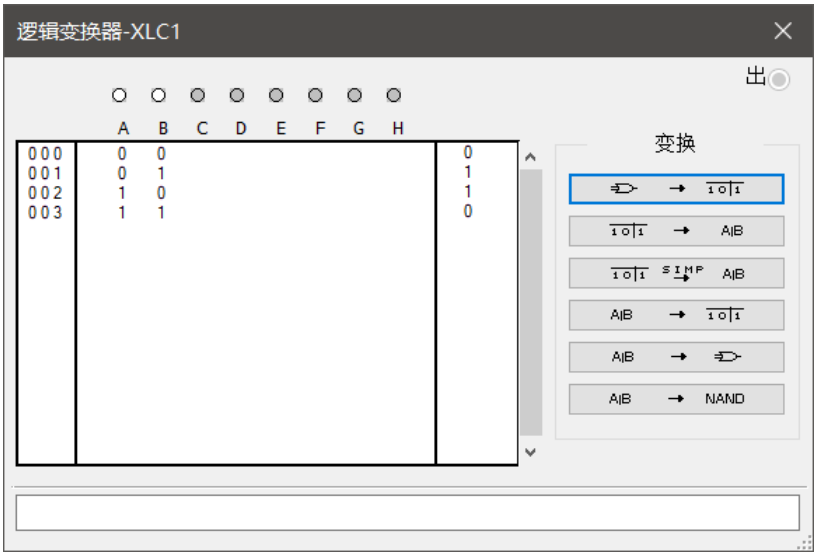
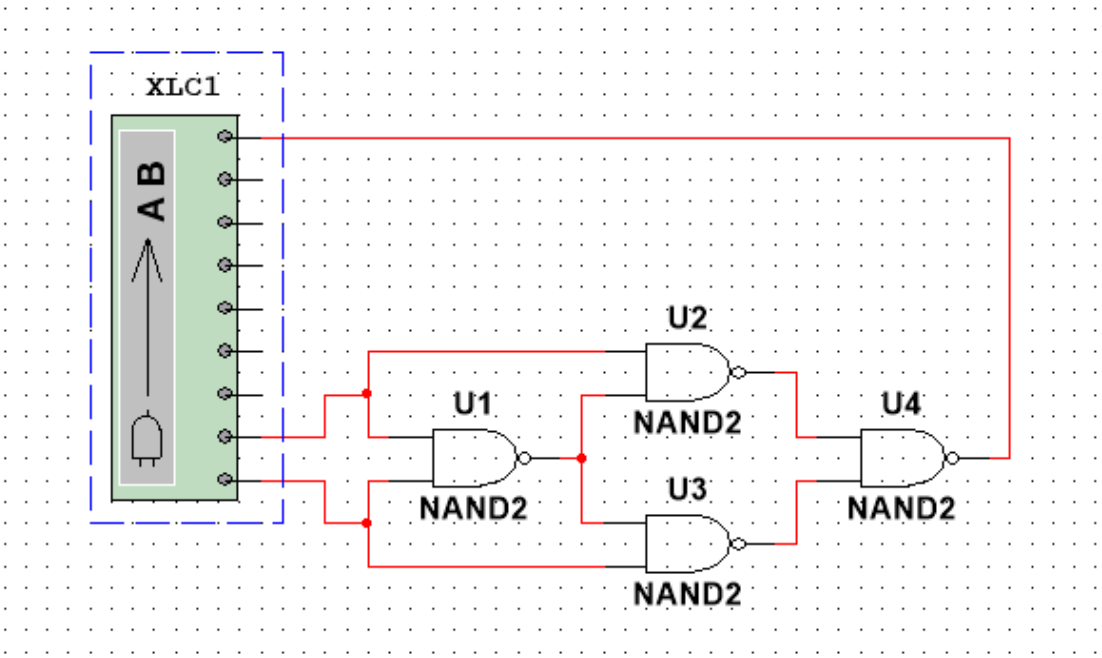
1. CMOS logic gate

An XOR gate can be built using NAND gate. The circuit diagram is shown below. Try building a XOR in multisim with NAND gate.

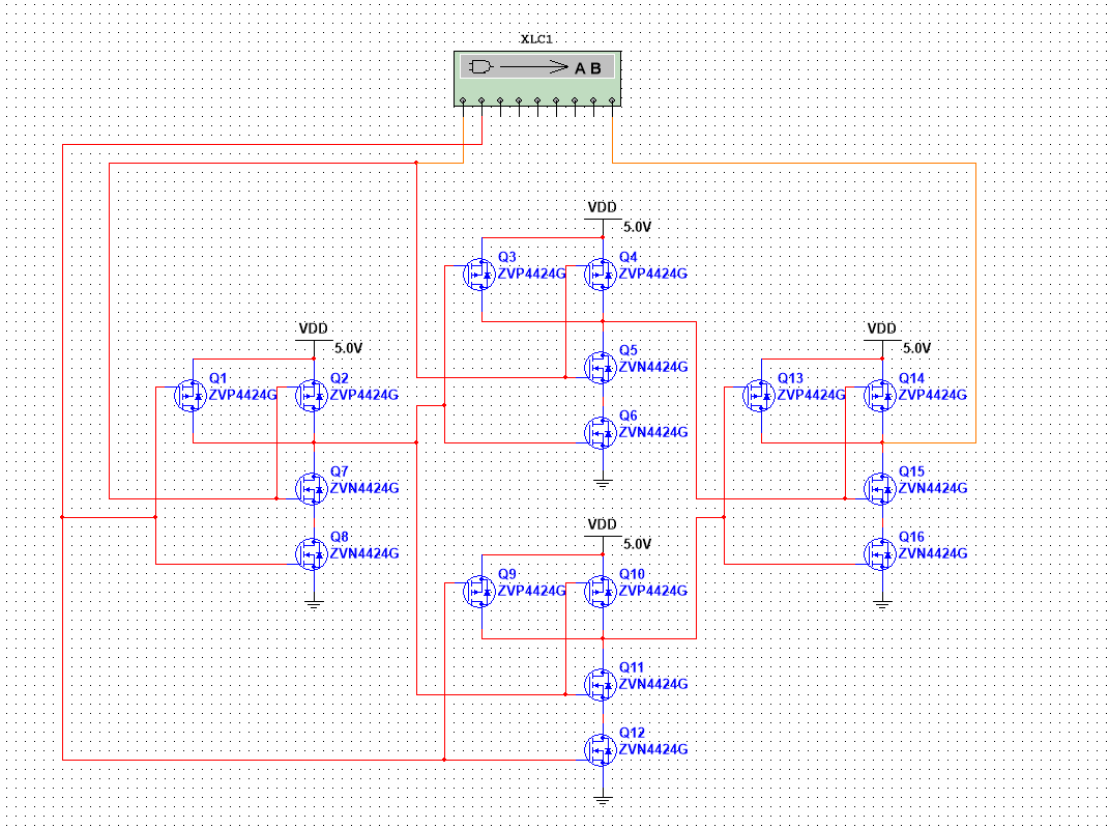
- Draw the truth table. Implement the circuit in multisim, use logic converter to generate its truth table, compare the simulation results with your answer

