

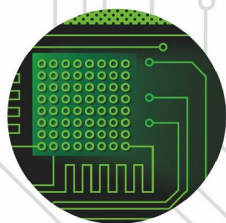
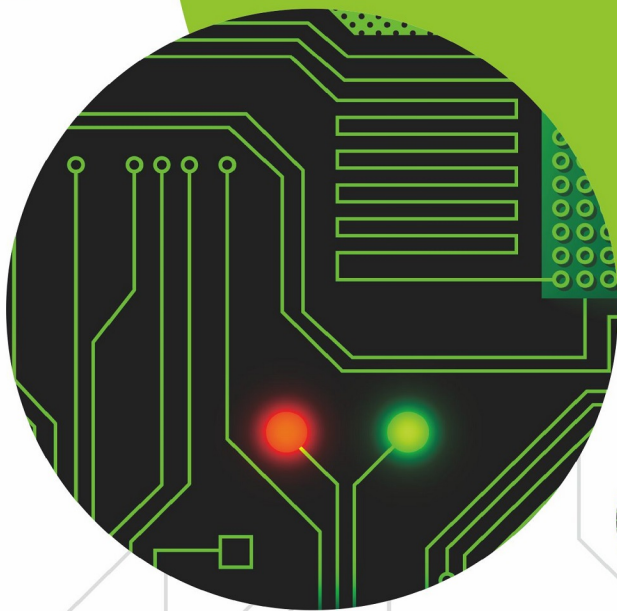
教育部高等学校电工电子基础课程教学指导分委员会推荐教材

数字电子技术

SHUZI DIANZI JISHU

(第二版)

俞阿龙 孙红兵 魏东旭 杨裕翠 戴金桥 杨军 编著



南京大学出版社

图书在版编目(CIP)数据

数字电子技术 / 俞阿龙等编著. — 2 版. — 南京:
南京大学出版社, 2026. 8. — ISBN 978-7-305-30330-2
I. TN79
中国国家版本馆 CIP 数据核字第 2026VC0352 号

出版发行 南京大学出版社
社 址 南京市汉口路 22 号 邮 编 210093
书 名 数字电子技术
SHUZI DIANZI JISHU
编 著 俞阿龙 孙红兵 魏东旭 杨裕翠 戴金桥 杨 军
责任编辑 高司洋 编辑热线 025-83686531
照 排 南京南琳图文制作有限公司
印 刷 南京京新印刷有限公司
开 本 787 mm×1092 mm 1/16 印张 23.75 字数 603 千
版 次 2026 年 8 月第 2 版 2026 年 8 月第 1 次印刷
ISBN 978-7-305-30330-2
定 价 69.00 元
网址: <http://www.njupco.com>
官方微博: <http://weibo.com/njupco>
官方微信号: njupress
销售咨询热线: (025) 83594756

* 版权所有, 侵权必究
* 凡购买南大版图书, 如有印装质量问题, 请与所购
图书销售部门联系调换

第二版前言

数字电子技术是高等学校电子信息类、电气类、自动化类等专业的必修课程,具有很强的工程性和实践性。教材建设是高等院校教育教学工作的重要组成部分,高质量的教材是人才培养的基本保证。随着我国高等教育人才培养模式趋于多样化,作为教学活动重要载体的教材,其多样化的特点也变得更加突出,以适应多样化的人才培养需求。编者于2011年出版《数字电子技术》教材第一版,并在2019年进行修订。为适应数字电子技术的发展,本次改版过程中,仍以应用型本科人才为培养目标,进一步加强理论与实践相结合,突出实用性、先进性、应用性以及课程与培养目标的达成度。为此,在内容的选取和安排上进一步优化:着重讲清数字电路的基本概念、基本知识、基本方法及工程应用;进一步弱化分立器件的讲述;进一步精简数字集成电路内部电路的分析,重点聚焦于功能与应用;在第10章中新增硬件描述语言简介;对课后练习题进行修订,并增设自我检测题,读者可通过扫描二维码获取自我检测题和部分练习题参考答案。

本版教材共分十章,内容涵盖绪论、逻辑代数基础、集成逻辑门电路、组合逻辑电路、集成触发器、时序逻辑电路、脉冲波形的产生与变换、数模和模数转换器、半导体存储器、可编程逻辑器件。

本书可作为高等学校电子信息类、电气类、自动化类等专业的教材或参考书,也可供相关工程技术人员参考。

本书修订改版过程中,我们参考了国内外相关书刊及网络文献,谨向所有被引用文献的作者致以谢意。

限于作者的水平,书中难免有疏漏和不妥之处,恳请读者批评指正。

编 者
2026年3月

目 录

第 1 章 绪 论	1
1.1 概 述	1
1.1.1 数字信号和数字电路	1
1.1.2 数字电路的特点和分类	3
1.2 数制	4
1.2.1 几种常用的数制	4
1.2.2 不同数制间的转换	6
1.3 二进制代码	8
1.3.1 二-十进制码	8
1.3.2 格雷码	9
1.3.3 ASCII 码	10
1.4 二进制数的算术运算	11
1.4.1 无符号二进制数的算术运算	11
1.4.2 有符号二进制数的算术运算	13
本章小结	14
自我检测题	15
练习题	16
第 2 章 逻辑代数基础	17
2.1 概 述	17
2.2 逻辑代数中的常用运算	18
2.2.1 基本逻辑运算	18
2.2.2 复合逻辑运算	20
2.3 逻辑代数中的基本定律和常用公式	23
2.3.1 逻辑代数中的基本定律	23
2.3.2 逻辑代数中的常用公式	24
2.3.3 逻辑代数中的三个基本规则	26

2.4	逻辑函数及其表示方法	26
2.4.1	逻辑函数的建立	27
2.4.2	逻辑函数的表示方法	27
2.4.3	逻辑函数的两种标准形式	30
2.5	逻辑函数的公式化简法	32
2.5.1	逻辑函数的最简表达式	33
2.5.2	逻辑函数的公式化简法	33
2.6	逻辑函数的卡诺图化简法	36
2.6.1	卡诺图的构成	36
2.6.2	逻辑函数的卡诺图表示法	38
2.6.3	逻辑函数的卡诺图化简法	40
	本章小结	45
	自我检测题	46
	练习题	47
第3章	集成逻辑门电路	51
3.1	概 述	51
3.2	半导体器件的开关特性	52
3.2.1	二极管的开关特性	53
3.2.2	三极管的开关特性	54
3.2.3	MOS管的开关特性	56
3.3	TTL集成逻辑门电路	58
3.3.1	TTL与非门	58
3.3.2	其他功能的TTL门电路	67
3.3.3	其他系列的TTL门电路	72
3.3.4	TTL集成逻辑门电路的使用注意事项	75
3.4	CMOS集成逻辑门电路	77
3.4.1	CMOS反相器	77
3.4.2	其他功能的CMOS门电路	79
3.4.3	高速CMOS门电路	83
3.4.4	CMOS数字集成逻辑门电路使用注意事项	84
3.5	TTL门电路与CMOS门电路的接口	85
3.5.1	TTL门电路驱动CMOS门电路	86

3.5.2	CMOS 门驱动 TTL 门	86
3.5.3	TTL 和 CMOS 门电路带负载时的接口电路	87
	本章小结	88
	自我检测题	89
	练习题	89
第 4 章	组合逻辑电路	94
4.1	概 述	94
4.1.1	组合逻辑电路的特点	94
4.1.2	组合逻辑电路的逻辑功能描述	94
4.2	组合逻辑电路的分析和设计	95
4.2.1	组合逻辑电路的分析	95
4.2.2	组合逻辑电路的设计	97
4.3	加法器	102
4.3.1	半加器和全加器	102
4.3.2	多位加法器	105
4.4	编码器	111
4.4.1	二进制编码器	112
4.4.2	二-十进制编码器	114
4.4.3	优先编码器	117
4.5	译码器和数据分配器	121
4.5.1	二进制译码器	121
4.5.2	二-十进制译码器	124
4.5.3	显示译码器	128
4.5.4	译码器的应用	131
4.5.5	数据分配器	135
4.6	数据选择器	137
4.6.1	4 选 1 数据选择器	137
4.6.2	8 选 1 数据选择器	139
4.6.3	数据选择器的应用	140
4.7	数值比较器	144
4.7.1	1 位数值比较器	144
4.7.2	多位数值比较器	145

* 4.8 组合逻辑电路中的竞争与冒险	149
4.8.1 产生竞争冒险的原因	149
4.8.2 冒险的分类	151
4.8.3 竞争冒险现象的判别	152
4.8.4 消除竞争冒险的方法	154
本章小结	157
自我检测题	158
练习题	159
第5章 集成触发器	164
5.1 概 述	164
5.2 基本 RS 触发器	165
5.2.1 由与非门组成的基本 RS 触发器	165
5.2.2 由或非门组成的基本 RS 触发器	167
5.3 同步触发器	170
5.3.1 同步 RS 触发器	170
5.3.2 同步 D 触发器	173
5.3.3 同步 JK 触发器	174
5.3.4 同步触发器的空翻	175
5.4 边沿触发器	176
5.4.1 边沿 JK 触发器	176
5.4.2 维持阻塞 D 触发器	178
5.4.3 T 触发器	181
5.5 主从触发器	181
5.5.1 主从 RS 触发器	181
5.5.2 主从 JK 触发器	184
5.5.3 主从 JK 触发器的一次翻转现象	186
5.6 触发器的逻辑功能及相互之间的转换	187
5.6.1 触发器的逻辑功能	187
5.6.2 触发器相互之间的转换	191
本章小结	192
自我检测题	192
练习题	193

第 6 章 时序逻辑电路	198
6.1 概 述	198
6.1.1 时序逻辑电路的特点与结构	198
6.1.2 时序逻辑电路的分类	199
6.1.3 时序逻辑电路功能的描述方法	200
6.2 时序逻辑电路的分析	202
6.2.1 同步时序逻辑电路的分析方法	202
6.2.2 异步时序逻辑电路的分析方法	210
6.3 寄存器和移位寄存器	214
6.3.1 寄存器	215
6.3.2 移位寄存器	217
6.3.3 移位寄存器的应用	223
6.4 计数器	226
6.4.1 异步计数器	226
6.4.2 同步计数器	230
6.4.3 集成计数器	238
6.4.4 利用计数器的级联获得大容量 N 进制计数器	241
* 6.5 同步时序逻辑电路的设计	245
6.5.1 同步时序逻辑电路的设计方法	245
6.5.2 同步时序逻辑电路的设计举例	246
本章小结	253
自我检测题	253
练习题	254
第 7 章 脉冲波形的变换与产生	262
7.1 概 述	262
7.2 555 定时器	262
7.2.1 555 定时器的电路结构	263
7.2.2 555 定时器的工作原理	263
7.3 施密特触发器	264
7.3.1 用 555 定时器组成的施密特触发器	265
7.3.2 集成施密特触发器	266

7.3.3 施密特触发器的应用·····	268
7.4 单稳态触发器·····	269
7.4.1 用 555 定时器组成的单稳态触发器·····	270
7.4.2 集成单稳态触发器·····	272
7.4.3 单稳态触发器的应用·····	275
7.5 多谐振荡器·····	277
7.5.1 用 555 定时器组成的多谐振荡器·····	277
7.5.2 用施密特触发器组成的多谐振荡器·····	280
7.5.3 用单稳态触发器组成的多谐振荡器·····	281
7.5.4 石英晶体多谐振荡器·····	281
本章小结·····	283
自我检测题·····	283
练习题·····	285
第 8 章 数模和模数转换器·····	291
8.1 概 述·····	291
8.2 D/A 转换器·····	292
8.2.1 权电阻网络 D/A 转换器·····	292
8.2.2 $R-2R$ 倒 T 型电阻网络 D/A 转换器·····	294
8.2.3 权电流型 D/A 转换器·····	298
8.2.4 D/A 转换器的主要参数·····	300
8.3 A/D 转换器·····	301
8.3.1 A/D 转换的一般过程·····	301
8.3.2 并联比较型 A/D 转换器·····	304
8.3.3 逐次逼近型 A/D 转换器·····	305
8.3.4 双积分型 A/D 转换器·····	308
8.3.5 A/D 转换器的主要参数·····	311
本章小结·····	312
自我检测题·····	313
练习题·····	314
第 9 章 半导体存储器·····	318
9.1 概 述·····	318

9.1.1 半导体存储器的分类·····	318
9.1.2 半导体存储器的主要技术指标·····	319
9.2 只读存储器 ·····	320
9.2.1 ROM 的电路结构 ·····	320
9.2.2 固定 ROM 的工作原理 ·····	320
9.2.3 可编程只读存储器·····	322
9.2.4 可擦除可编程只读存储器·····	323
9.2.5 ROM 的应用 ·····	327
9.3 随机存取存储器 ·····	330
9.3.1 RAM 的电路结构 ·····	330
9.3.2 RAM 中的存储单元 ·····	332
9.3.3 RAM 容量的扩展 ·····	334
本章小结·····	337
自我检测题·····	337
练习题·····	338
 第 10 章 可编程逻辑器件 ·····	 341
10.1 概 述·····	341
10.2 可编程逻辑器件的基本结构·····	342
10.2.1 PLD 的基本结构 ·····	342
10.2.2 PLD 器件的表示法 ·····	343
10.3 可编程阵列逻辑·····	345
10.3.1 PAL 的基本结构 ·····	345
10.3.2 PAL 的输出和反馈结构 ·····	346
10.4 通用阵列逻辑·····	347
10.4.1 GAL 的基本结构·····	347
10.4.2 GAL 的输出宏单元 ·····	348
10.5 现场可编程门阵列·····	349
10.5.1 FPGA 的基本结构 ·····	350
10.5.2 FPGA 的模块功能 ·····	350
10.6 硬件描述语言简介·····	354
10.6.1 Verilog HDL 基本语法简介 ·····	354
10.6.2 Verilog HDL 应用实例 ·····	357

本章小结·····	360
自我检测题·····	361
练习题·····	362
附 录 ·····	364
附录 1 数字电子技术模拟试卷(A、B) ·····	364
附录 2 自我检测题参考答案 ·····	364
附录 3 部分练习题参考答案 ·····	364
参考文献 ·····	365

第1章 绪 论

本章学习目的和要求

1. 了解模拟信号与数字信号、模拟电路与数字电路的区别与联系；
2. 掌握数字量、数制的概念及不同数制的互化；
3. 掌握数字电路中常用的二进制代码；
4. 掌握二进制数的算术运算方法。

随着电子技术的迅速发展,人类社会已经进入数字化时代。数字电子技术计算机、数字通信、智能手机、电子测量技术和工业控制系统等方面得到了广泛应用。数字电子技术是研究数字信号处理、传输和存储的电子技术,数字电路是数字电子技术的核心。本章首先介绍数字信号和数字电路的一些基本概念及其特点,接着介绍数字电路中常用的数制及不同数制之间的转换,然后介绍二进制代码的概念,最后介绍二进制数的算术运算方法。

1.1 概 述

1.1.1 数字信号和数字电路

在我们周围的生活中存在许多物理量,我们分析它们的信号波形可以发现有两类性质不同的物理量,一类是在时间和幅度均作连续变化的物理量,另一类是在时间和幅度上都不连续(离散)的物理量,这两类物理量分别称为模拟量和数字量。在电子电路中,将表示模拟量的信号称为模拟信号,将表示数字量的信号称为数字信号,如图 1.1.1 所示。自然界客观存在的一些物理量,如温度、湿度、速度、压力、应变、声音等通常是模拟量,它们可以通过转换器转换成模拟信号。用于传送、加工和处理模拟信号的电路称为模拟电路。像电子表的秒信号、生产流水线上记录零件个数的计数信号等是数字信号,这些信号的变化发生在一系

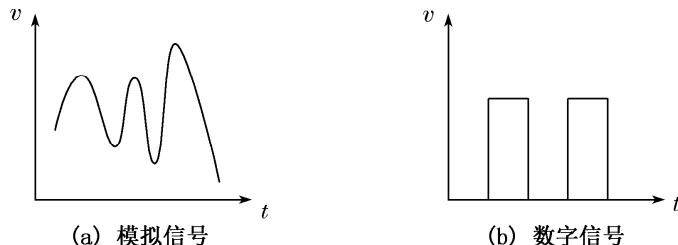


图 1.1.1 模拟信号和数字信号

列离散的瞬间,其值也是离散的。这种数字信号只有两个离散值,常用数字 0 和 1 来表示,代表两种对立的状态,称为逻辑 0 和逻辑 1,也称为二值数字逻辑。我们可以用高电平来表示逻辑 1,用低电平来表示逻辑 0,这称为正逻辑体制;当然也可以用低电平来表示逻辑 1,用高电平来表示逻辑 0,这称为负逻辑体制。如表 1.1.1 所示。

表 1.1.1 二值数字逻辑

	高电平	低电平
正逻辑关系	1	0
负逻辑关系	0	1

用于传送、加工和处理数字信号的电路称为数字电路。由于数字电路的各种功能是通过逻辑运算和逻辑判断来实现的,因此数字电路有时也称数字逻辑电路或逻辑电路。数字信号在电路中往往表现为突变的电压或电流。数字信号具有以下一些特点:

(1) 信号只有两个电压值,如 5 V 和 0 V。如果采用正逻辑体制,则 5 V 为高电平,0 V 为低电平。

(2) 信号从高电平变为低电平,或者从低电平变为高电平是一个突然变化的过程。如果采用正逻辑,图 1.1.1(b)所示的数字电压信号就变为如图 1.1.2 所示的逻辑信号,这是一种理想的矩形脉冲信号。

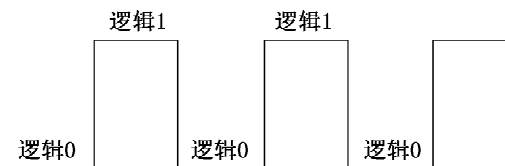


图 1.1.2 逻辑信号

在一个实际的数字系统中,矩形脉冲信号一般由电子器件的开关特性产生,其波形并没有那么理想,实际波形从低电平跳变到高电平(上升沿),或从高电平跳变到低电平(下降沿)其边沿并没有那么陡峭,而要经历一个过渡过程,如图 1.1.3 所示。它可用信号幅度(V_m)、信号的重复周期(T)、脉冲宽度(t_w)、占空比(q ,脉冲宽度 t_w 占整个周期 T 的百分比)等参数来描述,具体参数介绍如下。

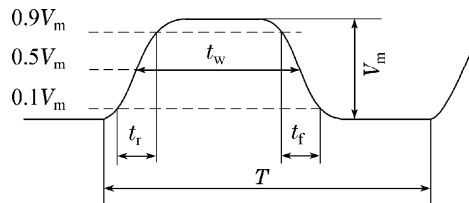


图 1.1.3 实际的矩形脉冲信号

脉冲幅度 V_m : 脉冲信号变化的最大值,单位一般为伏(V)。

脉冲上升时间 t_r : 脉冲信号波形从 $0.1V_m$ 上升到 $0.9V_m$ 所经历的时间。

脉冲下降时间 t_f : 脉冲信号波形从 $0.9V_m$ 下降到 $0.1V_m$ 所经历的时间。脉冲上升时间 t_r 和脉冲下降时间 t_f 越短,越接近于理想的矩形脉冲,单位为秒(s)、毫秒(ms)、微妙(μs)、纳秒(ns)等。

脉冲宽度 t_w : 由脉冲信号波形上升沿 $0.5V_m$ 到下降沿 $0.5V_m$ 之间的时间间隔,单位与 t_r 、 t_f 相同。

脉冲周期 T : 在周期性脉冲信号中, 任意两个相邻脉冲的上升沿(或下降沿)同一数值点之间的时间间隔, 单位与 t_r 、 t_f 相同。

脉冲频率 f : 单位时间内(每秒)出现的脉冲波形个数, 单位为赫兹(Hz)、千赫兹(kHz)、兆赫兹(MHz)等, 脉冲频率 $f=1/T$ 。

占空比 q : 表示脉冲宽度与整个周期的百分比, $q=\frac{t_w}{T}\times 100\%$ 。把占空比为 50% 的矩形脉冲信号称为方波信号。

1.1.2 数字电路的特点和分类

1. 数字电路的特点

(1) 电路结构简单, 容易制造, 便于集成、系列化生产, 成本低廉, 使用方便。由于数字电路是以二值数字逻辑为基础的, 只有 0 和 1 两个基本数字, 易于用电路来实现, 比如可用二极管、三极管的导通与截止这两个对立的状态来表示数字信号的逻辑 0 和逻辑 1。

(2) 由数字电路组成的数字系统, 工作准确可靠, 精度高, 抗干扰能力强。它可以通过整形很方便地去除叠加于传输信号上的噪声与干扰, 还可利用差错控制技术对传输信号进行查错和纠错。

(3) 数字电路不仅能完成数值运算, 而且能进行逻辑判断和运算, 这在控制系统中得到广泛应用。

(4) 数字信息便于长期保存, 比如可将数字信息存入磁盘、光盘等介质, 以便长期保存。

(5) 数字集成电路产品系列多、通用性强、成本低。

2. 数字电路的分类

数字电路可以按照电路组成结构、组成器件、逻辑功能等进行分类。

(1) 按电路组成结构分类

数字电路按其电路的组成结构分类, 可分为分立元件电路和集成电路两类。分立元件电路由二极管、三极管、电阻、电容等元件组成。集成电路则通过半导体制造工艺将这些元件做在一片芯片上。随着集成电路技术的不断发展, 具有体积小、重量轻、功耗小、价格低、可靠性高等特点的集成电路会逐步代替体积大、可靠性不高的分立电路。集成电路按集成度(在一块硅片上包含的逻辑门电路或组件的数量)分为: 小规模(SSI)、中规模(MSI)、大规模(LSI)、超大规模(VLSI)和甚大规模(ULSI)集成电路。如图 1.1.4 和表 1.1.2 所示。



集成电路

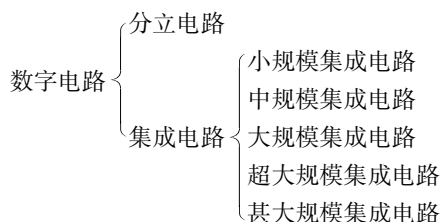


图 1.1.4 按电路组成结构分类图

表 1.1.2 数字集成电路的分类

分类	门的个数	典型集成电路
小规模	1~10	逻辑门、触发器
中规模	11~100	计数器、加法器
大规模	101~10 000	小型存储器、门阵列
超大规模	10 001~100 000	大型存储器、微处理器
甚大规模	100 001 以上	可编程逻辑器件、多功能专用集成电路

(2) 按所用器件分类

数字电路按电路所用器件分类,可以分为双极型和单极型电路。双极型电路即 TTL 型,是晶体管-晶体管逻辑门电路的简称,主要由双极型三极管组成。TTL 集成电路生产工艺成熟,产品参数稳定,工作可靠,开关速度高;单极型电路即 MOS 型,是金属-氧化物-半导体场效应管门电路的简称,主要由场效应管组成,优点是低功耗,抗干扰能力强。

(3) 按逻辑功能分类

数字电路按逻辑功能分类,可分为组合逻辑电路和时序逻辑电路。如果一个逻辑电路在任何时刻的输出状态只取决于当时的输入状态,与电路原来的状态无关,则该电路称为组合逻辑电路。如果在任一时刻,电路的输出状态不仅取决于当时的输入状态,还与前一时刻的状态有关,则该电路称为时序电路。

1.2 数制

数制是指计数的方法,是用一组固定的符号和统一的规则来表示数值的方法。在日常生活中,人们习惯于使用十进制,可是在数字系统中,为了便于电路的实现常使用二进制,但位数较多的二进制数不便于书写和阅读,于是,人们有时也使用八进制数或十六进制数。

1.2.1 几种常用的数制

1. 十进制

人类的祖先在长期的生产劳动实践中学会了用十个指头计数,因而产生了我们最熟悉的十进制数。任意一个十进制数 $(S)_{10}$ 可以表示为

$$(S)_{10} = k_{n-1}10^{n-1} + k_{n-2}10^{n-2} + \cdots + k_010^0 + k_{-1}10^{-1} + \cdots + k_{-m}10^{-m} = \sum_{i=-m}^{n-1} k_i 10^i \quad (1.2.1)$$

式中: k_i 可以是 0~9 十个数码中的任意一个, m 和 n 是正整数, 10^i 表示第 i 位的位权(简称权); k_i, m, n 均由 $(S)_{10}$ 决定, (S) 的下标与式中的 10 是十进制的基数。由于基数为 10,每个数位计满 10 就向高位进位,即逢十进一,所以称它为十进制计数制。如果用 R 表示 R 进制的基数,则式(1.2.1)可以写成通式

$$(S)_R = \sum_{i=-m}^{n-1} k_i R^i \quad (1.2.2)$$

【例 1.2.1】 将 $(123.45)_{10}$ 写成权表示的形式。

解 $(123.45)_{10} = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2}$

2. 二进制

在数字系统中,为了便于工程实现,广泛采用二进制计数。这是因为二进制表示的数,每一位只取数码 0 或 1,因而可以用具有两个不同状态的电子元件来表示,并且数据的存储和传送也可用简单而可靠的方式进行。二进制的基数是 2,它是以 2 为基数的计数体制,其计数规律是逢二进一。

