

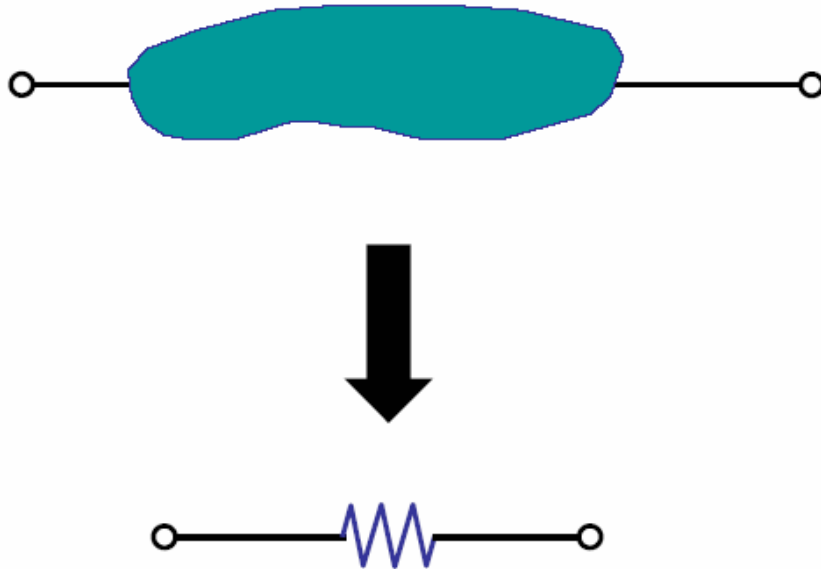
6.002

电路与
电子学

数字抽象

复习

离散物质同样遵守集总元件规律。



- 分析工具：KVL/KCL, 节点法，叠加定理，戴维宁和诺顿定理（叠加定理，戴维宁和诺顿定理仅适用线性电路）

今天

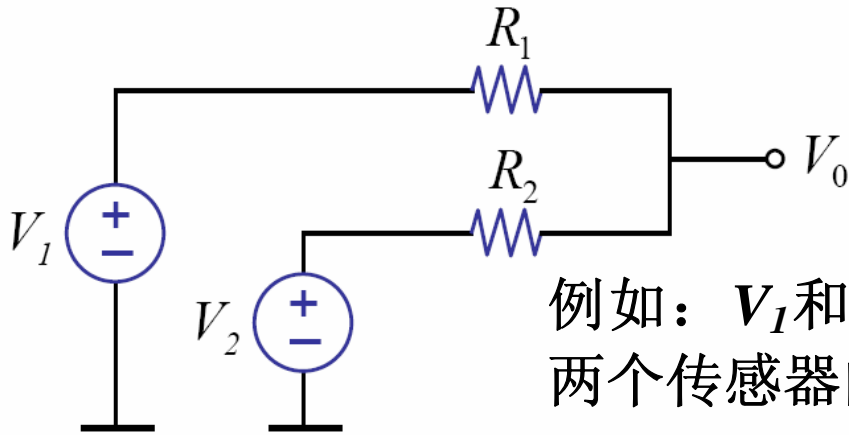
离散数值 \Longrightarrow 数字提取（抽象）

有意思的是, 我们很快就会发现前三讲中学到的解题工具足够用来分析简单的数字电路

阅读: **Agawal&Wang** 第五章

但是首先我们为什么引入数字信号？
过去……

模拟信号处理



例如： V_1 和 V_2 可以代表
两个传感器的输出

