1853790 庄镇华

习题 2

2.3 根据图 2.12 的四位超前进位加法器，请回答：

1. 从 X、Y 输入到 F 输出需要多长时间（假设异或门延迟 20ns,其他门为 10ns）?

答：由 Xi,Yi 产生所有的 Gi,Pi 需要 10ns,产生 Xi,Yi 异或需要 20ns,
由 Gi,Pi 产生所有的 Ci 需要 20ns,
最后产生 Fi 需要 20ns,
因此总共需要 50ns。

2. 如果扩大到 8 位，直接产生超前进位信号，将产生什么问题?

答：如果扩大到 8 位，直接产生超前进位信号，会导致处理最后几位的门电路输入端过多，输入的某个信号稍有波动就会产生误差。

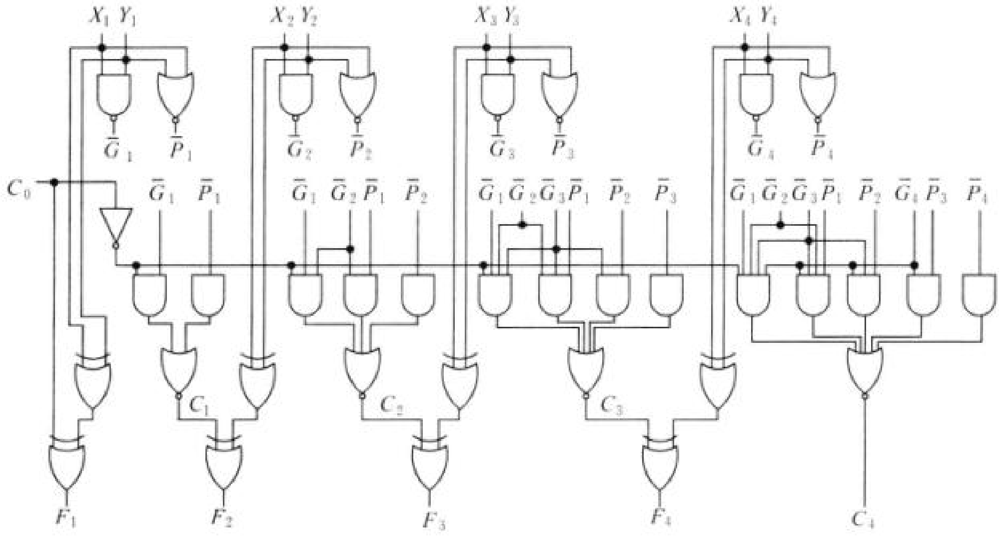
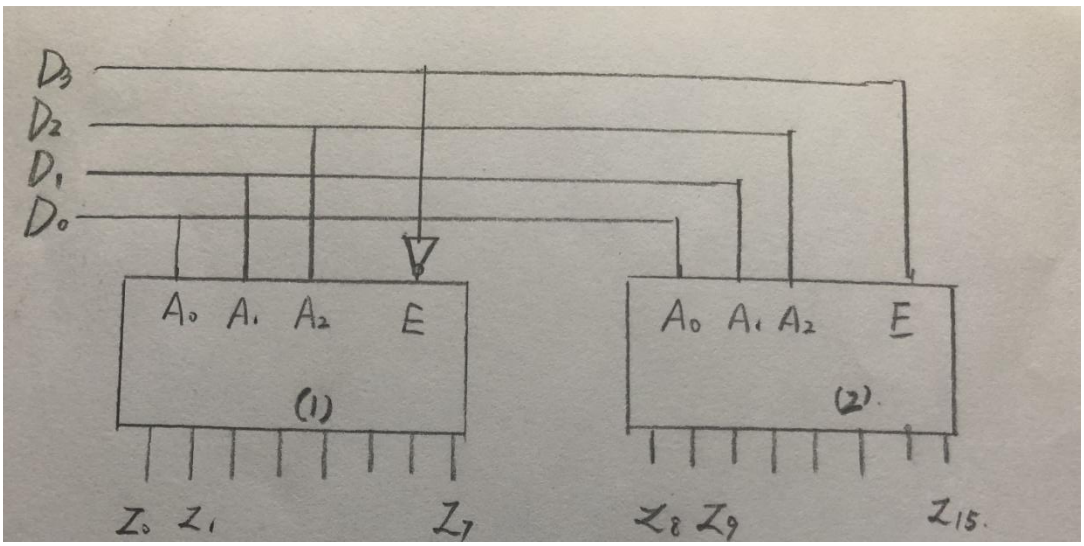


