

组合逻辑作业

(所有答案字体均已加粗)

一、填空题:

1. TTL 与非门的两个状态通常称为关态和开态，当输入全为高电平时对应的是**开**态，此时输出为**低电平**；当输入有一为低电平时，对应的是**关**态，此时输出为**高电平**。
2. 对于 ECL、TTL、CMOS 集成电路，按静态功耗低和高的顺序依次为**CMOS TTL ECL**，按工作速度快慢的顺序依次为**ECL、TTL、CMOS**，按抗干扰能力强弱的顺序依次为**CMOS，TTL，ECL**。
3. 逻辑变量和函数只有**0, 1**两种取值，而且它们只是表示两种不同的逻辑状态。
4. 逻辑函数 $F = \overline{A}C + \overline{B}D$ ，其对偶式 $F^* = (\overline{A} + C)(\overline{B} + D)$ ，其反函数 $\overline{F} = (A + \overline{C})(B + \overline{D})$ 。
5. 函数 $F = AB + AC + \overline{C}D + ACDE$ 的最简与或式是 **AC + AB + CD**。
6. 从结构看，组合逻辑电路由门电路构成，不含**记忆电路**，也不含**反馈电路**，信号从输入开始单向传输到输出。对于组合逻辑电路，任何时刻电路的输出仅由当时的决定。
7. 将加在电路若干输入端中的某一个输入端的信号变换成相应的一组二进制代码输出的过程叫做**编码**。
8. 将二进制代码所表示的信息翻译成对应输出的高低电平信号的过程称为**译码**； n 位二进制译码器有 **n** 个输入，有 **2^n** 个输出，工作时译码器只允许有一个输出有效。
9. 输出低电平有效的二-十进制译码器的输入 8421BCD 码 A3~A0 为 1101 时，其输出 $\overline{Y}_0 \sim \overline{Y}_9 =$ **1110111111**。

10. 消除组合逻辑电路中险象的常用方法有_____修改逻辑设计_____、_____选通法_____和_____滤波法_____。

11. 逻辑函数常用_____函数表达式_____、_____真值表_____、_____卡诺图_____和_____逻辑图_____四种方法表示，它们之间是可以相互转换的。

二、选择题：

1. 设两输入或非门的输入为 x 和 y, 输出为 z, 当 z 为低电平时, 有 (④)。

① x 和 y 同为高电平 ② x 为高电平, y 为低电平

③ x 为低电平, y 为高电平 ④ 以上都对

2. 逻辑函数 $F = A + BC(A + B)$, 当 ABC 的取值为 (②) 时, $F = 1$ 。

① 000 ② 011

③ 101

④ 111

3. 化简 $F = A(BC + \overline{BC}) + AB\overline{C} + A\overline{B}C$ 的结果为 (③)

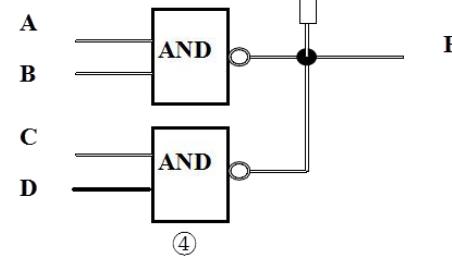
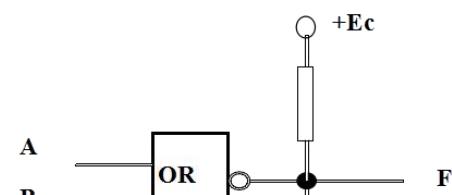
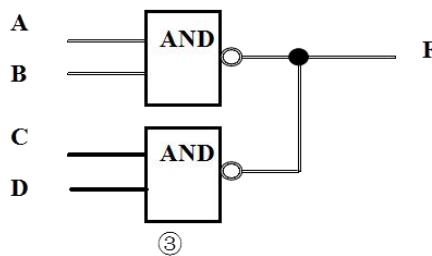
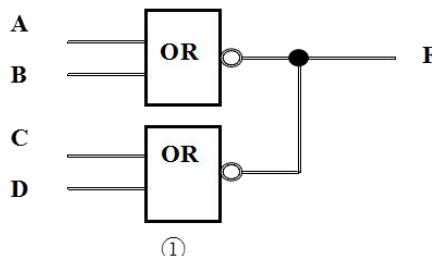
①. AC