利用叠加定理：

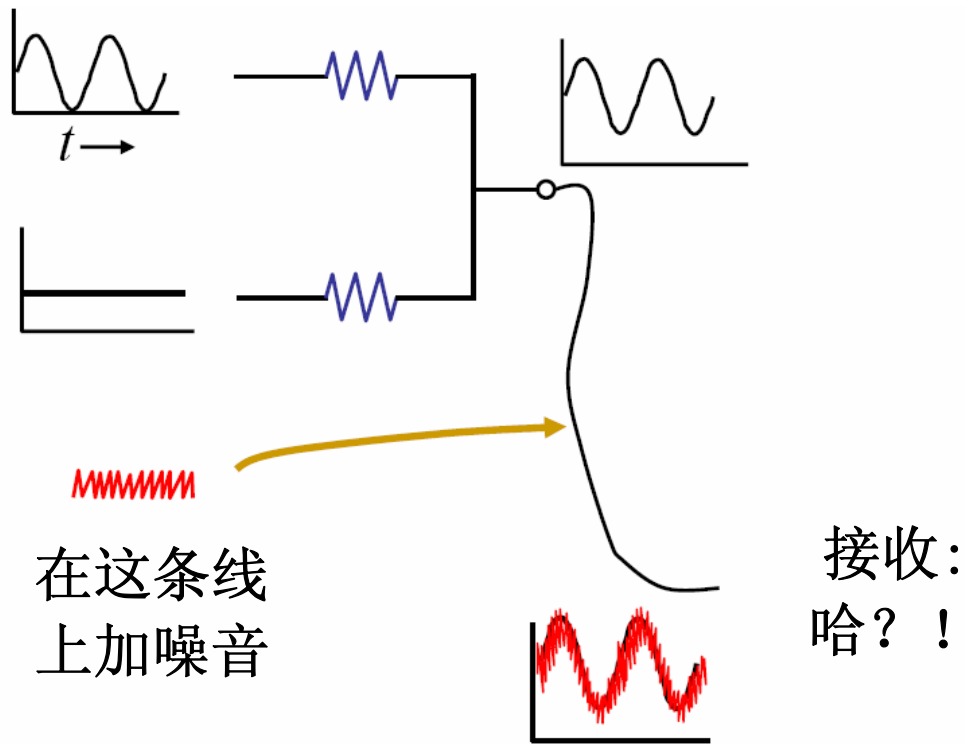
$$V_0 = \frac{R_2}{R_1 + R_2} V_1 + \frac{R_1}{R_1 + R_2} V_2$$

如果 $R_1 = R_2$

$$V_0 = \frac{V_1 + V_2}{2}$$

上述是一个“加法”电路。

噪声问题：



...噪声信号妨碍了我们对数值的微小差别加以区分的) 能力。例如3.1V和3.2V之间。

数值离散化

取值有两种可能

高

5 V

低

0 V

真

1

假

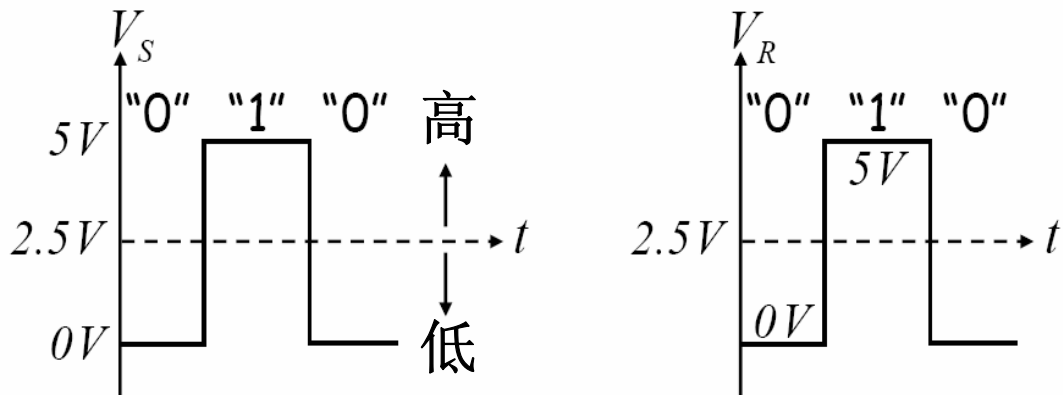
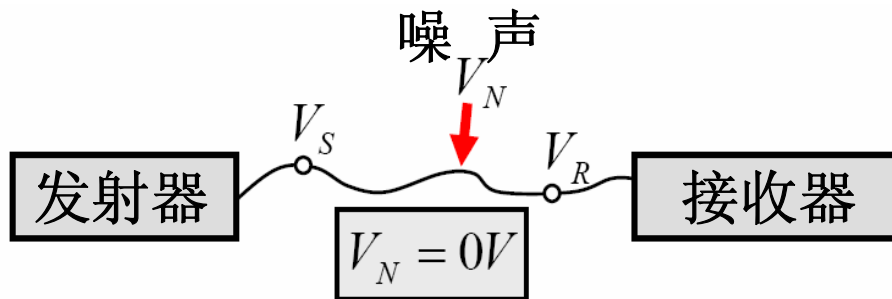
0

