

码距是根据任意两个合法码之间至少有几个二进制位不相同而确定的,若仅有一位不同,称其码距为1。例如,用4位二进制表示6种状态,则16种编码都用到了,此时码距为1,就是说,任何一个状态的4位码中的一位或几位出错,就变成另一个合法码,此时无查错能力。若用四个二进制位表示8个状态,就可以只用其中的8种编码,而把另8种编码作为非法编码,此时码距为2。

2.1.8 什么是奇偶校验码?

奇偶校验码的原理是:在每组代码中增加1个冗余位,使合法编码的最小码距由1增加到2。如果合法编码中有奇数个位发生了错误,这个编码就将成为非法的代码。增加的冗余位称为奇偶校验位。

1. 校验码的构成规则

偶校验:每个码字(包括校验位)中1的数目为偶数。

奇校验:每个码字(包括校验位)中1的数目为奇数。

2. 校验位的形成

设有效信息为 $D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$, 则

偶校验:在发送端求校验位 $P = D_7 \oplus D_6 \oplus D_5 \oplus D_4 \oplus D_3 \oplus D_2 \oplus D_1 \oplus D_0$

奇校验:在发送端求校验位 $P = D_7 \oplus D_6 \oplus D_5 \oplus D_4 \oplus D_3 \oplus D_2 \oplus D_1 \oplus D_0$

3. 校验原理

偶校验在接收端求 $P' = D_7 \oplus D_6 \oplus D_5 \oplus D_4 \oplus D_3 \oplus D_2 \oplus D_1 \oplus D_0 \oplus P$

奇校验在接收端求 $P' = D_7 \oplus D_6 \oplus D_5 \oplus D_4 \oplus D_3 \oplus D_2 \oplus D_1 \oplus D_0 \oplus P$

若 $P' = 0$, 则无错; 若 $P' = 1$, 则有错。

一般采用“异或”电路得到校验位。这种方式只能发现奇数个错误,且不能纠正错误。

2.1.9 什么是海明码?

海明校验的基本思想是将有效信息按某种规律分成若干组,每组安排一个校验位进行奇偶测试。在一个数据位组中加入几个校验位,会增加数据代码间的码距,当某一位发生变化时会引起校验结果发生变化,不同代码位上的错误会得出不同的校验结果。因此,海明码能检测出2位错误,并能纠正1位错误。

1. 求海明校验码的步骤

(1) 确定海明校验位的位数

设 K 为有效信息的位数, r 为校验位的位数, 则整个码字的位数 N 应满足不等式:

$$N = K + r \leq 2^r - 1$$

若要求海明码能检测出2位错误,则再增加1位校验位。

(2) 确定校验位的位置

位号(1~ N)为2的权值的那些位,即 $2^0, 2^1, 2^2, \dots, 2^{r-1}$ 的位置作为校验位,记作 $P_1, P_2, \dots, P_r$, 余下的为有效信息位。

(3) 分组

将 N 位分 r 组,第 i 位由校验位号之和等于 i 的那些校验位所校验。

(4) 校验位的形成

P_1 = 第一组中的所有位(除 P_1 外)求异或

P_2 = 第二组中的所有位(除 P_2 外)求异或

⋮

P_r = 第 r 组中的所有位(除 P_r 外)求异或

为了能检测两个错误,增加一位校验 P_{r+1} , 放在最高位。

P_{r+1} = 所有位(包括 $P_1, P_2, \dots, P_r$)共 N 位求异或。

2. 校验原理:在接收端分别求 $S_1, S_2, S_3, \dots, S_r$

S_1 = 第一组中的所有位(包括 P_1)求异或

S_2 = 第二组中的所有位(包括 P_2)求异或

⋮

S_r = 第 r 组中的所有位(包括 P_r)求异或

$S_{r+1} = P_{r+1} \oplus$ 所有位(包括 $P_1, P_2, \dots, P_r$)求异或

当 $S_{r+1} = 1$ 时有一位错,由 $S_r \dots S_3 S_2 S_1$ 的二进制编码指出出错位号,将其取反,即可纠错。

当 $S_{r+1} = 0$ 时无错或有偶数个错(两个错的可能性比较大),当 $S_r \dots S_3 S_2 S_1 = 0 \dots 000$ 时,接收的数或无错,或有两个错。

2.1.10 什么是循环码?