②. $A\overline{B}$

③. A

④. $B\overline{C}$

4. 为实现 $F = \overline{AB} \cdot \overline{CD}$, 下列电路接法正确的是 (④)。



5. 组合逻辑电路的竞争-冒险是由于 (③) 引起的。

- ① 电路不是最简
- ② 电路有多个输出
- ③ 构成电路的逻辑元件存在传输延迟
- ④ 电路使用不同的门电路

6. 下列四种类型的逻辑门中, 可以用 (④) 实现三种基本运算。

- ① 非门
- ② 与门
- ③ 或门
- ④ 或非门

7. 如需要判断两个二进制数的大小或相等, 可以使用 (④) 电路。

- ① 译码器
- ② 编码器
- ③ 数据选择器
- ④ 数值比较器

8. 函数 $F = AB + \overline{A}C + \overline{B}C + \overline{C}D + \overline{D}$ 的最简与或式为 (①) 。

- ① 1
- ② 0
- ③ AB
- ④ 以上均不是

9. 只考虑本位数而不考虑低位来的进位的加法称为 (②) 。

- ① 全加
- ② 半加
- ③ 全减
- ④ 半减

10. 能实现从多个输入端中选出一路作为输出的电路称为 (③) 。

- ① 触发器
- ② 计数器
- ③ 数据选择器
- ④ 译码器

11. 函数 $A \oplus B$ 与 $\overline{A} \oplus \overline{B}$ (③) 。

- ① 互为反函数
- ② 互为对偶式
- ③ 相等
- ④ 答案都不正确。

三、问答与计算题:

说明: 分析与计算题要求写出分析推导过程, 给出必要的公式。

1. 将逻辑函数 $F = ABC + \overline{A}BD + ABC\overline{D}$ 写成标准与或表达式。

三、

$$1. F = ABC(C + \bar{D}) \cancel{= ABC\bar{D}} + \bar{A}BD$$

$$\cancel{= ABC\bar{D} + ABD} = ABC\bar{D} + \bar{A}BD(C + \bar{C}) = BC + \bar{ABC}$$

$$= ABC + \bar{A}BD$$

2. 由卡诺图化简知 $F = \bar{B} + \bar{C}$

中国·北京 100191

37XUEYUANROADBEIJING 10019

2. 推导出函数 $F = \overline{AB + BC} + \overline{AC} + \overline{AB}$ 的最简与或式。

三、

$$1. F = ABC(C + \bar{D}) \cancel{= ABC\bar{D}} + \bar{A}BD$$

$$\cancel{= ABC\bar{D} + ABD} = ABC\bar{D} + \bar{A}BD(C + \bar{C}) = BC + \bar{ABC}$$