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

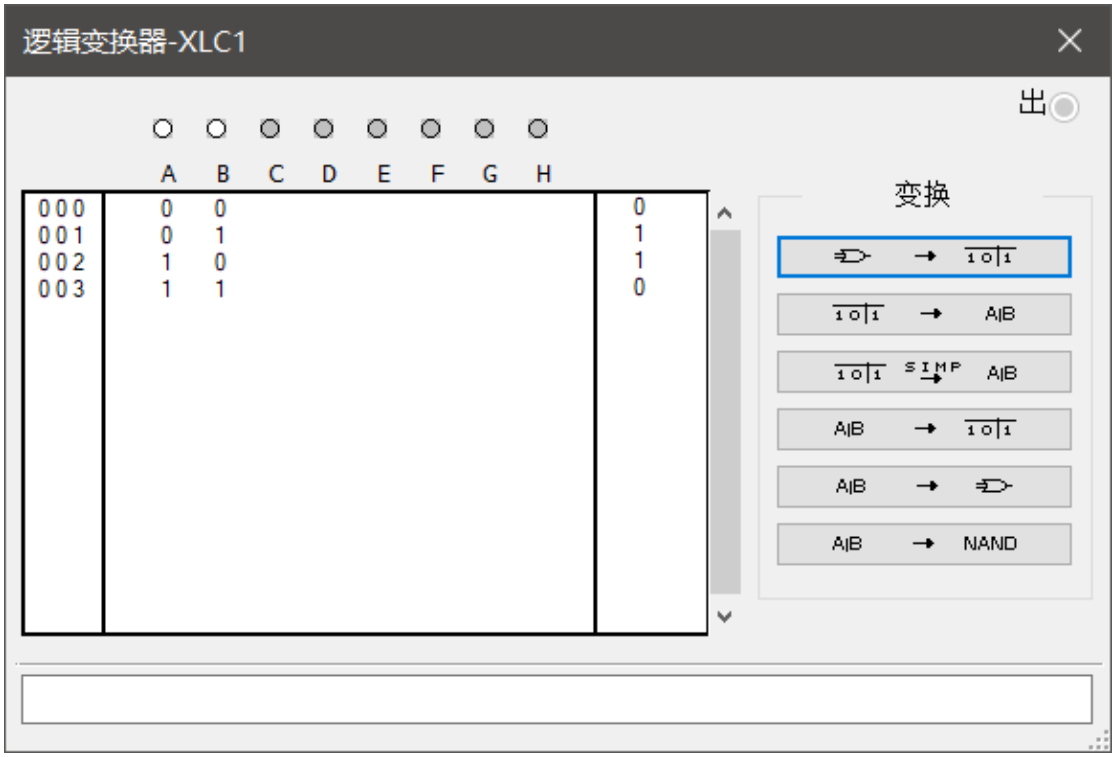




- Implement your design in Multisim, it should be CMOS-level implementation. If you don't know which MOSFET to use, you can use ZVN4424 and ZVP4424



- Use logic converter to generate the truth table of your design.



2. Combinational logic circuit exercises

- For each one of the truth tables, write down the Boolean equations in canonical sum of products form.

(a) $\bar{A}\bar{B}+A\bar{B}+AB$

(b) $\bar{A}\bar{B}\bar{C}+ABC$

(c) $\bar{A}\bar{B}\bar{C}+\bar{A}B\bar{C}+\bar{A}\bar{B}C+A\bar{B}C+ABC$

(d) $\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}B\bar{C}D+\bar{A}\bar{B}C\bar{D}+\bar{A}\bar{B}CD+A\bar{B}\bar{C}\bar{D}+A\bar{B}C\bar{D}+AB\bar{C}\bar{D}$

(e) $\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}\bar{B}CD+\bar{A}B\bar{C}D+\bar{A}BC\bar{D}+A\bar{B}\bar{C}D+A\bar{B}C\bar{D}+AB\bar{C}\bar{D}+ABCD$

- Simplify the logic using the K-map

(a) $\bar{A}\bar{B}+A\bar{B}+AB$

	B	1	0
A		B	$\bar{B}$
1	A	1	1
0	$\bar{A}$	0	1

$\bar{A}\bar{B}+A$

(b) $\bar{A}\bar{B}\bar{C}+ABC$

	BC	00	01	11	10
A		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
0	$\bar{A}$	1	0	0	0
1	A	0	0	1	0

$\bar{A}\bar{B}\bar{C}+ABC$

(c) $\bar{A}\bar{B}\bar{C}+\bar{A}B\bar{C}+\bar{A}\bar{B}C+A\bar{B}C+ABC$

	BC	00	01	11	10
A		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$

0	$\bar{A}$	1	0	0	1
1	A	1	1	1	0

$\bar{B}\bar{C}+AC+\bar{A}B\bar{C}$

(d) $\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}\bar{B}\bar{C}D+\bar{A}\bar{B}C\bar{D}+\bar{A}\bar{B}CD+A\bar{B}\bar{C}\bar{D}+A\bar{B}C\bar{D}+ABC\bar{D}$

	CD	00	01	11	10
AB		$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
00	$\bar{A}\bar{B}$	1	1	1	1
01	$\bar{A}B$	0	0	0	0
11	AB	0	0	0	1
10	$A\bar{B}$	1	0	0	1

