

# Digital Logic

Akhia<sup>1</sup>

2020 年 10 月 12 日

<sup>1</sup>E-mail:akhialomgir362856@gmail.com

# 目录

<b>1</b>	<b>基础</b>	<b>6</b>
1.1	二进制八进制十进制十六进制相互转化 . . . . .	7
1.1.1	十进制和其他进制的相互转换 . . . . .	7
1.1.2	二进制八进制十六进制的相互转换 . . . . .	7
1.1.3	8421BCD 码格雷码余 3 码与十进制之间的转换 . . . . .	7
1.2	十进制与原码反码补码之间的转换 . . . . .	7
1.3	校验法 . . . . .	7
<b>2</b>	<b>逻辑代数</b>	<b>9</b>
2.1	逻辑代数运算法则 . . . . .	10
2.1.1	逻辑运算 . . . . .	10
2.1.2	基本定律 . . . . .	11
2.1.3	代入规则 . . . . .	12
2.1.4	反演规则 . . . . .	12
2.1.5	对偶规则 . . . . .	12
2.1.6	常用公式 . . . . .	12
2.2	逻辑函数标准形式 . . . . .	13
2.3	逻辑函数化简 . . . . .	14
<b>3</b>	<b>组合逻辑电路</b>	<b>15</b>
3.1	逻辑门 . . . . .	16
3.2	逻辑函数实现 . . . . .	17
3.3	组合逻辑电路分析 . . . . .	19
3.4	组合逻辑电路设计 . . . . .	19

3.5	竞争与冒险 . . . . .	20
3.5.1	竞争 . . . . .	20
3.5.2	冒险 . . . . .	20
<b>4</b>	<b>同步时序电路</b>	<b>22</b>
4.1	结构 . . . . .	23
4.1.1	Mealy,Moore 型电路 . . . . .	24
4.2	描述 . . . . .	25
4.3	触发器 . . . . .	26
4.3.1	R-S 触发器 . . . . .	26
4.3.2	D 触发器 . . . . .	27
4.3.3	J-K 触发器 . . . . .	27
4.3.4	T 触发器 . . . . .	27
4.4	电路分析 . . . . .	28
4.4.1	隐含表 . . . . .	28
4.4.2	合并图 . . . . .	28
4.4.3	闭覆盖表 . . . . .	28
4.5	电路设计 . . . . .	28
<b>5</b>	<b>异步时序电路</b>	<b>29</b>
5.1	特性 . . . . .	30
5.2	脉冲异步分析与设计 . . . . .	30
5.3	电平异步分析与设计 . . . . .	30
5.3.1	电平异步竞争与冒险 . . . . .	31
<b>6</b>	<b>规模集成电路逻辑设计</b>	<b>32</b>
6.1	二进制并行加法器 . . . . .	33
6.1.1	逻辑功能 . . . . .	33
6.1.2	结构 . . . . .	33
6.1.3	原理 . . . . .	33
6.2	数值比较器 . . . . .	34
6.2.1	逻辑功能 . . . . .	34

6.2.2	结构	34
6.2.3	原理	34
6.3	译码器	35
6.3.1	逻辑功能	35
6.3.2	结构	35
6.3.3	原理	35
6.4	多路选择器	36
6.4.1	逻辑功能	36
6.4.2	结构	36
6.4.3	原理	36
6.5	计数器	37
6.5.1	逻辑功能	37
6.5.2	结构	37
6.5.3	原理	37
6.6	寄存器	38
6.6.1	逻辑功能	38
6.6.2	结构	38
6.6.3	原理	38
6.7	只读存储器 (ROM)	39
6.7.1	逻辑功能	39
6.7.2	结构	39
6.7.3	原理	39
6.8	可编程逻辑阵列 (PLA)	40
6.8.1	逻辑功能	40
6.8.2	结构	40
6.8.3	原理	40
6.9	可编程阵列逻辑 (PAL)	41
6.9.1	逻辑功能	41
6.9.2	结构	41
6.9.3	原理	41
6.10	通用阵列逻辑 (GAL)	42

6.10.1	功能	42
6.10.2	结构	42
6.10.3	原理	42
6.11	高密度可编程逻辑器件 (HDPLD)	43
6.11.1	功能	43
6.11.2	结构	43
6.11.3	原理	43

# 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		$\overline{\bar{A}} = A$
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: $\overline{A}BC$

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: $A + \overline{B} + C$

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  $F'$  (与或) 化简再求偶得到  $F$

### 2. 卡诺图简化

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

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

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

(a)  $2^n$  圈

(b) 卡诺圈尽量大

(c) 卡诺圈个数尽量少

(d) 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:  $\begin{matrix} A \\ B \end{matrix} \rightarrow \boxed{\&} \rightarrow F$

表达式:  $F = A \cdot B$

2. 或门

符号:  $\begin{matrix} A \\ B \end{matrix} \rightarrow \boxed{\geq 1} \rightarrow F$

表达式:  $F = A + B$

3. 非门

符号:  $A \rightarrow \boxed{1} \circ \rightarrow F$

表达式:  $F = \overline{A}$

4. 复合逻辑门

(a) 与非门

符号:  $\begin{matrix} A \\ B \end{matrix} \rightarrow \boxed{\&} \circ \rightarrow F$

表达式:  $F = \overline{AB}$

(b) 或非门

符号:  $\begin{matrix} A \\ B \end{matrix} \rightarrow \boxed{\geq 1} \circ \rightarrow F$

