
CS 61C逻辑、SDS、FSM

2020年秋季讨论6： 2020年10月5日

1 预检查

本节旨在对您进行概念性检查，以确定您是否从概念上理解此主题以及是否对此主题有任何误解。 请回答以下问题的对/错，并附上解释：

- 1.1 简化布尔逻辑表达式对硬件实现的性能没有影响。

错。实现相同逻辑的不同门布置可能具有不同的传播延迟，这会影响允许的时钟速度。

- 1.2 门越少，电路越快（假设它们都具有相同的延迟）。

错。传播延迟随着电路的深度而增加，因此具有更多门的宽电路可以比仅几个按顺序排列的门具有更小的延迟。

- 1.3 允许时钟到q的建立时间大于一个时钟周期。

错。如果触发器彼此连接，这会导致不稳定性。

2 布尔逻辑

在数字电子学中，根据输入得到某些输出（如真值表所示）通常很重要。真值表直接映射到布尔表达式，布尔表达式直接映射到逻辑门。然而，为了最小化实现电路所需的逻辑门的数量，简化长布尔表达式通常是有用的。

我们可以使用布尔代数的九个关键定律来简化表达式：

姓名	与表	OR表
交换	$AB = BA$	$A + B = B + A$
关联	$AB(C) = A(BC)$	$A + (B + C) = (A + B) + C$
鉴别	$1A = A$	$0 + A = A$
空值	$0A = 0$	$1 + A = 1$
吸收	$A(A + B) = A$	$A + AB = A$
分销	$(A + B)(A + C) = A + BC$	$A(B + C) = AB + AC$
幂等	$A(A) = A$	$A + A = A$
反向	$\overline{A(A)} = 0$	$\overline{A + A} = 1$
德摩根	$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}(\overline{B})$

2.1 用德摩根定律的多次迭代证明恒等式 $\overline{A} + \overline{AB} = \overline{A} + \overline{B}$ 。

$$\begin{aligned}
 \overline{A} + \overline{AB} &= \overline{\overline{A} + AB} \\
 &= \overline{A(\overline{A} + B)} \\
 &= \overline{A\overline{A} + AB} \\
 &= \overline{AB} \\
 &= \overline{A} + \overline{B}
 \end{aligned}$$

2.2 简化以下布尔表达式：

(a) $(A+B)(\overline{A+B})C$

$$\begin{aligned}
 (A+B)(\overline{A+B})C &= (A+BB)C \\
 &= \text{交流电}
 \end{aligned}$$

(b) $A\sim B\sim C\sim + A\sim BC\sim + ABC\sim + AB\sim C\sim + ABC + AB\sim C$

$$\begin{aligned}
 &A\overline{C}\overline{(B+B)} + A\overline{C}\overline{(B+B)} + AC(B+B) \quad A\sim C\sim + AC\sim + AC \\
 = &A\sim C\sim + AC\sim + AC \quad A\sim C\sim + AC\sim + AC\sim + AC \\
 &= \\
 &= (\overline{A} + A)C + A(C + C) \\
 &= C + A
 \end{aligned}$$

或者，使用2.1中的标识：

$$\begin{aligned}
 &A\overline{C}\overline{(B+B)} + A\overline{C}\overline{(B+B)} + AC(B+B) \quad A\sim C\sim + AC\sim + A \\
 = &C\overline{C}\sim(A\sim + A) \\
 &= +AC\overline{C}\sim + AC \\
 &= C\sim + A \\
 &=
 \end{aligned}$$

