计算机组成原理

(3) 数据校验

王浩宇,教授

haoyuwang@hust.edu.cn

https://howiepku.github.io/

目录

- **数据校验**
- 奇偶校验码
- 海明码
- 循环冗余校验码

校验码（数据校验）

- **数据校验原因**
  - 为减少和避免数据在计算机系统运行或传送过程中发生错误，在数据的编码上提供了检错和纠错的支持。
- **数据校验码的定义**
  - 能够发现某些错误或具有自动纠错能力的数据编码；
  - 也称检错码；
- **数据校验的基本原理是扩大码距；**
  - 码距：任意两个合法码之间不同的二进制位的最少位数；
  - 仅有一位不同时，称其码距为1
  - 整个编码系统中任意两个码字的的最小距离就是该编码系统的码距

码距及作用

- **设用3位二进制表示8种状态**
  - 8种编码都用到了，此时码距为1
  - 任何一种状态的三位码中的一位或几位出错，就变成另一个合法码
  - 无查错能力
- **若用四位二进制表示8个状态**
  - 只用其中的8种编码，而把另8种编码作为非法编码
  - 可使码距扩大为2
    - 注意：并不是任选8种编码都可扩大码距

码距及作用

信息序号	二进码字		
	a2	a1	a0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

码距为1

信息序号	二进码字			
	a3	a2	a1	a0
0	0	0	0	0
1	1	0	0	1
2	1	0	1	0
3	0	0	1	1
4	1	1	0	0
5	0	1	0	1
6	0	1	1	0
7	1	1	1	1

码距为2

校验码的类型

奇偶校验码

- 判断数据中1的个数设置1位校验位；
- 分奇校验和偶校验两种，只能检错，无纠错能力；

海明校验码

- 在奇偶校验的基础上增加校验位而得；
- 具有检错和纠错的能力；

循环冗余校验码（CRC）

- 通过模2的除法运算建立数据信息和校验位之间的约定关系；
- 具有很强的检错纠错能力。

奇偶校验码——概念

■奇偶校验原理

- 在数据中增加1个冗余位，使码距由1增加到2；
- 如果合法编码中有奇数个位发生了错误，就将成为非法代码。
- 增加的冗余位称为奇偶校验位。

■校验的类型

- 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。

■校验过程

- 发送端：按照校验类型，在发送数据后添加校验位P；
- 接收端：对接收到的数据（包括校验位）进行同样类型的校验，决定数据传输中是否存在错误；

奇偶校验码——校验原理

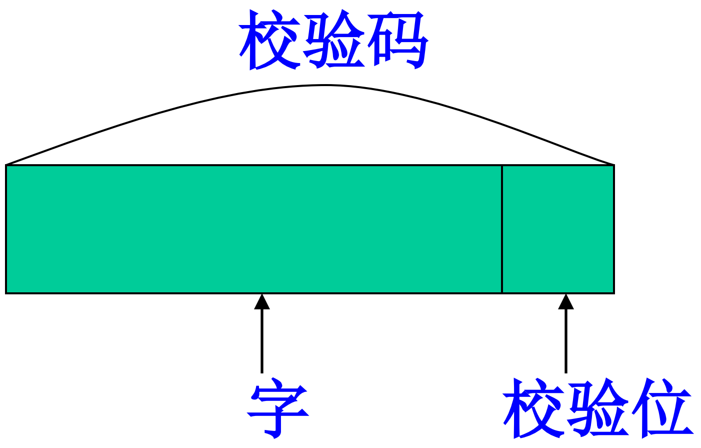
- **偶校验：在接收端求校验位**
  - $P' = D_7 \oplus D_6 \oplus D_5 \oplus D_4 \oplus D_3 \oplus D_2 \oplus D_1 \oplus D_0 \oplus P$
  - 若P’＝0，则无错；若P’＝1，则有错。
- **奇校验：在接收端求校验位**
  - $P' = D_7 \oplus D_6 \oplus D_5 \oplus D_4 \oplus D_3 \oplus D_2 \oplus D_1 \oplus D_0 \oplus P$
  - 若P’＝1，则无错；若P’＝0，则有错。
- **电路实现：**
  - 一般采用异或电路得到校验位。

