

# 《数字逻辑》

(第3次实验：在Logisim和FPGA开发板上实现组合逻辑电路)

厦门大学信息学院软件工程系 曾文华

2024年10月12日

# 目录

- **第一部分：在Logisim上实现组合逻辑电路**
  - （一）在Logisim上实现教材第4章的例题
  - （二）在Logisim上实现教材第4章的习题
- **第二部分：在FPGA开发板上实现组合逻辑电路**
  - （一）在FPGA开发板上实现教材第4章的例题
  - （二）在FPGA开发板上实现教材第4章的习题

# 第一部分：在Logisim上实现组合逻辑电路

设计文件：在Logisim上实现教材第4章的例题和习题.circ

- ❏ 例题4.1用ROM的内容
- ❏ 例题4.2用ROM的内容
- ❏ 例题4.3用ROM1的内容
- ❏ 例题4.3用ROM2的内容
- ❏ 例题4.4用ROM的内容
- ❏ 例题4.7用ROM1的内容
- ❏ 例题4.7用ROM2的内容
- ❏ 例题4.8用ROM的内容
- ❏ 例题4.9用ROM的内容
- ❏ 例题4.12 (a) 用ROM的内容
- ❏ 例题4.12 (b) 用ROM的内容
- ❏ 习题4.7 (a) 用ROM1的内容
- ❏ 习题4.7 (a) 用ROM2的内容
- ❏ 习题4.7 (b) 用ROM1的内容
- ❏ 习题4.7 (b) 用ROM2的内容

📄 例题4.1用ROM的内容 - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)  
v2.0 raw  
0 6\*1

# (一) 在Logisim上实现教材第4章的例题

## • 1、（验证实验）例题4.1的实现

- 在Logisim上实现例题4.1的**不一致电路**，该电路有3个输入、1个输出。

表 4.1 真值表

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

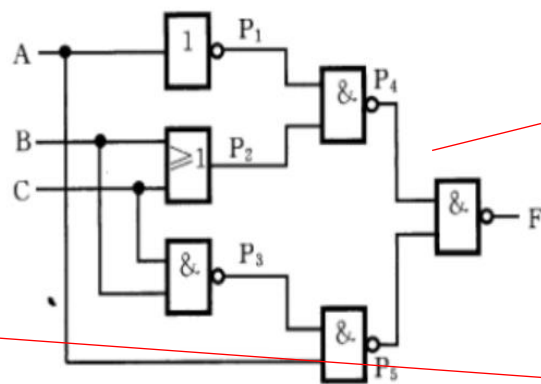
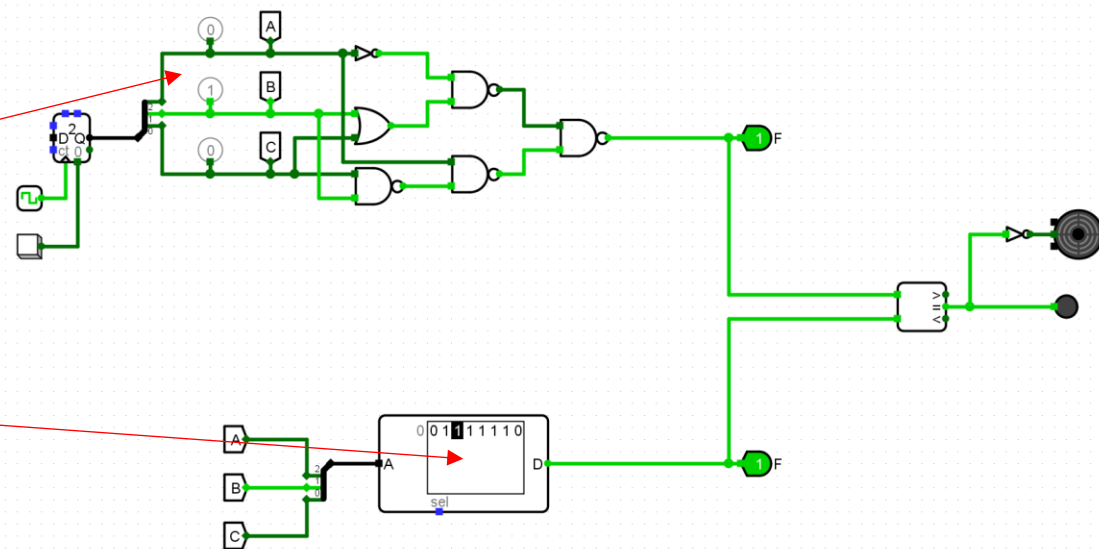


图4.2(a) 逻辑电路



## • 2、（验证实验）例题4.2的实现

- 在Logisim上实现例题4.2的半加器HA电路，该电路有2个输入、2个输出。

表 4.2 真值表			
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

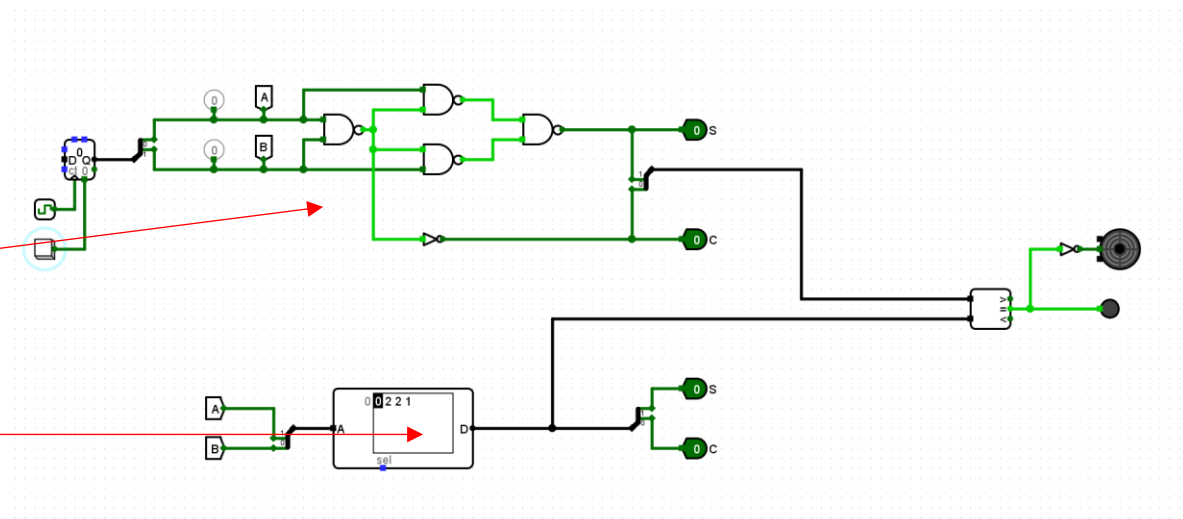
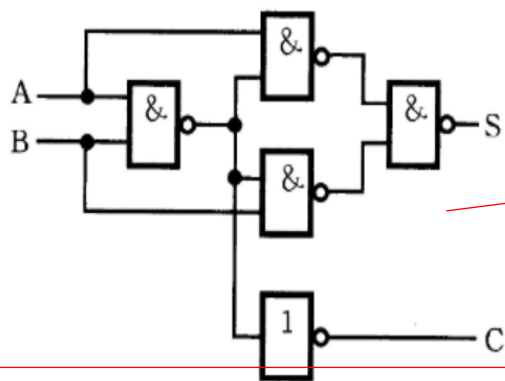


图4.3(a) 逻辑电路

### • 3、（验证实验）例题4.3的实现

- 在Logisim上实现例题4.3的将8421码转换为余3码的**代码转换电路**，该电路有4个输入、4个输出。

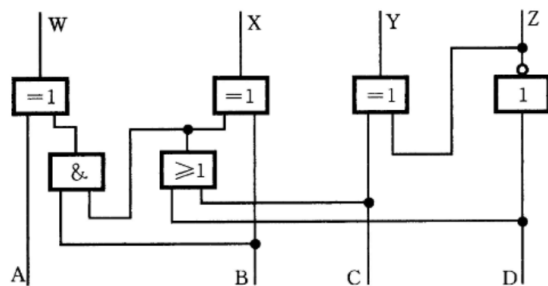
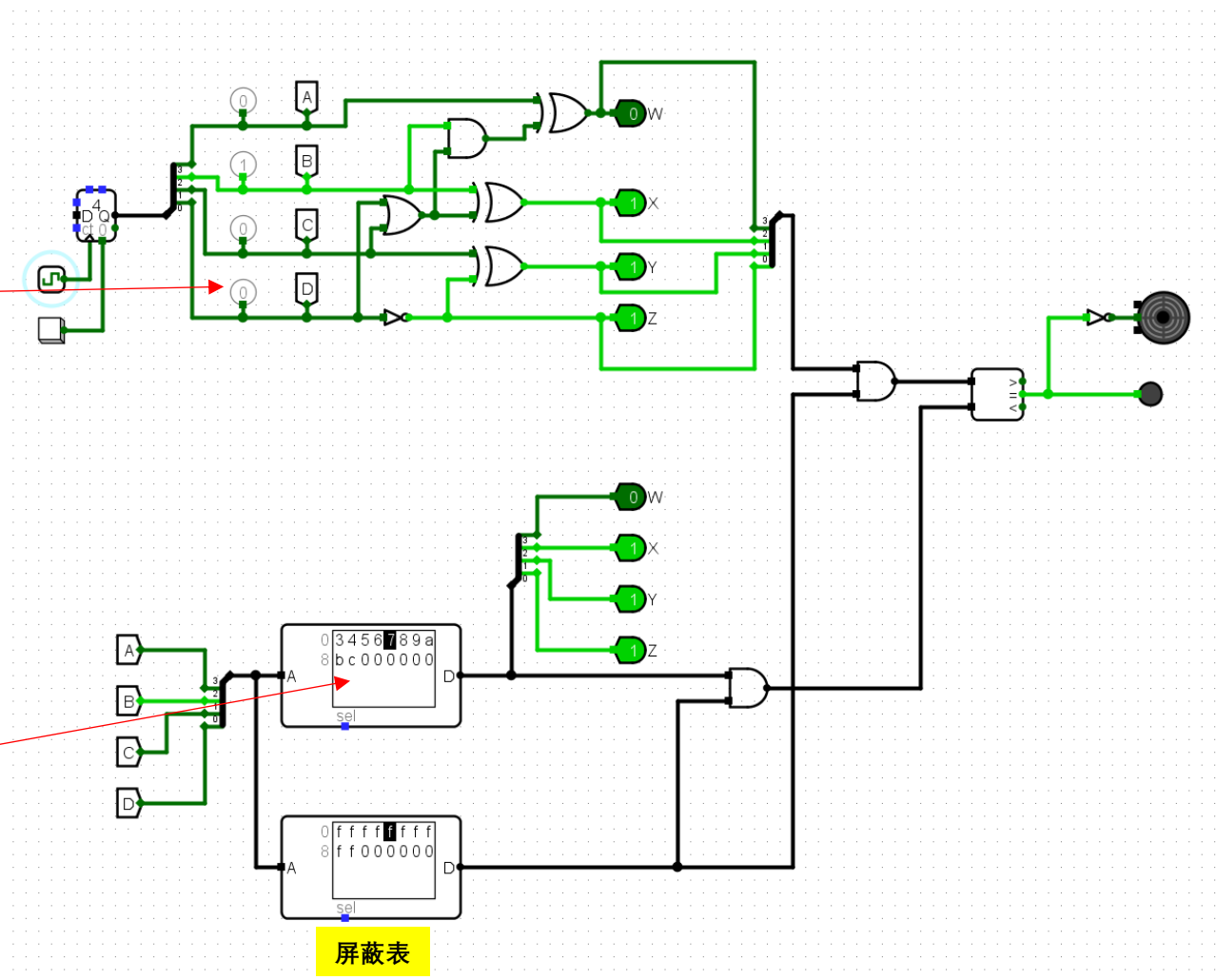


图 4.4 逻辑电路

表 4.3 真值表

ABCD	WXYZ	ABCD	WXYZ
0000	0011	0101	1000
0001	0100	0110	1001
0010	0101	0111	1010
0011	0110	1000	1011
0100	0111	1001	1100



## • 4、（验证实验）例题4.4的实现

- 在Logisim上实现例题4.4的3变量多数表决电路，该电路有3个输入、1个输出。

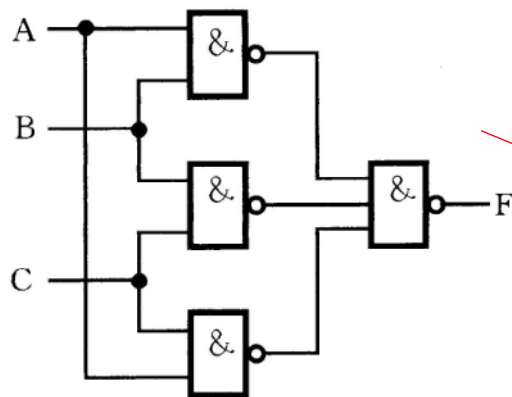
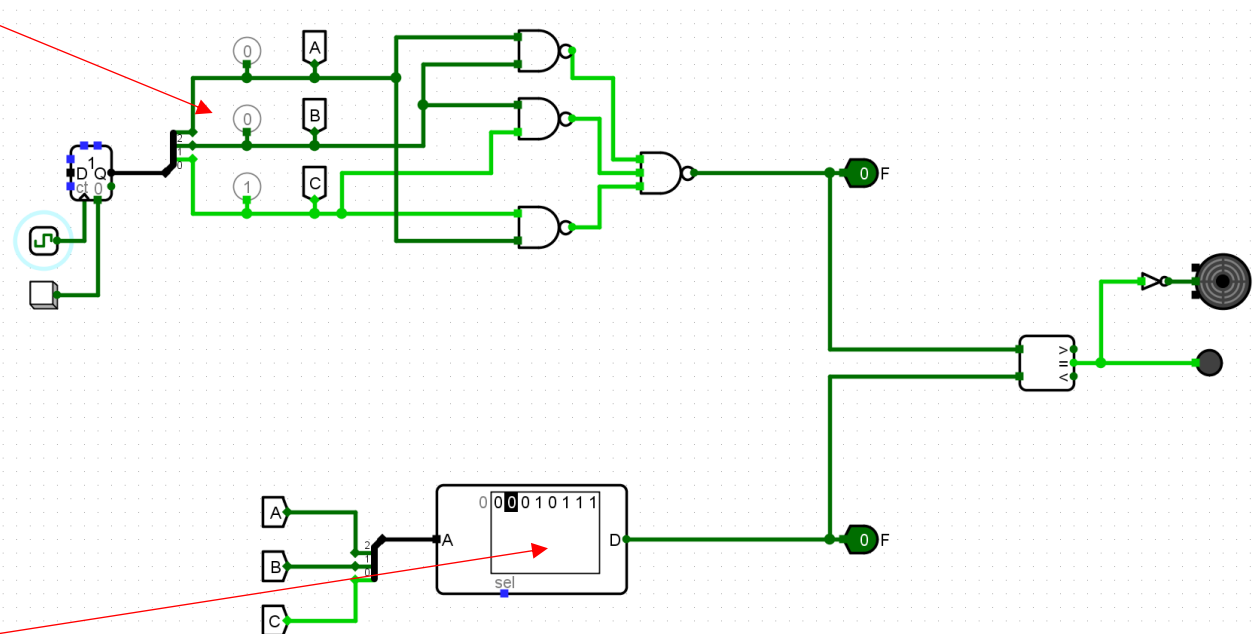


表 4.4 真值表

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



## • 5、（验证实验）例题4.5的实现

- 在Logisim上实现例题4.5的比较两个3位二进制数是否相等的**数值比较器电路**，该电路有6个输入、1个输出。

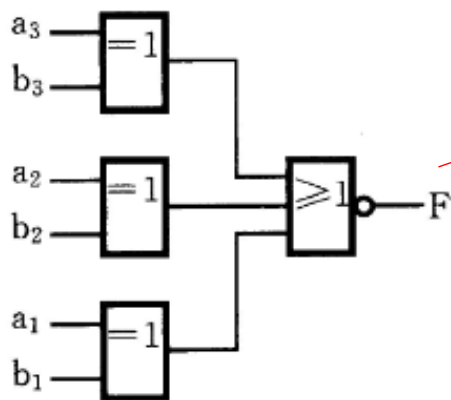
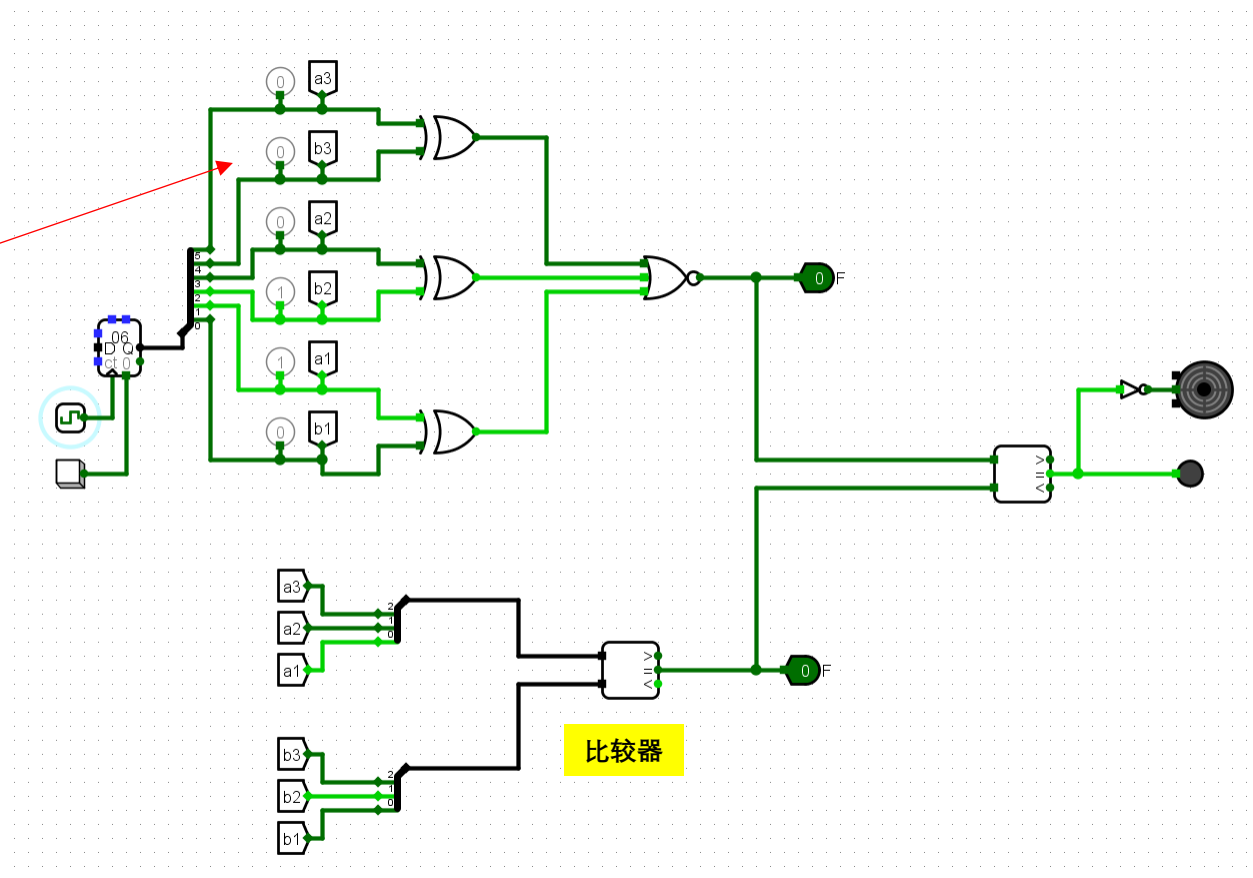