任意一个二进制数可以表示成

$$(S)_2 = \sum_{i=-m}^{n-1} k_i 2^i \quad (1.2.3)$$

式中: k_i 只能取 0 或 1; m, n 为正整数,它们由 $(S)_2$ 决定; 2^i 表示第 i 位的权。

【例 1.2.2】 将 $(1\ 011.01)_2$ 写成权表示的形式。

解 $(1\ 011.01)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$

3. 八进制

任意一个八进制数可以表示成

$$(S)_8 = \sum_{i=-m}^{n-1} k_i 8^i \quad (1.2.4)$$

式中: k_i 可取 0~7 八个数码之一; m 和 n 为正整数,它们由 $(S)_8$ 决定; 8^i 表示第 i 位的权。它是以 8 为基数的计数体制,其计数规律为逢八进一。

【例 1.2.3】 将八进制数 $(69.731)_8$ 写成权表示的形式。

解 $(69.731)_8 = 6 \times 8^{-1} + 9 \times 8^0 + 7 \times 8^{-1} + 3 \times 8^{-2} + 1 \times 8^{-3}$

4. 十六进制

任意一个十六进制数可以表示成

$$(S)_{16} = \sum_{i=-m}^{n-1} k_i 16^i \quad (1.2.5)$$

式中: k_i 可取 0, 1, 2, ..., 9, A(10), B(11), C(12), D(13), E(14), F(15)这十六个个数码之一; m 和 n 为正整数,它们由 $(S)_{16}$ 决定; 16^i 表示第 i 位的权。它是以 16 为基数的计数体制,其计数规律为逢十六进一。

【例 1.2.4】 将十六进制数 $(8BE6)_{16}$ 写成权表示的形式。

解 $(8BE6)_{16} = 8 \times 16^3 + B \times 16^2 + E \times 16^1 + 6 \times 16^0$

十进制、二进制、八进制和十六进制常用英文缩写表示,分别为 D(Decimal)、B(Binary)、O(Octal)、H(Hexadecimal),如 $(41)_D$ 表示十进制数 41。另外,还可以在一个数的后面加上上述英文的缩写表示是几进制数,如 41H 表示是十六进制数 41。

1.2.2 不同数制间的转换

将数由一种数制转换成另一种数制称为数制间的转换。因为日常生活中经常使用的是十进制数,而在数字系统中采用的是二进制数,所以就需要把输入的十进制数换算成计算机所能够接受的二进制数。数字系统在处理结束后,可以再把二进制数换算成人们所习惯的十进制数输出。

1. 其他进制数转换为十进制数

转换方法为将其他进制数按权展开,然后各项相加,就得到相应的十进制数。

【例 1.2.5】 将 $(10110.101)_2$ 和 $(8ED.C7)_{16}$ 转换为十进制数。

解 $(10110.101)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$
 $= 16 + 4 + 2 + 0.5 + 0.125 = (22.625)_{10}$

$(8ED.C7)_{16} = 8 \times 16^2 + 14 \times 16^1 + 13 \times 16^0 + 12 \times 16^{-1} + 7 \times 16^{-2} = (2285.7773)_{10}$

2. 十进制数转换为其他进制数

十进制的整数部分与小数部分分别转换。

整数部分采用“除基取余法”:整数部分逐次除以基数,依次记下余数,直至商为0。读数方向:从下到上。

小数部分采用“乘基取整法”:小数部分连续乘以基数,依次取整数,直至小数部分为0,或达到要求的精度。读数方向:从上到下。

【例 1.2.6】 将十进制数 37.48 转换成二进制数、八进制数,小数点后保留三位。

解 $(37.48)_{10} = (37)_{10} + (0.48)_{10}$

(1) 十进制数 37.48 转换成二进制数。

整数部分:

2	37	余数	
2	18	1	↑ 低位
2	9	0	
2	4	1	
2	2	0	
2	1	0	
	0	1	↓ 高位
$(37)_{10} = (100101)_2$			

小数部分:

0.48	整数	
× 2		↑ 高位
0.96	0	
0.96		↓ 低位
× 2	1	
1.92		
0.92		
× 2	1	
1.84		

$$(0.48)_{10} = (0.011)_2$$

所以, $(37.48)_{10} = (100101.011)_2$ 。

(2) 十进制数 37.48 转换成八进制数。

整数部分:

8	37	余数	↑ 低位
8	4	5	
	0	4	↑ 高位

$$(37)_{10} = (45)_8$$

小数部分:

0.48	整数		高位
× 8			
3.84	3		
0.84			
× 8			
6.72	6		
0.72			
× 8			
5.76	5		↓ 低位

$$(0.48)_{10} = (0.365)_8$$

所以, $(37.48)_{10} = (45.365)_8$ 。

3. 二进制与八进制、十六进制间的转换

(1) 二进制与八进制的转换

规则: 每三位二进制数相当于一位八进制数。

二进制数转换为八进制数: 以小数点为中心, 分别向左、右两边延伸, 每三位二进制数为一组, 用对应的八进制数来表示; 不足三位的, 用 0 补足。

八进制转换为二进制: 每位八进制数用三位二进制数来代替, 去掉多余的 0 (最前面和最后面的 0)。

(2) 二进制与十六进制的转换 (类似于二进制与八进制的转换)

规则: 每四位二进制数相当于一位十六进制数。

二进制数转换为十六进制数: 以小数点为中心, 分别向左、右划分延伸, 每四位二进制数用一位十六进制数来表示; 不足四位的, 用 0 补足。

十六进制转换为二进制: 每位十六进制数用四位二进制数来代替, 去掉多余的 0 (最前面和最后面的 0)。

【例 1.2.7】 将二进制数 1001101.010 转换为八进制数和十六进制数。

解 二进制 001 001 101.010

八进制 1 1 5. 2

所以, $(1001101.010)_2 = (115.2)_8$ 。

二进制 0100 1101.0100

十六进制 4 D. 4
所以, $(1001101.010)_2 = (4D.4)_{16}$ 。

【例 1.2.8】 将八进制数 $(752.36)_8$ 和十六进制数 $(4F8E.9C)_{16}$ 分别转换为二进制数。

解 八进制数 7 5 2. 3 6
 二进制数 111 101 010. 011 110
所以, $(752.36)_8 = (111101010.011110)_2$ 。

十六进制数 4 F 8 E. 9 C
 二进制数 0100 1111 1000 1110. 1001 1100
所以, $(4F8E.9C)_{16} = (0100111110001110.10011100)_2$ 。

1.3 二进制代码

数字系统可以处理文字、声音、图像、字符等信息。然而,由于在数字系统中采用二进制,所以这些信息在数字系统中必须以二进制代码的形式表示。将若干个二进制数码 0 或 1 按一定规则排列起来表示某种特定含义的代码,称之为二进制代码,或称为二进制码。将编制这类代码所遵循的规则称为码制。用 n 位二进制数可以表示 2^n 个不同的信息,给每一个信息规定一个具体的二进制代码,这个过程称为编码。下面介绍几种数字电路中常用的二进制代码。

1.3.1 二-十进制码

把 1 位十进制数中的 0~9 十个数字用 4 位二进制数表示的代码,称为二-十进制码或 BCD 码(Binary-Coded Decimal)。BCD 编码方法有很多种方案。表 1.3.1 中列出了常见的几种 BCD 码。

表 1.3.1 常用 BCD 码

十进制数	8421 码	2421 码	5421 码	余 3 码
0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 0 0 1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 0 1 0	0 0 1 0	0 1 0 1
3	0 0 1 1	0 0 1 1	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 0 0	0 1 0 0	0 1 1 1
5	0 1 0 1	1 0 1 1	1 0 0 0	1 0 0 0
6	0 1 1 0	1 1 0 0	1 0 0 1	1 0 0 1
7	0 1 1 1	1 1 0 1	1 0 1 0	1 0 1 0
8	1 0 0 0	1 1 1 0	1 0 1 1	1 0 1 1
9	1 0 0 1	1 1 1 1	1 1 0 0	1 1 0 0
位权	8 4 2 1 $b_3 b_2 b_1 b_0$	2 4 2 1 $b_3 b_2 b_1 b_0$	5 4 2 1 $b_3 b_2 b_1 b_0$	无权

1. 8421BCD 码

采用 4 位二进制数表示 1 位十进制数,即每一位十进制数用四位二进制表示。这是一种恒权码(每一位的位权是固定不变的),4 位二进制数各位权由高到低分别是 2^3 、 2^2 、 2^1 、 2^0 ,即 8、4、2、1。这种编码最自然,最简单,且书写方便、直观、易于识别。每组二进制代码按权展开求和就得到所代表的十进制数,它是最常用的 BCD 码。

【例 1.3.1】 将十进制数 34.15 转换为 8421 码。

解

3	4	.	1	5
0011	0100	.0001	0101	

$(34.15)_{10} = (110100.00010101)_{8421BCD}$

【例 1.3.2】 将 8421 码(100101100011)_{8421BCD}转换为十进制数。

解

1001	0110	0011
9	6	3

$(100101100011)_{8421BCD} = (963)_{10}$

2. 5421BCD 码

恒权码,从高位到低位的位权分别为 5、4、2、1。

3. 2421BCD 码

恒权码,从高位到低位的位权分别为 2、4、2、1。

4. 余 3 码

无权码(没有固定的位权,比 8421 码多余 3。余 3 码的特点是将 4 位二进制数的 16 种状态前后各去掉 3 种状态,剩下 10 种状态表示 0~9。即 0011(0)、0100(1)、0101(2)、…、1100(9),是一种无权码。当用二进制规则进行余 3 码加法时,当和的十进制值 ≤ 9 时,需减 3 来修正(因两个数相加余 6,而和只余 3);当和的十进制值 >9 时(即有进位),需加 3 来修正。

【例 1.3.3】 将 8421 码(0000)_{8421BCD}、(0001)_{8421BCD}、(0010)_{8421BCD}转换为余 3 码。

解

$(0000)_{8421BCD} = (0011)_{\text{余 3 码}};$
 $(0001)_{8421BCD} = (0100)_{\text{余 3 码}};$
 $(0010)_{8421BCD} = (0101)_{\text{余 3 码}}。$

1.3.2 格雷码

格雷码(Gray)也是一种常用的无权码,其特点是任意两组相邻代码之间只有一位不同,是无权码。格雷码常用于模拟量的转换中,当模拟量发生微小变化而可能引起数字量发生变化时,格雷码仅改变 1 位,这样与其他码同时改变两位或多位的情况相比更为可靠,可减少出错的可能性,典型的 4 位格雷码的编码如表 1.3.2 所示,表中同时给出了 4 位自然二进制码。

表 1.3.2 4 位格雷码

十进制数	二进制码	格雷码	十进制数	二进制码	格雷码
0	0 0 0 0	0 0 0 0	8	1 0 0 0	1 1 0 0
1	0 0 0 1	0 0 0 1	9	1 0 0 1	1 1 0 1
2	0 0 1 0	0 0 1 1	10	1 0 1 0	1 1 1 1
3	0 0 1 1	0 0 1 0	11	1 0 1 1	1 1 1 0
4	0 1 0 0	0 1 1 0	12	1 1 0 0	1 0 1 0
5	0 1 0 1	0 1 1 1	13	1 1 0 1	1 0 1 1
6	0 1 1 0	0 1 0 1	14	1 1 1 0	1 0 0 1
7	0 1 1 1	0 1 0 0	15	1 1 1 1	1 0 0 0

【例 1.3.4】 用格雷码表示十进制数 1、2。

解 (1)₁₀=(0001)_{格雷码}；
(2)₁₀=(0011)_{格雷码}。

1.3.3 ASCII 码

美国信息交换标准代码(American Standard Code for Information Interchange,简称 ASCII 码)是通信和计算机系统中使用得最广泛的一种代码。ASCII 码虽然是美国国家标准,但它已被国际标准化组织(ISO)认定为国际标准。ASCII 码已为世界公认,并在世界范围内通用。ASCII 码有 7 位版本和 8 位版本两种。国际上通用的是 7 位版本。7 位版本的 ASCII 码有 128 个元素,其中通用控制字符 34 个,阿拉伯数字 10 个,大、小写英文字母 52 个,各种标点符号和运算符号 32 个,表 1.3.3 为 ASCII 码的编码表。

表 1.3.3 美国信息交换标准代码(ASCII 码)

$b_4b_3b_2b_1$	$b_7b_6b_5$							
	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	“	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x

(续表)

$b_4 b_3 b_2 b_1$	$b_7 b_6 b_5$							
	000	001	010	011	100	101	110	111
1001	HT	EM	)	9	I	Y	I	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	—	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	—	o	DEL

比如：“A”的 ASCII 码值为 1000001(即 41H)，“a”的 ASCII 码值为 1100001(即 61H)，
“0”的 ASCII 码值为 0110000(即 30H)，“@”的 ASCII 码值为 1000000(即 40H)。

1.4 二进制数的算术运算

1.4.1 无符号二进制数的算术运算

在数字系统中,当两个二进制数仅表示数量的大小而不考虑其正负时,即所有位都用来表示数值,称它们为无符号数。无符号二进制数可以进行算术运算,这种运算称为无符号二进制数的算术运算。其运算规则与十进制数基本相同,差别仅在于进(借)位规则不同。在二进制数进行运算中,进位时“逢二进一”,借位是“借一当二”。无符号二进制数的基本算术运算包括加法、减法、乘法和除法。

1. 加法运算

(1) 加法进位规则:逢二进一。

(2) 二进制加法运算规则:

① $0+0=0$

② $0+1=1+0=1$

③ $1+1=10$ (向高位进位)

【例 1.4.1】求两个二进制数 $(1011011)_2$ 和 $(1010.01)_2$ 的和。

解

被加数

1011011

+

加数

1010.01

1100101.01

所以, $(1011011)_2+(1010.01)_2=(1100101.01)_2$ 。

2. 减法运算

(1) 减法借位规则:借一当二。

(2) 二进制减法运算法则:

① $0-0=1-1=0$

② $1-0=1$

③ $0-1=1$ (借一当二)

【例 1.4.2】 求两个二进制数 $(1110110)_2$ 和 $(1101.11)_2$ 的差。

解	被减数	1110110
	— 减数	1101.11
		1101000.01

所以, $(1110110)_2 - (1101.11)_2 = (1101000.01)_2$ 。

3. 乘法运算

二进制乘法运算规则:

(1) $0 \times 0 = 0$

(2) $0 \times 1 = 1 \times 0 = 0$

(3) $1 \times 1 = 1$

【例 1.4.3】 两个二进制数 $(1110)_2$ 和 $(1101)_2$ 的乘积。

解	被乘数	1110
	× 乘数	1101
	部分积	1110
	左移一位	0000
	左移一位	1110
	左移一位	1110
	乘积	10110110

所以, $(1110)_2 \times (1101)_2 = (10110110)_2$ 。

从执行乘法的过程可见,从乘数的低位开始每一位与被乘数相乘得到一个部分积,乘数的相应位是 0 时,部分积为 0,乘数的相应位是 1 时,部分积为被乘数,每一次的部分积均依次左移一位,部分积的个数是乘数的位数,将各部分积累加起来就得到最终乘积,即二进制乘法运算可归结为“加法与移位”。

4. 除法运算

二进制除法运算法则:

(1) $0 \div 0 = 0$

(2) $0 \div 1 = 0$

(3) $1 \div 1 = 1$

【例 1.4.4】 求两个二进制数 $(100110)_2$ 和 $(110)_2$ 之商。

解

$$\begin{array}{r}
 \text{除数} \cdots \cdots 110 \overline{) 100110 \cdots \cdots \text{被除数}} \\
 \underline{110} \\
 111 \\
 \underline{110} \\
 10 \cdots \cdots \text{余数} \quad (\text{误差以精度要求而定})
 \end{array}$$

所以, $(100110)_2 \div (110)_2 = (110)_2$, 余数 10。

可见, 二进制除法运算可归结为“减法与移位”。

1.4.2 有符号二进制数的算术运算

实际的二进制数与十进制数一样, 不仅有大小, 而且有正负, 这种带有正负号的二进制数称为有符号的二进制数。在数字系统中, 数的正负号也需要进行数值化处理。

1. 机器数与真值

数学中正数与负数是用该数的绝对值, 加上正、负符号来表示。由于数字系统中无论是数值还是数的符号, 都只能用 0 和 1 来表示。所以数字系统中, 为了表示正、负数, 把一个数的最高位作为符号位: 0 表示正数, 1 表示负数。比如, 用八个二进制位表示一个十进制数, 则正的 36 和负的 36 可表示为:

+36: 00100100;

-36: 10100100。

这种连同符号位一起数字化的数称为机器数, 而由机器数所表示的实际值称为真值。比如, 机器数 00101011 的真值为十进制的 +43 或二进制的 +0101011, 而机器数 1010011 的真值为十进制的 -43 或二进制的 -0101011。

2. 机器数的表示方法

(1) 原码: 正数的符号位用 0 表示, 负数的符号位用 1 表示, 数值部分用二进制形式表示, 称为该数的原码。

比如: $X = +81$, 则 $(X)_{\text{原}} = 01010001$; $Y = -81$, 则 $(Y)_{\text{原}} = 11010001$ 。

用原码表示一个数简单、直观、方便。但不能用它对两个同号数相减或两个异号数相加。

比如: 将十进制数“+36”与“-45”的原码直接相加:

$X = +36$, $(X)_{\text{原}} = 00100100$;

$Y = -45$, $(Y)_{\text{原}} = 10101101$ 。

$00100100 + 10101101 = 11010001$ 。

所得结果是一 81, 这显然是不对的, 这个问题可以用后面介绍的补码运算来解决。

(2) 反码: 正数的反码和原码相同, 负数的反码是对该数的原码除符号位外各位取反, 即“0”变“1”, “1”变“0”。

例如: $X = +81$, $Y = -81$ 。

$$(X)_{\text{原}} = 01010001, (X)_{\text{反}} = 01010001;$$

$$(Y)_{\text{原}} = 11010001, (Y)_{\text{反}} = 10101110。$$

(3) 补码:正数的补码与原码相同,负数的补码是对该数的原码除符号外各位取反,然后加1,即反码加1。

比如: $X=+81, Y=-81$,则有:

$$(X)_{\text{原}} = (X)_{\text{反}} = (X)_{\text{补}} = 01010001;$$

$$(Y)_{\text{原}} = 11010001;$$

$$(Y)_{\text{反}} = 10101110;$$

$$(Y)_{\text{补}} = 10101111。$$

3. 二进制数补码运算

设 S_1, S_2 均为 n 位正数,则二进制数的加、减运算补码运算的规则如下:

$$(1) (S_1 + S_2)_{\text{补}} = (S_1)_{\text{补}} + (S_2)_{\text{补}}$$

$$(2) (S_1 - S_2)_{\text{补}} = (S_1)_{\text{补}} + (-S_2)_{\text{补}}$$

$$(3) (-S_1 + S_2)_{\text{补}} = (-S_1)_{\text{补}} + (S_2)_{\text{补}}$$

$$(4) (-S_1 - S_2)_{\text{补}} = (-S_1)_{\text{补}} + (-S_2)_{\text{补}}$$

进行二进制补码运算时,必须注意是,参与运算的是补码,运算结果仍然是补码,要得到运算结果,还需要对运算结果再求一次补码,得到原码。

【例 1.4.5】 试用二进制补码求 $(37)_{10} - (39)_{10}$ 的值。

$$\begin{aligned} \text{解 } [(37)_{10} - (39)_{10}]_{\text{补}} &= [(37)_{10}]_{\text{补}} + [(-39)_{10}]_{\text{补}} = 0100101 + 1011001 \\ &= 1111110 \end{aligned}$$

即:

十进制数	原码	反码	补码
$(+37)_{10}$	0100101	0100101	0100101
$+(-39)_{10}$	1100111	1011000	<u>+1011001</u>
?			1111110

对补码(1111110)再求一次补码,得到原码(1000010)。

将原码(1000010)转化为十进制数,运算结果为-2。

所以, $(37)_{10} - (39)_{10} = (-2)_{10}$ 。

本章小结

数字信号在时间上和数值上均是离散的。对数字信号进行传送、加工和处理的电路称为数字电路。由于数字电路是以二值数字逻辑为基础的,即利用数字1和0来表示信息,因此,数字信息的存储、分析和传输要比模拟信息容易。

数字电路中用高电平和低电平分别来表示逻辑1和逻辑0,它和二进制数中的0和1正好对应。因此,数字系统中常用二进制数来表示数据。在二进制位数较多时,常用十六进制或八进制作为二进制的简写。

数制是用一组固定的符号和统一的规则来进行计数的方法。常用的数制有十进制、二进制、八进制和十六进制,各种进制之间可以互相转换。

常用的二进制代码主要有8421码、2421码、5421码、余3码和格雷码等,其中8421码使用最广泛。另外,格雷码由于可靠性高,也是一种常用码。

与十进制数算术运算类似,二进制数也有加、减、乘、除四种基本算术运算,加法是各种运算的基础。在进行有符号二进制数的算术运算时,常用原码、反码和补码来表示有符号数,采用补码可以进行有符号数的算术运算。

自我检测题

一、填空题

1. 电子电路中,被传递、加工和处理的信号有两类:一类是幅度随时间连续变化的信号,称为_____ ;另一类是在时间和幅度上都是离散的信号,称为_____。
2. 在数字系统中,由于使用二进制,为了表示数值、文字、符号等信息,一般采用一定位数的二进制数码按一定规则排列起来表示特定的含义,这种具有特定含义的二进制数码称为_____。
3. 二进制数 $(11.375)_{10}$ 等值的二进制数为_____。
4. 十进制数 $(56)_{10}$ 的8421BCD码是_____,等值的二进制数是_____。
5. 与二进制数 $(110.101)_2$ 等值的十进制数是_____。
6. 把十进制数103表示为十六进制数是_____。
7. $-(1011)_2$ 的原码、反码和补码分别是_____,_____,_____。
8. 采用补码运算 $(0110)_2 - (1101)_2$ 的运算结果其补码和原码分别是_____,_____。

二、选择题

1. 与模拟信号相比,数字信号具有_____。
A. 连续性 B. 数字性 C. 离散性 D. 对偶性
2. $(36.7)_{10}$ 的8421BCD码是_____。
A. 0110110.101 B. 00111110.1110 C. 00110110.0111 D. 1101110.111
3. 5421BCD码中表示十进制数9的编码是_____。
A. 1010 B. 1001 C. 1100 D. 1101
4. 字符X的ASCII码的十六进制数表示为_____。
A. 58H B. 48H C. 59H D. 4FH
5. 8位二进制补码 $(11111111)_2$ 所对应的十进制数的真实值是_____。
A. -1 B. 127 C. 256 D. 255
6. 十进制数-3用8位二进制补码表示,写成十六进制数为_____。
A. 03 B. FC C. FD D. 83
7. 与二进制数 $(0110.1001)_2$ 等值的八进制数为_____。

- A. 6.44 B. 6.41 C. 3.44 D. 3.41
8. 十进制数 27 若用余 3BCD 码表示,可写成_____。
- A. 01011010 B. 00100111 C. 00101010 D. 01010111

练习题

- 1.1 将下列二进制数转换为十进制数。
- (1) $(10101)_2$ (2) $(0.10101)_2$ (3) $(1010.101)_2$
- 1.2 写出下列八进制数的按权展开式。
- (1) $(247)_8$ (2) $(0.651)_8$ (3) $(465.43)_8$
- 1.3 将下列十六进制数转换为十进制数。
- (1) $(6BD)_{16}$ (2) $(0.7A)_{16}$ (3) $(8E.D)_{16}$
- 1.4 将下列十进制数转换为二进制数、八进制数和十六进制数(要求转换误差不大于 2^{-4})。
- (1) $(43)_{10}$ (2) $(2.718)_{10}$ (3) $(4.8)_{10}$
- 1.5 将下列二进制数转换为八进制数。
- (1) $(10111101)_2$ (2) $(0.11011)_2$ (3) $(1101011.1101)_2$
- 1.6 将下列二进制数转换为十六进制数。
- (1) $(1101111011)_2$ (2) $(0.10111)_2$ (3) $(110111.01111)_2$
- 1.7 将下列十进制数用 8421BCD 码表示。
- (1) $(2.618)_{10}$ (2) $(23.56)_{10}$ (3) $(127)_{10}$
- 1.8 将下列 8421BCD 码转化成等值的十进制数。
- (1) 10011000 (2) 10010111 (3) 01111001, 10000101 (4) 00110110, 0001
- 1.9 写出下列数字、字母或字符的 ASCII 码。
- (1) 7 (2) T (3) @
- 1.10 写出下列数的真值、原码、反码和补码(用八位二值数码表示)。
- (1) $(+127)_{10}$ (2) $(-88)_{10}$ (3) $(-101)_{10}$
- 1.11 写出下列有符号二进制补码所表示的十进制数。
- (1) 00101101 (2) 11101001 (3) 01011101 (4) 11011101
- 1.12 试用二进制补码计算下列各式的值。
- (1) $(14)_{10} - (16)_{10}$ (2) $(54)_{10} - (30)_{10}$ (3) $(-55)_{10} - (29)_{10}$