1010 1011 → 求校验码 →

偶校验码 1010 1011 1

奇校验码 1010 1011 0

奇偶校验码——例题



例1： 数据	奇校验码	偶校验码
0010 0001	0010 0001 1	0010 0001 0

例2：数据 ： 0111 0101 → 0111 0101 1

发送端 （门电路） 偶校验码

0110 0101 1

接收端

出错！

奇偶校验码——例题

数据 ： 0111 0101 → 0111 0101 0

发送端 （门电路） 奇校验码

0110 0111 0

接收端

正确

奇偶校验只能发现奇数个错误，且不能纠正错误！

海明码

- **海明码是1950年提出的；**
  - 只要增加少数的几位校验码，即可检测出多位出错，并能自动恢复一或几位出错信息；
- **实现原理：**
  - 在一个数据中加入几个校验位，每个校验位和某几个特定的信息位构成偶校验的关系；
  - 接收端对每个偶关系进行校验，产生校验因子；
  - 通过校正因子区分无错和码字中的n个不同位置的错误；
    - 不同代码位上的错误会得出不同的校验结果；

海明码——确定校验位的位数

- 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：
  - $N = K + r \leqq 2^r - 1$ **为什么?**
  - 通常称为（N，K）海明码
- 设某(7,4)海明码表示的码字长度为___位，校验位数为___位。
- 例如：数据$D_3D_2D_1D_0$ =1001
  - K=4，$r+5 \leqq 2^r$；
  - 可知，需要校验位3位$P_3P_2P_1$；

海明码——确定校验位的位置

- **数据表示**
  - 数据位D（$D_iD_{i-1}\cdots D_1D_0$）、校验位P（$P_jP_{j-1}\cdots P_2P_1$）
  - 海明码H （包括数据位和校验位）：$H_mH_{m-1}\cdots H_2H_1$；
- **分组原则**
  - 每个校验位P_i从低到高被分在海明码中位号2^{i-1}的位置；
- **例如：数据$D_3D_2D_1D_0$ =1001，校验位$P_3P_2P_1$**
  - 海明码共7位$H_7H_6\cdots H_2H_1$ ，各位分配如下：

H_7	H_6	H_5	H_4	H_3	H_2	H_1
D_3	D_2	D_1	P_3	D_0	P_2	P_1

海明码——校验分组

- **校验原则**
  - 海明码的每一位H_i有多个校验位校验，其关系是被校验的每一位位号等于校验它的各校验位的位号之和；
  - 每个信息位的位置写成用2的幂次之和的形式 ；
- **例如**
  - H_7参与H_1、H_2、H_4的校验；
  - H_6参与H_2、H_4的校验；
  - H_5参与H_1、H_4的校验；
  - H_3参与H_1、H_2的校验；
- **分组情况**

	H_7	H_6	H_5	H_4	H_3	H_2	H_1
	D_3	D_2	D_1	P_3	D_0	P_2	P_1
第一组P_1	√		√		√		
第二组P_2	√	√			√		
第三组P_3	√	√	√				

第一组（P_1、D_3、D_1、D_0）
第二组（P_2、D_3、D_2、D_0 ）
第三组（P_3、D_3、D_2、D_1 ）

海明码——校验位的形成

- 校验位形成公式
  - P_1 = 第一组中所有位(除P_1)求异或
  - ……
  - P_j = 第j组中所有位(除P_j)求异或

第一组（P_1、D_3、D_1、D_0）
第二组（P_2、D_3、D_2、D_0）
第三组（P_3、D_3、D_2、D_1）

- 为了能检测两个错误，增加一位校验P_{j+1}，放在**最高位**
  - P_{j+1} = 所有位(包括P_1，P_2，…，P_j)求异或

但不能纠错!

- 例如：
  - $P_1 = D_3 \oplus D_1 \oplus D_0 = 1 \oplus 0 \oplus 1 = 0$
  - $P_2 = D_3 \oplus D_2 \oplus D_0 = 1 \oplus 0 \oplus 1 = 0$
  - $P_3 = D_3 \oplus D_2 \oplus D_1 = 1 \oplus 0 \oplus 0 = 1$
  - $P_4 = D_3 \oplus D_2 \oplus D_1 \oplus D_0 \oplus P_3 \oplus P_2 \oplus P_1 = 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 1$