…就像两个阿拉伯数字 0 和 1

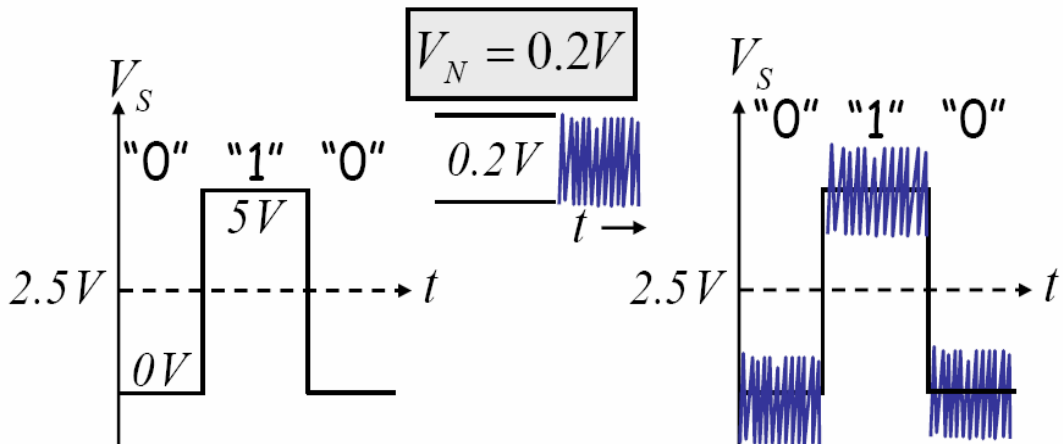
为什么这种离散化很有用?

(记住，大于 1 的数可以用多位二进制数表示和编码，就像用多位十进制数来表示大于 9 的数一样。例如，二进制数 **101** 等于十进制数 **5**)

数字系统



有噪声信号



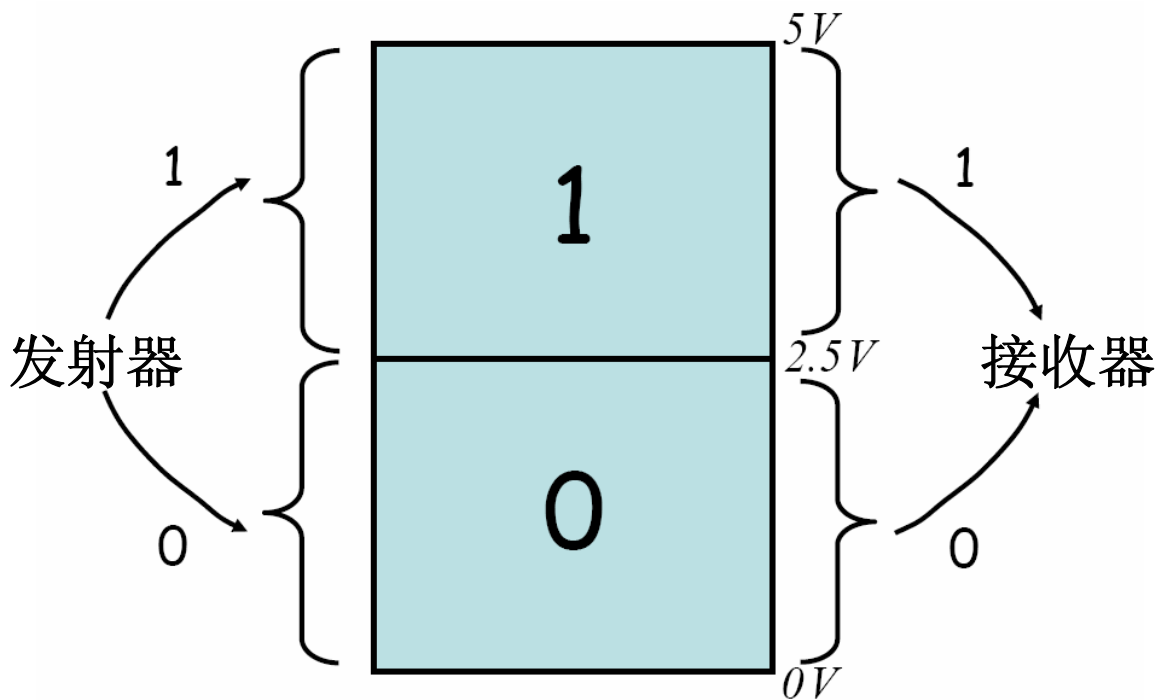
数字系统

好的抗噪声性能
大的“噪声裕量”