第 2 章 逻辑代数基础

本章学习目的和要求

1. 正确理解基本逻辑运算和复合逻辑运算；
2. 熟悉逻辑代数中的基本定律、常用公式和基本规则；
3. 掌握逻辑函数的概念和常用表示方法及其相互转换，了解逻辑函数的标准形式；
4. 掌握逻辑函数的变换和公式化简法；
5. 掌握逻辑函数的卡诺图化简法，会用卡诺图化简具有无关项的逻辑函数。

本章介绍分析和设计数字电路的数学工具——逻辑代数。首先介绍逻辑代数中的基本逻辑运算和复合逻辑运算，接着讲述逻辑代数中的基本定律、常用公式和三个基本规则，然后讲述逻辑函数及其表示方法，最后介绍如何应用这些定律、公式和规则来化简逻辑函数。

2.1 概 述

在数字逻辑电路中，可以用 0 和 1 组成的二进制数表示数量的大小，也可以用 0 和 1 表示事物两种不同的逻辑状态。例如，可以用 1 和 0 分别表示一件事情的是和非、真和假、有和无、好和坏，或者表示电路的通和断、电灯的亮和暗、门的开和关等。这里的 0 和 1 不是数值，而是逻辑 0 和逻辑 1 这两种对立的逻辑状态。这种只有两种对立逻辑状态的逻辑关系称为二值数字逻辑，简称数字逻辑。

这里所说的“逻辑”，是指事物之间的因果关系。当 0 和 1 两个二进制数码表示不同的逻辑状态时，它们之间可以按照指定的某种因果关系进行推理运算，这种运算称为逻辑运算。

逻辑运算使用的数学工具是逻辑代数，是 19 世纪中叶英国数学家乔治·布尔(George Boole)创立的一门研究客观事物逻辑关系的代数，因此，逻辑代数又称为布尔代数，后来被广泛应用于解决数字逻辑电路的分析与设计中。逻辑代数的有些运算公式在形式上和普通代数的运算公式相似，但是两者所包含的物理意义有本质的区别。逻辑代数中也用字母表示变量，这种变量称为逻辑变量。逻辑运算表示的是逻辑变量以及常量之间逻辑状态的推理运算，而不是数量之间的运算。虽然在二值逻辑中，每个变量的取值只有 0 和 1 两种可能，只能表示两种不同的逻辑状态，但是我们可以用多变量的不同状态组合表示事物的多种逻辑状态，处理复杂的逻辑问题。



布尔代数

2.2 逻辑代数中的常用运算

2.2.1 基本逻辑运算

逻辑代数中的变量称为逻辑变量,一般用大写英文字母 A,B,C,⋯来表示。逻辑变量只有两种取值,常用 0 和 1 来表示。这里的 0 和 1 不表示数量,也没有大小的意义,而只代表两种对立的状态。

逻辑代数的基本运算有与(AND)、或(OR)、非(NOT)三种。

1. 与运算(逻辑乘)

图 2.2.1 所示为一个简单的与逻辑电路,电压 E 通过开关 A 和 B 向灯泡 L 供电,只有开关 A 与 B 同时闭合时,灯泡 L 才亮。A 和 B 中只要有一个断开或者两个都断开时,灯泡 L 不亮。与逻辑的逻辑状态如表 2.2.1 所示。这个电路中开关 A、B 与灯泡 L 的逻辑关系是:“只有当一件事的几个条件全部具备之后,这件事才发生”,这种关系称为与逻辑。如果用 0 和 1 来表示开关和灯的状态,设开关断开和灯不亮均用 0 表示,而开关闭合和灯亮均用 1 表示,则可得出如表 2.2.2 所示形式的逻辑真值表(描述逻辑关系的表格),后面将详述,其中 L 表示灯的状态。

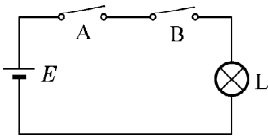


图 2.2.1 与逻辑电路举例

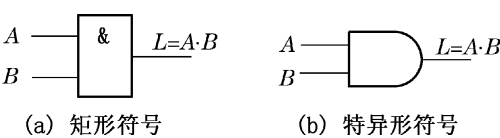


图 2.2.2 与门逻辑符号

表 2.2.1 与逻辑状态表

A	B	L
断开	断开	不亮
断开	闭合	不亮
闭合	断开	不亮
闭合	闭合	亮

表 2.2.2 与逻辑真值表

A	B	L
0	0	0
0	1	0
1	0	0
1	1	1

用逻辑表达式来描述与逻辑,则可写为

$$L=A \cdot B \tag{2.2.1}$$

式中小圆点“ $\cdot$ ”表示 A、B 的与运算,也称为逻辑乘。在不致引起混淆的前提下,乘号“ $\cdot$ ”可省略,写成“ $L=AB$ ”。与运算的规则: $0 \cdot 0=0$; $0 \cdot 1=0$; $1 \cdot 0=0$; $1 \cdot 1=1$ 。

在数字电路中能实现与逻辑运算的电路为与门电路,简称与门,其逻辑符号如图 2.2.2 所示,其中图 2.2.2(a)为矩形符号,图 2.2.2(b)为特异形符号。与运算可以推广到多个变量,其相应的逻辑表达式为

$$L=A \cdot B \cdot C \cdot \cdots$$

(2.2.2)

2. 或运算(逻辑加)

图 2.2.3 所示为一个简单的或逻辑电路,电压 E 通过开关 A 或 B 向灯泡供电。只要开关 A 或 B 闭合或两者均闭合,则灯亮。而当 A 和 B 均断开时,则灯不亮。此电路中,开关 A 、 B 与灯 L 的逻辑关系是:“当一件事情的几个条件中只要有一个条件得到满足,这件事就会发生”,这种关系称为或逻辑。或是指 A 闭合或 B 闭合,即任一个条件具备的意思。仿照前述,可以得出如表 2.2.3 所示的或逻辑电路状态表和如表 2.2.4 所示的用 0、1 表示的或逻辑真值表。

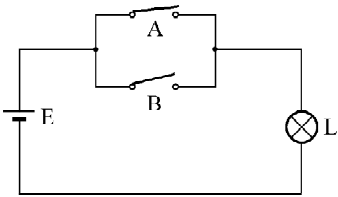


图 2.2.3 或逻辑电路举例

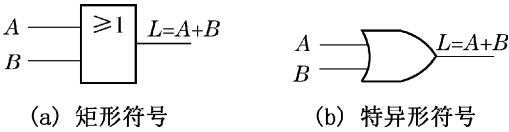


图 2.2.4 或门逻辑符号

表 2.2.3 或逻辑状态表

A	B	L
断开	断开	不亮
断开	闭合	亮
闭合	断开	亮
闭合	闭合	亮

表 2.2.4 或逻辑真值表

A	B	L
0	0	0
0	1	1
1	0	1
1	1	1

用逻辑表达式来描述,则可写为

$$L=A+B$$

(2.2.3)

式中:符号“+”表示 A 、 B 或运算,也称为逻辑加。或运算的规则: $0+0=0$; $0+1=1$; $1+0=1$; $1+1=1$ 。

能实现或运算的电路称为或门电路,简称或门,其逻辑符号如图 2.2.4 所示,其中图 2.2.4(a)为矩形符号,图 2.2.4(b)为特异形符号。或运算同样也可以推广到多个变量,其相应的逻辑表达式为

$$L=A+B+C+\cdots$$

(2.2.4)

3. 非运算(逻辑反)

非运算(逻辑反)是逻辑的否定,当条件具备时,结果不会发生;而条件不具备时,结果一定会发生。如图 2.2.5 所示,电压 E 通过 R 向灯泡供电,开关 A 闭合时,灯 L 不亮;开关 A 断开时,灯 L 才会亮。其逻辑关系就是“一件事情的发生是与其相反的条件为依据”。若用 0 和 1 来表示开关 A 和灯泡 L 的状态, A 断开和灯不亮用 0 表示,而 A 闭合和灯亮用 1 表示,可得出如表 2.2.5 所示的非逻辑真值表。显而易见,对于二值逻辑 0 和 1,有“ $\overline{0}=1$, $\overline{1}=0$ ”。很容易看到, L 与 A 总是处于相反的逻辑状态。

用逻辑表达式来描述,则可写为

$$L=\overline{A}$$

(2.2.5)

式中字母 A 上方的短横线表示非运算,读作“非”或“反”。在逻辑运算中,通常将 A 称为原变量,而将 $\overline{A}$ 称为反变量或非变量。

能实现非运算的电路称为非门电路,简称非门,也称反相器,其逻辑符号如图 2.2.6 所示,小圆圈表示非运算。

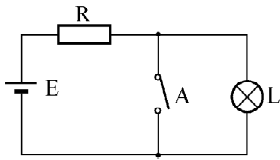


图 2.2.5 非逻辑电路举例

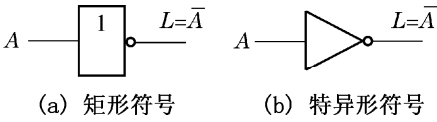


图 2.2.6 非门逻辑符号

表 2.2.5 非逻辑真值表

A	L
0	1
1	0

2.2.2 复合逻辑运算

在实际应用中,除了与、或、非三种基本逻辑运算以外,还广泛应用与、或、非的不同组合的复合逻辑运算,最常见的复合逻辑运算有与非、或非、与或非、异或和同等。

1. 与非运算

“与”与“非”的复合运算称为与非运算,它是先与后非的复合运算,如图 2.2.7 所示。能实现与非运算的电路称为与非门电路,简称与非门,与非门的逻辑符号如图 2.2.8 所示,逻辑表达式为 $L=\overline{AB}$,真值表见表 2.2.6。扩展到多个变量时,相应的逻辑表达式为

$$L=\overline{ABC\cdots}$$

(2.2.6)

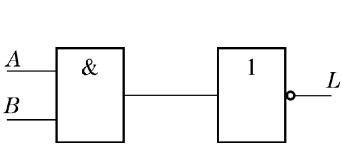


图 2.2.7 与非运算的组合电路

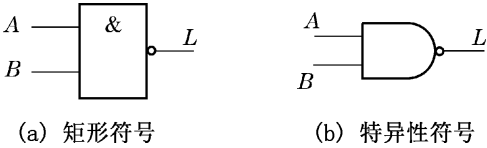


图 2.2.8 与非门逻辑符号

表 2.2.6 与非逻辑真值表

A	B	L	A	B	L
0	0	1	1	0	1
0	1	1	1	1	0

2. 或非运算

“或”和“非”的复合运算称为或非运算，它是先或后非的复合运算，如图 2. 2. 9 所示。能实现或非运算的电路称为或非门电路，简称或非门，或非门的逻辑符号如图 2. 2. 10 所示，逻辑表达式为 $L=\overline{A+B}$ ，真值表见表 2. 2. 7。扩展到多个变量时，逻辑表达式为

$$L=\overline{A+B+C+\cdots}$$

(2. 2. 7)

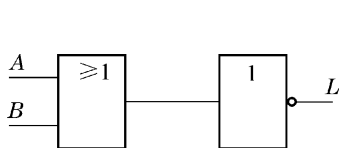


图 2. 2. 9 或非运算组合电路

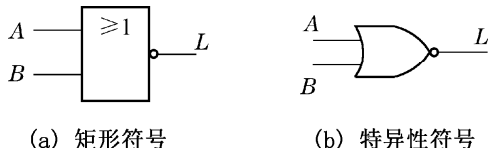


图 2. 2. 10 或非门逻辑符号

表 2. 2. 7 或非逻辑真值表

A	B	L	A	B	L
0	0	1	1	0	0
0	1	0	1	1	0

3. 与或非运算

“与”“或”和“非”的复合运算称为与或非运算，它是实现先与后或再非的复合运算，如图 2. 2. 11 所示。能实现与或非运算的电路称与或非门电路，简称与或非门，与或非运算逻辑符号如图 2. 2. 12 所示，表 2. 2. 8 是与或非逻辑真值表。相应的逻辑表达式为

$$L=\overline{AB+CD}$$

(2. 2. 8)

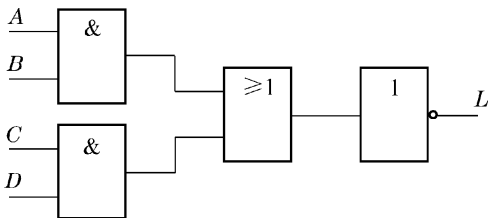


图 2. 2. 11 与或非运算组合电路

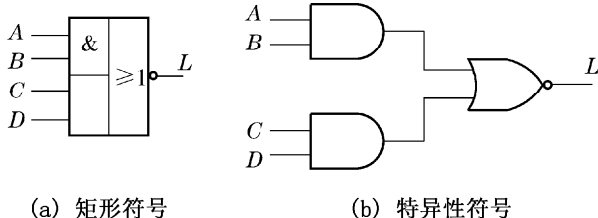


图 2. 2. 12 与或非门逻辑符号

表 2. 2. 8 与或非逻辑真值表

A	B	C	D	L	A	B	C	D	L
0	0	0	0	1	0	1	0	1	1
0	0	0	1	1	0	1	1	0	1
0	0	1	0	1	0	1	1	1	0
0	0	1	1	0	1	0	0	0	1
0	1	0	0	1	1	0	0	1	1

(续表)

A	B	C	D	L	A	B	C	D	L
1	0	1	0	1	1	1	0	1	0
1	0	1	1	0	1	1	1	0	0
1	1	0	0	0	1	1	1	1	0

4. 异或运算

异或运算的逻辑关系是：当两个输入信号相同时，输出为 0；当两个输入信号不同时，输出为 1。能实现异或运算的电路称为异或门电路，简称异或门，其逻辑符号如图 2. 2. 13 所示，表 2. 2. 9 为异或逻辑真值表，相应的逻辑表达式为

$$L=A\bar{B}+\bar{A}B=A\oplus B$$
 (2. 2. 9)

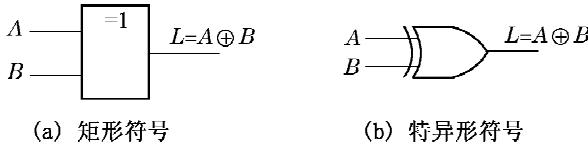


图 2. 2. 13 异或门逻辑符号

表 2. 2. 9 异或逻辑真值表

A	B	L
0	0	0
0	1	1
1	0	1
1	1	0

5. 同或运算

同或和异或的逻辑关系刚好相反：当两个输入信号相同时，输出为 1，当两个输入信号不同时，输出为 0。即异或非运算结果等于同或，同或非运算结果等于异或。能实现同或运算的电路称为同或门电路，简称同或门。同或门的逻辑符号和同或逻辑真值表分别如图 2. 2. 14 和表 2. 2. 10 所示。逻辑表达式为

$$L=\overline{AB}+AB=A\odot B=\overline{A\bar{B}+\bar{A}B}=\overline{A\oplus B}$$
 (2. 2. 10)

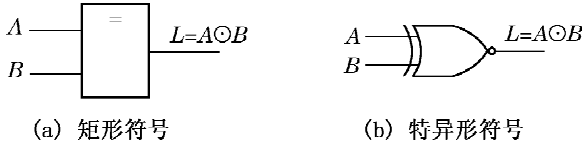


图 2. 2. 14 同或门逻辑符号

表 2.2.10 同或逻辑真值表

A	B	L
0	0	1
0	1	0
1	0	0
1	1	1

2.3 逻辑代数中的基本定律和常用公式

和普通代数一样,逻辑代数也有一系列的定律、定理和规则,用它们对逻辑函数表达式进行处理,可以完成对逻辑电路的化简、变换、分析和设计。

2.3.1 逻辑代数中的基本定律

逻辑代数中的基本定律见表 2.3.1,包括 9 个定律,即交换律、结合律、分配律、互补律、0-1 律、还原律、重叠律、吸收律和反演律。其中有的定律与普通代数定律相似,有的定律与普通函数不同,使用时切勿混淆。

表 2.3.1 逻辑代数中的基本定律

基本定律	与	或	非
0-1 律	$A \cdot 1 = A$ $A \cdot 0 = 0$	$A + 0 = A$ $A + 1 = 1$	
互补律	$A \bar{A} = 0$	$A + \bar{A} = 1$	
重叠律	$AA = A$	$A + A = A$	
交换律	$AB = BA$	$A + B = B + A$	
结合律	$A(BC) = (AB)C$	$A + (B + C) = (A + B) + C$	
分配律	$A(B + C) = AB + AC$	$A + BC = (A + B)(A + C)$	
反演律	$\overline{AB} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A} \bar{B}$	
吸收律	$A(A + B) = A$ $A(\bar{A} + B) = AB$	$A + AB = A$ $A + \bar{A}B = A + B$	
还原律			$\bar{\bar{A}} = A$

由与、或、非三种基本运算可以逻辑推理得到表 2.3.1 中的与、或、非运算的基本定律公式。例如,根据与运算 $L = A \cdot B$ 和或运算 $L = A + B$ 的运算规则,“ $0 \cdot 0 = 0, 0 \cdot 1 = 0, 1 \cdot 0 = 0, 1 \cdot 1 = 1$ ”和“ $0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1, 1 + 1 = 1$ ”,可以推理得到表 2.3.1 中的 0-1 律、互补律、重叠律公式;根据非运算法则“ $\bar{0} = 1, \bar{1} = 0$ ”,很容易推导出还原律公式 $\bar{\bar{A}} = A$;表中略为复杂的公式可用其他更简单的公式来证明。

【例 2.3.1】 证明吸收律 $A + \overline{A}B = A + B$ 。

证明 $A + \overline{A}B = A(B + \overline{B}) + \overline{A}B = AB + A\overline{B} + \overline{A}B = AB + AB + A\overline{B} + \overline{A}B$
 $= A(B + \overline{B}) + B(A + \overline{A}) = A + B$

表中的定律公式还可以用真值表来证明,即检验等式两边函数的真值表是否一致。

【例 2.3.2】 用真值表证明反演律 $\overline{AB} = \overline{A} + \overline{B}$ 和 $\overline{A + B} = \overline{A}\overline{B}$ 。

证明 分别列出两公式等号两边函数的真值表即可得证,见表 2.3.2 和表 2.3.3。

表 2.3.2 证明 $\overline{AB} = \overline{A} + \overline{B}$ 的真值表

A	B	$\overline{AB}$	$\overline{A} + \overline{B}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

表 2.3.3 证明 $\overline{A + B} = \overline{A}\overline{B}$ 的真值表

A	B	$\overline{A + B}$	$\overline{A}\overline{B}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

本例所证明的反演律,也称摩根定律,或德摩根定律,它是由英国数学家奥古斯塔斯·德摩根(Augustus De Morgan)在 19 世纪提出,此定律在逻辑函数化简和变换中经常用到。

2.3.2 逻辑代数中的常用公式

表 2.3.4 中列出了几个常用公式,这些公式是利用基本定律和基本公式导出的。直接运用这些导出公式可以给化简逻辑函数的工作带来很大方便。

表 2.3.4 化简逻辑函数的几个常用公式

序 号	公 式
1	$AB + A\overline{B} = A$
2	$A + BC = (A + B)(A + C)$
3	$(A + B)(\overline{A} + C)(B + C) = (A + B)(\overline{A} + C)$
4	$AB + \overline{A}C + BC = AB + \overline{A}C; AB + \overline{A}C + BCD = AB + \overline{A}C$
5	$A\overline{AB} = A\overline{B}; \overline{AAB} = \overline{A}$

现将表 2.3.4 中的各式证明如下:

1. $AB + A\bar{B} = A$

证明 $AB + A\bar{B} = A(B + \bar{B}) = A \cdot 1 = A$

这个公式的含义是,当两个乘积项相加时,若它们分别含 B 和 $\bar{B}$ 两个因子而其他因子相同,则两项一定能合并,且可以将 B 和 $\bar{B}$ 两个因子消去。

2. $A + BC = (A + B)(A + C)$

证明 $A + BC = AA + AC + AB + BC$
 $= A(A + C) + B(A + C)$
 $= (A + B)(A + C)$

该式说明,变量 A 与另两个变量的乘积项 BC 之“和”等于变量 A 分别与另两个变量的和之“积”。可以利用它化简逻辑函数,也可以用来对逻辑函数的与或形式和或与形式之间进行变换。

3. $(A + B)(\bar{A} + C)(B + C) = (A + B)(\bar{A} + C)$

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

上式表明,在逻辑函数的或与(和之积)表达式中,若两个和项中分别包含 A 和 $\bar{A}$,而这两个和项的其余部分组成第三个和项时,则这第三个和项是多余的,可以消去。

4. $AB + \bar{A}C + BC = AB + \bar{A}C$

证明 $AB + \bar{A}C + BC = AB + \bar{A}C + BC(A + \bar{A})$
 $= AB + \bar{A}C + ABC + \bar{A}BC$
 $= AB(1 + C) + \bar{A}C(1 + B)$
 $= AB + \bar{A}C$

此式说明,如果两个乘积项中分别包含 A 和 $\bar{A}$ 两个因子,而这两个乘积项的其余因子组成第三个乘积项时,则这第三个乘积项是多余的,可以消去。

由此,不难导出: $AB + \bar{A}C + BCD = AB + \bar{A}C$ 。

5. $A\bar{A}B = A\bar{B}; \overline{AAB} = \bar{A}$

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

上式说明,当 A 和一个乘积项的非相乘且为乘积项的因子时,则 A 这个因子可以消去。

$$\overline{AAB} = \overline{A(\bar{A} + B)} = \overline{A + AB} = \overline{A(1 + B)} = \bar{A}$$

这就说明,当 $\bar{A}$ 和一个乘积项的非相乘时,且 A 为乘积项的因子时,其结果就等于 $\bar{A}$ 。

从这里的证明可知,这些常用公式都是从基本公式导出的结果。显然,还能够推导出更多的常用公式。

2.3.3 逻辑代数中的三个基本规则

1. 代入规则

代入规则的基本内容是:对于任何一个逻辑等式,以某个逻辑变量或逻辑函数同时取代等式两端任何一个逻辑变量后,等式依然成立。

利用代入规则可以方便地扩展公式。例如,在反演律 $\overline{AB} = \overline{A} + \overline{B}$ 中用 BC 去代替等式中的 B ,则新的等式仍成立: $\overline{ABC} = \overline{A} + \overline{BC} = \overline{A} + \overline{B} + \overline{C}$,以此类推,摩根定理对任意多个变量都成立。

2. 对偶规则

将一个逻辑函数 L 进行变换,将 L 中的与“ $\cdot$ ”换成或“ $+$ ”,或“ $+$ ”换成与“ $\cdot$ ”;“ 0 ”换成“ 1 ”,“ 1 ”换成“ 0 ”,所得新函数表达式叫做 L 的对偶式,用 L' 表示。

变换时仍然要保持“先括号,然后与,最后或”的运算顺序。

例如, $L = (A+B)(A+\overline{C})$,则 $L' = AB + A\overline{C}$ 。

对偶规则的基本内容是:如果两个逻辑函数表达式相等,那么它们的对偶式也一定相等。

利用对偶规则可以帮助我们减少公式的记忆量。例如,表 2.3.1 中的左边一列公式和右边一列公式就互为对偶,只需记住一边的公式就可以了。因为利用对偶规则,不难得出另一边的公式。

3. 反演规则

根据摩根定律,由原函数 L 的表达式,求它的非函数 $\overline{L}$ 时,可以将 L 中的与“ $\cdot$ ”换成或“ $+$ ”,或“ $+$ ”换成与“ $\cdot$ ”;再将原变量换为非变量(如 A 换成 $\overline{A}$),非变量换为原变量;并将 1 换成 0 , 0 换成 1 ;那么所得的逻辑函数式就是 $\overline{L}$ 。这个规则称为反演规则。

利用反演规则,可以比较容易地求出一个函数的反函数。

【例 2.3.3】 求函数 $L = \overline{AC} + B\overline{D}$ 的反函数。

解 $\overline{L} = (A + \overline{C}) \cdot (\overline{B} + D)$

【例 2.3.4】 求函数 $L = A \cdot B + C + \overline{D}$ 的反函数。

解 $\overline{L} = \overline{A + B \cdot C \cdot D}$

运用反演规则时必须注意以下两个原则:

- (1) 仍需遵守“先括号内运算、然后与运算、最后或运算”的运算优先次序;
- (2) 不属于单个变量上的非号应保留不变。

2.4 逻辑函数及其表示方法

从前面介绍的逻辑运算中可以知道,逻辑变量分为两种:输入逻辑变量和输出逻辑变量。描述输入逻辑变量和输出逻辑变量之间的因果关系的函数称为逻辑函数。由于逻辑变量是只取 0 或 1 的二值逻辑变量,因此逻辑函数也是二值逻辑函数。由于逻辑函数是从生

