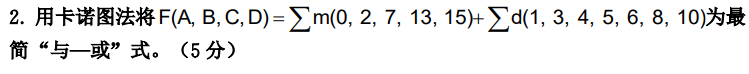
# 一、化简逻辑表达式（第一大题）

## 第1题：用代数法化简，用真值表或卡诺图得出答案后倒推理，速度快又不易错。

## 第2题：用卡诺图化简，摘取两道题作样本

（2021弘毅A、B卷一）

图示, 工程绘图

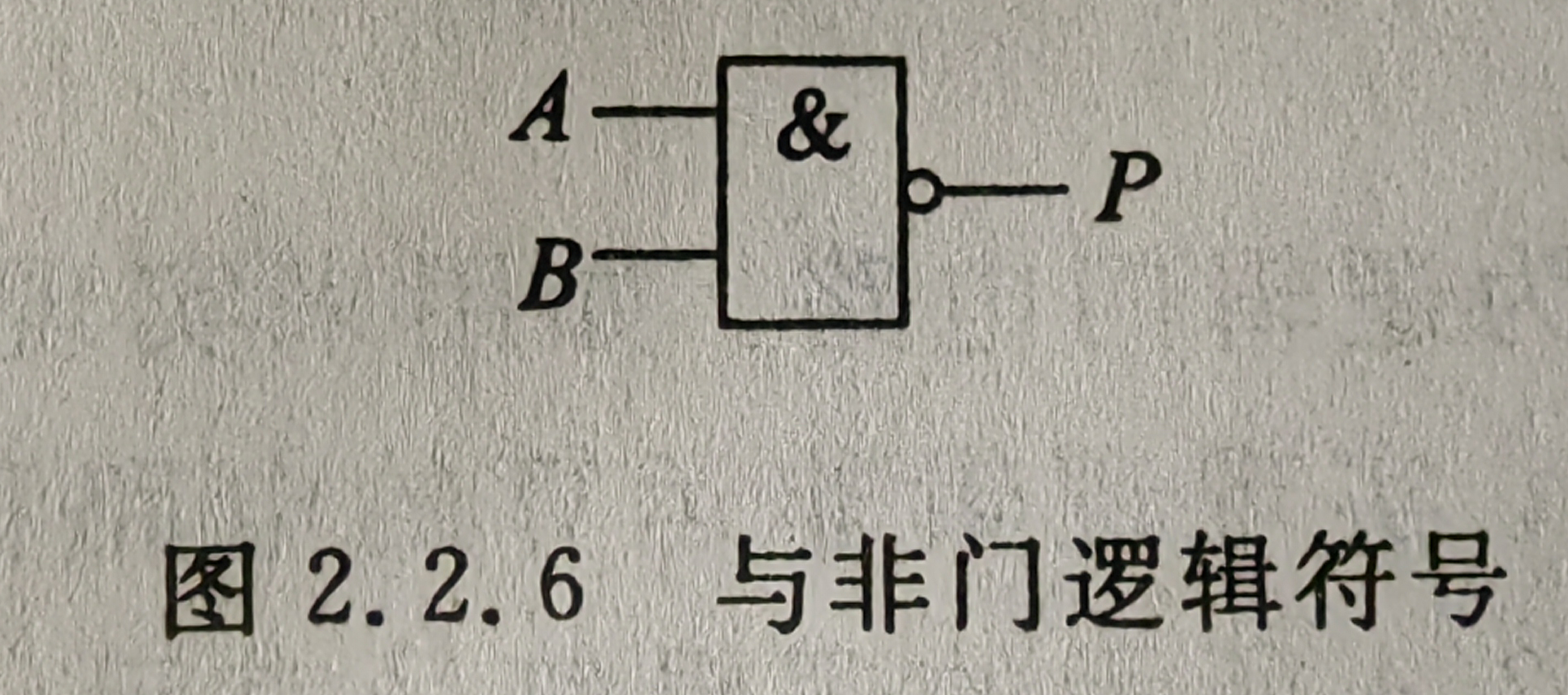
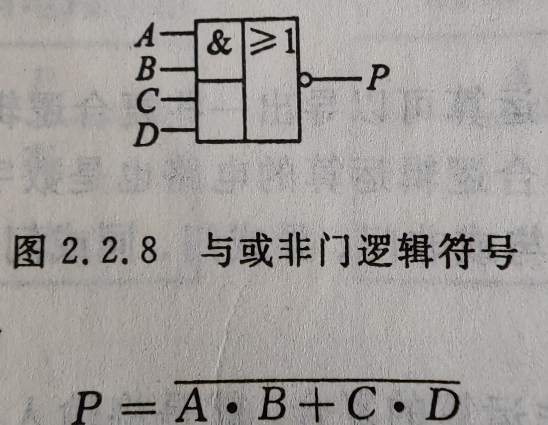
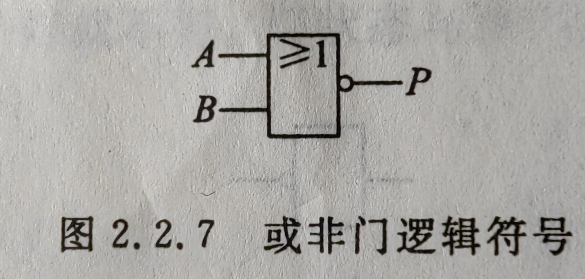
描述已自动生成

