

《计算机组成原理》

（第二讲习题答案）

厦门大学信息学院软件工程系 曾文华

2022年2月24日

第2章 数据信息的表示

- 2.1 数据表示的作用
- 2.2 数值数据的表示
- 2.3 非数值数据的表示
- 2.4 数据信息的校验

习题 (P53-56)

- 2.2
- 2.4
- 2.5
- 2.6
- 2.7
- 2.9
- 2.10
- 2.13
- 2.16
- 2.17
- 2.18

习题答案（P53-56）

- 2.1: 解释下列名词:

1. 真值: +1010, -0.1101
2. 机器码: 由符号和数值一起编码表示的二进制数称为机器码（或机器数）。
3. 原码: 正数的原码数值位为本身, 符号位为0; 负数的原码数值位为本身, 符号位为1。
4. 反码: 正数的反码数值位为本身, 符号位为0; 负数的反码数值位为本身取反, 符号位为1。
5. 补码: 正数的补码数值位为本身, 符号位为0; 负数的补码数值位为本身取反加1, 符号位为1。
6. 移码: 整数的移码为补码的符号位取反, 移码的符号位为0时表示负数, 移码的符号位为1时表示正数, 小数没有移码。
7. 模: 也称为模数, 例如4位二进制数整数的模为16, 二进制小数的模为2。
8. 定点数: 定点数的小数点是固定的; 定点小数, 小数点在符号位的右边; 定点整数, 小数点在最右边。
9. 浮点数: 浮点数的小数点位置是不固定的, 小数点位置可以浮动, 故称为浮点数。
10. 溢出: 包括正上溢、负上溢、正下溢、负下溢。
11. 精度溢出: 对于小数还存在精度的问题, 所有不在数轴上的小数都超出了定点小数所能表示的精度, 无法表示, 此时定点小数发生精度溢出, 只能采用舍入的方法近似表示。
12. 浮点数规格化: 尾数为纯小数, 且尾数的绝对值大于等于0.5、小于1, 即尾数的最高有效位为1（尾数用原码表示）。非规格化的浮点数可以通过左规或右规的方法变为规格化的浮点数。
13. 隐藏位: 当尾数采用原码表示时, 规格化的尾数最高有效位一定是1, 可以将最高有效位的1隐藏, 从而节省1位存储空间, 被隐藏的这一位称为隐藏位。

14. BCD码: Binary Coded Decimal, 二进制编码的十进制数
15. 有权码: 十进制数的4位二进制数码的每一位都有确定的权值。8421码是一种有权码。
16. 无权码: 十进制数的4位二进制数码的每一位没有确定的权值。余3码是一种无权码。
17. BID码: Binary Integer Decimal, 十进制整数的二进制表示
18. DPD码: Densely Packed Decimal, 紧凑十进制编码
19. 二进制浮点数: 浮点数的阶码和尾数都用二进制数表示, 且阶码为移码, 尾数为小数。
20. 十进制浮点数: 浮点数的基数为10, 不是2; 尾数是定点整数, 不是定点小数。
21. ASCII码: American Standard Code for Information Interchange, 美国信息交换标准代码, 用7位二进制数表示128个字符。
22. 机内码: 汉字机内码是计算机内部存储、处理加工和传输汉字时所用的统一编码(如国标码、区位码、Unicode编码等)。
23. 字形码: 字形码是汉字的输出码, 也称字型码。
24. 字库: 也称汉字库, 存放汉字字形码的字库。
25. 码距: 码距(又称海明距离), 两个编码对应二进制位不同的个数称为码距。
26. 校验码: 校验码是具有发现错误或纠正错误能力的数据编码。
27. 多重奇偶校验: 多重奇偶校验: 将原始数据分成若干个校验组, 每个数据位至少位于两个或两个以上的校验组, 当某个数据位出错时, 能在多个检错位中被指出(能够改变多个检错位), 从而可以知道是哪一位数据位出错。
28. ECC码: 海明码本质上是一种多重奇偶校验码, 它是一种既能检错也能纠错的校验码(Error-Correcting Codes, ECC)。
29. 海明码: 假设原始数据为: $D_k \dots D_2 D_1$, 共 k 位; 校验位为: $P_r \dots P_2 P_1$, 共 r 位; 则海明校验码为: $H_n \dots H_2 H_1$, 共 n 位, $n=k+r$, 也称 (n,k) 码
30. CRC码: 循环冗余校验(Cyclic Redundancy Check, CRC)是一种基于模2运算的校验码

