

Digital Logic

Akhia¹

2020 年 9 月 20 日

¹E-mail:akhialomgir362856@gmail.com

目录

Chapter 1

基础

1.1 二进制八进制十进制十六进制相互转化

1.1.1 十进制和其他进制的相互转换

1. 其他进制转换为十进制：

各进制数按权展开并相加

2. 十进制转换为其他进制：

(a) 整数：除以基数取余，直到商为零，逆序

(b) 小数：乘以基数取整，顺序

1.1.2 二进制八进制十六进制的相互转换

以小数点为界向两侧划分，按基数划分组，不够则补零

1.1.3 8421BCD 码格雷码余 3 码与十进制之间的转换

1. 8421BCD
2. G
3. 余 3

1.2 十进制与原码反码补码之间的转换

符号位 0，正数反码补码和原码相同

符号位 1，负数反码数值取反，补码在反码最低有效位上加一

1.3 校验法

1. 奇偶校验码: 可以验证传输过程是否产生了错误
2. 奇校验: 为二进制添加一位校验码, 使 1 的数量为奇数
3. 偶校验: 为二进制添加一位校验码, 使 1 的数量为偶数
4. 海明码: 传输过程中错一位概率大, 通过海明码可以验证是哪位出错

Chapter 2

逻辑代数

2.1 逻辑代数运算法则

2.1.1 逻辑运算

1. $F = A + B$ ($F = A \vee B$)

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

2. $F = A \cdot B$ ($F = A \wedge B$)

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

3. $F = \overline{A}$

A	F
0	1
1	0

2.1.2 基本定律

1	$A + B = B + A$	$AB = BA$
2	$A + (B + C) = (A + B) + C$	$A(BC) = (AB)C$
3	$A + (BC) = (A + B)(A + C)$	$A(B + C) = AB + AC$
4	$A + 0 = A, A + 1 = 0$	$A \cdot 1 = A, A \cdot 0 = 0$
5	$A + \bar{A} = 1$	$A \cdot \bar{A} = 0$
6	$A + A = A$	$A \cdot A = A$
7		$\bar{\bar{A}} = A$
8	$\overline{A + B} = \bar{A} \cdot \bar{B}$	$\overline{AB} = \bar{A} + \bar{B}$

2.1.3 代入规则

2.1.4 反演规则

$$F \left\{ \begin{array}{l} 1 \Longleftrightarrow 0 \\ + \Longleftrightarrow \bullet \\ A \Longleftrightarrow \overline{A} \end{array} \right\} \overline{F} \quad (2.1)$$

2.1.5 对偶规则

$$F \left\{ \begin{array}{l} 1 \Longleftrightarrow 0 \\ + \Longleftrightarrow \bullet \end{array} \right\} F' \quad (2.2)$$

2.1.6 常用公式

2.2 逻辑函数标准形式

1. 最小项及标准与或式

最小项: 与项包含全部 n 个变量, 全部以原变量或反变量的形式出现, 且只出现一次

例如: $A\bar{B}C$

函数简写作: $\sum m(0, 1, 2, 3 \dots)$

2. 最大项及标准或与式

最大项: 或项包含全部 n 个变量, 全部以原变量或反变量的形式出现, 且只出现一次

例如: $A + \bar{B} + C$

函数简写作: $\prod M(0, 1, 2, 3 \dots)$

3. 两者转换

(a) 代数转换法

- i. 转换到最小项之和
- ii. 转换为一般与或式
- iii. 将非最小项拓展到最小项
 - i. 转换到最大项之积
 - ii. 转换为一般或与式
 - iii. 将非最大项转换到最大项

(b) 真值表转换法

- i. 1: $\sum m(0, 2, 4, 6 \dots)$
- ii. 0: $\prod M(1, 3, 5, 7 \dots)$

2.3 逻辑函数化简

1. 代数化简化

(a) 与或

(b) 并项法 $AB + A\bar{B} = A$

(c) 吸收法 $A + AB = A$

(d) 消去法 $A + \bar{A}B = A + B$

(e) 配项法 $A \cdot 1 = 1, A + \bar{A} = 1$

(f) 或与

(g) 定理法

(h) 求偶得到 F' () 化简再求偶得到 F

2. 卡诺图简化

$\begin{smallmatrix} CD \\ AB \end{smallmatrix}$	00	01	11	10
00	$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}\bar{B}\bar{C}D$	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}\bar{B}CD$
01	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}\bar{B}CD$	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}B\bar{C}D$
11	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}B\bar{C}D$	$\bar{A}BC\bar{D}$	$\bar{A}BCD$
10	$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}\bar{C}D$	$A\bar{B}C\bar{D}$	$A\bar{B}CD$

(a) $AB + A\bar{B} = A$

(b) $\bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC = \bar{A}\bar{C} + \bar{A}C = \bar{A}$