图 4.6 逻辑电路





## • 6、（验证实验）例题4.6的实现

- 在Logisim上实现例题4.6的**乘法器电路**（有2种不同的实现方式），该电路用于产生两个2位二进制数相乘的积，该电路有4个输入、4个输出。

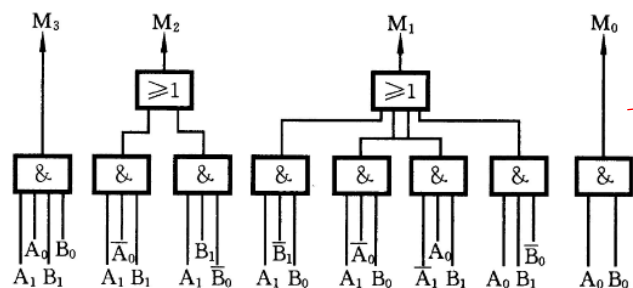


图 4.7 方法 I 的逻辑电路

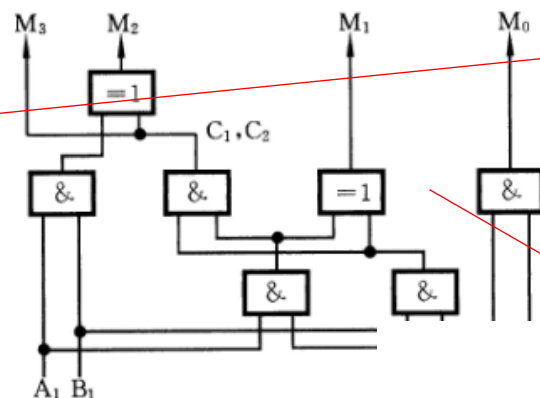
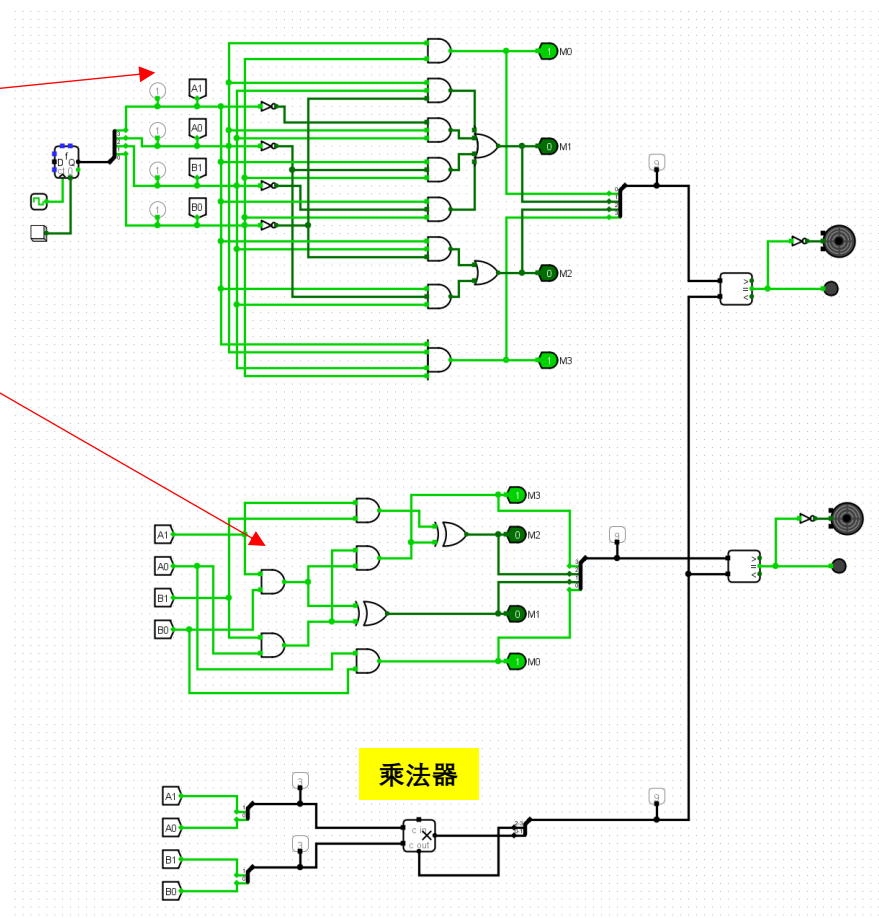


图 4.8 方法 II 的逻辑电路

表 4.5 真值表

A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	M <sub>3</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>	A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	M <sub>3</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1	0	1	0	0	1	0	0
0	0	1	1	0	0	0	0	1	0	1	1	0	1	1	0
0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1	1	1	0	1	0	0	1	1
0	1	1	0	0	0	1	0	1	1	1	0	0	1	1	0
0	1	1	1	0	0	1	1	1	1	1	1	1	0	0	1



## • 7、（验证实验）例题4.7的实现

- 在Logisim上实现例题4.7的判别以余3码表示的1位十进制数**是否为合数的电路**，该电路有4个输入、1个输出。

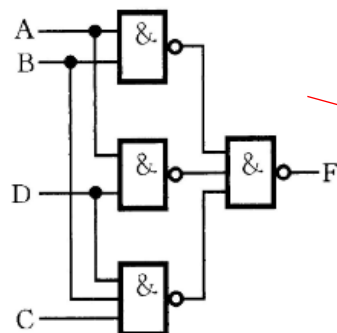
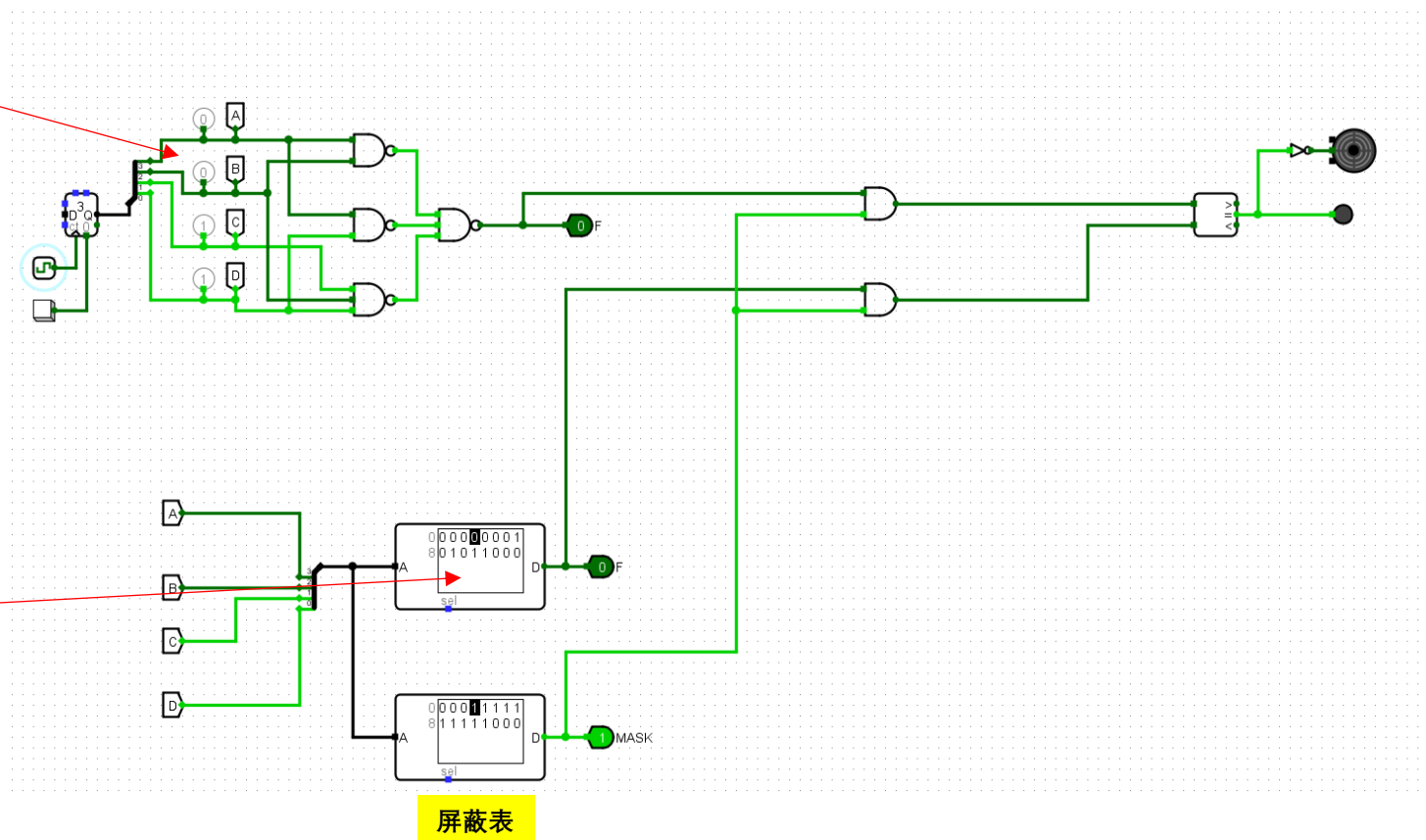


图 4.10 逻辑电路

表 4.6 真值表

A	B	C	D	F	A	B	C	D	F
0	0	0	0	d	1	0	0	0	0
0	0	0	1	d	1	0	0	1	1
0	0	1	0	d	1	0	1	0	0
0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	1	1	0	0	1
0	1	0	1	0	1	1	0	1	d
0	1	1	0	0	1	1	1	0	d
0	1	1	1	1	1	1	1	1	d



## • 8、（验证实验）例题4.8的实现

- 在Logisim上实现例题4.8的**组合逻辑电路**（有2种不同的实现方式），该电路有3个输入、2个输出。

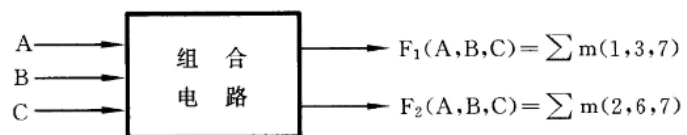
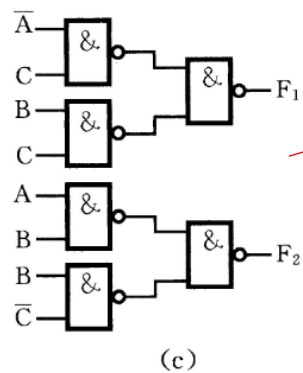
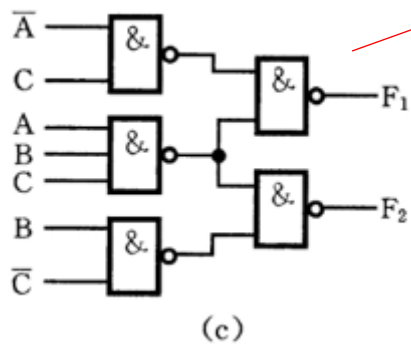


图 4.11 组合逻辑电路结构框图



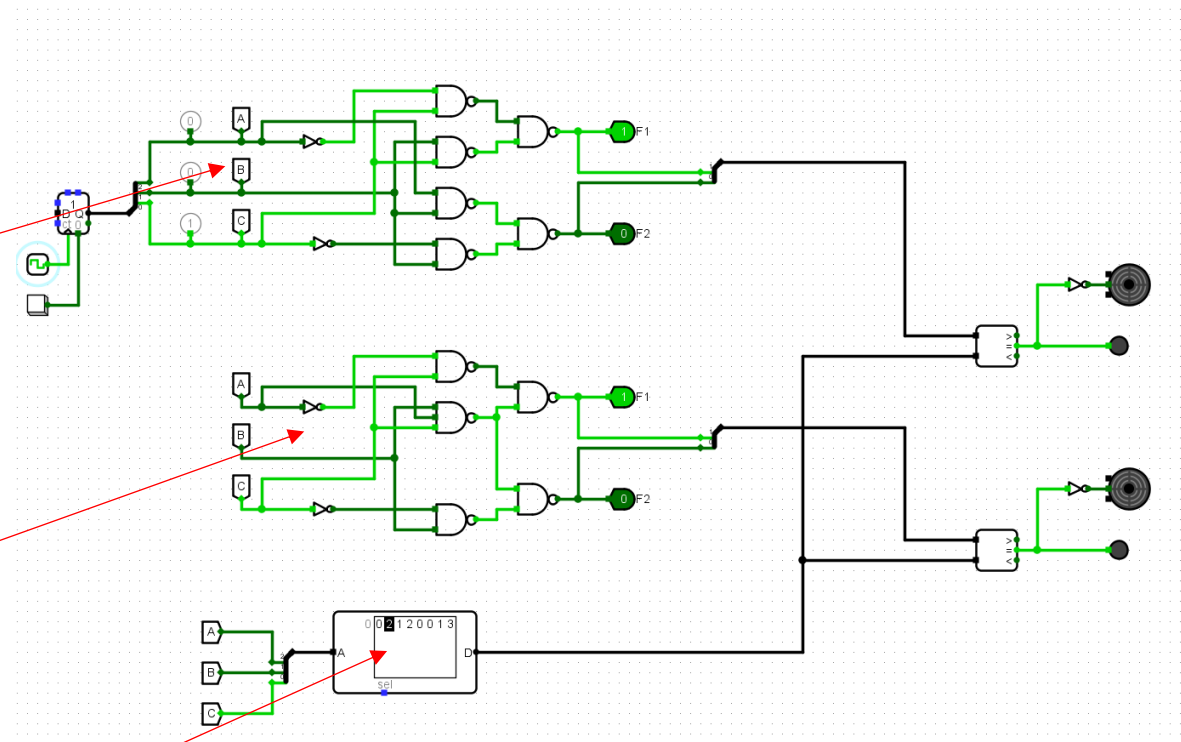
(c)



(c)

真值表

A	B	C	F1	F2
0	0	0	0	0
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	1
1	1	1	1	1



- 9、(验证实验) 例题4.9的实现

- 在Logisim上实现例题4.9的**全加器FA电路**（有2种不同的实现方式），该电路有3个输入、2个输出。

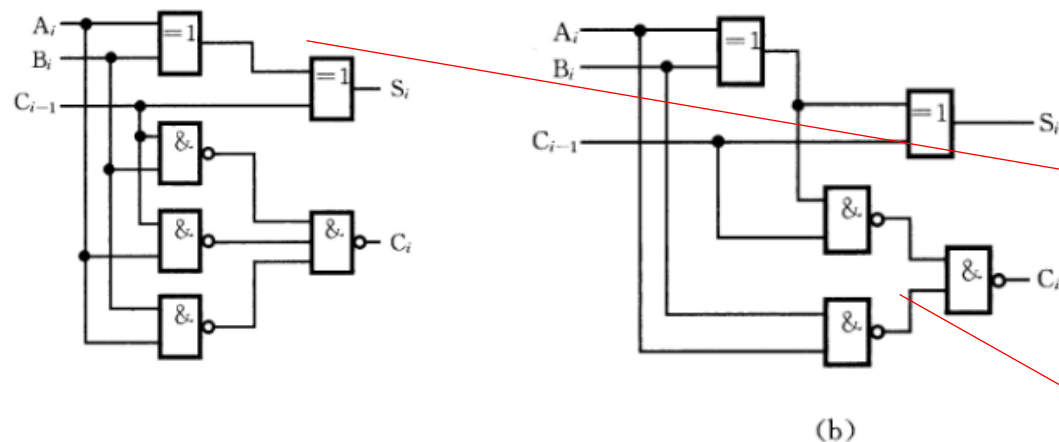
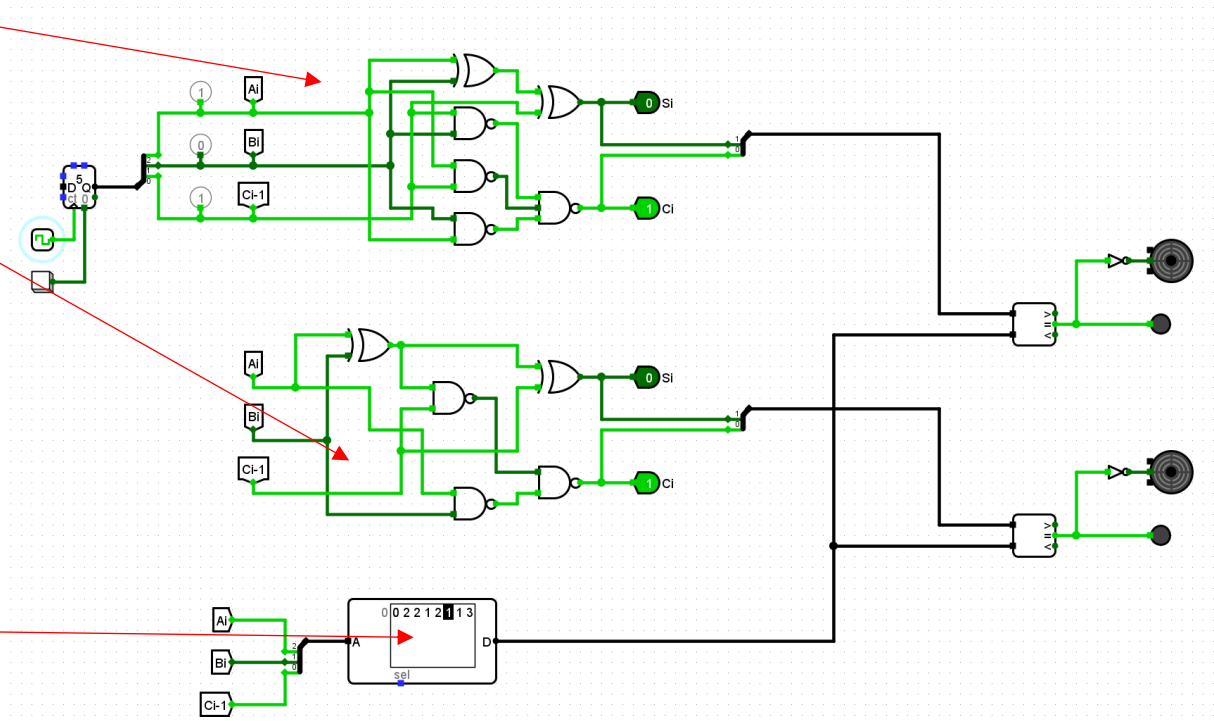


表 4.7 全加器真值表

$A_i$	$B_i$	$C_{i-1}$	$S_i$	$C_i$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



## • 10、（验证实验）例题4.10的实现

- 在Logisim上实现例题4.10的**组合逻辑电路**（有2种不同的实现方式），该电路有4个输入、1个输出。

- $F = \neg A \cdot B + B \cdot \neg C + A \cdot \neg B \cdot C + A \cdot C \cdot \neg D$

- $F = \neg(\neg(B \cdot \neg(A \cdot C)) + \neg(A \cdot C \cdot \neg(B \cdot D)))$