(c) $\overline{A(B\sim C\sim + BC)}$

$$\begin{aligned}
 \overline{A(B\bar{C}+BC)} &= \overline{A\sim+B\sim C\sim+BC} \\
 &= \overline{A\sim+B\sim C\sim} BC \\
 &= \overline{A\sim} (B+C) (\overline{B\sim+C\sim}) \\
 &= A\sim+BC\sim+B\sim C \\
 &=
 \end{aligned}$$

$$(d) \quad \overline{A} (A+B) + (B+AA) (\overline{A}+B)$$

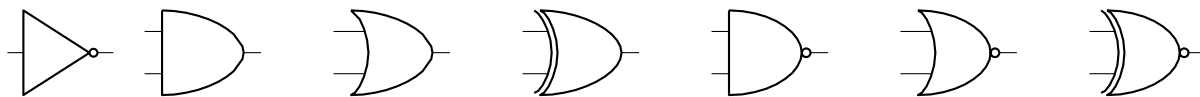
$$\begin{aligned}
 \overline{A} (A+B) + (B+AA) (\overline{A}+B) &= \overline{AA} + \overline{AB} + (B+AA) (\overline{A}+B) \\
 &= \overline{AB} + (B+A) (\overline{A}+B) \\
 &= \overline{AB} + (BA+AA+BB+\overline{AB}) \\
 &= \overline{AB} + (BA+A+AB) \\
 &= \overline{AB} + A \\
 &= A+B
 \end{aligned}$$

或者，

$$\begin{aligned}
 \overline{A} (A+B) + (B+AA) (\overline{A}+B) &= \overline{A} (A+B) + (A+B) \overline{A} (A+B) \\
 &= (A+B) (\overline{A}+A+B) \\
 &= A+B
 \end{aligned}$$

3 逻辑门

3.1 标记以下逻辑门：



NOT, AND, OR, XOR, NAND, NOR, XNOR

3.2 将以下内容转换为输入信号A和B的布尔表达式：

(a) 与非

$$A\sim B\sim + A\sim B + AB\sim$$

或者，使用和积规则： $A\sim + B\sim$

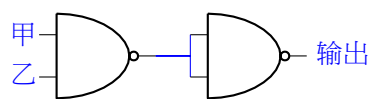
(b) 异或

$$A\sim B + AB\sim$$

(c) 异或

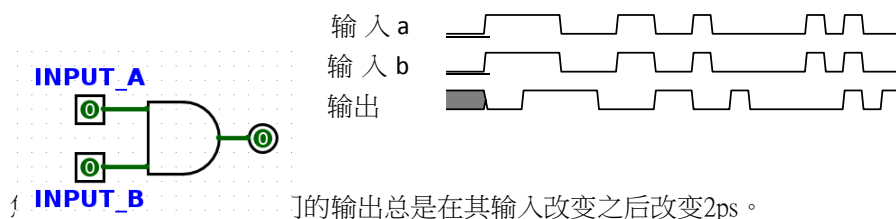
$$A\sim B\sim + AB$$

3.3 仅使用“与非”门创建“与”门。



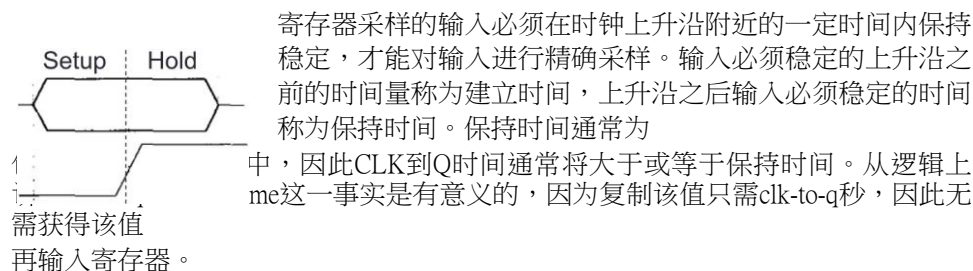
4 州简介

有两种基本类型的电路：组合逻辑电路和状态元件。**组合逻辑电路简单地基于它们的输入在与它们相关的任何传播延迟之后改变**。例如，如果与门（如下图所示）具有2ps的相关传播延迟，则其输出将基于其输入而改变，如下所示：

