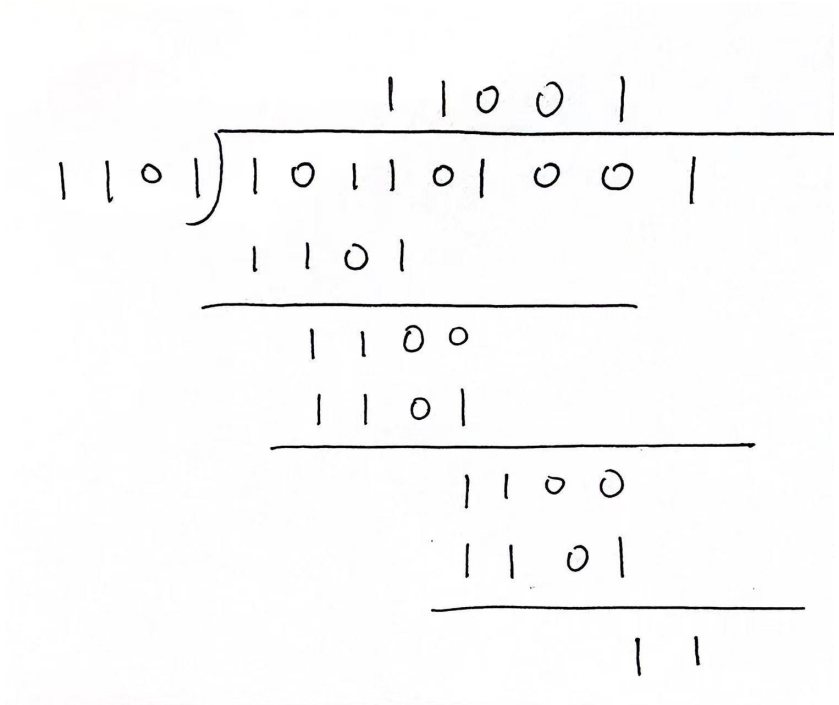
除完余数为1，补齐得3位余数001

故余数为001，发送码字为101001001



例：接收到的信息为101101001，生成多项式为$G(x) = x^3 + x^2 + 1$，判断传输是否误码。

解：作除法：

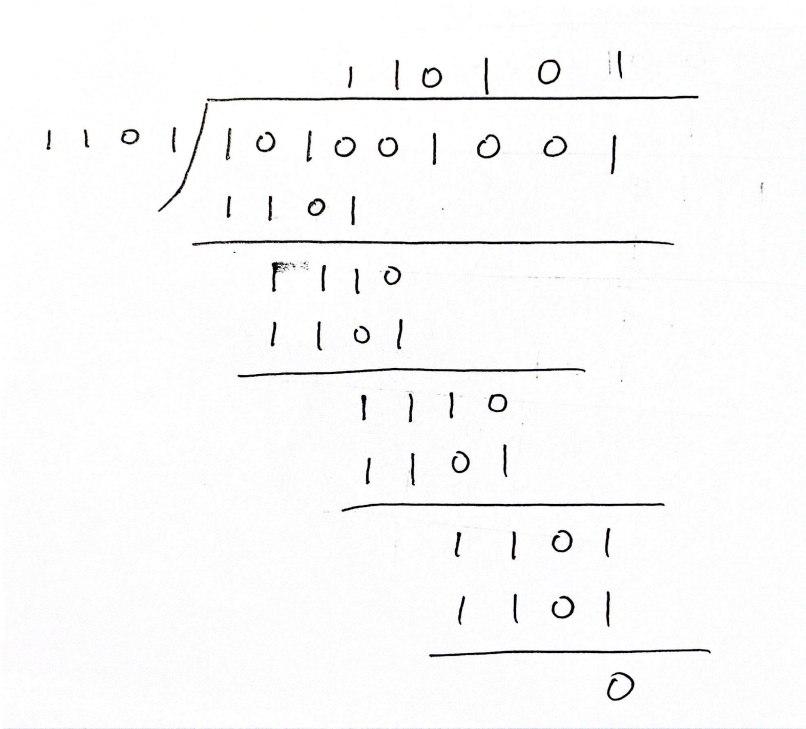


可见，余数为11，不为0，故为误码

6.1 概述

例：接收到的信息为101001001，生成多项式为$G(x) = x^3 + x^2 + 1$，判断传输是否误码。

解：作除法：



可见，余数为0，故不是误码

➢6.1.3纠错码的基本原理

□ 差错编码中的名词解释:

码长：编码码字的码元总位数称为码字的长度，简称码长。

码重：码字中，非零码元的数目称为码字的重量，简称码重。

码距：两个等长码字之间对应位不同的数目称为这两个码字的距离，简称码距。

最小码距：在全体码字集合中，码字距离的最小值称为最小码距。

分组码一般用（n,k）表示，其中 n 是编码码字的码长，k 是码字中信息码元的位数。$n-k=r$是码字中校验码元的数。如果用二进制码元表示码字，共有 2^k 个不同的信息组，对应有 2^k个不同的编码码字，称为**许用码字**。其余 2^n-2^k 个未被选用的码字称为**禁用码字**

□最小距离与检、纠错能力

纠错误的抗干扰能力完全取决于许用码字之间的码距，码距的最小值越大，说明码字间的最小差别越大，抗干扰能力就越强。

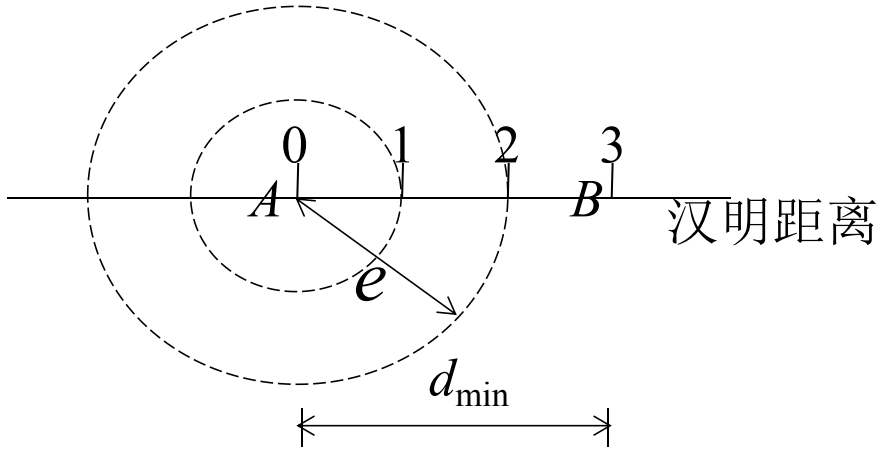
检错能力：如果一个线性码能检出长度≤ e 个码元的任何错误图样，称码的检错能力为 e 。