$\bar{A}\bar{B}+A\bar{B}\bar{D}+ABC\bar{D}$

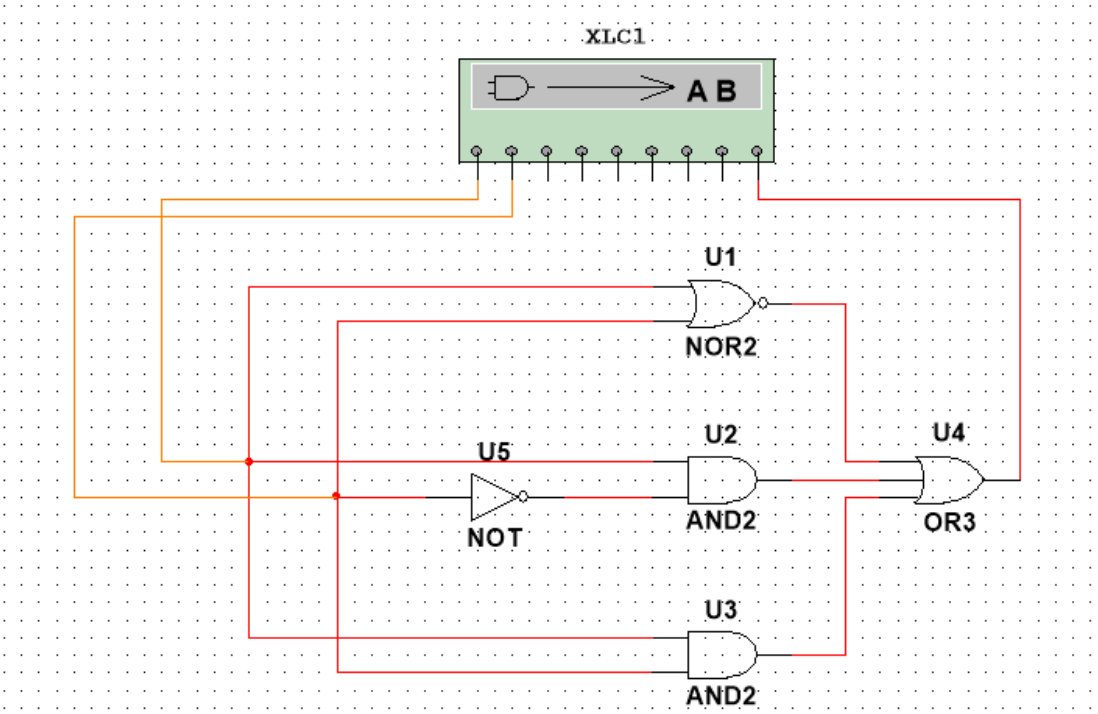
(e) $\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}\bar{B}CD+\bar{A}B\bar{C}D+\bar{A}BC\bar{D}+A\bar{B}CD+AB\bar{C}D+ABC\bar{D}+ABCD$

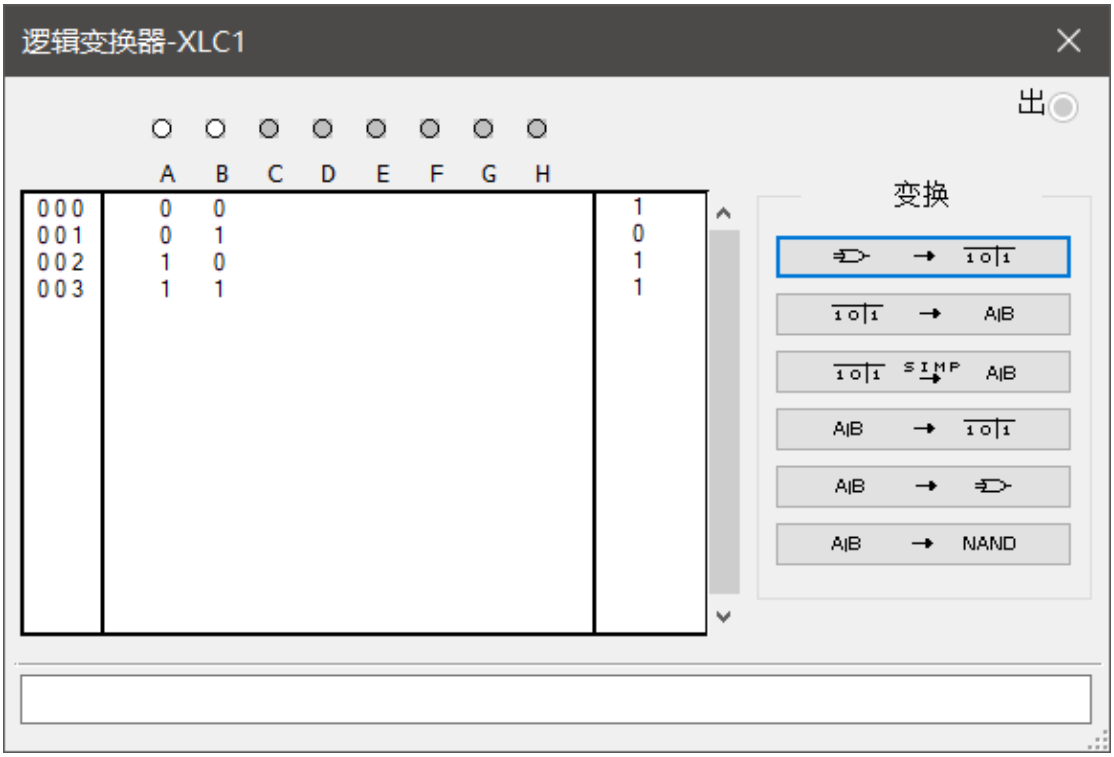
	CD	00	01	11	10
AB		$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
00	$\bar{A}\bar{B}$	1	0	1	0
01	$\bar{A}B$	0	1	0	1
11	AB	1	0	1	0
10	$A\bar{B}$	0	1	0	1

$\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}\bar{B}CD+\bar{A}B\bar{C}D+\bar{A}BC\bar{D}+A\bar{B}CD+AB\bar{C}D+ABC\bar{D}+ABCD$