对于 1: “噪声裕量” $5V$ to $2.5V = 2.5V$

对于 1: “噪声裕量” $0V$ to $2.5V = 2.5V$

电压域值和逻辑值

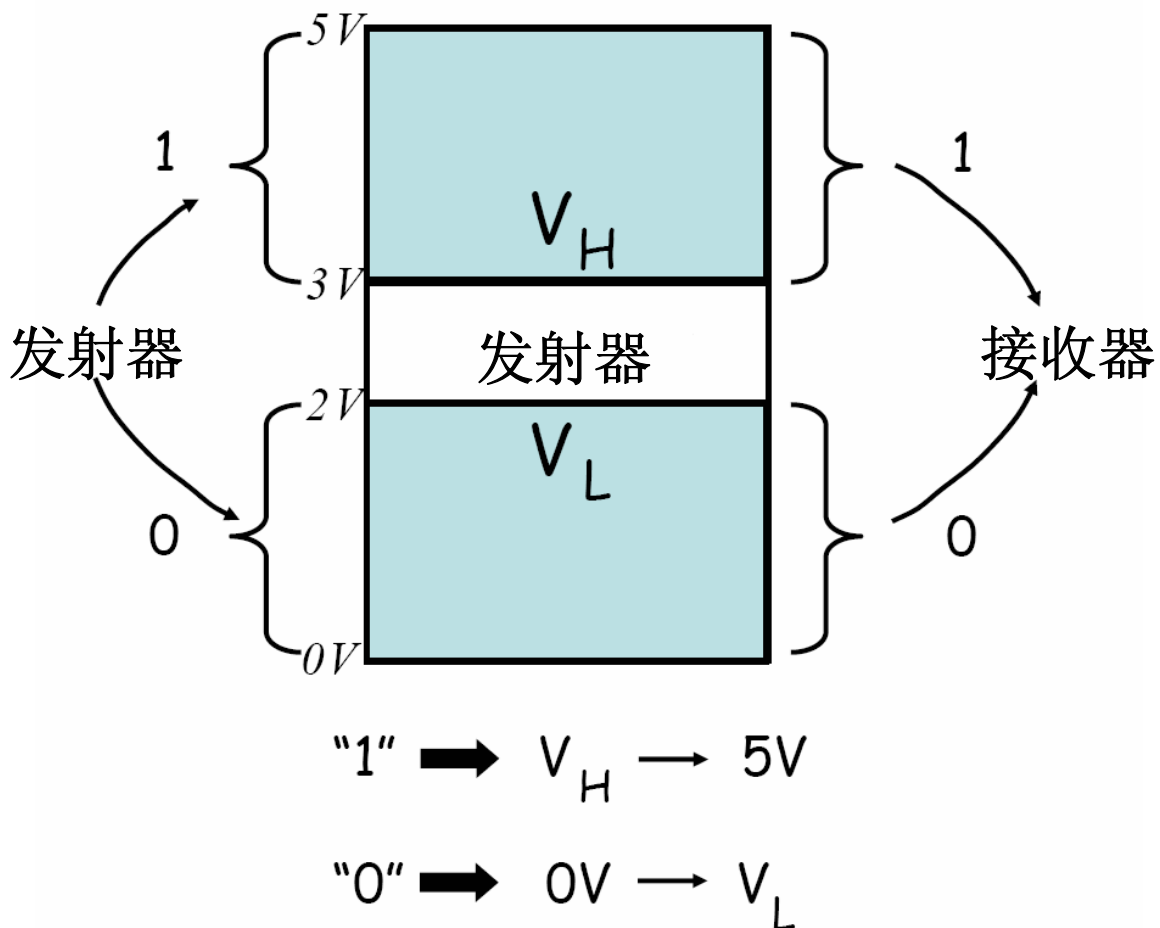


但是，但是，但是……

2.5V 会是什么情况呢？

