

EDA总复习

1 绪论

- 设计思路：自上而下，逐步分解
- EDA技术的构成要素
 - 实现载体：可编程逻辑器件(PLD)
 - 描述方式：HDL/图形/混合
 - 软件开发环境：Quartus
 - 硬件开发环境：DE2-115开发板
- EDA的设计流程
 - 设计准备
 - 设计输入：原理图&硬件描述语言
 - 综合 - 功能仿真
 - 适配 - 时序仿真
 - 编程下载 - 在线测试

2 数制与码制

- 数字电路所处理的各种数字信号是以数码形式给出的。
- 数制：表示数量的大小
 - 采用进位计数制构成多位数码
 - 多位数码中每一位的构成方法和从低位到高位进位的规则成为数制
- 码制：表示不同事物或事物的不同状态：
 - 不同事物的代号为代码
 - 编制代码的规则成为码制
- 补码
 - 简易转换
 - 正数的补码和它的原码相同
 - 负数的补码=数值位取反再+1
 - 通过补码，将减一个数用加上该数的补码来实现
- 格雷码
 - 特点
 - 每一位的状态变化都按一定的顺序循环： $k^{*}\{0\}, k^{*}\{11\}, k^{*}\{0\}$
 - 编码顺序依次变化，按表中顺序变化时，相邻代码只有一位改变状态
 - 例：3位格雷码
 - 000, 001, 011, 010
 - 110, 111, 101, 100

- ASCII码：七位二进制代码，共128个

3 逻辑代数基础

- 概述
 - 逻辑：事物的因果关系
 - 逻辑运算的数学基础：逻辑代数
 - 在二值逻辑中的变量取值：0/1
- 三种基本运算
 - 与：条件同时具备，结果发生
 - $Y = A \cdot B$
 - 或：条件之一具备，结果发生
 - $Y = A + B$
 - 非：条件不具备，结果发生
 - $Y = A'$

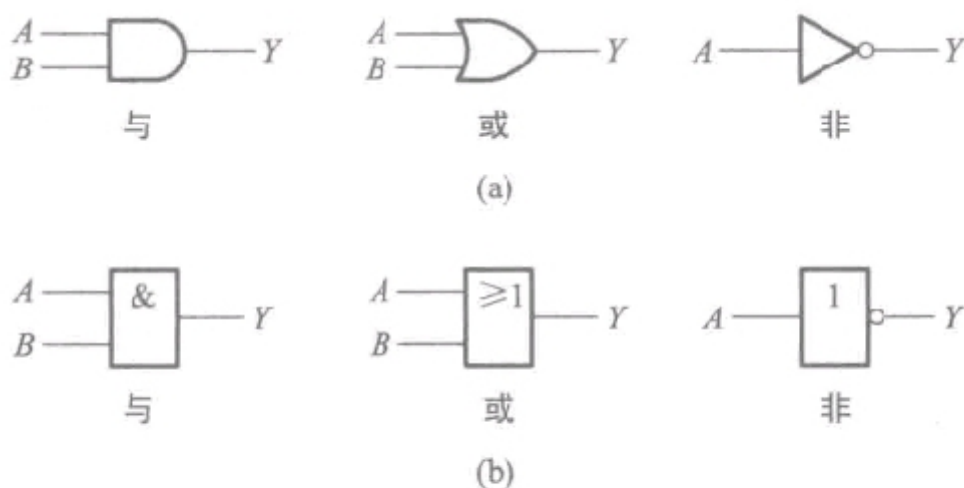


图 2.2.2 与、或、非的图形符号

(a) 特定外形符号 (b) 矩形轮廓符号

- 几种常用的复合逻辑运算
 - 与非： $Y = (A \cdot B)'$
 - 或非： $Y = (A + B)'$
 - 与或非： $Y = (A \cdot B + C \cdot D)'$
 - 异或： $Y = A \oplus B$ 或 $Y = A' \cdot B + A \cdot B'$
 - 同或： $Y = A \odot B$
- 重点基本公式/常用公式
 - $A + BC = (A + B)(A + C)$
 - $(AB)' = A' + B'$
 - $(A + B)' = A'B'$