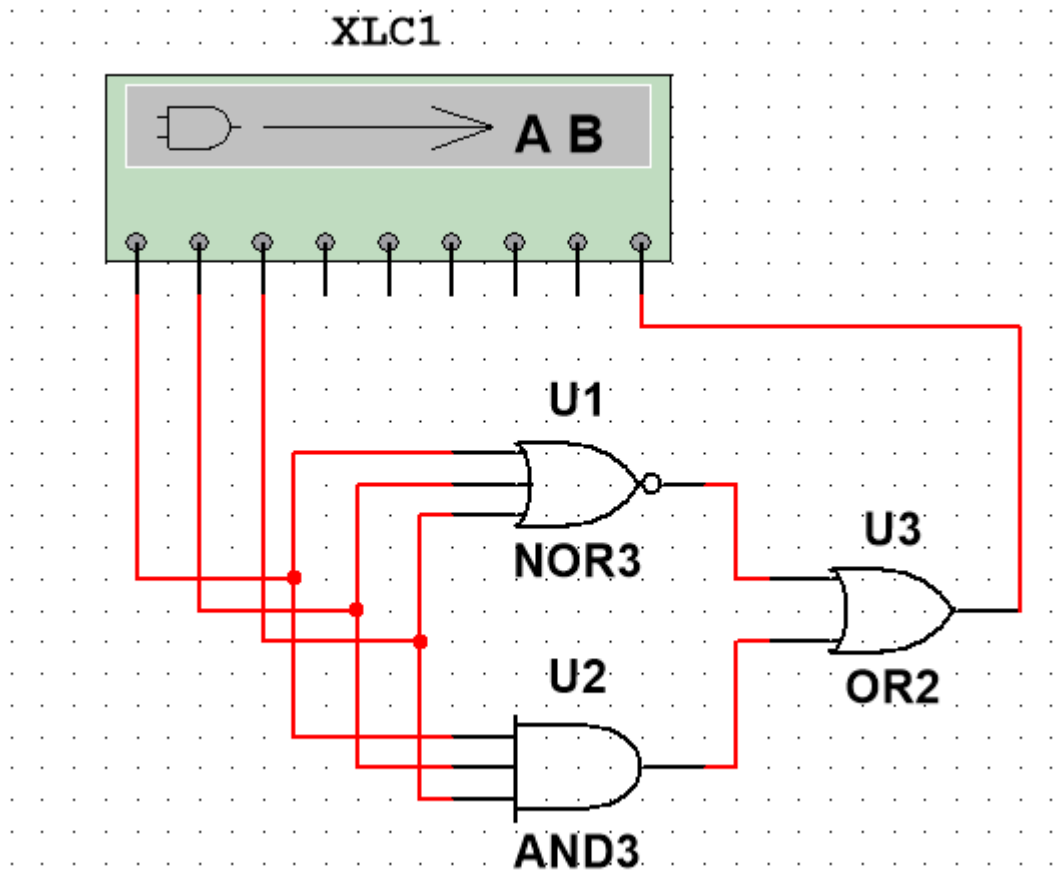
- Implement the circuits in Multisim using logic gates. Use logic converter to generate their truth tables.

(a)





(b)



逻辑变换器-XLC1

出

	A	B	C	D	E	F	G	H	
000	0	0	0						1
001	0	0	1						0
002	0	1	0						0
003	0	1	1						0
004	1	0	0						0
005	1	0	1						0
006	1	1	0						0
007	1	1	1						1

变换

1 0 1 →

1 0 1 → A|B

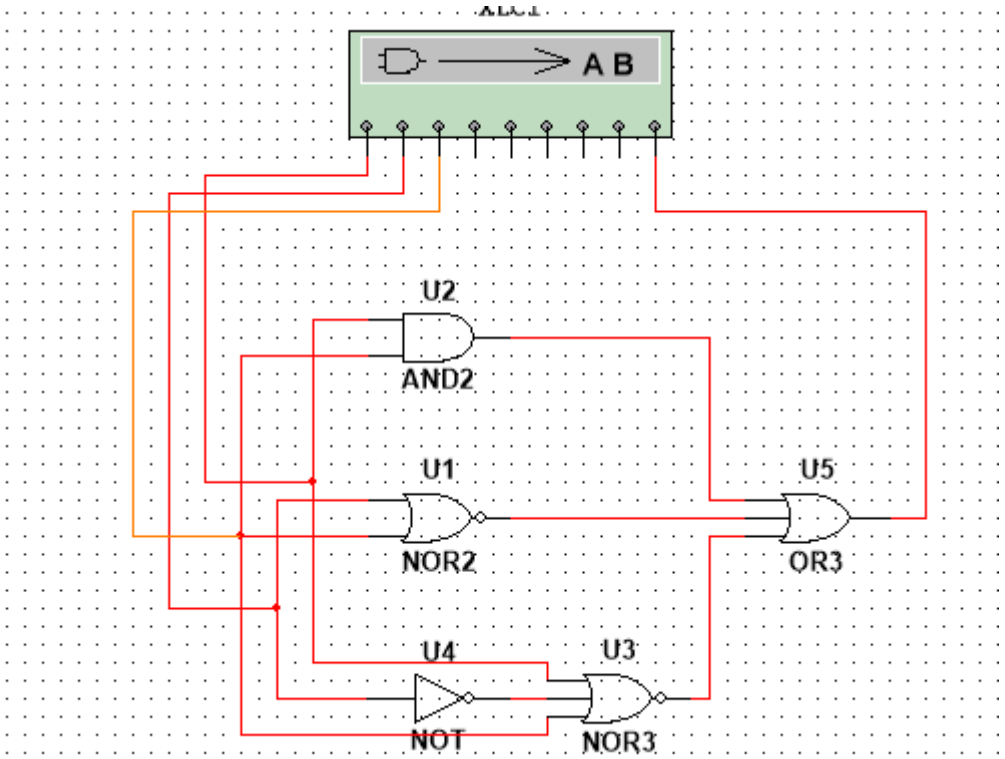
1 0 1 SIMP A|B

A|B → 1 0 1

A|B →

A|B → NAND

(c)



逻辑变换器-XLC1

出

	A	B	C	D	E	F	G	H	
000	0	0	0						1
001	0	0	1						0
002	0	1	0						1
003	0	1	1						0
004	1	0	0						1
005	1	0	1						1
006	1	1	0						0
007	1	1	1						1

