



中国科学技术大学
University of Science and Technology of China

《数字电路》习题课

一、数字电路基础

黄慎宜

2025年11月19日

School of Microelectronics, University of Science and Technology of China

- ◆ 本习题课分为**考点总结**、**习题讲解**和**补充题**三部分
- ◆ **考点总结**部分会给出和该页讲解内容直接相关的题目，供大家熟悉知识点（部分题目会给出解答）
- ◆ **习题讲解**部分为一些习题的详细解答过程
- ◆ **补充题**大多为一些有意义的开放讨论题和往年真题，供大家自行复习，会以独立的文档给出简答

注1：本习题课中的习题大多选自历年期中期末真题和考研真题，练习价值很大

注2：期中期末真题选自《数字电路》、《数字逻辑电路》和《模拟与数字电路》三个课程的考试真题，由于三门课内容几乎一致，在备注的时候不做额外区分

◆ **考点总结**

◆ **习题讲解**

◆ **补充题**

1. 数制转换

□ 十进制转换为二进制

➤ 整数除2取余倒数；小数乘2取整正序

□ 二进制转换为十进制

□ 十进制转换为八进制、16进制，可以**通过转换二进制过渡**

□ 注意**有效数字**问题

2. 二进制运算

□ 均通过转换为二进制加法运算实现

➤ 减法：**加上补码**

➤ 乘法：移位相加

Q1：将 $(12.6)_{10}$ 转换为二进制，要求二进制数保留小数点以后4位有效数字

Q2：将 $(1010111.0101)_2$ 转换为十进制和十六进制

Q3：如果一个等式 $325 + 42 = 411$ 成立，请问该等式采用的是几进制



3. 二进制补码运算

- 首先根据运算的操作数和结果的最大值确定补码的位数
- 结果验证：补码可以看作一种加权码，与一般二进制码的区别在于最高位的权值要加上负号（为什么？）

★ 原码、反码、补码的概念

符号位为 0（正数）→ 三者无差异

符号位为 1（负数）→ 反码：符号位不变，数值位按位取反
补码：反码+1

注意：0 的补码为 0

Q4：用二进制补码完成减法
 $(21)_{10} - (5)_{10}$

Q5：将下列十进制数转换为 5 位二进制补码，并进行减法运算，指出 5 位结果是否会产生溢出？

① $(30)_{10} - (9)_{10}$

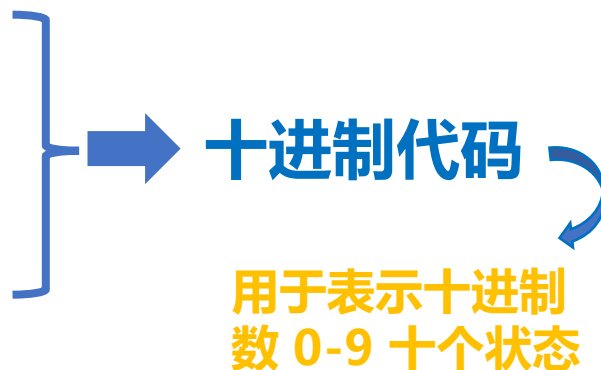
② $(18)_{10} - (12)_{10}$

③ $(-16)_{10} - (21)_{10}$



4. 常用编码及其转换

- 8421码 (BCD代码)
- 余3码
- 余3循环码
- 格雷码



□ 二进制码 -> 格雷码

- ✓ 格雷码的最高位和二进制码最高位相同
- ✓ 从左到右，逐一将二进制码相邻两位异或，作为格雷码下一位

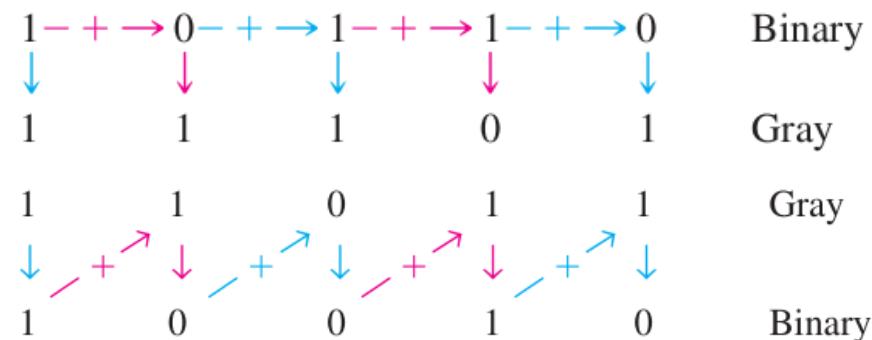
□ 格雷码 -> 二进制码

- ✓ 二进制码的最高位和格雷码最高位相同
- ✓ 从左到右，逐一将产生的二进制码和下一位相邻的格雷码异或，作为二进制码下一位

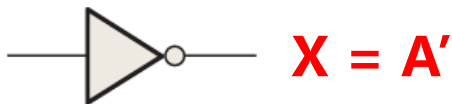
Q6: 格雷码的特点是?

Q7:

- ① 将二进制码10110转换为格雷码
- ② 将格雷码11011转换为二进制码



1. 基本逻辑运算



Inverter truth table.

Input		Output
A		X
LOW (0)		HIGH (1)
HIGH (1)		LOW (0)



Truth table for a 2-input AND gate.

Inputs		Output