活和生产实践中抽象出来的,所以只有那些能明确地用“是”或“否”做出回答的事物,才能定义为逻辑函数。

2.4.1 逻辑函数的建立

下面通过一个例子来介绍逻辑函数的建立步骤。

【例 2.4.1】 三个人表决一件事情,表决结果按“少数服从多数”的原则决定。试用逻辑函数表达表决意见与表决结果之间的逻辑关系。

解 第一步:设置自变量和因变量。将三人的意见设置为自变量 A 、 B 、 C ,并规定只能有同意或不同意两种意见。将表决结果设置为因变量 L ,显然也只有通过和没通过这两种情况。

第二步:状态赋值。对于自变量 A 、 B 、 C ,设同意为逻辑“1”,不同意为逻辑“0”。对于因变量 L ,设事情通过为逻辑“1”,没通过为逻辑“0”。

第三步:根据题意及上述设置规定,列出函数的真值表如表 2.4.1 所示。

由真值表可以看出,当自变量 A 、 B 、 C 取确定值后,因变量 L 的值就完全确定了。所以, L 就是 A 、 B 、 C 的函数。 A 、 B 、 C 称为输入逻辑变量, L 称为输出逻辑变量。

表 2.4.1 例 2.4.1 真值表

输 入	输 出	输 入	输 出
$A \ B \ C$	L	$A \ B \ C$	L
0 0 0	0	1 0 0	0
0 0 1	0	1 0 1	1
0 1 0	0	1 1 0	1
0 1 1	1	1 1 1	1

一般地说,若输入逻辑变量 A 、 B 、 C …的取值确定以后,输出逻辑变量 L 的值也唯一地确定了,就称 L 是 A 、 B 、 C …的逻辑函数,写作:

$$L=f(A,B,C,\cdots) \quad (2.4.1)$$

逻辑函数与普通代数中的函数相比较,我们必须牢记两个突出的特点:

- (1) 逻辑变量和逻辑函数只能取两个值 0 或 1。
- (2) 函数和变量之间的关系是由“与”“或”“非”三种基本逻辑运算决定的。

2.4.2 逻辑函数的表示方法

常用的逻辑函数表示方法有逻辑真值表、逻辑表达式(逻辑函数式或简称逻辑式)、逻辑图、波形图、卡诺图等。本小节只讨论前面四种方法,用卡诺图表示逻辑函数的方法将在后面再做专门介绍。

1. 真值表

逻辑真值表是将输入逻辑变量的各种可能的取值和相应的函数值排列在一起而组成的

表格。为了避免遗漏,各变量的取值组合一般应该按照二进制数递增的次序排列。如前面介绍过的表 2.2.4~2.2.8、表 2.4.1 等。

真值表表示逻辑函数具有以下特点:

- (1) 直观明了。输入变量取值一旦确定后,即可在真值表中查出相应的函数值。
- (2) 把一个实际的逻辑问题抽象成一个逻辑函数时,使用真值表是最方便的。所以,在设计逻辑电路时,常常是先根据设计要求列出真值表。

真值表的主要缺点是:无法运用逻辑代数的公式、定律进行运算,当变量比较多时,表比较大,显得过于繁琐。

2. 逻辑表达式

逻辑表达式是指由逻辑变量和“与”“或”“非”三种运算所构成的表达式。这是一种用公式表示逻辑函数的方法。

例如, $L(A,B)=A\bar{B}+\bar{A}B$ 。

由真值表转换为逻辑函数表达式的方法是:在真值表中依次找出函数值等于 1 的变量组合,变量值为 1 的写成原变量,变量值为 0 的写成反变量,把组合中各个变量相乘。这样,对应于函数值为 1 的每一个变量组合就可以写成一个乘积项。然后,把所有的乘积项相加,就得到相应的函数表达式(与或表达式)了。

例如,用此方法可以直接由表 2.4.1 写出“三人表决”函数的逻辑表达式:

$$L=\bar{A}BC+A\bar{B}C+AB\bar{C}+ABC$$

反之,由逻辑表达式也可以转换成真值表。方法为:画出真值表的表格,将变量及变量的所有取值组合按照二进制递增的次序列入表格左边,然后按照表达式,依次对变量的各种取值组合进行运算,求出相应的函数值,填入表格右边对应的位置,即得真值表。

【例 2.4.2】 列出函数 $L=\bar{A}\bar{B}+AB$ 的真值表。

解 该函数有两个变量,有 4 种取值的可能组合,将他们按顺序排列起来即得真值表,如表 2.4.2 所示。

表 2.4.2 $L=\bar{A}\bar{B}+AB$ 的真值表

A	B	L	A	B	L
0	0	1	1	0	0
0	1	0	1	1	1

逻辑表达式是我们最先接触逻辑函数的表示方法,这种表示方法的优点有:

- (1) 用基本逻辑符号高度抽象而概括地表示各个变量之间的逻辑关系,书写简洁方便。
- (2) 便于运用逻辑代数的公式、定律进行运算和变换。
- (3) 便于用逻辑图实现函数。只要用相应的门电路的逻辑符号代替表达式中的有关运算,即可得到逻辑图。

逻辑表达式的主要缺点是:同一个逻辑函数有不同的表达式,不易判断各个表达式彼此是否相等;而且在逻辑函数比较复杂时难以看出逻辑函数的值,没有真值表直观。

3. 逻辑图

用与、或、非等逻辑符号表示逻辑函数中各变量之间的逻辑关系所得到的图形称为逻辑

图。它是由逻辑符号及它们之间的连线而构成的图形。

由逻辑函数表达式可以画出其相应的逻辑图。

【例 2.4.3】 画出逻辑函数 $L=AB+\overline{A}\overline{B}$ 的逻辑图。

解 如图 2.4.1 所示。

由逻辑图也可以写出其相应的函数表达式。

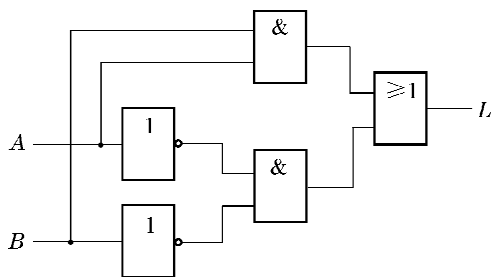


图 2.4.1 例 2.4.3 的逻辑图

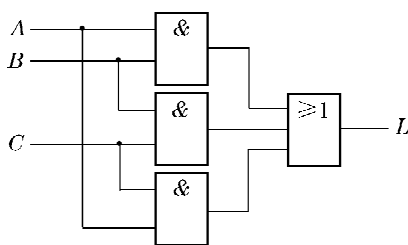


图 2.4.2 例 2.4.4 的逻辑图

【例 2.4.4】 写出如图 2.4.2 所示逻辑图的逻辑函数表达式。

解 该逻辑图是由基本的“与”“或”逻辑符号组成的,可由输入至输出逐步写出逻辑表达式: $L=AB+BC+AC$ 。

逻辑电路图中的逻辑符号,与实际使用的电路器件有着明显的联系,所以它比较接近工程实际;但是对于稍复杂一些的逻辑电路图,我们很难直接看出其功能,而且逻辑图不易书写(绘制),不易记忆。

4. 波形图

如果将逻辑函数输入变量每一种可能出现的取值与对应的输出值按时间顺序依次排列起来,得到的波形为该逻辑函数的波形图。这种波形图也称为时序图。在逻辑分析仪和一些计算机仿真工具中,经常以这种波形图的形式给出分析结果。此外,也可以通过示波器实验观察这些波形图,以检验实际逻辑电路的功能是否正确。

如果用波形图来描述逻辑函数 $L=AB+BC$,则只需将表 2.4.3 所示的真值表给出的输入变量与对应的输出变量取值依时间顺序排列起来,就可以得到所需要的波形图了,如图 2.4.3 所示。

表 2.4.3 $L=AB+BC$ 的真值表

A	B	C	L
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

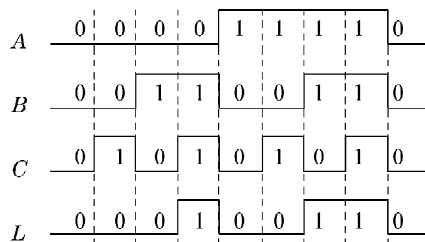


图 2.4.3 $L=AB+BC$ 的波形图

以上通过几个简单的例子说明了逻辑函数的四种表示方法,这四种方法各有特点。实际应用中的逻辑函数往往比较复杂,应当根据具体的实际情况选用合适的表示方法。

2.4.3 逻辑函数的两种标准形式

在讲述逻辑函数的标准形式之前,先介绍一下最小项和最大项的概念,然后再介绍逻辑函数的“最小项之和”及“最大项之积”这两种标准形式。

1. 最小项和最大项

(1) 最小项

在 n 个变量的逻辑函数中,若 m 为包含 n 个因子的乘积项,而且这 n 个变量均以原变量或反变量的形式在 m 中仅出现一次,则称 m 为该组变量的最小项。

例如, A 、 B 、 C 三个变量的最小项有 $\overline{A}\overline{B}\overline{C}$ 、 $\overline{A}\overline{B}C$ 、 $\overline{A}B\overline{C}$ 、 $\overline{A}BC$ 、 $A\overline{B}\overline{C}$ 、 $A\overline{B}C$ 、 $AB\overline{C}$ 和 ABC ,共 8 个(即 2^3 个)。 n 个变量的最小项总共有 2^n 个。

输入变量的每一组取值都使一个对应的最小项的值等于 1。例如,在三变量 A 、 B 、 C 的最小项中,当 $A=1$ 、 $B=0$ 、 $C=1$ 时, $A\overline{B}C=1$ 。如果把 $A\overline{B}C$ 的取值 011 看作一个二进制数,那么它所表示的十进制数就是 3。为了今后使用方便,将 $A\overline{B}C$ 这个最小项记作 m_3 。按照这一约定,就得到了三变量最小项的编号表,如表 2.4.4 所示。

表 2.4.4 三变量逻辑函数的最小项及编号

最小项	变量取值 A B C	编号	最小项	变量取值 A B C	编号
$\overline{A}\overline{B}\overline{C}$	0 0 0	m_0	$A\overline{B}\overline{C}$	1 0 0	m_4
$\overline{A}\overline{B}C$	0 0 1	m_1	$A\overline{B}C$	1 0 1	m_5
$\overline{A}B\overline{C}$	0 1 0	m_2	$AB\overline{C}$	1 1 0	m_6
$\overline{A}BC$	0 1 1	m_3	ABC	1 1 1	m_7

根据同样的道理,我们将 A 、 B 、 C 、 D 这 4 个变量的 16 个最小项记作 $m_0 \sim m_{15}$ 。

从最小项的定义出发,可以证明它具有如下的重要性质:

- ① 在输入变量的任何取值下必有一个最小项,且仅有一个最小项的值为 1。
- ② 全体最小项之和为 1。
- ③ 任意两个最小项的乘积为 0。
- ④ 具有相邻性的两个最小项之和可以合并成一项并消去一对因子。

若两个最小项只有一个因子不同,则称这两个最小项具有相邻性。例如, $\overline{A}\overline{B}C$ 和 $\overline{A}\overline{B}\overline{C}$ 两个最小项仅第三个因子不同,所以它们具有相邻性。这两个最小项相加时一定能合并成一项并将一对不同的因子消去,即

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

(2) 最大项

在 n 个变量逻辑函数中,若 M 为 n 个变量之和,而且这 n 个变量均以原变量或反变量的形式在 M 中只出现一次,则称 M 为该组变量的最大项。

例如,三变量 A, B, C 的最大项有 $(\overline{A}+\overline{B}+\overline{C})$ 、 $(\overline{A}+\overline{B}+C)$ 、 $(\overline{A}+B+\overline{C})$ 、 $(\overline{A}+B+C)$ 、 $(A+\overline{B}+\overline{C})$ 、 $(A+\overline{B}+C)$ 、 $(A+B+\overline{C})$ 、 $(A+B+C)$, 共 8 个(即 2^3 个)。对于 n 个变量, 则有 2^n 个最大项。可见, n 个变量的最大项数目和最小项数目是相等的。

输入变量的每一组取值都使一个对应的最大项的值为 0。例如, 在三变量 A, B, C 的最大项中, 当 $A=1, B=1, C=0$ 时, $\overline{A}+\overline{B}+C=0$ 。若将使最大项为 0 的 A, B, C 取值视为一个二进制数, 并以其对应的十进制数给最大项编号, 则 $(\overline{A}+\overline{B}+C)$ 可记作 M_6 。由此得到的三变量最大项编号表(表 2.4.5)。

表 2.4.5 三变量最大项的编号表

最大项	使最大项为 0 的变量取值 $A \quad B \quad C$	对应的十进制数	编号
$A+B+C$	0 0 0	0	M_0
$A+B+\overline{C}$	0 0 1	1	M_1
$A+\overline{B}+C$	0 1 0	2	M_2
$A+\overline{B}+\overline{C}$	0 1 1	3	M_3
$\overline{A}+B+C$	1 0 0	4	M_4
$\overline{A}+B+\overline{C}$	1 0 1	5	M_5
$\overline{A}+\overline{B}+C$	1 1 0	6	M_6
$\overline{A}+\overline{B}+\overline{C}$	1 1 1	7	M_7

根据最大项的定义同样也可以得到它的主要性质:

- ① 在输入变量的任何取值下必有一个最大项, 而且只有一个最大项为 0。
- ② 全体最大项之积为 0。
- ③ 任意两个最大项之和为 1。
- ④ 只有一个变量不同的两个最大项的乘积等于各相同变量之和。

如果将表 2.4.4 和表 2.4.5 加以对比则可发现, 最大项和最小项之间存在如下关系:

$$M_i = \overline{m_i} \quad (2.4.2)$$

例如, $m_5 = A\overline{B}C$, 则 $\overline{m_5} = \overline{A\overline{B}C} = \overline{A}+B+\overline{C} = M_5$ 。

2. 逻辑函数的最小项之和形式

首先将给定的逻辑函数式化为若干乘积项之和的形式(亦称“积之和”形式), 然后再利用基本公式 $A+\overline{A}=1$ 将每个乘积项中缺少的因子补全, 这样就可以将与或的形式化为最小项之和的标准形式。这种标准形式在逻辑函数的化简以及计算机辅助分析和设计中得到了广泛的应用。

例如, 给定逻辑函数为

$$L(A, B, C) = A\overline{B}C + BC$$

则可化为

$$L(A, B, C) = A\overline{B}C + (A+\overline{A})BC = A\overline{B}C + ABC + \overline{A}BC = m_3 + m_5 + m_7$$

或写作

$$L(A, B, C) = \sum m(3, 5, 7)$$

【例 2.4.5】 将逻辑函数 $L(A, B, C) = AB + \bar{A}C$ 转换成最小项表达式。

解 该函数为三变量函数, 而表达式中每项只含有两个变量, 不是最小项。要变为最小项, 就应补齐缺少的变量, 办法为将各项乘以 1, 如 AB 项乘以 $(C + \bar{C})$ 。

$$\begin{aligned} L(A, B, C) &= AB + \bar{A}C = AB(C + \bar{C}) + \bar{A}C(B + \bar{B}) \\ &= ABC + AB\bar{C} + \bar{A}BC + \bar{A}\bar{B}C \\ &= m_7 + m_6 + m_3 + m_1 \end{aligned}$$

或者

$$L(A, B, C) = \sum m(1, 3, 6, 7)$$

在逻辑函数包含的最小项个数较多时, 这种用最小项下标编号来表示最小项的方法比较简便。

3. 逻辑函数的最大项之积形式

利用逻辑代数的基本定律和公式, 首先我们一定能把任何一个逻辑函数式化成若干多项式相乘的或与形式(也称为“和之积”形式)。然后再利用基本公式 $A\bar{A} = 0$ 将每个多项式中缺少的变量补齐, 就可以将函数式的或与形式化成最大项之积的形式了。

【例 2.4.6】 将逻辑函数 $L = \bar{A}B + AC$ 化成最大项之积的形式。

解 首先利用公式 $A + BC = (A + B)(A + C)$ 将 L 化成或与形式:

$$\begin{aligned} L &= \bar{A}B + AC \\ &= (\bar{A}B + A)(\bar{A}B + C) \\ &= (A + B)(\bar{A} + C)(B + C) \end{aligned}$$

然后在第一个括号内加上 $C\bar{C}$, 第二个括号内加上 $B\bar{B}$, 第三个括号内加上 $A\bar{A}$, 于是得到

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

或写作

$$L(A, B, C, D) = \prod M(0, 1, 4, 6)$$

2.5 逻辑函数的公式化简法

根据逻辑函数表达式, 可以画出相应的逻辑图。然而, 在进行逻辑运算时常常会看到, 同一个逻辑函数可以写成不同的逻辑表达式, 而这些逻辑表达式的繁简程度又相差甚远。逻辑表达式越是简单, 它所表示的逻辑关系越明显, 构成逻辑电路实现逻辑函数时所用的电子器件就越少, 越有利于降低成本, 提高数字系统的可靠性。因此, 经常需要通过化简的手段找出逻辑函数的最简形式。

2.5.1 逻辑函数的最简表达式

由于一个逻辑函数的表达式不是唯一的,可以有多种形式。例如,有一个逻辑函数表达式为

$$L = AC + \overline{A}B$$

式中 AC 和 $\overline{A}B$ 两项都是由与(逻辑乘)运算把变量连接起来的,故称为与项(乘积项),然后由或运算将这两个与项连接起来。这种类型的表达式称为与-或逻辑表达式,或称为逻辑函数表达式的“积之和”形式。

在若干个逻辑关系相同的与-或表达式中,将其中包含的与项数最少,且每个与项中变量数最少的表达式称为最简与-或表达式。

一个与-或表达式易于转换为其他类型的函数式。例如,上面的与-或表达式经过变换,可以得到与其对应的与非-与非表达式、或-与表达式、或非-或非表达式以及与-或非表达式四种表达式。例如:

$$\begin{aligned} L &= AC + \overline{A}B && \text{与-或表达式} \\ &= (A+B)(\overline{A}+C) && \text{或-与表达式} \\ &= \overline{\overline{AC} \cdot \overline{\overline{A}B}} && \text{与非-与非表达式} \\ &= \overline{\overline{A+B} + \overline{\overline{A}+C}} && \text{或非-或非表达式} \\ &= \overline{\overline{A} \overline{C} + \overline{A} \overline{B}} && \text{与-或非表达式} \end{aligned}$$

在上述多种表达式中,与-或表达式是逻辑函数的最基本表达形式。因此,在化简逻辑函数时,通常是将逻辑式化简成最简与-或表达式,然后再根据需要转换成其他形式。

最简与-或表达式的标准是:

- (1) 与项最少,即表达式中“+”号最少。
- (2) 每个与项中的变量数最少,即表达式中“ $\cdot$ ”号最少。

与项最少,可以使电路实现时所需的逻辑门的个数最少;每个与项中的变量数最少,可以使电路实现时所需逻辑门的扇入系数即输入端个数最少。这样就可以保证电路最简,成本最低。

对于其他类型的电路,也可以得出类似的“最简”标准。例如或-与表达式,其“最简”的标准可以变更为:或项最少;每个或项中的变量数最少。

2.5.2 逻辑函数的公式化简法

用公式化简法化简逻辑函数,就是直接利用逻辑代数的基本公式和基本规则进行化简。公式化简法没有固定的步骤,常用的化简方法有以下几种。

- (1) 并项法。运用公式 $A + \overline{A} = 1$,将两项合并为一项,消去一个变量。如:

$$L = AB\bar{C} + ABC = AB(\bar{C} + C) = AB$$

$$\begin{aligned} L &= A(BC + \bar{B}\bar{C}) + A(B\bar{C} + \bar{B}C) = ABC + A\bar{B}\bar{C} + AB\bar{C} + A\bar{B}C \\ &= AB(C + \bar{C}) + A\bar{B}(C + \bar{C}) = AB + A\bar{B} = A(B + \bar{B}) = A \end{aligned}$$

(2) 吸收法。运用吸收律 $A + AB = A$ 消去多余的与项。如：

$$L = A\bar{B} + A\bar{B}(C + DE) = A\bar{B}$$

(3) 消去法。运用吸收律 $A + \bar{A}B = A + B$ 消去多余的因子。如：

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

$$L = \bar{A} + AB + \bar{B}E = \bar{A} + B + \bar{B}E = \bar{A} + B + E$$

(4) 配项法。先通过乘以 $A + \bar{A} (=1)$ 或加上 $A\bar{A} (=0)$, 增加必要的乘积项, 再用以上方法化简。如：

$$L = AB + \bar{A}C + BCD = AB + \bar{A}C + BCD(A + \bar{A}) = AB + \bar{A}C + ABCD + \bar{A}BCD = AB + \bar{A}C$$

$$\begin{aligned} L &= AB\bar{C} + \bar{A}BC \cdot \bar{A}\bar{B} = AB\bar{C} + \bar{A}BC \cdot \bar{A}\bar{B} + AB \cdot \bar{A}\bar{B} = AB(\bar{C} + \bar{A}\bar{B}) + \bar{A}BC \cdot \bar{A}\bar{B} \\ &= AB \cdot \bar{A}BC + \bar{A}BC \cdot \bar{A}\bar{B} = \bar{A}BC(AB + \bar{A}\bar{B}) = \bar{A}BC \end{aligned}$$

使用配项法要注意防止越配越繁。通常对逻辑函数进行化简, 要灵活运用上述方法技巧, 才能将逻辑函数化为最简。

【例 2.5.1】 化简逻辑函数 $L = A\bar{B} + A\bar{C} + A\bar{D} + ABCD$ 。

解 $L = A(\bar{B} + \bar{C} + \bar{D}) + ABCD = A\bar{B}CD + ABCD = A(\bar{B}CD + BCD) = A$

【例 2.5.2】 化简逻辑函数 $L = AD + A\bar{D} + AB + \bar{A}C + BD + A\bar{B}EF + \bar{B}EF$ 。

解 $L = A + AB + \bar{A}C + BD + A\bar{B}EF + \bar{B}EF$ (利用 $A + \bar{A} = 1$)

$$= A + \bar{A}C + BD + \bar{B}EF \text{ (利用 } A + AB = A \text{)}$$

$$= A + C + BD + \bar{B}EF \text{ (利用 } A + \bar{A}B = A + B \text{)}$$

【例 2.5.3】 化简逻辑函数 $L = AB + A\bar{C} + \bar{B}C + \bar{C}B + \bar{B}D + \bar{D}B + ADE(F + G)$ 。

解 $L = A\bar{B}C + \bar{B}C + \bar{C}B + \bar{B}D + \bar{D}B + ADE(F + G)$ (利用反演律 $A(B + \bar{C}) = A\bar{B}C$)

$$= A + \bar{B}C + \bar{C}B + \bar{B}D + \bar{D}B + ADE(F + G) \text{ (利用 } A + \bar{A}B = A + B \text{)}$$

$$= A + \bar{B}C + \bar{C}B + \bar{B}D + \bar{D}B \text{ (利用 } A + AB = A \text{)}$$

$$= A + \bar{B}C(D + \bar{D}) + \bar{C}B + \bar{B}D + \bar{D}B(C + \bar{C}) \text{ (配项法)}$$

$$= A + \bar{B}CD + \bar{B}C\bar{D} + \bar{C}B + \bar{B}D + \bar{D}BC + \bar{D}B\bar{C}$$

$$= A + \bar{B}C\bar{D} + \bar{C}B + \bar{B}D + \bar{D}BC \text{ (利用 } A + AB = A \text{)}$$

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

$$= A + C\bar{D} + \bar{C}B + \bar{B}D \text{ (利用 } A + \bar{A} = 1 \text{)}$$

【例 2.5.4】 化简逻辑函数 $L = A\bar{B} + B\bar{C} + \bar{B}C + \bar{A}B$ 。

解法 1 $L = A\bar{B} + B\bar{C} + \bar{B}C + \bar{A}B + A\bar{C}$ (增加冗余项 $A\bar{C}$)

$$= A\bar{B} + \bar{B}C + \bar{A}B + A\bar{C} \text{ (消去 1 个冗余项 } B\bar{C} \text{)}$$

$$= \bar{B}C + \bar{A}B + A\bar{C} \text{ (再消去 1 个冗余项 } A\bar{B} \text{)}$$

解法 2 $L = A\bar{B} + B\bar{C} + \bar{B}C + \bar{A}B + \bar{A}C$ (增加冗余项 $\bar{A}C$)

$$= A\bar{B} + B\bar{C} + \bar{A}B + \bar{A}C \text{ (消去 1 个冗余项 } \bar{B}C \text{)}$$

$$=A\bar{B}+B\bar{C}+\bar{A}C \quad (\text{再消去1个冗余项}\bar{A}B)$$

由上例可知,逻辑函数的化简结果不是唯一的。

【例 2.5.5】 已知逻辑函数表达式为 $L=AB\bar{D}+\bar{A}BD+\bar{A}BD+\overline{ABCD}+\overline{ABCD}$ 。要求:

- (1) 最简的与-或逻辑函数表达式,并画出相应的逻辑图;
- (2) 仅用与非门画出最简表达式的逻辑图。

解 $L=AB(D+\bar{D})+\bar{A}\bar{B}\bar{D}+\bar{A}\bar{B}D(C+\bar{C})$ (分配律)

$$=AB+\bar{A}\bar{B}\bar{D}+\bar{A}\bar{B}D \quad (\text{利用 } A+\bar{A}=1)$$

$$=AB+\bar{A}\bar{B}(D+\bar{D}) \quad (\text{利用 } A+\bar{A}=1)$$

$$=AB+\bar{A}\bar{B} \quad (\text{与-或表达式})$$

$$=\overline{\overline{AB+\bar{A}\bar{B}}} \quad (\text{先利用 } \overline{\overline{A}}=A, \text{再利利用摩根定律})$$

$$=\overline{\overline{AB}} \cdot \overline{\overline{\bar{A}\bar{B}}} \quad (\text{与非-与非表达式})$$

最简与-或表达式的逻辑图如图 2.5.1 所示,使用与非门的等效逻辑图如图 2.5.2 所示。

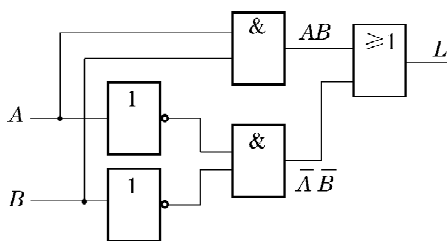


图 2.5.1 例 2.5.5 的逻辑电路

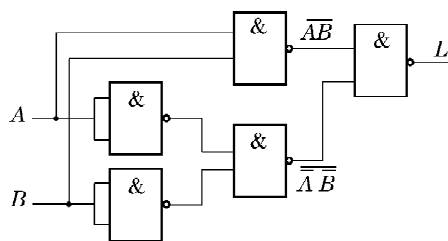


图 2.5.2 改用与非门实现的例 2.5.5 的逻辑电路

图 2.5.1 所示为根据最简与-或表达式画出的逻辑图,它用到与门、或门、和非门 3 种类型的逻辑门;图 2.5.2 所示为根据与非-与非表达式画出的逻辑图,它只用到两个输入端与非门一种类型的逻辑门。通常在一片集成电路器件内部有多个同类型的门电路,所以利用摩根定律对逻辑函数表达式进行变换,可以减少门电路的种类和集成电路的数量,具有一定的实际意义。

将与-或表达式变换为与非-与非表达式时,首先对与-或表达式取两次非,然后根据摩根定律分开下面的取非线。将与-或表达式变换成或非-或非表达式时,先对与-或表达式中的每个乘积项单独取两次非,后按照摩根定律分开下面的取非线,最后对整个表达式取两次非。下面再举一例说明这种形式的逻辑函数变换。

【例 2.5.6】 试对逻辑表达式 $L=\overline{A}BC+A\bar{B}C$ 进行变换,仅用或非门画出该表达式的逻辑图。

解 仿照上例,只用或非门来实现也是可以的,只需把函数向或非形式进行变换:

$$\begin{aligned} L &= \overline{\overline{\overline{A}BC}} + \overline{\overline{\overline{A}BC}} \\ &= \overline{A+B+C} + \overline{\overline{A}+\overline{B}+\overline{C}} \quad (\text{摩根定律}) \\ &= \overline{\overline{\overline{\overline{A}BC}} + \overline{\overline{\overline{A}BC}}} \quad (\text{或非-或非表达式}) \end{aligned}$$

仅用或非门画出的逻辑图如图 2.5.3 所示。

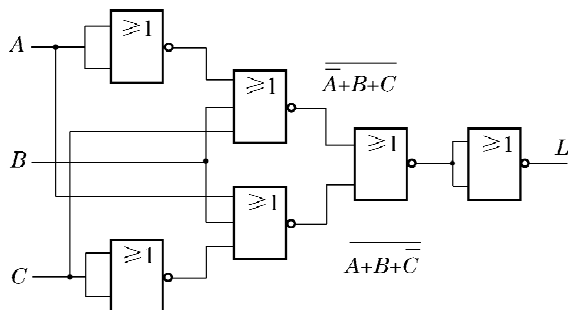


图 2.5.3 例 2.5.6 的逻辑电路图

在实际应用中,有时最简逻辑表达式并不是最好的选择,比如某函数表达式形式上是最简的,但是实现的成本却不是最低的(如器件种类多,使成本反而较高,使性价比较低)或者还会出现问题而不能使用(参阅*4.8 组合逻辑电路的竞争冒险)。所以,化简和变换的原则方向是正确的,但是要根据应用实际具体情况灵活掌握。

公式化简法的优点是不受变量数目的限制。缺点是:没有固定的步骤可循;需要熟练运用各种公式和定律;需要一定的技巧和经验;有时较难判定化简结果是否最简。因此,在变量数不多的情况下,通常多采用卡诺图化简法化简。

2.6 逻辑函数的卡诺图化简法

卡诺图是美国工程师卡诺(Karnaugh)首先提出的,它是一种按逻辑相邻原则排列成的方格图,利用相邻项合并的原则来使逻辑函数得到化简。由于卡诺图化简法简单、直观,而且可靠,因而得到了广泛的应用。

2.6.1 卡诺图的构成

在逻辑函数的真值表中,输入变量的每一种组合都和一个最小项对应,这种真值表称为最小项真值表。将逻辑函数真值表中的最小项排列成矩阵,并且矩阵的横向和纵向的逻辑变量的取值按照格雷码的顺序排列(即相邻的数码只有一位码不同),这样构成的图形称为卡诺图。图 2.6.1 分别为二变量、三变量、四变量、五变量的卡诺图。

由于相邻的最小项只有一个变量不同而其余变量都相同,如 ABC 和 $\overline{A}BC$ 、 $ABCD$ 和 $A\overline{B}CD$,而相邻最小项可以利用公式 $A + \overline{A} = 1$ 来消去一个变量,如 $ABC + \overline{A}BC = BC$ 、 $ABCD + A\overline{B}CD = ACD$ 。逻辑函数的化简实质上就是相邻最小项的合并。

卡诺图的排列特点是具有很强的相邻性。

首先是直观相邻性,只要小方格在几何位置上相邻(不管上下左右),它代表的最小项在逻辑上一定是相邻的。

		B	
		$\bar{B}$	B
A	$\bar{A}$	m_0 $\bar{A}\bar{B}$	m_1 $\bar{A}B$
	A	m_2 $A\bar{B}$	m_3 AB

		B	
		0	1
A	0	0	1
	1	2	3

(a) 二变量 (A, B) 最小项的卡诺图

		BC			
		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$BC\bar{C}$
A	$\bar{A}$	m_0 $\bar{A}\bar{B}\bar{C}$	m_1 $\bar{A}\bar{B}C$	m_3 $\bar{A}B\bar{C}$	m_2 $\bar{A}BC$
	A	m_4 $A\bar{B}\bar{C}$	m_5 $A\bar{B}C$	m_7 $AB\bar{C}$	m_6 ABC

		BC			
		00	01	11	10
A	0	0	1	3	2
	1	4	5	7	6

(b) 三变量 (A, B, C) 最小项的卡诺图

		CD			
		$\bar{C}\bar{D}$	$\bar{C}D$	CD	$CD\bar{D}$
AB	$\bar{A}\bar{B}$	m_0 $\bar{A}\bar{B}\bar{C}\bar{D}$	m_1 $\bar{A}\bar{B}\bar{C}D$	m_3 $\bar{A}\bar{B}C\bar{D}$	m_2 $\bar{A}\bar{B}CD$
	$\bar{A}B$	m_4 $\bar{A}B\bar{C}\bar{D}$	m_5 $\bar{A}B\bar{C}D$	m_7 $\bar{A}B\bar{C}D$	m_6 $\bar{A}BCD$
AB	AB	m_{12} $AB\bar{C}\bar{D}$	m_{13} $AB\bar{C}D$	m_{15} $ABC\bar{D}$	m_{14} $ABCD$
	$A\bar{B}$	m_8 $A\bar{B}\bar{C}\bar{D}$	m_9 $A\bar{B}\bar{C}D$	m_{11} $A\bar{B}C\bar{D}$	m_{10} $A\bar{B}CD$

		CD			
		00	01	11	10
AB	00	0	1	3	2
	01	4	5	7	6
AB	11	12	13	15	14
	10	8	9	11	10

(c) 四变量 (A, B, C, D) 最小项的卡诺图

		CDE							
		$\bar{C}\bar{D}\bar{E}$	$\bar{C}\bar{D}E$	$\bar{C}D\bar{E}$	$\bar{C}DE$	$C\bar{D}\bar{E}$	$C\bar{D}E$	$CD\bar{E}$	CDE
AB	$\bar{A}\bar{B}$	m_0	m_1	m_3	m_2	m_6	m_7	m_5	m_4
	$\bar{A}B$	m_8	m_9	m_{11}	m_{10}	m_{14}	m_{15}	m_{13}	m_{12}
AB	AB	m_{24}	m_{25}	m_{27}	m_{26}	m_{30}	m_{31}	m_{29}	m_{28}
	$A\bar{B}$	m_{16}	m_{17}	m_{19}	m_{18}	m_{22}	m_{23}	m_{21}	m_{20}

		CDE							
		000	001	011	010	110	111	101	100
AB	00	0	1	3	2	6	7	5	4
	01	8	9	11	10	14	15	13	12
AB	11	24	25	27	26	30	31	29	28
	10	16	17	19	18	22	23	21	20

(d) 五变量 (A, B, C, D, E) 最小项的卡诺图

图 2.6.1 二到五变量最小项的卡诺图

其次是对边相邻性,即与中心轴对称的左右两边和上下两边的小方格也具有相邻性。所以,四角的最小项也都是相邻的。

凡是两个相邻的最小项,它们在图中也是相邻的。所以,二变量的最小项有 2 个最小项与之相邻,三变量的最小项有 3 个最小项与之相邻,四变量的最小项有 4 个最小项与之相邻,五变量的最小项有 5 个最小项与之相邻,以此类推。

从图 2.6.1 可以看出,随着逻辑函数变量的增多,相应的卡诺图的复杂程度也成倍地增加。所以,卡诺图一般只适用于 5 个变量以内的情况。

2.6.2 逻辑函数的卡诺图表示法

用卡诺图表示逻辑函数,就是把逻辑函数的最小项表达式中,每一个最小项对应的方格填上 1,其余的方格填上 0(也可以空着都不填),这样就得到了该逻辑函数的卡诺图。

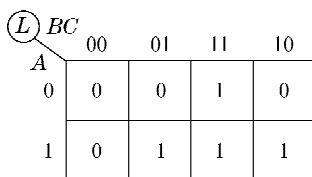
1. 利用真值表填卡诺图

【例 2.6.1】 某逻辑函数的真值表如表 2.6.1 所示,给出该逻辑函数的卡诺图。

解 该函数为三变量,先画出三变量卡诺图,然后根据表 2.6.1 将 8 个最小项 L 的取值 0 或者 1 填入卡诺图中对应的 8 个小方格中即可,如图 2.6.2 所示。

表 2.6.1 某逻辑函数的真值表

A	B	C	L
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



$\bar{L}$ BC	00	01	11	10
A 0	0	0	1	0
1	0	1	1	1

图 2.6.2 例 2.6.1 的卡诺图

2. 根据逻辑表达式填卡诺图

(1) 如果逻辑表达式为最小项表达式,则只要将函数式中出现的最小项在卡诺图对应的小方格中填入 1,没出现的最小项则在卡诺图对应的小方格中填入 0(或者空着)。

【例 2.6.2】 用卡诺图表示逻辑函数 $L = \bar{A}\bar{B}\bar{C} + \bar{A}BC + AB\bar{C} + ABC$ 。

解 该函数为三变量,且为最小项表达式,写成简化形式 $L = m_0 + m_3 + m_6 + m_7$ 。然后画出三变量卡诺图,将卡诺图中 m_0 、 m_3 、 m_6 、 m_7 对应的小方格填 1,其他小方格填 0,如图 2.6.3 所示。

$\textcircled{L}$ BC		00	01	11	10
		0	1	0	1
A		0	1	0	1
		1	0	0	1

图 2.6.3 例 2.6.2 的卡诺图

$\textcircled{L}$ CD		00	01	11	10
		0	0	0	0
AB		00	0	1	0
		01	0	1	0
		11	0	1	0
		10	1	1	1

图 2.6.4 例 2.6.3 的卡诺图

(2) 如果逻辑表达式不是最小项表达式,可将“与-或表达式”先化成最小项表达式,再填入卡诺图。也可直接填入,直接填入的具体方法是:分别找出每一个与项所包含的所有小方格,全部填入 1。

【例 2.6.3】 用卡诺图表示逻辑函数 $L = A\bar{B} + B\bar{C}\bar{D}$ 。

$$\begin{aligned}
 \text{解 } L &= A\bar{B} + B\bar{C}\bar{D} \\
 &= A\bar{B}(C + \bar{C})(D + \bar{D}) + (A + \bar{A})B\bar{C}\bar{D} \\
 &= \bar{A}\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}\bar{D} + A\bar{B}C\bar{C}\bar{D} + A\bar{B}C\bar{D}\bar{D} + A\bar{B}C\bar{D}\bar{D} + A\bar{B}C\bar{D}\bar{D} \\
 &= \sum m(5, 8, 9, 10, 11, 13)
 \end{aligned}$$

按照上式,将 $\sum m(5, 8, 9, 10, 11, 13)$ 中的对应的 6 个方格填 1,其余填 0,得到的卡诺图如图 2.6.4 所示。

只要是与-或表达式,也可以直接填图。方法是:把每一个与项所属于的方格都填上 1,其余填 0。例如本例中,与项 $A\bar{B}$ 应该包含既属于 A 同时又属于 $\bar{B}$ 的方格。从图 2.6.4 可知,只有最下边一排四个格子既属于 A 同时又属于 $\bar{B}$,所以最下边四个格子都填 1,而与项 $B\bar{C}\bar{D}$ 则是既属于 B 又同时属于 $\bar{C}$ 和 $\bar{D}$ 的方格只有第 2 列中间的两个,把它们填上 1,得到的结果与先化成最小项表达式再填图完全一样,就是图 2.6.4。

(3) 如果逻辑表达式不是“与-或表达式”,可先将其化成“与-或表达式”,再填入卡诺图。

【例 2.6.4】 画出 $L(A, B, C, D) = (\bar{A}\bar{B} + \bar{C}\bar{D})(\bar{A}\bar{B} + C\bar{D})$ 的卡诺图。

解 现将函数化成与-或表达式:

$$\begin{aligned}
 L(A, B, C, D) &= \bar{A}\bar{B} + \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} \\
 &= \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} \\
 &= \sum m(0, 1, 2, 3)
 \end{aligned}$$

本题按照 $\sum m(0, 1, 2, 3)$ 的四个最小项,在最上边一排四个格子里填 1,其余填 0。但是在化为与或表达式时,就可以直接填图,不一定要进一步化成最小项表达式。例如,把属于 $\bar{A}\bar{B}$ 的最上边一排四个格子先填上 1,另外两项分别是第一行的第一格和第四格,显然已经填上,再填该格子依然是 1,故结果和化成最小项再填图的结果是一样的,如图 2.6.5 所示。

$\textcircled{L}$ AB	CD	00	01	11	10
	00	1	1	1	1
	01	0	0	0	0
	11	0	0	0	0
	10	0	0	0	0

图 2.6.5 例 2.6.4 的卡诺图

$\textcircled{L}$ AB	CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
	$\bar{A}\bar{B}$	m_0	m_1	m_3	m_2
	$\bar{A}B$	m_4	m_5	m_7	m_6
	AB	m_{12}	m_{13}	m_{15}	m_{14}
	$A\bar{B}$	m_8	m_9	m_{11}	m_{10}

图 2.6.6 卡诺图化简原理

2.6.3 逻辑函数的卡诺图化简法

1. 卡诺图化简逻辑函数的原理

由于卡诺图具有循环邻接的特性,若图中两个相邻的格子均为 1,则两个最小项可以合并为一项,同时消去一个变量;若图中 4 个相邻的格子均为 1,则 4 个最小项合并为一项,同时消去 2 个变量;8 个相邻的格子均为 1,则 8 个最小项合并为一项,同时消去 3 个变量,以此类推。我们只要把个数为 2^n 的相邻的方格用一个包围圈圈起来,这样就可以按照每个包围圈合并化简为 1 项,如图 2.6.6 所示。

图中:

$$\bar{A}\bar{B}CD + \bar{A}\bar{B}C\bar{D} = \bar{A}\bar{B}C \quad (m_3 \text{ 和 } m_2 \text{ 合并})$$

$$\bar{A}BCD + \bar{A}BC\bar{D} = \bar{A}BC \quad (m_7 \text{ 和 } m_6 \text{ 合并})$$

$$\bar{A}BC + \bar{A}\bar{B}C = \bar{A}C \quad (m_3, m_2 \text{ 和 } m_7, m_6 \text{ 合并})$$

$$ABC + A\bar{B}C = AC \quad (m_{10}, m_{11}, m_{14}, m_{15} \text{ 合并})$$

$$AC + \bar{A}C = C \quad (m_2, m_3, m_6, m_7, m_{10}, m_{11}, m_{14}, m_{15} \text{ 合并})$$

总之, 2^n 个相邻的最小项结合,可以消去 n 个取值不同的变量而合并为 1 项。

2. 用卡诺图合并最小项的原则

用卡诺图化简逻辑函数,就是在卡诺图中找相邻的最小项,即画圈。常见的 2 个、4 个、8 个最小项相邻合并的画圈方法如图 2.6.7 所示。

为了保证将逻辑函数化到最简,画圈时必须遵循以下原则:

- (1) 圈要尽可能大,这样消去的变量就多。但是每个圈内只能含有相邻项 2^n 个 ($n=0, 1, 2, 3, \dots$)。要特别注意对边相邻性和四角相邻性。
- (2) 圈的个数尽量少,这样化简后的逻辑函数的与项就少。
- (3) 卡诺图中所有取值为 1 的方格均要被圈过,即不能漏下取值为 1 的最小项。
- (4) 取值为 1 的方格可以被重复圈在不同的包围圈中,但在新画的包围圈中至少要含有 1 个未被圈过的 1 的方格,否则该包围圈是多余的。

3. 用卡诺图化简逻辑函数的步骤

- (1) 画出逻辑函数的卡诺图。

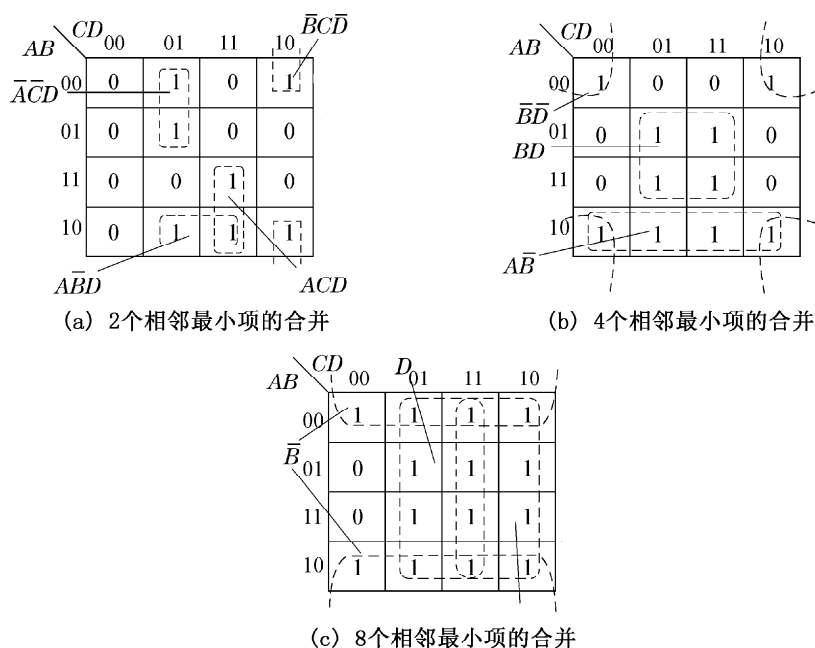


图 2.6.7 相邻最小项的画圈合并

(2) 合并相邻的最小项,即根据前述原则画圈。

(3) 写出化简后的表达式。每一个圈写一个最简与项,规则是,取值为1的变量用原变量表示,取值为0的变量用反变量表示,将这些变量相与。然后将所有与项进行逻辑加,即得最简与-或表达式。

【例 2.6.5】 用卡诺图化简逻辑函数:

$$L(A,B,C,D) = \sum m(0,2,3,4,6,7,10,11,13,14,15)$$

解 (1) 由表达式画出卡诺图如图 2.6.8(a)所示。

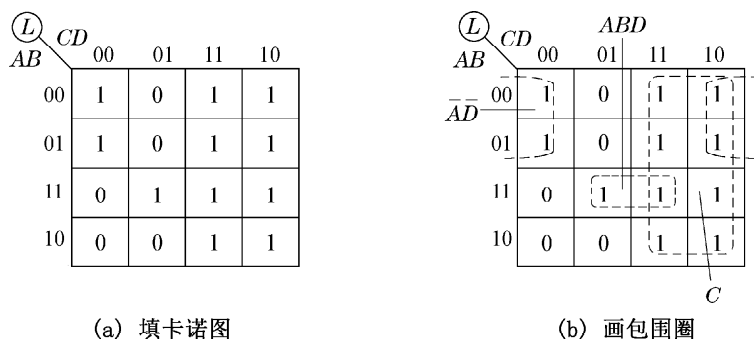


图 2.6.8 例 2.6.5 的卡诺图化简

(2) 画包围圈合并最小项,如图 2.6.8(b)所示,得简化的与-或表达式:

$$L = C + \overline{A}\overline{D} + ABD$$

注意:图中的包围圈 $\overline{A}\overline{D}$ 是利用了对边相邻性。

【例 2.6.6】用卡诺图化简逻辑函数： $L=AD+A\bar{B}\bar{D}+\bar{A}\bar{B}CD+\bar{A}\bar{B}C\bar{D}$ 。

解 (1) 由表达式画出卡诺图如图 2.6.9(a)所示。

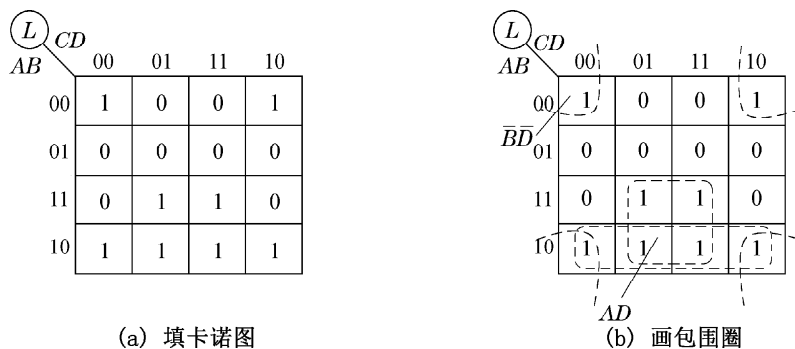


图 2.6.9 例 2.6.6 的卡诺图化简

(2) 画包围圈合并最小项,如图 2.6.9(b)所示,得简化的与-或表达式:

$$L=AD+\bar{B}\bar{D}$$

注意:图中的长方形虚线圈是多余的,应去掉(最简逻辑表达式中不写这一项);图中的包围圈 $\bar{B}\bar{D}$ 是利用了四角相邻性。

【例 2.6.7】某逻辑函数的真值表如表 2.6.2 所示,用卡诺图化简该逻辑函数。

表 2.6.2 例 2.6.7 真值表

A	B	C	L
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

解法 1 (1) 由真值表画出卡诺图,如图 2.6.10 所示。

(2) 画包围圈合并最小项,如图 2.6.10(a)所示,得简化的与-或表达式:

$$L=\bar{B}C+\bar{A}B+A\bar{C}$$

解法 2 (1) 由表达式画出卡诺图,如图 2.6.10 所示。

(2) 画包围圈合并最小项,如图 2.6.10(b)所示,得简化的与-或表达式:

$$L=A\bar{B}+B\bar{C}+\bar{A}C$$

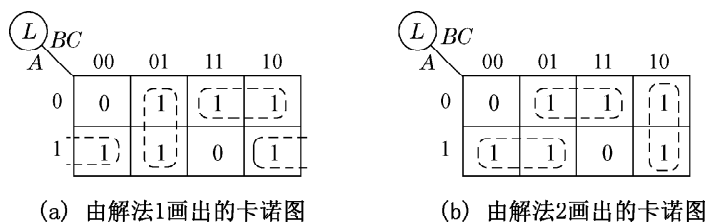


图 2.6.10 例 2.6.7 卡诺图

本例表明,一个逻辑函数的真值表是唯一的,卡诺图也是唯一的,但是包围圈可以有不同的画法,所以得到的化简结果有时不是唯一的。

4. 卡诺图化简逻辑函数的另一种方法——圈0法

当一个逻辑函数用卡诺图表示后,里面的0很少且相邻性很强,这时用圈0法更简便。但要注意,圈0后,先写出反函数 $\bar{L}$,再取非,才能得到原函数 L 。

【例 2.6.8】 已知逻辑函数的卡诺图如图 2.6.11 所示,分别用“圈0法”和“圈1法”写出其最简与-或式。

解 (1) 用圈0法画包围圈如图 2.6.11(a)所示,得

$$\bar{L} = B\bar{C}\bar{D}$$

对 $\bar{L}$ 取非,得

$$L = \overline{B\bar{C}\bar{D}} = \bar{B} + \bar{C} + D$$

(2) 用圈1法画包围圈如图 2.6.11(b)所示,得

$$L = \bar{B} + \bar{C} + D$$

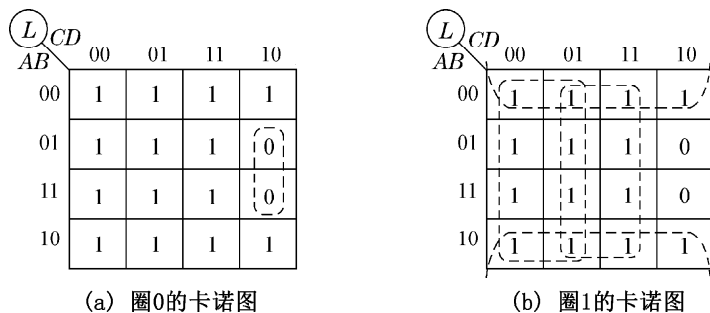


图 2.6.11 例 2.6.8 的卡诺图

5. 具有无关项的逻辑函数的化简

(1) 什么是无关项

逻辑函数所有 2^n 个最小项中有时会有一些最小项是受约束的项(不允许出现)或者是任意项(有这些项还是无这些项对逻辑函数没有影响),这些约束项和任意项统称无关最小项或称无关项。例如,8421BCD码中的1010~1111就是不允许出现的约束项。由于无关项在逻辑函数中要么不会出现,要么对逻辑函数无影响,因此这些无关最小项在卡诺图中相应的方格中是1或是0都无所谓。在填卡诺图时,这些无关最小项在相应的方格中填“×”,以示区别。在画包围圈时,可把“×”当1看待,也可把“×”当0看待。究竟把“×”当1还是

