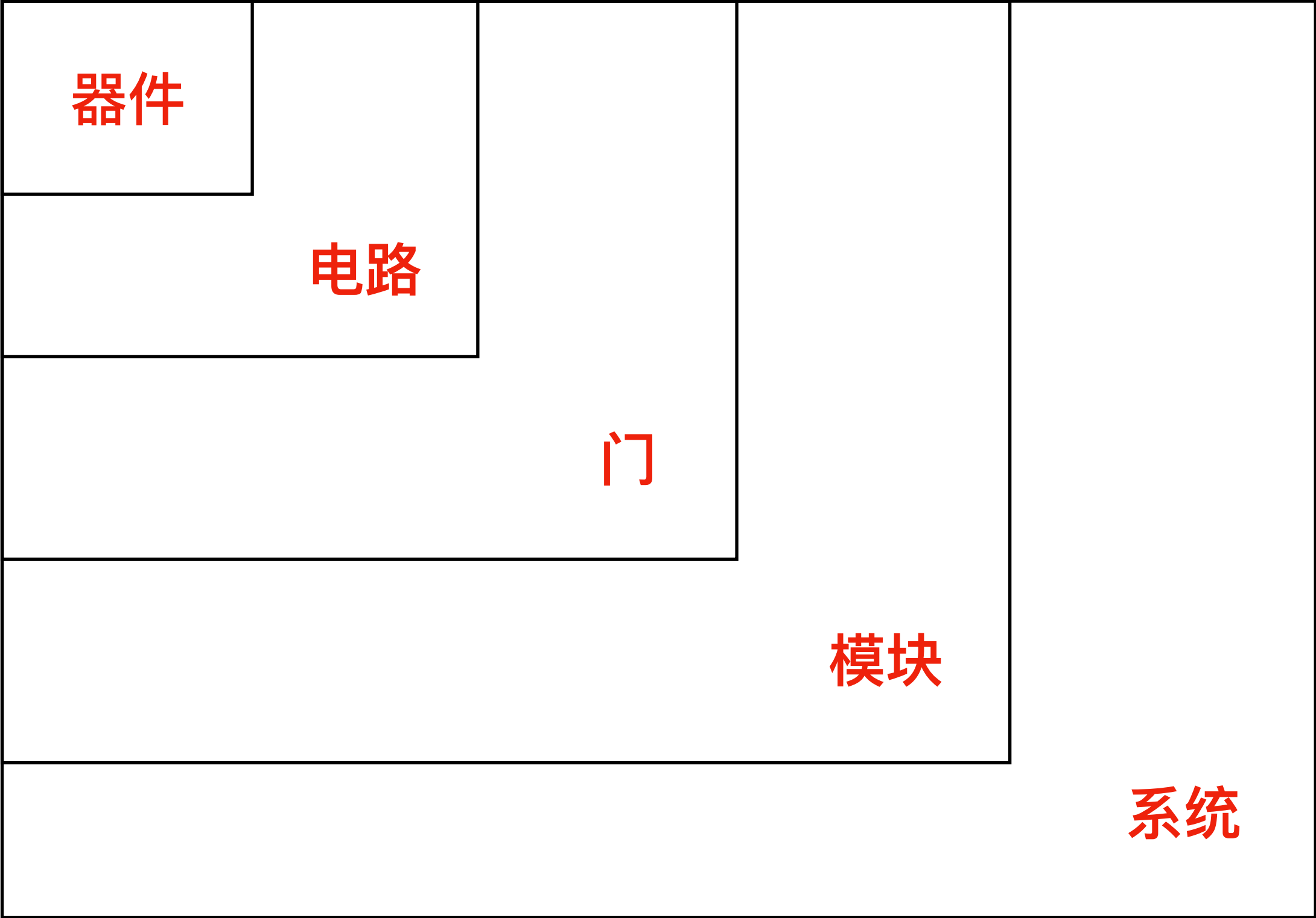


# 2-数字逻辑电路基础

搜狐焦点计算机基础学习系列课程

李少鹏 2019-02

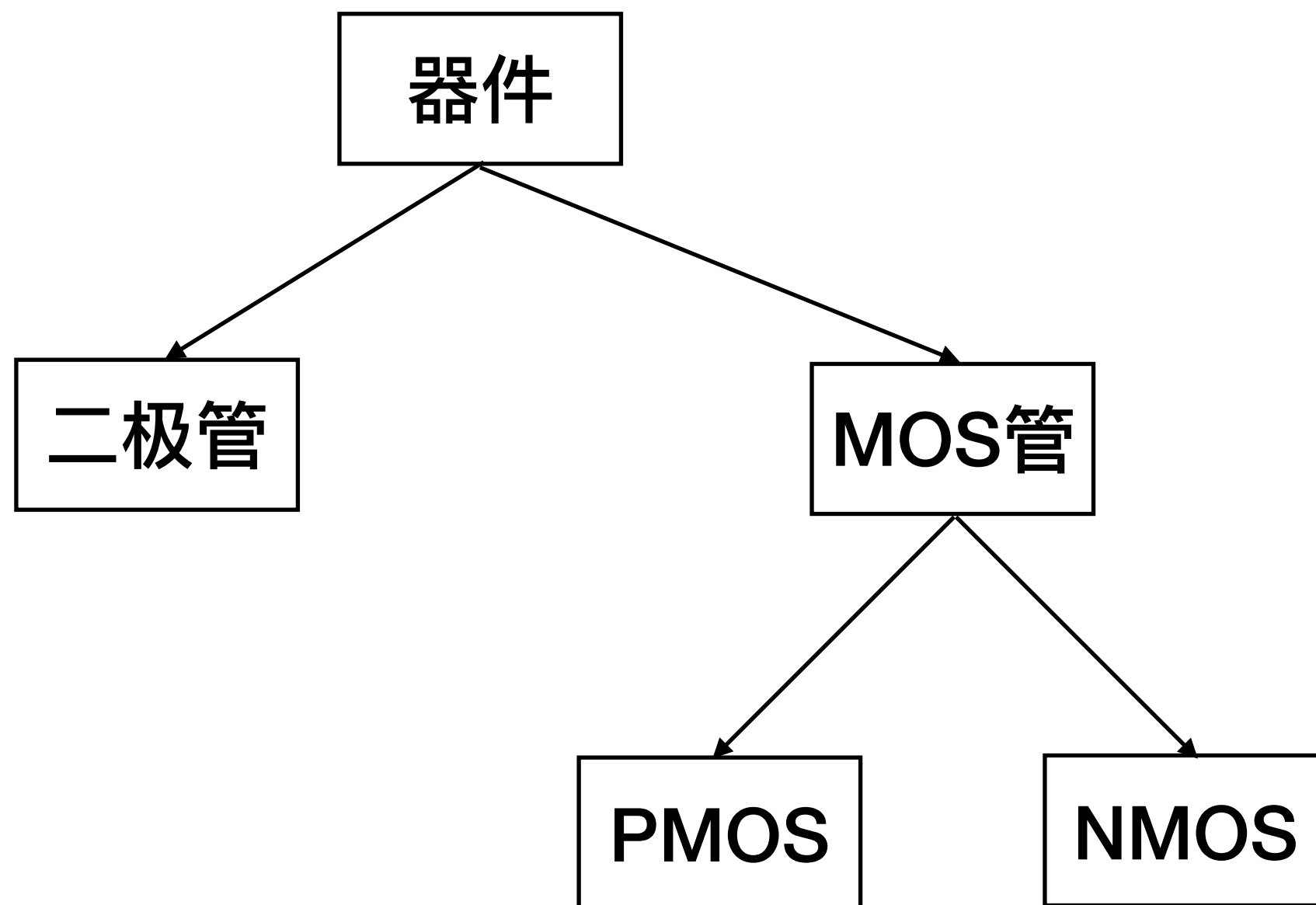
# 层级



**器件： 关键词**

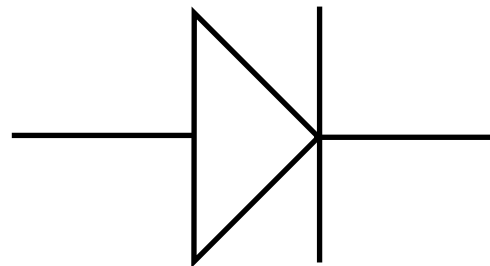
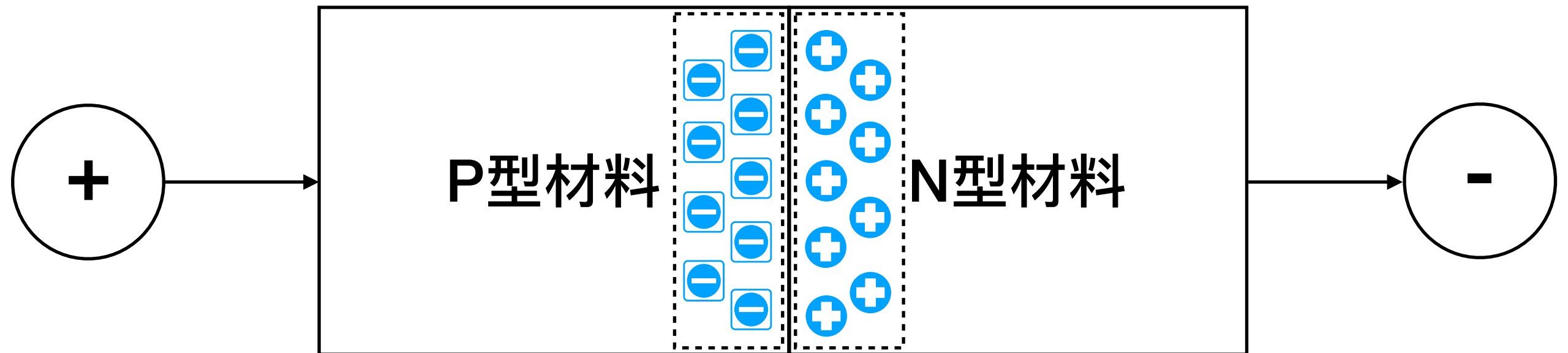
**二极管、PMOS管、NMOS管**