纠错能力：如果线性码能纠正长度≤ t 个码元的任意错误图样，称码的纠错能力为 t 。

6.1 概述

分组码的最小码距 d_{min} 与检错和纠错能力之间满足下列关系：

(1)为检测e个错码，要求最小码距 $d_{\min} \geq e+1$



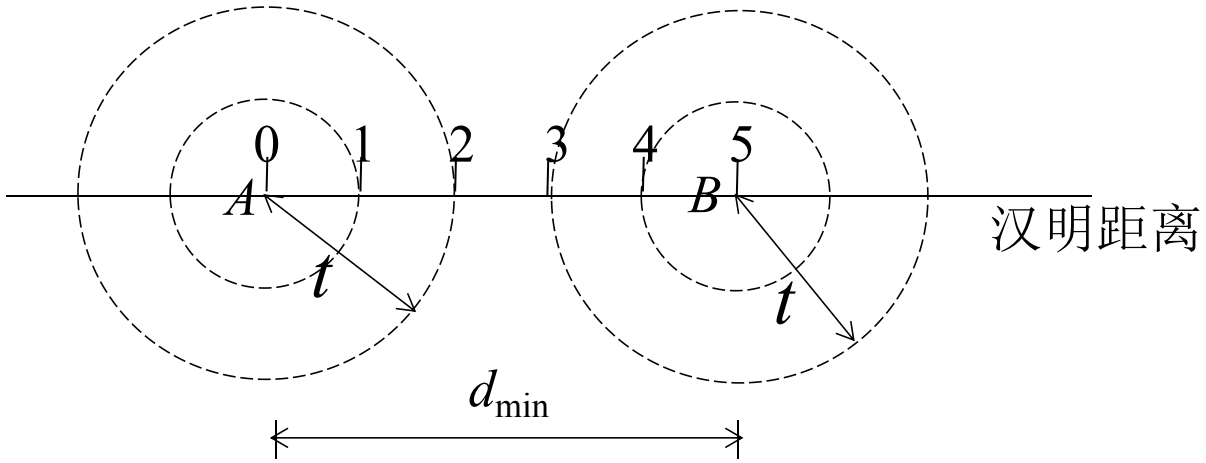
【证】设码组A位于O点，$d_{\min}=3$，

若A错1位，将位于以O点为圆心，以1为半径的圆。

若A错2位，将位于以O点为圆心，以2为半径的圆。

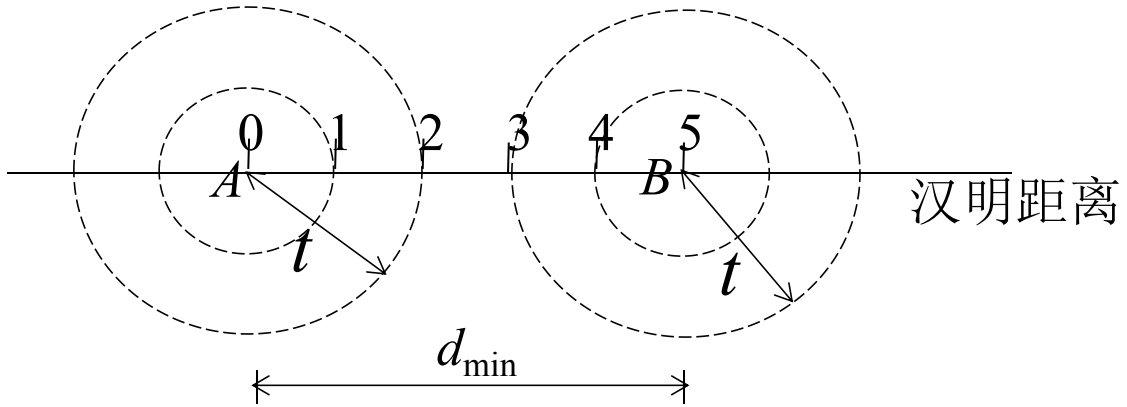
若A错3位，将位于以O点为圆心，以3为半径的圆。 可能错成B，无法检错。

(2)为了纠正t个错码，要求最小码距$d_{min} \geq 2t+1$



【例】设码组A和B的距离为5。码组A或B若发生不多于2位错码，则其位置均不会超出半径为2以原位置为圆心的圆。这两个圆是不重叠的。判决规则为：若接收码组落于以A为圆心的圆上就判为A，若落于以B为圆心的圆上就判决为码组B。这样，就能够纠正2位错码。

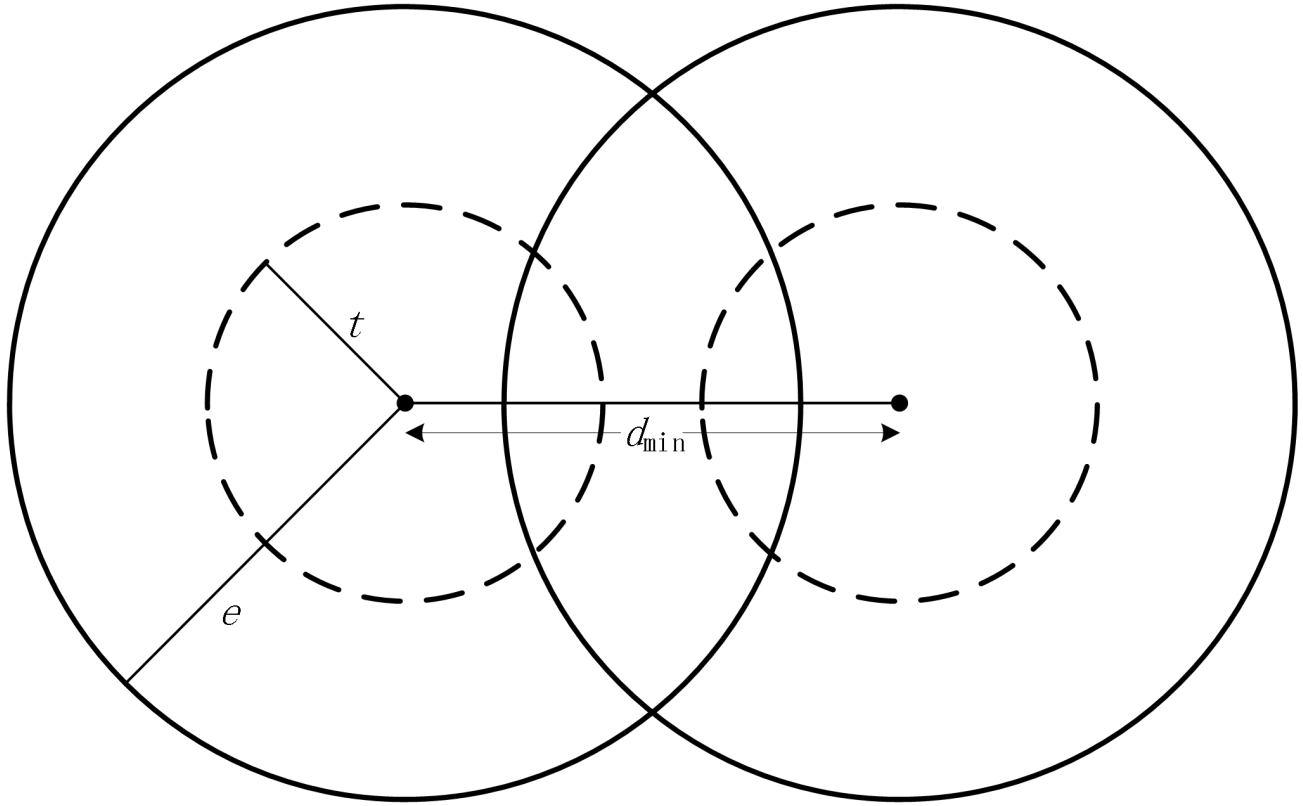
(3)为纠正t个错码，同时检测e个错码，要求最小码距$d_{min} \geqslant e+t+1$