海明码——接收端校验（1/2）

- 接收端接收到数据后，分别求S_1，S_2，S_3，…，S_j
  - S_1 = 第一组中所有位(包括P_1)求异或
  - …
  - S_j = 第j组中所有位(包括P_j)求异或
  - $S_{j+1} = P_{j+1} \oplus$所有位(包括P_1，P_2，…，P_j)求异或
- 当S_{j+1} = 1时，有一位出错；
  - 由$S_j \ldots S_3 S_2 S_1$的编码指出出错位号，将其取反，即可纠错。
- 当S_{j+1} = 0时，无错或有偶数个错（两个错的可能性比较大）；
- 当$S_j \ldots S_3 S_2 S_1 = 0 \ldots 0\,00$时，接收的数无错，否则有两个错。

海明码——接收端校验（2/2）

- 同上例，接收端接收的数据为
- 接收端求S
  - $S_1 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$
  - $S_2 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$
  - $S_3 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$
  - $S_4 = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 0$
- 若接收端接收到错误的数据
  - $S_1 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$
  - $S_2 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$
  - $S_3 = 1 \oplus 1 \oplus 1 \oplus 0 = 1$
  - $S_4 = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 1$

H_8	H_7	H_6	H_5	H_4	H_3	H_2	H_1
P_4	D_3	D_2	D_1	P_3	D_0	P_2	P_1
1	1	1	0	1	1	0	0

第一组（P_1、D_3、D_1、D_0）
第二组（P_2、D_3、D_2、D_0）
第三组（P_3、D_3、D_2、D_1）

无错误！

S_4=1，有错误！
$S_3S_2S_1$=110，H_6位有错，应取反！

【练习】设待校验的数据为 D7～D0＝10101011，写出其海明校验码。

【解】

①确定海明校验位的位数

因为$K=8$， 由$N=K+r\leq 2^r-1$，得$9+r\leq 2^r$，校验位的位数为$r=4$。

②确定校验位的位置

i： 12 11 10 9 8 7 6 5 4 3 2 1

D_7 D_6 D_5 D_4 P_4 D_3 D_2 D_1 P_3 D_0 P_2 P_1

③分组（N位分r组）

位号i	12	11	10	9	8	7	6	5	4	3	2	1
	D_7	D_6	D_5	D_4	P_4	D_3	D_2	D_1	P_3	D_0	P_2	P_1
	1	0	1	0		1	0	1		1		
第一组(P_1)		√		√		√		√		√		
第二组(P_2)		√	√			√	√			√		
第三组(P_3)	√					√	√	√				
第四组(P_4)	√	√	√	√								

【练习】设待校验的数据为D7～D0＝10101011，写出其海明校验码

④校验位的形成

P1=D6⊕ D4 ⊕ D3 ⊕ D1 ⊕ D0 ＝ 1;

P2=D6⊕ D5 ⊕ D3 ⊕ D2 ⊕ D0 ＝ 1

P3=D7⊕ D3 ⊕ D2 ⊕ D1 ＝1；

P4=D7⊕ D6 ⊕ D5 ⊕ D4 ＝0

所以，信息码10101011的海明校验码为：1010 0 101 1 1 11

海明码的纠错与检错能力

- 一个系统能纠正一位差错时，码距最小是3；
  - 码距为3时，或能纠正一位错，或能检测二位错；
  - 但不能同时纠正一位错并检测二位错。
- 码距为1至7时，海明码的纠错和检错能力如右表：
- 码距越大，纠错能力越强，但数据冗余也越大，即编码效率低了。