嗯……这是禁止的情况

例如：

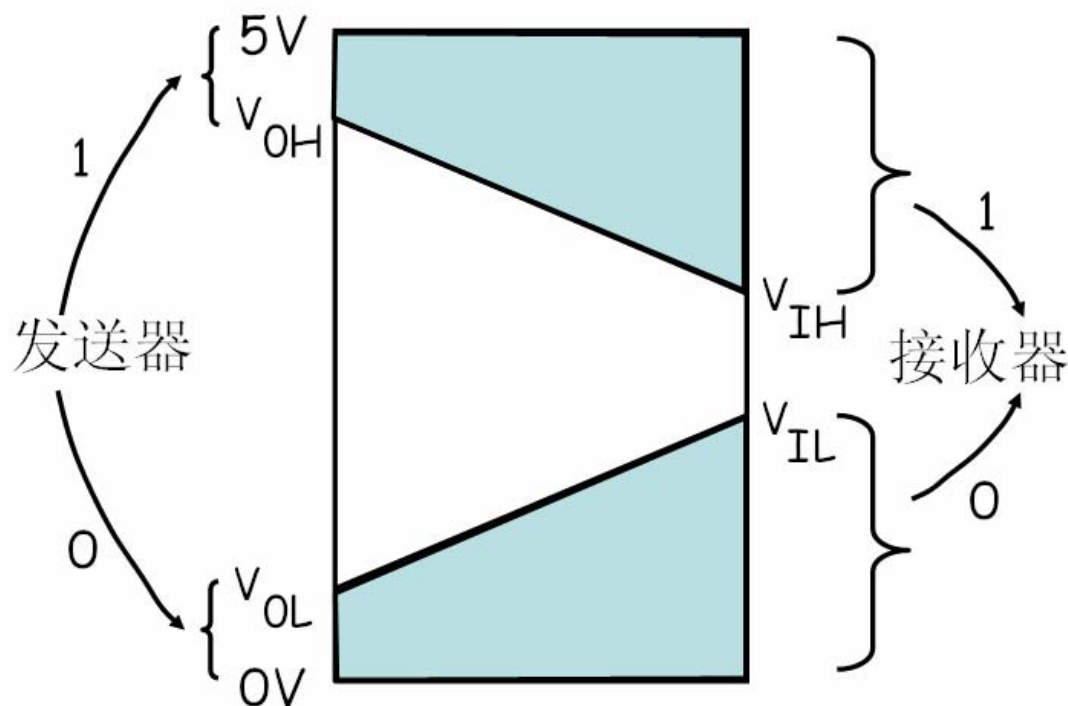


但是，但是，但是……

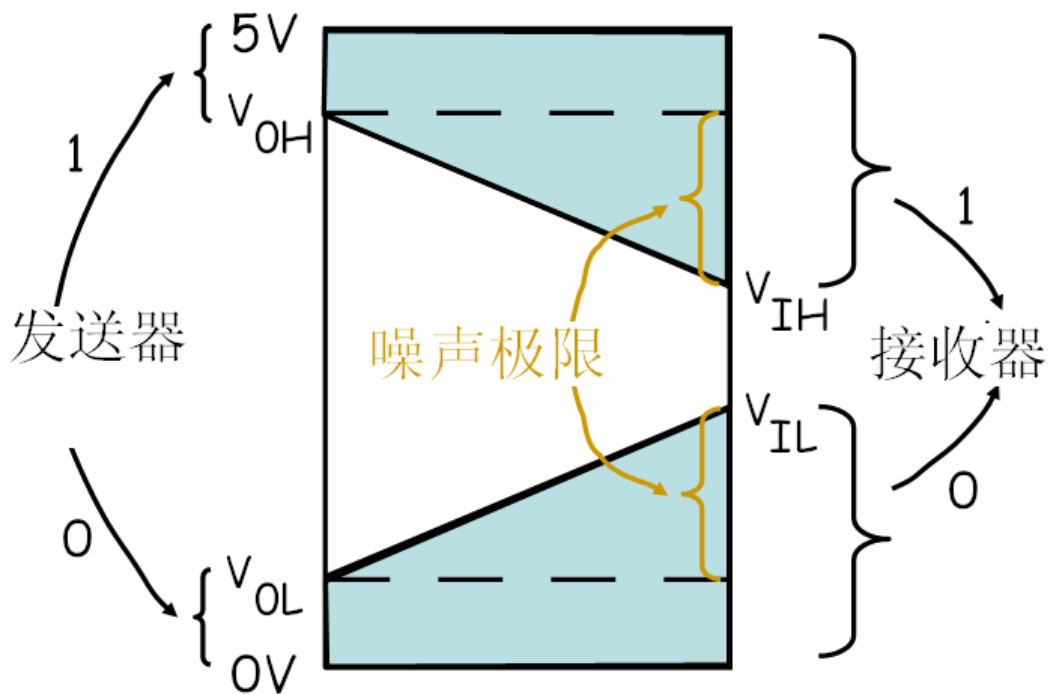
哪里是噪声裕量的范围呢？

如果发送器发送 1 呢： V_{IH} ?