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1



Truth table for a 2-input OR gate.

Inputs		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1



Truth table for a 2-input NAND gate.

Inputs		Output
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0



Truth table for a 2-input NOR gate.

Inputs		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0



Truth table for an exclusive-OR gate.

Inputs		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

$$X = A \oplus B$$



Truth table for an exclusive-NOR gate.

Inputs		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

$$X = A \odot B$$

2. 逻辑函数式的化简

① 公式法化简

1. $A + 0 = A$

2. $A + 1 = 1$

3. $A \cdot 0 = 0$

4. $A \cdot 1 = A$

5. $A + A = A$

6. $A + \bar{A} = 1$

7. $A \cdot A = A$

8. $A \cdot \bar{A} = 0$

9. $\bar{\bar{A}} = A$

10. $A + AB = A$

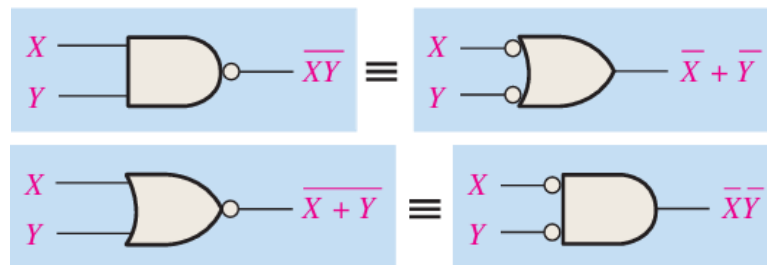
11. $A + \bar{A}B = A + B$

12. $(A + B)(A + C) = A + BC$

补充: $AB + A'C + BC = AB + A'C$

★ 德摩根 (DE Morgan's) 定律:

$$\begin{cases} (A + B)' = A'B' \\ (AB)' = A' + B' \end{cases}$$



Q1: 证明公式11、12, 以及补充公式

Q2: (2016•期末) 将 $Y = AC + B'C + BD' + CD' + A(B + C') + A'BCD' + AB'DE$ 化简为最简与或式

Q3: (2021•期末) 将 $Y = (AB'C'D + AC'DE + B'DE + AC'D'E)'$ 化简为最简与或式



2. 逻辑函数式的化简

② 逻辑代数基本定理

★ **反演定理**。对逻辑式 Y ，将其中 \cdot 和 $+$ 互换， 0 和 1 互换，**原变量和反变量互换**，则得到的结果为 Y'

注意：

- a. 仍需遵循“先括号、然后乘、最后加”的运算优先次序
- b. 不属于单个变量上的反号应该保留不变

例：

$$Y = A(B + C) + CD$$
$$\Rightarrow Y' = (A' + B'C')(C' + D')$$

★ **对偶定理**。逻辑式中 \cdot 和 $+$ 互换， 0 和 1 互换
若两逻辑式相等，则它们的对偶式也相等

Q4：利用对偶定理证明上页中基本公式12

Q5：若 $Y = ((AB' + C)' + D)' + C$ ，利用反演定理求 Y'



2. 逻辑函数式的化简

③ 不同形式逻辑函数式之间的转换

- a. 最简**与或**式
- b. 最简**或与**式
- c. **与非-与非**形式
- d. **或非-或非**形式

.....