(a) 卡诺圈尽量大

(b) 卡诺圈个数尽量少

(c) 每个 1 可以被多个卡诺圈包含

Chapter 3

组合逻辑电路

3.1 逻辑门

1. 与门

符号: $A \text{---} \boxed{\&} \text{---} F$

表达式: $F = A \cdot B$

2. 或门

符号: $A \text{---} \boxed{\geq 1} \text{---} F$

表达式: $F = A + B$

3. 非门

符号: $A \text{---} \boxed{1} \text{---} F$

表达式: $F = \overline{A}$

4. 复合逻辑门

(a) 与非门

符号: $A \text{---} \boxed{\&} \text{---} F$

表达式: $F = \overline{AB}$

(b) 或非门

符号: $A \text{---} \boxed{\geq 1} \text{---} F$

表达式: $F = \overline{A + B}$

(c) 与或非门

表达式: $F = \overline{AB + CD}$

(d) 异或门

符号: $A \text{---} \boxed{= 1} \text{---} F$

表达式: $F = A \oplus B = A\overline{B} + \overline{A}B$

同或运算: $F = A \odot B = AB + \overline{A}\overline{B}$

1. 根据输入逐级写出输出
2. 化简逻辑功能
3. 列出真值表
4. 讨论功能

3.2 逻辑函数实现

$$\begin{aligned} F(A, B, C) &= AB + \overline{A}C \\ &= (\overline{A} + B)(A + C) \\ &= \overline{[(A + C) + (\overline{A} + B)]} \\ &= \overline{(\overline{A}B + \overline{A})} \end{aligned} \tag{3.1}$$

1. 与非

(a) 化为最简 ** 与或 ** 式

(b) 变换为 ** 与非²** 式

2. 或非

(a) 化为最简 ** 或与 ** 式

(b) 变换为 ** 或非²** 式

3. 与或非

(a) 化为最简 ** 与或 ** 式

(b) 变换为 ** 与或非 ** 式

4. 异或 (部分才能实现, 但简单)

3.3 组合逻辑电路分析

1. 输入标字母
2. 从输入端按深度一层层写出逻辑函数
3. 用前一层输出代入后一层并继续重复
4. 简化逻辑函数, 判断合理性
5. 列出逻辑电路的真值表
6. 判断功能并评价完善

3.4 组合逻辑电路设计

1. 根据逻辑要求构建真值表
2. 根据真值表写出逻辑函数
3. 将逻辑函数化简并转换成适当形式

3.5 竞争与冒险

原因: 信号传输延迟

用输入、输出时序图表示

3.5.1 竞争

竞争: 输入信号通过不同途径达到输出端的时间不同 (随机过程)

1. 非临界竞争: 不会产生错误
2. 临界竞争: 导致逻辑错误

3.5.2 冒险

冒险 (暂时、瞬态现象): 输出端的尖脉冲

OUT_EXPECTED		OUT_ERROR	
静态冒险	偏 1	1	1-0-1
	偏 0	0	0-1-0
动态冒险	偏 1	0-1	0-1-0-1
	偏 0	1-0	1-0-1-0

1. 代数判别法

从函数表达式结构判别

- (a) 如果某变量同时以原变量反变量形式存在
- (b) 将其他变量可能的取值代入
- (c) 如果出现 $x + \bar{x}$ 或 $x\bar{x}$ 则可能产生冒险

2. 卡诺图判别法

- (a) 画出各 ** 与项 ** 对应卡诺圈
- (b) 如果两卡诺圈 ** 相切 ** (存在共用的相邻最小项) 则可能产生冒险

Chapter 4

同步时序电路

4.1 结构

1. 组合电路
2. 储存电路
1. 输出函数

$$Z_i = f_i(x_1, \dots, x_n, y_1, \dots, y_r), \quad i = 1, \dots, m \quad (4.1)$$

2. 激励函数

$$Y_j = g_j(x_1, \dots, x_n, y_1, \dots, y_r), \quad j = 1, \dots, r \quad (4.2)$$

统一的时钟信号 (不能太短) 来临后, 电路状态才改变, 且只有一次。

时钟信号起同步作用

信号来临前	现态	y^n
信号来临后	次态	y^{n+1}

4.2 描述

1. 状态表
2. 状态图

4.3 触发器

1. 储存电路
2. 能储存一位二进制数
3. 在任一时刻只处于一种稳态

4.3.1 R-S 触发器

1. 基本型 (锁存器)
2. 直接复位-置位
3. 组成:
4. 交叉耦合或非门
5. 交叉耦合与非门
6. 时钟型
7. 4与非

空翻现象: 由于时钟信号宽度而多次翻转, 可由主从触发器 (串联) 解决

4.3.2 D 触发器

为了解决 R-S 触发器输入同为 1 时触发器状态不确定问题

1. 单输入端
2. 输入信号转换为互补信号

4.3.3 J-K 触发器

为了解决 R-S 触发器输入同为 1 时触发器状态不确定问题, 同时使触发器有两个输入端

4.3.4 T 触发器

JK 端合并为 T 端

4.4 电路分析

4.5 电路设计

Chapter 5

异步时序电路

5.1 特性

5.2 脉冲异步分析与设计

5.3 电平异步分析与设计

5.3.1 电平异步竞争与冒险

Chapter 6

规模集成电路逻辑设计

- 6.1 二进制并行加法器
- 6.2 数值比较器
- 6.3 译码器
- 6.4 多路选择器
- 6.5 计数器
- 6.6 寄存器
- 6.7 只读存储器
- 6.8 可编程逻辑阵列
- 6.9 可编程阵列逻辑
- 6.10 通用阵列逻辑
- 6.11 高密度可编程逻辑器件