【证】设d_0 =5，则能检4个错码，或纠2个错码。但是，不能同时作到两者。

例：发送A，若错3位，收端认为是B错2位，从而错“纠”为B。

6.1 概述



实线圆为检错范围，可交叉；虚线为纠错范围，不可交叉

6.2 线性分组码

➢ 线性分组码的基本概念

线性码：按照一组线性方程构成的代数码。即每个码字的校验码元是信息码元的线性组合。

代数码：建立在代数基础上的编码。

分组码：每一码组的校验码元仅与本组中的信息码元有关。

线性分组码：按照一组线性方程构成的分组码。

一个 k 比特信息的分组码可以映射到一个码长为 n 的码字上。当校验码元与信息码元之间为线性关系时，则称为线性分组码。

6.2 线性分组码

➢ 线性分组码的基本概念

□ 汉明重量/最小重量

设线性分组码码字为 $\mathbf{c} = (c_{n-1}\ c_{n-2} \ldots c_{n-k}\ c_{n-k-1} \ldots c_0)$，高 k 位为信息位，低 $n-k$ 位为校验位。C 为 $\mathbf{c}$ 所有可能码字的集合。$w(c)$ 为码字 $\mathbf{c}$ 的汉明重量。$W(C)$为集合 C 的汉明重量，表示 $\mathbf{c}$ 所有可能码字的重量。显然线性分组码里有且仅有一个全零码字，因此记 $w_{min}(C)$ 为 C 中非零码字的**最小重量**：

$$w_{\min}(C) = \min\left\{w(\mathbf{c}) : \mathbf{c} \in C, \mathbf{c} \neq 0\right\}$$

令 $\mathbf{v}$ 和 $\mathbf{w}$ 是集合 C 中两个不同的 n 维向量，记 $\mathbf{v}$ 和 $\mathbf{w}$ 之间汉明距离为 $d(\mathbf{v}, \mathbf{w})$。对于线性码，由于其在二元域上的封闭性，有 $\mathbf{v} - \mathbf{w} = \mathbf{c}$，（$\mathbf{c} \in C$，$\mathbf{c} \neq 0$），因此集合 C 中两个不同码字的**汉明距离等于集合中某一非零码字的汉明重量**，即：

$$d(\mathbf{v}, \mathbf{w}) = w(\mathbf{c})$$

6.2 线性分组码

➢ 生成矩阵和校验矩阵

□ 生成矩阵

一个二进制 (n, k) 线性分组码 C 是所有二进制 n 维向量组成的向量空间 $\boldsymbol{V}$的一个 k 维子空间，因此 C 中存在 k 个线性无关的码字 $\boldsymbol{g}_{k-1}$，$\boldsymbol{g}_{k-2}$，…，$\boldsymbol{g}_0$，使得 C 中每个码字 $\boldsymbol{c}$ 都是这 k 个码字的线性组合，即

$$\mathbf{c} = u_{k-1}\mathbf{g}_{k-1} + u_{k-2}\mathbf{g}_{k-2} + \cdots + u_0\mathbf{g}_0$$

可以把 $\boldsymbol{g}_0$，$\boldsymbol{g}_1$，…，$\boldsymbol{g}_{k-1}$表示为一个 $k \times n$ 矩阵，即

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_{k-1} \\ \mathbf{g}_{k-2} \\ \vdots \\ \mathbf{g}_0 \end{bmatrix} = \begin{bmatrix} g_{k-1,n-1} & g_{k-1,n-2} & \cdots & g_{k-1,0} \\ g_{k-2,n-1} & g_{k-2,n-2} & \cdots & g_{k-2,0} \\ \vdots & \vdots & \ddots & \vdots \\ g_{0,n-1} & g_{0,n-2} & \cdots & g_{0,0} \end{bmatrix}$$

6.2 线性分组码

令 $\mathbf{u}=(u_{k-1}, u_{k-2}, \cdots, u_0)$ 是待编码的消息，其对应编码后码字 $\mathbf{c}=(c_{n-1}\ c_{n-2}\ \cdots\ c_{n-k}\ c_{n-k-1}\ \cdots\ c_0)$ 与 $\mathbf{G}$ 的关系可如下给出：

$$\mathbf{c}=\mathbf{u}\cdot\mathbf{G}=(u_{k-1},u_{k-2},\cdots,u_0)\cdot\begin{bmatrix}\mathbf{g}_{k-1}\\ \mathbf{g}_{k-2}\\ \vdots\\ \mathbf{g}_0\end{bmatrix}=u_{k-1}\mathbf{g}_{k-1}+u_{k-2}\mathbf{g}_{k-2}+\cdots+u_0\mathbf{g}_0$$

因此，码字 $\mathbf{c}$ 是矩阵 $\mathbf{G}$ 的行向量关于 $\mathbf{u}$ 的线性组合，称 $\mathbf{G}$ 为 (n, k) 线性分组码 C 的**生成矩阵。**