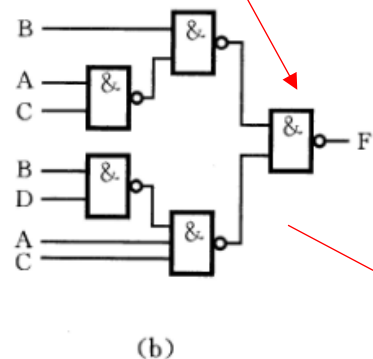
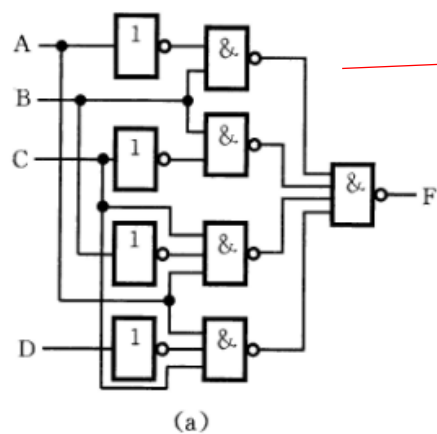
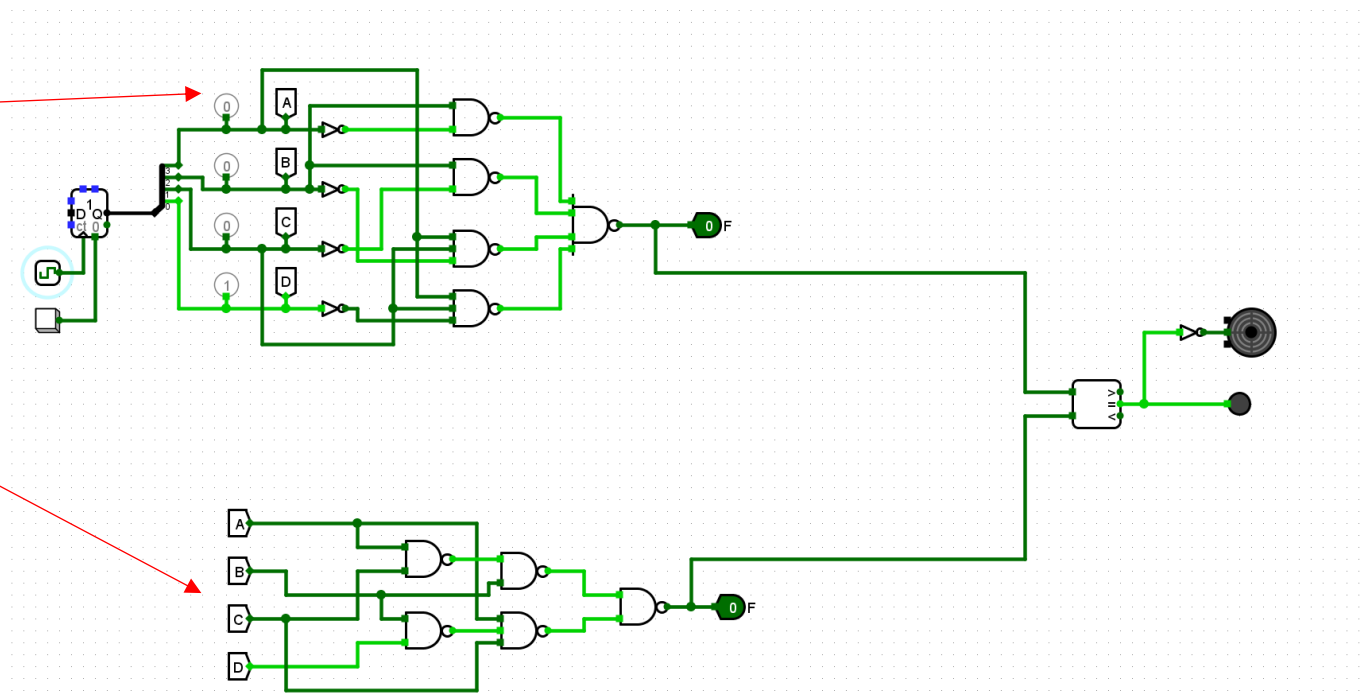


图 4.17 逻辑电路



## • 11、（验证实验）例题4.11的实现

- 在Logisim上实现例题4.11的**组合逻辑电路**（有2种不同的实现方式），该电路有3个输入、1个输出。

- $F = \neg A \cdot B + B \cdot \neg C + A \cdot \neg B \cdot C$

- $F = (A \cdot C) \oplus B$

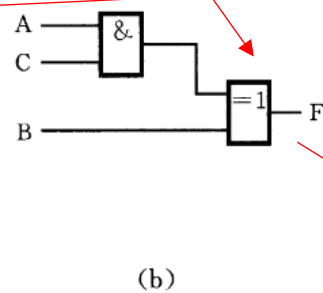
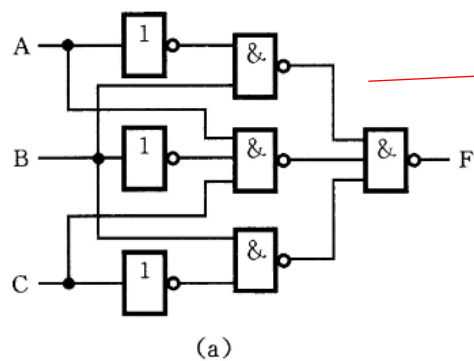
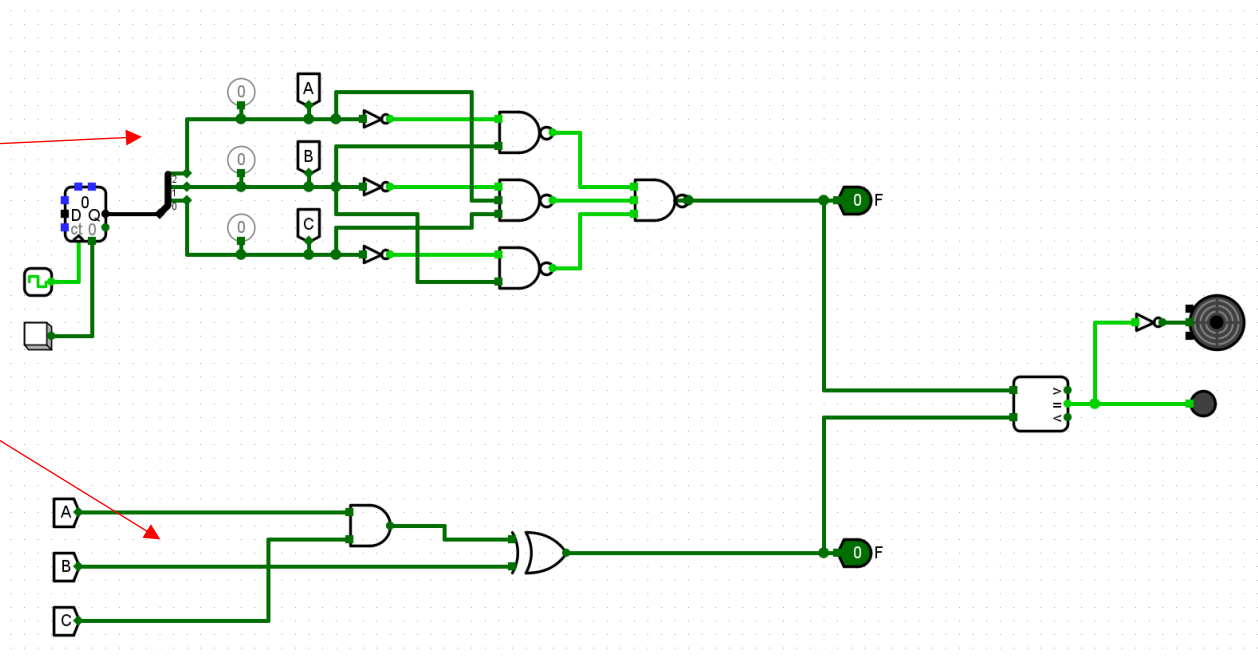


图 4.18 逻辑电路



- 12、(验证实验) 例题4.12的实现 (a)

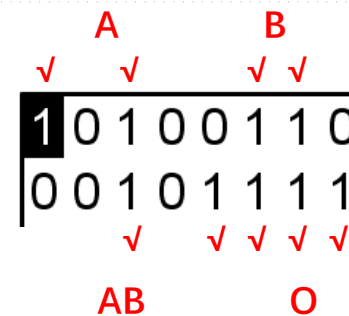
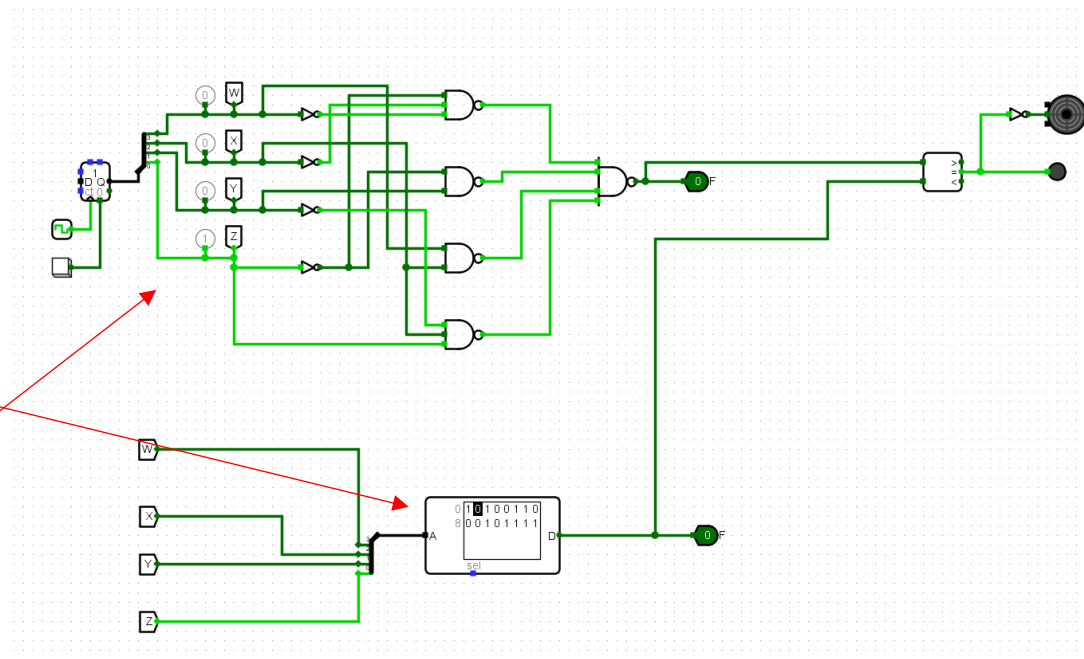
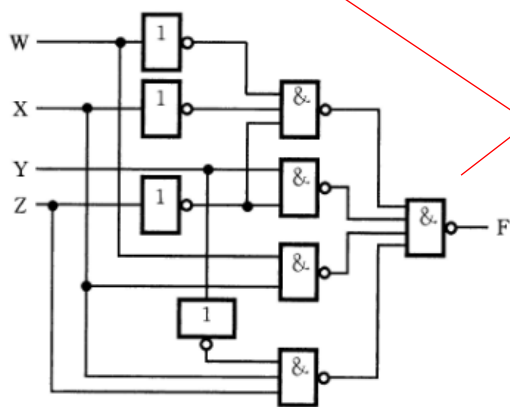
- 在Logisim上实现例题4.12的判断献血者与受血者**血型是否相容的电路**（采用血型编码I），该电路有4个输入、1个输出。

表 4.8 血型相容规则表

受血 献 血	A	B	AB	O
A	✓		✓	
B		✓	✓	
AB			✓	
O	✓	✓	✓	✓

表 4.9 血型编码(1)

血 型	献		受	
	W	X	Y	Z
A	0	0	0	0
B	0	1	0	1
AB	1	0	1	0
O	1	1	1	1



# • 13、（验证实验）例题4.12的实现（b）

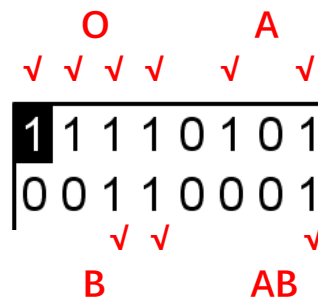
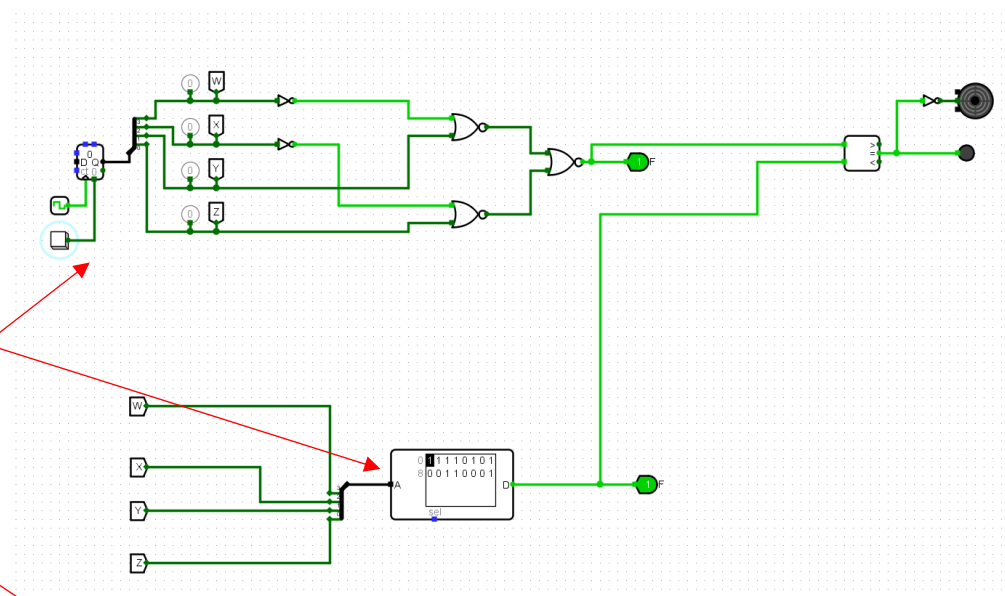
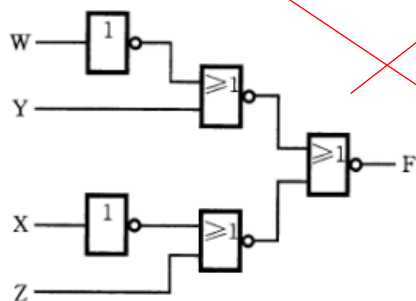
- 在Logisim上实现例题4.12的判断献血者与受血者**血型是否相容的电路**（采用血型编码II），该电路有4个输入、1个输出。

表 4.8 血型相容规则表

受血 献血	A	B	AB	O
A	✓		✓	
B		✓	✓	
AB			✓	
O	✓	✓	✓	✓

表 4.10 血型编码(2)

血 型	献		受	
	W	X	Y	Z
O	0	0	0	0
A	0	1	0	1
B	1	0	1	0
AB	1	1	1	1





## • 14、（验证实验）例题4.16的实现

- 在Logisim上实现例题4.16增加冗余项前的组合逻辑电路（左下图）和增加冗余项后的组合逻辑电路（图4.23），该电路有3个输入、1个输出。

- $F = A \cdot B + \overline{A} \cdot C$

- $F = A \cdot B + \overline{A} \cdot C + B \cdot C$

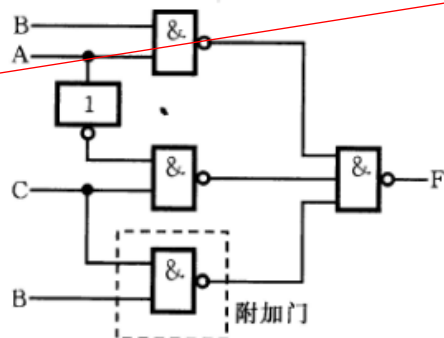
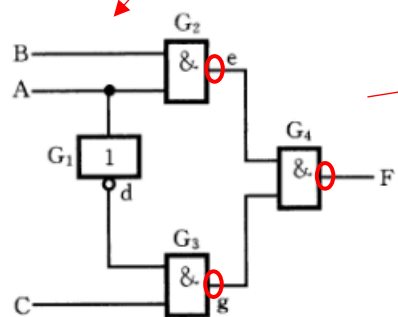
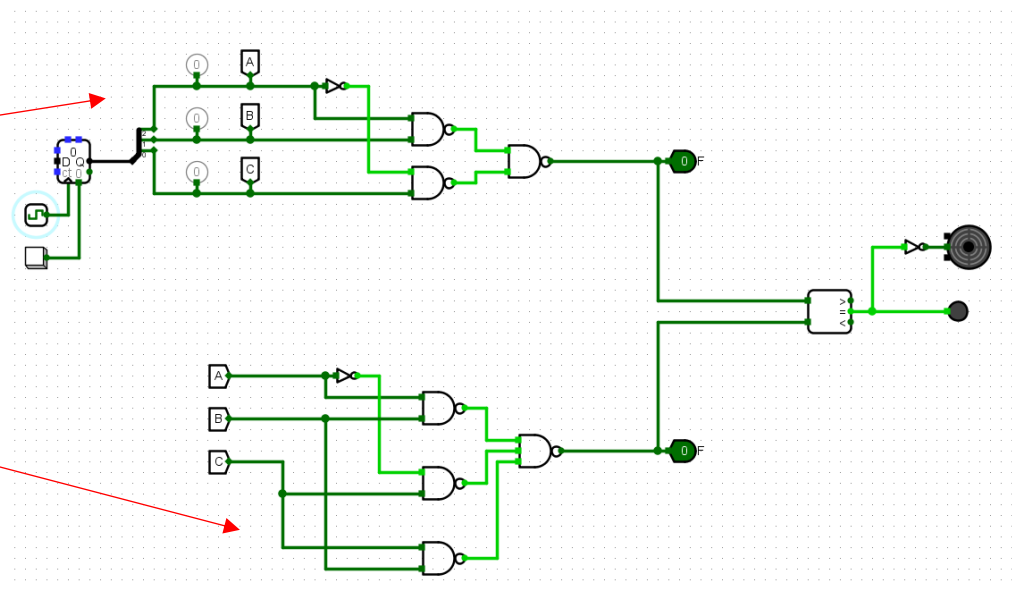