# 器件：结构图

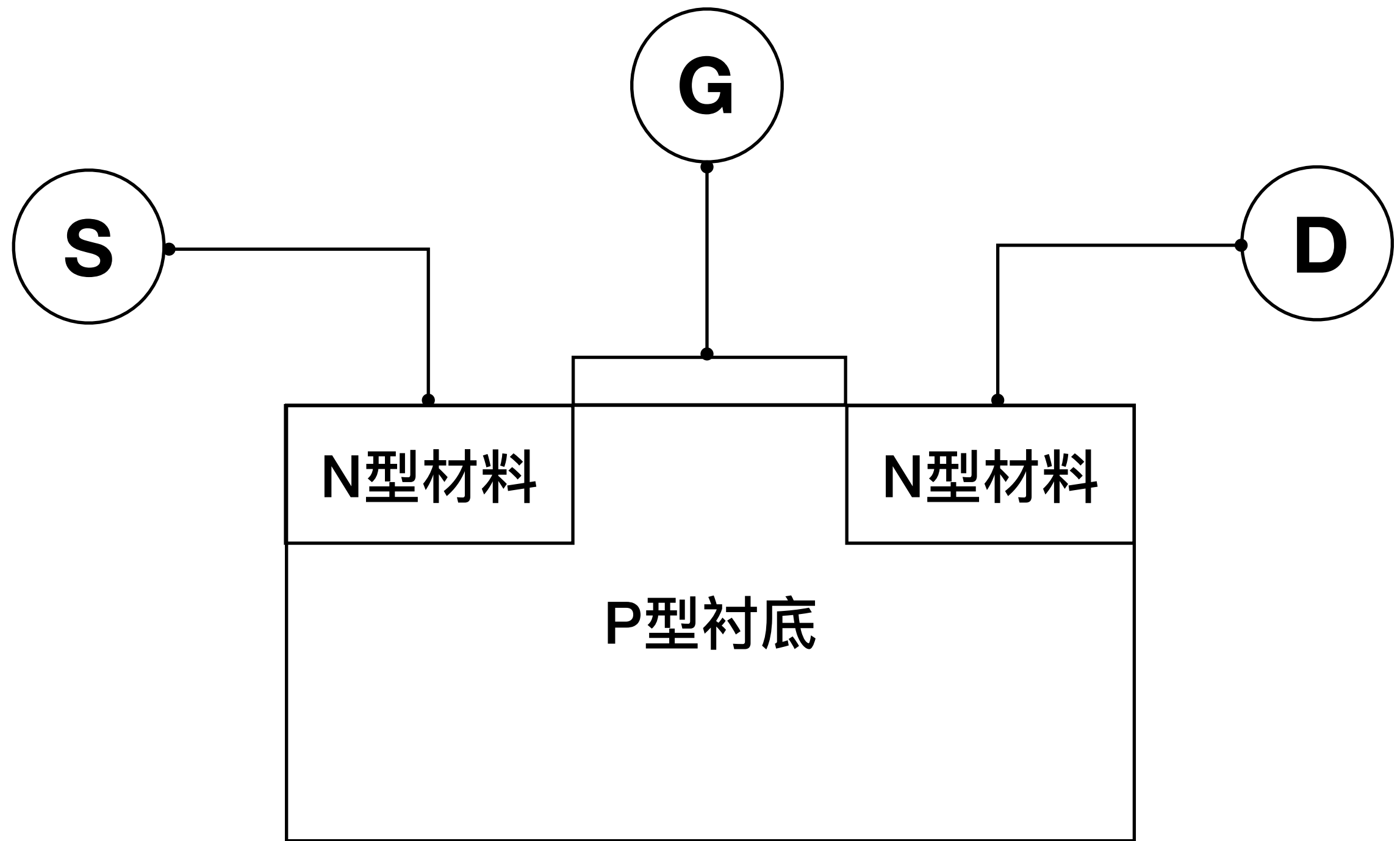


**CMOS电路所需核心器件**

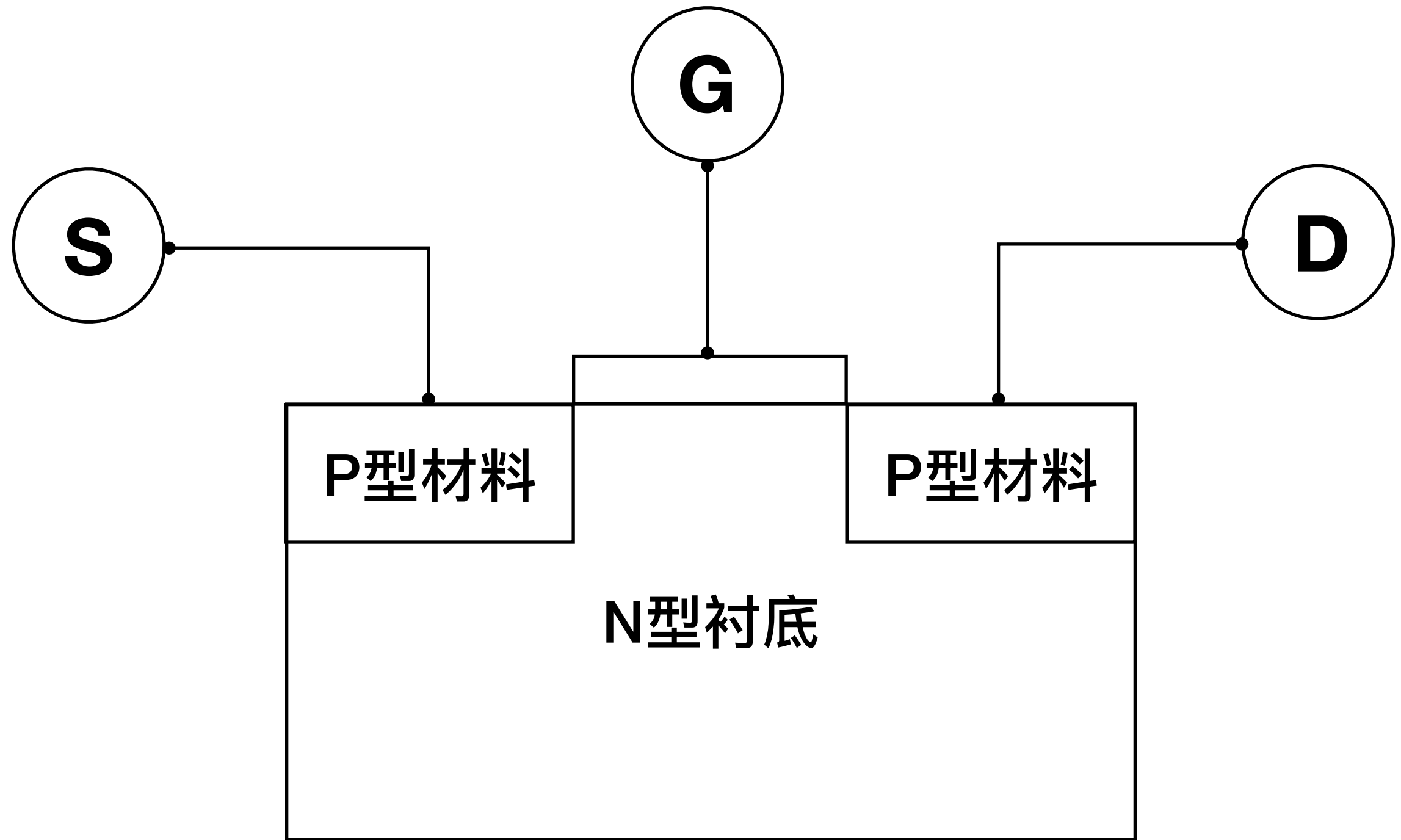
# 器件：二极管



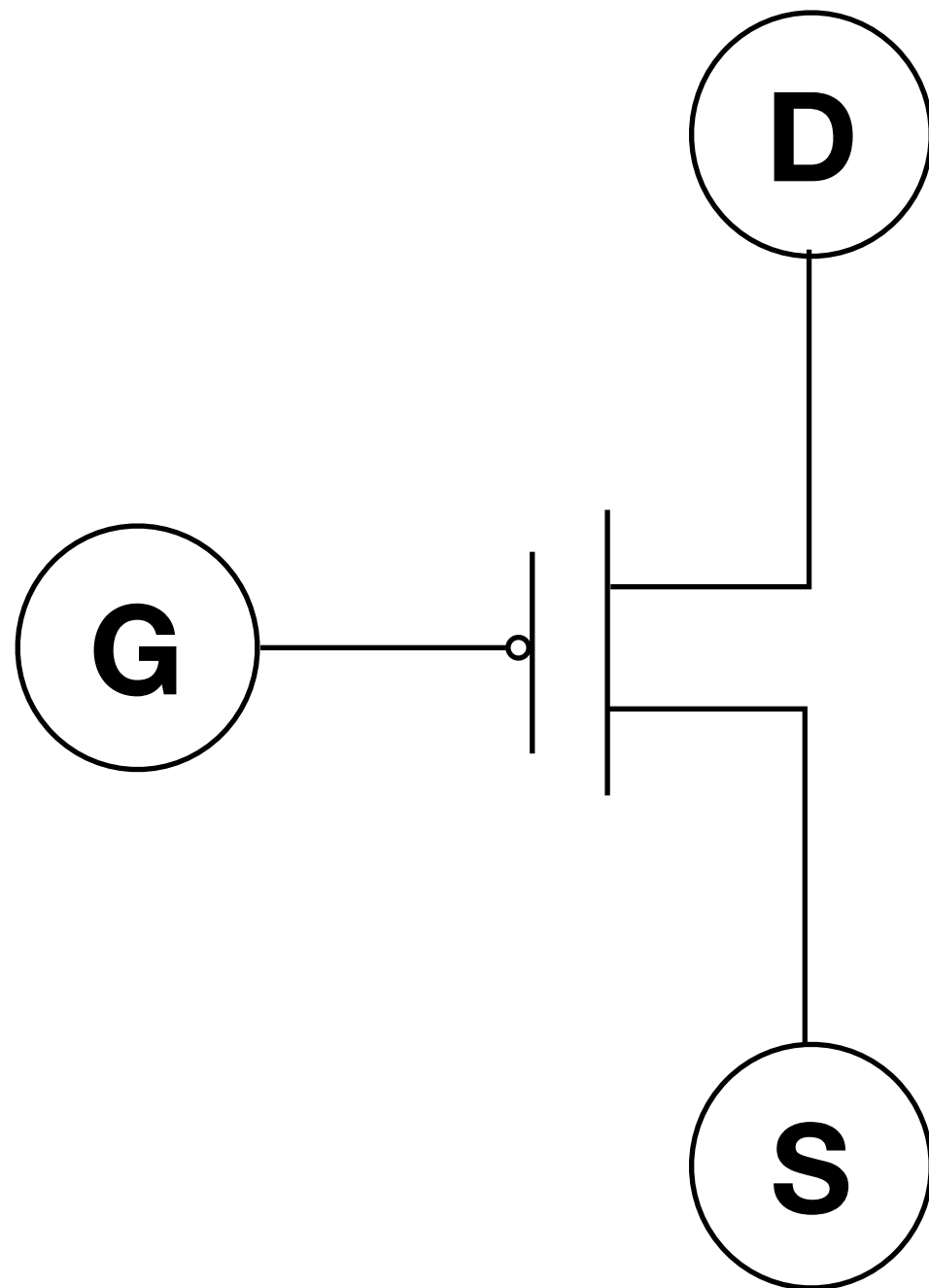
# 器件：NMOS管



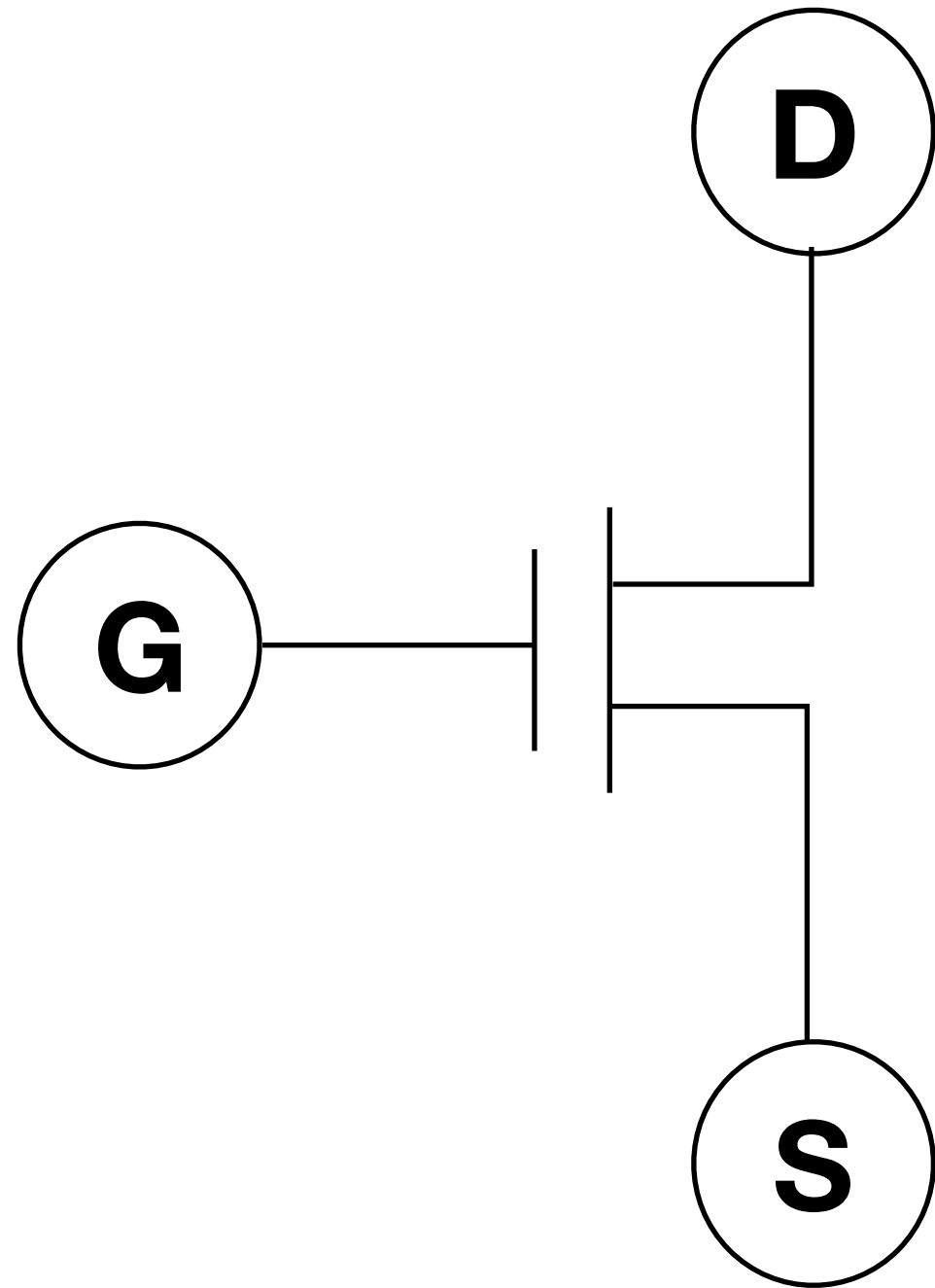
# 器件：PMOS管



# 器件：PMOS和NMOS的逻辑表示



PMOS