6.2 线性分组码

生成矩阵 **G** 可以通过线性变换转化成如下系统形式：

$$\mathbf{G}=[\mathbf{I}_k \mid \mathbf{P}]=\begin{bmatrix} 1 & 0 & \cdots & 0 & p_{k-1,n-k-1} & p_{k-1,n-k-2} & \cdots & p_{k-1,0} \\ 0 & 1 & \cdots & 0 & p_{k-2,n-k-1} & p_{k-2,n-k-2} & \cdots & p_{k-2,0} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & p_{0,n-k-1} & p_{0,n-k-2} & \cdots & p_{0,0} \end{bmatrix}$$

$\mathbf{I}_k$为一个$k\times k$的单位矩阵

$\mathbf{P}$为一个$k\times(n\text{-}k)$的矩阵，$p_{i,j}=0$或1

具有以上形式的生成矩阵称为典型生成矩阵

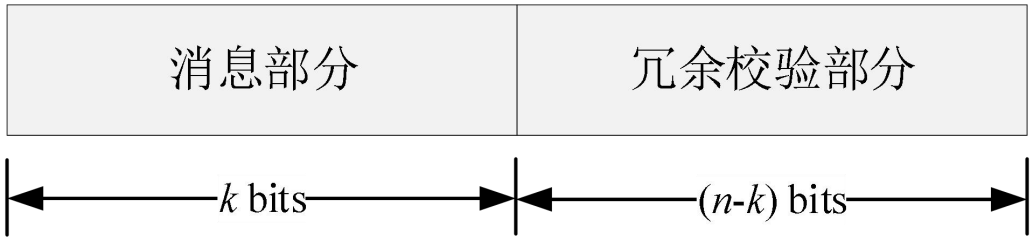
G 矩阵的性质：

① **G** 矩阵的各行是线性无关的；

② **G** 的各行本身就是一个码组。

6.2 线性分组码

线性系统分组码：用标准生成矩阵 **G** 编码得到的码字，前面 k 位为信息部分，后面 $r=n-k$ 位为冗余校验部分，这种**消息部分在前冗余校验部分在后**的线性分组码称为线性系统分组码。



高 k 个比特与信息比特一致，即

$$c_{n-k+i}=u_i,\quad i=0,1,\cdots,k-1$$

低 n-k 个比特是信息比特的线性组合

$$c_j=u_{k-1}p_{k-1,j}+u_{k-2}p_{k-2,j}+\cdots+u_0p_{0,j},\quad j=0,1,\cdots,n-k-1$$

这 n-k 个由信息比特线性求和得到的码比特称为**一致校验比特（简称校验比特）**

6.2 线性分组码

□ 校验矩阵

对任何一个由 k 个线性独立的行向量组成的 $k\times n$ 矩阵 $\mathbf{G}$，均存在一个由 n-k 个线性独立的行向量组成的$(n\text{-}k)\times n$ 矩阵 $\mathbf{H}$，使得 $\mathbf{G}$ 与 $\mathbf{H}$ 正交，即矩阵 $\mathbf{G}$ 与 $\mathbf{H}$ 的关系为:

$$\mathbf{G}\mathbf{H}^{\mathrm{T}}=0$$

当且仅当 $\mathbf{c}\mathbf{H}^{\mathrm{T}}=0$（$n$-$k$ 维全零向量)时，二进制 n 维向量 $\mathbf{c}\in V$ 是 C 中的码字，C 被称为 $\mathbf{H}$ 的零空间，$\mathbf{H}$ 被称为 C 的校验矩阵。

一个 (n, k) 线性分组码 C 不仅可以通过生成矩阵 $\mathbf{G}$ 确定，还可以由其校验矩阵 $\mathbf{H}$ 定义。

6.2 线性分组码

线性系统码的校验矩阵 **H** 具有如下形式：

$$\mathbf{H}=\left[\mathbf{P}^{\mathrm{T}}\,|\,\mathbf{I}_{n-k}\right]=\left[\begin{array}{cccc|cccc} p_{k-1,n-k-1} & p_{k-2,n-k-1} & \cdots & p_{0,n-k-1} & 1 & 0 & \cdots & 0 \\ p_{k-1,n-k-2} & p_{k-2,n-k-2} & \cdots & p_{0,n-k-2} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{k-1,0} & p_{k-2,0} & \cdots & p_{0,0} & 0 & 0 & \cdots & 1 \end{array}\right]$$

具有以上形式的校验矩阵称为典型校验矩阵

H 矩阵的性质：

① **H** 的行数等于校验位的个数(n-k)。**H** 的每行“1”的位置表示相应码元之间存在的校验关系；

② **H** 的各行应该是线性无关的，否则得不到(n-k)个线性无关的校验关系式。

6.2 线性分组码

例：设线性分组码码字为 $\mathbf{c}=[c_6\ c_5\ c_4\ c_3\ c_2\ c_1\ c_0]$，其中高 4 位 $c_6\ c_5\ c_4\ c_3$ 是信息码元，低 3 位 $c_2\ c_1\ c_0$ 为校验码元，是由信息码元模 2 加得到，可以用下列线性方程组来描述：

$$\begin{cases} c_2 = c_6 + c_5 + c_4 \\ c_1 = c_6 + c_5 + c_3 \\ c_0 = c_6 + c_4 + c_3 \end{cases}$$

用生成矩阵 $\mathbf{G}$ 表示该线性方程组

$$\mathbf{c}=[c_6\ c_5\ c_4\ c_3]\cdot\mathbf{G}=[c_6\ c_5\ c_4\ c_3]\cdot[\mathbf{I}_k\mid\mathbf{P}]=\left[\begin{array}{cccc|ccc} 1&0&0&0&1&1&1\\ 0&1&0&0&1&1&0\\ 0&0&1&0&1&0&1\\ 0&0&0&1&0&1&1 \end{array}\right]$$