图 4.23 增加冗余项后的逻辑电路

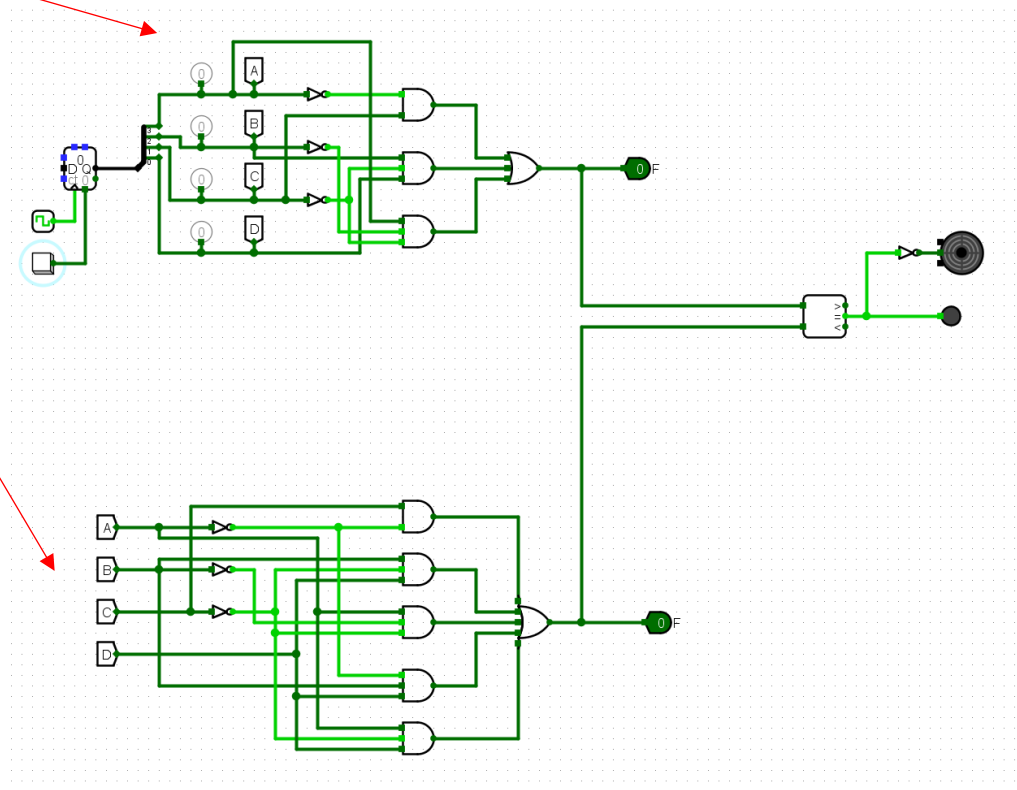


## • 15、（验证实验）例题4.17的实现

- 在Logisim上实现例题4.17**增加冗余项前**的组合逻辑电路和**增加冗余项后**的组合逻辑电路，该电路有4个输入、1个输出。

- $F = \neg A \cdot C + B \cdot \neg C \cdot D + A \cdot \neg B \cdot \neg C$

- $F = \neg A \cdot C + B \cdot \neg C \cdot D + A \cdot \neg B \cdot \neg C + \neg A \cdot B \cdot D + A \cdot \neg C \cdot D$



## (二) 在Logisim上实现教材第4章的习题

### • 1、（验证实验）习题4.2的实现

- 在Logisim上实现习题4.2的电路（图4.28）以及改用异或门实现的电路，该电路有3个输入，1个输出。

4.2 分析图 4.28 所示的组合逻辑电路：(1) 指出在哪些输入取值下，输出 F 的值为 1；  
(2) 改用异或门实现该电路的逻辑功能。

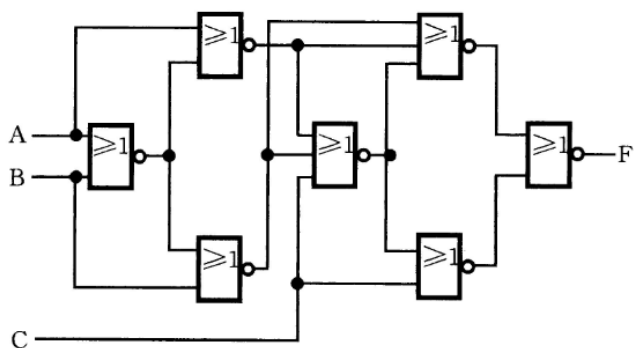
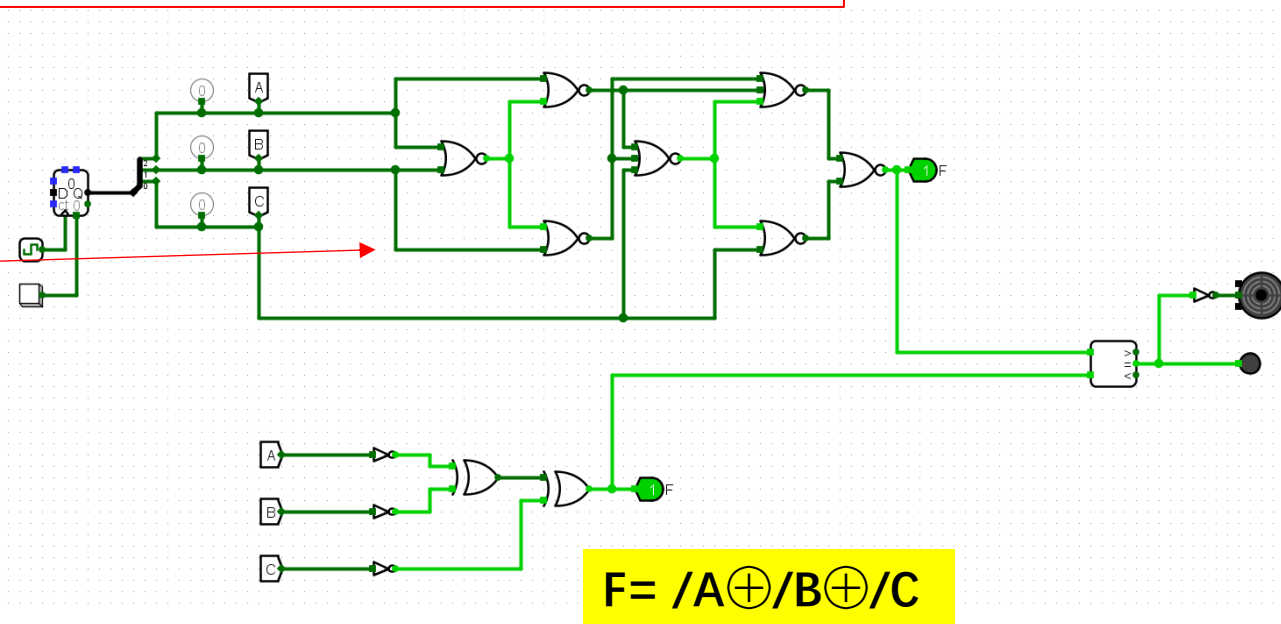


图 4.28 组合逻辑电路



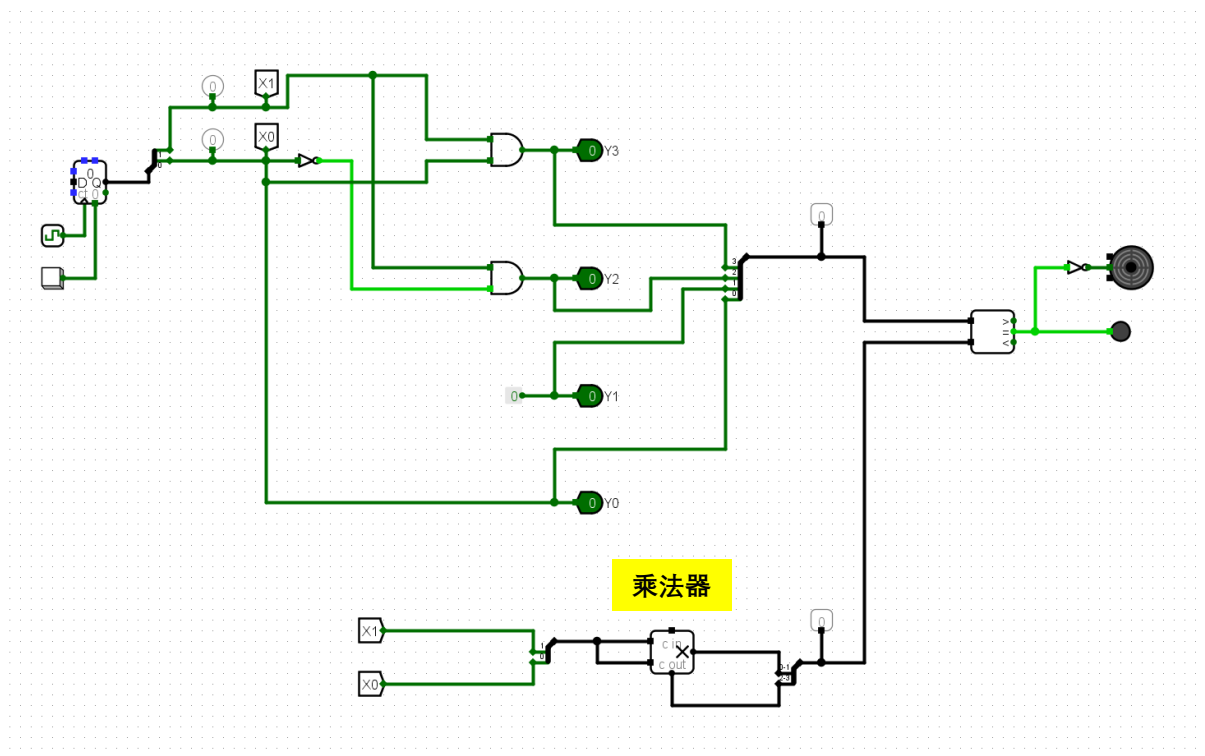
$$F = \neg A \oplus \neg B \oplus C$$

## • 2、（验证实验）习题4.6（1）的实现

- 在Logisim上实现习题4.6（1）要求的组合逻辑电路的功能，该电路有2个输入，4个输出。

4.6 假定  $X=AB$  代表一个 2 位二进制数，试设计满足如下要求的逻辑电路（Y 也用二进制数表示）：

(1)  $Y=X^2$

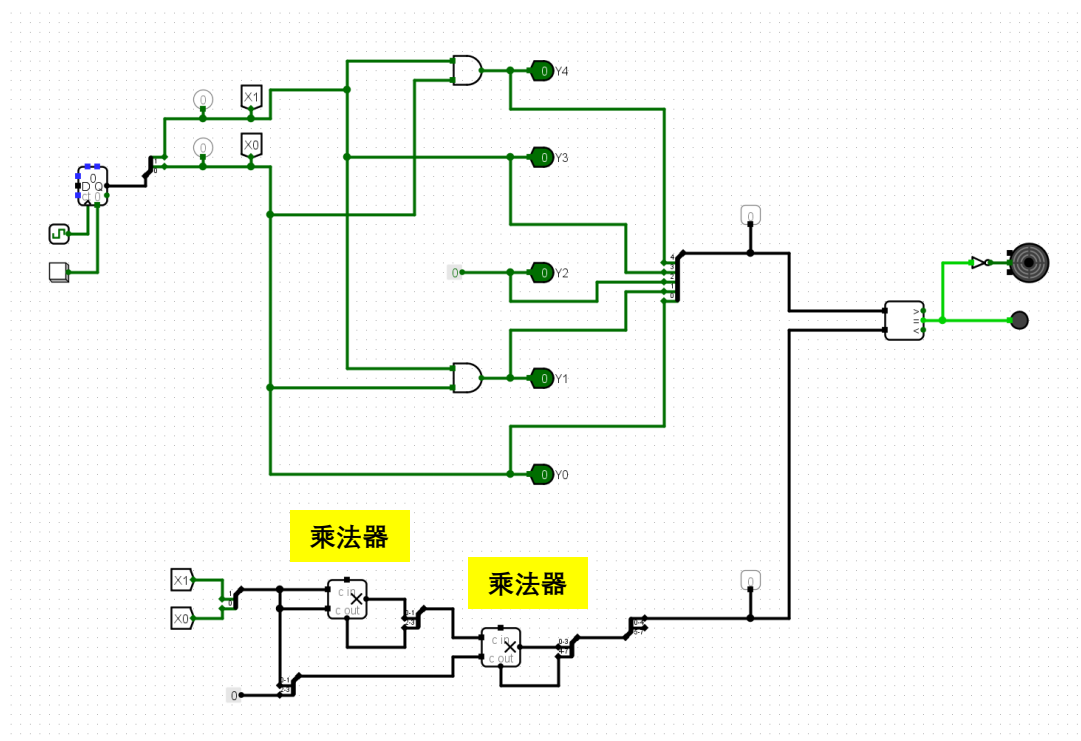


### • 3、（验证实验）习题4.6（2）的实现

- 在Logisim上实现习题4.6（2）要求的组合逻辑电路的功能，该电路有2个输入，5个输出。

4.6 假定  $X=AB$  代表一个 2 位二进制数，试设计满足如下要求的逻辑电路（Y 也用二进制数表示）：

$$(2) Y=X^3$$



## • 4、（验证实验）习题4.7的实现

- 在Logisim上用与非门实现习题4.7要求的组合逻辑电路的功能，该电路有4个输入，1个输出。

4.7 用与非门设计一个组合逻辑电路,该电路输入为 1 位十进制数的 2421 码,当输入的数字为素数时,输出 F 为 1,否则 F 为 0。

0-9中：2、3、5、7为素数

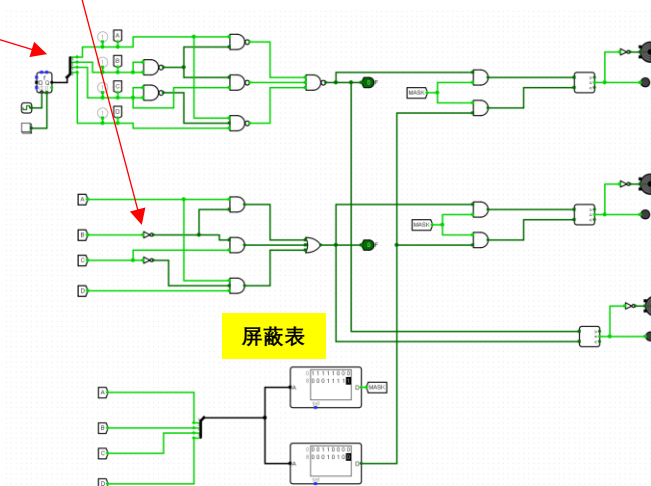
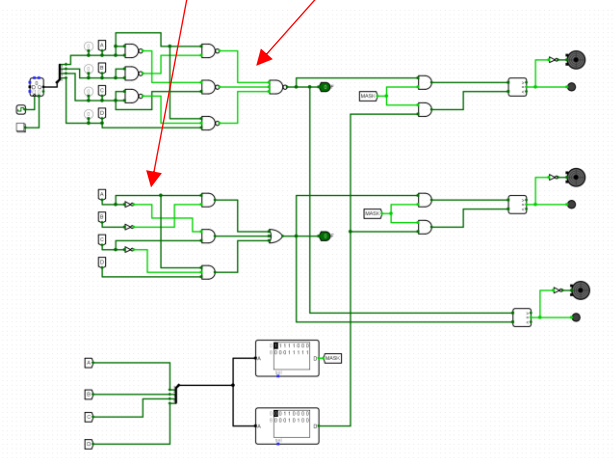
- 答：  $F = A \cdot B + \bar{A} \cdot C + A \cdot \bar{C} \cdot D$       或者       $F = A \cdot B + \bar{B} \cdot C + A \cdot \bar{C} \cdot D$

“与非门”实现

- $F = A \cdot B + \bar{A} \cdot C + A \cdot \bar{C} \cdot D = \overline{\overline{A \cdot B} \cdot \overline{\bar{A} \cdot C} \cdot \overline{A \cdot \bar{C} \cdot D}} = \overline{\overline{A \cdot B} \cdot \overline{\bar{A} \cdot C} \cdot \overline{A \cdot \bar{C} \cdot D}}$
- $F = A \cdot B + \bar{B} \cdot C + A \cdot \bar{C} \cdot D = \overline{\overline{A \cdot B} \cdot \overline{\bar{B} \cdot C} \cdot \overline{A \cdot \bar{C} \cdot D}} = \overline{\overline{A \cdot B} \cdot \overline{\bar{B} \cdot C} \cdot \overline{A \cdot \bar{C} \cdot D}}$

表 1.3 常用的 3 种 BCD 码

十进制字符	8421 码	2421 码	余 3 码
0	0000	0000	0011
1	0001	0001	0100
2	0010	0010	0101
3	0011	0011	0110
4	0100	0100	0111
5	0101	1011	1000
6	0110	1100	1001
7	0111	1101	1010
8	1000	1110	1011
9	1001	1111	1100



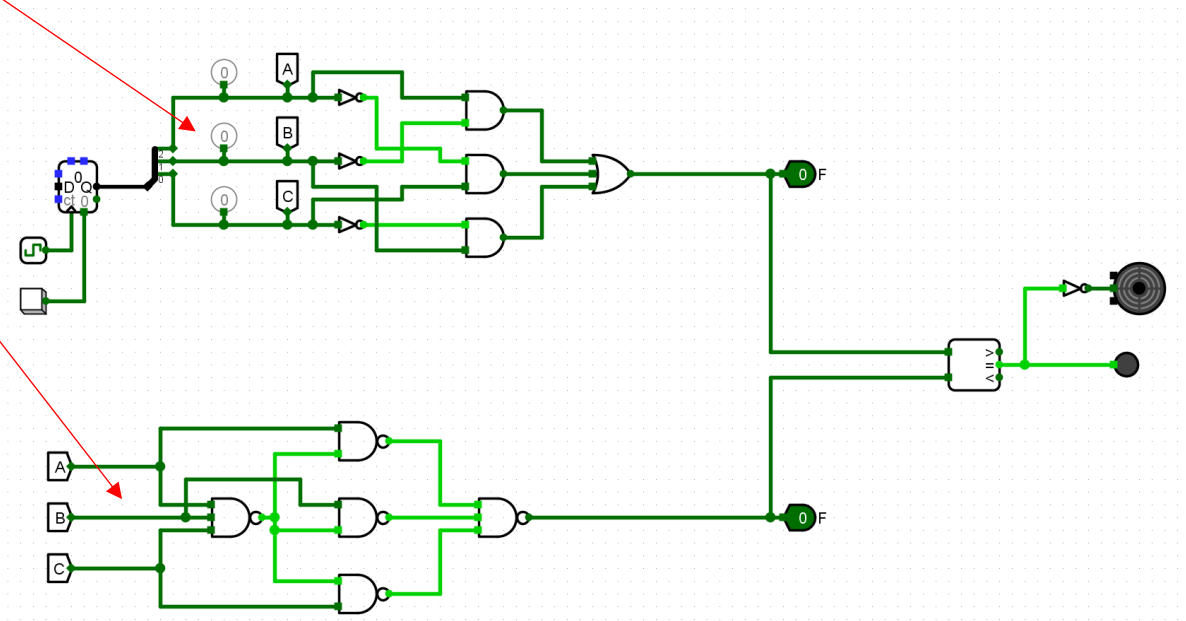
## • 5、（验证实验）习题4.11（1）的实现

- 在Logisim上实现习题4.11（1）要求的组合逻辑电路（ $F=A\cdot B+\bar{A}\cdot C+B\cdot \bar{C}$ ），以及用与非门实现的电路，该电路有3个输入，1个输出。

4.11 在输入不提供反变量的情况下,用与非门组成实现下列函数的最简电路。

(1)  $F=A\bar{B}+\bar{A}C+B\bar{C}$

- 答:  $F = \overline{\overline{A\cdot B\cdot C}} \cdot \overline{\overline{B\cdot A\cdot C}} \cdot \overline{\overline{C\cdot A\cdot B}}$