图 2.12 四位超前进位加法器

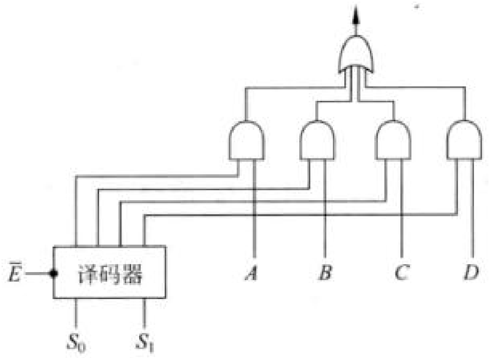
2.5

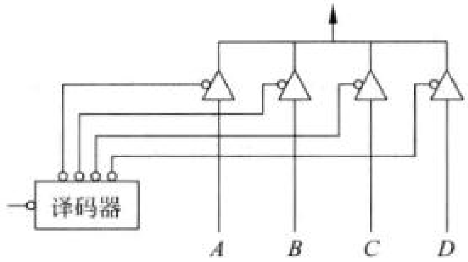
画出逻辑图：用 2 个有 3 输入 8 输出译码器功能的芯片组成具有 16 输出的译码器。



2.6

把 4 个寄存器的输出送到某一组输出线上，可使用四选一多路选择器，也可使用三态门。用四选一实现和用三态门实现，对开门信号的要求有什么差别？





答：对于四选一实现，如果输入 S0 和 S1 之间的信号有延迟，或者译码器内部电路的不一致性，译码器的输出端会产生一些毛刺，但时间很短，很快会稳定。如果需要去除毛刺，可以在译码器上加入译码允许信号 E，当 E 位于高电平时，译码器的 4 个输出均为低电位。

对于三态门实现，要求三态门的开门信号不能同时有 2 个或者 2 个以上处于低电平。

2.7

设 A 为锁存器，B 为 D 触发器，设输入信号和触发信号关系如下图所示，画出输出端波形图（设 A、B 原状态均为 0）

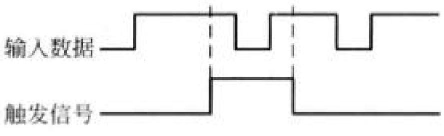
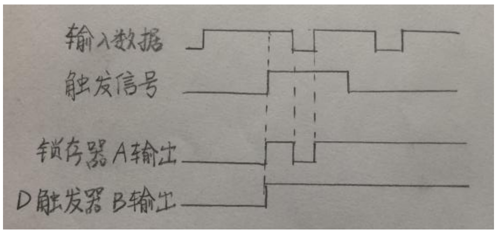


图 2.16 输入数据和触发信号的时序



2.8

使用 JK 触发器构成 4 位二进制同步计数器，请写出 J、K 表达式，为什么这些表达式比 2.2.3 节列出的 1 位十进制同步计数器表达式简单？

答：设 4 位二进制数从低到高：QD,QC,QB,QA

则

$$Q_D = \overline{Q_D}, J_D = 1, \overline{K_D} = 0$$

$$Q_C = \overline{Q_D}Q_C + \overline{Q_C}Q_D, J_C = Q_D, \overline{K_C} = \overline{Q_D}$$

$$Q_B = (\overline{Q_C} \oplus \overline{Q_D})Q_B + \overline{Q_B}Q_CQ_D, J_B = Q_CQ_D, \overline{K_B} = (\overline{Q_C} \oplus \overline{Q_D})$$

$$Q_A = Q_A(\overline{Q_B} + \overline{Q_B}Q_C + \overline{Q_B}\,\overline{Q_C}Q_D) + \overline{Q_A}Q_BQ_CQ_D, J_A = Q_BQ_CQ_D, \overline{K_A} = (\overline{Q_B} + \overline{Q_B}Q_C + \overline{Q_B}\,\overline{Q_C}Q_D)$$

因为 JK 触发器更适用于二进制的计数，计数范围是满的，不会从 1001 直接跳跃到 0000。

习题 3

3.7 已知 X=0.1011，Y=-0.0101，试求：
[X]补，[-X]补，[Y]补，[-Y]补，[X/2]补，[X/4]补，[2X]补，[Y/2]补，[Y/4]补，[2Y]补，[-2Y]补
答：[X]补=0.1011
[-X]补=1.0101
[Y]补=1.1011
[-Y]补=0.0101
[X/2]补=0.0101
[X/4]补=0.0010
[2X]补为溢出
[Y/2]补=1.1101
[Y/4]补=1.1110
[2Y]补=1.0110
[-2Y]补=0.1010
注意判别数据是否溢出