变换

1 0 1 →

1 0 1 → A|B

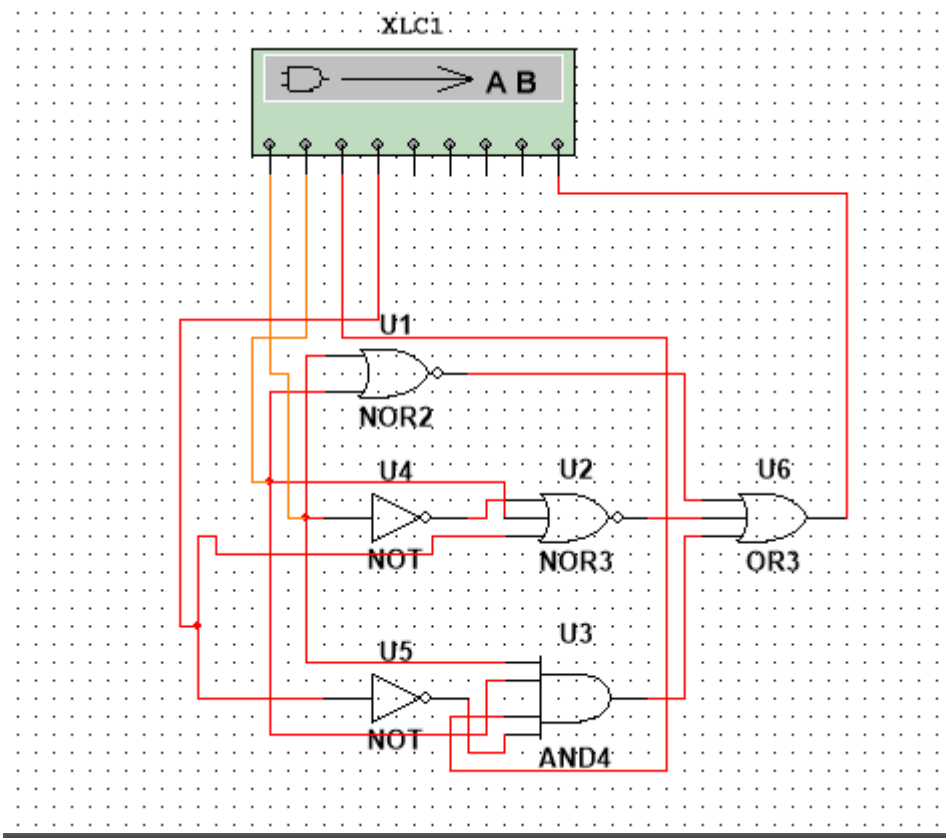
1 0 1 SIMP A|B

A|B → 1 0 1

A|B →

A|B → NAND

(d)