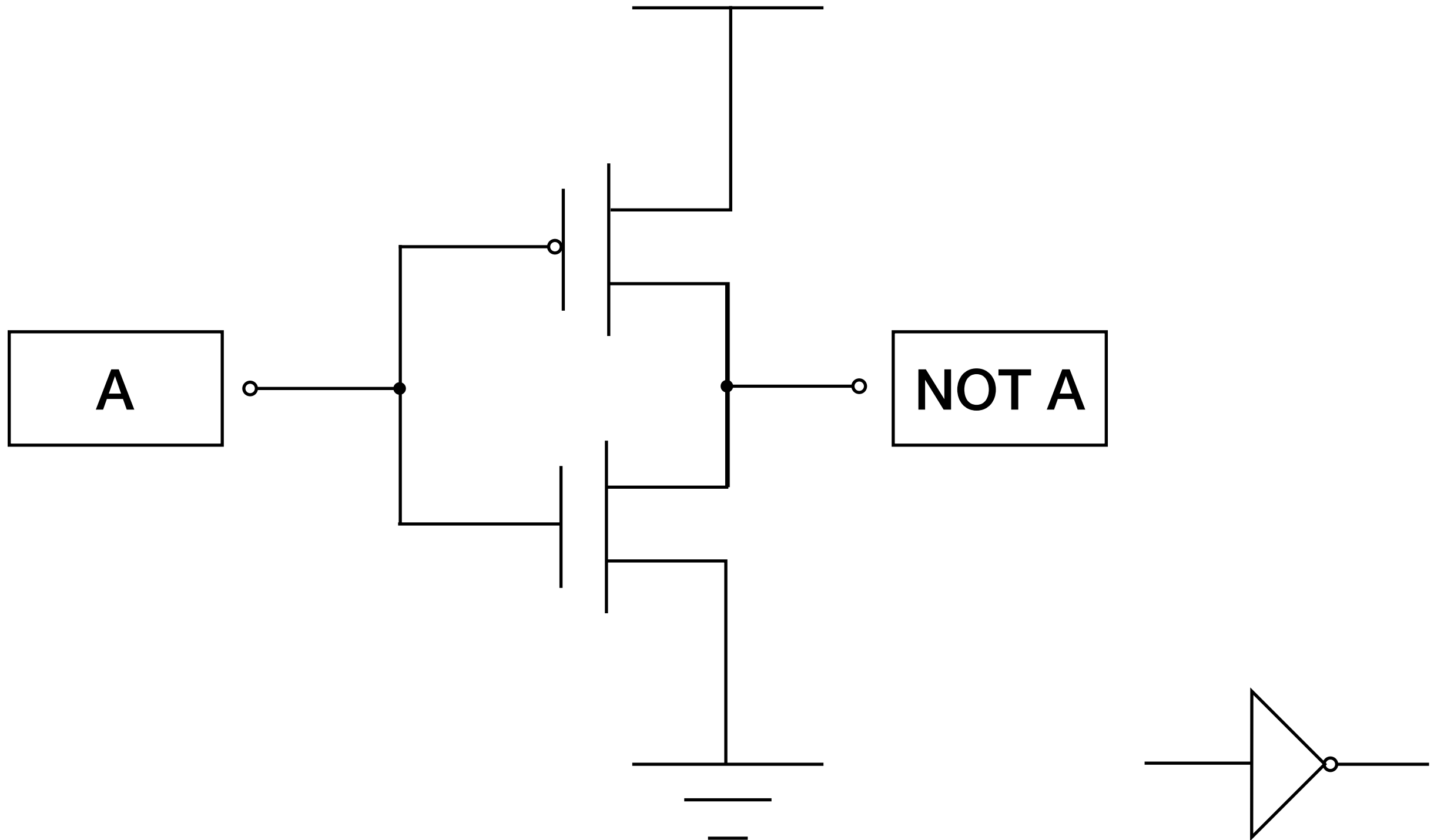
NMOS



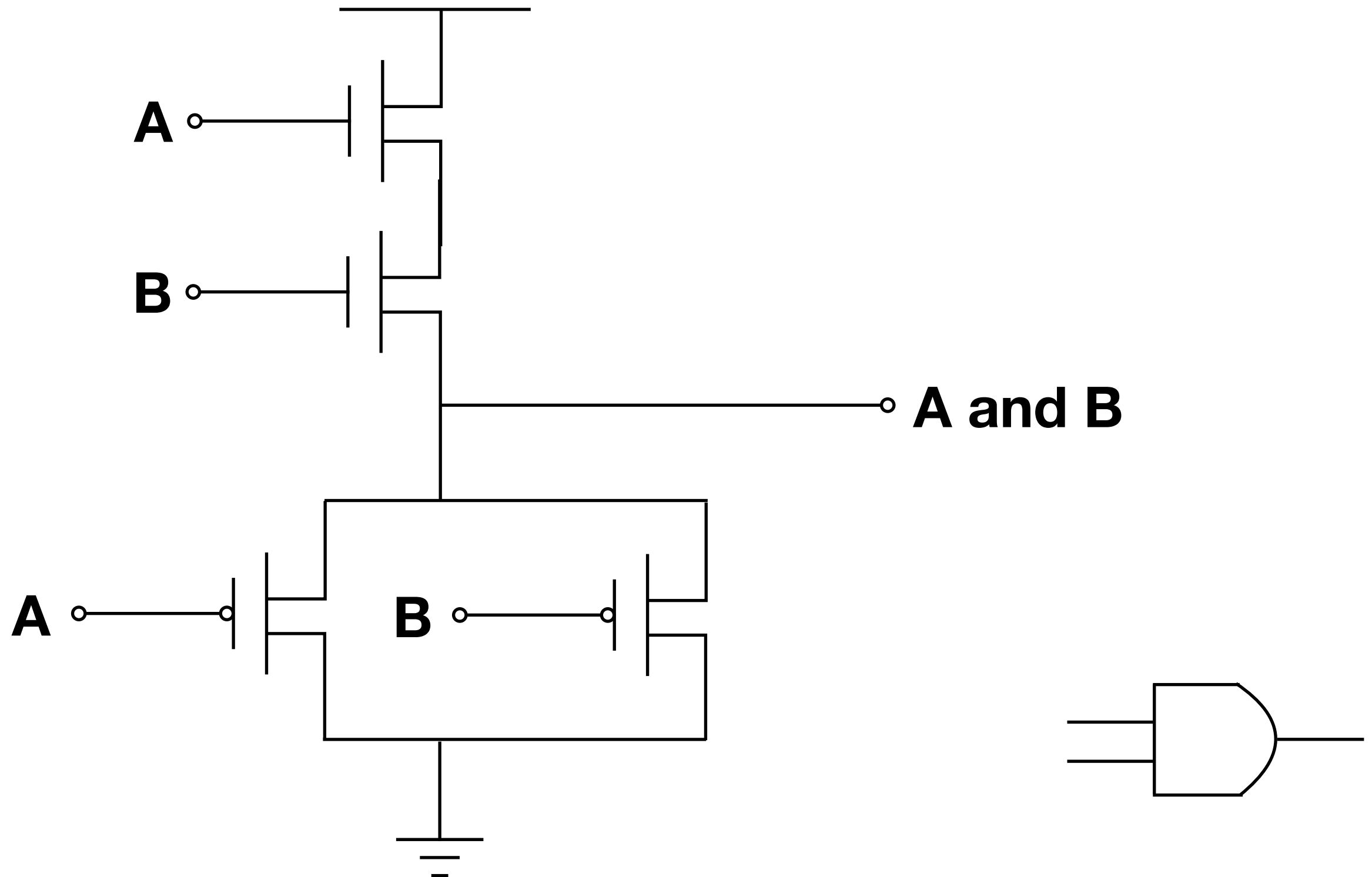
# 门：关键词

## 非门、与门、或门

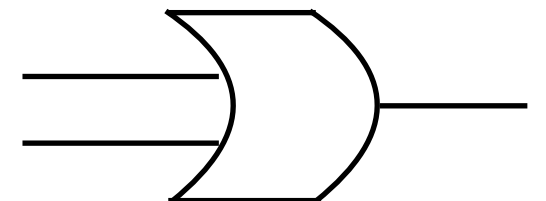
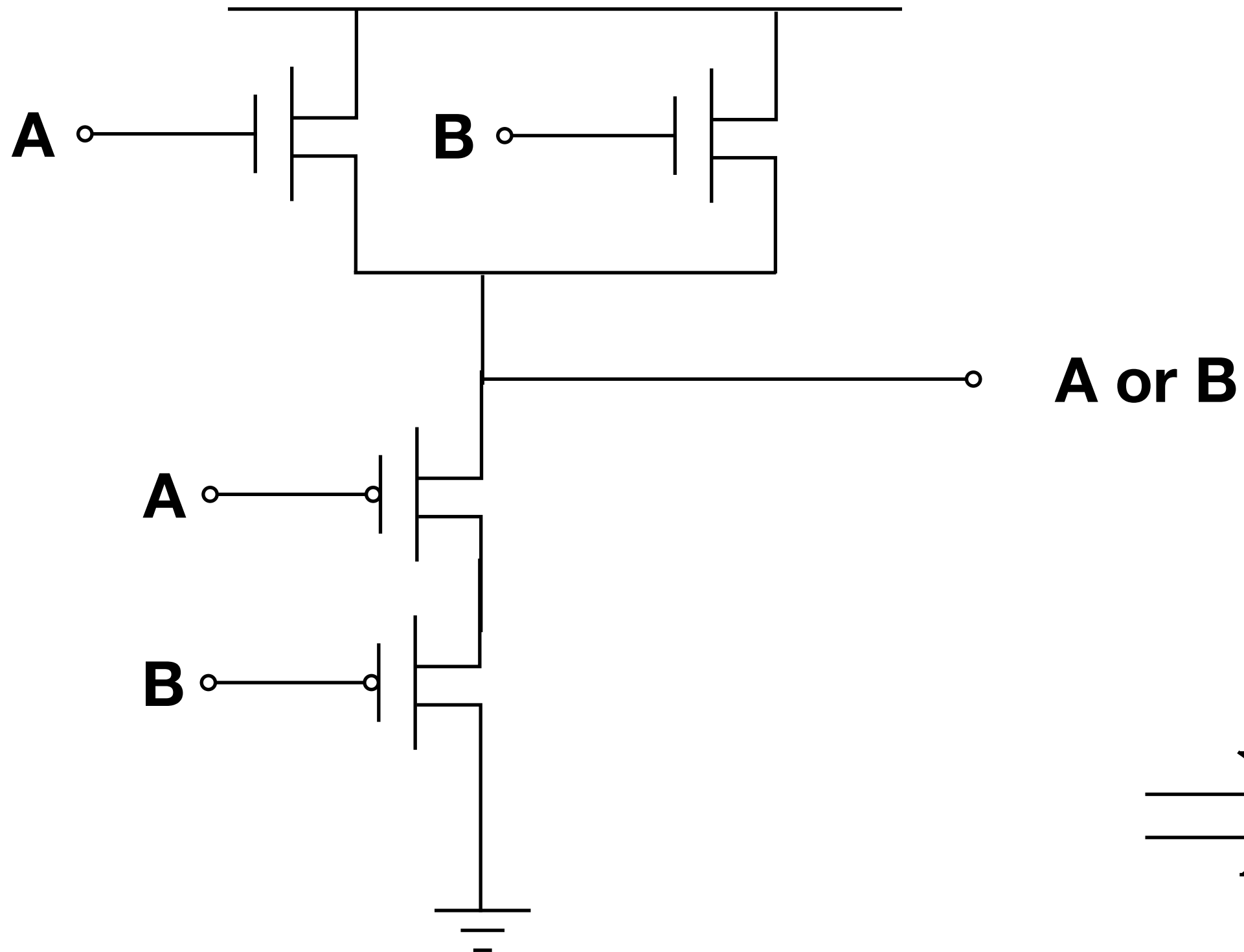
# 门：非门（反相器）



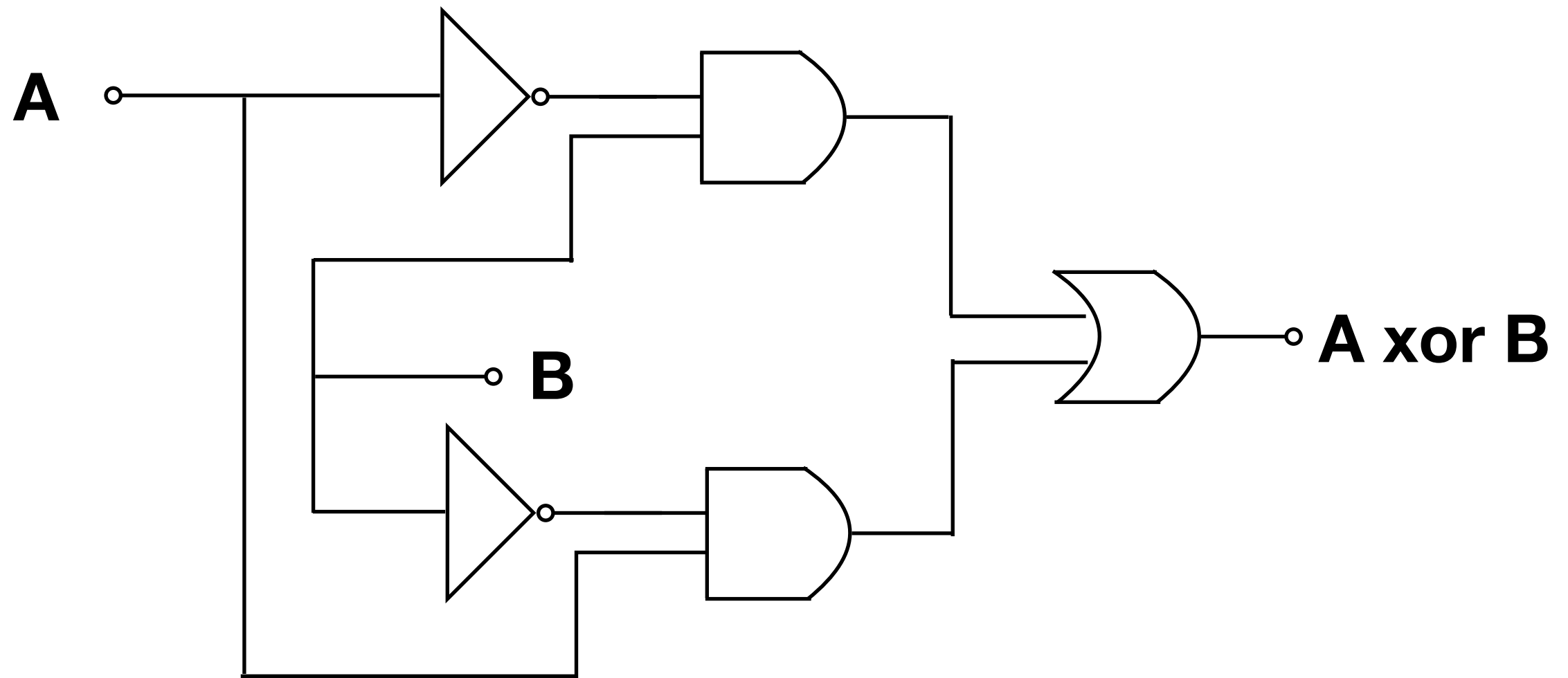