逻辑函数的描述方法

- 真值表

- 遍历输入变量所有可能的取值组合
- 输出对应的取值
- 逻辑式：与或非的运算式
- 逻辑图：逻辑图形符号+逻辑电路的实现
- 卡诺图：使用卡诺图化简
 - 用卡诺图表示逻辑函数
 - 找出可合并的最小项
 - 化简后的乘积项相加
- 两种标准形式：最小项之和/最大项之积
 - 最小项
 - 编号举例

最小项	取值	对应	编号
	$A B C$	十进制数	
$A'B'C'$	0 0 0	0	m_0
$A'B'C$	0 0 1	1	m_1
$A'BC'$	0 1 0	2	m_2
$A'BC$	0 1 1	3	m_3
$AB'C'$	1 0 0	4	m_4
$AB'C$	1 0 1	5	m_5
ABC'	1 1 0	6	m_6
ABC	1 1 1	7	m_7

- 在输入任一变量任一取值下，有且仅有一个最小项的值为1

4 Verilog设计初步&语法

- 理解
 - 在一个Verilog模块中，所有过程块(always 语句块)、连续赋值语句 assign 以及实例引用都是并行的。
 - 它们同时执行，没有先后
 - 表示的是一种通过变量名相互连接的关系
- 标识符
 - 可是任意一组字母、数字以及符号 \$ 和 _ 的组合。
 - 区分大小写
 - 必须以字母或者下划线开头
- 数据类型

5 组合电路

6 存储电路：触发器

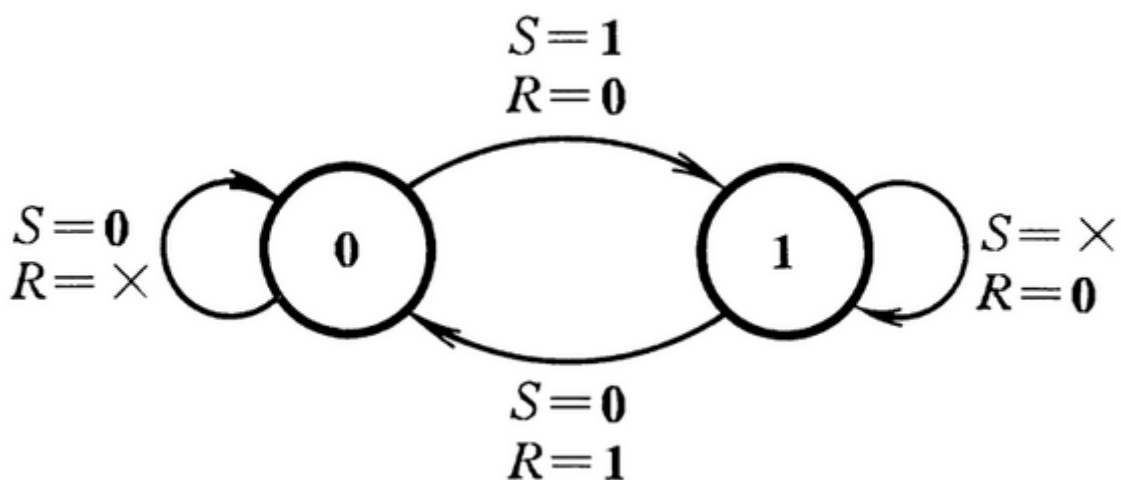
- 触发器的描述方法
 - 功能表：输入端为何值时，下一次输出为何种结果
 - 状态表：分现态的不同情况讨论，输入为何种组合时，输出为何种结果。
 - 状态图：状态转移情况。
 - 特性表：输入端、目前状态分别为不同取值时，下一状态的输出情况。
 - 特性方程：下一状态怎么决定；对输入的约束
 - 波形图：最直观