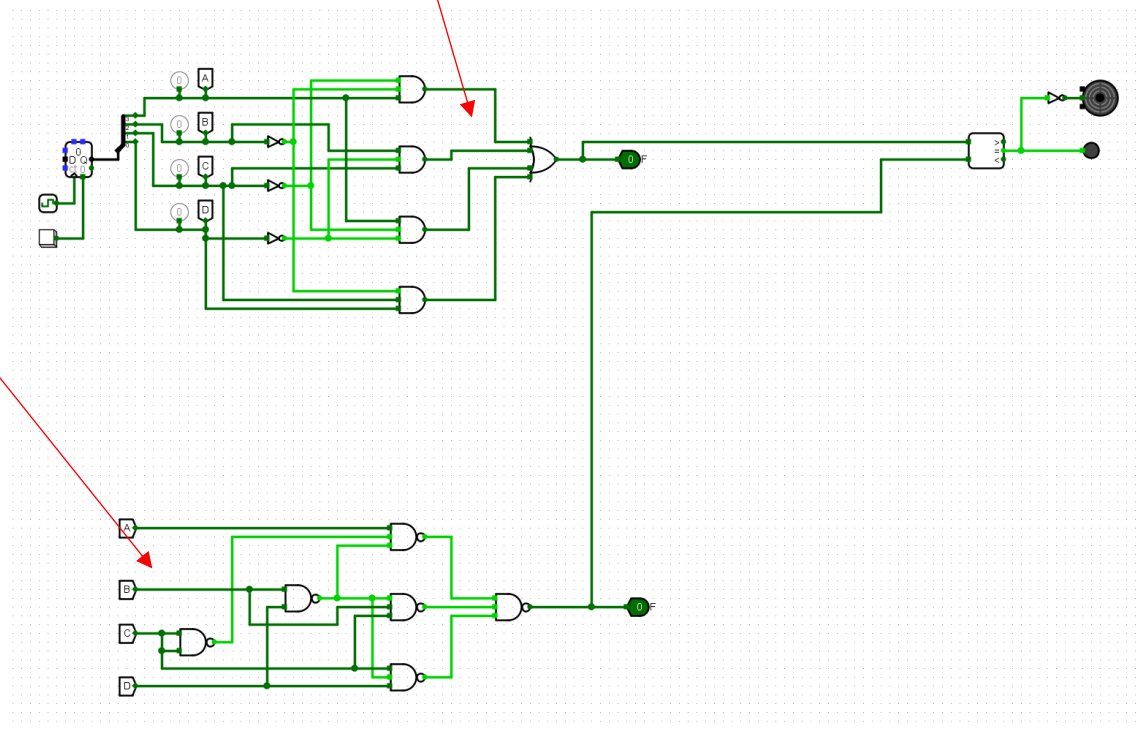
## • 6、（验证实验）习题4.11（2）的实现

- 在Logisim上实现习题4.11（2）要求的组合逻辑电路（ $F=A\cdot B\cdot C+B\cdot C\cdot D+A\cdot C\cdot D+B\cdot C\cdot D$ ），以及用与非门实现的电路，该电路有4个输入，1个输出。

4.11 在输入不提供反变量的情况下,用与非门组成实现下列函数的最简电路。

$$(2) F = A\bar{B}\bar{C} + B\bar{C}\bar{D} + A\bar{C}\bar{D} + \bar{B}CD$$

- 答:  $F = \neg(\neg(\neg(B\cdot D)\cdot A)\neg(C\cdot C)) \cdot \neg(\neg(B\cdot D)\cdot B\cdot C) \cdot \neg(\neg(B\cdot D)\cdot C\cdot D)$





## • 7、（设计实验，课后完成）习题4.1的实现

- 在Logisim上实现习题4.1的电路（图4.27）以及其简化电路，该电路有3个输入，1个输出。

4.1 分析图 4.27 所示的组合逻辑电路,说明电路功能,并画出其简化逻辑电路图。

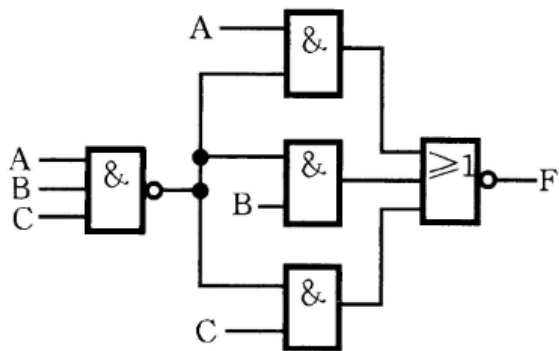


图 4.27 组合逻辑电路

## • 8、（设计实验，课后完成）习题4.3的实现

- 在Logisim上实现习题4.3的电路（图4.29），该电路有4个输入，4个输出。

4.3 分析图 4.29 所示组合逻辑电路,列出真值表,说明该电路的逻辑功能。

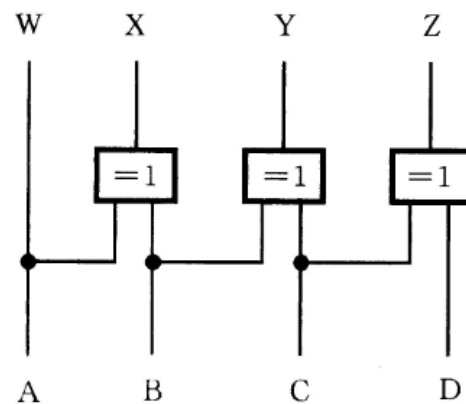


图 4.29 组合逻辑电路

## • 9、（设计实验，课后完成）习题4.4的实现

- 在Logisim上实现习题4.4要求的组合逻辑电路的功能，该电路有4个输入，1个输出。

4.4 设计一个组合逻辑电路,该电路输入端接收两个 2 位二进制数  $A=A_2A_1, B=B_2B_1$ 。当  $A>B$  时,输出  $Z=1$ ,否则  $Z=0$ 。

- 10、（设计实验，课后完成）习题4.5的实现

- 在Logisim上实现习题4.5要求的代码转换电路的功能，该电路有4个输入，4个输出。

4.5 设计一个代码转换电路,将 1 位十进制数的余 3 码转换成 2421 码。

## • 11、（设计实验，课后完成）习题4.8的实现

- 在Logisim上实现习题4.8要求的组合逻辑电路的功能，该电路有4个输入，1个输出。

4.8 设计一个“四舍五入”电路。该电路输入为1位十进制数的8421码，当其值大于或等于5时，输出F的值为1，否则F的值为0。

## • 12、（设计实验，课后完成）习题4.9的实现

- 在Logisim上实现习题4.9要求的组合逻辑电路的功能，该电路有4个输入，1个输出。

4.9 设计一个检测电路,检测 4 位二进制码中 1 的个数是否为偶数。若为偶数个 1,则输出为 1,否则输出为 0。

## • 13、（设计实验，课后完成）习题4.10的实现

- 在Logisim上实现习题4.10要求的加/减法器的功能，该电路有4个输入，2个输出。

4.10 设计一个加/减法器,该电路在  $M$  控制下进行加、减运算。当  $M=0$  时,实现全加器功能;当  $M=1$  时,实现全减器功能。

## • 14、（设计实验，课后完成）习题4.12（3）的实现

- 在Logisim上实现习题4.12（3）要求的组合逻辑电路（ $F_3=(A+\bar{B})\cdot(\bar{A}+\bar{C})$ ，增加冗余项前），以及增加冗余项后的电路。

4.12 下列函数描述的电路是否可能产生险象？在什么情况下产生险象？若产生险象，试用增加冗余项的方法消除。

$$(3) F_3 = (A + \bar{B})(\bar{A} + \bar{C})$$



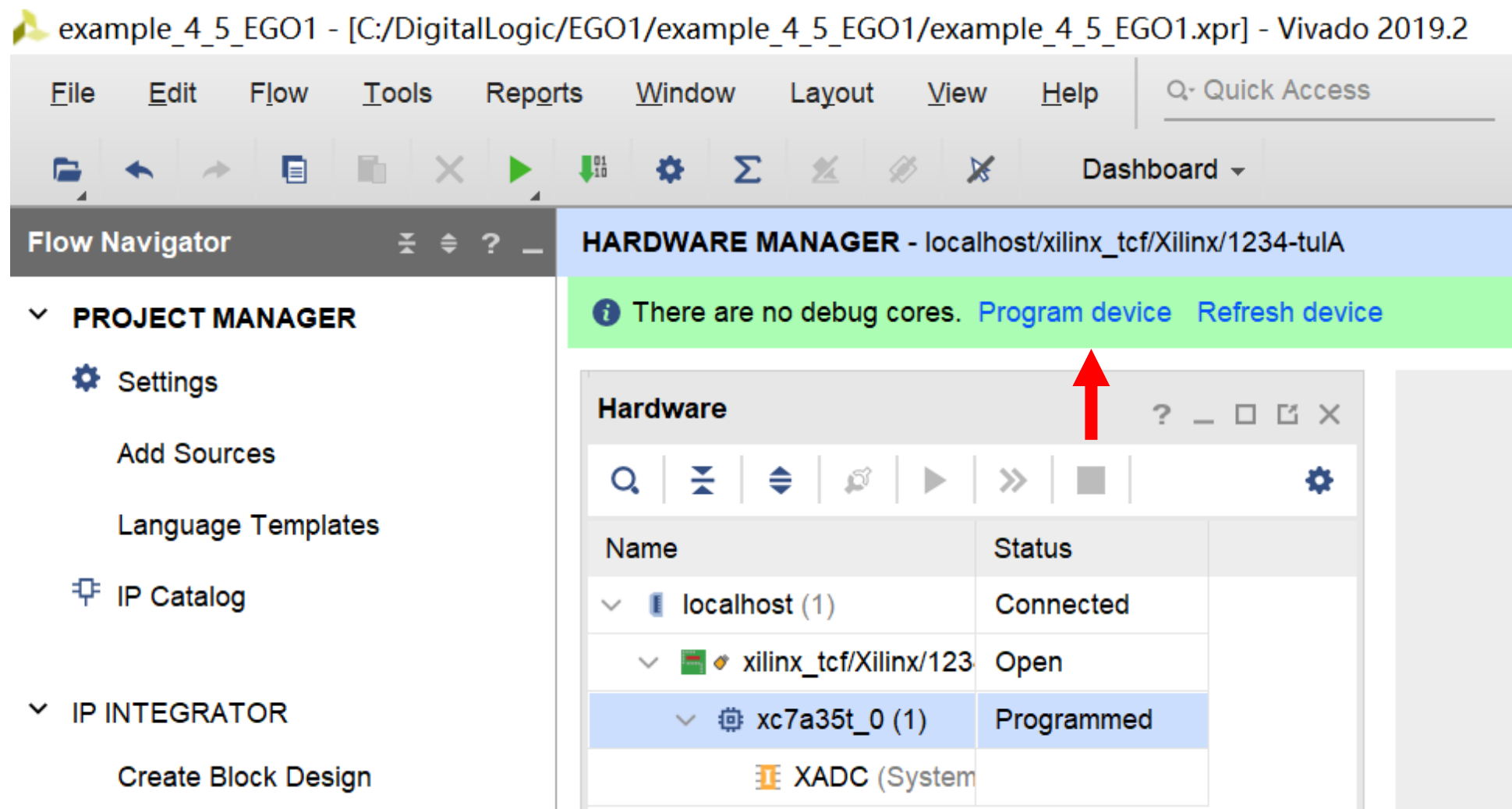
# 第二部分：在FPGA开发板上实现组合逻辑电路

Local Disk (C:) > 数字逻辑 (2024年下半年) > 第3次实验文档 (发给学生) > EGO1

名称	修改日期	类型	大小
example_4_1_1_EGO1	2024/10/6 8:42	文件夹	
example_4_1_EGO1	2024/10/6 8:42	文件夹	
example_4_3_1_EGO1	2024/10/6 8:42	文件夹	
example_4_3_EGO1	2024/10/6 8:42	文件夹	
example_4_5_1_EGO1	2024/10/6 8:42	文件夹	
example_4_5_EGO1	2024/10/6 8:42	文件夹	
example_4_9_1_EGO1	2024/10/6 8:43	文件夹	
example_4_9_2_EGO1	2024/10/6 8:43	文件夹	
example_4_9_EGO1	2024/10/6 8:43	文件夹	
example_4_12a_1_EGO1	2024/10/6 8:42	文件夹	
example_4_12a_EGO1	2024/10/6 8:42	文件夹	
example_4_16_1_EGO1	2024/10/6 8:42	文件夹	
example_4_16_EGO1	2024/10/6 8:42	文件夹	
EGO1.xdc	2023/12/24 13:00	XDC 文件	18 KB
example_4_1_1_EGO1.v	2023/10/26 16:42	V 文件	1 KB
example_4_1_EGO1.v	2023/10/26 16:46	V 文件	2 KB
example_4_3_1_EGO1.v	2023/10/26 21:21	V 文件	3 KB
example_4_3_EGO1.v	2023/10/26 22:26	V 文件	2 KB
example_4_5_1_EGO1.v	2023/10/26 22:25	V 文件	1 KB
example_4_5_EGO1.v	2023/10/26 22:26	V 文件	2 KB
example_4_9_1_EGO1.v	2023/10/29 9:30	V 文件	2 KB
example_4_9_2_EGO1.v	2023/10/29 9:30	V 文件	1 KB
example_4_9_EGO1.v	2023/10/27 10:25	V 文件	2 KB
example_4_12a_1_EGO1.v	2023/10/27 11:48	V 文件	2 KB
example_4_12a_EGO1.v	2023/10/27 11:15	V 文件	2 KB
example_4_16_1_EGO1.v	2023/10/29 10:30	V 文件	1 KB
example_4_16_EGO1.v	2023/10/27 13:00	V 文件	2 KB

请将\第3次实验文档（发给学生）\EGO1\目录中的所有文件，拷贝到实验室电脑（或自己电脑）D:\DigitalLogic\EGO1\目录中

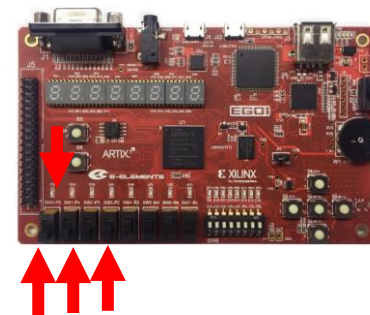
- 如果在下载bit文件时，出现不了这个界面，请将USB线从电脑上拔下来，再插上！



# (一) 在FPGA开发板上实现教材第4章的例题

## • 1、（验证实验）例题4.1的实现

- 分别用结构化描述方式和行为描述方式实现例题4.1的不一致电路，该电路有3个输入、1个输出。
- 输入为开发板上最左边的3个拨动开关，输出为开发板上最左边的LED灯。



EGO1开发板

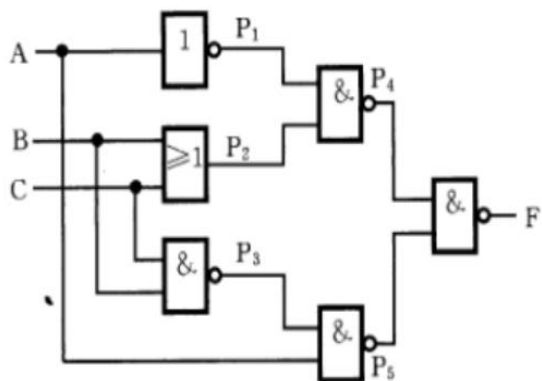


图4.2(a) 逻辑电路

表 4.1 真值表

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

## 在EGO1开发板上实现例题4.1的设计文件：

example\_4\_1\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//结构化描述方式（子模块采用行为描述方式）实现例题4.1的不一致电路，输入为最左边的3个拨动开关，输出为最左边的LED灯。

```
`timescale 1ns / 1ps
```

```
module not_gate(  
    input a,  
    output reg f  
);  
    always @(*) //行为描述方式  
    begin  
        f <= ~a;  
    end  
endmodule
```

```
module or_gate(  
    input a,  
    input b,  
    output reg f  
);  
    always @(*) //行为描述方式  
    begin  
        f <= a | b;  
    end  
endmodule
```

```
module nand_gate(  
    input a,  
    input b,  
    output reg f  
);  
    always @(*) //行为描述方式  
    begin  
        f <= ~(a & b);  
    end  
endmodule
```

```
module example_4_1(  
    input sw_pin[7:0],  
    output [15:0] led_pin  
);  
    wire p1, p2, p3, p4, p5;
```

```
    not_gate U1(a(sw_pin[0]),f(p1));  
    or_gate U2(a(sw_pin[1]),b(sw_pin[2]),f(p2));  
    nand_gate U3(a(sw_pin[1]),b(sw_pin[2]),f(p3));  
    nand_gate U4(a(p1),b(p2),f(p4));  
    nand_gate U5(a(p3),b(sw_pin[0]),f(p5));  
    nand_gate U6(a(p4),b(p5),f(led_pin[0]));  
endmodule
```

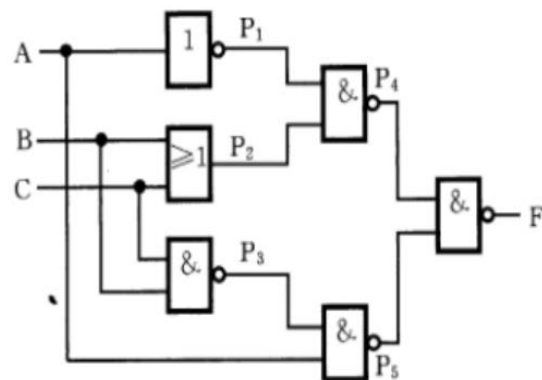


图4.2(a) 逻辑电路

## 结构化描述方式

example\_4\_1\_1\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//采用行为描述方式实现例题4.1的不一致电路，输入为最左边的3个拨动开关，输出为最左边的LED灯。

```
`timescale 1ns / 1ps
```

```
module example_4_1_1(  
    input sw_pin[7:0],  
    output reg [15:0] led_pin  
);
```

## 行为描述方式

```
    always @(*) //行为描述方式  
    begin  
        case({sw_pin[0],sw_pin[1],sw_pin[2]})  
            0: led_pin[0] <= 0; // ABC=000 F=0  
            1: led_pin[0] <= 1; // ABC=001 F=1  
            2: led_pin[0] <= 1; // ABC=010 F=1  
            3: led_pin[0] <= 1; // ABC=011 F=1  
            4: led_pin[0] <= 1; // ABC=100 F=1  
            5: led_pin[0] <= 1; // ABC=101 F=1  
            6: led_pin[0] <= 1; // ABC=110 F=1  
            7: led_pin[0] <= 0; // ABC=111 F=0  
        endcase  
    end  
endmodule
```

表 4.1 真值表

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

## • 2、（验证实验）例题4.3的实现

- 分别用结构化描述方式和行为描述方式实现例题4.3的将8421码转换为余3码的代码转换电路，该电路有4个输入、4个输出。
- 输入为开发板上最左边的4个拨动开关，输出为开发板上最左边的4个LED灯。