# 门：与门 (AND)



# 门：或门 (OR)



# 门：异或门 (XOR)



**$A \text{ xor } B : (A \text{ and } \neg B) \text{ or } (\neg A \text{ and } B)$**

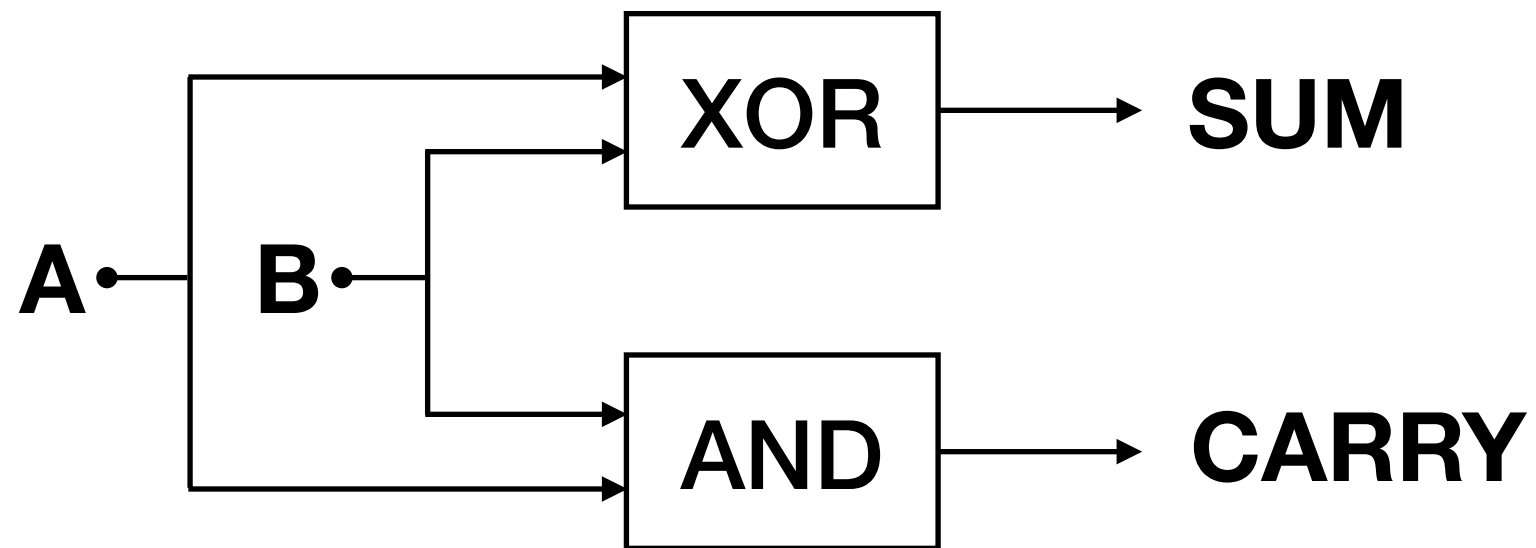
# 门: 6+1

门	中文	A-B=0-0	A-B=1-1	A-B=1-0
AND	与门	0	1	0
NAND	与非门	1	0	1
OR	或门	0	1	1
NOR	或非门	1	0	0
XOR	异或门	0	0	1
XNOR	同或门	1	1	0
		1	1	1
		0	0	0