6.1 SR触发器

- 描述
 - 输入端：SR触发器有两个输入端：Set (S) 和Reset (R) 。
 - 状态：SR触发器有四种状态：
 - S=0, R=0 (保持)：输出端保持原状
 - S=0, R=1 (复位)：输出端被置为0
 - S=1, R=0 (设置)：输出端被置为1
 - S=1, R=1 (被禁止)：输出端的值无法确定
- 特性方程

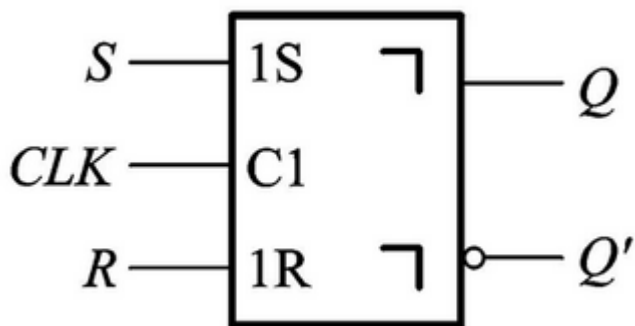
$$\begin{cases} Q^* = S + R'Q \\ SR = 0 \end{cases}$$

- 状态转换图



- 符号

- 输出端的拐角符号意味着Q的值在时钟下降沿被改变。

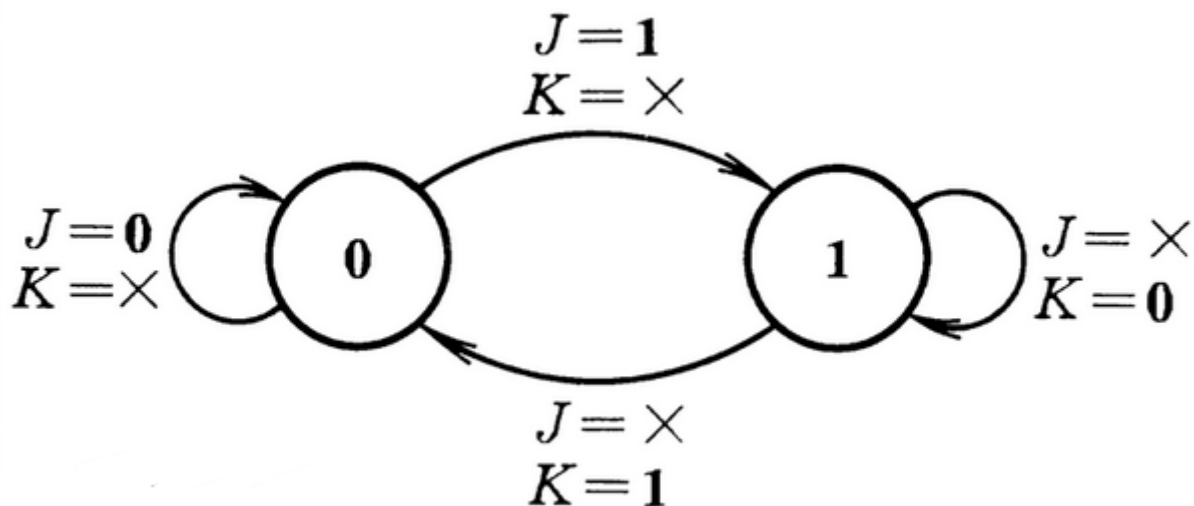


6.2 JK触发器

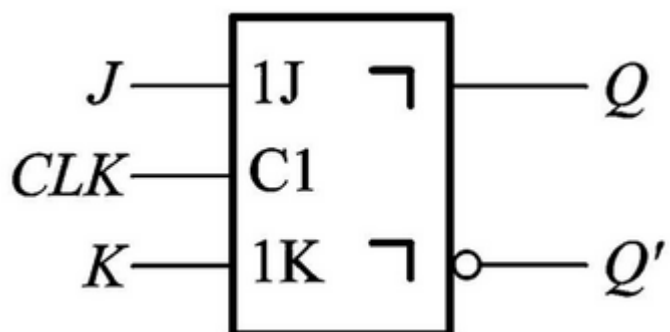
- 描述
 - 输入端：JK触发器有两个输入端：J（置位）和K（复位）。
 - 状态：JK触发器有四种状态：
 - J=0, K=0（保持）：输出端保持原状
 - J=0, K=1（复位）：输出端被置为0
 - J=1, K=0（置位）：输出端被置为1
 - J=1, K=1（切换）：输出端与上一状态相反
- 特性方程