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:  $\begin{matrix} A \\ B \end{matrix} \rightarrow \boxed{= 1} \rightarrow F$

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

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

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。

#### 4.1.1 Mealy,Moore 型电路

1. moore 的输出仅与当前电路状态有关
2. mealy 的输出与当前电路状态和输入都有关



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：交叉耦合或非门交叉耦合与非门
4. 时钟型两个控制与非门+两个基本耦合与非门

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

1. 根据电路，列出输出函数表达式和激励函数表达式
2. 建立状态转移真值表
3. 作出电路状态表，画出状态图
4. 用文字和时间图描述电路逻辑功能

### 4.4.1 隐含表

### 4.4.2 合并图

### 4.4.3 闭覆盖表

## 4.5 电路设计

1. 根据逻辑要求，作出原始状态图和状态表
2. 状态简化
3. 状态编码
4. 求出激励函数和输出函数表达式
5. 画出逻辑电路图

## Chapter 5

### 异步时序电路

## 5.1 特性

1. 电路没有统一的时钟信号，电路状态改变直接由外部输入信号变化引起
2. 类别：脉冲电平
3. 每一时刻仅允许一个输入发生变化
4. 只有电路进入一个新的稳定状态时才允许输入变化
5. Mealy 模型：输出不仅与输入状态有关，还与二次状态有关
6. Moore 模型：输出仅与二次状态有关
7. 研究工具：脉冲：状态图，状态表电平：状态流程图，时序图
8. 电平信号：基本信号
9. 脉冲信号：连续两次电平跳变

## 5.2 脉冲异步分析与设计

脉冲异步时序电路分析设计方法与同步时序电路相似，但有输入信号限制：

1. 不允许同时出现两个及以上的输入脉冲
2. 对于  $n$  个输入端的电路，仅有  $n + 1$  种不同的输入信号组合
3. 第二个脉冲到达必须在第一个脉冲引起的电路响应结束后

## 5.3 电平异步分析与设计

信号限制：

1. 同时只允许一个输入电平发生变化，而且且一定要是相邻的

2. 输入电平变化必须在第一个变化引起的电路响应结束后
1. 根据逻辑电路图，写出激励函数和输出函数表达式
2. 列出状态流程表
3. 作出时序图
4. 说明电路逻辑功能

### 5.3.1 电平异步竞争与冒险

#### 本质冒险

输入的原变量与反变量由于延迟没有同时保持互补状态，输入信号通过反馈回路的延迟小于通过反相器的延迟，电路出现不正常转移。

要消除本质冒险，需要选择元件延迟特性或在反馈回路中加入足够的延迟。

## Chapter 6

### 规模集成电路逻辑设计



## 6.1 二进制并行加法器

### 6.1.1 逻辑功能

产生两个二进制算术和。

### 6.1.2 结构

由全加器构成进位链无法完成超前进位。

74283

### 6.1.3 原理

## **6.2 数值比较器**

### **6.2.1 逻辑功能**

比较两个正数而确定其相对大小。

### **6.2.2 结构**

7485

### **6.2.3 原理**

## 6.3 译码器

### 6.3.1 逻辑功能

将  $n$  个输入变量变换为  $2^n$  个输出函数，每个输出对应一个最小项。

### 6.3.2 结构

74138

### 6.3.3 原理

## **6.4 多路选择器**

### **6.4.1 逻辑功能**

多路输入，单路输出，从多个输入中选择一个信号输出。

### **6.4.2 结构**

74153

### **6.4.3 原理**

## **6.5 计数器**

### **6.5.1 逻辑功能**

对输入脉冲信号进行计数。

### **6.5.2 结构**

分为同步，异步。

74193

### **6.5.3 原理**

## **6.6 寄存器**

### **6.6.1 逻辑功能**

存放数据或运算结果，具有接收数据、储存数据或传入数据的功能。

### **6.6.2 结构**

74194

### **6.6.3 原理**

## **6.7 只读存储器（ROM）**

### **6.7.1 逻辑功能**

只读不写的存储器，断电后依然保存数据。

### **6.7.2 结构**

### **6.7.3 原理**

## **6.8 可编程逻辑阵列（PLA）**

### **6.8.1 逻辑功能**

解决 ROM 存在地址译码和储存单元必须一一对应而浪费空间的缺陷，“与”阵列、“或”阵列都可以编程。

### **6.8.2 结构**

### **6.8.3 原理**



## **6.9 可编程阵列逻辑(PAL)**

### **6.9.1 逻辑功能**

可编程逻辑阵列（PLA）的“与”阵列可编程，而“或”阵列固定。

### **6.9.2 结构**

### **6.9.3 原理**

## 6.10 通用阵列逻辑（GAL）

### 6.10.1 功能

与 PAL 类似，但不采用熔丝 I/O，而是使用 OLMC，可以重复改写。  
改进：

1. 采用  $E^2CMOS$  功耗低，速度快，可以电擦写和反复编程
2. 输出结构配置了输出逻辑宏单元，可以编程选择输出组态
3. 有加密单元，防复制，增加保密性

### 6.10.2 结构

### 6.10.3 原理

## **6.11 高密度可编程逻辑器件（HDPLD）**

### **6.11.1 功能**

EPLD,CPLD,FPGA

### **6.11.2 结构**

### **6.11.3 原理**