3.8
设十进制数 X=（+128.75）x2^{-10}
（1）若（Y）$_2$=X，用定点数表示 Y 值
（2）设用 21 个二进制位表示浮点数，阶码用 5 位，其中阶符用 1 位，尾数用 16 位，其中符号用 1 位。阶码的基数为 2，写出阶码和尾数均用原码表示的 Y 的机器数。
（3）写出阶码和尾数均用反码表示 Y 的机器数
（4）写出阶码和尾数均用补码表示 Y 的机器数
答：（1）0.001000000011
（2）阶码：10010 尾数：0.100000001100000
（3）阶码：11101 尾数：0.100000001100000
(4)阶码：11110 尾数：0.100000001100000

3.9
设机器字长 16 位。定点表示时，数值 15 位，符号位 1 位，浮点表示时，阶码 6 位，其中阶符 1 位，尾数 10 位，其中数符 1 位，阶码的基数为 2.试求
（1）定点原码整数表示时，最大正数和最小负数各是多少？
（2）定点原码小数表示时，最大正数和最小负数各是多少？
（3）浮点原码表示时，最大浮点数和最小浮点数各是多少？
绝对值最小的呢？（非 0）估算表示的十进制值的有效数字位数
答：（1）最大正数：2^15-1
最小负数：1-2^15
绝对值最小为 1
（2）最大正数：1-2^(-15)
最小负数：2^(-15)-1

(3) 最大浮点数：(1-2^-9)x2^31
(4) 最小浮点数：(2^-9-1)x2^31
绝对值最小的浮点数：2^(-9)x2^(-31)=2^(-40)
有效数字位数：2^(-9)=10^(-e) e=3

3.10
设机器字长16位，阶码7位，其中阶符1位，尾数9位，其中数符1位（阶码的基数为2）。若阶码和尾数均用补码表示，说明在尾数规格化和不规格化两种情况下，它所能表示的最大正数，非零最小正数，绝对值最大的负数以及绝对值最小的负数各是那几个数？写出机器数，并给出十进制值（不采用隐藏位）。若阶码用移码，尾数仍用补码，上述各值有变化吗？若有变化，请列出。
答：
（1）阶码和尾数均用补码：

尾数规格化：
最大正数：(1-2^-8)x2^63
非零最小正数：2^(-65)
绝对值最大的负数：-2^63
绝对值最小的负数：-(2^-1+2^-8)x2^(-64)

尾数不规格化：
最大正数：(1-2^-8)x2^63
非零最小正数：2^(-72)
绝对值最大的负数：-2^63
绝对值最小的负数：-2^(-72)

（2）阶码用移码，尾数仍用补码：
若用移码，上述各值无影响，但是考虑到下溢，当阶为-64，就认为是下溢，也把尾数也置成全零，化为机器零，所以不能表示阶为-64的那些数。

尾数规格化：
非零最小正数：2^(-64)
绝对值最小的负数：-(2^-1+2^-8)x2^(-63)

尾数不规格化：
非零最小正数：2^(-71)
绝对值最小的负数：-2^(-71)

3.11
按下列要求设计一个尽可能短的浮点数格式（阶的基数取2）
(1) 数值范围为 $1.0x10^{+-38}$
(2) 有效数字为十进制7位
(3) 0的机器数为全0

答：10^38 约为 2^127,所以阶码取 8 位，其中符号位 1 位。10^-7 约为 2^-24，尾数取 24 位，另加一个符号（数符）阶码采用移码，才能使全 0 为 0 的机器数。
阶码：0-7
数符：8
尾数：9-32

3.21
设浮点数 X、Y 的阶码（补码形式）和尾数（原码形式）如下：
X：阶码 0001，尾数 0.1010；Y：阶码 1111，尾数 0.1001.设基数为 2.
（1）求 X+Y（阶码运算用补码，尾数运算用补码）
（2）求 XY（阶码运算用移码，尾数运算用原码一位乘）
（3）求 X/Y（阶码运算用移码，尾数运算用原码加减交替法）
（1）对齐：Y：阶码 0001 尾数 0.0010
X+Y 阶码 0001 尾数 0.1100
（2）XY 阶码补码 0000 移码 1000