6.2 线性分组码

$$\begin{cases} c_2 = c_6 + c_5 + c_4 \\ c_1 = c_6 + c_5 + c_3 \\ c_0 = c_6 + c_4 + c_3 \end{cases} \longrightarrow \begin{cases} 1\cdot c_6 + 1\cdot c_5 + 1\cdot c_4 + 0\cdot c_3 + 1\cdot c_2 + 0\cdot c_1 + 0\cdot c_0 = 0 \\ 1\cdot c_6 + 1\cdot c_5 + 0\cdot c_4 + 1\cdot c_3 + 0\cdot c_2 + 1\cdot c_1 + 0\cdot c_0 = 0 \\ 1\cdot c_6 + 0\cdot c_5 + 1\cdot c_4 + 1\cdot c_3 + 0\cdot c_2 + 0\cdot c_1 + 1\cdot c_0 = 0 \end{cases}$$

$$[c_6\ c_5\ c_4\ c_3\ c_2\ c_1\ c_0]^{\mathrm{T}} \cdot \boldsymbol{H} = [0\ \ 0\ \ 0]$$

$$\boldsymbol{H} = \left[\boldsymbol{P}^{\mathrm{T}} \mid \boldsymbol{I}_{n-k}\right] = \left[\begin{array}{cccc|ccc} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$

6.2 线性分组码

根据上式可以得到（7,4）线性分组码的全部码字，如下表所示：

序号	码 字		序号	码 字	
	信息码元	校验码元		信息码元	校验码元
0	0 0 0 1	0 0 0	8	1 0 0 0	1 1 1
1	0 0 0 1	0 1 1	9	1 0 0 1	1 0 0
2	0 0 1 0	1 0 1	10	1 0 1 0	0 1 0
3	0 0 1 1	1 1 0	11	1 0 1 1	0 0 1
4	0 1 0 0	1 1 0	12	1 1 0 0	0 0 1
5	0 1 0 1	1 0 1	13	1 1 0 1	0 1 0
6	0 1 1 0	0 1 1	14	1 1 1 0	1 0 0
7	0 1 1 1	0 0 0	15	1 1 1 1	1 1 1

最小码距 d_{min}=3，它能纠正一个错误或检测两个错误

线性分组码主要性质：

- ✓ 任意两许用码字之和仍为一许用码字，也就是说，线性分组码具有封闭性；
- ✓ 分组码的最小码距等于非零码字的最小码重。

6.2 线性分组码

汉明码

- 一种可以纠正单个随机错误的线性分组码，其特点是：无论码长 n 为多少，汉明码最小码距 $d_0=3$；码长 n 与校验码元个数 r 满足关系式 $n = 2^r - 1$，且 $r \geq 2$。

极化(Polar)码

- 在编码时通过信道极化处理，使各个子信道呈现出不同的可靠性，当码长持续增加时，部分信道将趋向于容量接近于 1 的完美信道，另一部分信道趋向于容量接近于 0 的纯噪信道，选择在容量接近于 1 的信道上直接传输信息以逼近信道容量。

低密度奇偶校验(LDPC)码

- 一种具有稀疏校验矩阵的线性分组码，几乎适用于所有的信道，它的性能逼近香农极限，且描述和实现简单，译码过程可进行并行操作，因此适合于硬件实现。

6.2 线性分组码

- 汉明码：能够纠正1位错码且编码效率较高的一种线性分组码
  - 汉明码的构造原理
    - 在偶数监督码中，加一位监督位a_0，它满足：

$$a_{n-1} \oplus a_{n-2} \oplus \cdots \oplus a_0 = 0$$

接收端计算：$S = a_{n-1} \oplus a_{n-2} \oplus \cdots \oplus a_0$

若$S=0$，就认为无错码；若$S=1$，就认为有错码。上式称为监督关系式，S称为校正子。由于校正子S只有两种取值，故它只能代表有错和无错这两种信息，而不能指出错码的位置。

6.2 线性分组码

- 监督位增加一位，则增加一个监督关系式。两个校正子（S1,S2）有4种组合，能表示4种信息。1种表示无错，其余3种可用来指示一个错码的3种不同位置。同理，*r*个监督关系式能指示1位错码的(2^r-1)个可能位置。
- 设码长为*n*，信息位数为*k*，则监督位数*r*＝*n*－*k*。如果希望用r个监督位构造出*r*个监督关系式来指示1位错码的*n*种可能位置，则要求

$$2^r-1\geq n \qquad 或 \qquad 2^r\geq k+r+1$$

6.2 线性分组码

【例】：设汉明码(n, k)中$k = 4$，为纠正1位错码，要求监督位数$r \geq 3$。若取$r = 3$，则$n = k + r = 7$。用$a_6\ a_5 \ldots a_0$表示这7个码元。现规定校正子S_1、S_2和S_3的值与错码位置的对应关系如下表，求所有码字、监督矩阵和生成矩阵。

$S_1\ S_2\ S_3$	错码位置	$S_1\ S_2\ S_3$	错码位置
001	a_0	101	a_4
010	a_1	110	a_5
100	a_2	111	a_6
011	a_3	000	无错码

【解】由表中可见，仅当一位错码的位置在a_2、a_4、a_5或a_6时，校正子S_1为1，否则S_1为零。这就意味着a_2、a_4、a_5和a_6四个码元构成偶数监督关系：

$$S_1 = a_6 \oplus a_5 \oplus a_4 \oplus a_2$$

同理：

$$S_2 = a_6 \oplus a_5 \oplus a_3 \oplus a_1$$

$$S_3 = a_6 \oplus a_4 \oplus a_3 \oplus a_0$$

设a_6、a_5、a_4、a_3为信息码，a_2、a_1、a_0为监督码，编码时监督码应使校正子S_1、S_2、S_3为零，即：

$$\begin{cases} a_6 \oplus a_5 \oplus a_4 \oplus a_2 = 0 \\ a_6 \oplus a_5 \oplus a_3 \oplus a_1 = 0 \\ a_6 \oplus a_4 \oplus a_3 \oplus a_0 = 0 \end{cases}$$

上式移项解出监督位：

$$\begin{cases} a_2 = a_6 \oplus a_5 \oplus a_4 \\ a_1 = a_6 \oplus a_5 \oplus a_3 \\ a_0 = a_6 \oplus a_4 \oplus a_3 \end{cases}$$

6.2 线性分组码

给定信息位后，可直接按上式算出监督位，从而得到所有码字，见下表：

信息位 $a_6a_5a_4a_3$	监督位 $a_2a_1a_0$	信息位 $a_6a_5a_4a_3$	监督位 $a_2a_1a_0$
0000	000	1000	111
0001	011	1001	100
0010	101	1010	010
0011	110	1011	001
0100	110	1100	001
0101	101	1101	010
0110	011	1110	100
0111	000	1111	111

可见：

$d_{min} = 3$，

可纠正1位错码。

6.2 线性分组码

将:

$$\begin{cases} a_6 \oplus a_5 \oplus a_4 \oplus a_2 = 0 \\ a_6 \oplus a_5 \oplus a_3 \oplus a_1 = 0 \\ a_6 \oplus a_4 \oplus a_3 \oplus a_0 = 0 \end{cases}$$