图示, 工程绘图

描述已自动生成

# 图示, 示意图 描述已自动生成二、组合逻辑电路（第二题、第三题、第五题）

## ①分析组合逻辑电路（第二题）：很简单

****

**异或：相异得1：**黑色的钟表

低可信度描述已自动生成墙上的钟表

中度可信度描述已自动生成

**同或：相同得1**

钟表的特写

描述已自动生成图示, 示意图

描述已自动生成

**1、（2021弘毅 A卷 二）**

分析图 1 所示组合逻辑电路，回答问题：

1. 求输出函数的逻辑表达式；（4 分）

2. 列出函数真值表；（4 分）

3. 判断此逻辑电路功能。（4 分）

图示

描述已自动生成

**2、（2021弘毅 B卷 二）**

分析图 1 所示的逻辑电路：

1、求输出函数与输入变量的表达式、并化简（4 分）；

2、求函数真值表与函数最小项表达式（4 分）；

3、改用异或门实现该电路的逻辑功能（4 分）；

图示, 工程绘图

描述已自动生成

**3、（2020弘毅 二）**

分析图 1 所示组合逻辑电路，输入 ABCD 为 8421 的 BCD 码。

1. 求 W、X、Y、Z 的逻辑表达式；（4 分）

2. 列出输出对应的真值表；（4 分）

3. 判断此逻辑电路功能。（4 分）

图示

描述已自动生成

**4、（2017 计院 四）**

分析如图所示组合逻辑电路（每问4分）

① 写出输出函数表达式

② 列出真值表

③ 说明电路功能

图示, 示意图

描述已自动生成

**5、（2015 计院 三1）**

分析图1所示组合逻辑电路，其中16引脚的集成电路为三－八译码器74138。要求：

① 写出函数表达式（4分）

② 列出函数真值表（4分）

③ 说明电路功能 （4分）

图示

描述已自动生成

**6、（2019 计院 一）**

分析如下图所示组合逻辑电路（每问 6 分，共 18 分）。

1、写出表达式

2、列出真值表

3、说明逻辑功能

图示, 示意图

描述已自动生成

## ②组合逻辑电路的应用

**芯片真值表课本位置（随着书版本不同可能有一两页的误差）**

**8-3编码器（P114）**

**优先编码器74LS138（P120）**

**二-十进制编码器74LS42(P125)**

**七段译码器74LS48(P129)**

**4位二进制同步可预置加法计数器74LS161(P210)**

**单时钟可逆十六进制计数器74LS191(P211)**

**74LS90（P215）**

**1、（2021弘毅 A卷 三）**

设 ABCD 输入值为 8421BCD 码，试设计一个电路判断其输入数值是否为质数的电路，当输入为质数（含 1）时 F=1，否则 F=0。

1. 写出函数 F 与 ABCD 的真值表；（3 分）

2. 写出逻辑函数表达式并化简；（3 分）

3. 用适当逻辑门电路实现该函数并画出逻辑电路图；（3 分）

4. 用三—八译码器 74LS138 及适当逻辑门电路实现该电路的逻辑功能；（3 分）

5. 用中规模集成电路八选一 74LS151 及适当逻辑门电路实现其逻辑功能；（3 分）

6. 用 Verilog HDL 语言设计该逻辑电路；（4 分）

7. 用如图2所示的可编程逻辑器件PLA设计该逻辑函数。（3 分）

图示

描述已自动生成

**2、（2021弘毅 B卷 三）**

设 ABCD 输入值为 8421BCD 码，试设计一个“四舍五入”电路。当其输入数值大于或等于五时F=1，否则 F=0。

1. 写出函数 F 与 ABCD 的真值表；（3 分）

2. 写出逻辑函数表达式并化简；（3 分）

3. 用适当门电路实现该函数并画出逻辑电路图；（3 分）

4. 用三—八译码器74LS138及适当门电路实现该电路的逻辑功能；（3 分）

5. 用中规模集成电路八选一74LS151及适当门电路实现其逻辑功能；（3 分）

6. 用 Verilog HDL 语言设计该逻辑电路；（4 分）

7. 用如图 2 所示的可编程逻辑器件 PLA 设计该逻辑函数。（3 分）

**3、（2020弘毅 三）**

为预防新冠通常会检测核酸 NAT（Nucleic Acid Test）、IgG、IgM 三项指标。IgG 是抗体中分子量最小的一种，该抗体产生时间晚、维持时间长、消失慢、 浓度高；IgM 是抗体中分子量最大的一种，一经感染，快速产生作用，维持时间短、消失快。依据三项指标的检测（NAT、IgG、IgM）结果确定该人员是否需要被隔离，如果需要被隔离 F 为 1，否则 F 为 0。其防疫要求如下：

