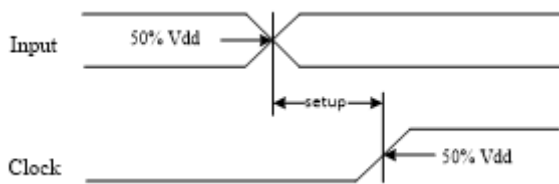
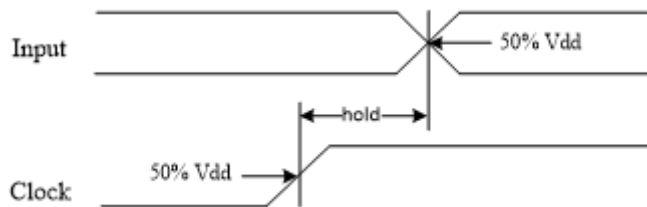


由D触发器结构图可知，数据D要经过TG1才能到达前级反馈回路，延时大小取决于TG1的门延时，此外前级反馈回路建立稳定的电路状态也需要一定时间。时钟上沿会关闭TG1，这就要求时钟上沿到来之前，数据D已经到达前级反馈回路，并且其中的反相器已经完成负载电容的充放电，建立了稳定的电平。

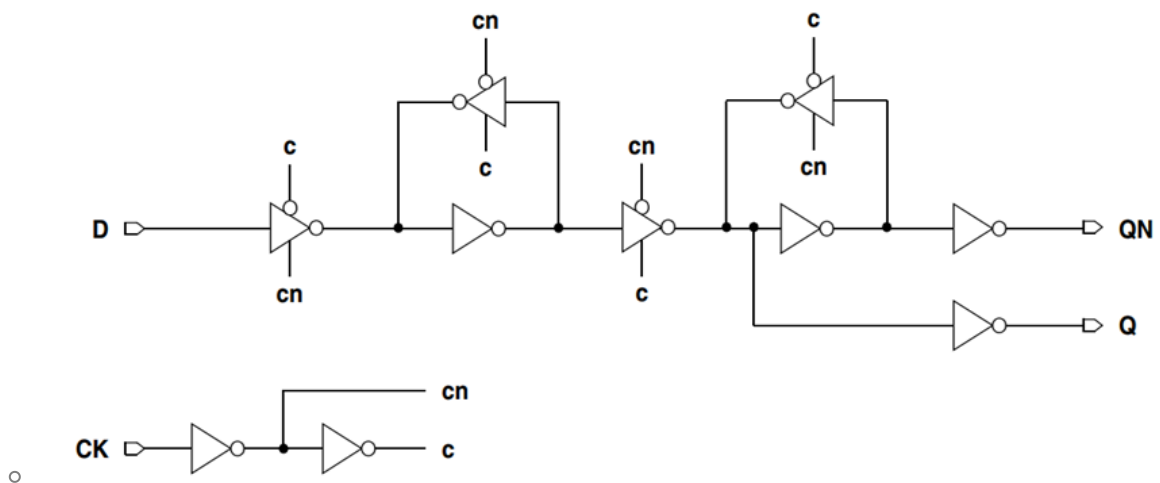


**保持时间 (Hold Time)：** 数据信号在时钟沿到达之后必须保持稳定的最小时间。如果数据过早变化，可能会导致保持时间违例。

时钟信号传播到TG1控制端是有延时的，TG1从开启状态转为关闭状态也需要时间，此外，前级反馈回路中的TG2开启并建立稳定的电路状态也是需要时间的，在这些动作完成之前，不希望数据D发生变化，否则会影响数据D的锁存。保持时间就是为了保证TG1完全关闭了，同时前级反馈回路已经稳定后，数据D才允许变化。