➤ 要想得到c或d的形式，一般先将逻辑表达式转化为a或b的形式，然后利用反演定理处理

例：利用2输入端**与非**门产生如下逻辑函数

$$Y = AC + BC'$$

【解】 $Y = AC + BC' = ((AC + BC')')'$
 $= ((AC)'(BC')')'$

核心思想：利用反演定理 $Y = (Y')'$

Q6：证明公式11、12，以及补充公式

Q7：将下列十进制数转换为5位二进制补码，并进行减法运算，指出5位结果是否会产生溢出？

- ① $(30)_{10} - (9)_{10}$
- ② $(18)_{10} - (12)_{10}$
- ③ $(-16)_{10} - (21)_{10}$



2. 逻辑函数式的化简

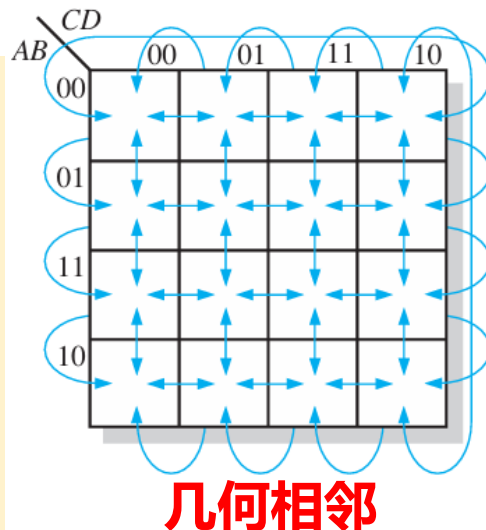
④ 卡诺 (Karnaugh) 图化简法

- 将函数化为**最小项之和**的形式
- 画出表示该逻辑函数的**卡诺图**
- 找出可以合并的最小项
- 选取化简后的乘积项

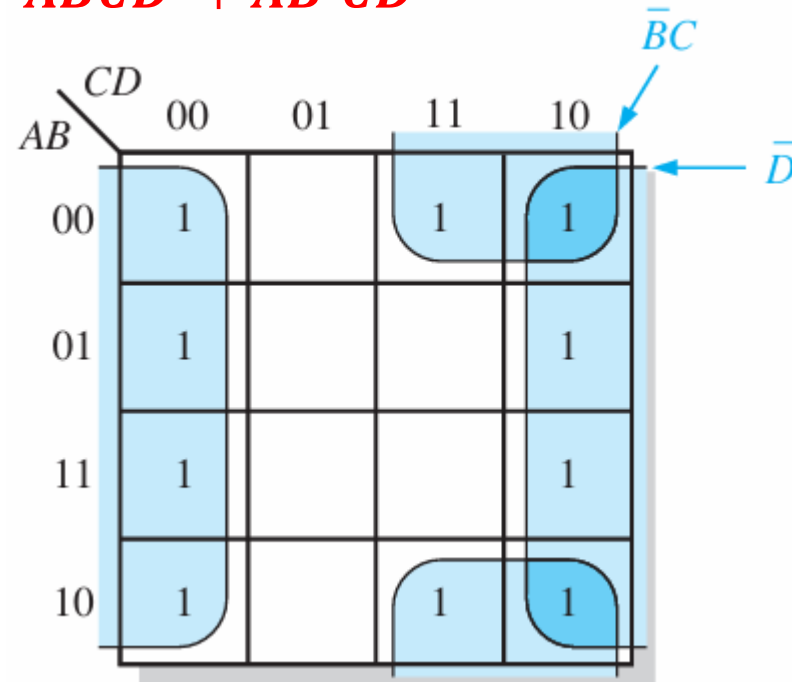
选取原则:

- ✓ 覆盖卡诺图中**所有的1**
- ✓ 圈出的矩形**数目最少**
- ✓ 圈出的矩形**尽可能大**
 - 每个乘积项包含的因子最少

核心思想：利用几何相邻表征逻辑相邻



Q8: 化简 $Y = B'C'D' + A'BC'D' + ABC'D' + A'B'CD + AB'CD + A'B'CD' + A'BCD' + ABCD' + AB'CD'$



2. 逻辑函数式的化简

⑤ 最小项之和与最大项之积

最小项：所有变量均出现且仅取一种取值组合为**1**的**积项**

e.g.:

三变量最小项 $A'BC'$ ，该最小项为1时 $A = 0, B = 1, C = 0$ ，对应的十进制数为2，用 m_2 表示

最大项：所有变量均出现且仅取一种取值组合为**0**的**和项**

e.g.:

三变量最大项 $A' + B + C$ ，该最大项为0时 $A = 1, B = 0, C = 0$ ，对应的十进制数为4，用 M_4 表示

Q9：将逻辑函数 $Y = AB'C'D + A'CD + AC$ 展开为最小项之和的形式

Q10：将逻辑函数 $Y = A'B + AC$ 展开为最大项之积的形式

➤ 提示：
使用基本公式
 $A + BC = (A + B)(A + C)$



2. 逻辑函数式的化简

⑥ 无关项

约束项

函数值始终为0
的项

任意项

函数值是0或1均可，不
会影响电路功能
(不完全定义的逻辑函数)

在逻辑函数式中可以写入也可以删除

可用于化简逻辑函数，在卡诺图中用×表示：

- 既可以当作1，也可以当作0
- 应使相邻最小项矩形组合最大、且组合数目最少

Q11: 化简具有约束的逻辑函

数: $Y = A'B'C'D' + A'BCD + AB'C'D'$

给定的约束条件为: $A'B'CD + A'BC'D + ABC'D' + AB'C'D + ABCD + ABCD' + AB'CD' = 0$

 该函数也可以写为

$Y(A, B, C, D)$

$= \Sigma m(1, 7, 8)$

$+ d(3, 5, 9, 10, 12, 14, 15)$



2. 逻辑函数式的化简

⑦ 多输出逻辑函数式的化简

- 有几个输出，就画几张卡诺图
- 方法与单输出类似，均为合并最小项
- 需要注意公共项的处理

⑧ 多变量逻辑函数式的化简

- 卡诺图一般只适用于不超过4变量逻辑函数
- 使用公式法消除部分变量
- 观察逻辑函数，是否存在可退化变量
 - 比如变量A和B始终同时出现
- 当变量不超过4时，再使用卡诺图化简

Q12: (2021·期末) 化简

多输出逻辑函数:

$$\begin{cases} Y_1(A, B, C, D) = \sum (3, 4, 5, 6, 7, 8, 9, 12, 13, 14, 15) \\ Y_2(A, B, C, D) = \sum (2, 3, 4, 6, 7, 12, 14) \\ Y_3(A, B, C, D) = \sum (2, 6, 8, 9) \end{cases}$$

Q13: 化简: $Y = AC +$

$$AC'D + AB'E'F + \\ B(D \oplus E) + BC'DE' + \\ BC'D'E + ABE'F$$



◆ 知识点总结

◆ 习题讲解

◆ 补充题

1. 利用二进制补码列算式计算: $-20+17$

20与17的和为37, 故采用6位二进制补码

-20: **1**010100 \longrightarrow 补码为1101100 (反码+1)

17: **0**010001 \longrightarrow 补码为其本身

符号位

1101100

+

0010001

数值位

1 111101

本题是否可以使用5位二进制补码?
如果题目改为计算-20-17呢?

验算: 对应**十进制**数为

$$\begin{aligned} & (-1) \times 2^6 + 2^5 + 2^4 + 2^3 \\ & + 2^2 + 2^0 = -3 \quad \checkmark \end{aligned}$$

2. (2021·期中) 将十进制数202.1分别转换为二进制数、八进制数和十六进制数 (保留一位小数)

先转换为二进制, 然后再转换为8进制和16进制

➤ 整数部分: $(202)_{10} = (11001010)_2$

➤ 小数部分: $\times 2$ 取整后, 正序排列

$0.1 \times 2 = 0.2$	0
$0.2 \times 2 = 0.4$	0
$0.4 \times 2 = 0.8$	0
$0.8 \times 2 = 1.6$	1
$0.6 \times 2 = 1.2$	1
$0.2 \times 2 = 0.4$	0
$0.4 \times 2 = 0.8$	0
$0.8 \times 2 = 1.6$	1

$(11001010.0)_2$

小数部分

11001010.00011001

八进制:

$011 \mid 001 \mid 010$
3 1 2
 $.000 \mid 110 \mid 010$
0 6 2

$(312.062)_8$

保留一位小数
 $(312.1)_8$

十六进制:

$1100 \mid 1010$
C A
 $0001 \mid 1001$
1 9

$(CA.19)_{16}$

保留一位小数
 $(CA.2)_{16}$

3. (2021·期中) 已知 $L(A, B, C) = AB' + BC'$, 求:

① L 的真值表

A	B	C	L
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

3. (2021·期中) 已知 $L(A, B, C) = AB' + BC'$, 求:

② L 的标准与或式、标准或与式

根据真值表可以立即写出:

$$\begin{aligned} L(A, B, C) &= \Sigma m(2, 4, 5, 6) \Rightarrow \text{标准与或式} \\ &= ((m_2 + m_4 + m_5 + m_6)')' \\ &= (m_0 + m_1 + m_3 + m_7)' \\ &= m_0' m_1' m_3' m_7' \\ &= M_0 M_1 M_3 M_7 \\ &= \Pi M(0, 1, 3, 7) \Rightarrow \text{标准或与式} \end{aligned}$$

3. (2021·期中) 已知 $L(A, B, C) = AB' + BC'$, 求:

③ 使用卡诺图化简法, 求 L 反函数的最简或与式

求最简或与式 \rightarrow 圈 0

反函数的最简或与式 \rightarrow 圈 1

$$L = AB' + BC'$$

$$L' = (AB' + BC')'$$

$$= (A' + B)(B' + C)$$

注意: 本题其实不需要用卡洛图, 直接对原式取反即可; 使用卡诺图是为了展示该方法在最简或与式化简中的作用

		BC			
		00	01	11	10
A	0	0	0	0	1
	1	1	1	0	1

4. 化简一组多输出逻辑函数：

$$Y_1 = \Sigma m(1, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15)$$

$$Y_2 = \Sigma m(1, 3, 4, 5, 6, 7, 12, 14)$$

$$Y_3 = \Sigma m(3, 7, 10, 11)$$

注意：公共项的使用

AB \ CD				
	00	01	11	10
00	0	1	0	0
01	1	1	1	1
11	1	1	1	1
10	0	0	1	1

$$Y_1 = B + AB'C + A'C'D$$

AB \ CD				
	00	01	11	10
00	0	1	1	0
01	1	1	1	1
11	1	0	0	1
10	0	0	0	0

$$Y_2 = A'C'D + A'CD + BD'$$

AB \ CD				
	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	0	0	0	0
10	0	0	1	1

$$Y_3 = A'CD + AB'C$$

5. (2020·期末) 试用**公式法**将逻辑函数式 $Y = AC + B'C + BD' + CD' + AB + A'BCD' + AC'$ 化简成最简与或式

$$\begin{aligned}
 & \text{= } A + AB = A \\
 Y &= \underline{AC + AC' + AB} + \boxed{A'BCD'} + B'C + \boxed{BD'} + CD' \\
 &= A + \boxed{BD'} + B'C + (B + B')CD' \\
 &= A + \boxed{BD'} + \boxed{B'C} + \boxed{BCD'} + \boxed{B'CD'} \\
 &= A + \boxed{BD'} + \boxed{B'C}
 \end{aligned}$$

注：

此类题型需要大家熟练掌握逻辑代数基本公式（比如本题中反复使用 $A + AB = A$ ），并需要熟悉添项/拆项等技巧

6. (2017·期末) 用卡诺图求逻辑函数 $Y(A, B, C, D) = \Sigma m(2, 3, 4, 5, 6, 7, 12, 13) + d(10, 14, 15)$ 的最简与或式

化简结果为:

$$Y = B + A'C$$

要求:

- ① 所有的1都被圈出
- ② 所有的0都不能被圈出
- ③ 圈的数量尽量少
- ④ 圈的大小尽量大
- ⑤ X可以被圈, 也可以不被圈

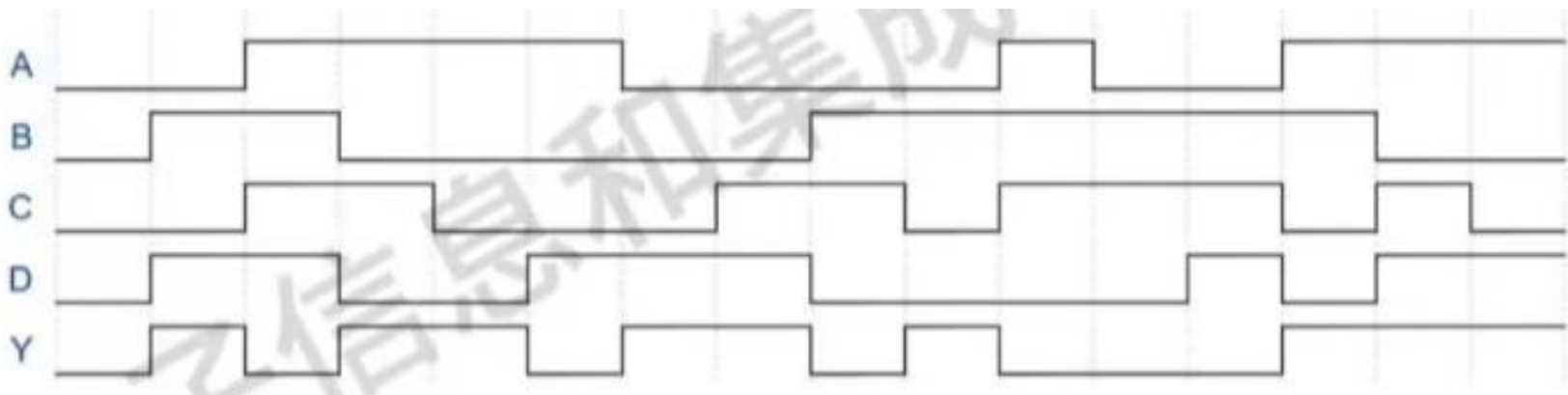
AB \ CD	CD			
	00	01	11	10
00	0	0	1	1
01	1	1	1	1
11	1	1	x	x
10	0	0	0	x

◆ 知识点总结

◆ 习题讲解

◆ 补充题

1. 简述摩尔定律的内容
2. (2025·复旦) 用余三码表示十进制数 8 是 _____
3. (2023·清华) 将表达式 $F = A'C'D' + A'BD + AB' + B'CD'$ 化为与或非的形式
4. (2019·期末) 求逻辑函数式 $Y = (A + B + C)(A' + B + C')(A + C' + D')(A' + D)(B + C + D')$ 的最简与或式
5. (2025·复旦) 已知电路波形图如下，写出输出Y的最简与或式和最简或与式。



6. (2021·期末) 已知四变量函数 Y_1 和 Y_2 :

$$\begin{cases} Y_1(A, B, C, D) = \Sigma m(2, 3, 4, 6, 8, 10, 12, 14) \\ Y_2(A, B, C, D) = \Sigma m(0, 1, 2, 5, 7, 8, 12, 14) + d(3, 9, 10) \end{cases}$$

① 求 Y_1 的最简 “与或非” 式

② 求 Y_2 的最简 “与或” 式

③ 求复合函数 $Y_1 \oplus Y_2$ 最小项之和的形式

7. (2021·复旦) 与最小项 AB 相邻的最小项分别为 ____ 和 ____

8. (2020·复旦) 函数 $Y = AB + CD$ 的对偶式为 _____, 反函数为 _____

9. (2020·复旦) $A \oplus 1 = \underline{\hspace{1cm}}$, $A \oplus 0 = \underline{\hspace{1cm}}$

10. (2021•期中) 已知十进制数 $X=5$, $Y=7$, 用 4 位二进制补码形式列竖式计算 $X+Y$ 和 $X-Y$, 将结果转换为十进制, 并判断运算结果是否溢出
11. (2021•期中) 用代数法将 $(B' \oplus C + (A'B)') \cdot (A' + B'C') + A'C)$ 化为最简与或式
12. (2024•期末) $(14.375)_{10} = (\underline{\hspace{2cm}})_2$
13. (2024•期末) 用反演定理写出 $((AB' + C)'D)' + B'C$ 的反逻辑式 , 再用对偶定理写出该式的对偶式 。
14. (2024•期末) 试用卡诺图化简逻辑函数 $Y = \Sigma m(1, 3, 4, 7, 11, 13) + \Sigma \phi(2, 5, 10, 12, 14, 15)$, 其中 ϕ 表示无关项, 并将最简与或式转换为“或非-或非”形式

感谢各位聆听!