保持发送器发送的信号严格在域值范围内！

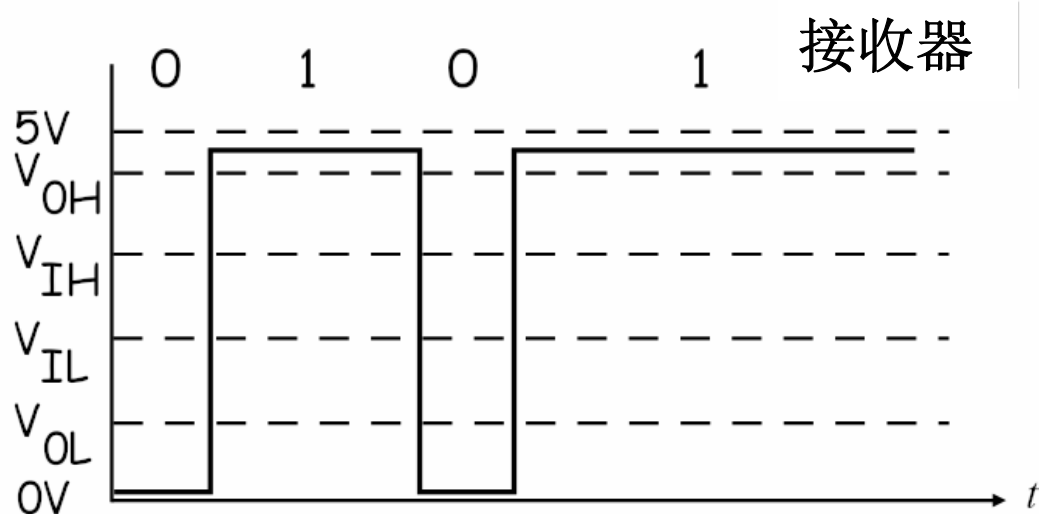
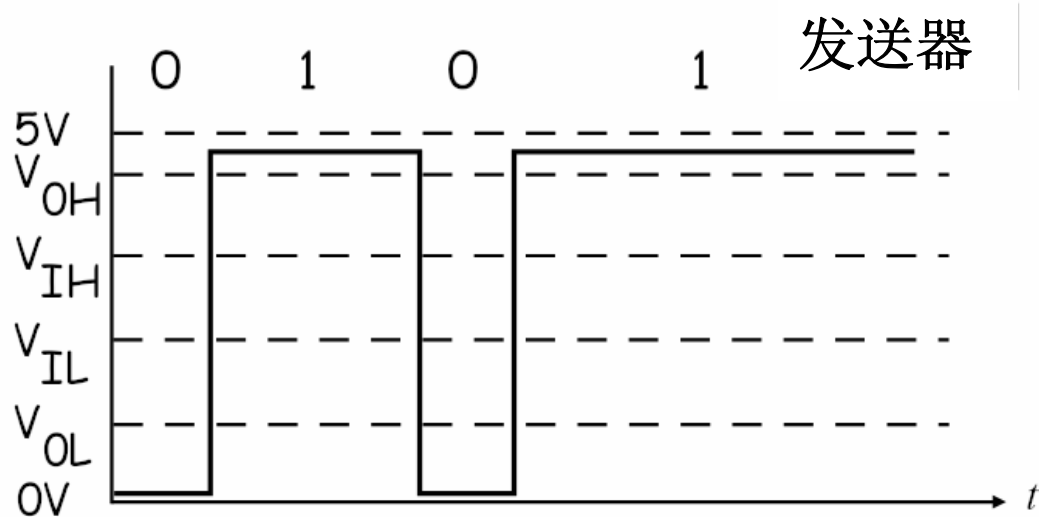


但是，但是，但是
那里是噪声容限的极限呢？
如果发送器发送 1 呢？
保证发送器发送的信号严格满足域值标准！



“1” 噪声容限： $V_{IH} - V_{OH}$

“0” 噪声容限： $V_{IL} - V_{OL}$



数字系统遵守静态规律：如果 输入信号在输入信号的域值范围内则系统可以保证输出信号大小在输出信号域值范围

数字信号处理

回想一下，我们仅有两个值

1,0 \Rightarrow 自然的绘制逻辑图：T, F
 \Rightarrow 也可以表现数字

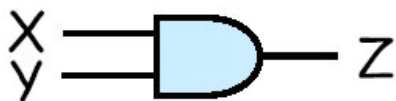
处理数字信号

布尔逻辑

⇒ 如果 **X** 是真，同时 **Y** 也是真
那么 **Z** 是真，否则 **Z** 是假。

⇒ $Z = X$ 与 Y
 $Z = X \cdot Y$
布尔等式

X, Y, Z
是数字信号
"0", "1"