**模块： 关键词**

**半加器、加法器、全加器、ALU**

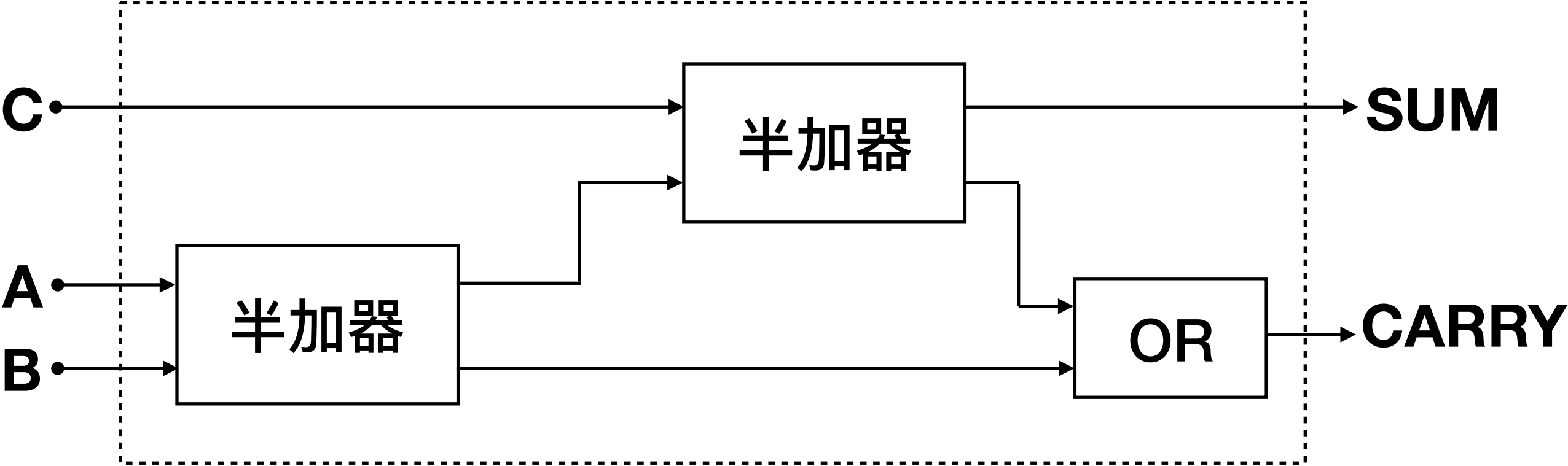
# 模块：半加器



输入A	输入B	输出CARRY	输出SUM
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	0

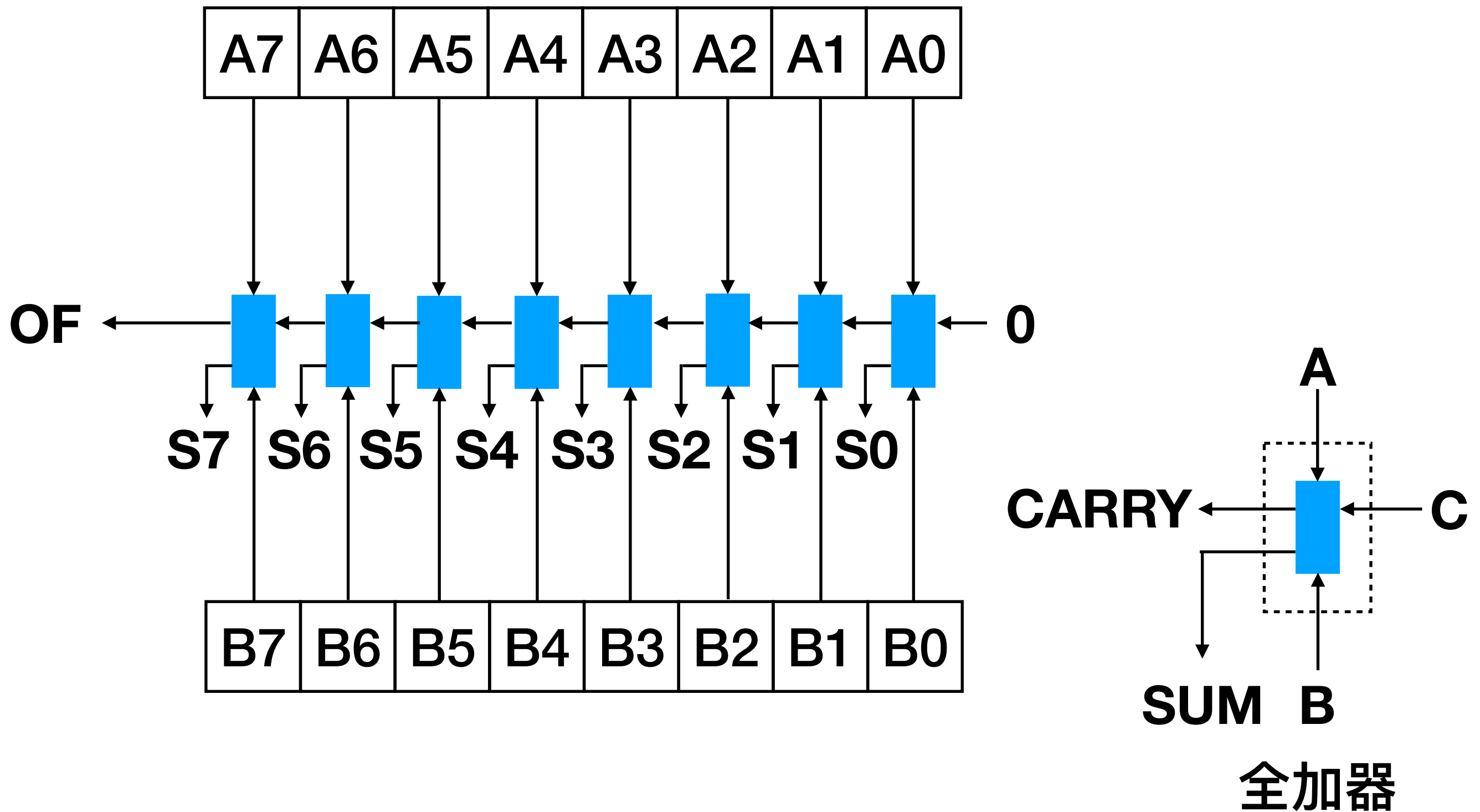


# 模块：全加器

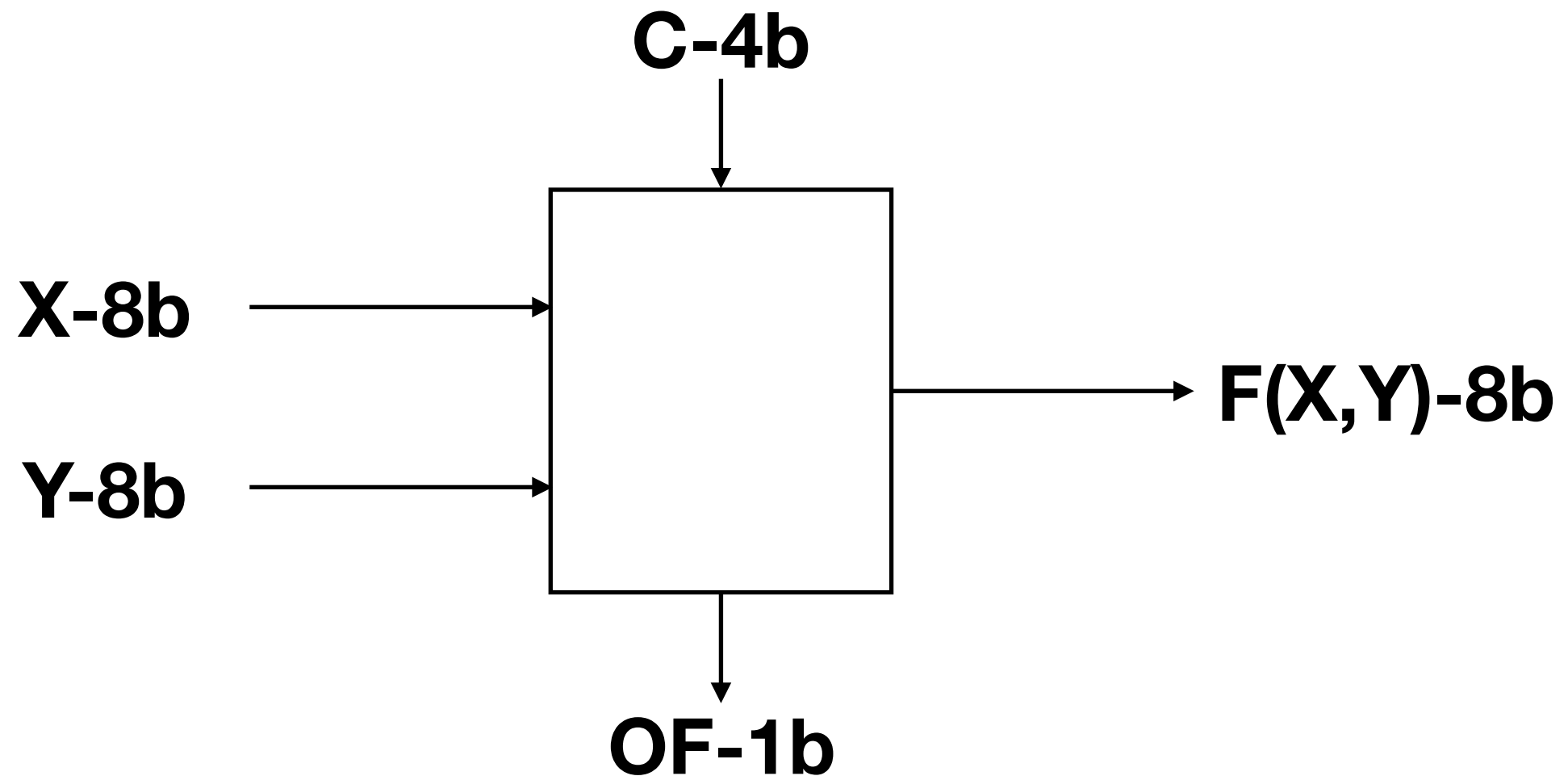


输入A	输入B	输入C	输出CARRY	输出SUM
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