改写为:

$$\left.\begin{array}{l} 1\cdot a_6 + 1\cdot a_5 + 1\cdot a_4 + 0\cdot a_3 + 1\cdot a_2 + 0\cdot a_1 + 0\cdot a_0 = 0 \\ 1\cdot a_6 + 1\cdot a_5 + 0\cdot a_4 + 1\cdot a_3 + 0\cdot a_2 + 1\cdot a_1 + 0\cdot a_0 = 0 \\ 1\cdot a_6 + 0\cdot a_5 + 1\cdot a_4 + 1\cdot a_3 + 0\cdot a_2 + 0\cdot a_1 + 1\cdot a_0 = 0 \end{array}\right\}$$

上式可以表示成如下矩阵形式：

$$\begin{bmatrix}1110100\\1101010\\1011001\end{bmatrix}\begin{bmatrix}a_6\\a_5\\a_4\\a_3\\a_2\\a_1\\a_0\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix}\quad（模2）$$

简记为$\boldsymbol{H}\cdot\boldsymbol{A}^{\mathrm{T}}=\boldsymbol{0}^{\mathrm{T}}$ 。$\boldsymbol{H}$为监督矩阵，$\boldsymbol{H}=\begin{bmatrix}1110100\\1101010\\1011001\end{bmatrix}$

将

$$\begin{cases} a_2 = a_6 \oplus a_5 \oplus a_4 \\ a_1 = a_6 \oplus a_5 \oplus a_3 \\ a_0 = a_6 \oplus a_4 \oplus a_3 \end{cases}$$

改成矩阵形式：

$$\begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 1110 \\ 1101 \\ 1011 \end{bmatrix} \begin{bmatrix} a_6 \\ a_5 \\ a_4 \\ a_3 \end{bmatrix}$$

转置写成

$$[a_2 a_1 a_0] = [a_6 a_5 a_4 a_3]\begin{bmatrix}111\\110\\101\\011\end{bmatrix} = [a_6 a_5 a_4 a_3]\boldsymbol{Q}$$

将**Q**的左边加上**1**个$\boldsymbol{k \times k}$阶单位方阵，就构成生成矩阵$\boldsymbol{G}$：

$$\boldsymbol{G} = [I_k \boldsymbol{Q}] = \left[\begin{array}{c:c}1000 & 111\\0100 & 110\\0010 & 101\\0001 & 011\end{array}\right]$$

则：$[a_6 a_5 a_4 a_3 a_2 a_1 a_0] = [a_6 a_5 a_4 a_3]\cdot \boldsymbol{G}$　即：$\boldsymbol{A} = [a_6 a_5 a_4 a_3]\cdot \boldsymbol{G}$

6.2 线性分组码

➢ 编码与译码

□ 校正子与差错检测

假设 **c** 为要通过有噪信道传输的码字，**y** 为信道输出端接收到的码字，由于噪声影响，**c** 与 **y** 可能不同。用**错误图样** **e** 表示收发码字之差：

$$\begin{aligned}\mathbf{e}&=\mathbf{c}+\mathbf{y}\\&=[c_{n-1}\ c_{n-2}\ \cdots\ c_0]+[y_{n-1}\ y_{n-2}\ \cdots\ y_0]\\&=[e_{n-1}\ e_{n-2}\ \cdots\ e_0]\end{aligned}$$

若 $\mathbf{e}=0$，则 $\mathbf{y}=\mathbf{c}$，传输无误码；若 $\mathbf{e}=1$，则传输有误码。

可见

错误图样 **e** 反映了接收码组的出错情况

6.2 线性分组码

令 $\mathbf{s}=\mathbf{y}\mathbf{H}^{\mathrm{T}}$，称为伴随式或校正子。则接收端利用接收到的码字 $\mathbf{y}$ 计算校正子

$$\mathbf{s}=\mathbf{y}\mathbf{H}^{\mathrm{T}}=(\mathbf{c}+\mathbf{e})\mathbf{H}^{\mathrm{T}}=\mathbf{e}\mathbf{H}^{\mathrm{T}}$$

由此可见，伴随式 $\mathbf{s}$ 与错误图样 $\mathbf{e}$ 之间有确定的线性变换关系，与发送码字 $\mathbf{y}$ 无关。接收端译码器的任务就是从伴随式确定错误图样，然后用接收到的码字减去错误图样以恢复发送端发送的正确码字。

由以上分析可知，(n, k) 线性分组码译码的三个基本步骤：

1）计算接收码字 $\mathbf{c}$ 的伴随式 $\mathbf{s}$；

2）根据 $\mathbf{s}$ 找出错误图样 $\mathbf{e}$，判定误码位置；

3）根据 $\mathbf{e}$ 纠正错误，得到正确的码字 $\mathbf{c}=\mathbf{e}+\mathbf{y}$(模 2 加)

6.2 线性分组码

□ 伴随式与错误图样

(7,4)线性分组码的伴随式 **s** 与错误图样 **e** 的对应关系如下表所示。从表和(7,4)线性分组码的校验矩阵 **H** 可知，当 **s** 为非零向量时，校正子 **s** 的转置与 **H** 的列向量一一对应。因此，当校正子 **s** 的转置等于 **H** 的第 i 列向量时，错码的位置为 c_i 。

序号	错误位置	**e**	**s**
		$e_6e_5e_4e_3e_2e_1e_0$	$s_2s_1s_0$
0	无	0 0 0 0 0 0 0	0 0 0
1	y_0	0 0 0 0 0 0 1	0 0 1
2	y_1	0 0 0 0 0 1 0	0 1 0
3	y_2	0 0 0 0 1 0 0	1 0 0
4	y_3	0 0 0 1 0 0 0	0 1 1
5	y_4	0 0 1 0 0 0 0	1 0 1
6	y_5	0 1 0 0 0 0 0	1 1 0
7	y_6	1 0 0 0 0 0 0	1 1 1