⇒  与门

⇒ 真值表表示法：

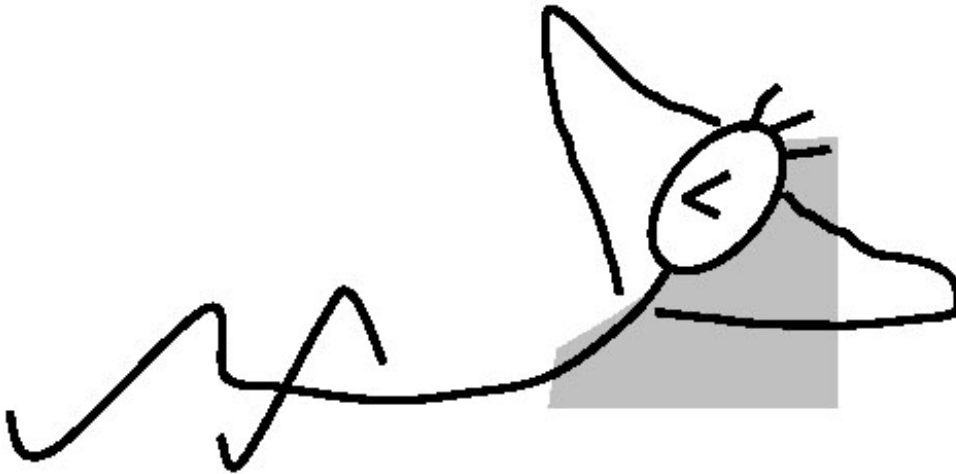
X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

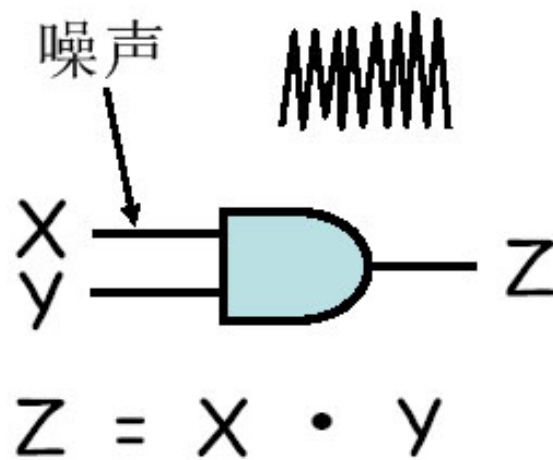
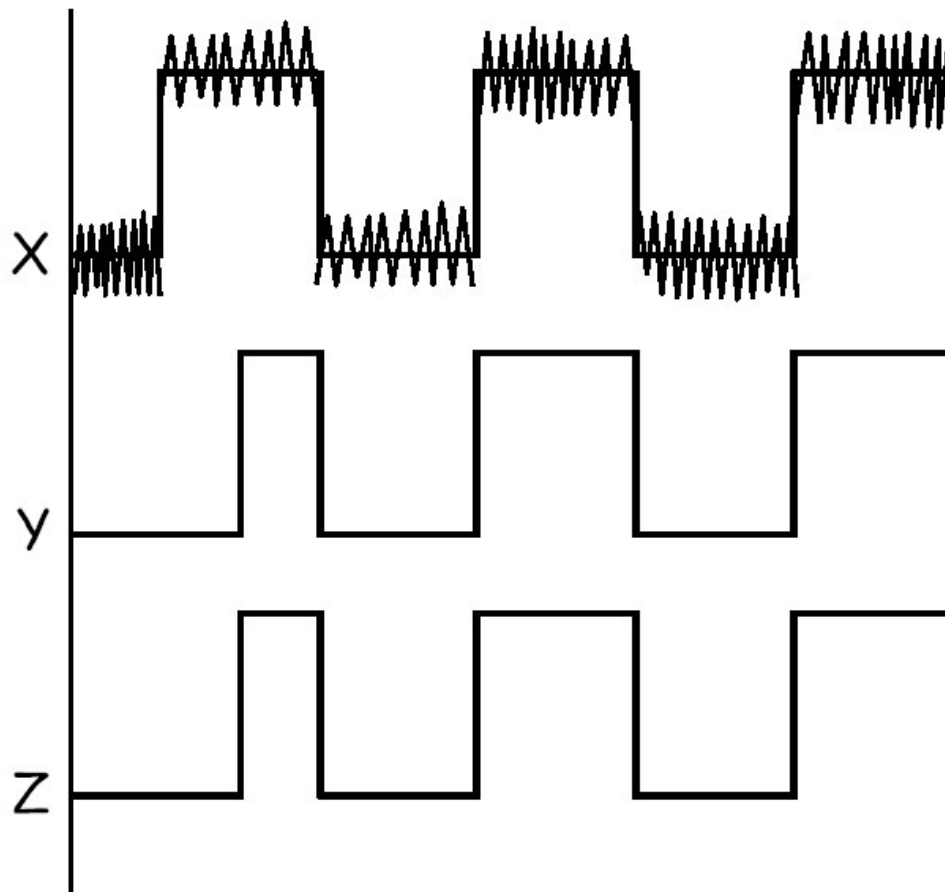
列举所有的输入组合