# 模块：N位加法器 (N=8)



# 模块：ALU-The Arithmetic Logic Unit

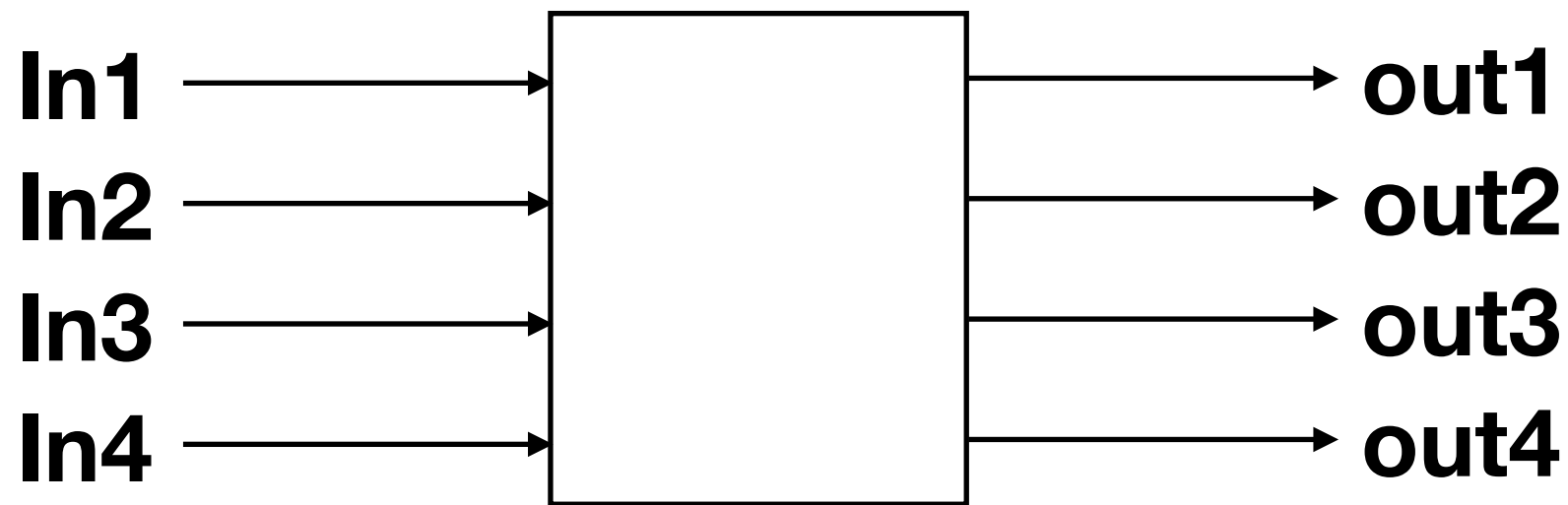


C	F(X,Y)
0001	X+Y
0010	X-Y
0011	Y-X
0100	X&Y

**系统： 关键词**

**组合逻辑电路、时序逻辑电路**

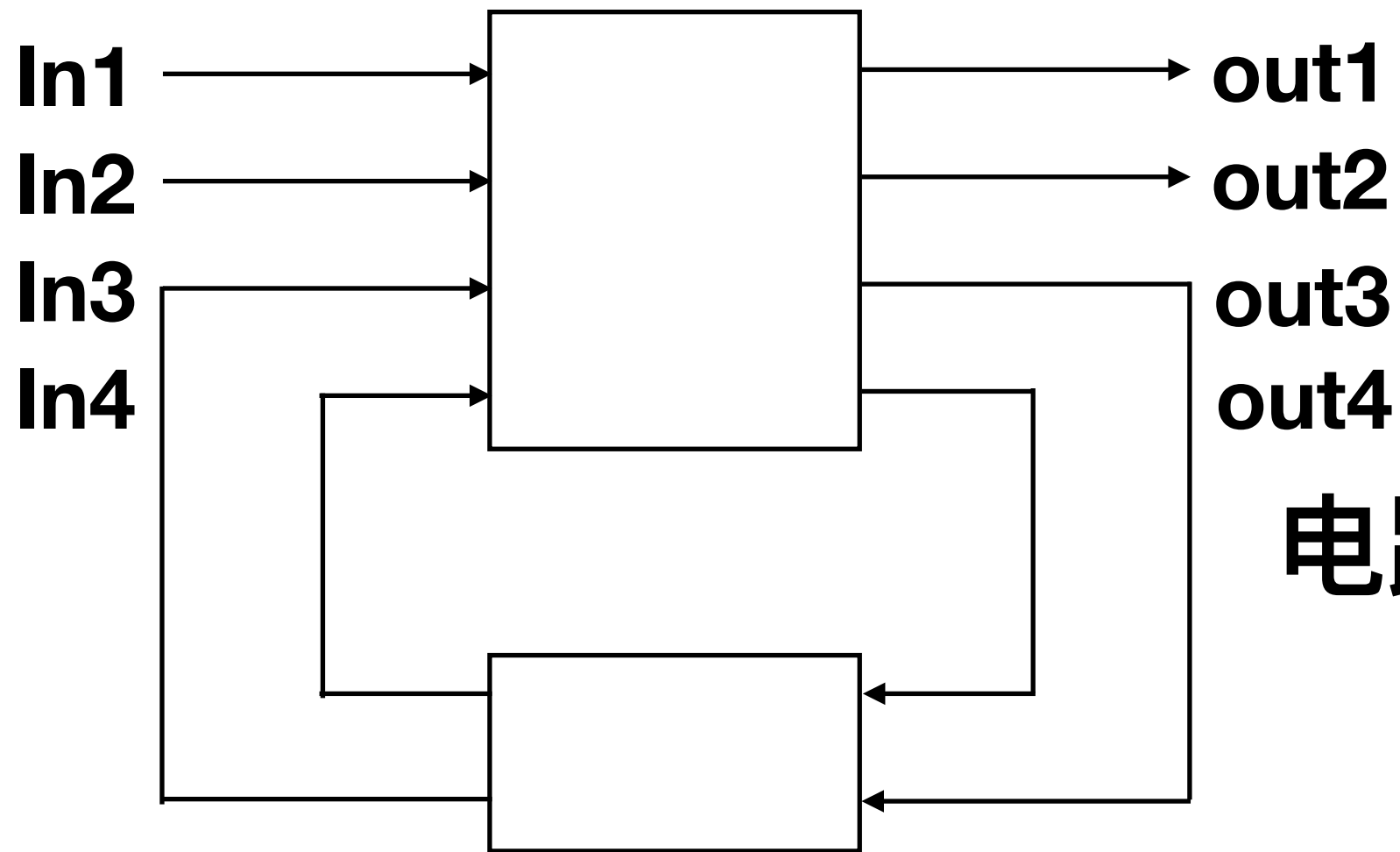
# 系统： 组合逻辑电路



本次输出只与本次输入有关系

总数：  $2^4 * 2^4$  个

# 系统： 时序逻辑电路



**电路中有环**

**本次输出与历史输入有关**

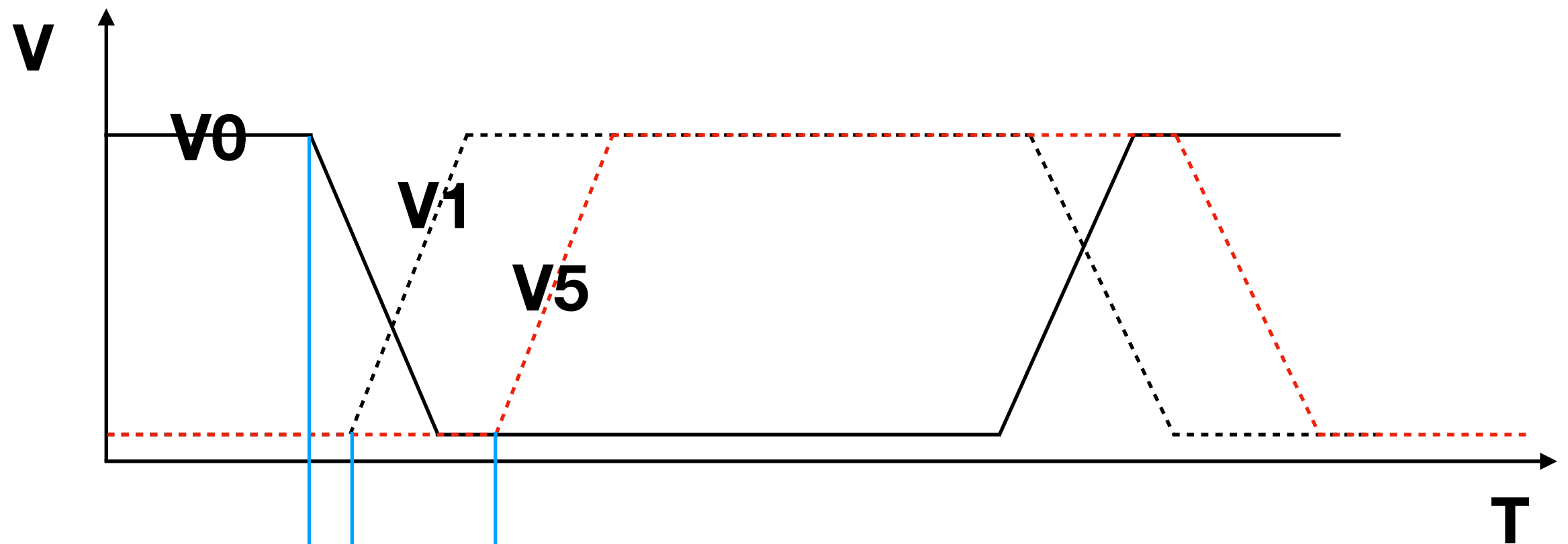
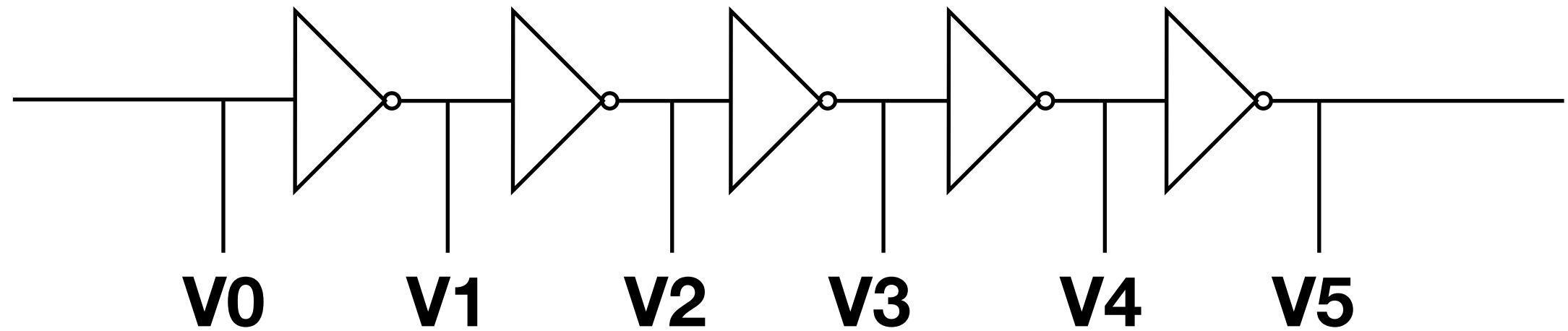
# 系统：时序逻辑电路——环的问题