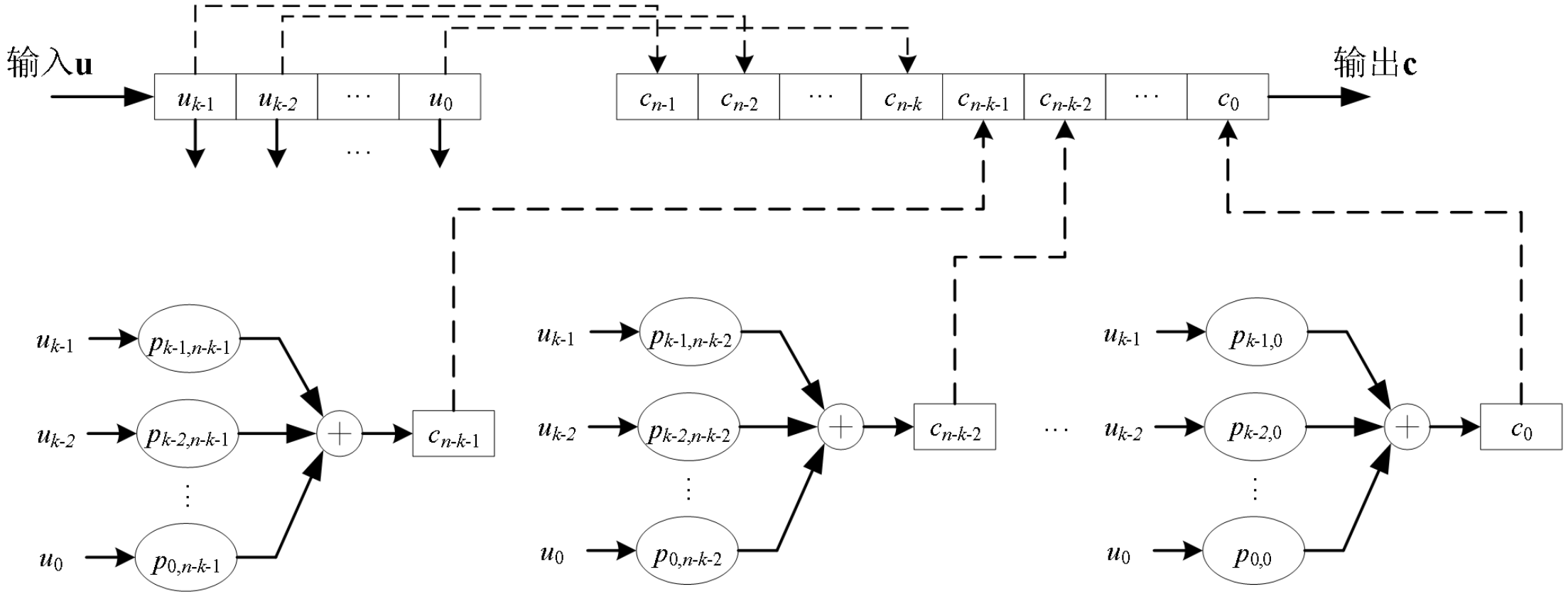
例

当 **s** = [0 0 1] 时，它等于 **H** 的第 1 列向量，错码的位置为c_1

6.2 线性分组码

编码和译码电路

$$c = uG$$

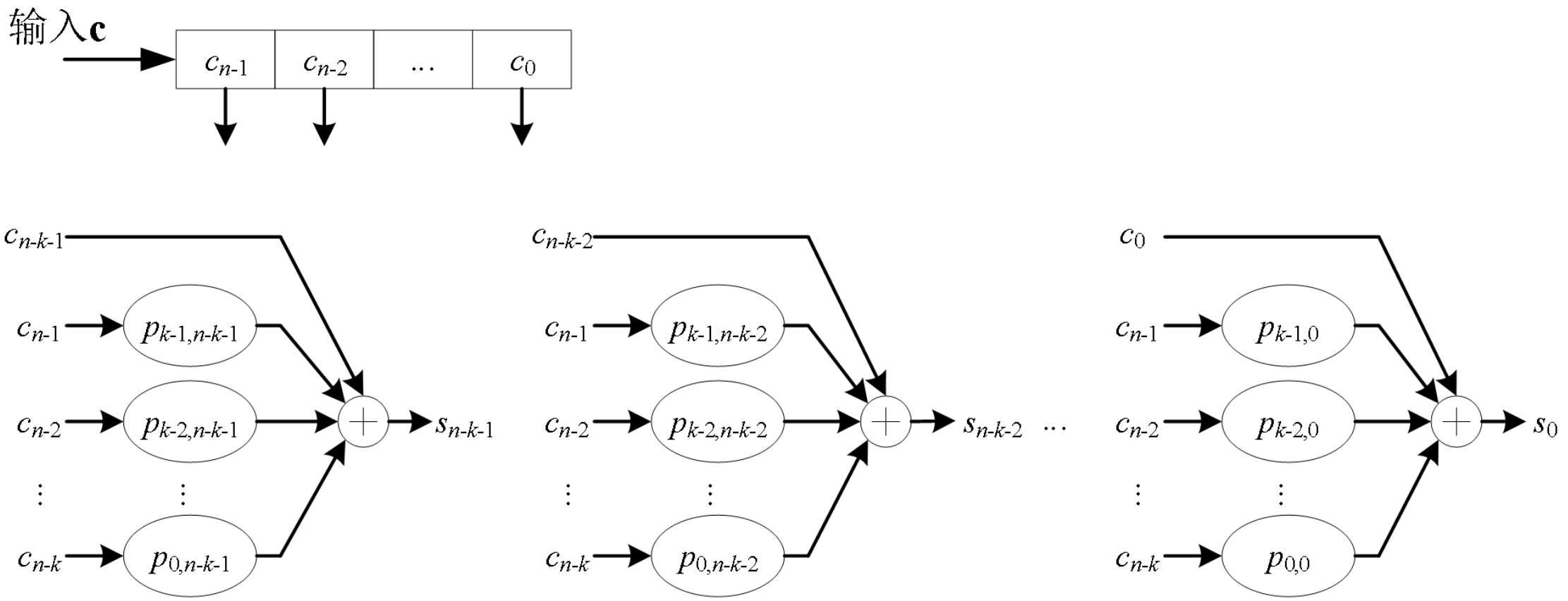


(n, k) 线性系统码的编码电路图

6.2 线性分组码

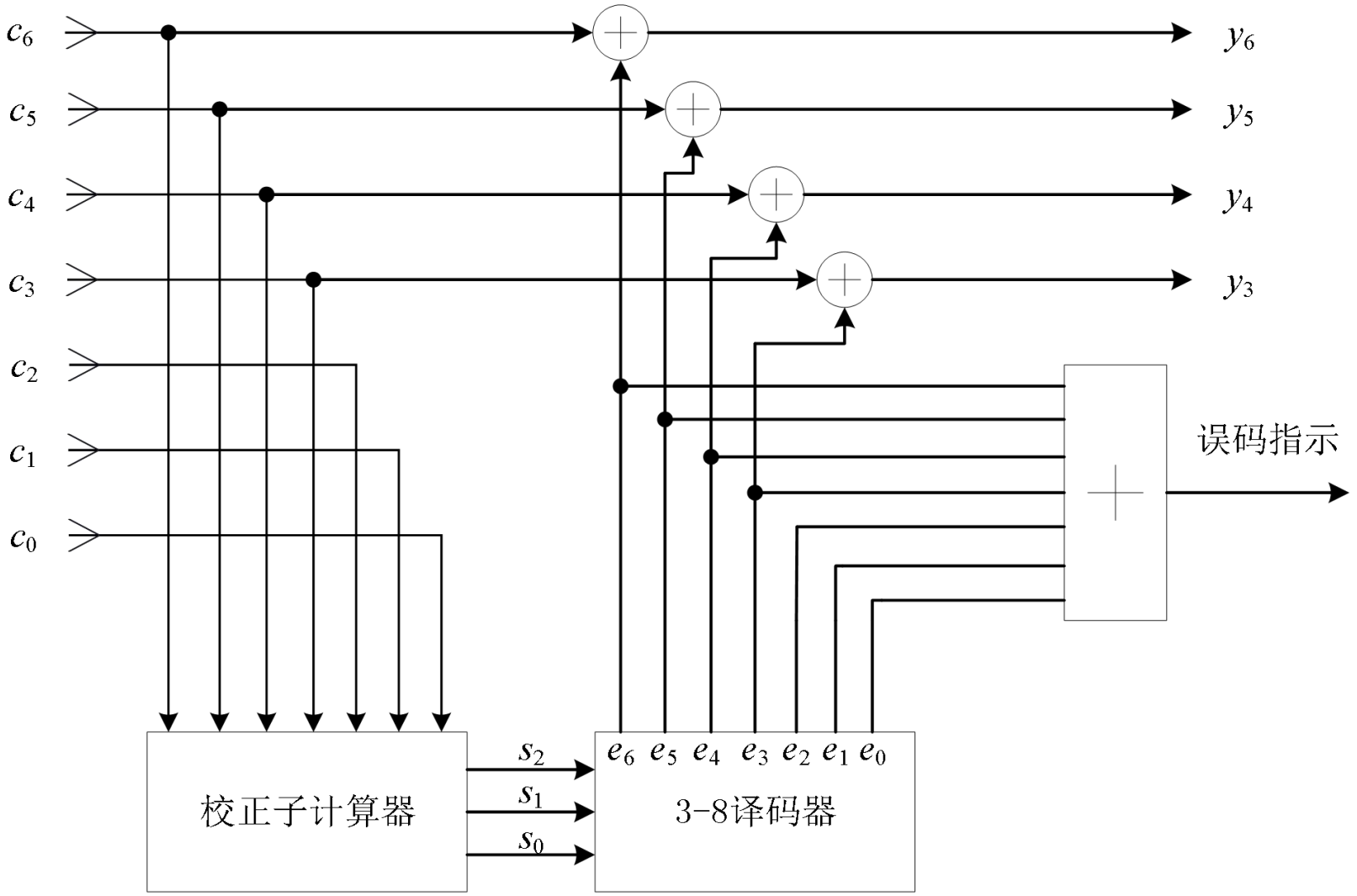
校正子电路

$$s = yH^T$$



6.2 线性分组码

- **译码器电路原理图**



6.2 线性分组码

例：设线性分组码的生成矩阵为

$$G=\begin{bmatrix}0&0&1&0&1&1\\1&0&0&1&0&1\\0&1&0&1&1&0\end{bmatrix}$$

（1）确定（n, k）码中的n, k;

（2）求典型监督矩阵H;

（3）写出监督方程;

（4）列出所有码字;

（5）确定最小码距;

（6）列出错误图样表;

（7）若收到码字000011，是否错码？如何纠正

解：（1）生成矩阵G是k行n列，所以k=3, n=6, 该（n, k）码为（6, 3）码

（2）将G进行初等行运算，原矩阵的第2, 3, 1分别作为典型阵的第1, 2, 3行，得典型生成矩阵为：

$$G=\begin{bmatrix}1&0&0&1&0&1\\0&1&0&1&1&0\\0&0&1&0&1&1\end{bmatrix}=[I_kQ]$$

其中：$$Q=\begin{bmatrix}1&0&1\\1&1&0\\0&1&1\end{bmatrix}\qquad P=Q^T=\begin{bmatrix}1&1&0\\0&1&1\\1&0&1\end{bmatrix}$$

于是：$$H=[PI_r]=\begin{bmatrix}1&1&0&1&0&0\\0&1&1&0&1&0\\1&0&1&0&0&1\end{bmatrix}$$

（3）监督码元也信息码元之间的关系称为监督方程，监督矩阵H的每行中“1”的位置表示相应码元之间存在的监督关系，即：