码距	码 能 力	
	检错	纠错
1	0	0
2	1	0
3	2 或 1	
4	2 并 1	
5	2 并 2	
6	3 并 2	
7	3 并 3	

目录

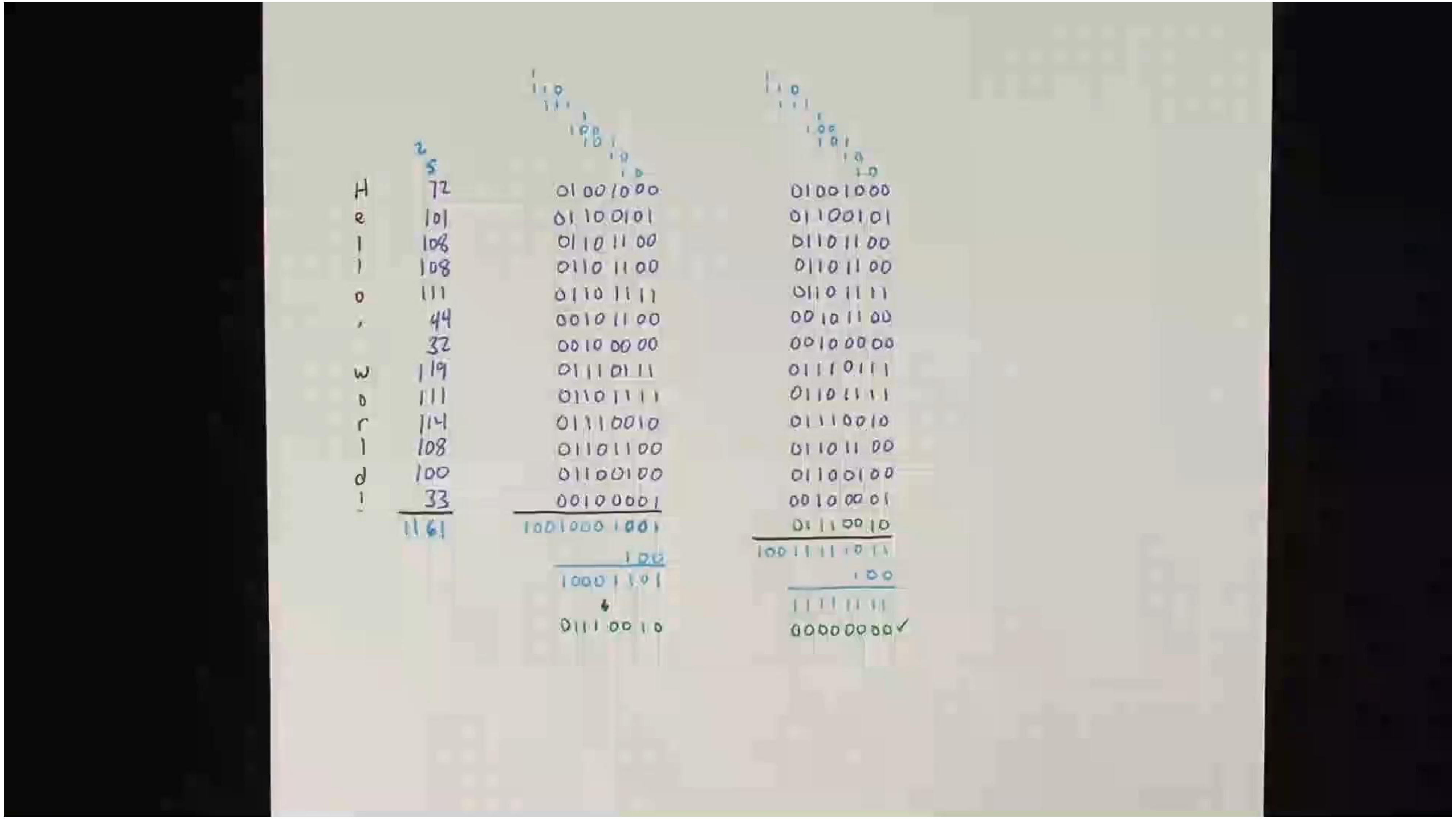
- 数据校验
- 奇偶校验码
- 海明码
- **循环冗余校验码**

CRC校验的基本原理

- 在要发送的帧后面附加一个数（校验码），生成一个新帧发送给接收端
- 附加的校验码使所生成的新帧能与发送端和接收端共同选定的某个特定数整除（模2除法）

- 到达接收端后，再把接收到的新帧除以这个选定的除数。
- 因为在发送端发送数据帧之前就已通过附加一个数，做了“去余”处理（也就已经能整除了），所以结果应该是没有余数
- 如果有余数，则表明该帧在传输过程中出现了差错。

CRC校验的基本原理



https://www.bilibili.com/video/BV13F411F77r/

CRC校验的基本原理

- **模2运算**
  - 模2加减法运算
    - 没有进位和借位的二进制加法和减法运算
    - 相同的两个二进制数的模2加法与减法的运算结果相同，异或运算
    - 0±0 = 0， 0±1 = 1， 1±0 = 1， 1±1 = 0
  - 模2乘法运算
    - 按照模2加法运算求部分积之和，运算过程中不考虑进位
  - 模2除法运算
    - 按模2减，求部分余数，不借位

CRC校验的基本原理

模2除法运算上商原则

①部分余数首位为1时，商为1，减除数；

②部分余数首位为0时，商为0，减0；

③当部分余数的位数小于除数的位数时，该余数即为最后余数

1 0 1	商
1 0 1) 1 0 0 1 0	首位为 1
1 0 1	模 2 减，商 1
0 1 1	首位为 0
0 0 0	减 0，　商 0
1 1 0	首位为 1
1 0 1	模 2 减，商 1
1 1	余数位数少于除数，为最后余数