$$Q^* = JQ' + K'Q$$

- 状态转换图



- 符号

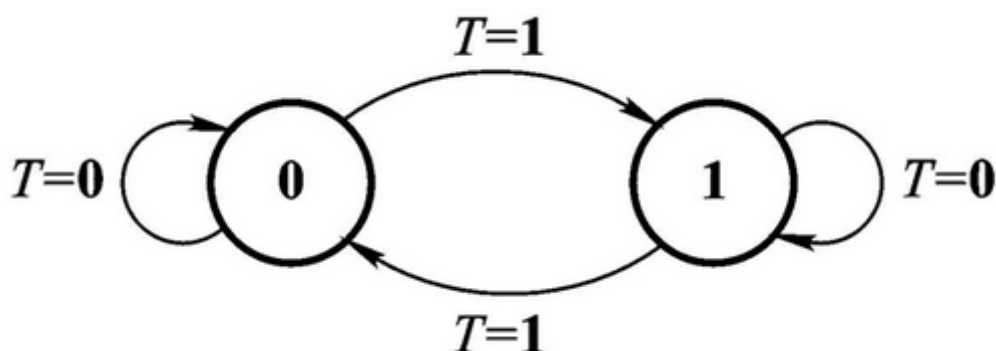


6.3 T触发器

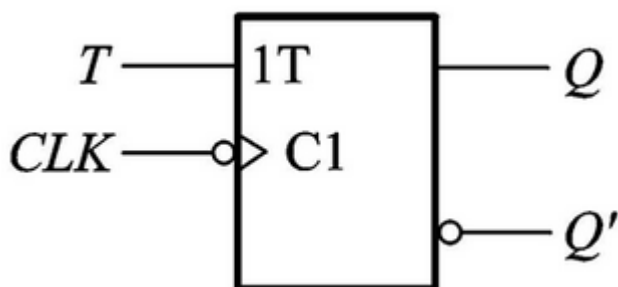
- 描述
 - T触发器的名称来源于“Toggle”，即翻转的意思。
 - T触发器有一个输入端（T）和一个时钟输入端（CLK）。
 - 当时钟信号的边沿（例如上升沿或下降沿）到来时，如果T输入为高电平（1），则触发器的输出会翻转（从0变为1或从1变为0）。
 - 如果T输入为低电平（0），则触发器的输出保持不变。
- 特性方程