二进制信息沿一条线逐位在部件之间或计算机之间传送称为串行传送。CRC(Cyclic Redundancy Check)码可以发现并纠正信息存储或传送过程中连续出现的多位错误。

1. CRC 码的编码原理

CRC 码是用多项式 $M(x) \cdot x^r$ 除以称为生成多项式的 $G(x)$ (产生校验码的多项式)所得的余数作为校验位。为了得到 r 位余数(校验位), $G(x)$ 必须是 $r+1$ 位。

设所得余数表达式为 $R(x)$, 商为 $Q(x)$ 。将余数拼接在信息位左移 r 位后空出的 r 位上,就构成这个有效信息的 CRC 码。该 CRC 码可用多项式表示为

$$\begin{aligned} M(x) \cdot x^r + R(x) &= [Q(x) \cdot G(x) + R(x)] + R(x) \\ &= [Q(x) \cdot G(x)] + [R(x) + R(x)] \\ &= Q(x) \cdot G(x) \end{aligned}$$

因此所得 CRC 码可被 $G(x)$ 表示的数码除尽。

将编好的循环校验码称为 (n, k) 码。

2. CRC 码的编码电路

若 $G(x) = x^r + g_{r-1}x^{r-1} + \dots + g_2x^2 + g_1x^1 + 1, g_i \in \{0, 1\}, i \in \{r-1, \dots, 2, 1\}$, 则 CRC 码的编码电路如图 2.1.1 所示, 当 $g_i = 1$ 时, 开关 g_i 闭合, 当 $g_i = 0$ 时, 开关 g_i 断开。开始发送数据时, 开关 k 置于 k_m 位置, 开关 R 同时闭合, 信息位由输入端从高到低串行输入, 一边从码字输出端输出, 一边进入编码电路循环运算。当信息发送完毕时, 寄存器 $R_0, R_1, \dots, R_{r-1}$ 中的内容即为冗余位的值。此时, 开关 R 断开, 开关 k 置于 k_r 位置, 冗余位就紧接在信息位的后面输出。至此发送方编码发送完毕。

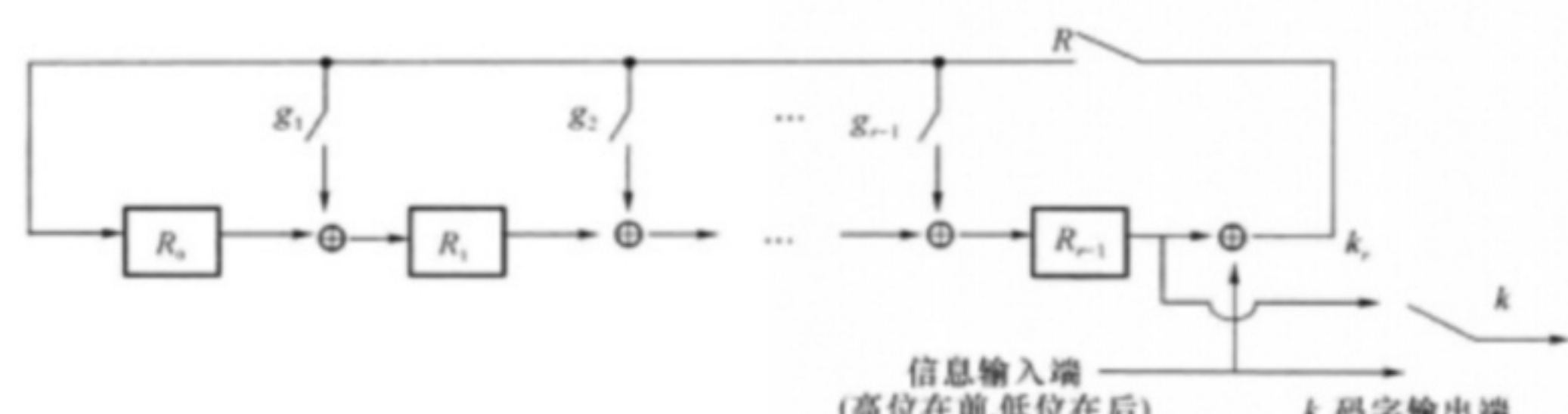


图 2.1.1 CRC 码的编码电路

3. CRC 的译码与纠错

在 CRC 校验中, 发送方已知信息序列 $M(x)$ 和生成多项式 $G(x)$, 发送方便可利用编码电路形成传输序列 $T(x)$ 进行发送。 $T(x)$ 经信道传输后到达接收方, 接收方收到信息 $T'(x)$, 若 $T'(x) = T(x)$, 则传输无错, 否则就是出错。接收方只是已知 $G(x)$ 和收到的 $T'(x)$, 但不知道发送方发送的 $T(x)$ 。判断 $T'(x)$ 是否有错的方法为: 将收到的循环校验码用约定的生成多项式 $G(x)$ 去除, 如果码字无误则余数应为 0, 如有某一位出错, 则余数不为 0, 不同位数出错余数不同。更换不同的待测码字可以证明余数与出错位的对应关系是不变的, 只与码制和生成多项式有关。对于其他码制或选用其他生成多项式, 出错模式将发生变化。