逻辑变换器-XLC1

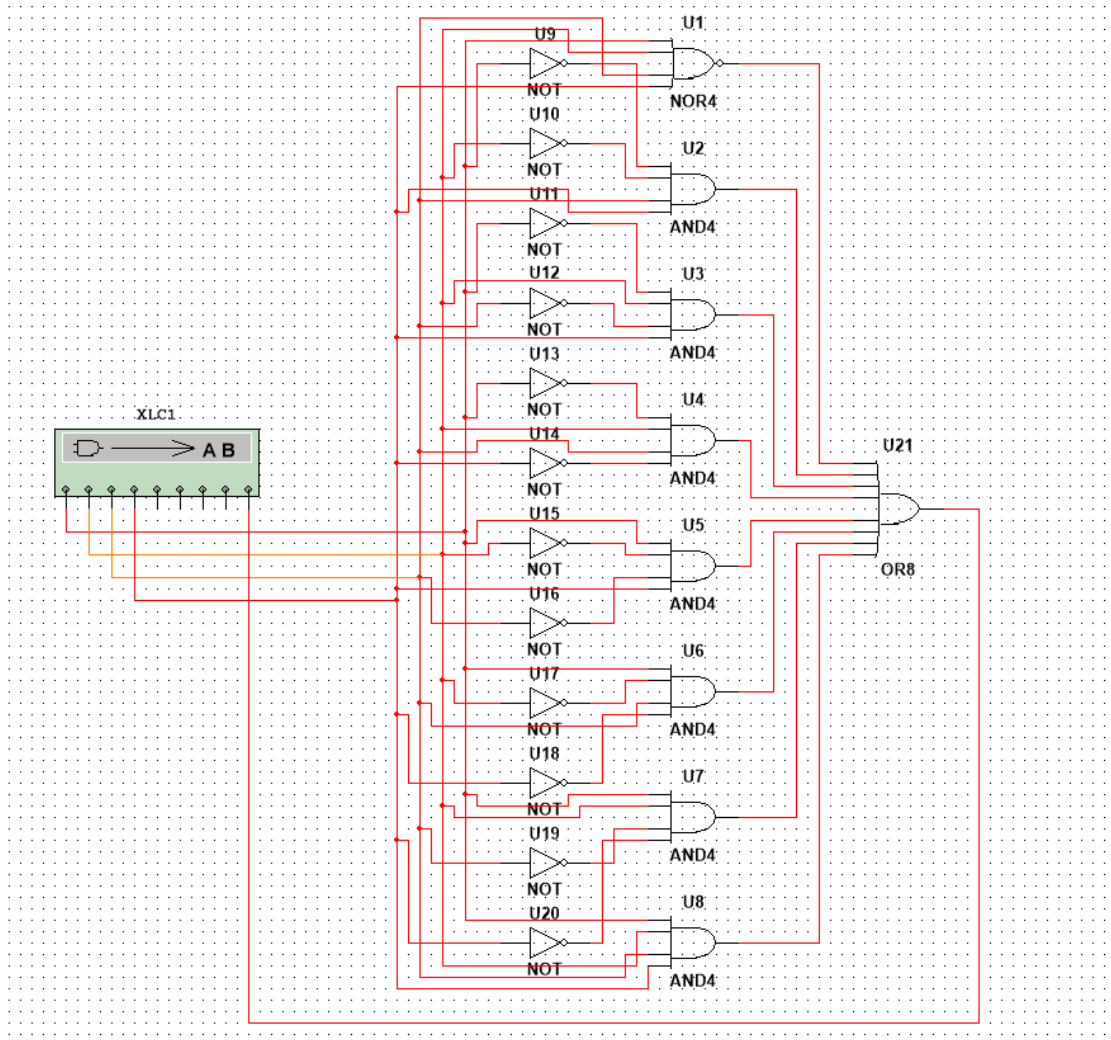
出

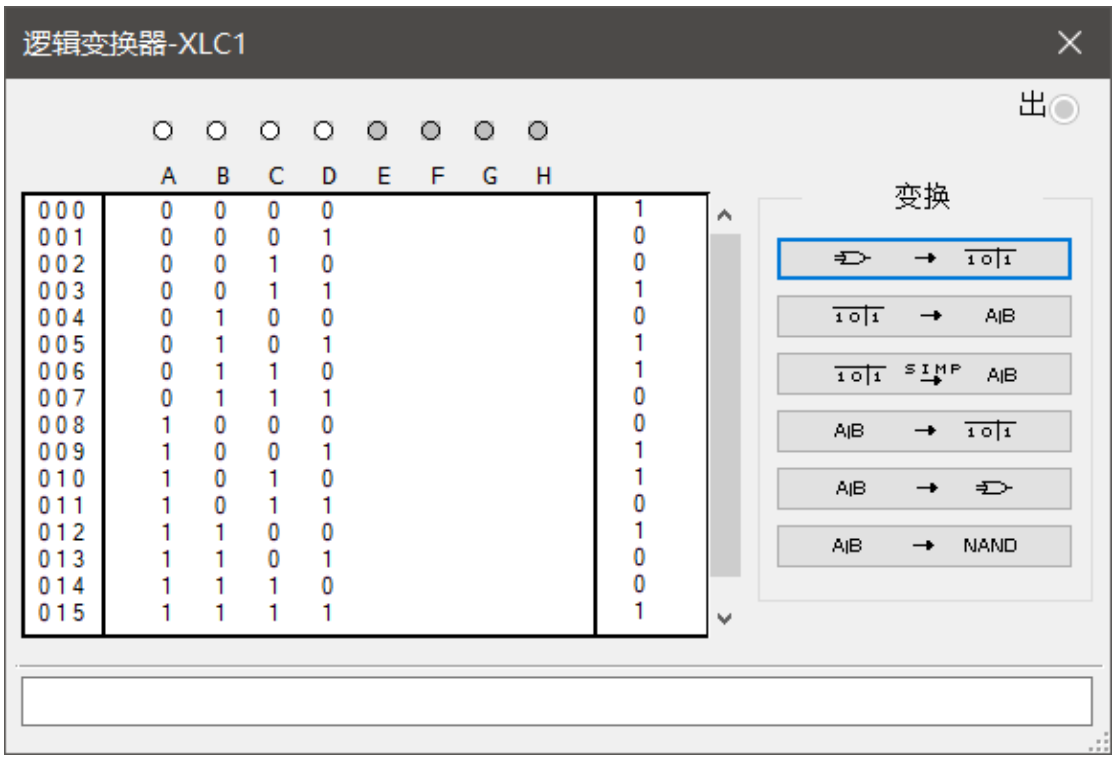
	A	B	C	D	E	F	G	H	
000	0	0	0	0					1
001	0	0	0	1					1
002	0	0	1	0					1
003	0	0	1	1					1
004	0	1	0	0					0
005	0	1	0	1					0
006	0	1	1	0					0
007	0	1	1	1					0
008	1	0	0	0					1
009	1	0	0	1					0
010	1	0	1	0					1
011	1	0	1	1					0
012	1	1	0	0					0
013	1	1	0	1					0
014	1	1	1	0					1
015	1	1	1	1					0

变换

AB NAND

(e)


U9
NOT
U10
NOT
U11
NOT
U12
NOT
U13
NOT
U14
NOT
U15
NOT
U16
NOT
U17
NOT
U18
NOT
U19
NOT
U20
NOT
U1
NOR4
U2
AND4
U3
AND4
U4
AND4
U5
AND4
U6
AND4
U7
AND4
U8
AND4
U21
OR8
XLC1
A B


逻辑变换器-XLC1
出
A B C D E F G H
000 0 0 0 0 1
001 0 0 0 1 0
002 0 0 1 0 0
003 0 0 1 1 1
004 0 1 0 0 0
005 0 1 0 1 1
006 0 1 1 0 1
007 0 1 1 1 0
008 1 0 0 0 0
009 1 0 0 1 1
010 1 0 1 0 1
011 1 0 1 1 0
012 1 1 0 0 1
013 1 1 0 1 0
014 1 1 1 0 0
015 1 1 1 1 1
变换
AB
AB
SIMP
AB
AB
AB
AB
NAND

3. Majority voting and 14-segment display

Referring to the majority voting system introduced in the lecture, design a logic circuit with Multisim and show the voting result with a 14-segment display as follows (T for Trump and b for Biden)

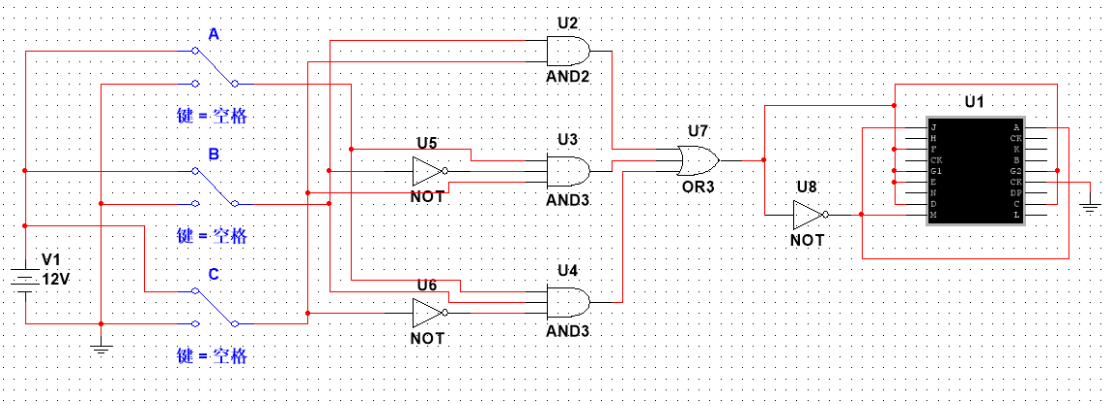
- Show the design procedures(Truth table, Boolean equations, etc) and final circuit schematic.