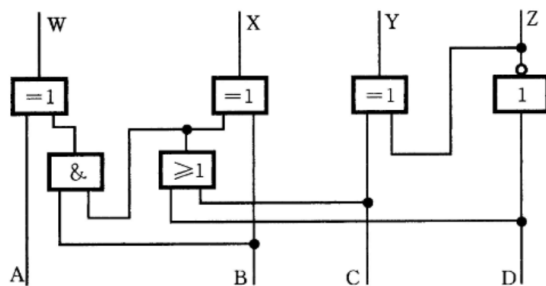
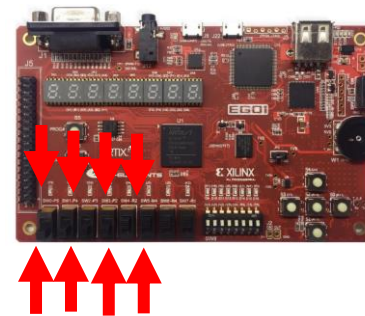


图 4.4 逻辑电路

表 4.3 真值表

ABCD	WXYZ	ABCD	WXYZ
0000	0011	0101	1000
0001	0100	0110	1001
0010	0101	0111	1010
0011	0110	1000	1011
0100	0111	1001	1100



EGO1开发板

在EGO1开发板上实现例题4.3的设计文件：

example\_4\_3\_EGO1.v - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)  
//结构化描述方式(子模块采用行为描述方式)实现例题4.3的不一致电路,输入为最左边的4个拨动开关,输出为最左边的4个LED灯。

```
`timescale 1ns / 1ps

module not_gate(
    input a,
    output reg f
);
    always @(*)
    begin
        f <= ~a;
    end
endmodule

module or_gate(
    input a,
    input b,
    output reg f
);
    always @(*)
    begin
        f <= a | b;
    end
endmodule

module and_gate(
    input a,
    input b,
    output reg f
);
    always @(*)
    begin
        f <= a & b;
    end
endmodule

module xor_gate(
    input a,
    input b,
    output reg f
);
    always @(*)
    begin
        f <= a ^ b;
    end
endmodule
```

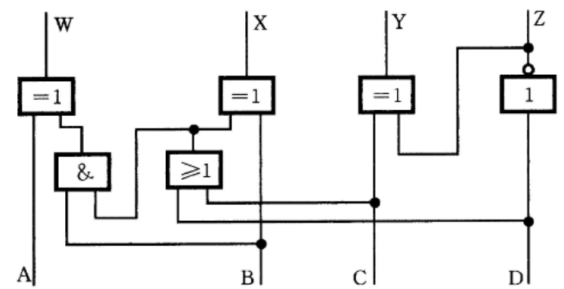


图 4.4 逻辑电路

结构化描述方式

```
module example_4_3(
    input sw_pin[7:0],
    output [15:0] led_pin
);
    wire p1, p2;

    or_gate U2(a(sw_pin[2]),b(sw_pin[3]),f(p1));
    and_gate U1(a(sw_pin[1]),b(p1),f(p2));
    xor_gate U3(a(sw_pin[0]),b(p2),f(led_pin[0]));
    xor_gate U4(a(p1),b(sw_pin[1]),f(led_pin[1]));
    not_gate U6(a(sw_pin[3]),f(led_pin[3]));
    xor_gate U5(a(sw_pin[2]),b(led_pin[3]),f(led_pin[2]));

endmodule
```

行为描述方式

```
example_4_3_1_EGO1.v - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
//采用行为描述方式实现例题4.1的代码转换电路,输入为最左边的4个拨动开关,输出为最左边的4个LED灯。

`timescale 1ns / 1ps

module example_4_3_1(
    input sw_pin[7:0],
    output reg [15:0] led_pin
);
    //8个拨动开关
    //16个led灯

    always @(*)
    begin
        case((sw_pin[0],sw_pin[1],sw_pin[2],sw_pin[3]))
            0: begin
                led_pin[0] <= 0;
                led_pin[1] <= 0;
                led_pin[2] <= 1;
                led_pin[3] <= 1;
            end
            1: begin
                led_pin[0] <= 0;
                led_pin[1] <= 1;
                led_pin[2] <= 0;
                led_pin[3] <= 0;
            end
            2: begin
                led_pin[0] <= 0;
                led_pin[1] <= 1;
                led_pin[2] <= 0;
                led_pin[3] <= 1;
            end
            3: begin
                led_pin[0] <= 0;
                led_pin[1] <= 1;
                led_pin[2] <= 1;
                led_pin[3] <= 0;
            end
            4: begin
                led_pin[0] <= 0;
                led_pin[1] <= 1;
                led_pin[2] <= 1;
                led_pin[3] <= 1;
            end
        endcase
    end
endmodule
```

表 4.3 真值表

ABCD	WXYZ	ABCD	WXYZ
0000	0011	0101	1000
0001	0100	0110	1001
0010	0101	0111	1010
0011	0110	1000	1011
0100	0111	1001	1100

### • 3、（验证实验）例题4.5的实现

- 分别用结构化描述方式和行为描述方式实现例题4.5的比较两个3位二进制数是否相等的数值比较器电路，该电路有6个输入、1个输出。
- 输入为开发板上最左边的3个拨动开关和最右边的3个拨动开关，输出为开发板上最左边的1个LED灯。

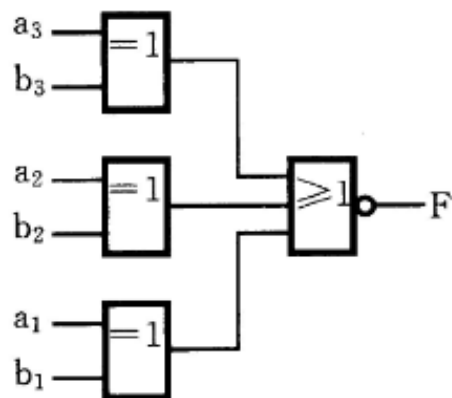


图 4.6 逻辑电路



EGO1开发板

- 在EGO1开发板上实现**例题4.5**的设计文件：

example\_4\_5\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//结构化描述方式（子模块采用行为描述方式）实现例题4.5的比较两个3位二进制数是否相等的数值比较器

```
`timescale 1ns / 1ps
```

```
module nor_gate(
    input a,
    input b,
    input c,
    output reg f
);
    always @(*)           //行为描述方式
    begin
        f <= ~(a | b | c);
    end
endmodule
```

```
module xor_gate(
    input a,
    input b,
    output reg f
);
    always @(*)           //行为描述方式
    begin
        f <= a ^ b;
    end
endmodule
```

```
module example_4_3(
    input sw_pin[7:0],
    output [15:0] led_pin
);
    wire p1, p2, p3;
```

//8个拨动开关  
//16个led灯

```
xor_gate U1(.a(sw_pin[0]),.b(sw_pin[5]),.f(p1));
xor_gate U2(.a(sw_pin[1]),.b(sw_pin[6]),.f(p2));
xor_gate U3(.a(sw_pin[2]),.b(sw_pin[7]),.f(p3));
nor_gate U4(.a(p1),.b(p2),.c(p3),.f(led_pin[0]));
```

//结构化描述方式  
//结构化描述方式  
//结构化描述方式  
//结构化描述方式

endmodule

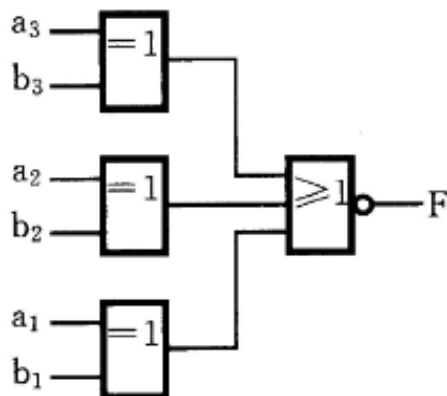


图 4.6 逻辑电路

结构化描述方式

example\_4\_5\_1\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//采用行为描述方式实现例题4.5的数值比较器电路，输入为开发板上最左边的3个拨动开关和

```
`timescale 1ns / 1ps
```

```
module example_4_3_1(
    input sw_pin[7:0],
    output reg [15:0] led_pin
);
    //8个拨动开关
    //16个led灯
```

```
    always @(*)           //行为描述方式
    begin
        if(sw_pin[0]==sw_pin[5] & sw_pin[1]==sw_pin[6] & sw_pin[2]==sw_pin[7])
            led_pin[0] <= 1;
        else
            led_pin[0] <= 0;
        end
    endmodule
```

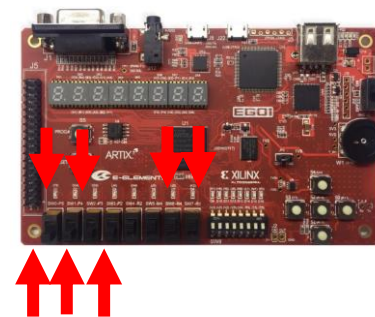
行为描述方式

如果 $a_3a_2a_1=b_3b_2b_1$ ，则 $F=1$ ；否则， $F=0$

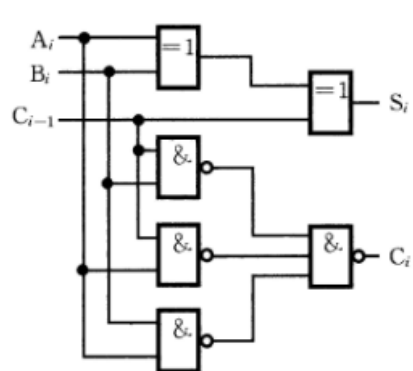


## • 4、（验证实验）例题4.9的实现

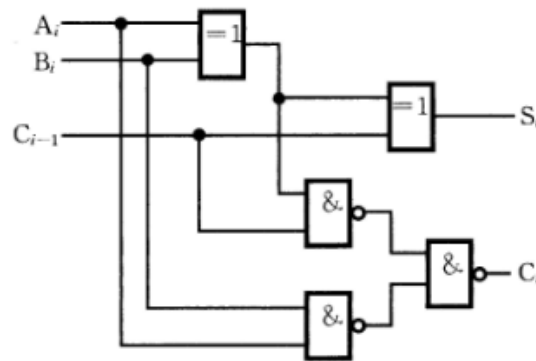
- 请分别用**结构化描述方式**（有2个不同的电路）、**行为描述方式**和**数据流描述方式**实现例题4.9的**全加器FA电路**，该电路有3个输入、2个输出，该实验有**3个小实验**（同时实现2个不同电路的结构化描述方式，行为描述方式，数据流描述方式）。
- （结构化描述方式）** **输入**为开发板上最左边的3个拨动开关，**输出**为开发板上最左边的2个LED灯（电路I）和最右边的2个LED灯（电路II）。
- （行为描述方式）** **输入**为开发板上最左边的3个拨动开关，**输出**为开发板上最左边的2个LED灯。
- （数据流描述方式）** **输入**为开发板上最左边的3个拨动开关，**输出**为开发板上最左边的2个LED灯。



EGO1开发板



电路I



电路II

表 4.7 全加器真值表

$A_i$	$B_i$	$C_{i-1}$	$S_i$	$C_i$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

## 在EGO1开发板上实现例题4.9的设计文件：

example\_4\_9\_EGO1.v - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
begin
    f <= ~(a & b);
end
endmodule
```

```
module nand3_gate(
    input a,
    input b,
    input c,
    output reg f
);
```

```
always @(*) //行为描述方式
begin
    f <= ~(a & b & c);
end
endmodule
```

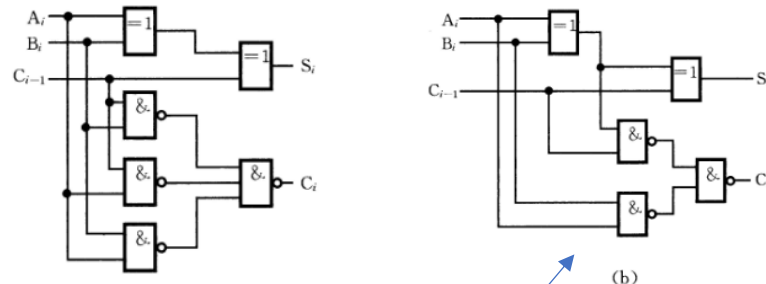
```
module xor_gate(
    input a,
    input b,
    output reg f
);
always @(*) //行为描述方式
begin
    f <= a ^ b;
end
endmodule
```

```
module example_4_9(
    input sw_pin[7:0],
    output [15:0] led_pin
);
wire p1, p2, p3, p4, p5, p6, p7;
```

```
xor_gate U1(a(sw_pin[0]),b(sw_pin[1]),f(p1)); //结构化描述方式
nand2_gate U2(a(sw_pin[0]),b(sw_pin[1]),f(p2)); //结构化描述方式
nand2_gate U3(a(sw_pin[0]),b(sw_pin[2]),f(p3)); //结构化描述方式
nand2_gate U4(a(sw_pin[1]),b(sw_pin[2]),f(p4)); //结构化描述方式
xor_gate U5(a(p1),b(sw_pin[2]),f(led_pin[0])); //结构化描述方式
nand3_gate U6(a(p2),b(p3),c(p4),f(led_pin[1])); //结构化描述方式
```

```
xor_gate U7(a(sw_pin[0]),b(sw_pin[1]),f(p5)); //结构化描述方式
nand2_gate U8(a(p5),b(sw_pin[2]),f(p6)); //结构化描述方式
nand2_gate U9(a(sw_pin[0]),b(sw_pin[1]),f(p7)); //结构化描述方式
xor_gate U10(a(p5),b(sw_pin[2]),f(led_pin[6])); //结构化描述方式
nand2_gate U11(a(p6),b(p7),f(led_pin[7])); //结构化描述方式
```

```
endmodule
```



## 结构化描述方式

example\_4\_9\_1\_EGO1.v - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)  
//采用行为描述方式实现例题4.9的全加器FA电路，输入为开发板上最左边的3个拨

```
`timescale 1ns / 1ps

module example_4_9_1(
    input sw_pin[7:0], //8个拨动开关
    output reg [15:0] led_pin //16个led灯
);
```

```
always @(*) //行为
begin
    case({sw_pin[0],sw_pin[1],sw_pin[2]})
    0: begin
        led_pin[0] <= 0;
        led_pin[1] <= 0;
    end
    1: begin
        led_pin[0] <= 1;
        led_pin[1] <= 0;
    end
    2: begin
        led_pin[0] <= 1;
        led_pin[1] <= 0;
    end
    3: begin
        led_pin[0] <= 0;
        led_pin[1] <= 1;
    end
    4: begin
        led_pin[0] <= 1;
        led_pin[1] <= 0;
    end
    5: begin
        led_pin[0] <= 0;
        led_pin[1] <= 1;
    end
    6: begin
        led_pin[0] <= 0;
        led_pin[1] <= 1;
    end
    7: begin
        led_pin[0] <= 1;
        led_pin[1] <= 1;
    end
    end
endcase
endmodule
```

## 行为描述方式

表 4.7 全加器真值表

$A_i$	$B_i$	$C_{i-1}$	$S_i$	$C_i$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

example\_4\_9\_2\_EGO1.v - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//采用数据流描述方式实现例题4.9的全加器FA电路，输入为开发板上最左边的3个

```
`timescale 1ns / 1ps
```

```
module example_4_9_1(
    input sw_pin[7:0],
    output [15:0] led_pin
);
```

```
assign {led_pin[1], led_pin[0]} = sw_pin[0] + sw_pin[1] + sw_pin[2];
```

```
endmodule
```

## 数据流描述方式

```
assign {led_pin[1], led_pin[0]} = sw_pin[0] + sw_pin[1] + sw_pin[2];
```

## • 5、（验证实验）例题4.12的实现（a）

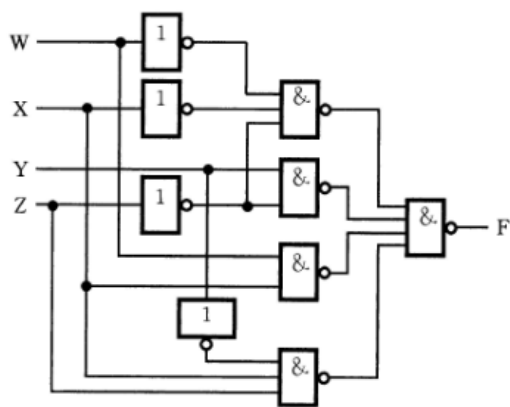
- 请分别用**结构化描述方式**和**行为描述方式**实现例题4.12的判断献血者与受血者**血型是否相容的电路**（采用血型编码I），该电路有4个输入、1个输出。
- 输入**为开发板上最左边的2个拨动开关和最右边的2个开关，**输出**为开发板上最左边的LED灯。

表 4.8 血型相容规则表

受血 献血	A	B	AB	O
A	✓		✓	
B		✓	✓	
AB			✓	
O	✓	✓	✓	✓

表 4.9 血型编码(1)

血 型	献		受	
	W	X	Y	Z
A	0	0	0	0
B	0	1	0	1
AB	1	0	1	0
O	1	1	1	1



EGO1开发板

在EGO1开发板上实现例题4.12 (a) 的设计文件:

example\_4\_12a\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
always @(*)           //行为描述方式
begin
    f <= ~(a & b & c);
end
endmodule
```

```
module nand4_gate(
    input a,
    input b,
    input c,
    input d,
    output reg f
);
always @(*)           //行为描述方式
begin
    f <= ~(a & b & c & d);
end
endmodule
```

```
module not_gate(
    input a,
    output reg f
);
always @(*)           //行为描述方式
begin
    f <= ~a;
end
endmodule
```

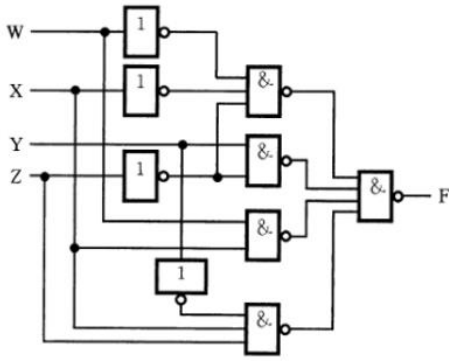
```
module example_4_12a(
    input sw_pin[7:0],
    output [15:0] led_pin
);
    wire p1, p2, p3, p4, p5, p6, p7, p8;
```

```
not_gate U1(.a(sw_pin[0]),.f(p1));           //结构化描述方式
not_gate U2(.a(sw_pin[1]),.f(p2));           //结构化描述方式
not_gate U3(.a(sw_pin[7]),.f(p3));           //结构化描述方式
not_gate U4(.a(sw_pin[6]),.f(p4));           //结构化描述方式

nand3_gate U5(.a(p1),.b(p2),.c(p3),.f(p5));   //结构化描述方式
nand2_gate U6(.a(sw_pin[6]),.b(p3),.f(p6));   //结构化描述方式
nand2_gate U7(.a(sw_pin[0]),.b(sw_pin[1]),.f(p7)); //结构化描述方式
nand3_gate U8(.a(p4),.b(sw_pin[1]),.c(sw_pin[7]),.f(p8)); //结构化描述方式

nand4_gate U9(.a(p5),.b(p6),.c(p7),.d(p8),.f(led_pin[0])); //结构化描述方式
```

endmodule



结构化描述方式

```
if((sw_pin[0]==0 & sw_pin[1]==0) & (sw_pin[6]==0 & sw_pin[7]==0) | (sw_pin[0]==0 & sw_pin[1]==0) & (sw_pin[6]==1 &
sw_pin[7]==0) | (sw_pin[0]==0 & sw_pin[1]==1) & (sw_pin[6]==0 & sw_pin[7]==1) | (sw_pin[0]==0 & sw_pin[1]==1) & (sw_pin[6]==1 &
sw_pin[7]==0) | (sw_pin[0]==1 & sw_pin[1]==0) & (sw_pin[6]==1 & sw_pin[7]==0) | (sw_pin[0]==1 & sw_pin[1]==1) &
(sw_pin[6]==0 & sw_pin[7]==0) | (sw_pin[0]==1 & sw_pin[1]==1) & (sw_pin[6]==0 & sw_pin[7]==1) | (sw_pin[0]==1 & sw_pin[1]==1)
& (sw_pin[6]==1 & sw_pin[7]==0) | (sw_pin[0]==1 & sw_pin[1]==1) & (sw_pin[6]==1 & sw_pin[7]==1))
```

example\_4\_12a\_1\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//采用行为描述方式实现例题4.12血型是否相容的电路（采用血型编码I），输入为开发板上最左边的2个拨动开关和最右边的2个开关，输出为开发板上最左边的LED灯。