在实际工作中, 接收方一边接收信息, 一边利用译码电路进行差错检测。

若 $G(x) = x^r + g_{r-1}x^{r-1} + \dots + g_2x^2 + g_1x^1 + 1, g_i \in \{0, 1\}, i \in \{r-1, \dots, 2, 1\}$, 则译码电路如图 2.1.2 所示, 当 $g_i = 1$ 时, 开关 g_i 闭合, 当 $g_i = 0$ 时, 开关 g_i 断开。接收方将收到的码字信息逐位送入译码电路计算余数, 在码字信息全部进入译码电路后, 寄存器 $R_0, R_1, \dots, R_{r-1}$ 中就寄存了余数 $E(x)$ 。将余数各位进行或运算。得错误特征位 E , 若 $E = 0$, 则接收正确, 将码字中的最末 r 位去掉, 得到原发数据; 若 $E = 1$, 则表示收到的码字有错。

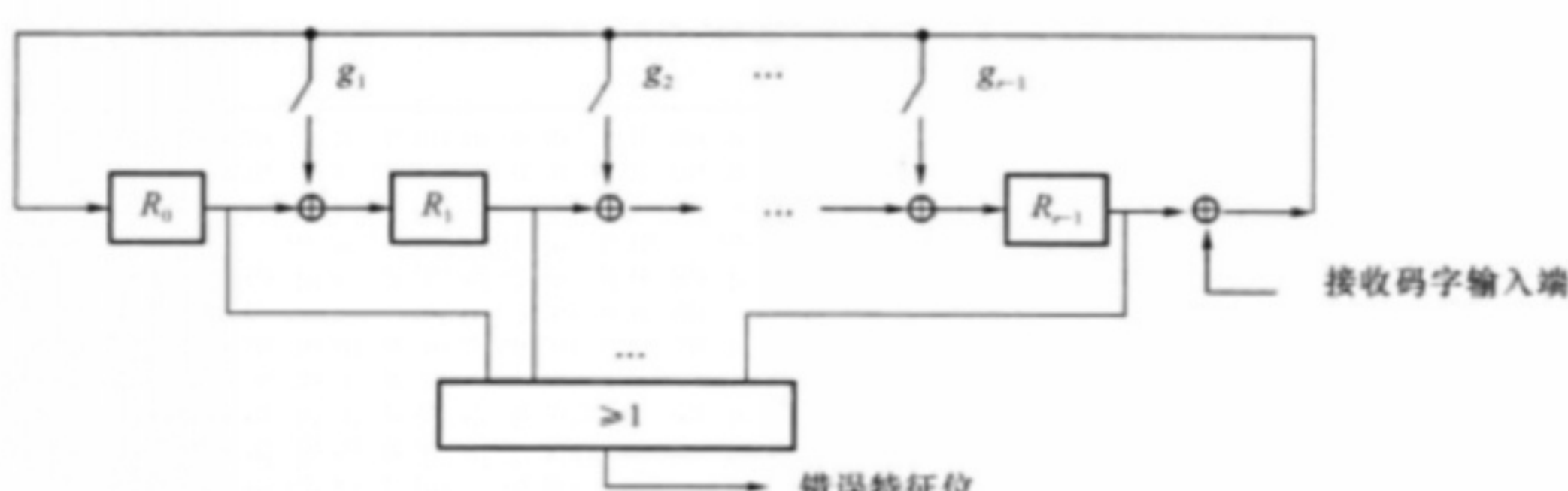


图 2.1.2 CRC 的译码电路

4. 关于生成多项式

并不是任何一个 $(r+1)$ 位多项式都可以作为生成多项式。从检错及纠错的要求出发, 生成多项式应能满足下列要求:

- 任何一位发生错误都应使余数不为 0。
- 不同位发生错误应当使余数不同。
- 对余数继续作模 2 除, 应使余数循环。

将这些要求反映为数学关系是比较复杂的, 对一个 (n, k) 码来说, 可将 $(x^n - 1)$ 分解为若干质因子, 根据编码所要求的码距选取其中的因式或若干因式的乘积作为生成多项式。