#### • D触发器电路结构

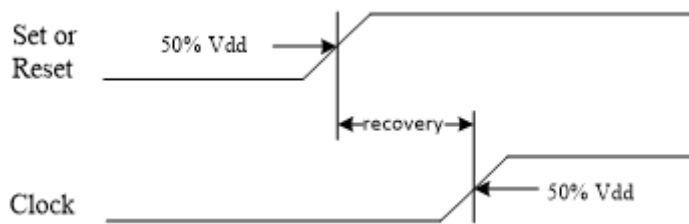


### 三、结合电路原理图，解释D触发器的恢复时间 (recovery time) 和释放时间 (removal time) 。

**恢复时间 (recovery time)**

恢复时间是指在时钟信号 (CK) 的有效边沿到来之前，数据输入信号 (D) 需要保持稳定的最短时间。换句话说，D信号必须在时钟的有效边沿到来前  $t_{\text{retrc}}$  时间内稳定，确保触发器能够正确采样数据。

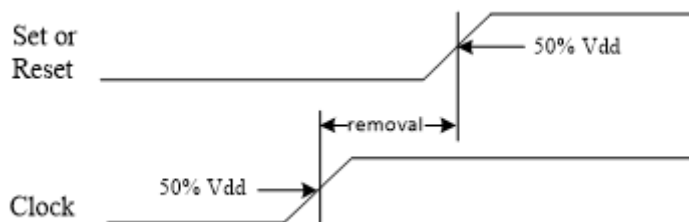
- 在电路中，D信号通过传输门 TG1 和 TG2 传递到触发器的主锁存部分。
- 当时钟信号 CK 为低电平 (CKN 为高电平) 时，TG1 导通，D 数据被锁存到 TG2 的输入节点。
- 当时钟信号 CK 上升沿到来时，TG1 关闭，TG2 导通，主锁存的数据被传递到从锁存器。
- 如果 D 信号在 CK 上升沿到来之前没有稳定 (例如发生跳变)，主锁存器可能采样到错误的数据，导致触发器输出错误。



释放时间 (removal time)

释放时间是指在时钟信号 (CK) 的有效边沿到来之后, 异步复位或置位信号 (如复位信号 Reset 或置位信号 Set) 需要保持无效状态的最短时间。换句话说, 异步控制信号必须在时钟有效边沿之后保持稳定 trem时间, 确保触发器的状态正确更新

- 在触发器中, 异步复位或置位信号通常直接控制输出节点 Q 和 QN。
- 当异步复位信号 (例如 Reset) 被释放时, 触发器恢复到正常工作状态。
- 如果复位信号在时钟有效边沿之后立即变化, 触发器可能无法正确更新输出状态, 导致逻辑错误。



## 四、同步复位和异步复位哪个好? 异步复位和异步置位哪个好? 为什么?

同步复位 (**Synchronous Reset**) :

- 复位信号在时钟信号的边沿 (通常是上升沿或下降沿) 时起作用。
- 复位信号必须与时钟信号同步, 只有在时钟边沿到来时, 电路才会响应复位信号。
- 复位逻辑通常被集成到时序逻辑中。

异步复位 (**Asynchronous Reset**) :

- 复位信号不依赖时钟信号, 可以在任意时间直接起作用。
- 一旦复位信号有效, 电路会立即进入复位状态, 而不需要等待时钟边沿

### 同步复位VS异步复位

#### 1. 同步复位

##### 1. 优点

1. 确保电路同步
2. 复位只发生在有效时钟沿, 可以过滤毛刺

##### 2. 缺点

1. 复位信号的有效时长必须大于时钟周期, 才能真正被系统识别并完成复位。同时还要考虑如: 时钟偏移、组合逻辑路径延时、复位延时等因素
2. 需要消耗额外的逻辑资源