`timescale 1ns / 1ps

```
module example_4_12a_1(
    input sw_pin[7:0],           //8个拨动开关
    output reg [15:0] led_pin    //16个led灯
);
```

```
always @(*)           //行为描述方式
begin
    if((sw_pin[0]==0 & sw_pin[1]==0) & (sw_pin[6]==0 & sw_pin[7]==0) | (sw_pin[0]==0 & sw_pin[1]==0) & (sw_pin[6]==1 & sw_pin[7]==0) | (sw_pin[0]==0
led_pin[0] <= 1;
    else
        led_pin[0] <= 0;
    end
end
```

endmodule

行为描述方式

表 4.9 血型编码(1)

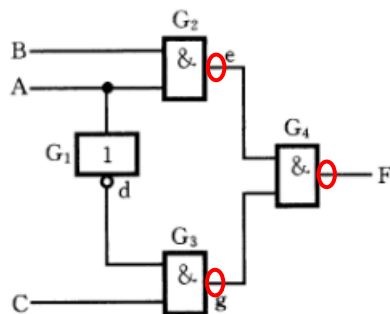
血 型	献		受	
	W	X	Y	Z
A	0	0	0	0
B	0	1	0	1
AB	1	0	1	0
O	1	1	1	1

表 4.8 血型相容规则表

受血 献血	A	B	AB	O
A	✓		✓	
B		✓	✓	
AB			✓	
O	✓	✓	✓	✓

## • 6、（验证实验）例题4.16的实现

- 分别用结构化描述方式和数据流描述方式实现例题4.16增加冗余项前的组合逻辑电路 ( $F=A \cdot B + \neg A \cdot C$ )，以及增加冗余项后的组合逻辑电路 ( $F=A \cdot B + \neg A \cdot C + B \cdot C$ )，该电路有3个输入、1个输出。
- 输入为开发板上最左边的3个拨动开关。
- 输出为开发板上最左边的LED灯（增加冗余项前）和最右边的LED灯（增加冗余项后）。



增加冗余项前的电路

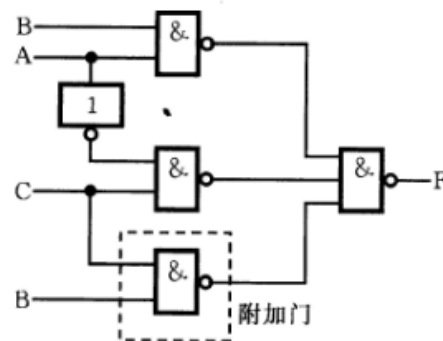
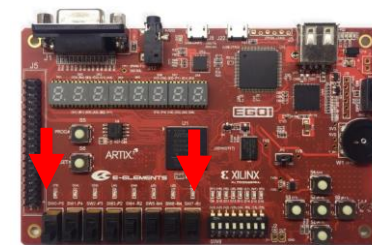


图 4.23 增加冗余项后的逻辑电路



EGO1开发板

- 在EGO1开发板上实现例题4.16的设计文件：

example\_4\_16\_EGO1.v - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
);
always @(*)           //行为描述方式
begin
    f <= ~a;
end
endmodule
```

```
module nand2_gate(
    input a,
    input b,
    output reg f
);
always @(*)           //行为描述方式
begin
    f <= ~(a & b);
end
endmodule
```

```
module nand3_gate(
    input a,
    input b,
    input c,
    output reg f
);
always @(*)           //行为描述方式
begin
    f <= ~(a & b & c);
end
endmodule
```

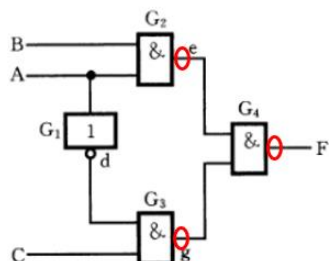
```
module example_4_16(
    input sw_pin[7:0],
    output [15:0] led_pin
);
wire p1, p2, p3, p4, p5, p6, p7;
```

## 结构化描述方式

```
nand2_gate U1(a(sw_pin[1]),b(sw_pin[0]),f(p1));           //结构化描述方式
not_gate U2(a(sw_pin[0]),f(p2));                         //结构化描述方式
nand2_gate U3(a(p2),b(sw_pin[2]),f(p3));                 //结构化描述方式
nand2_gate U4(a(p1),b(p3),f(led_pin[0]));                //结构化描述方式
```

```
nand2_gate U5(a(sw_pin[1]),b(sw_pin[0]),f(p4));           //结构化描述方式
not_gate U6(a(sw_pin[0]),f(p5));                         //结构化描述方式
nand2_gate U7(a(p5),b(sw_pin[2]),f(p6));                 //结构化描述方式
nand2_gate U8(a(sw_pin[2]),b(sw_pin[1]),f(p7));           //结构化描述方式
nand3_gate U9(a(p4),b(p6),c(p7),f(led_pin[7]));          //结构化描述方式
```

```
endmodule
```



增加冗余项前的电路

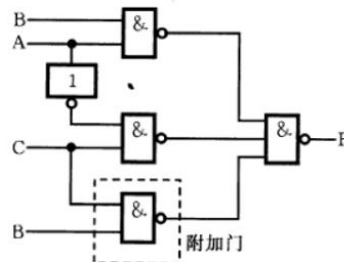


图 4.23 增加冗余项后的逻辑电路

example\_4\_16\_1\_EGO1.v - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

//采用数据流描述方式实现例题4.16的组合逻辑电路（增加冗余项前），以及增加冗余项后的组合逻辑电路（

```
`timescale 1ns / 1ps
```

```
module example_4_9_1(
    input sw_pin[7:0],
    output [15:0] led_pin
);
```

## 数据流描述方式

```
assign led_pin[0] = (sw_pin[0] & sw_pin[1]) | (~sw_pin[0] & sw_pin[2]);
```

```
assign led_pin[7] = (sw_pin[0] & sw_pin[1]) | (~sw_pin[0] & sw_pin[2]) | (sw_pin[1] & sw_pin[2]);
```