①当 NAT 为阳性（为 1），不管其他指标如何都为疑似感染者需要被隔离；

②当 NAT 为阴性（为 0），如果 IgG 和 IgM 都为阳性（均为 1），则该人员为疑似感染者需要被隔离，其他情况（单阳）视为安全的不需要被隔离。根据要求回答下列问题：

1. 写出函数 F 与 N(NAT)、G(IgG)、M(IgM)真值表；（4 分）

2. 写出逻辑函数表达式并化简；（4 分）

3. 用与非门实现该函数并画出逻辑电路图；（4 分）

4. 用二-四译码器 74LS139 及适当门电路实现该电路的逻辑功能；（4 分）

5. 用中规模集成电路八选一 74LS151 及适当门电路实现其逻辑功能；（4分）

6. 用 Verilog HDL 语言设计该逻辑电路；（4 分）

7. 用如图 2 所示的可编程逻辑器件 PLA 设计该逻辑函数。（4 分）

图示

描述已自动生成

**4、（2021弘毅 A卷 五）法二不接A0？**

现有 6 个数字逻辑功能部件需要轮流工作，每个功能部件都有一个片选端 ，且低电平有效。试用计数器74LS161和3-8 译码器74LS138设计产生各个功能部件的片选信号。以便在 6 个 CP 时钟周期内，每个功能部件都能够分时依次被选通，并依此循环工作。具体功能要求如图 4 波形图（提示：将计数器输出作为译 码器输入）。

1. 请说明设计思路；（8 分）

2. 画出实现原理图。（8 分）

图示

描述已自动生成图示, 示意图

描述已自动生成

**5、（2021弘毅 B卷 五）**

用两片4位二进制同步计数器74LS193构成模147十进制计数器。

**6、（2020弘毅 五）**

用 2 片 74LS161 级联构成一个五十三进制计数器，请画出相应的电路连线图，要求有分析过程。

1. 使用异步清零法实现，并画出电路图；（5 分）

墙上的钟表

描述已自动生成

2. 使用同步置位法实现，并画出电路图。（5 分）

手机屏幕截图

中度可信度描述已自动生成

# 三、时序逻辑电路

## ①时序逻辑电路的分析

**1、（2021 弘毅 A卷 四）**

分析图 3 的时序逻辑电路，回答问题： 图示

描述已自动生成

1. 写出激励函数与输出函数的表达式；（4 分）

2. 写出次态方程组；（4 分）

3. 作出电路的状态表及状态图；（6分）

4. 分析电路的逻辑功能。（4 分）

**2、（2021 弘毅 B卷 四）**

分析图 3 的时序逻辑电路，回答问题：

1. 写出激励函数与输出函数的表达式；（6 分）

2. 写出次态方程组；（4 分）

3. 作出电路的状态表及状态图；（6 分）

4. 分析电路的逻辑功能。（4 分）

图示, 示意图

描述已自动生成

**3、（2020 弘毅 四）**

分析图 3 时序逻辑电路，回答问题：

1. 写出输出函数与激励函数表达式；（4 分）

2. 写出电路次态真值表；（4 分）

3. 列出状态表与状态图；（4 分）

4. 分析电路的逻辑功能。（4 分）

图示, 示意图

描述已自动生成

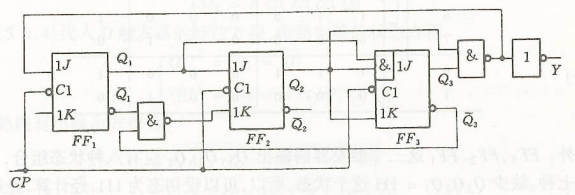
**4、（2016 计院 四2）**

分析图2所示同步时序逻辑电路（每问4分）

① 写出激励函数和输出函数表达式，指出电路属于哪种模型？

② 作出状态表和状态图

③ 说明电路功能，作出Q3、Q2、Q1、Y的波形图



**5、（2015 计院 三2）**

分析图2所示电路，设Q1Q2=00，输入x和CP的波形如图，要求：

①写出激励函数表达式（4分）

②判断电路是同步还是异步时序逻辑电路，是Moore型还是Mealy型（4分）

③画出输出Q1、Q2的波形图。（4分）

墙上的钟表

中度可信度描述已自动生成

## ②设计同步时序逻辑电路

**1、（2021 弘毅 A卷 六）**

用正边缘 D 触发器作为存储元件，设计一个同步时序逻辑电路“110”序列检测器。该电路有一个输入 *x* 和一个输出 Z，当随机输入信号中出现“110”序列时，输出一个 1 信号。典型输入/输出序列如下：