CRC校验的基本原理

- 增加冗余码（校验位）

有效信息(k位)	校验信息(r位)

$N=k+r \le 2^r - 1$

- 生成多项式G(x)

收发双方约定的一个(r + 1)位生成多项式G(X)，发送方利用G(X)对信息多项式做模2除运算,生成校验码。接收方利用G(X)对收到的编码多项式做模2除运算检测差错及错误定位。

- G(x)应满足的条件

A、最高位和最低位必须为1；

B、当被传送信息（CRC码）任何一位发生错误时，被生成多项式做除后应该使余数不为0；

C、不同位发生错误时，模2除运算后余数不同；

D、CRC编码的非0余数具有循环特性：对不为0余数继续进行模2除运算应使余数循环

CRC校验的基本原理

- 常见生成多项式G(x)

N	K	码距d	G(x)多项式	G(x)
7	4	3	x^3+x+1	1011
7	4	3	x^3+x^2+1	1101
7	3	4	$x^4+x^3+x^2+1$	11101
7	3	4	x^4+x^2+x+1	10111
15	11	3	x^4+x+1	10011
15	7	5	$x^8+x^7+x^6+x^4+1$	111010001
31	26	3	x^5+x^2+1	100101
31	21	5	$x^{10}+x^9+x^8+x^6+x^5+x^3+1$	11101101001
63	57	3	x^6+x+1	1000011
63	51	5	$x^{12}+x^{10}+x^5+x^4+x^2+1$	1010000110101

CRC编码方法

(1)根据待校验信息的长度k，按照 $k+r \le 2^r - 1$ 确定校验位r的位数 如对4位信息 1100 进行CRC编码，根据 $4+r \le 2^r - 1$

得 $r_{min} = 3$

(2)根据r 和生成多项式的选择原则，选择位数为 r +1 的生成多项式G(X)= 1011

(3)进行下列变化

有效信息(k位)	校验信息(r位)	⇨	1100	000

即：将待校验的二进制信息Q(X)逻辑左移 r 位,得到Q(X)′

CRC编码方法

(4)对Q(X)’按模2运算法则除G(x)，求CRC编码中的r位校验信息

```
            1 1 1 0
1011  ／ 1 1 0 0 0 0 0
         1 0 1 1
           1 1 1 0
           1 0 1 1
             1 0 1 0
             1 0 1 1
                 0 1 0
```