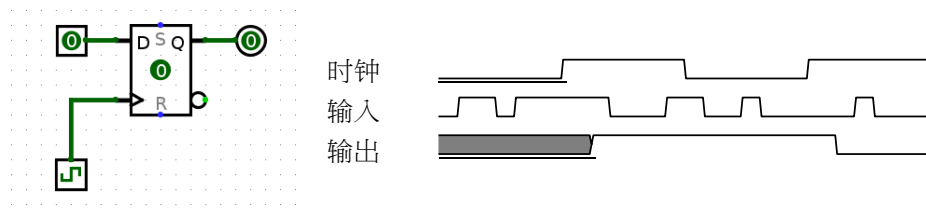


另一方面，即使在输入改变之后，状态元素也可以记住它们的输入。状态元件基于时钟信号改变值。例如，上升沿触发寄存器在时钟的上升沿对其输入进行采样（当时钟信号从0变为1时）。

与逻辑门一样，寄存器在其输出反映被采样的输入之前也存在与其相关的延迟。这称为clk-to-q延迟。（“Q”通常表示输出）。这是时钟信号上升沿与寄存器输出反映输入变化之间的时间。

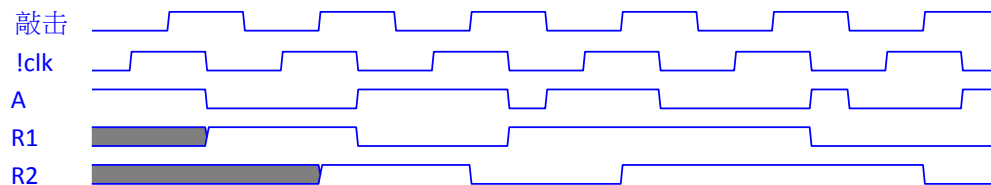
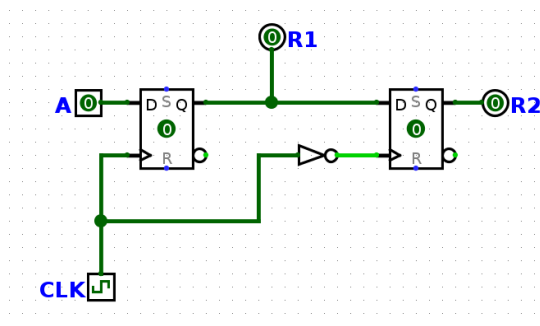
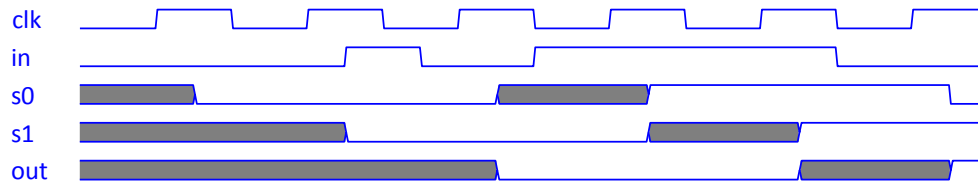
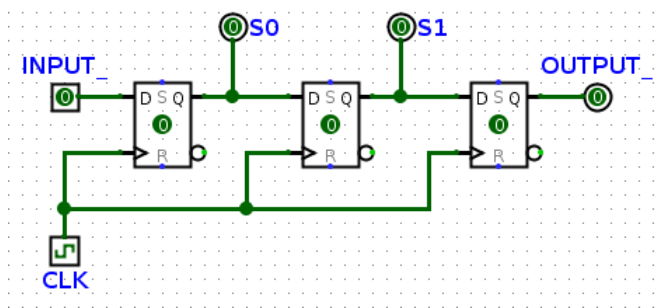


对于以下寄存器电路，假设设置值为2.5ps,保持时间为1.5ps,CLK-to-Q时间为1.5ps。时钟信号的周期为13ps。

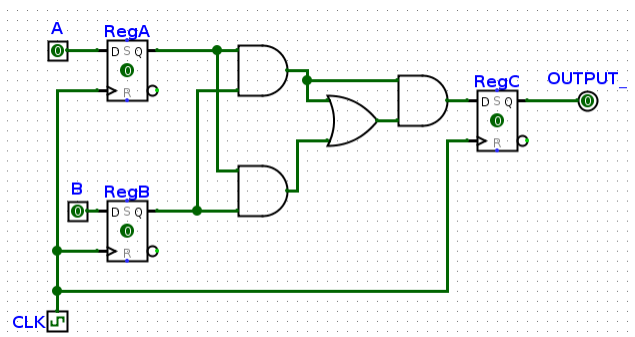


您会注意到，上图中的输出值不会在时钟上升沿之后立即改变。时钟周期时间必须足够小，使得寄存器的输入在保持时间内不变，并且必须足够大，以考虑clk-to-q时间、建立时间和组合逻辑延迟。