00.0000	0.1001	
+00.1010		+X
00.1010		
00.0101	0.0100	右移
00.0010	1.0010	右移
00.0001	0.1001	右移
+00.1010		+X
00.1011		
00.0101	1.0100	右移

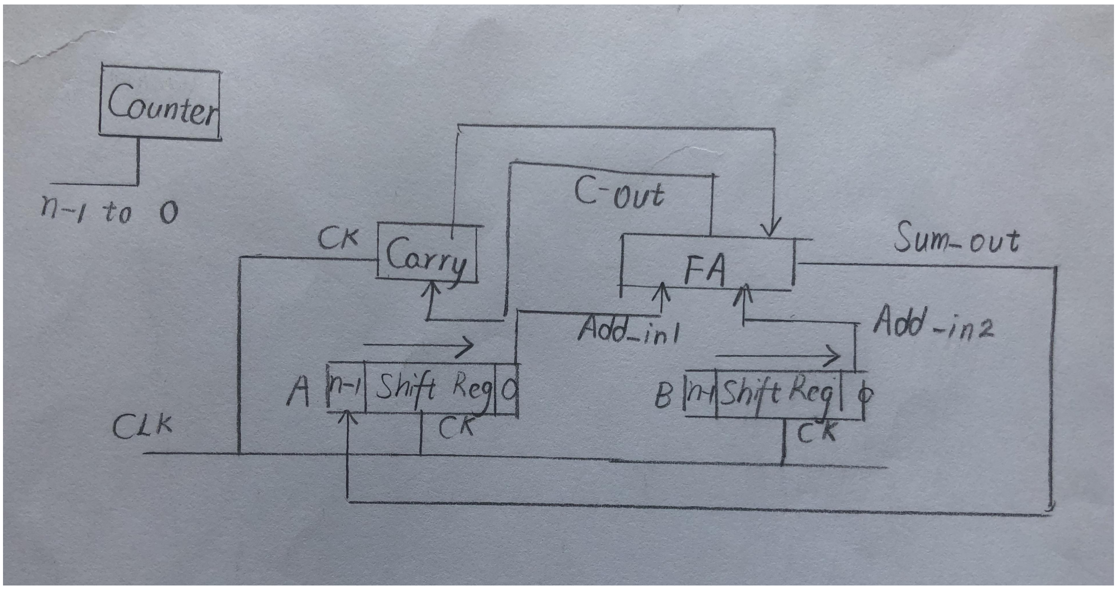
左规：尾数：0.10110100 阶：0111
舍入：尾： 0.1011 阶：0111
（3）X/Y 阶码补码 0010 移码 1010
[Y] 补=0.1001 [-Y]补=1.0111

00.1010	00000	+[-Y]补
+11.0111		
00.0001		商 1
00.0001	00001	
00.0010	00010	左移
+11.0111		+[-Y]补
11.1001		商 0
11.1001	00010	
11.0010	00100	左移
+00.1001		+[Y]补
11.1011		商 0
11.1011	00100	
11.0110	01000	左移
+00.1001		+[Y]补
11.1111		商 0
11.1111	01001	

11.1110　　10010　　左移
+00.1001　　　　　　+[Y]补
00.0111　　　　　　商 1
00.0111　　10011
右规：尾数：0.10011 阶码：1011
舍入：尾 0.1001
不溢出
阶码：1011 尾数：0.1001

3.22
浮点加减乘除运算各在什么情况下会发生溢出？
答：阶码溢出的情况下

3.24
现有一串行加法器，计算两个 n 位数据之和（不带符号位），已知相加两数存放在 A,B 寄存器中，请画出能实现（A）+（B）-> A 的逻辑图。图中只准用一个一位加法器，逐位进行计算。
答：分析：
一位加法器：各位串行计算
寄存器要有移位功能
i 位进位和 i+1 位的操作数一起计算 全加器
n 位数据加法 使用计数器确定加法是否完成
解答：