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 输入x | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 输出Z | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

1. 作出原始状态图、状态表及二进制状态表；（6 分）

2. 求D触发器的各输入端的激励表达式及输出函数表达式；（6分）

3. 画出相应电路图并说明该电路是否能够自启动；（5 分）

4. 用 Verilog HDL 语言描述状态图。（5 分）

**2、（2021 弘毅 B卷 六）**

用 T 触发器作为存储元件，设计一个 2 位二进制减 1 计数器。电路工作状态受输入信号 x 的控制。当 x =0 时,电路状态不变;当 x =1 时，在时钟脉冲作用下进行减 1 计数。计数器有一个输出 Z，当产生借位时 Z 为 1,其他情况下 Z 为 0。

1. 作出原始状态图、状态表及二进制状态表；（6 分）

2. 求 D 触发器的各输入端的激励表达式及输出函数表达式；（6分）

3. 画出相应电路图、并分析该电路是否能够自启动；（6 分）

4. 用 Verilog HDL 语言描述状态图。（6 分）

**3、（2020 弘毅 六）**

用 J-K 触发器作为存储元件，设计一个“101”序列检测器。该电路有一 个输入 x 和一个输出 Z，当随机输入信号中出现“101”序列时，输出一个 1 信号。典型输入/输出序列如下：

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 输入x | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 输出Z | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |

1. 作出原始状态图、状态表；（4 分）

2. 将原始状态化简；（4 分）

3. 对状态进行分配，并画出二进制状态表；（4 分）

4. 求 J-K 触发器的各输入端的激励表达式及输出函数表达式；（4 分）

5. 画出相应电路图；（4 分）

6. 分析该电路是否能够自启动。（4 分）

**4、（2018 计院 七）**

用T或D触发器设计一个同步十进制加1 计数器，输出z表示进位（20分，每问4分）。

1、形成原始状态图

2、求出输出方程和驱动方程

3、画出逻辑图

4、画出计数状态波形图

5、检查电路能否自启动

**5、（2016 计院 五）**

用上升沿触发的D触发器Mleay模型设计一个异步6进制加1计数器，该电路对输入端X进行计数，输出Z表示进位。D触发器激励表如下：

表格

描述已自动生成

**6、（2015 计院 五2）**

用T触发器作为存储元件，设计一个Mealy型模4可逆同步计数器，输入X=0时加1计数、X=1时减1计数，当有进位/借位时输出Z为1，否则Z为0。

要求：①作出状态图和状态表（4分）

②确定激励函数和输出函数（4分）

③画出逻辑图（4分）

**Verilog HDL专题  
1、组合逻辑电路的设计实现（课本P102-P103、该资料二②123）  
2、时序逻辑电路（该资料三②123）**

**3、计数器**

**module cnt\_10(rst,en,clk,q,cout);//十进制加法器**

**input rst,en,clk;**

**output reg[3:0] q;**

**output reg cout;**

**always@(posedge clk or negedge rst)**

**begin**

**if(~rst) begin q<=4'b0000; cout<=1'b0;end//是0就执行此行**

**else if(en)**

**if(q==4'b1001) begin q<=4'b0000;cout<=1'b1; end**

**else begin q<=q+1'b1; cout<=1'b0;end**

**end**

**endmodule**

**译码器**

**module DECODER7(BCD4,LED7); //W模块名为DECODER7，两个端口**

**input [3:0] BCD4; //输入为变量为BCD4，四位**

**output[6:0] LED7; //输出变量为LED7，七位**

**reg [6:0] LED7; //W定义输出为寄存器类型**

**always@(BCD4) //过程块,当敏感量BCD4改变时,执行过程块**

**begin**

**case(BCD4) //分支语句，BCD4改变时执行对应的分支语句**

**4’b0000:LED7=Tb0111111;**

**//当输入为四位二进制数0000时,将0111111输出，显示十进制0**

**4’b0001:LED7=7’b0000110;**

**4’b0010:LED7=7’b1011011;**

**4’b0011:LED7=7’b1001111;**

**4’b0100:LED7=7’b1100110;**

**4’b0101:LED7=7’b1101101;**

**4’b0110:LED7=7’b1111101;**

**4’b0111:LED7=7’b0000111;**

**4’b1000:LED7=7’b1111111;**

**4’b1001:LED7=7’b1101111;**

**default:LED7=7’b00000O0;**

**//当以上条件都不满起时,将 0000000送输出显示**

**endcase //分支语句结束**

**end //过程块结束**

**endmodule //程序结束**