1 for Biden

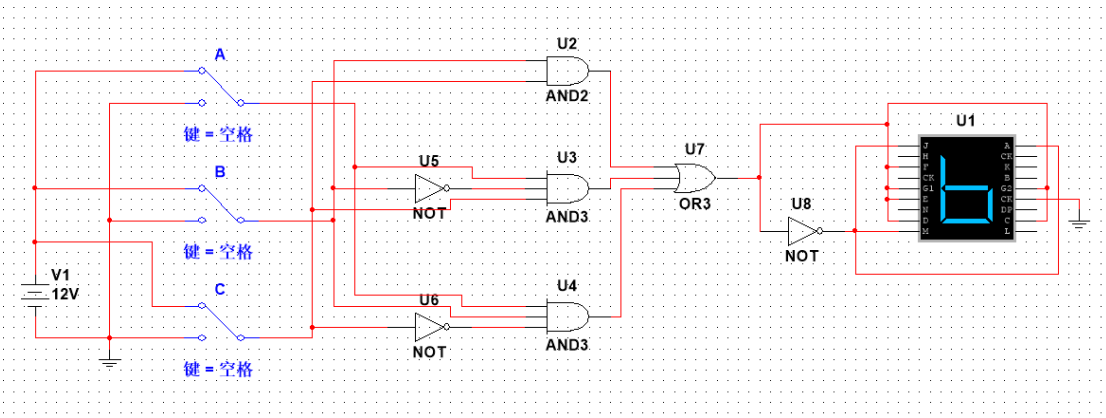
0 for Trump

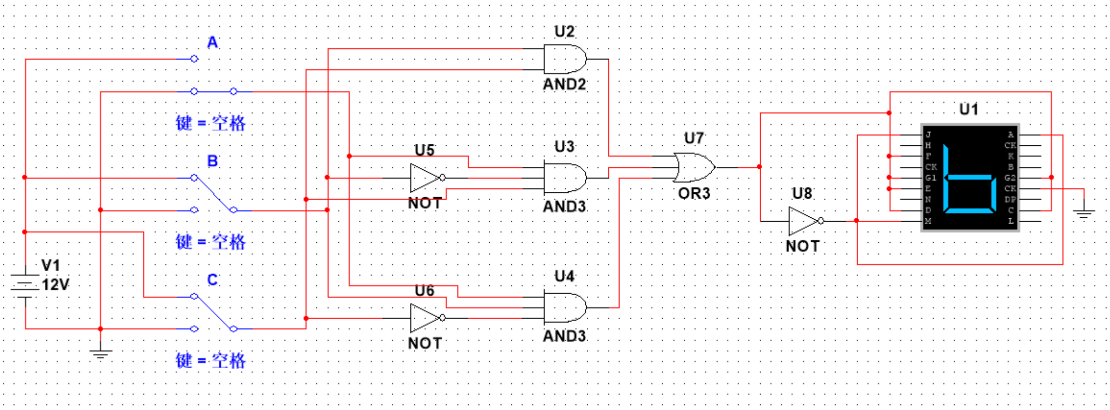
	BC	00	01	11	10
A		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
0	$\bar{A}$	0	0	1	0
1	A	0	1	1	1