(5)用得到的余数替换Q(X)’的最后r位即可得到对应的CRC编码

1100 000 ⟹ 1100 010　即为1100 的CRC 编码

CRC的检错与纠错

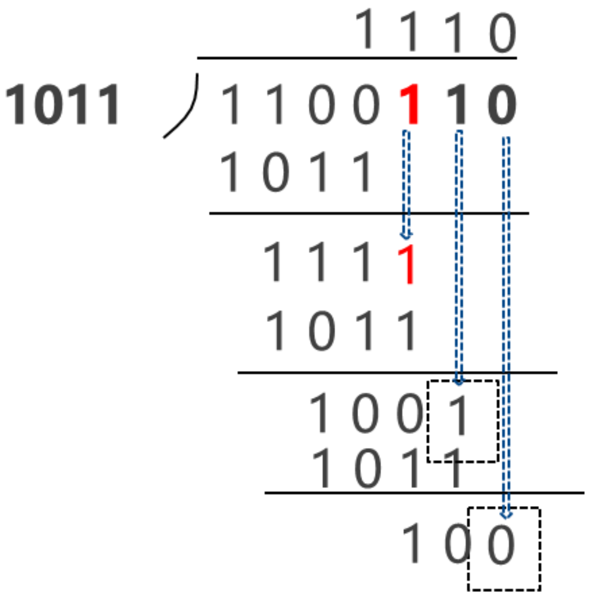
接收方利用G(X)对收到的编码多项式做模2除运算

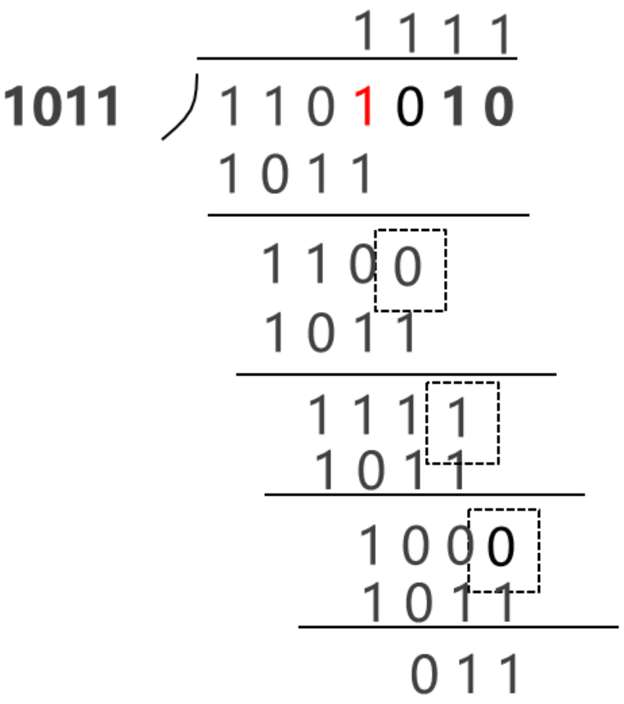
```
            1 1 1 0
       ┌─────────────
1011   │ 1 1 0 0 0 1 0
         1 0 1 1
         ─────────
           1 1 1 0
           1 0 1 1
           ─────────
             1 0 1 1
             1 0 1 1
             ─────────
                 0 0 0
```

余数为0说明传输没有错误

CRC的检错与纠错

接收方利用G(X)对收到的**有错**编码多项式做模2除运算





余数不为0说明传输有错

CRC的检错与纠错

- (7,4）编码不同数位出错对应的余数

A_1~A_7	余数	出错位
1100010	000	无
1100011	001	7
1100000	010	6
1100110	100	5
1101010	011	4
1110010	110	3
1000010	111	2
0100010	101	1

G(X) = 1011

A_1~A_7	余数	出错位
1100101	000	无
1100100	001	7
1100111	010	6
1100001	100	5
1101101	101	4
1110101	111	3
1000101	011	2
0100101	110	1

G(X) = 1101

CRC的检错与纠错

- 一位出错情况下余数的循环特性

```
           1110
1011 ) 1100110
       1011
       ----
        111
        1011
        ----
         100
         1011
         ----
            100
```

```
          1111
1011 ) 1101010
       1011
       ----
        110
        1011
        ----
         111
         1011
         ----
          100
          1011
          ----
           011
```

```
          1
1011 ) 1000
       1011
       ----
        011
```

CRC的检错与纠错

- 利用出错情况下余数的循环特性进行纠错

A_1~A_7	余数	出错位
1100010	000	无
1100011	001	7
1100000	010	6
1100110	100	5
1101010	011	4
1110010	110	3
1000010	111	2
0100010	101	1

若余数不为0，一边对余数补0继续做模2除，同时让被检测的校验码循环左移，当余数为101时，出错位也移到A1位置。通过异运算纠正后继续循环左移和执行余数模2除法，直到修改后的出错位回原位。不需对每一位提供纠正电路。

当位数增多时，循环码校验能有效地降低硬件代价，这是它得以广泛应用的主要原因。

CRC的纠错示例

A_1~A_7	余数	出错位
1100010	000	无
1100011	001	7
1100000	010	6
1100110	100	5
1101010	011	4
1110010	110	3
1000010	111	2
0100010	101	1

情况说明	校验码当前情况	运算	余数
发现出错，开始纠错	1100011	1100011与G(X)模二除	001
校验码循环左移，同时余数001补0，成为0010，与G(X) 模二除	1000111	0010与G(X)模二除	010
校验码循环左移，同时余数010补0，成为0100，与G(X) 模二除	0001111	0100与G(X)模二除	100
校验码循环左移，同时余数100补0，成为1000，与G(X) 模二除	0011110	1000与G(X)模二除	011
校验码循环左移，同时余数011补0，成为0110，与G(X) 模二除	0111100	0110与G(X)模二除	110
校验码循环左移，同时余数110补0，成为1100，与G(X) 模二除	1111000	1100与G(X)模二除	111
校验码循环左移，同时余数111补0，成为1110，与G(X) 模二除	1110001	1110与G(X)模二除	101
发现当前余数位101，已知101代表第一位出错，把第一位与1进行异或运算(或取反运算)，纠错	0110001	无运算	无
校验码循环左移，同时余数101补0，成为1010，与G(X) 模二除	1100010	1010与G(X)模二除	001

https://blog.csdn.net/limaodx/article/details/116559204