• 2.2 选择题

- ① B: $(-126=1000\ 0010; -125=0101\ 1001; -32=1110\ 0000; -3=1111\ 1101)$
- ② A: $(si=usi=65535=FFFFH=-1)$
- ③ B: $(\text{unsigned short } x=65530=FFFAH, \text{ unsigned int } y = 0000\ FFAH)$
- ④ D: $(\text{short } si=-32767=1000,0000,0000,0001B \text{ (补码)}=8001H, \text{ unsigned short } usi = 8001H=32769)$
- ⑤ A: $(x=-8.25=-1000.01=1.00001 \times 2^3, E=127+3=130=10000010, M=00001, \text{ 单精度浮点数}=1\ 100\ 0001\ 0\ 000\ 0100\ 0000\ 0000\ 0000\ 0000 = C104\ 0000H)$
- ⑥ A: $(C640\ 0000H=1100\ 0110\ 0100\ 0000\ 0000\ 0000\ 0000\ 0000, \text{ 符号位}=1, E=1000\ 1100=140, M=100\ 0000\ 0000\ 0000\ 0000\ 0000, e=E-127=13, \text{ 单精度浮点数}=(1.1)_2 \times 2^{13} = -1.5 \times 2^{13})$
- ⑦ D: $(\text{单精度浮点数能表示的最大正数}: E=254, M=11111111111111111111111111111111 \text{ (23个1)}, \text{ 单精度浮点数}=2^{E-127} \times 1.M = 2^{127} \times (2-2^{-23}) = 2^{128} - 2^{104}; \text{ 教材表2.6和表2.7})$
- ⑧ A: $(\text{单精度浮点数最小规格化正数}: E=1, M=000000000000000000000000 \text{ (23个0)}, \text{ 单精度浮点数}=2^{E-127} \times 1.M = 2^{-126} \times (1.0)_2 = 1.0 \times 2^{-126}; \text{ 教材表2.6和表2.7})$
- ⑨ A: $(x=(f1)=CC90\ 0000H=1100\ 1100\ 1001\ 0000\ 0000\ 0000\ 0000\ 0000, \text{ 符号位}=1, E=1001\ 1001=153, M=001\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000, e=E-127=26, \text{ 单精度浮点数 } x = -(1.001)_2 \times 2^{26} = -1.125 \times 2^{13}; y=(f2)=B0C0\ 0000H=1011\ 0000\ 1100\ 0000\ 0000\ 0000\ 0000\ 0000, \text{ 符号位}=1, E=0110\ 0001=97, M=100\ 0000\ 0000\ 0000\ 0000\ 0000, e=E-127=-30, \text{ 单精度浮点数 } y = -(1.1)_2 \times 2^{-30} = -1.5 \times 2^{-30}; \text{ 故 } x < y, \text{ 且符合相同})$
- ⑩ B: (I: 整数转浮点再转回整数, 没有问题; II: 浮点转整数再转回浮点, 可能有问题; III: 单精度浮点转双精度浮点再转回单精度浮点, 没有问题; IV: 大数+小数-大数, 可能有问题)
- ⑪ C: $(\text{海明码}: k+r \leq 2^r-1; k=8, r=4; 8+4 \leq 2^4-1)$

• 2.3 回答下列问题:

① 为什么计算机中采用二进制进行数据表示和运算?

- 答: 二进制由于数码最少、容易与简单的物理状态对应、算术逻辑运算电路更容易实现等优势, 成为现代计算机中数据表示的不二之选, 采用二进制可以表示任何数据信息。

② 相对于奇偶校验, 交叉奇偶校验的检错与纠错能力的提高需要付出哪些方面的代价?

- 答: 交叉奇偶校验增加了校验位, 例如4个7位二进制数, 简单的奇偶校验只需要4位校验位(每一个1位), 而交叉奇偶校验需要12位校验位。

③ 为什么计算机中采用补码表示带符号的整数?