In1	In2	In3	In4	Out1	Out2	Out3	Out4
1	1	0	0	0	0	1	1
1	1	1	1	1	1	0	0

**电路无法到达稳定状态**

**Out3、Out4以系统的固有电路延时为周期  
进行震荡**

# 系统：电路延时特性



延迟时间为皮秒级别



# 系统： 电路延时时间

$$1\text{秒} = 10^3\text{毫秒} \quad (\text{ms})$$

$$1\text{秒} = 10^6\text{毫秒} \quad (\mu\text{s})$$

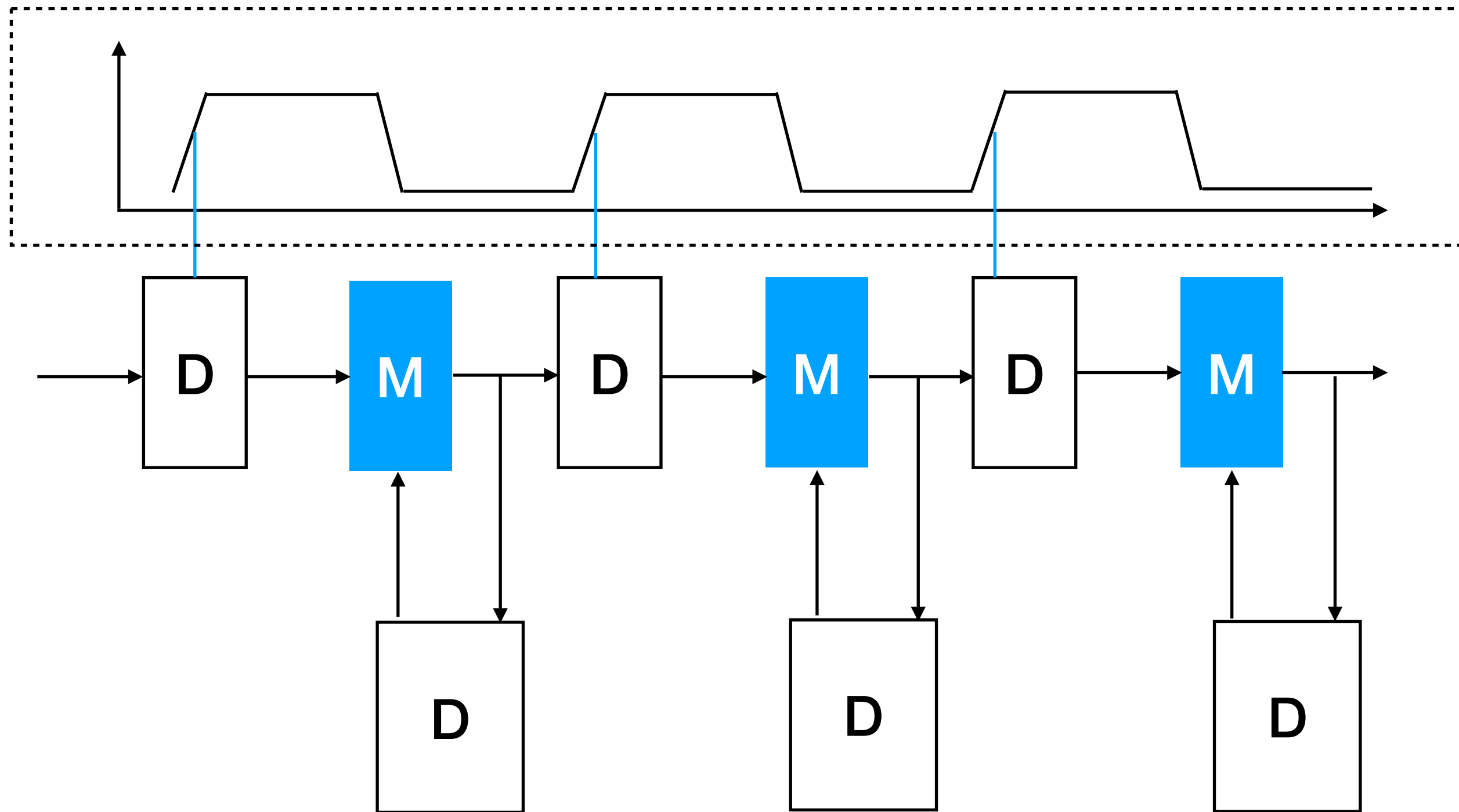
$$1\text{秒} = 10^9\text{纳秒} \quad (\text{ns})$$