- 4.1 对于以下2个电路，填写时序图。时钟周期（上升沿到上升沿）为8ps。对于每个寄存器，clk-to-q延迟为2ps,建立时间为4ps,保持时间为2ps。“非”门具有2ps传播延迟



- 4.2 在下面的电路中，RegA和RegB的建立、保持和clk-to-q时间为4ns,所有逻辑门的延迟为5ns,RegC的建立时间为6ns。RegC的最长允许放置时间是多少?该电路的最小可接受时钟周期时间是多少?它对应的时钟频率是多少?



RegC的最大允许保持时间是RegC输入改变所需的时间，因此（A或B的clk-to-q）+ 最短CL时间 = $4 + (5 + 5) = 14 \text{ ns}$ 。

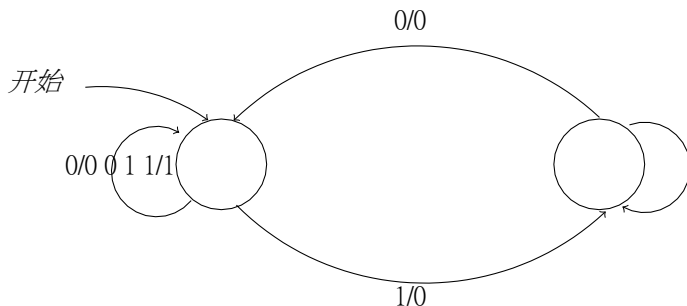
可接受的最小时钟周期时间为：clk-to-q + 最长CL时间 + 建立时间 = $4 + (5 + 5 + 5) + 6 = 25 \text{ ns}$ 。

25 ns对应的时钟频率为 $(1 / (25 \times 10^{-9})) \text{ s}^{-1} = 40 \text{ MHz}$

5个有限状态机

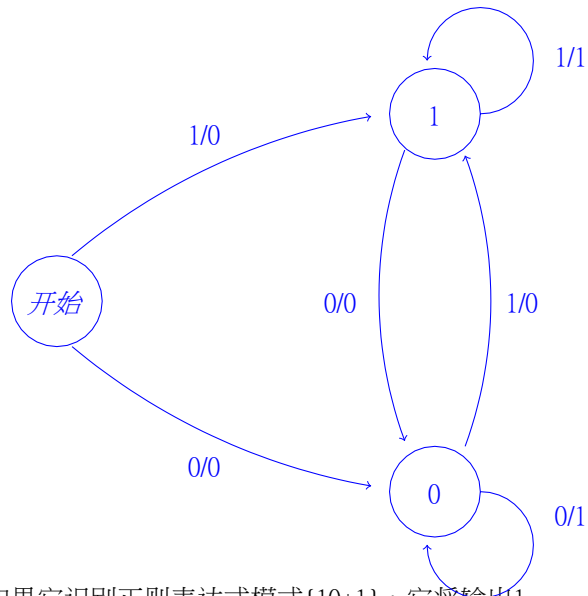
自动机是接收输入并使用各种状态产生输出的机器。有限状态机是一种简单的自动机，其中下一个状态和输出仅取决于当前状态和输入。每个状态都由一个圆圈表示，每个适当的有限状态机都有一个起始状态，用标签“Start”或指向它的单个箭头表示。状态之间的每个转换都标记为[input]/[output]。

- 5.1 以下 FSM 在位串中检测到什么模式？它会为输入位串“011001001110”输出什么？



如果检测到模式“11”，则FSM输出1。FSM将输出“001000000110”

- 5.2 每当有两个重复位作为最近位时，填写以下FSM以输出1,否则输出0。您可能不需要所有状态。



5.3 写入一个FSM,如果它识别正则表达式模式{10+1},它将输出1。