$$= ABC + \bar{A}BD$$

2. 由卡诺图化简知 $F = \bar{B} + \bar{C}$

中国·北京 100191

37XUEYUANROADBEIJING 10019

3. 列出下述问题的真值表，利用最小项推导法写出其逻辑函数表达式，利用公式简化法进行简化并给出逻辑电路图。最后，写出完整的 Verilog HDL 程序。

设计一个投票表决器，三个投票人分别为 A、B、C，同意为“1”，不同意为“0”。按规定只要二人以上同意才能通过，输出为“1”表示通过，为“0”表示不通过。

Combinational Analysis

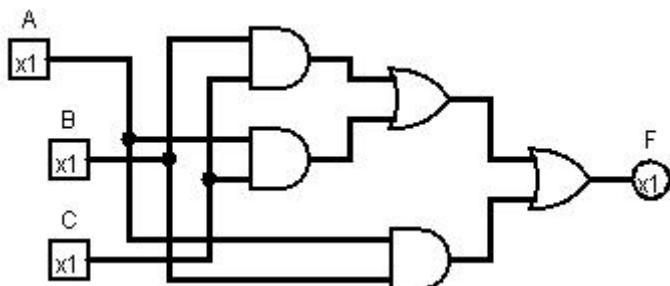
文件 编辑 项目 模拟 窗口 帮助

输入 输出 真值表 表达式 最小项

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

构建电路

真值表



表决器

$$F = B C + A C + A B$$

```

module decider (
    input A,
    input B,
    input C,
    output F
);

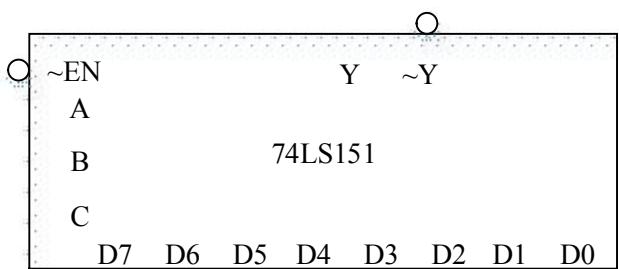
    assign F = (BC + AC + AB);
endmodule
  
```

4. 用逻辑代数证明下列等式:

$$BC + D + \overline{D}(\overline{B} + \overline{C})(AD + B) = B + D$$

4.
$$\begin{aligned} & BC + D + (\overline{D}\overline{B} + \overline{C}\overline{D})(AD + B) \\ &= BC + D + B\overline{C}\overline{D} \\ &= D + BC(C + \overline{C}\overline{D}) = D + BC(CD + \overline{C}\overline{D} + C\overline{D}) = D + BC(1 + \overline{C}\overline{D}) \\ &= B + D \end{aligned}$$

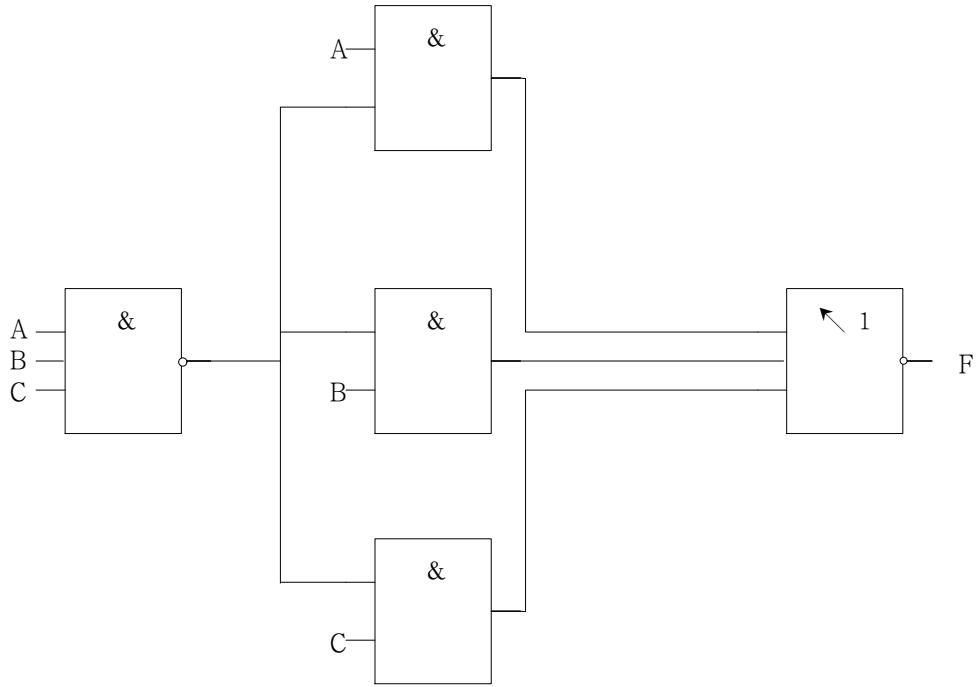
5. 下图所示为 8 选 1 数据选择器 CT74151, 若要实现 $Y = \overline{ABC} + \overline{ABC} + AC$, 问 $D_0 \sim D_7$ 中哪些管脚需要输入高电平, 哪些管脚需要输入低电平?