- 答: 0的补码是唯一的, 同样长度的二进制数整数, 补码的表示范围比原码多一个(如4位二进制数, 原码整数表示范围为-7~+7, 补码整数表示范围为-8~+7); 采用补码进行运算时, 符号位可以直接参与运算, 运算时符号位的进位作为模会被自动舍弃, 因此补码的减法运算可以转换成加法运算, 大大方便了二进制运算。

④ 浮点数的表示范围和精度分别由什么决定?

- 答: 浮点数的表示范围由阶码决定、精度由尾数决定。

⑤ 汉字输入码、机内码和字形码在汉字处理过程中各有何作用?

- 答: 汉字输入码就是使用英文键盘输入汉字时的编码; 汉字机内码是计算机内部存储、处理加工和传输汉字时所用的统一编码(如国标码、区位码、Unicode编码等); 汉字字形码是汉字的输出码, 也称字型码。

⑥ 在机内码中如何区分ASCII字符和汉字字符？

- 答：ASCII码的最高有效位是0，汉字字符的最高有效位是1。

⑦ 为什么现代处理器中又开始支持十进制浮点数运算？

- 答：二进制浮点数的最大问题是不能精确表示十进制数（例如 $(0.7)_{10} = (0.1011001100110011\dots)_2$ ），不精确的二进制浮点数表示会给运算带来很多误差问题，因此在财务金额运算中是不允许采用二进制浮点数的，必须使用十进制浮点数的表示和运算。

⑧ 如何识别浮点数的正负？浮点数能表示的数值范围和数值的精度取决于什么？

- 答：浮点数的第一位为符号位，该位是0表示正数，该位是1表示负数；浮点数的表示范围由阶码决定、精度由尾数决定。

⑨ 浮点数有两个0会带来什么问题？

- 答：IEEE754单精度浮点数有两个机器0，+0和-0；两个机器0使浮点数少了一个表示的值。

⑩ 简述CRC校验码的检错原理，CRC能纠错吗？

- 答：CRC编码：原始数据左移r位，然后与生成多项式相除（模2运算），得到余数，将原始数据和余数拼接起来就是CRC码。CRC解码：将接收到的CRC码与生成多项式相除（模2运算），得到余数，如果余数=0，则没有错误；如果余数不等于0，表示出错，并且根据余数的值，就可以知道是哪一位出错。因此CRC码可以发现并纠正错误。

- 2.4 写出下列各数的原码、反码和补码：

- 答：

真值	原码	反码	补码
0	0.00...0	0.00...0	0.00...0
-0	1.00...0	1.11...1	0.00...0
0.10101	0.10101	0.10101	0.10101
-0.10101	1.10101	1.01010	1.01011
0.11111	0.11111	0.11111	0.11111
-0.11111	1.11111	1.00000	1.00001
-0.10000	1.10000	1.01111	1.10000
0.10000	0.10000	0.10000	0.10000

- 2.5 已知数的补码表示形式，求数的真值：
- 答：此题目补码字长参看题目加上符号位共6位

补码	真值	补码	真值
$[x]_{\text{补}} = 0.10010$	$x = 0.10010$	$[x]_{\text{补}} = 1.10010$	$x = -0.01110$
$[x]_{\text{补}} = 1.11111$	$x = -0.00001$	$[x]_{\text{补}} = 1.00000$	$x = -1.00000$
$[x]_{\text{补}} = 0.10001$	$x = 0.10001$	$[x]_{\text{补}} = 1.00001$	$x = -0.11111$

- **2.6 给出在32位计算机中上述程序段的输出结果并分析原因：**

- **答：**

- **输出结果如下：**

x= 4294967295= -1;

u= 2147483648= -2147483648

- **因为x=-1，对应的补码是FFFF FFFFH，无符号数FFFF FFFFH对应的真值是4294967295**
 - **因为u= 2147483648，对应的机器码=8000 0000H，补码8000 0000H对应的真值为-2147483648**
 -

- 2.7 分析下列几种情况下所能表示的数据范围：

- 答：补码最小值可表示到： -2^{n-1}

- 1) 16 位无符号数：0 ~ 1111 1111 1111 1111，即 $0 \sim 2^{16}-1=65535$
- 2) 16 位原码定点小数：1.111 1111 1111 1111 ~ 0.111 1111 1111 1111，即 $-(1-2^{-15}) \sim 1-2^{-15}$
- 3) 16 位补码定点小数：1.000 0000 0000 0000 ~ 0.111 1111 1111 1111，即 $-1 \sim 1-2^{-15}$
- 4) 16 位补码定点整数：1000 0000 0000 0000 ~ 0111 1111 1111 1111，即 $-2^{15} \sim 2^{15}-1$