当 0, 应根据化简需要而定。

【例 2.6.9】 在十字路口有红绿黄三色交通信号灯, 规定红灯亮停, 绿灯亮行, 黄灯亮等一等, 试分析车行与三色信号灯之间逻辑关系。

解 设红、绿、黄灯分别用 A 、 B 、 C 表示, 且灯亮为 1, 灯灭为 0。车用 L 表示, 车行 $L=1$, 车停 $L=0$ 。列出该函数的真值表见表 2.6.3。

表 2.6.3 例 2.6.9 的真值表

红灯 A	绿灯 B	黄灯 C	车 L
0	0	0	×
0	0	1	0
0	1	0	1
0	1	1	×
1	0	0	0
1	0	1	×
1	1	0	×
1	1	1	×

显而易见, 在这个函数中, 有 5 个最小项是不会出现的, 如 $\overline{ABC}$ (三个灯都不亮)、 $AB\overline{C}$ (红灯绿灯同时亮) 等。因为一个正常的交通灯系统不可能出现这些情况, 如果出现了, 车可以行也可以停, 即逻辑值任意。

带有无关项的逻辑函数的最小项表达式为

$$L = \sum m(\quad) + \sum d(\quad)$$

如本例函数可写成 $L = \sum m(2) + \sum d(0, 3, 5, 6, 7)$ 。

(2) 具有无关项的逻辑函数的化简

化简具有无关项的逻辑函数时, 要充分利用无关项可以当 0 也可以当 1 的特点, 尽量扩大卡诺圈, 使逻辑函数更简单。

画出例 2.6.9 的卡诺图如图 2.6.12 所示。如果不考虑无关项, 包围圈只能包含一个最小项, 如图 2.6.12(a) 所示, 写出表达式为 $L = \overline{A}B\overline{C}$; 如果把与它相邻的三个无关项当作 1, 则包围圈可包含 4 个最小项, 如图 2.6.12(b) 所示, 写出表达式为 $L = B$, 其含义为: 只要绿灯亮, 车就行。

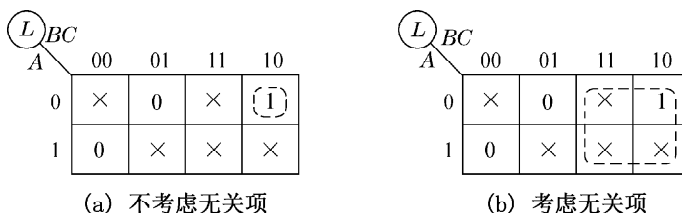


图 2.6.12 例 2.6.9 的卡诺图

注意:在考虑无关项时,哪些无关项当作1,哪些无关项当作0,要以尽量扩大卡诺圈、减少圈的个数,使逻辑函数更简单为原则。

【例 2.6.10】 某逻辑函数输入是 8421BCD 码(即不可能出现 1010~1111 这 6 种输入组合),其逻辑表达式为: $L(A,B,C,D) = \sum m(1,4,5,6,7,9) + \sum d(10,11,12,13,14,15)$,用卡诺图法化简该逻辑函数。

解 (1) 画出四变量卡诺图,如图 2.6.13(a)所示。将 1、4、5、6、7、9 号小方格填入 1;将 10、11、12、13、14、15 号小方格填入 $\times$ 。

(2) 合并最小项。与 1 方格圈在一起的无关项被当作 1,没有圈的无关项被当作 0。注意:1 方格不能漏; $\times$ 方格根据需要,可以圈入,也可以放弃。

(3) 写出逻辑函数的最简与-或表达式: $L = B + \overline{CD}$ 。

如果不考虑无关项,如图 2.6.13(b)所示,写出表达式为 $L = \overline{AB} + \overline{BCD}$,可见不是最简。

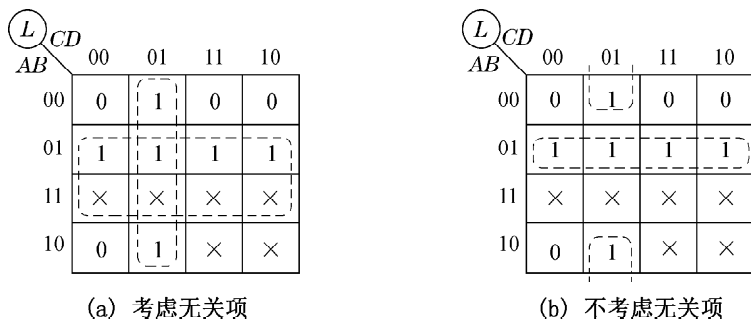


图 2.6.13 例 2.6.10 的卡诺图

卡诺图化简法的优点是简单、直观,有一定的化简步骤可循,且容易化到最简,化简结果可靠。但是当逻辑变量超过 5 个时,就失去了简单、直观的优点。

本章小结

与、或、非运算是逻辑代数中的三种基本逻辑运算,由它们可以组合或演变成几种基本的复合逻辑运算,如与非、或非、异或、同或和与或非等。本章的主要内容是逻辑代数的公式和定律、逻辑函数的表示方法和逻辑函数的化简方法。

为了进行逻辑运算,必须熟练掌握表 2.3.1 中的基本定律。而表 2.3.4 中的常用公式,可以由基本定律导出。

逻辑函数有五种常用的表示方法,即真值表、逻辑表达式、逻辑图、波形图和卡诺图。它们各有其特点:真值表全面反映事物的逻辑关系,一目了然;逻辑表达式便于运算和演变;逻辑图接近工程实际;波形图便于分析和测试;卡诺图便于化简。这五种表示方法可以任意地互相转换,其转换方法是分析和设计数字电路的必要工具,在实际中可根据需要选用。

逻辑函数的化简是分析、设计数字电路的重要环节。实现同样的功能,电路越简单,成本就越低而且工作越可靠。公式化简法的优点是其使用不受任何条件的限制。但由于这种方法没有固定的步骤可循,所以在化简一些复杂的逻辑函数时不仅需要熟练地运用各种公式和定律,而且需要有一定的运算技巧和经验,有时还难于判断化简的结果是否最简。

卡诺图是逻辑函数的图形表示法。它利用几何位置的相邻性直观地表示了逻辑函数各最小项之间在逻辑上的相邻性。卡诺图化简法的优点是简单、直观,而且有一定的化简步骤可循。初学者容易掌握这种方法,而且化简过程中也便于检查化简结果的准确性,避免差错。然而在逻辑变量超过5个时,将失去简单、直观的优点。在利用卡诺图化简逻辑函数时,要充分、合理地利用无关项。

在设计数字电路的过程中,有时由于受电子器件供货的限制只能选用某种逻辑功能类型的器件,这时就需要将逻辑函数式变换为与之对应的形式。而在使用器件的逻辑功能类型不受限制的情况下,为了减少所用器件的数目,往往不限于使用单一逻辑功能的门电路。这时希望得到的最简逻辑表达式可能既不是单一的与或式,也不是单一的与非式,而是一种混合的形式。因此,究竟将函数式化成什么形式最为有利,还要根据选用哪些种类的电子器件而定。

自我检测题

一、填空题

1. 逻辑代数中最基本的运算是_____、_____、_____。
2. 逻辑函数 $L=A+B+C$ 的反函数是_____。
3. 逻辑函数 $L=B+A+C$ 的对偶式是_____。
4. 逻辑函数 $L=ABC$ 的反演式是_____。
5. 同或门的逻辑图为_____。
6. n 个逻辑变量构成的最小项有_____个。
7. 实现 $L=A\oplus B$ 至少需要使用_____个两输入端的与非门。
8. 逻辑函数 $L(A,B,C,D) = \sum m(0,2,3,4,8) + \sum d(10,11,12,13,14,15)$ 的最简与或表达式为_____。

二、选择题

1. 逻辑函数 $L=A\oplus(A\oplus B)=$ _____。
A. B B. A C. $A\oplus B$ D. $A \cdot B$
2. 逻辑函数 $L=A\bar{B}+BC$, 使 $L=1$ 的 ABC 的取值组合为_____。
A. $ABC=000$ B. $ABC=010$ C. $ABC=101$ D. $ABC=110$
3. 全体最小项之和为_____。
A. A B. $\bar{A}$ C. 1 D. 0
4. 任何两个最小项的乘积为_____。
A. 1 B. 0 C. $\bar{AB}$ D. $\overline{A+B}$
5. 使 $L(A,B,C,D)=\bar{A}B+C\bar{D}$ 为 1 的最小项有_____个。
A. 8 B. 7 C. 6 D. 5
6. 下列_____是逻辑变量 A,B,C,D 中的最小项。
A. $\bar{A}BC$ B. $ABCD$ C. BC D. $\bar{A}B+CD$
7. 将函数式 $L(A,B,C)=\bar{A}BC+AC+\bar{B}C$ 化成最小项之和的形式为_____。

- A. $L = \sum m(1,3,4,7)$ B. $L = \sum m(1,2,4,7)$
 C. $L = \sum m(1,3,6,7)$ D. $L = \sum m(1,3,5,7)$
8. 函数 $L=AB+BC+AC$ 转换成与非-与非式为_____。
- A. $L=\overline{\overline{AB}BCAC}$ B. $L=\overline{AB+BC+AC}$
 C. $L=\overline{\overline{AB+BC+AC}}$ D. $L=\overline{ABBCAC}$

练习题

2.1 列真值表证明下列各式。

- (1) $\overline{\overline{AB}}=A+B$
 (2) $\overline{\overline{A+B}}=AB$
 (3) $\overline{A\overline{B}+\overline{AB}}=\overline{AB}+AB$
 (4) $\overline{\overline{AB}+AB}=A\overline{B}+\overline{AB}$
 (5) $\overline{AB+CD}=(\overline{A+B})(\overline{C+D})$
 (6) $A+B+C=\overline{\overline{ABC}}$

2.2 用逻辑代数基本定律和常用公式证明下列等式。

- (1) $ABC+A\overline{B}C+AB\overline{C}=AB+AC$
 (2) $A+A\overline{B}C+\overline{ACD}+(\overline{C+D})E=A+CD+E$
 (3) $A\overline{B}+BD+DCE+D\overline{A}=A\overline{B}+D$
 (4) $A+A\overline{B}(B+C)=A+\overline{BC}$
 (5) $\overline{AB+\overline{AB}+C}=(A\oplus B)C$
 (6) $A\oplus B\oplus C=ABC+(A+B+C)\overline{AB+BC+CA}$
 (7) $BC+AD=(B+A)(B+D)(A+C)(C+D)$
 (8) $A\oplus B\oplus C=A\odot B\odot C$
 (9) $ABC+A\overline{B}C+\overline{AB}C+\overline{ABC}=A\oplus B\oplus C$
 (10) $A+A\overline{B}C+\overline{ACD}+(\overline{C+D})E=A+CD+E$

2.3 写出函数 $Y(ABCD)=\overline{AB+C}\cdot D+AC$ 的对偶式和反函数。

2.4 写出如题表 2.6 所示真值表中各个函数的逻辑表达式,并画出逻辑图。

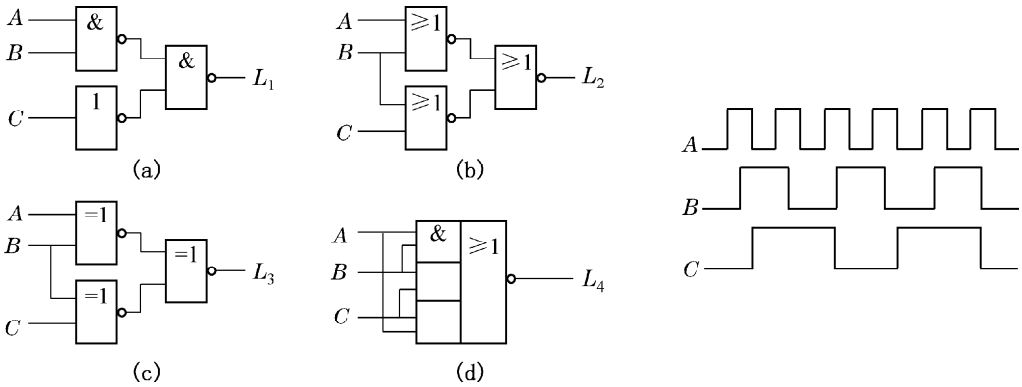
题表 2.4

A B C	L_1	L_2	L_3	L_4
0 0 0	0	0	0	0
0 0 1	0	1	0	1
0 1 0	1	1	0	1

(续表)

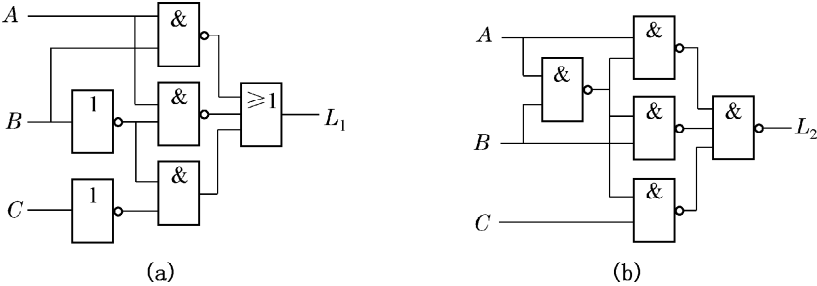
A B C	L_1	L_2	L_3	L_4
0 1 1	0	0	1	1
1 0 0	1	1	0	0
1 0 1	0	0	1	0
1 1 0	1	0	1	0
1 1 1	0	1	1	1

2.5 写出题图 2.5 各个电路输出信号的逻辑表达式,并对应 A、B、C 的波形画出各个输出信号的波形。



题图 2.5

2.6 写出如题图 2.6 所示各逻辑图的输出函数表达式,并列出真值表。



题图 2.6

2.7 已知逻辑函数式为 $L = A + \overline{B}C + \overline{A}B\overline{C}$, 写出它对应的真值表并画出逻辑电路图。

2.8 将下列逻辑函数表达式用最小项的形式表示。

- (1) $L = A\overline{C} + \overline{A}B + CB$
- (2) $L = BC + \overline{A}B + A\overline{C}$
- (3) $L = A\overline{C}D + \overline{B}C\overline{D} + ABCD$

2.9 将下列逻辑函数用最大项表达式表示。

$$(1) L = AB + \overline{A}C$$

$$(2) L = \overline{A}\overline{B} + \overline{B}C + A\overline{C}$$

$$(3) L = A\overline{B} + ACD + \overline{B}\overline{C}\overline{D}$$

2.10 应用逻辑代数基本定律和常用公式化简下列逻辑表达式。

$$(1) L = A + ABC + \overline{A}BC$$

$$(2) L = A + \overline{\overline{A}DB} + \overline{\overline{B} + \overline{CD}}$$

$$(3) L = A\overline{B}CD + ABCD + A\overline{C}D$$

$$(4) L = (AB + A\overline{B} + \overline{A}D)(A + B + \overline{A}BD)$$

$$(5) L = \overline{A \oplus B \oplus C} \cdot \overline{B \oplus C}$$

$$(6) L = (A + B + \overline{C})(A + B + C)$$

$$(7) L = (A + B)C + \overline{A}C + AB + ABC + \overline{B}C$$

$$(8) L = A\overline{B}\overline{C}D + AB\overline{C}D + A\overline{B} + A\overline{D} + A\overline{B}C$$

2.11 函数 L 的真值表如题表 2.11 所示。试求：

(1) 分别写出用最小项编号和最大项编号表示的 L 的表达式。

(2) 分别写出用逻辑变量 A, B, C 表示的最小项之和、最大项之积的表达式。

(3) 给出用最小项编号表示的 $\overline{L}$ 的表达式。

(4) 用公式法化简, 得到 L 的最简与或式, 并画出与非门实现的逻辑图。

题表 2.11

A	B	C	L
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

2.12 用卡诺图表示下述逻辑函数, 并化简成最简与-或式。

$$(1) L(A, B, C, D) = \sum m(1, 2, 3, 4, 5, 6)$$

$$(2) L(A, B, C, D) = \sum m(1, 5, 6, 7, 11, 12, 13, 14, 15)$$

$$(3) L(A, B, C, D) = \sum m(0, 1, 2, 5, 6, 7, 14, 15)$$

$$(4) L(A, B, C, D) = \sum m(0, 2, 5, 7, 8, 11, 13, 15)$$

$$(5) L(A, B, C, D) = \overline{A}\overline{B}C + AD + B\overline{D} + C\overline{D} + A\overline{C}$$

$$(6) L(A, B, C, D) = \overline{B}CD + B\overline{C} + \overline{A}CD + A\overline{B}C$$

$$(7) L(A, B, C, D) = \overline{A}\overline{B} + \overline{B}\overline{C} + AC + \overline{B}C$$

$$(8) L(A, B, C, D) = B + \overline{\overline{A + \overline{CD + \overline{ABD}}}}$$

2.13 用卡诺图法化简下列具有无关项的逻辑函数为最简与-或表达式。

$$(1) L(A, B, C, D) = \sum m(0, 2, 3, 4, 8) + \sum d(10, 11, 12, 13, 14, 15)$$

$$(2) L(A, B, C, D) = \sum m(5, 6, 7, 8, 9) + \sum d(10, 11, 12, 13, 14, 15)$$

$$(3) L(A, B, C, D) = \sum m(1, 5, 8, 12) + \sum d(3, 7, 10, 11, 14, 15)$$

$$(4) L(A, B, C, D) = \sum m(2, 4, 6, 8, 9) + \sum d(12, 13, 14, 15)$$

$$(5) L(A, B, C, D) = \sum m(0, 2, 4, 5, 6, 11, 12) + \sum d(8, 9, 10, 13, 14, 15)$$

$$(6) L(A, B, C, D) = \sum m(1, 3, 5, 8, 9, 13) + \sum d(7, 10, 11, 14, 15)$$

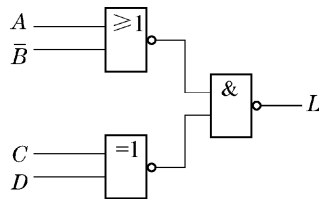
2.14 门电路组成如题图 2.14 所示逻辑电路。

(1) 按图直接写出 L 的表达式(不化简)。

(2) 根据反演规则, 写出 L 的反函数 $\overline{L}$ 。

(3) 根据对偶规则, 写出 L 的对偶函数 L' 。

(4) 将函数 L 化简为最简与-或表达式。



题图 2.14

2.15 (1) 已知逻辑函数 $L(A, B, C, D) = \overline{A}\overline{C}\overline{D} + \overline{A}B\overline{C} + \overline{B}C\overline{D}$, 约束条件为 $AB + A\overline{C} = 0$, 写出 L 的最简与非-与非式。

(2) 已知逻辑函数 $L(A, B, C, D) = \sum m(0, 2, 6, 8)$, 约束条件为 $AB + AC = 0$, 写出 L 的最简与-或式。

第3章 集成逻辑门电路

本章学习目的和要求

1. 熟悉半导体器件的开关特性,了解 TTL 门电路、ECL 门电路、CMOS 门电路的结构特点、工作原理和电气特性;
2. 掌握 TTL、CMOS 基本逻辑门、OC 门(OD 门)、三态门和传输门的逻辑功能;
3. 掌握门电路逻辑功能的分析方法;
4. 熟悉常用集成逻辑门电路的主要参数指标和选用方法;
5. 掌握逻辑门在应用中的接口问题。

逻辑代数中的各种逻辑运算都是用相应的逻辑门电路实现的。逻辑门电路是组成数字电路的基本逻辑单元,门电路中的半导体器件一般工作在开关状态。本章首先介绍半导体器件的开关特性,然后重点讲述 TTL 和 CMOS 集成门电路的电路结构、工作原理、逻辑功能和特点,以应用为目的,着重讲述门电路的外部特性及使用方法,对内部工作过程只作一般介绍。最后介绍 TTL 门电路与 CMOS 门电路的接口问题。

3.1 概 述

用于实现基本逻辑运算和复合逻辑运算的单元电路称为逻辑门电路,简称门电路。与前面所讲的基本逻辑运算和复合逻辑运算相对应,常用的门电路在逻辑功能上有与门、或门、非门、与非门、或非门、与或非门、异或门、同或门等多种。

在数字电路中,用高、低电平分别表示二值逻辑的 1 和 0 两种逻辑状态。如果高电平表示逻辑 1,低电平表示逻辑 0,这种表示方法称为正逻辑;反之,如果高电平表示逻辑 0,低电平表示逻辑 1,则称这种表示方法为负逻辑。图 3.1.1 为正负逻辑示意图。

对于同一电路,可以采用正逻辑,也可以采用负逻辑。正逻辑和负逻辑两种体制不牵涉到逻辑电路本身的好坏问题。但是所选用的逻辑体制不同,同一电路实现的逻辑功能也就不同。如某逻辑电路输入电平 V_A 和 V_B 与输出电平 V_L 的对应关系如表 3.1.1 所示,其中 L 表示低电平, H 表示高电平。采用正逻辑得到的逻辑函数真值表如表 3.1.2 所示,所对应的输出逻辑表达式为 $L=A+B$;如果采用负逻辑,真值表如表 3.1.3 所示,其输出逻辑函数表达式为 $L=\overline{AB}$ 。本书如无说明,一律采用正逻辑。

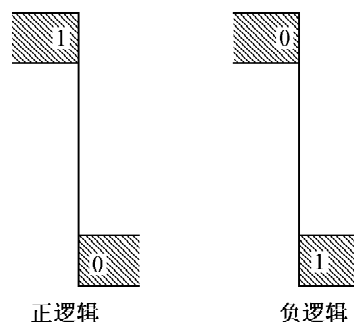


图 3.1.1 正逻辑与负逻辑示意图

表 3.1.1 电路输入、输出电平

V_A	V_B	V_L
L	L	H
L	H	L
H	L	L
H	H	L

表 3.1.2 正逻辑真值表

V_A	V_B	V_L
0	0	1
0	1	0
1	0	0
1	1	0

表 3.1.3 负逻辑真值表

V_A	V_B	V_L
1	1	0
1	0	1
0	1	1
0	0	1

由图 3.1.1 可以看出,表示逻辑 0、1 的低、高电平都有一个允许的范围,高电平有一下限值 $V_{H(\min)}$,低电平有一上限值 $V_{L(\max)}$ 。数字电路中,如果电平取值范围在 $V_{H(\min)}$ 和 $V_{L(\max)}$ 之间,则容易造成逻辑混乱,应当避免。在实际工作中只要能区分出高、低电平就可以了。正因为逻辑电路的高、低电平是一个允许的范围,数字电路中无论是对元器件参数精度的要求,还是对供电电源稳定度的要求,都比模拟电路要低一些。