组合门 提取

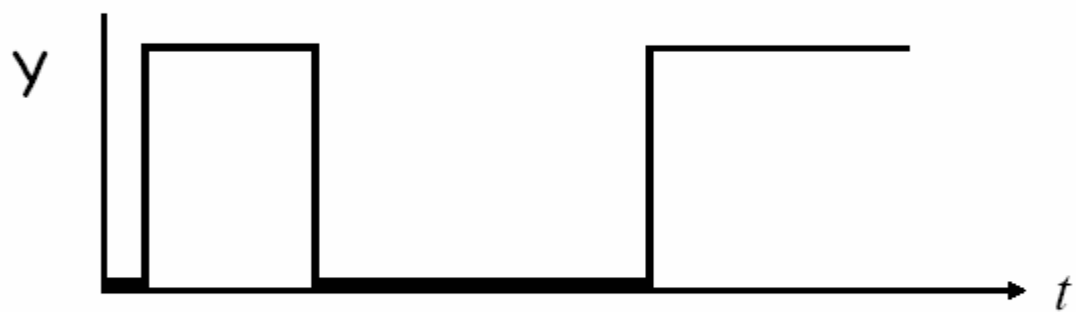
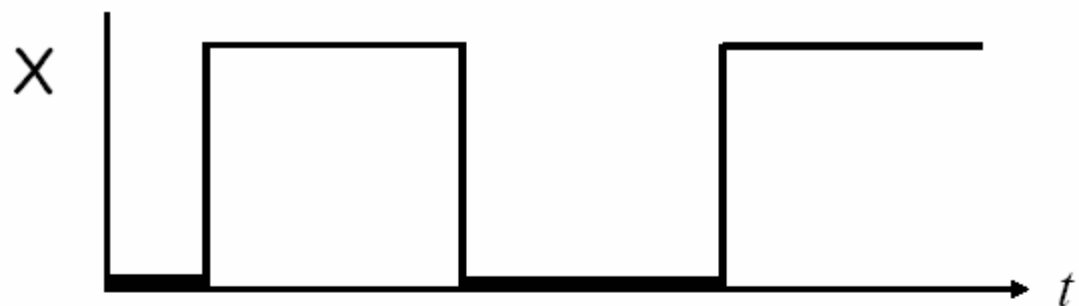
- 静态规则
- 输出仅是输入的函数.

数字逻辑设计不用关心
内部门电路的结构。





复习举例



$$Z = X \cdot y$$

复习...

另一个门电路的例子

如果 (A 是真) 或 (B 是真)

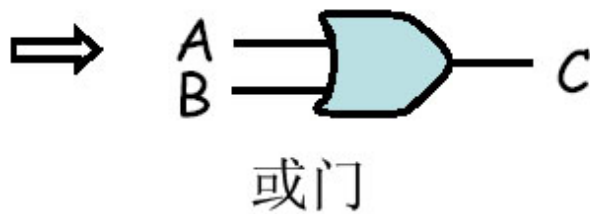
那么 C 是真

否则 C 是假

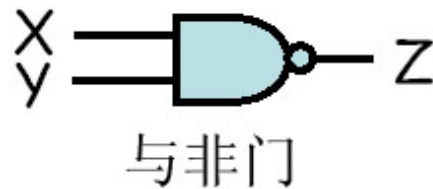
$$\Rightarrow C = A + B$$

布尔等式

或



更多门



$$Z = \overline{X \cdot Y}$$

布尔恒等式

$$X \cdot 1 = X$$

$$X \cdot 0 = 0$$

$$X + 1 = 1$$

$$X + 0 = X$$

$$\overline{\overline{1}} = 1$$

$$\overline{\overline{0}} = 0$$

$$AB + AC = A \cdot (B + C)$$

数字电路

实现 : 输出 = $A + \overline{B \cdot C}$