低: **D0, D1, D3, D6**

高: **D2, D4, D5, D7**

6. 分析下图的组合逻辑电路, 说明电路功能, 并写出最简与或式, 画出对应的真值表。



$$F = ABC + \overline{ABC}$$

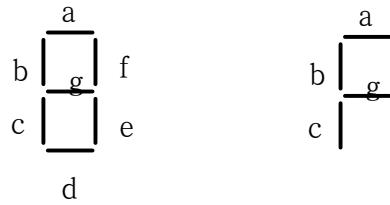
ABC 必须要全 0 或全 1, $F = 1$, 所以该电路为“一致性电路”

Combinational Analysis																																							
文件	编辑	项目	模拟 窗口 帮助																																				
输入	输出	真值表	表达式 最小项																																				
<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>				A	B	C	F	0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	1	0	1	0	1	1	0	0	1	1	1	1
A	B	C	F																																				
0	0	0	1																																				
0	0	1	0																																				
0	1	0	0																																				
0	1	1	0																																				
1	0	0	0																																				
1	0	1	0																																				
1	1	0	0																																				
1	1	1	1																																				
构建电路																																							

真值表

7. 7 段数码管是由 7 个独立的发光管构成的，每个发光管有一个驱动控制

信号。当驱动控制信号为高电平(逻辑 1)时, 则信号对应的发光管发光。现需设计 7 段数码管的控制电路, 使之能够根据 4 位输入 $x[3:0]$ 显示 $0 \sim 9$, A ~ F 共 16 个图案。7 段数码管控制电路输出信号为各数码管的驱动控制信号, 即 a, b, c, d, e, f, g。数码管各段的定义和 16 进制数 “F” (对应 abcdefg 的二进制输出为 1110001) 的显示如下图所示。



a) 各段定义 b) 16进制数
 F的显示

- (1) 请给出 7 段数码管控制电路的输入输出信号真值表。
 - (2) 根据真值表写出各输出信号的逻辑表达式, 并化简。
 - (3) 采用结构描述法, 用 Verilog 语言实现上述的数码管控制器。
- (1)

Combinational Analysis

文件 编辑 项目 模拟 窗口 帮助

输入 输出 真值表 表达式 最小项

A3	A2	A1	A0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x

构建电路

真值表

(2)

$$\begin{aligned}
 a &= \sim A_2 \sim A_0 + A_1 + A_2 A_0 + A_3 \\
 b &= \sim A_2 + \sim A_1 \sim A_0 + A_1 A_0 \\
 c &= \sim A_1 + A_0 + A_2 \\
 d &= \sim A_2 \sim A_0 + \sim A_2 A_1 + A_1 \sim A_0 + A_2 \sim A_1 A_0 + A_3 \\
 e &= \sim A_2 \sim A_0 + A_1 \sim A_0 \\
 f &= \sim A_1 \sim A_0 + A_2 \sim A_1 + A_2 \sim A_0 + A_3 \\
 g &= \sim A_2 A_1 + A_2 \sim A_1 + A_2 \sim A_0 + A_3
 \end{aligned}$$

(3)

```

module seg_dec(num,a_g);
input[3:0]      num;
output[6:0]     a_g;//a_g[6:0]->(a,b,c,d,e,f,g)
reg[6:0]        a_g;

```

```
always@(num)begin
  case(num)
    4'd0: a_g<=7'b111_1110;
    4'd1: a_g<=7'b011_0000;
    4'd2: a_g<=7'b110_1101;
    4'd3: a_g<=7'b111_1100;
    4'd4: a_g<=7'b011_0011;
    4'd5: a_g<=7'b101_1011;
    4'd6: a_g<=7'b101_1111;
    4'd7: a_g<=7'b111_0000;
    4'd8: a_g<=7'b111_1111;
    4'd9: a_g<=7'b111_1011;
    default: a_g<=7'b000_0001;
  endcase
end
endmodule
```