最初的逻辑门电路都是用若干个分立的半导体器件和电阻、电容连接而成的,这种分立元件门电路组成大规模的数字电路是非常困难的,严重地制约了数字电路的普遍应用。目前这种分立元件门电路已经很少采用,但是学习它们有助于理解门电路的基本工作原理和逻辑功能。把大量的全部元件及连线制作在一块半导体芯片上的门电路称为集成逻辑门电路。随着集成电路工艺水平的不断提高,今天已经能够把大量的门电路集成在一块很小的半导体芯片上,构成功能复杂的“片上系统”,为数字电路的应用开辟了无限广阔的天地。

数字集成电路按所用半导体器件的不同,可分为双极型、单极型和混合型三种。在数字集成电路发展的历史过程中,首先得到推广应用的是双极型的 TTL(Transistor-Transistor Logic)电路。由于集成电路体积小、重量轻、可靠性好,因而在大多数领域里迅速取代了分立器件组成的数字电路。直到 20 世纪 80 年代初,这种采用双极型三极管组成的 TTL 型集成电路一直是数字集成电路的主流产品。

不过,TTL 电路的功耗比较大,这是它自身的一个严重缺点。因此,用 TTL 电路只能制作成中小规模集成电路。CMOS(Complementary Metal-Oxide-Semiconductor)集成电路出现于 20 世纪 60 年代后期,它最突出的优点在于功耗极低,所以非常适合于制作大规模集成电路。随着 CMOS 制作工艺的不断进步,无论在工作速度还是在驱动能力上,目前 CMOS 电路都不比 TTL 电路逊色。因此,CMOS 电路已经逐渐取代 TTL 电路而成为当前数字集成电路的主流产品。只是在现有的一些设备中仍旧在使用 TTL 电路,所以掌握 TTL 电路的基本工作原理和使用知识仍然是必要的。本章将重点讨论 TTL 和 CMOS 这两种目前应用较为广泛的集成逻辑门电路。

3.2 半导体器件的开关特性

在数字电路中,经常将半导体二极管、三极管和场效应管作为开关元件使用,它们在电路中的工作状态有时导通,有时截止,并能在信号的控制下进行两种状态的转换。这是一种非线性的大信号运用。一个理想的开关,接通时阻抗应为零,断开时阻抗应为无穷大,而这两个状态之间的转换应该是瞬间完成的。但实际上半导体开关器件在导通时具有一定的内阻,而截止时仍有一定的反向电流,又由于它本身具有惰性(如三极管中存在着势垒电容和

扩散电容,场效应管中存在着极间电容),因此,两个状态之间的转换需要时间,转换时间的长短反映了该器件开关速度的快慢。下面先讨论半导体二极管、三极管和 MOS 管的开关特性,然后再介绍逻辑门电路。

3.2.1 二极管的开关特性

半导体二极管是由一个 PN 结加上引出导线和管壳构成的半导体器件。由于二极管具有单向导通的特性,因此,在数字电路中,二极管可作为一个受电压控制的开关来使用。

1. 二极管的开关作用

二极管开关电路如图 3.2.1 所示。以硅二极管输入信号 v_i 为合适的跳变电压信号为例来说明二极管的开关作用,其中高电平为 V_{IH} ,低电平为 V_{IL} 。

当 $v_i = V_{IH}$ 时,二极管充分导通,产生较大的导通电流 i_D ,二极管两端电压约为 0.7 V,此时二极管相当于开关闭合,且有 0.7 V 压降,其等效电路如图 3.2.2(a) 所示。与理想开关相比,它有端电压 V_D ,即“开关”两端电压不为 0。

当 $v_i = V_{IL}$ 时,二极管因反向偏置而截止, $i_D \approx 0$,此时二极管相当于开关断开,其等效电路如图 3.2.2(b) 所示。与理想开关相比,它有反向饱和电流 I_S ,即流过“开关”的电流不为 0。

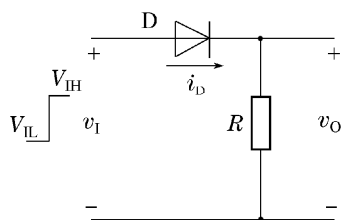


图 3.2.1 二极管开关电路

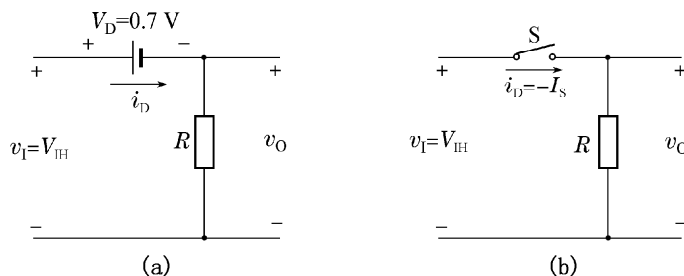


图 3.2.2 二极管工作在开关状态

图 3.2.3(a) 所示是硅二极管的伏安特性,图 3.2.3(b) 为被折线化的二极管的伏安特性,图 3.2.3(c) 所示是理想化的二极管的伏安特性。

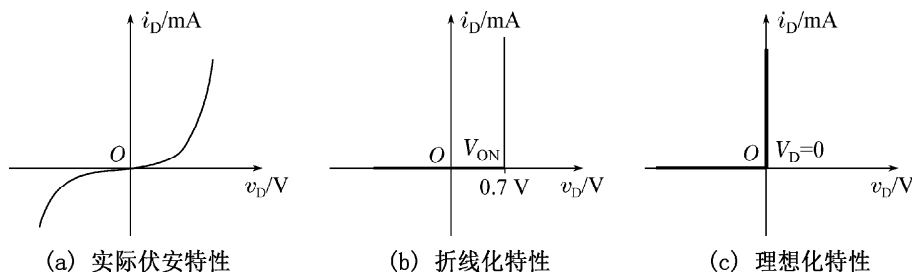


图 3.2.3 硅二极管伏安特性

2. 开关特性

在图 3.2.1 所示的二极管开关电路中,当输入信号 v_i 是跳变的电压时,流过二极管电流 i_D 的波形滞后于输入电压 v_i 的变化,如图 3.2.4 所示。

因为二极管由一个 PN 结组成,所以当外加输入信号突变时,其空间电荷区的电荷有一个积累和释放的过程,如同电容器的充、放电一样,表现出一定的电容效应,称其为结电容。由于结电容的存在,当外加电压由反向跳变为正向时,PN 结内部要建立起足够的电荷梯度才开始形成正向扩散电流,因而正向导通电流的建立要滞后于输入电压正跳变的时刻。二极管由反向截止转换为正向导通所需的时间,一般称为开启时间。通常二极管正向导通时电阻很小,与二极管内 PN 结等效电容(一般为 pF 量级)并联之后,电容作用不明显,所以转换时间很短,一般可以忽略不计。当外加电压由正向跳变为反向时,出于 PN 结内尚存在一定数量的存储电荷,所以有较大的瞬态反向电流,用 I_R 表示,随着存储电荷的释放。反向电流逐渐减小并趋近于零,最后稳定在一个微小的数值,用 I_s 表示,称作反向饱和电流。将反向电流从它的峰值衰减到它十分之一值所经过的时间用 t_{re} 表示,称作反向恢复时间。因为二极管反向截止时等效电阻很大,PN 结等效电容作用明显,充、放电时间较长,一般开关二极管的关断时间大约是几纳秒。

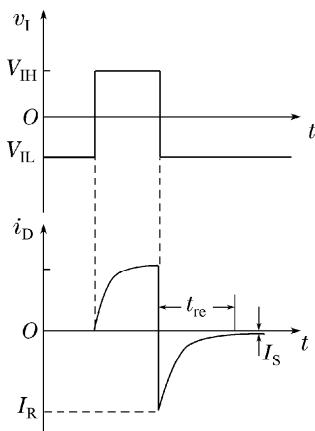


图 3.2.4 二极管的开关时间

由此可知,当二极管外加跳变电压且信号变化频率很高时,将因结电容的充、放电而失去单向导电特性,二极管将不能起到开关作用。因此,对二极管的最高工作频率应当有一定的限制。

3.2.2 三极管的开关特性

一个半导体三极管由管芯、三个引出电极和外壳组成。三个电极分别称为基极(base),集电极(collector)和发射极(emitter)。内部由三层 P 型和 N 型半导体结合在一起而构成,有 NPN 型和 PNP 型两种。因为在工作时有电子和空穴两种载流子参与导电过程,故称这类三极管为双极型三极管(Bipolar Junction Transistor, BJT)。

1. 三极管的开关作用

半导体三极管能当作开关使用,图 3.2.5 是三极管基本开关电路。三极管有三个工作区:截止、放大、饱和。当工作在饱和区时,管压降很小,接近于短路;当工作在截止区时,反向电流很小,接近于断路。若把三极管的开关作用对应于有触点开关的“断开”和“闭合”,所以只要使三极管工作在饱和区和截止区,就可以把它看成开关的通、断两个状态。二极管是用其正极和负极两极作为开关的两端接在电路中,开关的通、断受其两端电路控制,而三极管(以共发射极电路为例)是用其集电极、发射极两极作为开关的两端接在电路里,其开关的通、断则受基极控制。

下面以 NPN 型硅管为例来说明三极管的放大作用,三极管基本开关电路如图 3.2.5 所示。

当输入电压 v_i 为低电平 V_{IL} 时,使基极和发射极之间的电压 v_{BE} 小于其门限电压,此时 BJT 发射结和集电结均为反向偏置,只有很小的反向漏电流 I_{EBO} 和 I_{CBO} 分别流过两个结,故 $i_B \approx 0, i_C \approx 0, v_{CE} \approx V_{CC}$, 对应于图 3.2.6 中的 A 点。这时集电极回路中的 c、e 极之间近似于开路,相当于开关断开一样,三极管的这种工作状态称为截止。

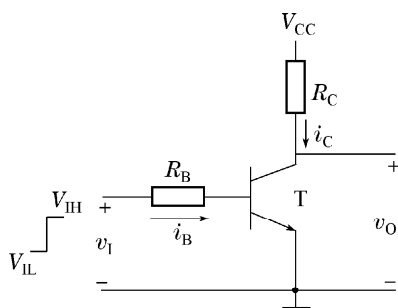


图 3.2.5 三极管基本开关电路

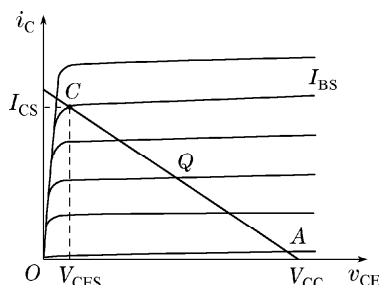


图 3.2.6 三极管输出特性曲线

当输入电压 v_i 为高电平 V_{IH} 时,调节 R_B ,使 $i_C \approx \frac{V_{CC}}{R_C}$, 则三极管工作在图 3.2.6 中的 C 点,集电极电流 i_C 已接近于最大值 $\frac{V_{CC}}{R_C}$, 由于 i_C 受到 R_C 的限制,它已不像放大区那样随着 i_B 的增加而成比例地增加了,此时集电极电流达到饱和,对应的基极电流称为基极临界饱和电流 I_{BS} , 而集电极电流称为集电极饱和电流 I_{CS} 。此后,如果再增加基极电流,但集电极电流基本上保持在 I_{CS} 而不再增加,集电极电压 $v_{CE} = V_{CC} - I_{CS}R_C$, 约 $0.2\text{ V} \sim 0.3\text{ V}$ 。这个电压称为三极管的饱和压降 V_{CES} , 它也基本上不随 i_B 增加而改变。由于 V_{CES} 很小,集电极回路中的 c、e 极之间近似于短路,相当于开关闭合一样。三极管的这种工作状态称为饱和。

2. 三极管的开关特性

三极管的开关过程和二极管一样,也是内部电荷“建立”和“消散”的过程。因此,三极管饱和与截止两种状态的相互转换也是需要一定的时间才能完成的。

在图 3.2.5 所示电路的输入端加入一个幅度在 V_{IH} 和 V_{IL} 之间变化的理想矩形脉冲如图 3.2.7(a), 则输出电流 i_C 的波形如图 3.2.7(b) 所示。

图中四个参数都是以集电极电流的变化为基准的,用来描述三极管的开关特性,称为三极管的开关时间参数:

延迟时间 t_d : 集电极电流从 0 上升到 10% 饱和电流所经历的时间。

上升时间 t_r : 集电极电流从饱和电流的 10% 上升到 90% 所经历的时间。

存储时间 t_s : 基极输入电压下降到低电平时起到集电极电流下降到饱和电流的 90% 时

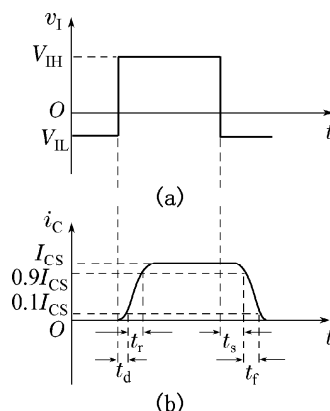


图 3.2.7 三极管的开关特性

的时间。

下降时间 t_f : 集电极电流从饱和电流的 90% 下降到 10% 所经历的时间。

通常把 $t_{on} = t_d + t_r$ 称为开通时间, 是三极管内部建立基区电荷的时间, 它反映了三极管从截止到饱和所需的时间; 把 $t_{off} = t_s + t_f$ 称为关闭时间, 是管子内部存储电荷消散的时间, 它反映的是三极管从饱和到截止所需的时间。

开通时间 t_{on} 和关闭时间 t_{off} 总称为三极管的开关时间, 而且 $t_{off} > t_{on}$ 。它随管子类型不同而有很大差别, 一般都在纳秒数量级, 可以从器件手册中查到。

三极管的开关时间限制了三极管开关运用的速度。开关时间越短, 开关速度越高。其中关闭时间 t_{off} 与工作时三极管饱和导通的深度—— I_B/I_{BS} 有关, 饱和程度越深即 I_B/I_{BS} 比值愈大, t_{off} 就越长, 反之则越短。所以, 加快三极管开关速度的一条重要措施, 就是限制三极管工作时的饱和深度, 即减小 I_B/I_{BS} 的比值。

3.2.3 MOS 管的开关特性

MOS 管是金属-氧化物-半导体场效应管的简称。MOS 管也有三个工作区: 截止区、非饱和区(也称可变电阻区)、饱和区(也称恒流区)。MOS 管作开关使用时, 通常工作在截止区和可变电阻区。在数字电路中, 用得最多的是 N 沟道增强型 MOS 管和 P 沟道增强型 MOS 管, 它们是构成 CMOS 数字集成电路的基本开关元件。由于 P 沟道增强型 MOS 管和 N 沟道增强型 MOS 管在结构上是对称的, 两者工作原理和特点也无本质区别, 只是在 PMOS 管中, 栅源电压 V_{GS} 、漏源电压 V_{DS} 、开启电压 V_{TP} 均为负值。下面以 N 沟道增强型 MOS 管为例, 说明 MOS 管的开关特性及工作特点。

1. MOS 管的开关作用

图 3.2.8(a) 所示为 NMOS 管组成的开关电路, 其实是 NMOS 管构成的反相器。 $v_I = v_{GS}$, $v_O = v_{DS}$, V_T 为开启电压。图 3.2.8(b) 为 NMOS 管的输出特性曲线, 其中斜线为直流负载线。

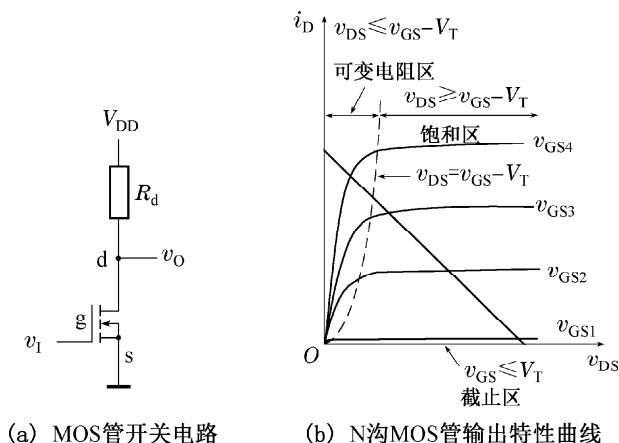


图 3.2.8 MOS 管开关电路及输出特性曲线

当 $v_i < V_T$ 时, MOS 管处于截止状态, $i_D = 0$, 输出电压 $v_o = V_{DD}$ 。

当 $v_i > V_T$, 并且比较大, 使得 $v_{DS} > v_{GS} - V_T$ 时, MOS 管工作在饱和区。随着 v_i 的增加, i_D 增加, v_{DS} 随之下降, MOS 管最后工作在可变电阻区。从特性曲线的可变电阻区可以看到, 当 v_{GS} 一定时, d、s 之间可近似等效为线性电阻。 v_{GS} 越大, 输出特性曲线越倾斜, 等效电阻越小。此时 MOS 管可以看成是一个受 v_{GS} 控制的可变电阻。 v_{GS} 的取值足够大时使得 R_d 远远大于 d、s 之间的等效电阻时, 电路输出为低电平。

由此可见, MOS 管相当于一个由 v_{GS} 控制的无触点开关, 当输入为低电平时, MOS 管截止, 相当于开关“断开”, 输出为高电平, 其等效电路如图 3.2.9(a) 所示; 当输入为高电平时, MOS 管工作在可变电阻区, 相当于开关“闭合”, 输出为低电平, 其等效电路如图 3.2.9(b) 所示。图中 R_{on} 为 MOS 管导通时的等效电阻, 大约在 $1\text{ k}\Omega$ 以内。

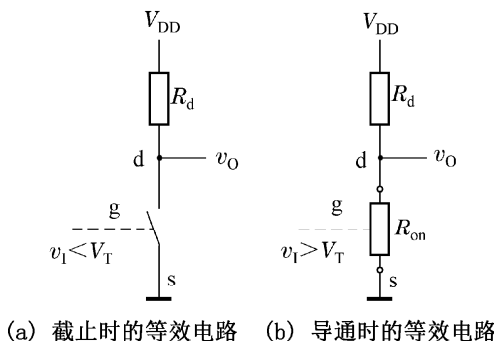


图 3.2.9 MOS 管开关等效电路

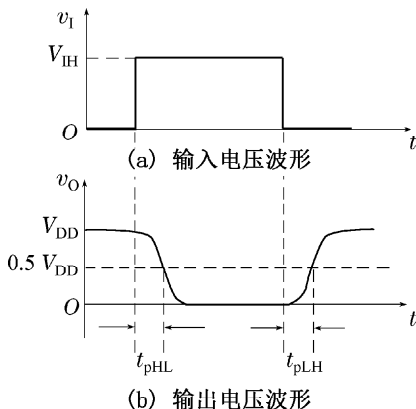


图 3.2.10 MOS 管开关电路波形

2. MOS 管的开关特性

在图 3.2.8(a) 所示 MOS 管的开关电路的输入端, 加一个理想的脉冲波形, 如图 3.2.10(a) 所示。由于 MOS 管是单极型器件, 它只有一种载流子参与导电, 没有超量存贮电荷存在, 也不存在存贮时间, 因而 MOS 管本身固有的开关时间是很小的, 它与由寄生电容造成的影响相比, 完全可以忽略。MOS 管中栅极与衬底间电容 C_{gb} (即数据手册中的输入电容 C_i)、漏极与衬底间电容 C_{db} 、栅极与漏极电容 C_{gd} 、布线电容等寄生电容构成了 MOS 管的输入和输出电容, 虽然这些电容很小, 但是因为 MOS 管输入电阻很高, 导通电阻也达几百欧姆, 负载的等效电阻也很大, 因而输入、输出电路的充、放电时间常数较大, 所以 MOS 管开关电路的开关时间, 主要取决于输入、输出电路的充、放电时间。这就使其在断开和闭合两种状态之间转换时, 不可避免地受到电容充、放电过程的影响。输出电压的波形已不是和输入一样的理想脉冲, 上升沿和下降沿都变得缓慢了, 而且输出电压的变化滞后于输入电压的变化。

由半导体开关器件可构成分立元件的门电路, 有关分立元件门电路知识可扫描二维码学习。



3.3 TTL 集成逻辑门电路

TTL 集成逻辑门电路自从问世几十年来,经过电路结构的不断改进和集成工艺的逐步完善,至今仍然广泛应用。因为这种类型电路的输入端和输出端均为三极管结构,所以称为三极管-三极管逻辑电路(Transistor-Transistor Logic),简称 TTL 电路。

3.3.1 TTL 与非门

1. TTL 与非门的电路结构及工作原理

(1) 电路结构

典型的 TTL 与非门电路可以把电路分成三部分:输入级、中间级(或称倒相级)和输出级,如图 3.3.1 所示。

首先,分析输入级。仔细分析后我们发现,电路中的 D_1 、 D_2 、 D_3 、 D_4 的 P 区是相连的,如图 3.3.2(a)所示。我们可用集成工艺将它们做成一个多发射极三极管,如图 3.3.2(b)所示。这样它既是四个 PN 结,不改变原来的逻辑关系,又具有三极管的特性。一旦满足了放大的外部条件,它就具有放大作用,为迅速消散 T_2 饱和时的超量存储电荷提供足够大的反向基极电流,从而大大提高了关闭速度。

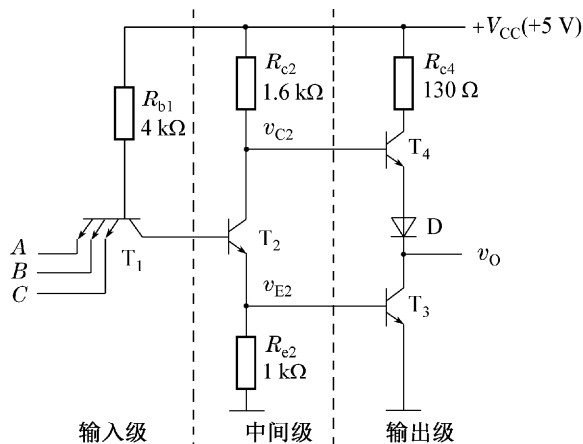


图 3.3.1 TTL 与非门电路

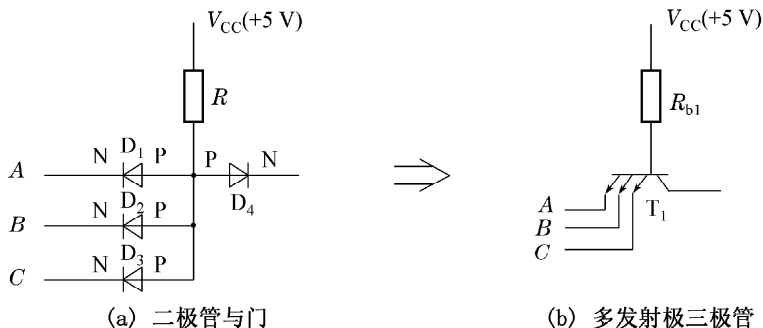


图 3.3.2 TTL 与非门的输入级

其次,分析中间级。为提高输出管的开通速度,由 T_2 、 R_{c2} 和 R_{e2} 组成中间级,它是一个三极管倒相电路,从 T_2 的集电极和发射极分别送出两个相位相反的信号驱动 T_3 和 T_4 管。同时在电路的开通过程中利用 T_2 的放大作用,为输出管 T_3 提供较大的基极电流,加速了输出管的导通。另外, T_2 和电阻 R_{c2} 、 R_{e2} 组成的放大器有两个反相的输出端 v_{c2} 和 v_{e2} ,以产生

两个互补的信号去驱动 T_3 、 T_4 组成的推拉式输出级。

最后,分析输出级。输出级应有较强的负载能力,为此将三极管的集电极负载电阻 R_c 换成由三极管 T_4 、二极管 D 和 R_{c4} 组成的有源负载。由于 T_3 和 T_4 受两个互补信号 v_{c2} 和 v_{E2} 的驱动,所以在稳态时,它们总是一个导通,另一个截止。这种结构,称为推拉式输出级。

(2) 工作原理

因为该电路的输出高低电平分别为 3.6 V 和 0.3 V,所以在下面的分析中假设输入高电平也分别为 3.6 V 和 0.3 V。

① 输入全为高电平 3.6 V

如果 v_{B1} 低于 3.6 V, T_1 的发射结反偏截止。令电源为 +5 V, T_1 的集电结和 T_2 、 T_3 的发射结串联, V_{CC} 经 R_{b1} 使三个 PN 结全部导通, $v_{B1} = 0.7 \times 3 = 2.1$ (V), 从而保证 T_1 的发射结因反偏而截止。电路中各点电压值如图 3.3.3 中所标注,此时 T_1 的发射结反偏,而集电结正偏,其偏置电压相当于把发射极与集电极颠倒,故又称为倒置放大工作状态。倒置状态下三极管的电流放大倍数很小,只有 0.01 倍左右。因 T_1 的发射结有一 1.5 V 的反偏电压,使得 T_1 的基极电流全部流入 T_2 的基极,使 T_2 、 T_3 饱和导通。