#### 2. 异步复位

## 1. 优点

1. 异步复位信号识别方便，而且可以很方便的使用全局复位
2. 厂商目标库内的触发器都有异步复位端口，可以节约逻辑资源

## 2. 缺点

1. 复位信号容易受到毛刺信号影响
2. 复位结束时刻恰在亚稳态窗口内时，无法决定现在的复位状态是1还是0，会导致亚稳态

3. 🔴 同步复位更适合高频电路，时序可控，可靠性更高；异步复位适用于需要快速响应的场景，但需处理信号抖动和亚稳态问题；大多数现代设计倾向于使用同步复位

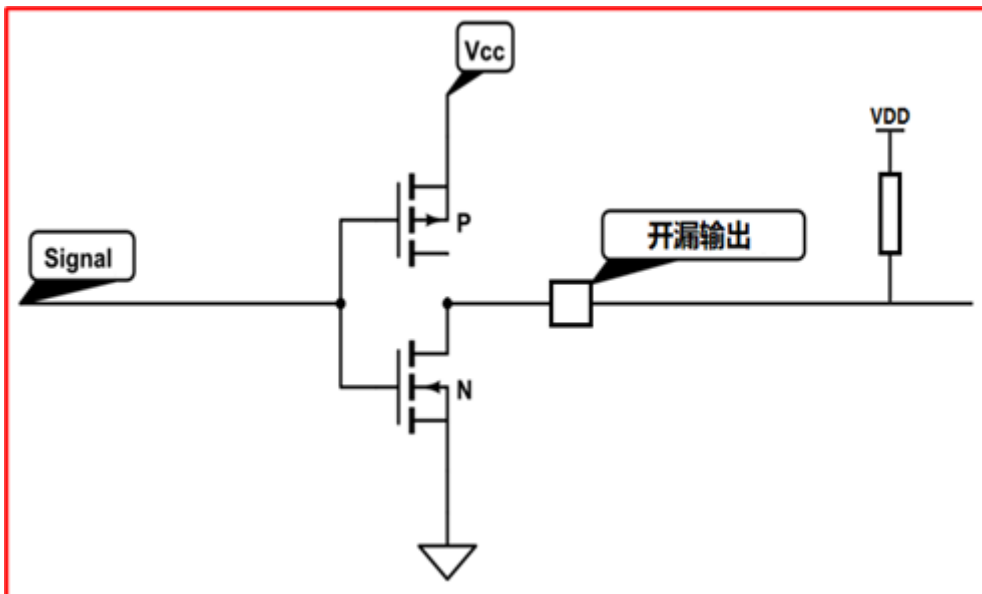
## 异步置位 (Asynchronous Set) :

- 异步置位 (Asynchronous Set) 是一种不依赖时钟信号的置位操作。当异步置位信号有效时，电路的状态会立即被置为逻辑1
- 置位信号不依赖时钟信号，可以在任意时间直接起作用。
- 一旦置位信号有效，电路会立即进入置位状态，而不需要等待时钟边沿。

## 异步复位VS异步置位

- 异步复位通常比异步置位更常见，因为大多数电路的初始化状态是复位为0，复位操作在系统启动和故障恢复中更为重要。
- 异步置位适用于需要快速将状态置为1的特殊场景，但在实际应用中使用频率较低。

## 五、何为开漏输出？有什么用？



开漏输出是一种特殊的输出结构，核心特点是使用一个**N沟道 MOSFET**（图中的N管）作为开关，漏极 (Drain) 直接连接到输出端，而没有内部的上拉电阻。输出端需要通过外部上拉电阻连接到电源 **VDD** 才能实现逻辑高电平。

## 工作原理

- 输出低电平（逻辑0）

- 当信号 SignalSignal 控制 P管导通时，N管也导通，输出端直接与地相连
- 此时，输出电压为逻辑低电平（0V）
- 输出高电平（逻辑1）
  - 当信号 SignalSignal 控制 P管截止时，N管也截止，输出端悬空（高阻态）
  - 此时，输出电压由外部上拉电阻拉到 VDD逻辑高电平）

作用：开漏输出是一种灵活的输出结构

1. 实现多设备信号共享（如 I<sup>2</sup>C 总线）。
2. 实现不同电压域之间的电平转换：通过外部上拉电阻连接到不同电压的电源，实现电平转换。
3. 驱动大电流或高压负载：开漏输出可以通过外部电路（如继电器、LED）驱动大电流或高压负载。负载电流由外部电源提供
4. 实现中断信号等的共享逻辑：多个设备的中断信号可以通过开漏输出连接到一个共享的中断引脚，当任意一个设备触发中断时，中断信号线被拉低到低电平，通知主控设备处理中断

截止时间 2024年11月30日 23:59