$$Q^* = TQ' + T'Q$$

- 状态转换图



- 符号
 - 时钟端的 > 符号意味着该触发器为边沿触发。单个 > 为上升沿，加上 o 则表示下降沿。

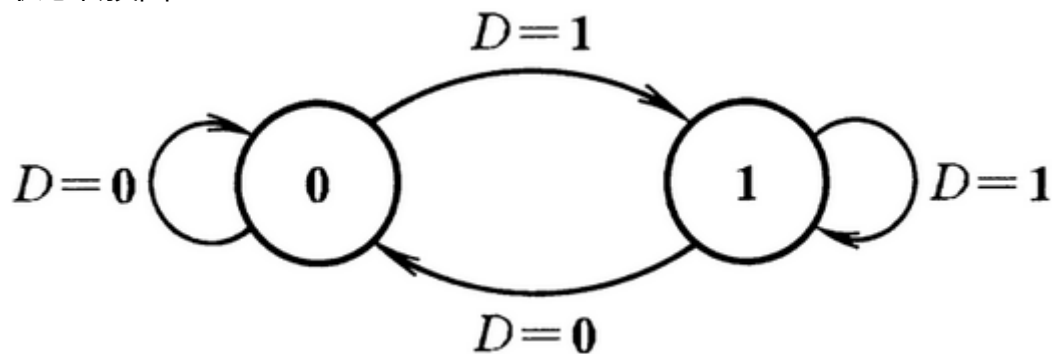


6.4 D触发器

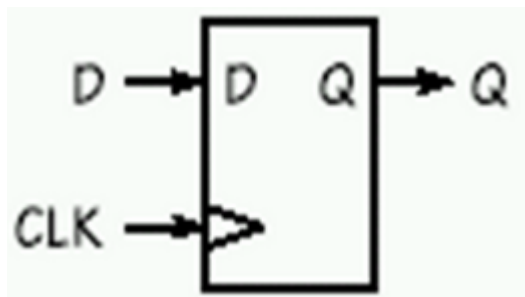
- 描述
 - D触发器有一个数据输入端（D）和一个时钟输入端（CLK）。
 - 当时钟信号的边沿到来时，D触发器将D输入端的值传递到输出端（Q）。
 - 其输出端（Q）在时钟信号的有效边沿保持D输入端的值，直到下一个有效边沿到来。
- 特性方程

$$Q^* = D$$

- 状态转换图

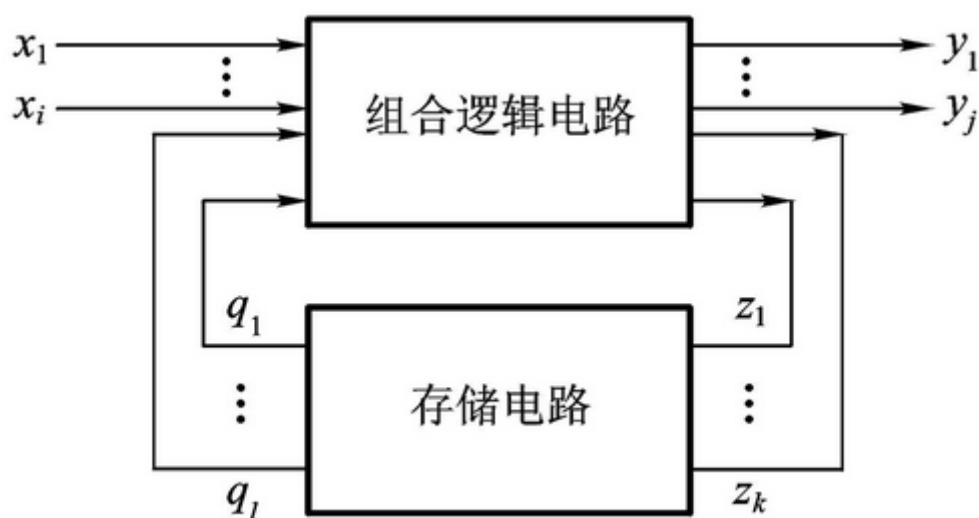


- 符号



时序逻辑电路

- 任一时刻的输出不仅取决于该时刻的输入，还与电路原来的状态有关。
- 结构上包含组合逻辑电路与存储电路，存储器状态与输入变量共同决定输出。



- 描述方程

- 驱动方程：描述了当前状态如何根据上一个时刻的状态和当前输入进行变化。

- $Z = F(X, Q)$

- Q表示当前时刻的状态。即目前存储电路的输出。

- 状态方程(状态转移方程)：是驱动方程的具体实现形式，通常用于表示状态寄存器的更新规则。

- $Q^* = H(Z, Q)$

- 实质就是触发器的规则。根据上一状态与输入得出下一状态。

- 输出方程

- $Y = F(X, Q)$

- 描述了电路的输出如何根据当前状态和输入进行变化。
- 时序电路分析
 - 从给定电路中写出每个存储电路中每个触发器的驱动方程（输入端的逻辑式）
 - 将其代入触发器的特性方程，得到状态方程。
 - 由给定电路得出输出方程。
 - 由状态方程得到状态转移表，由输出方程得到对应状态输出。
 - 画状态转移图 考到的时候只要画出主状态就行