$$1\text{秒} = 10^{12}\text{皮秒} \quad (\text{ps})$$

**系统：实现时序逻辑电路要解决的问题**

**如何统一系统中各个模块的传播延时**

# 系统：时序逻辑电路的解决思路



**D：触发器    M：模块**

# 触发器：总览

**RS锁存器、D锁存器、D触发器**

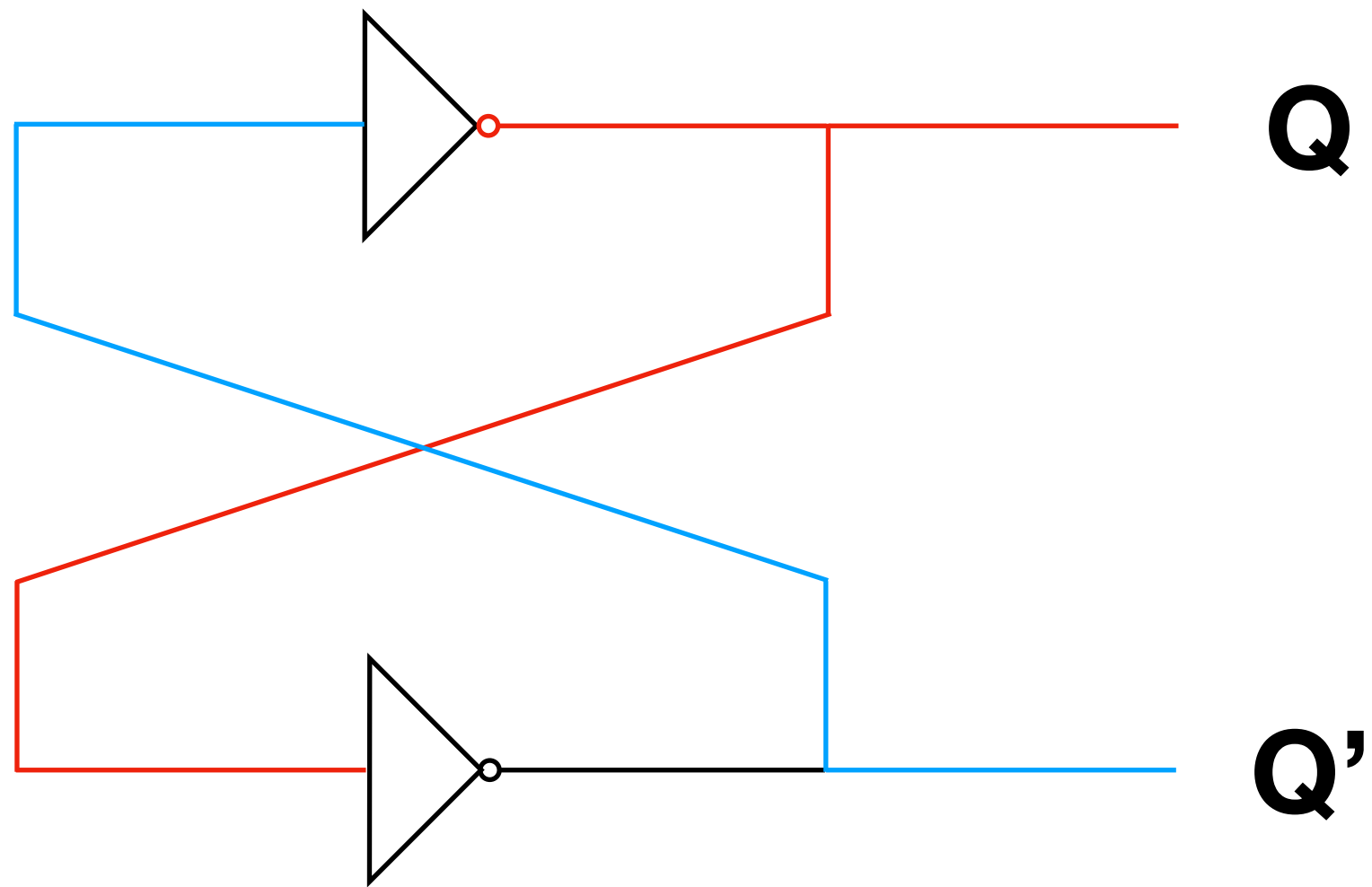
# 触发器：形象理解

触发器就像一扇门  
在每个时钟上升沿打开一次  
放过一个信号  
然后又立刻关上

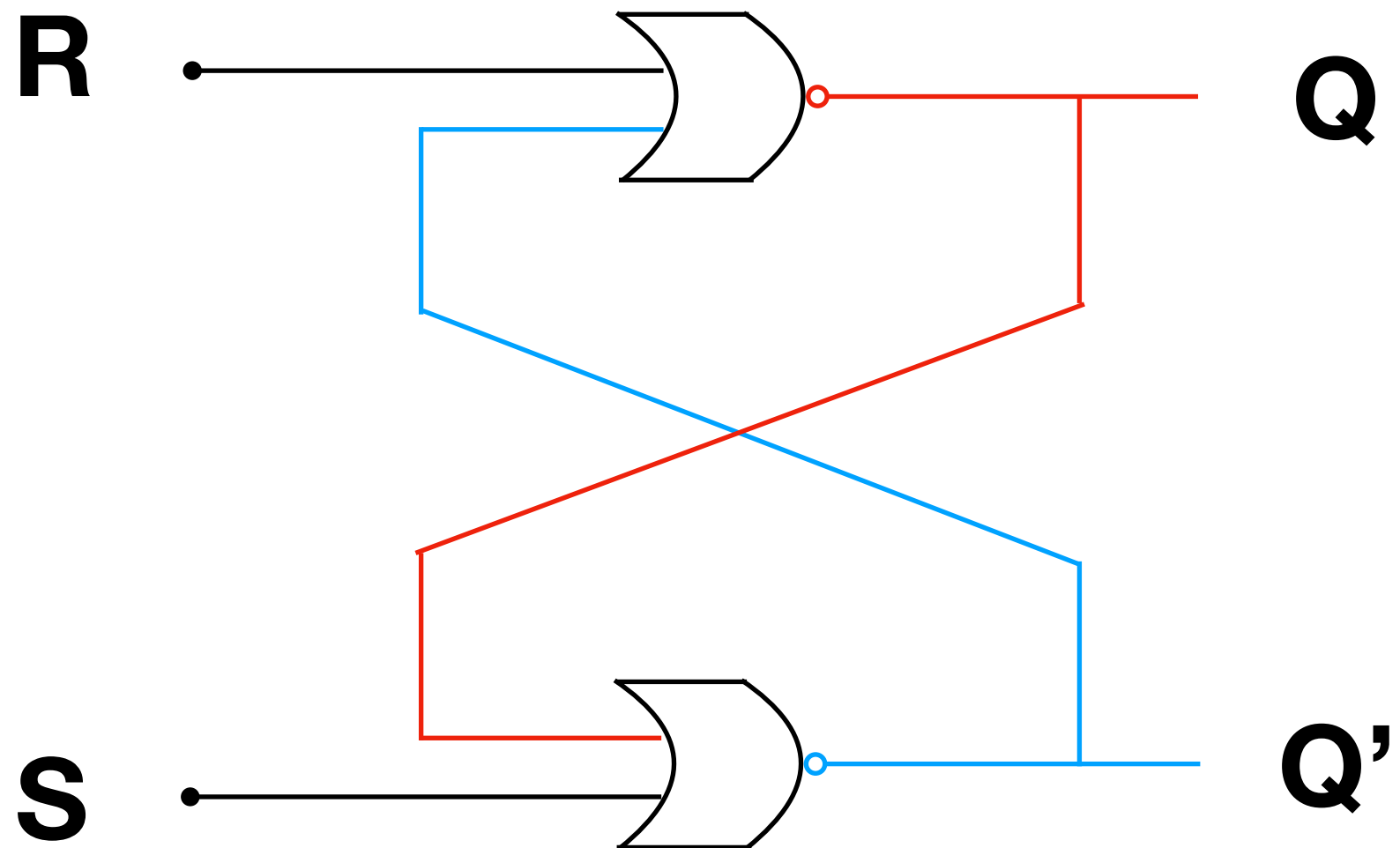
# 触发器：问题转化

## 如何实现寄存器与触发器？

# 触发器：双稳态电路



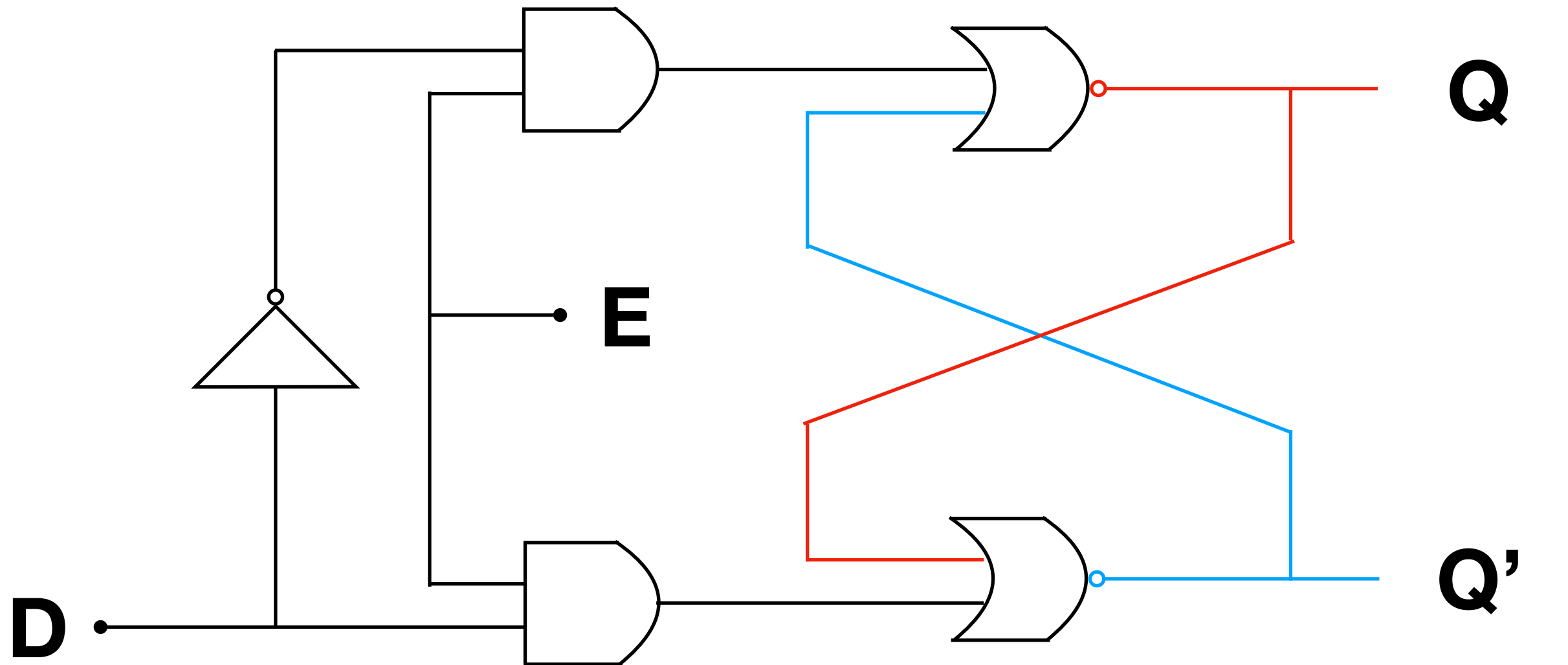
# 触发器：RS锁存器



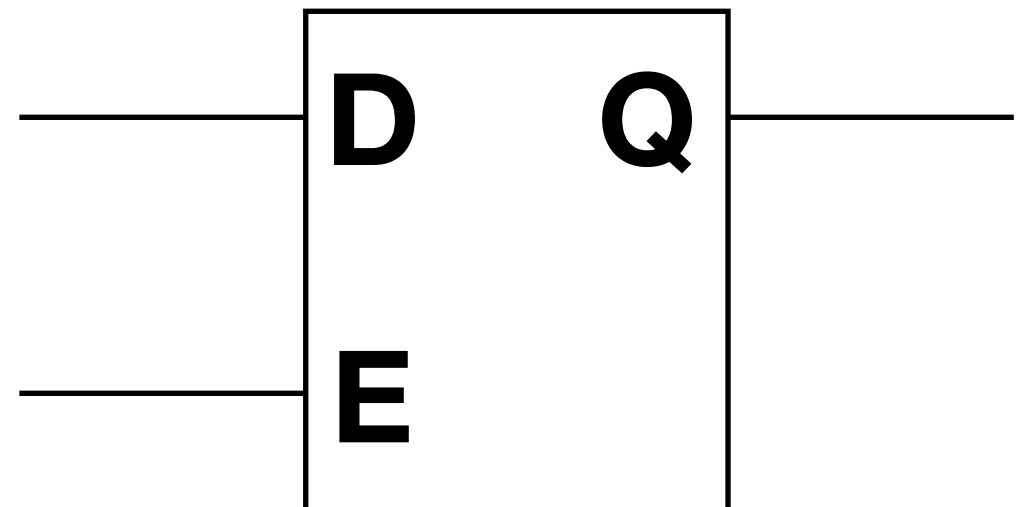
R	S	Q	Q'	用途
0	0	-	-	保持
1	0	0	1	RESET
0	1	1	0	SET
1	1	?	?	不用



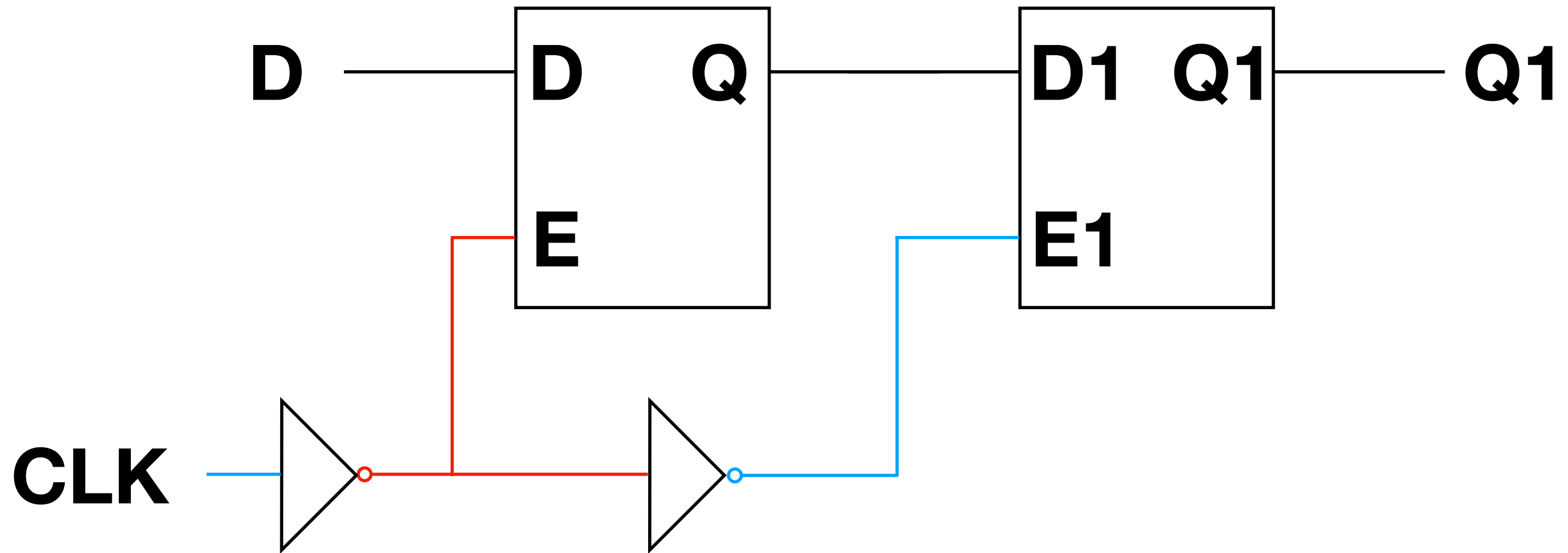
# 触发器：D锁存器



寄存器实现

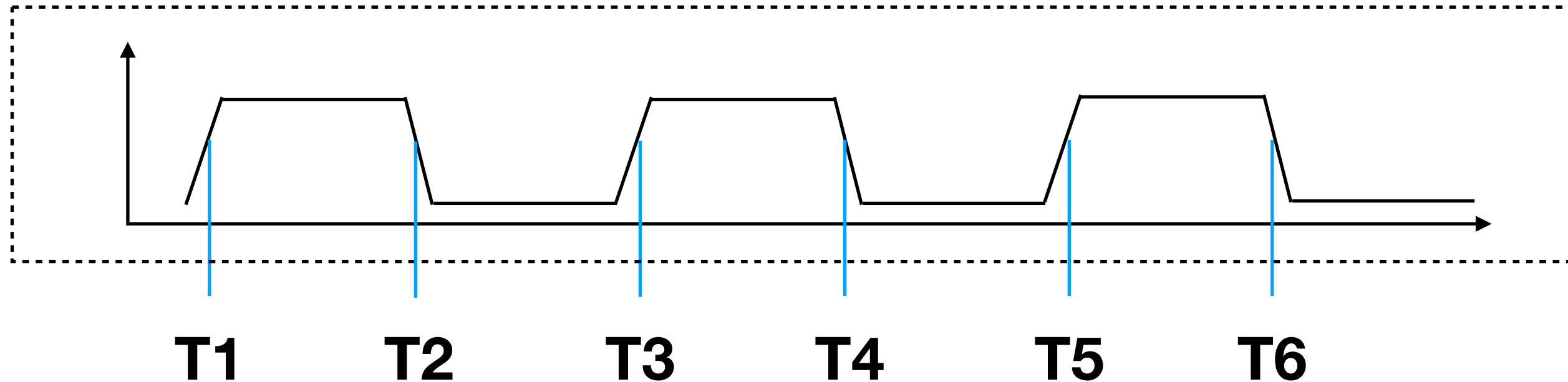


# 触发器：D触发器（上升沿触发）



触发器实现

# 触发器：D触发器（时序状态）



时间	CLK	E	E1	Q	Q1
<T1	0	1	0	D	-
T1-T2	1	0	1	-	Q
T2-T3	0	1	0	D	-
T3-T4	1	0	1	-	Q

# CPU

**CPU是一种复杂的时序逻辑电路**