$$a_5 \oplus a_4 \oplus a_2 = 0$$

$$a_4 \oplus a_3 \oplus a_1 = 0$$

$$a_5 \oplus a_3 \oplus a_0 = 0$$

（4）设A为许用码组，则由

$$A=[a_6 \quad a_5 \quad a_4]\bullet G=[a_6 \quad a_5 \quad a_4]\bullet\begin{bmatrix}1&0&0&1&0&1\\0&1&0&1&1&0\\0&0&1&0&1&1\end{bmatrix}$$

可得全部码字，见下表：

信息位	监督位	信息位	监督位
$a_5a_4a_3$	$a_2a_1a_0$	$a_5a_4a_3$	$a_2a_1a_0$
000	000	100	101
001	011	101	110
010	110	110	011
011	101	111	000

6.2 线性分组码

（5）由于线性码的最小码距就是码的最小重量（全“0”码除外），所以该码的最小码距d_{min}=3

（6）错误图样。根据 $S = EH^T$ 可得错误图样，见下表：

错码位置	$S_1S_2S_3$	错码位置	$S_1S_2S_3$
a_5	101	a_1	010
a_4	110	a_0	001
a_3	011	无错	000
a_2	100		

注：错码位置为a_5，对应E=100000。其它类似

（7）

$$S = R \cdot H^{T} = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1] \times \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [0 \quad 1 \quad 1]$$

对照错误图样表可知，有错，a_3错，故正确码组应该为001011

错码位置	$S_1S_2S_3$	错码位置	$S_1S_2S_3$
a_5	101	a_1	010
a_4	110	a_0	001
a_3	011	无错	000
a_2	100		