- **2.8**

- **答:**

- **机器码为8位，8位二进制数的模为256。**

- **2.9 用IEEE754 32位单精度浮点数标准表示下列十进制数。**
- **答：参考教材P27-30**
- **(1) $-(6+5/8)=-110.101=-1.10101 \times 2^2$, $E=127+2=129=1000\ 0001$, $M=101\ 0100\ 0000\ 0000\ 0000\ 0000$; 单精度浮点数= $1\ 1000\ 0001\ 101\ 0100\ 0000\ 0000\ 0000\ 0000$ =**C0D40000H****
- **(2) $3.1415927=3.243F6BH=11.0010\ 0100\ 0011\ 1111\ 0110\ 1011=1.1\ 0010\ 0100\ 0011\ 1111\ 0110\ 1011 \times 2^1$, $E=127+1=128=1000\ 0000$, $M=1\ 0010\ 0100\ 0011\ 1111\ 0110\ 11$ (最后两个11舍入, 并进位到上一位); 单精度浮点数= $0\ 1000\ 0000\ 1\ 0010\ 0100\ 0011\ 1111\ 0110\ 11$ =**40490FDBH****
- **(3) $64000=11111010000000000=1.1111010000000000 \times 2^{15}$, $E=127+15=142=1000\ 1110$, $M=111\ 1010\ 0000\ 0000\ 0000\ 0000$; 单精度浮点数= $0\ 100\ 0111\ 0\ 111\ 1010\ 0000\ 0000\ 0000\ 0000$ =**477A0000H****

- **2.10 求与单精度浮点数43490000H对应的十进制数**
- **答：首先展开成二进制形式，之后按照IEEE754标准逐个对应**
 - **43940000H=0100 0011 1001 0100 0000 0000 0000 0000，E=10000111=135，M=001 0100 0000 0000 0000 0000，浮点数=1.Mx2^{E-127}=1.00101x2⁸=10010 1000=296**

- 2.11

- 答:

- 单精度浮点数能表示的最大数（最大正数）：E=254，e=254-127=127，M=11...11，1.M=(2-2⁻²³)，fmax = $2^{127} \times (2 - 2^{-23})$
- 单精度浮点数能表示的最小数（最小负数）：为最大数的负数，fmin = -fmax = $-2^{127} \times (2 - 2^{-23})$
- 单精度浮点数能表示的最小正数：E=1，e=1-127=-126，M=00...00，1.M=1，最小正数= $2^{-126} \times 1.0$
- 单精度浮点数能表示的最大负数：为最小正数的负数，最大负数= $-2^{-126} \times 1.0$
-

- **2.12**

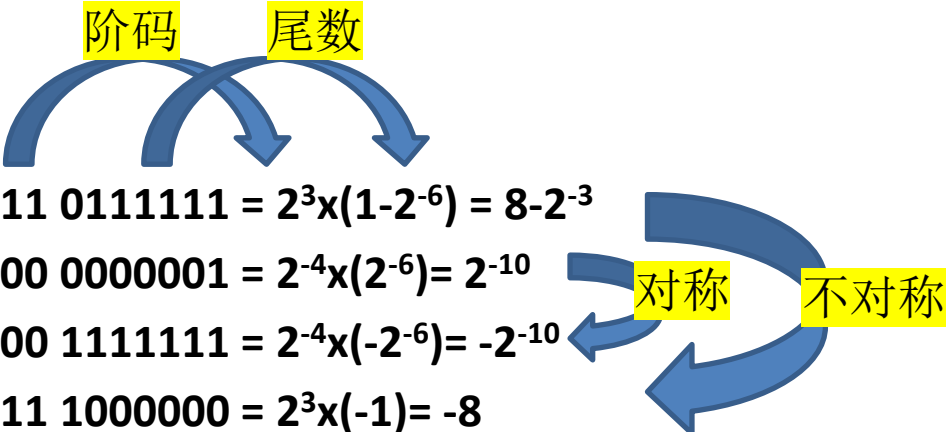
- 答:

- (1) 有可能, 例如, $N_1=2^3 \times 0.1=100=4$; $N_2=2^4 \times 0.001=10=2$; 此时 $m < n$, 但是 $N_1 > N_2$
- (2) 不可能, 因为 M_1 和 M_2 是规格化数, 则 M_1 和 M_2 都在 $[0.5, 1)$ 范围内, 即 M_1 和 M_2 不可能相差 2 倍或 2 倍以上; 若 $m < n$, 则两个数的阶码会相差 2 倍或 2 倍以上; 因此, 若 $m < n$, 一定有 $N_1 < N_2$

- **2.13** 设二进制浮点数的阶码为3位，尾数7位。用模2补码写出它们所能表示的最大正数，最小正数，最大负数和最小负数，并将之转换为十进制数。

• 答：

- 阶码为3位（1位阶符）：-4~3；1,00~0,11
- 尾数为7位（1位尾符）：-1 ~ +(1-2⁻⁶)；1.000000 ~ 0.111111

- 
- The diagram illustrates the bit patterns for the maximum and minimum positive and negative numbers. It shows that the representation is asymmetric around zero. The bit patterns are: 011 0111111 for the maximum positive number, 100 0000001 for the minimum positive number, 100 1111111 for the maximum negative number, and 011 1000000 for the minimum negative number. Arrows indicate the mapping from the bit patterns to the values. The labels '阶码' (Exponent) and '尾数' (Mantissa) are shown above the bit patterns. The labels '对称' (Symmetric) and '不对称' (Asymmetric) are shown next to the values.
- 最大正数=011 0111111 = $2^3 \times (1-2^{-6}) = 8-2^{-3}$
 - 最小正数=100 0000001 = $2^{-4} \times (2^{-6}) = 2^{-10}$
 - 最大负数=100 1111111 = $2^{-4} \times (-2^{-6}) = -2^{-10}$
 - 最小负数=011 1000000 = $2^3 \times (-1) = -8$

- **2.14**

- 答:

- $57/128 = 0.0111001 = 0.111001 \times 2^{-1}$; 阶码=-1=1,111; 尾数=0.111001000; 57/128=1111 0111001000

- $-69/128 = -0.1000101 = -0.1000101 \times 2^0$; 阶码=0=0,000; 尾数=1.011101100; -69/128=0000 1011101100

- **2.15**

- **答：**

- 原始数据：**01011011**
- 奇校验码：**01011011 0**
- 偶校验码：**01011011 1**

- 接收到的数据=**01011010**（最后一位出错了）

- 如果是奇校验，则接收到的奇校验码=**01011010 0**，此时**1**的个数是偶数个，说明出错了（假设只有**1**位出错）

- 如果是偶校验，则接收到的奇校验码=**01011010 1**，此时**1**的个数是奇数个，说明出错了（假设只有**1**位出错）

- 2.16 已知表2.27交叉校验表求：

- 则X1,X2,X3,X4处的比特分别是？
- 则X5,X6,X7,X8处的比特分别是？
- 则X9,X10,X11,X12处的比特分别是？
- 则Y1,Y2处的字符分别是？

表2.27 ASCII交叉校验

字符	7位ASCII码							HP
3	0	X1	X2	0	0	1	1	0
Y1	1	0	0	1	0	0	X3	1
+	X4	1	0	1	0	1	1	0
Y2	0	1	X5	X6	1	1	1	1
D	1	0	0	X7	1	0	X8	0
=	0	X9	1	1	1	X10	1	1
VP	0	0	1	1	1	X11	1	X12

- 2.16

- 答：

- 3的ASCII码=33H=011 0011，因此X1=1、X2=1
- +的ASCII码=2BH=010 1011，因此X4=0
- D的ASCII码=44H=100 0100，因此X7=0、X8=0
- =的ASCII码=3DH=011 1101，因此X9=1、X10=0

字符	7位ASCII码							HP
3	0	1	1	0	0	1	1	0
Y1	1	0	0	1	0	0	X3	1
+	0	1	0	1	0	1	1	0
Y2	0	1	X5	X6	1	1	1	1
D	1	0	0	0	1	0	0	0
=	0	1	1	1	1	0	1	1
VP	0	0	1	1	1	X11	1	X12