3.25
如果采用偶校验，下述两个数据的校验位的值是什么？
（1）0101010
（2）0011011
答：（1）1 （2）0

3.26

设有 16 个信息位，如果采用海明校验，至少需要设置多少个校验位？应放在哪些位置上？

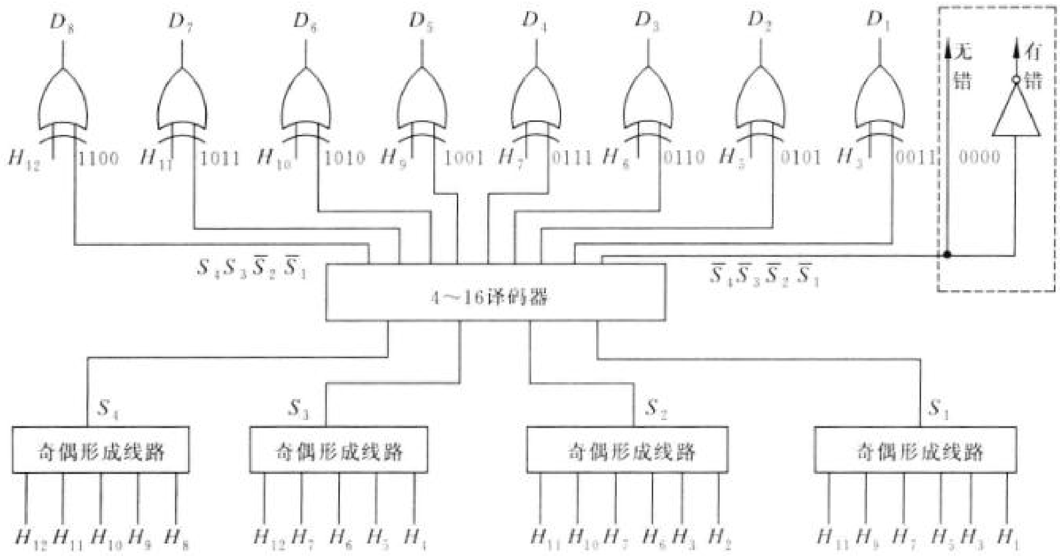
答：2^(r-1) >= 16 + r，r = 6

分别排列在海明码字的第 1、2、4、8、16、22 位

3.27

设有 8 位有效信息，试为之编写海明校验线路。说明编码方法，并分析所选方案具有怎样的检错与纠错能力。若 8 位信号为 01101101，海明码是何值？

答：2^(r-1) >= 8 + r, r = 5



校验位分别排列在海明码字的第 1、2、4、8、13 位

P5 D8 D7 D6 D5 P4 D4 D3 D2 P3 D1 P2 P1

H13 H12 H11 H10 H9 H8 H7 H6 H5 H4 H3 H2 H1

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

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

$$P3 = D2 \oplus D3 \oplus D4 \oplus D8$$

$$P4 = D5 \oplus D6 \oplus D7 \oplus D8$$

能检测与自动校正一位错，并发现两位错。

01101101 的海明码是 1 0110 0110 0111

3.28

现有 4 个数：00001111、11110000、00000000 和 11111111，请回答：

（1）其码距为多少？最多能纠正或发现多少位错？如果出现数据 00011111，应纠正成什么数？当已经知道出错位时如何纠正？

（2）如果再加上两个数 00110000 和 11001111（共 6 个数），其码距为多少？能纠正或发现多少位错？

答：（1）四个数之间有四位不同，故码距为四，能纠正一位错和发现两位错。

如果出现 00011111，纠正为 00001111，方法是取出错位的反码

（2)码距是根据数据中任意两个合法码之间最少有几位二进制位不同而确定的。
00110000 和 00000000 以及 11110000 都仅有 2 位不同，因此码距为 1。
能发现 1 位错误，不能纠正错误。
注意：当码距 d 为奇数时，可检查 d-1 个错，纠正（d-1）/2 位错；
码距 d 为偶数时，可检查 d/2 位错，纠正 d/2-1 位错。

3.29

现有 4 位二进制数，请回答：
（1）如果是无符号数，能表示的数据个数是多少？
（2）如果内有 1 位符号位，则用原码、补码或反码表示时，所表示的数据个数各是多少？

答：（1）16
（2）原码：15 补码：16 反码：15

3.30

规格化浮点数阶码有 P 位，尾数有 M 位，各自包含 1 位符号位，且都用补码表示，该数的基数为 2，请说出用这种格式能表示的全部不同的规格化数的总个数。当基数为 8 时，又能表示多少个规格化数（不考虑机器零）？

答：对于同一阶码，尾数能表示的个数为 2^M 个
当 r=2 时，能表示的规格化数为 1/2*2^M=2^(M-1)。阶码的个数为 2^P,因此能表示的全部规格化浮点数为 2^（P+M-1）个。
当 r=8 时，浮点数尾数前三位不全为零的数为规格化数。同一阶码能表示的规格化数为 7/8*2^M，因此能表示的全部规格化浮点数为 7/8*2^（P+M）个。