```
endmodule
```

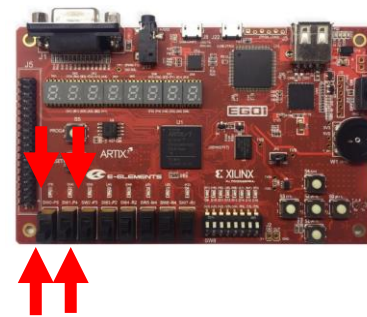
$$F = A \cdot B + \neg A \cdot C$$

$$F = A \cdot B + \neg A \cdot C + B \cdot C$$



## • 7、（设计实验，课后完成）例题4.2的实现

- 请分别用结构化描述方式、行为描述方式和数据流描述方式实现例题4.2的半加器HA电路，该电路有2个输入、2个输出。
- 输入为开发板上最左边的2个拨动开关，输出为开发板上最左边的2个LED灯。
- EGO1开发板（结构化描述方式）：工程命名为example\_4\_2\_EGO1，设计文件命名为example\_4\_2\_EGO1.v，约束文件为EGO1.xdc。
- EGO1开发板（行为描述方式）：工程命名为example\_4\_2\_1\_EGO1，设计文件命名为example\_4\_2\_1\_EGO1.v，约束文件为EGO1.xdc。
- EGO1开发板（数据流描述方式）：工程命名为example\_4\_2\_2\_EGO1，设计文件命名为example\_4\_2\_2\_EGO1.v，约束文件为EGO1.xdc。



EGO1开发板

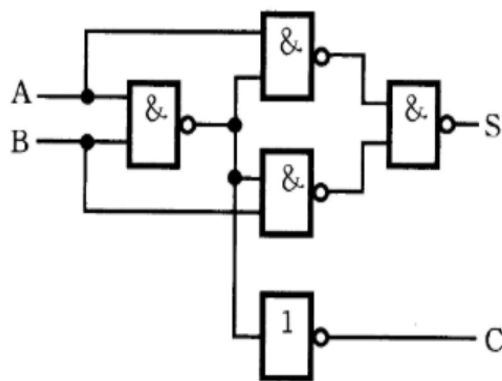


图4.3(a) 逻辑电路

表 4.2 真值表			
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

## • 8、（设计实验，课后完成）例题4.12的实现（b）

- 请分别用**结构化描述方式**和**行为描述方式**实现例题4.12的判断献血者与受血者**血型是否相容的电路**（采用血型编码II），该电路有4个输入、1个输出。
- 输入**为开发板上最左边的2个拨动开关和最右边的2个开关，**输出**为开发板上最左边的LED灯。
- EGO1开发板（结构化描述方式）：工程命名为**example\_4\_12b\_EGO1**，设计文件命名为example\_4\_12b\_EGO1.v，约束文件为EGO1.xdc。
- EGO1开发板（行为描述方式）：工程命名为**example\_4\_12b\_1\_EGO1**，设计文件命名为example\_4\_12b\_1\_EGO1.v，约束文件为EGO1.xdc。



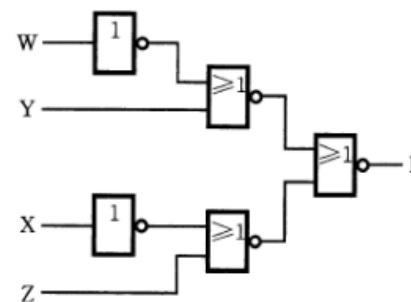
EGO1开发板

表 4.8 血型相容规则表

受血 献血	A	B	AB	O
A	✓		✓	
B		✓	✓	
AB			✓	
O	✓	✓	✓	✓

表 4.10 血型编码(2)

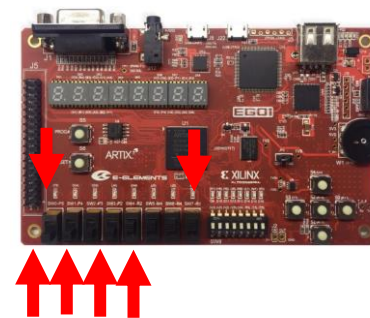
血 型	献		受	
	W	X	Y	Z
O	0	0	0	0
A	0	1	0	1
B	1	0	1	0
AB	1	1	1	1





## • 9、（设计实验，课后完成）例题4.17的实现

- 请分别用**结构化描述方式**和**数据流描述方式**实现例题4.17增加冗余项前的组合逻辑电路（ $F = \neg A \cdot C + B \cdot \neg C \cdot D + A \cdot \neg B \cdot \neg C$ ），以及增加冗余项后的组合逻辑电路（ $F = \neg A \cdot C + B \cdot \neg C \cdot D + A \cdot \neg B \cdot \neg C + \neg A \cdot B \cdot D + A \cdot \neg C \cdot D$ ），该电路有4个输入、1个输出。
- **输入**为开发板上最左边的4个拨动开关。
- **输出**为开发板上最左边的LED灯（**增加冗余项前**）和最右边的LED灯（**增加冗余项后**）。
- EGO1开发板（结构化描述方式）：工程命名为**example\_4\_17\_EGO1**，设计文件命名为example\_4\_17\_EGO1.v，约束文件为EGO1.xdc。
- EGO1开发板（数据流描述方式）：工程命名为**example\_4\_17\_1\_EGO1**，设计文件命名为example\_4\_17\_1\_EGO1.v，约束文件为EGO1.xdc。



EGO1开发板

## • 10、（选做实验）例题4.4的实现

- 请分别用**结构化描述方式**和**行为描述方式**实现例题4.4的**3变量多数表决电路**，该电路有3个输入、1个输出。
- **输入**为开发板上最左边的3个拨动开关，**输出**为开发板上最左边的1个LED灯。

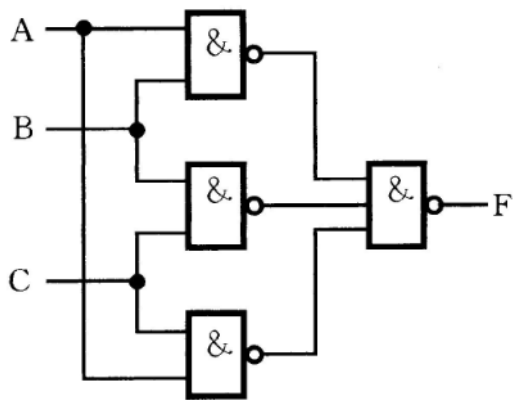
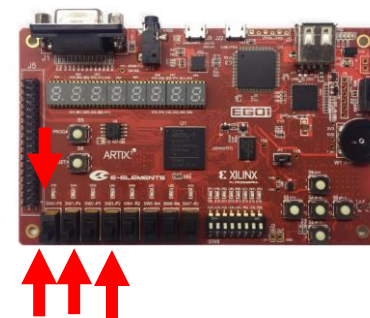


表 4.4 真值表

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



EGO1开发板

## • 11、（选做实验） 例题4.6的实现

- 请分别用**结构化描述方式（有2个不同的电路）**和**行为描述方式**实现例题4.6的**乘法器电路**（用于产生两个2位二进制数相乘的积），该电路有4个输入、4个输出，该实验有**3个小实验**（方法I电路的结构化描述方式，方法II电路的结构化描述方式，行为描述方式）。
- 输入**为开发板上最左边的2个拨动开关和最右边的2个拨动开关，**输出**为开发板上最左边的4个LED灯。

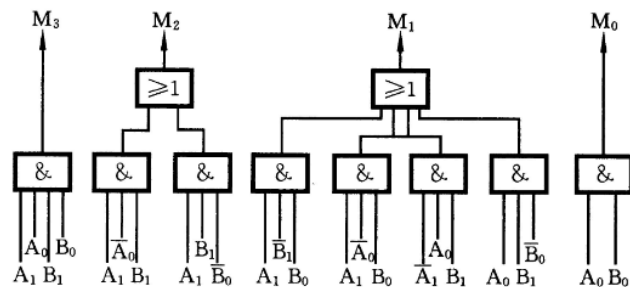


图 4.7 方法 I 的逻辑电路

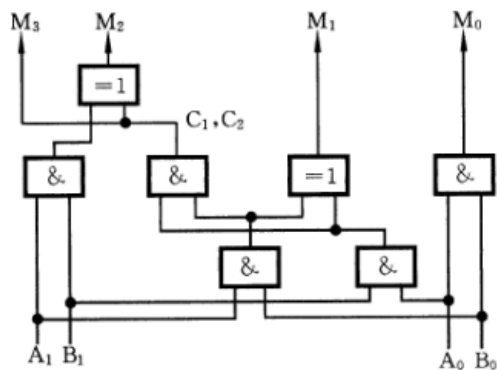
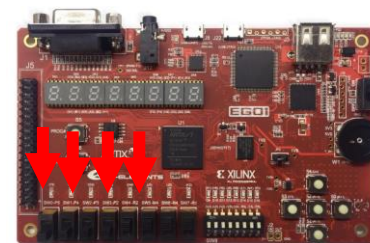


图 4.8 方法 II 的逻辑电路

表 4.5 真值表

A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	M <sub>3</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>	A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	M <sub>3</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1	0	1	0	0	1	0	0
0	0	1	1	0	0	0	0	1	0	1	1	0	1	1	0
0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1	1	1	0	1	0	0	1	1
0	1	1	0	0	0	1	0	1	1	1	0	0	1	1	0
0	1	1	1	0	0	1	1	1	1	1	1	1	0	0	1



EGO1开发板

## • 12、（选做实验） 例题4.7的实现

- 请分别用结构化描述方式和行为描述方式实现例题4.7的判别以余3码表示的1位十进制数是否为合数的电路，该电路有4个输入、1个输出。
- 输入为开发板上最左边的4个拨动开关，输出为开发板上最左边的1个LED灯。

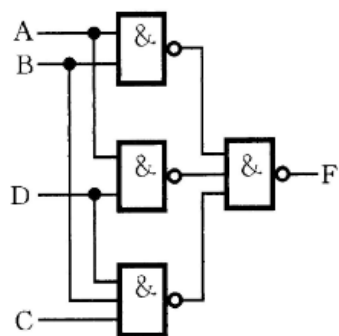
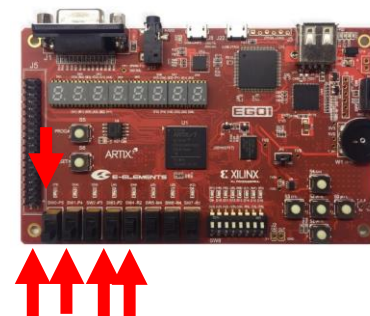


图 4.10 逻辑电路

表 4.6 真值表

A	B	C	D	F	A	B	C	D	F
0	0	0	0	d	1	0	0	0	0
0	0	0	1	d	1	0	0	1	1
0	0	1	0	d	1	0	1	0	0
0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	1	1	0	0	1
0	1	0	1	0	1	1	0	1	d
0	1	1	0	0	1	1	1	0	d
0	1	1	1	1	1	1	1	1	d



EGO1开发板

## • 13、（选做实验）例题4.8的实现

- 请分别用**结构化描述方式（有2个不同的电路）**和**行为描述方式**实现例题4.8的**组合逻辑电路**，该电路有3个输入、2个输出，该实验有**3个小实验**（电路I的结构化描述方式，电路II的结构化描述方式，行为描述方式）。
- 输入**为开发板上最左边的3个拨动开关，**输出**为开发板上最左边的2个LED灯。

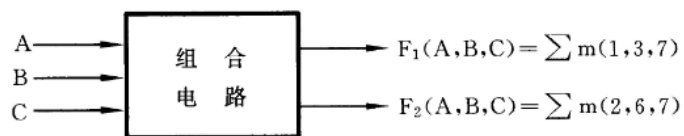
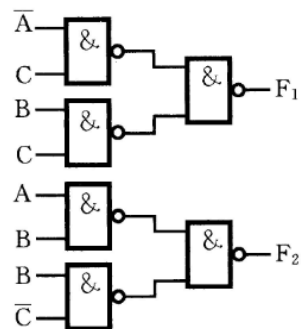
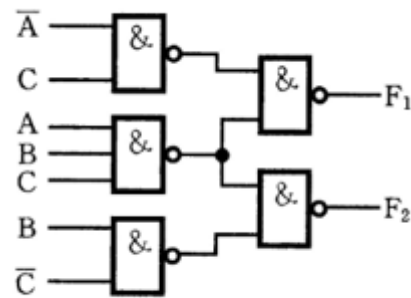


图 4.11 组合逻辑电路结构框图



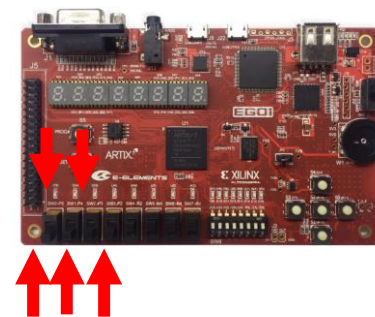
电路I



电路II

真值表

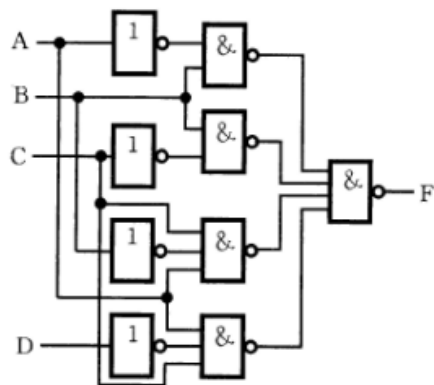
A	B	C	F1	F2
0	0	0	0	0
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	1
1	1	1	1	1



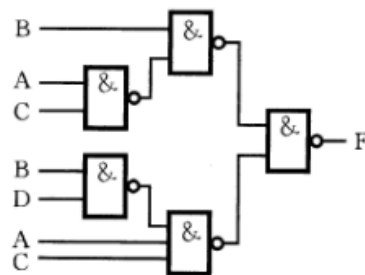
EGO1开发板

## • 14、（选做实验） 例题4.10的实现

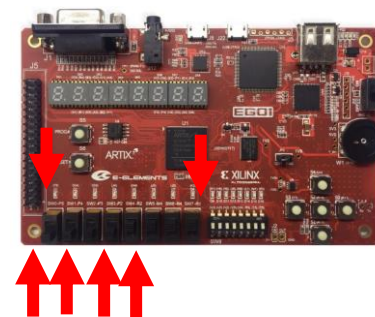
- 请用**结构化描述方式（有2个不同的电路）**实现例题4.10的**组合逻辑电路**，该电路有4个输入、1个输出。
- **输入**为开发板上最左边的4个拨动开关，**输出**为开发板上最左边的LED灯（**简化前**的电路）和最右边的LED灯（**简化后**的电路）。



简化前的电路



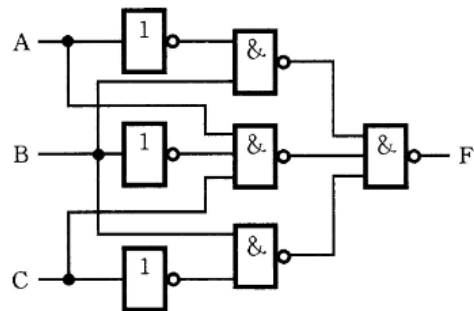
简化后的电路



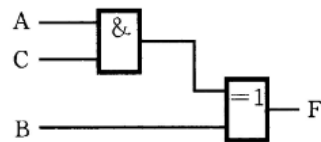
EGO1开发板

## • 15、（选做实验）例题4.11的实现

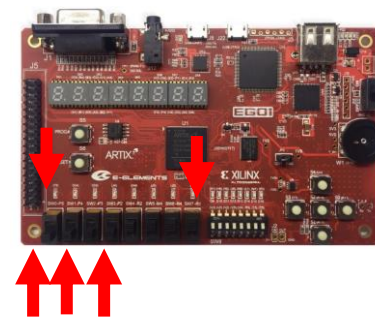
- 请用结构化描述方式（有2个不同的电路）实现例题4.11的组合逻辑电路，该电路有3个输入、1个输出。
- 输入为开发板上最左边的3个拨动开关，输出为开发板上最左边的LED灯（简化前的电路）和最右边的LED灯（简化后的电路）。



简化前的电路



简化后的电路

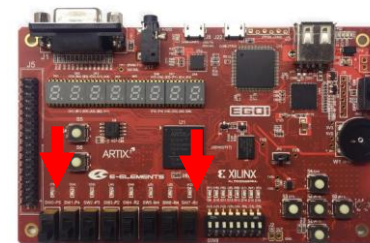


EGO1开发板

## (二) 在FPGA开发板上实现教材第4章的习题

### • 1、（选做实验）习题4.1的实现

- 请采用**结构化描述方式**在FPGA开发板上实现习题4.1的电路（图4.27），以及**简化后**的电路。
- **输入**为开发板上最左边的3个拨动开关（ABC），**输出**为开发板上最左边的LED灯（**简化前**）和最右边的LED灯（**简化后**）。
- EGO1开发板：工程命名为**exercise\_4\_1\_EGO1**，设计文件命名为exercise\_4\_1\_EGO1.v，约束文件为EGO1.xdc。



EGO1开发板

4.1 分析图 4.27 所示的组合逻辑电路,说明电路功能,并画出其简化逻辑电路图。

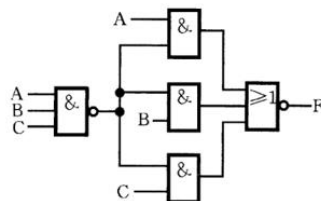
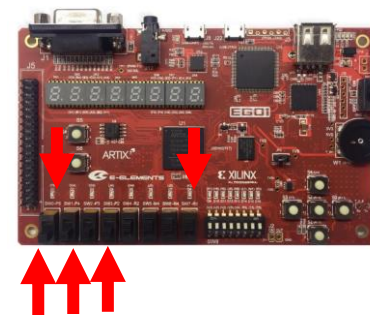


图 4.27 组合逻辑电路



## • 2、（选做实验）习题4.2的实现

- 请采用**结构化描述方式**在FPGA开发板上实现习题4.2的电路（图4.28），以及**改用异或门**实现的电路。
- **输入**为开发板上最左边的3个拨动开关（ABC），**输出**为开发板上最左边的LED灯（图4.28的电路）和最右边的LED灯（**改用异或门实现的电路**）。



EGO1开发板

4.2 分析图 4.28 所示的组合逻辑电路：(1) 指出在哪些输入取值下，输出 F 的值为 1；  
(2) 改用异或门实现该电路的逻辑功能。

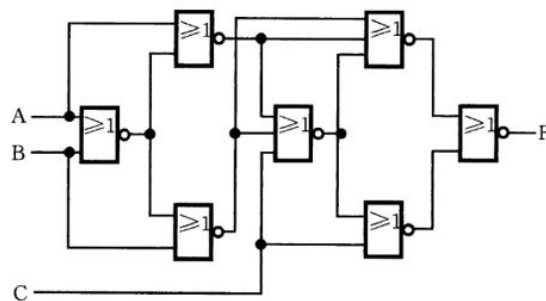


图 4.28 组合逻辑电路

### • 3、（选做实验）习题4.3的实现

- 请分别采用**结构化描述方式**（根据图4.29）和**行为描述方式**（根据真值表）在FPGA开发板上实现习题4.3的电路。
- **输入**为开发板上最左边的4个拨动开关（ABCD），**输出**为开发板上最左边的4个LED灯（WXYZ）。

4.3 分析图 4.29 所示组合逻辑电路,列出真值表,说明该电路的逻辑功能。

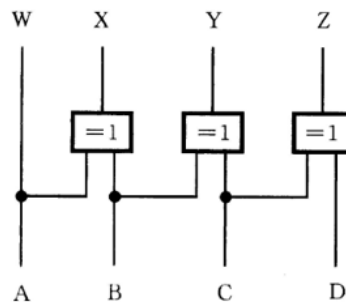
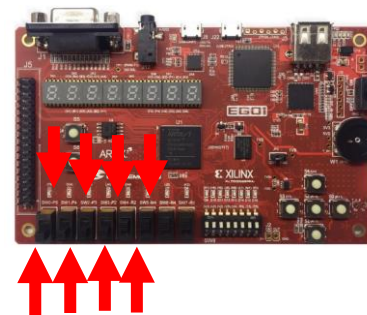


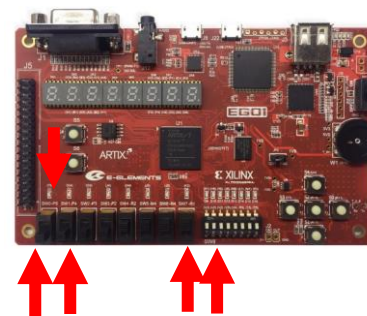
图 4.29 组合逻辑电路



EGO1开发板

## • 4、（选做实验）习题4.4的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.4要求的组合逻辑电路。
- **输入**为开发板上最左边的2个拨动开关（ $A_2A_1$ ）和最右边的2个拨动开关（ $B_2B_1$ ），**输出**为开发板上最左边的LED灯（Z）。

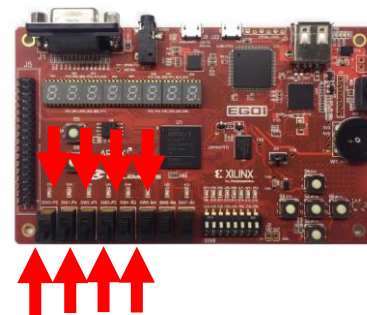


EGO1开发板

4.4 设计一个组合逻辑电路,该电路输入端接收两个 2 位二进制数  $A=A_2A_1$ ,  $B=B_2B_1$ 。当  $A>B$  时,输出  $Z=1$ ,否则  $Z=0$ 。

## • 5、（选做实验）习题4.5的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.5要求的代码转换电路。
- **输入**为开发板上最左边的4个拨动开关（余3码），**输出**为开发板上最左边的4个LED灯（2421码）。

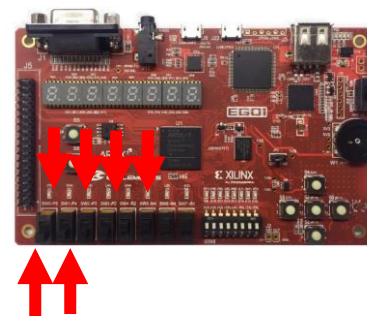


EGO1开发板

4.5 设计一个代码转换电路,将 1 位十进制数的余 3 码转换成 2421 码。

## • 6、（选做实验）习题4.6（1）的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.6（1）要求的组合逻辑电路。
- **输入**为开发板上最左边的2个拨动开关（ $X=AB$ ），**输出**为开发板上最左边的4个LED灯（ $Y=Y_3Y_2Y_1Y_0$ ）。



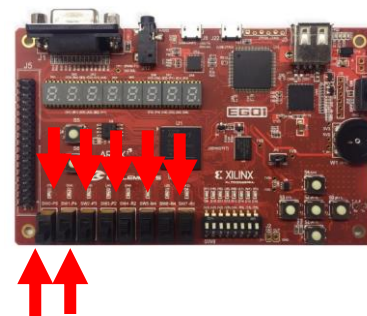
EGO1开发板

4.6 假定  $X=AB$  代表一个 2 位二进制数，试设计满足如下要求的逻辑电路（ $Y$  也用二进制数表示）：

$$(1) Y = X^2$$

## • 7、（选做实验）习题4.6（2）的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.6（2）要求的组合逻辑电路。
- **输入**为开发板上最左边的2个拨动开关（ $X=AB$ ），**输出**为开发板上最左边的5个LED灯（ $Y=Y_4Y_3Y_2Y_1Y_0$ ）。



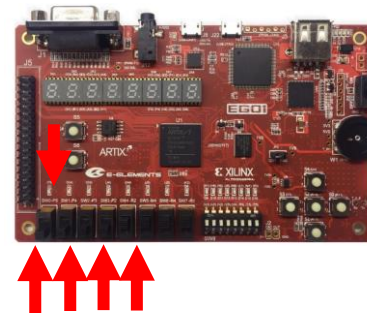
EGO1开发板

4.6 假定  $X=AB$  代表一个 2 位二进制数，试设计满足如下要求的逻辑电路（ $Y$  也用二进制数表示）：

$$(2) Y = X^3$$

## • 8、（选做实验）习题4.7的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.7要求的组合逻辑电路。
- **输入**为开发板上最左边的4个拨动开关（2421码），**输出**为开发板上最左边的LED灯（F）。

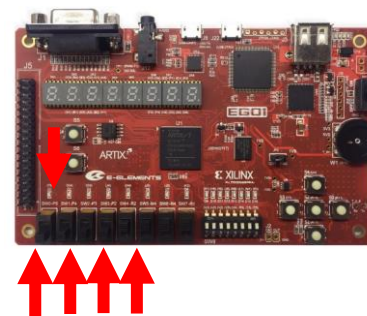


EGO1开发板

4.7 用与非门设计一个组合逻辑电路,该电路输入为 1 位十进制数的 2421 码,当输入的数字为素数时,输出 F 为 1,否则 F 为 0。

## • 9、（选做实验）习题4.8的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.8要求的组合逻辑电路。
- **输入**为开发板上最左边的4个拨动开关（8421码），**输出**为开发板上最左边的LED灯（F）。



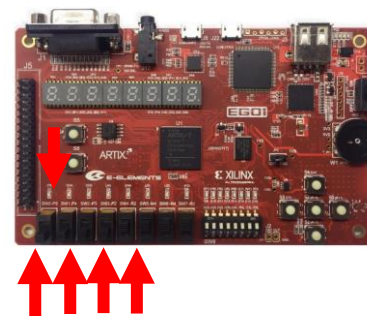
EGO1开发板

4.8 设计一个“四舍五入”电路。该电路输入为 1 位十进制数的 8421 码,当其值大于或等于 5 时,输出 F 的值为 1,否则 F 的值为 0。



## • 10、（选做实验）习题4.9的实现

- 请分别采用**行为描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.9要求的组合逻辑电路。
- **输入**为开发板上最左边的4个拨动开关（4位二进制数），**输出**为开发板上最左边的LED灯。

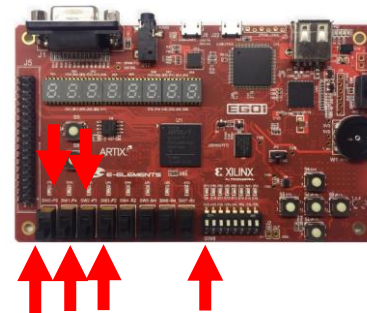


EGO1开发板

4.9 设计一个检测电路,检测 4 位二进制码中 1 的个数是否为偶数。若为偶数个 1,则输出为 1,否则输出为 0。

## • 11、（选做实验）习题4.10的实现

- 请分别采用**数据流描述方式**（根据题目要求）和**结构化描述方式**（根据设计好的电路）在FPGA开发板上实现习题4.10要求的组合逻辑电路。
- **输入**为开发板上最左边的3个拨动开关（ $A_i$ 、 $B_i$ 、 $C_{i-1}$ ）和最右边的1个开关（ $M$ ），**输出**为开发板上最左边的2个LED灯（ $S_i$ 、 $C_i$ ）。

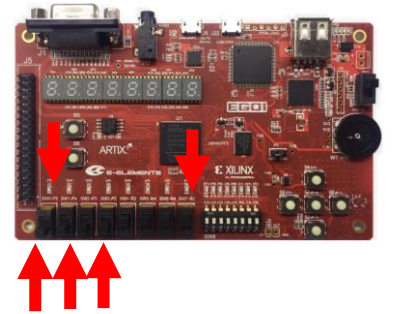


EGO1开发板

4.10 设计一个加/减法器,该电路在  $M$  控制下进行加、减运算。当  $M=0$  时,实现全加器功能;当  $M=1$  时,实现全减器功能。

## • 12、（选做实验）习题4.11（1）的实现

- 请采用**结构化描述方式**在FPGA开发板上实现习题4.11（1）的组合逻辑电路（ $F=A \cdot B + \overline{A} \cdot C + B \cdot \overline{C}$ ），以及用与非门实现的电路。
- **输入**为开发板上最左边的3个拨动开关（ABC），**输出**为开发板上最左边的LED灯（ $F=A \cdot B + \overline{A} \cdot C + B \cdot \overline{C}$ ）和最右边的LED灯（**用与非门实现**）。

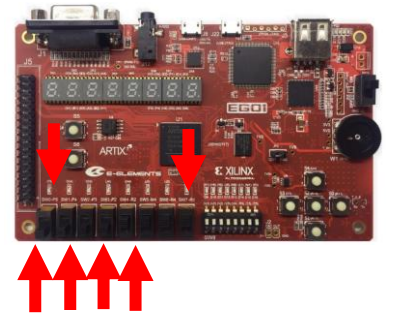


EGO1开发板

4.11 在输入不提供反变量的情况下,用与非门组成实现下列函数的最简电路。  
(1)  $F = A\overline{B} + \overline{A}C + B\overline{C}$

### • 13、（选做实验）习题4.11（2）的实现

- 请采用**结构化描述方式**在FPGA开发板上实现习题4.11（2）的组合逻辑电路（ $F=A\cdot/B\cdot/C + B\cdot C\cdot/D + A\cdot/C\cdot/D + /B\cdot C\cdot D$ ），以及用与非门实现的电路。
- **输入**为开发板上最左边的4个拨动开关（ABCD），**输出**为开发板上最左边的LED灯（ $F=A\cdot/B\cdot/C + B\cdot C\cdot/D + A\cdot/C\cdot/D + /B\cdot C\cdot D$ ）和最右边的LED灯（**用与非门实现**）。



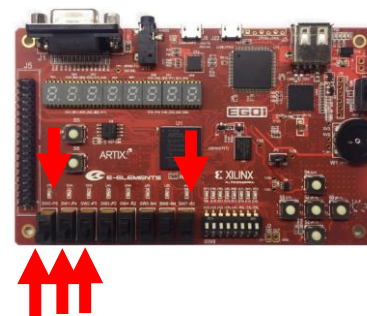
EGO1开发板

4.11 在输入不提供反变量的情况下,用与非门组成实现下列函数的最简电路。

$$(2) F = A\bar{B}\bar{C} + BC\bar{D} + A\bar{C}\bar{D} + \bar{B}CD$$

## • 14、（选做实验）习题4.12（3）的实现

- 请采用**结构化描述方式**在FPGA开发板上实现习题4.12（3）的组合逻辑电路（ $F=(A+B) \cdot (\overline{A}+C)$ ），以及增加冗余项后的电路。
- **输入**为开发板上最左边的3个拨动开关（ABC），**输出**为开发板上最左边的LED灯（增加冗余项前， $F=(A+B) \cdot (\overline{A}+C)$ ）和最右边的LED灯（增加冗余项后）。



EGO1开发板

4.12 下列函数描述的电路是否可能产生险象？在什么情况下产生险象？若产生险象，试用增加冗余项的方法消除。

$$(3) F_3 = (A + \overline{B})(\overline{A} + \overline{C})$$

# 实验要求

- 1、在Logisim上完成**验证实验**，将实验过程中的主要结果通过截屏（或拍照）的方式，拷贝到实验报告中，并给予适当的文字说明。
- 2、在FPGA开发板上完成**验证实验**，将实验过程中的主要结果通过截屏（或拍照）的方式，拷贝到实验报告中，并给予适当的文字说明。
- 3、在Logisim上完成**设计实验**，设计文件命名为：**第四章习题的实现.circ**。
- 4、在FPGA开发板上完成**设计实验**，工程文件、设计文件、约束文件请严格按照规定的要求命名。
- 5、实验报告命名为：**学号+姓名+第3次实验报告.docx**。
- 6、将设计文件、实验报告打包压缩成1个压缩文件，命名为：**学号+姓名+第3次实验.zip**（或**.rar**），并上传到FTP上，上传截止日期：**2024年10月20日晚上24点**。
- 7、鼓励有兴趣的同学完成选做实验。

**Thanks**