- 2.16

- 答：

- 因为是偶校验，因此X5=1，X6=0，X11=1，X3=1，X12=1
- Y1=100 1001=49H，为I的ASCII码，Y1=I；
- Y2=011 0111=37H，为7的ASCII码，Y2=7

字符	7位ASCII码							HP
3	0	1	1	0	0	1	1	0
I	1	0	0	1	0	0	1	1
+	0	1	0	1	0	1	1	0
7	0	1	1	0	1	1	1	1
D	1	0	0	0	1	0	0	0
=	0	1	1	1	1	0	1	1
VP	0	0	1	1	1	1	1	1

- 2.17 设8位有效信息为01101110，试写出它的海明校验码，给出过程，说明分组检测方式，并给出指错字以及逻辑表达式。如果接收方收到的有效信息变成01101111，说明如何定位错误并纠错。

• 答：

— 原始数据=01101110=D1D2D3D4D5D6D7D8，k=8，r=4，n=k+r=12

— 海明码校验位：

- $P1=D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 1$
- $P2=D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 1$
- $P3=D2 \oplus D3 \oplus D4 \oplus D8 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$
- $P4=D5 \oplus D6 \oplus D7 \oplus D8 = 1 \oplus 1 \oplus 1 \oplus 0 = 1$

— 海明码=P1P2D1P3D2D3D4P4D5D6D7D8=110011011110（校验位P位于 2^n ，数据位D顺次排列）

— 接收到的数据=01101111（D8出错）

— 接收到的海明码=110011011111（D8出错）

— 海明码检错位：

- $G1=P1 \oplus D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7 = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 0$
- $G2=P2 \oplus D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7 = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 0$
- $G3=P3 \oplus D2 \oplus D3 \oplus D4 \oplus D8 = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 1$
- $G4=P4 \oplus D5 \oplus D6 \oplus D7 \oplus D8 = 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1$

— $G4G3G2G1=1100$ ，表示第12位出错，即D8位出错（从左往右数）

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	
	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
	P1	P2	D1	P3	D2	D3	D4	P4	D5	D6	D7	D8	D9	D10	D11	
	√		√		√		√		√		√		√		√	
		√	√			√	√			√	√			√	√	
				√	√	√	√					√	√	√	√	
								√	√	√	√	√	√	√	√	

每个校验位P负责检测的小组如下（参考唐朔飞老师PPT）：

$$P1 = D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7 \oplus D9 \oplus D11$$

$$G1 = P1 \oplus D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7 \oplus D9 \oplus D11$$

$$P2 = D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7 \oplus D10 \oplus D11$$

$$G2 = P2 \oplus D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7 \oplus D10 \oplus D11$$

$$P3 = D2 \oplus D3 \oplus D4 \oplus D8 \oplus D9 \oplus D10 \oplus D11$$

$$G3 = P3 \oplus D2 \oplus D3 \oplus D4 \oplus D8 \oplus D9 \oplus D10 \oplus D11$$

$$P4 = D5 \oplus D6 \oplus D7 \oplus D8 \oplus D9 \oplus D10 \oplus D11$$

$$G4 = P4 \oplus D5 \oplus D6 \oplus D7 \oplus D8 \oplus D9 \oplus D10 \oplus D11$$

- 2.18

- 答:

- 原始数据=1001
- 生成多项式=1101, $r=3$
- 将原始数据左移 r 位得: 1001 000
- 用1001 000 除以生成多项式1101 (模2运算), 得到余数=011
- CRC码=1001 011
- 接收到的数据=1101
- 接收到的CRC码=1101 011
- 用1101 011 除以生成多项式1101 (模2运算), 得到余数=011
- 根据余数011, 查表知道是第2位出错 (从左往右数)

$$\begin{array}{r}
 1101 \overline{) 1001000} \\
 \underline{1101} \\
 1000 \\
 \underline{1101} \\
 1010 \\
 \underline{1101} \\
 1110 \\
 \underline{1101} \\
 011
 \end{array}$$

$$\begin{array}{r}
 1101 \overline{) 1101011} \\
 \underline{1101} \\
 0000 \\
 \underline{0000} \\
 0001 \\
 \underline{0000} \\
 0011 \\
 \underline{0000} \\
 011
 \end{array}$$

Thanks