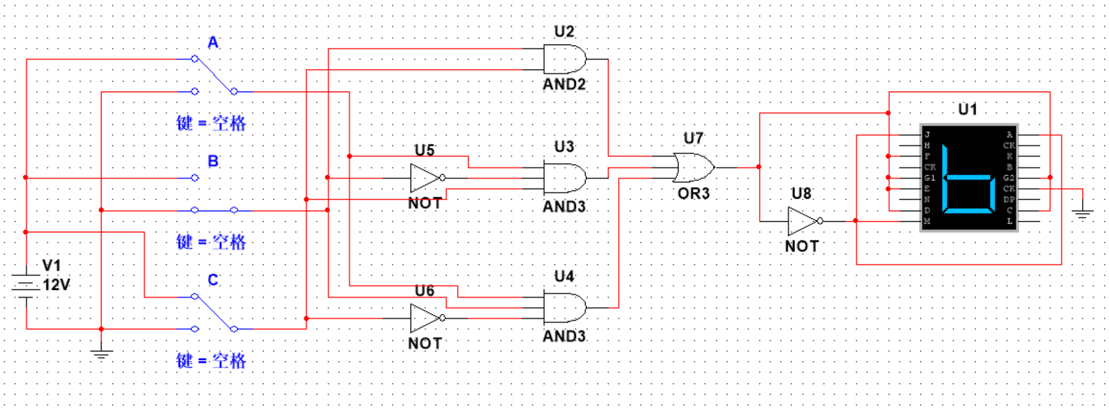
$BC+A\bar{B}C+AB\bar{C}$

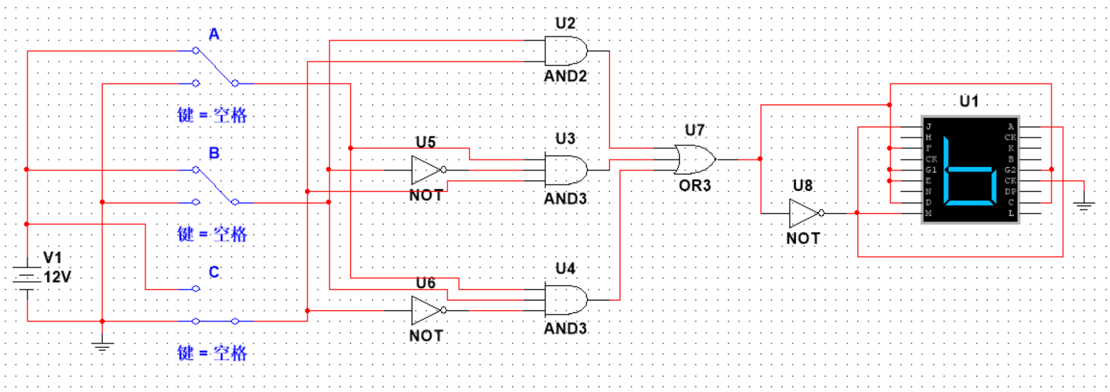


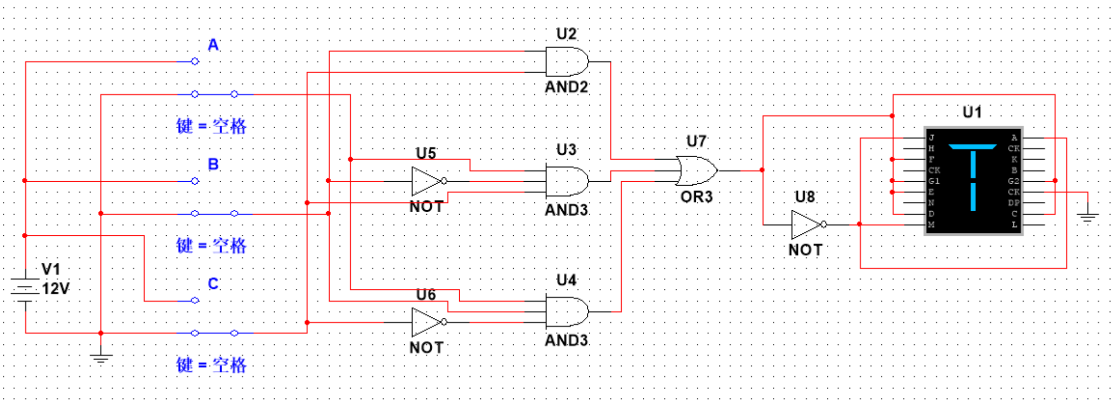
- Show the result pictures of eight input conditions.

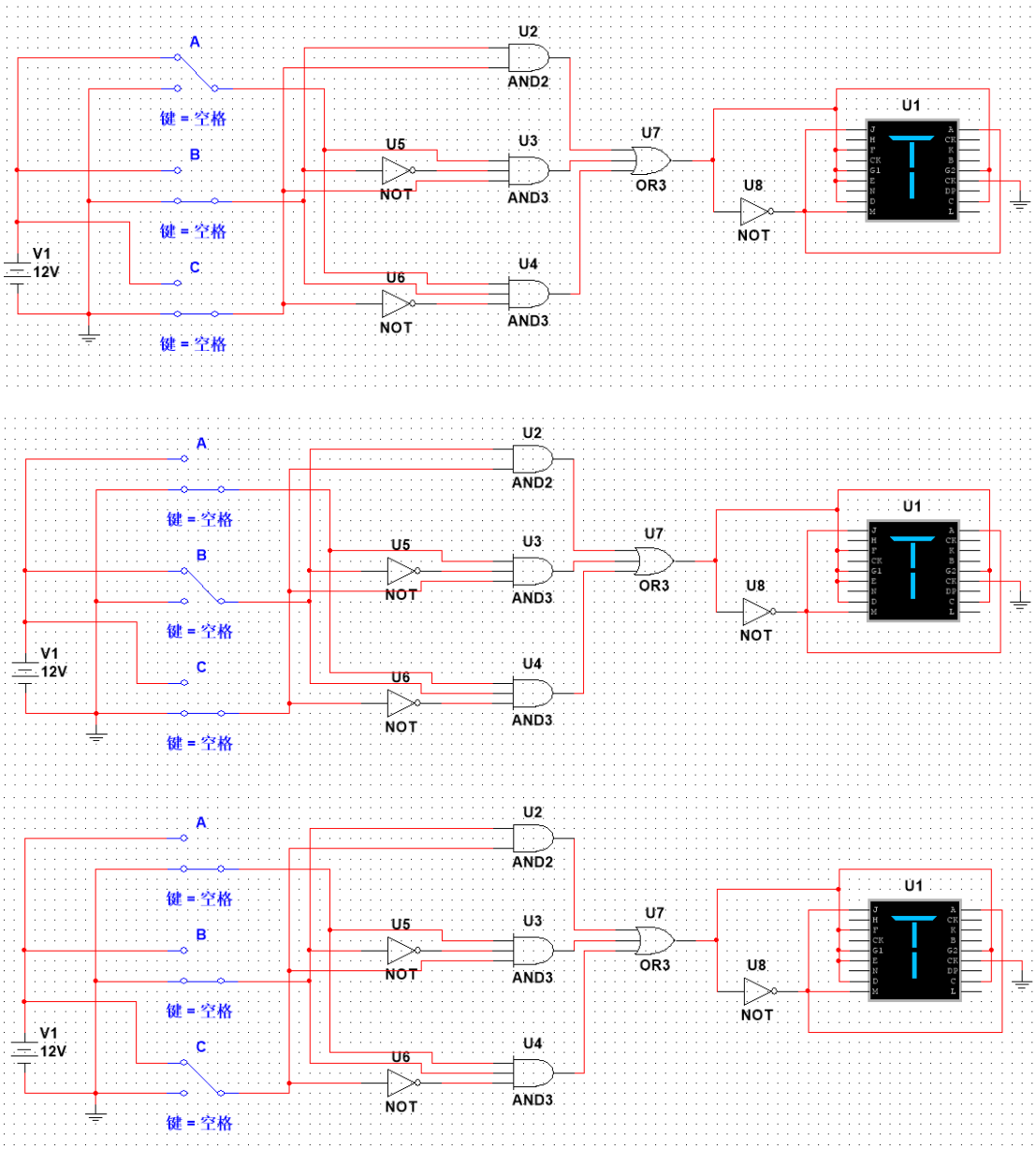



A
B
C
键 = 空格
键 = 空格
键 = 空格
V1
12V
U2
AND2
U3
AND3
U4
AND3
U5
NOT
U6
NOT
U7
OR3
U8
NOT
U1


A
B
C
键 = 空格
键 = 空格
键 = 空格
V1
12V
U2
AND2
U3
AND3
U4
AND3
U5
NOT
U6
NOT
U7
OR3
U8
NOT
U1


A
B
C
键 = 空格
键 = 空格
键 = 空格
V1
12V
U2
AND2
U3
AND3
U4
AND3
U5
NOT
U6
NOT
U7
OR3
U8
NOT
U1


A
B
C
键 = 空格
键 = 空格
键 = 空格
V1
12V
U2
AND2
U3
AND3
U4
AND3
U5
NOT
U6
NOT
U7
OR3
U8
NOT
U1



4. MCU development

- Make your esp32's onboard LED blink with 25Hz frequency for one second, then 5Hz for one second. Repeat this cycle (25Hz 5Hz 25Hz 5Hz 25Hz……) for 100 times, then turn off the LED

```
import machine
import time
pin2 = machine.Pin(2 , machine.Pin.OUT)
for i in range(100):
    for i in range(25):
        pin2.value(1)
        time.sleep(0.02)
        pin2.value(0)
        time.sleep(0.02)
```

```
    for i in range(10):
        pin2.value(1)
        time.sleep(0.1)
        pin2.value(0)
        time.sleep(0.02)
pin2.value(0)
```