由于 T_3 饱和导通,输出电压为 $v_O = V_{CES3} \approx 0.3$ V。这时, $v_{E2} = v_{B3} = 0.7$ V, 而 $v_{CE2} = 0.3$ V, 故有 $v_{C2} = v_{E2} + v_{CE2} = 1$ V。1 V 的电压作用于 T_4 的基极, 小于 T_4 发射结和二极管 D 的两个 PN 结串联所需的导通电压, 所以 T_4 和二极管 D 都截止。于是实现了与非门的逻辑功能之一: 输入全为高电平时, 输出为低电平。我们称这时电路为导通状态, 也称为开门状态。

② 输入端有低电平 0.3 V

输入端为低电平 0.3 V 时, 输入端为 0.3 V 的发射结导通, T_1 的基极电位被钳位到 $V_{B1} = 1$ V。 T_1 饱和导通, 其基极电流经 R_{b1} 进入 T_1 的基极, 从低电平输入端流出, 其集电极电流从 T_2 基极流向输入端, 抽走了 T_2 基区的存储电荷。另外, 从电源经 R_{c2} 、 T_2 集电极流向 T_2 基极的是反向饱和电流, 其值很小, 故 T_1 深度饱和, 使 T_2 基极电位低于 0.6 V, 于是 T_2 、 T_3 都截止。由于 T_2 截止, 流过 R_{c2} 的电流仅为 T_4 的基极电流, 这个

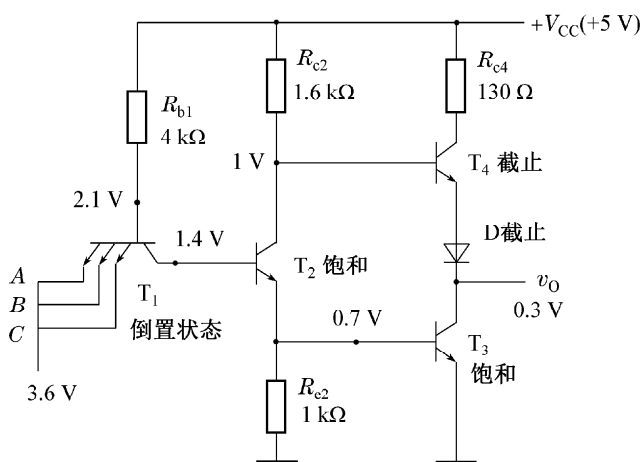


图 3.3.3 输入全为高电平时电路各点的工作电压

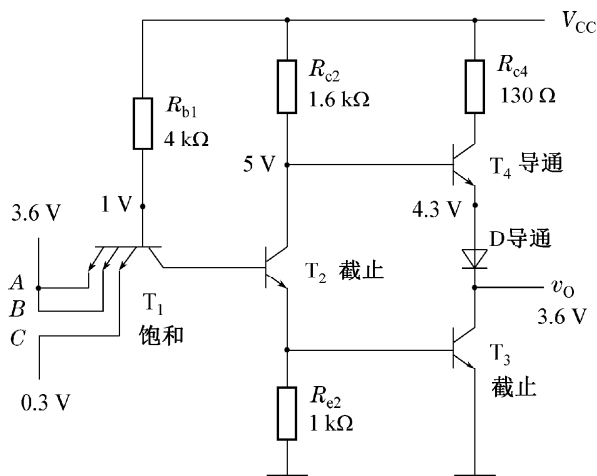


图 3.3.4 输入有低电平时电路各点的工作电压

电流较小,在 R_{c2} 上产生的压降也较小,可以忽略,所以 $v_{B1} \approx V_{CC} = 5\text{ V}$,使 T_4 和 D 导通,则有

$$v_O \approx V_{CC} - v_{BE4} - V_D = 5 - 0.7 - 0.7 = 3.6(\text{V})$$

可见,实现了与非门的逻辑功能的另一功能:只要输入有低电平时,输出就为高电平。我们称这时电路为截止状态,也称为关门状态。

综合上述两种情况,归纳成如表 3.3.1 所示。可见,该电路满足与非的逻辑功能, $L = \overline{ABC}$,是一个与非门电路。

表 3.3.1 TTL 与非门典型电路的工作状态

输入	输出	T_1	T_2	T_3	T_4	D	门的状态
有 0	为 1	深饱	截止	截止	导通	导通	关门
全 1	为 0	倒置	饱和	深饱	截止	截止	开门

2. TTL 与非门的开关速度

(1) TTL 与非门提高工作速度的原理

① 采用多发射极三极管加快了存储电荷的消散过程。如图 3.3.5 所示,设电路原来输出低电平,当电路的某一输入端突然由高电平变为低电平, T_1 的一个发射结导通, v_{B1} 变为 1 V。由于 T_2 、 T_3 原来是饱和的,基区中的超量存储电荷还来不及消散, V_{B2} 仍维持 1.4 V。在这个瞬间, T_1 为发射结正偏,集电结反偏,工作于放大状态,其基极电流 $i_{B1} = (V_{CC} - v_{B1})/R_{b1}$,集电极电流 $i_{C1} = \beta_1 i_{B1}$ 。这个 i_{C1} 正好是 T_2 的反向基极电流 i_{B2} ,可将 T_2 过剩的存储电荷迅速地拉走,促使 T_2 管迅速截止。 T_2 管迅速截止又使 T_4 管迅速导通,而使 T_3 管的集电极电流加大,使 T_3 的超量存储电荷从集电极消散而达到截止,正是由于 T_1 的放大作用加快了 T_3 从饱和到截止、 T_4 从截止到饱和的转换。

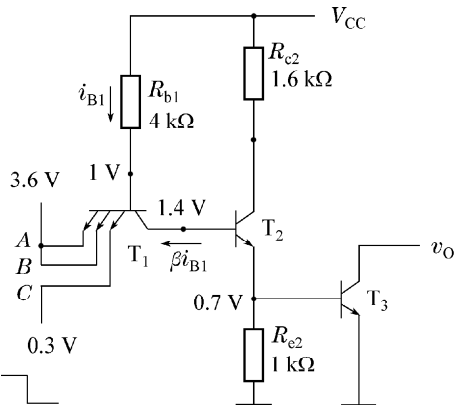


图 3.3.5 多发射极三极管消散 T_2 存储电荷的过程

② 采用了推拉式输出级,输出阻抗比较小,加快了给负载电容充、放电的进程。

输出端接有负载电容 C_L 的情况如图 3.3.6 所示。当输出电压由低变高时, T_3 截止, T_4 导通, T_4 为发射极输出器,输出电阻小,可使 C_L 迅速充电,故上升沿好。当输出电压由高变低时, T_3 饱和导通,其电流完全用来驱动负载,而且 T_3 等效为一个小电阻, C_L 通过 T_3 的饱和和小电阻放电也很快,故下降沿也好。推拉式输出级明显提高了对负载电容 C_L 的充、放电速度。

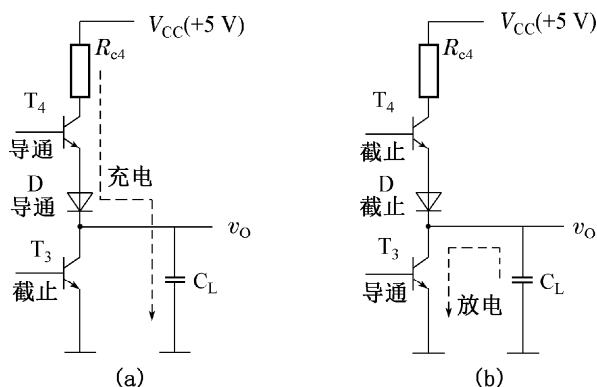


图 3.3.6 推拉式输出级给负载电容充、放电

(2) TTL 与非门传输延迟时间 t_{pd}

TTL 与非门的开关速度常用传输延迟时间 t_{pd} 来表示。当与非门输入一个脉冲波形时,其输出波形有一定的延迟,如图 3.3.7 所示。定义以下两个延迟时间:

导通延迟时间 t_{PHL} :从输入波形上升沿的中点到输出波形下降沿的中点所经历的时间。

截止延迟时间 t_{PLH} :从输入波形下降沿的中点到输出波形上升沿的中点所经历的时间。

与非门的传输延迟时间 t_{pd} 是 t_{PHL} 和 t_{PLH} 的平均值,即

$$t_{pd} = \frac{t_{PLH} + t_{PHL}}{2} \quad (3.3.1)$$

一般 TTL 与非门传输延迟时间 t_{pd} 的值为几纳秒~十几纳秒。

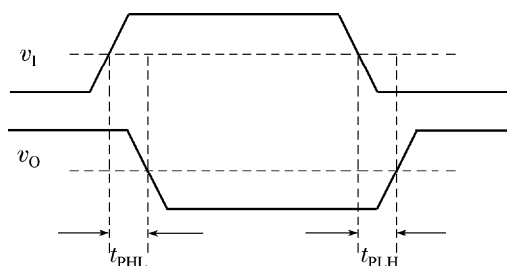


图 3.3.7 TTL 与非门的传输时间

3. TTL 与非门的电压传输特性

对于实际应用来说,TTL 与非门电路的外特性是应该掌握的重点。外特性是通过集成电路的引出端测得的电路特性。

(1) 电压传输特性曲线

与非门的电压传输特性曲线是指与非门的输出电压与输入电压之间的对应关系曲线,即 $v_o = f(v_i)$,它反映了电路的静态特性。其测试方法如图 3.3.8 所示,输入电压从 0 V 起逐步增大到 5 V,测得与非门对应的输出电压的值,绘出与非门的电压传输特性曲线如图 3.3.9 所示。此曲线可以分为四个区域来描述。

① AB 段(截止区)

在曲线 AB 段中输入电压 v_i 很低, $0 \text{ V} < v_i < 0.6 \text{ V}$, T_1 处于深度饱和状态(设深度饱和压降 $V_{CES1} \approx 0.1 \text{ V}$), T_2 、 T_3 截止, T_4 、 D 导通,输出电压 $v_o = 3.6 \text{ V}$ 且基本保持恒定。这时 T_2 的基极电位 $v_{B2} = v_i + V_{CES1}$,随着 v_i 的增加 v_{B2} 也在增加;当 $v_{B2} = 0.7 \text{ V}$ 时(这时 $v_i = v_{B2} - V_{CES1} = 0.6 \text{ V}$), T_2 开始导通,以后开始有了 i_{C2} ,使得 R_{C2} 上的压降增大, v_o 开始下降。所以

B 点坐标等于 $v_1(B)=0.6\text{ V}$ 。 B 点就是 T_2 开始导通点。

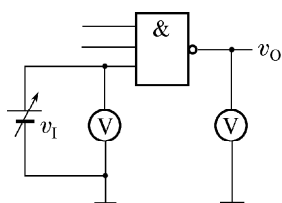


图 3.3.8 传输特性的测试方法

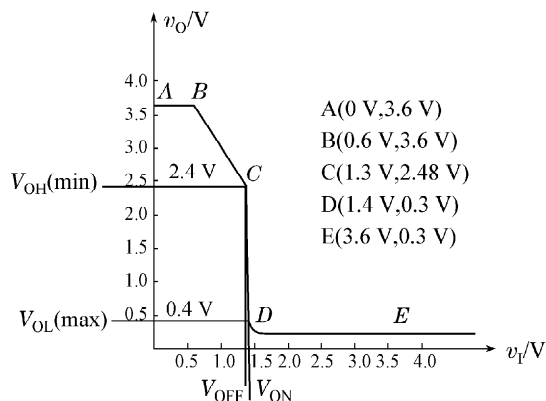


图 3.3.9 TTL 与非门的电压传输特性

② BC 段(线性区)

$v_1 > 0.6\text{ V}$ ，即过了 B 点以后， T_2 导通。在这段区间内， T_1 仍处于饱和状态， T_2 处于放大状态。因为 $v_{B2} < 1.4\text{ V}$ ， $v_{B3} < 0.7\text{ V}$ ，故 T_3 仍截止， T_4 、 D 仍导通。随着 v_1 的增加， v_{B2} 增加， v_{C2} 下降， v_O 下降。这段曲线近似为线性变化，故称为线性区。

当输入电压增加到 1.3 V 时， $v_{B2} = 1.4\text{ V}$ ， $v_{E2} = 0.7\text{ V}$ ， T_3 开始导通， C 点就是 T_3 开始导通的点。

③ CD 段(过渡区)

v_1 继续升高， v_{B2} 、 v_{E2} 也升高，当 v_{E2} 升高到 0.7 V 时， T_3 开始导通，并且随着 v_1 的增加由导通转变为饱和，输出电压急剧下降。 CD 段就是电路由输出高电平转换为低电平的阶段，故称为过渡区或转折区。 D 点坐标为 $v_1(D) \approx 1.4\text{ V}$ ， $v_O(D) \approx 0.3\text{ V}$ 。

由于这段曲线中 T_2 、 T_3 、 T_4 都处于放大状态， v_1 的微小增加都会使 T_2 的电流迅速增加，从而使 T_3 的电流迅速增加，使 T_3 迅速进入饱和区，故这段曲线很陡。

④ DE 段(饱和区)

进入饱和区以后， v_1 再增加， v_O 无明显变化，但是电路内部过程尚未完全结束。随着 v_1 继续升高， T_1 管转为倒置工作， T_1 的基极电流完全注入 T_2 ，使 T_2 进入饱和， v_{C2} 下降为 1 V ， T_4 、 D 截止，电路进入输出低电平的稳定状态。

(2) 几个重要参数

从 TTL 与非门的电压传输特性曲线上，我们可以定义几个重要的电路指标。

① 输出高电平电压 V_{OH} ： V_{OH} 的理论值为 3.6 V ，产品规定输出高电压的最小值 $V_{OH(min)} = 2.4\text{ V}$ ，即大于 2.4 V 的输出电压就可称为输出高电压 V_{OH} 。

② 输出低电平电压 V_{OL} ： V_{OL} 的理论值为 0.3 V ，产品规定输出低电压的最大值 $V_{OL(max)} = 0.4\text{ V}$ ，即小于 0.4 V 的输出电压就可称为输出低电压 V_{OL} 。

由上述规定可以看出，TTL 门电路的输出高低电压都不是一个值，而是一个范围。

③ 关门电平电压 V_{OFF} ：是指输出电压下降到 $V_{OH(min)}$ 时对应的输入电压。显然，只要 $v_1 < V_{OFF}$ ， V_o 就是高电压，所以 V_{OFF} 就是输入低电压的最大值，在产品手册中常称为输入低电平电压，用 $V_{IL(max)}$ 表示。从电压传输特性曲线上看， $V_{IL(max)}(V_{OFF}) \approx 1.3\text{ V}$ ，产品规定

$$V_{\text{IL}(\text{max})} = 0.8 \text{ V}。$$

④ 开门电平电压 V_{ON} :是指输出电压下降到 $V_{\text{OL}(\text{max})}$ 时对应的输入电压。显然,只要 $v_1 > V_{\text{ON}}$, V_o 就是低电压,所以 V_{ON} 就是输入高电压的最小值,在产品手册中常称为输入高电平电压,用 $V_{\text{IH}(\text{min})}$ 表示。从电压传输特性曲线上看, $V_{\text{IH}(\text{min})}$ (V_{ON}) 略大于 1.3 V ,为了保证产品可靠,留有余地,规定 $V_{\text{IH}(\text{min})} = 2 \text{ V}$ 。

⑤ 阈值电压 V_{th} :决定电路截止和导通的分界线,也是决定输出高、低电压的分界线。从电压传输特性曲线上看, V_{th} 的值界于 V_{OFF} 与 V_{ON} 之间,而 V_{OFF} 与 V_{ON} 的实际值又差别不大,所以,近似为 $V_{\text{th}} \approx V_{\text{OFF}} \approx V_{\text{ON}}$ 。 V_{th} 是一个很重要的参数,在近似分析和估算时,常把它作为决定与非门工作状态的关键值,即 $v_1 > V_{\text{th}}$,与非门开门,输出低电平; $v_1 < V_{\text{th}}$,与非门关门,输出高电平。 V_{th} 又常被形象化地称为门槛电压, V_{th} 的值为 $1.3 \sim 1.4 \text{ V}$ 。

(3) 抗干扰能力

抗干扰能力是指电路在干扰信号的作用下,维持原来逻辑状态的能力。实际上,TTL门电路的输出高低电平电压不是一个值,而是一个范围,如图 3.3.10 所示。同样,它的输入高低电平电压也有一个范围,把在保证电路输出逻辑值不变(或者说 v_o 变化的范围不超过允许限度)的条件下,输入电平电压的允许波动范围称为输入端噪声容限。

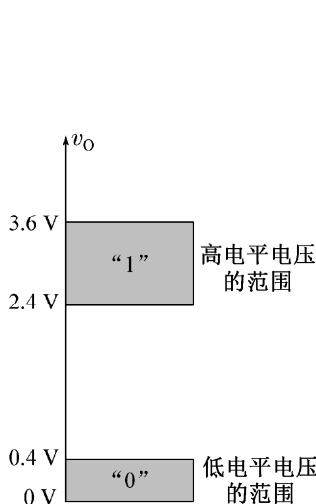


图 3.3.10 输出高低电平的电压范围

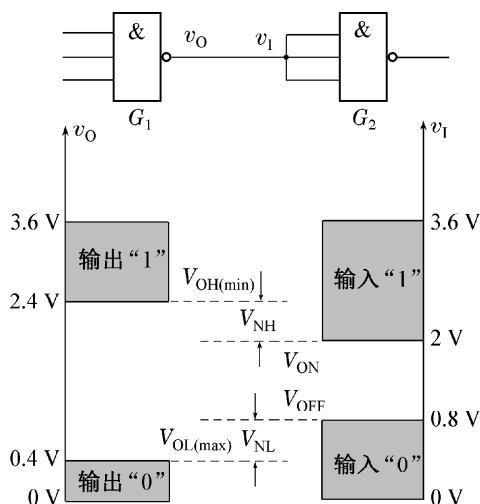


图 3.3.11 噪声容限图解

图 3.3.11 中,若前一个门 G_1 输出为低电压,则后一个门 G_2 输入也为低电压。如果由于某种干扰,使 G_2 的输入低电压高于了输出低电压的最大值 $V_{\text{OL}(\text{max})}$,从电压传输特性曲线上看,只要这个值不大于 V_{OFF} , G_2 的输出电压仍大于 $V_{\text{OH}(\text{min})}$,即逻辑关系仍是正确的。因此,在输入低电压时,把关门电压 V_{OFF} 与 $V_{\text{OL}(\text{max})}$ 之差称为低电平噪声容限,用 V_{NL} 来表示,即低电平噪声容限:

$$V_{\text{NL}} = V_{\text{OFF}} - V_{\text{OL}(\text{max})} = 0.8 \text{ V} - 0.4 \text{ V} = 0.4 \text{ V} \quad (3.3.2)$$

若前一个门 G_1 输出为高电压,则后一个门 G_2 输入也为高电压。如果由于某种干扰,使 G_2 的输入低电压低于了输出高电压的最小值 $V_{\text{OH}(\text{min})}$,从电压传输特性曲线上看,只要这个值不小于 V_{ON} , G_2 的输出电压仍小于 $V_{\text{OL}(\text{max})}$,逻辑关系仍是正确的。因此,在输入高电压

时,把 $V_{OH(\min)}$ 与开门电压 V_{ON} 之差称为高电平噪声容限,用 V_{NH} 来表示,即高电平噪声容限:

$$V_{NH} = V_{OH(\min)} - V_{ON} = 2.4 \text{ V} - 2.0 \text{ V} = 0.4 \text{ V} \quad (3.3.3)$$

7400 系列门电路高电平噪声容限 V_{NH} 和低电平噪声容限 V_{NL} 一般均约为 0.4 V 。

噪声容限反映了门电路的抗干扰能力指标。显然,噪声容限越大,电路的抗干扰能力越强。由此可知,二值数字逻辑中的“0”和“1”都是允许有一定容差的,这也正是数字电路的一个突出的特点。

4. TTL 与非门的带负载能力

在数字系统中,门电路的输出端一般都要与其他门电路的输入端相连,称为带负载。像图 3.3.12 所示的那样,前边的与非门 G_0 叫做驱动门,后边的 $G_1, G_2, G_3, \dots, G_n$ 叫做负载门。负载门的多少会对驱动门的特性带来怎样的影响? 一个驱动门门电路最多允许带几个同类的负载门? 下面就来讨论这个问题。

(1) 输入低电平电流 I_{IL} 与输入高电平电流 I_{IH}

这是两个与带负载能力有关的电路参数指标。

① 输入低电平电流 I_{IL}

输入低电平电流 I_{IL} 是指当门电路的输入端接低电平时,从门电路输入端流出的电流。

根据图 3.3.13 中 TTL 与非门电路的数据,可以算出 $I_{IL} = \frac{V_{CC} - v_{BI}}{R_{b1}} = \frac{5 - 1}{4} = 1(\text{mA})$ 。

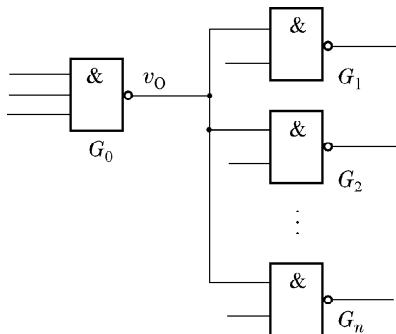


图 3.3.12 门电路带 n 个负载

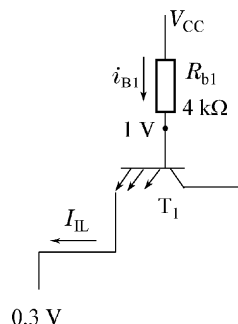


图 3.3.13 输入低电平电流 I_{IL}

② 输入高电平电流 I_{IH}

输入高电平电流 I_{IH} 是指当门电路的输入端接高电平时,流入输入端的电流。有两种不同的情况,如图 3.3.14 所示。

a. 寄生三极管效应。当与非门一个输入端(如 A 端)接高电平,其他输入端接低电平,这时 $I_{IH} = \beta_p i_{BI}$, β_p 为寄生三极管的电流放大系数。

b. 倒置工作状态。当与非门的输入端全接高电平,这时, T_1 的发射结反偏,集电结正偏,工作于倒置的放大状态。这时 $I_{IH} = \beta_i i_{BI}$, β_i 为倒置放大的电流放大系数。

由于 β_p 和 β_i 的值都远小于 1,所以 I_{IH} 的数值比较小。

射极电流 i_{E4} , 同时又是负载门的输入高电平电流 I_{IH} , 如图 3.3.16 所示。在负载电流较小时, 由于驱动门 T_4 工作在射极输出状态, 电路的输出阻抗很低, 负载电流的变化使驱动门 V_{OH} 变化很小。如果负载门的个数增加, 则拉电流增大, 即驱动门的 T_4 管发射极电流 i_{E4} 增加, R_{C4} 上的压降增加。最终将使 T_4 的 $b-c$ 结变为正向偏置, T_4 进入饱和状态。这时, T_4 将失去射极跟随能力, 输出阻抗加大, 因而 V_{OH} 便随着输出电流的增加而下降。所以使用时, 拉电流负载也要受到限制。前面提到过输出高电平不得低于 $V_{OH(\min)} = 2.4 \text{ V}$ 。因此, 把输出高电平时允许拉出输出端的电流定义为输出高电平电流 I_{OH} , 这也是门电路的一个参数。由此可得出, 输出高电平时所能驱动同类门最多的个数为

$$N_{OH} = \frac{I_{OH}}{I_{IH}} \quad (3.3.5)$$

式中 N_{OH} 称为输出高电平时的扇出系数。 N_{OH} 在设计应用计算中也应该取其整数部分。

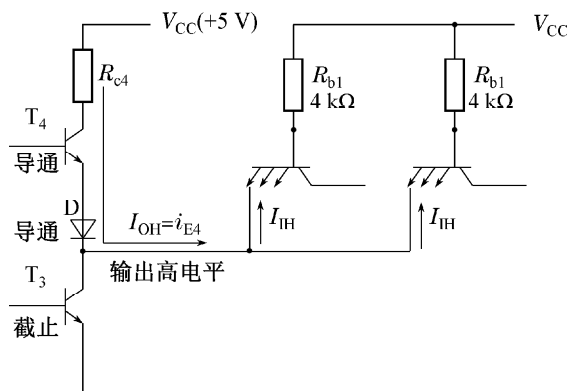


图 3.3.16 带拉电流负载

通常基本的 TTL 与非门电路, 其扇出系数约为 10, 而性能更好的门电路的扇出数最高可达 30~50。一般 TTL 器件的数据手册中, 并不给出扇出系数, 而须用计算或用实验的方法求得, 并要注意在设计时留有余地, 以保证数字电路或系统能正常地运行。由于输出低电平电流 I_{OL} 一般大于输出高电平电流 I_{OH} , N_{OL} 也不一定等于 N_{OH} , 因而在实际工程设计中, 常取两者中的较小值作为门电路的扇出系数, 用 N_O 表示。

【例 3.3.1】 试计算基本的 TTL 与非门 7410 带同类门时的扇出系数。

解 (1) 从 TTL 数据手册可查到 7410 的参数如下:

$$I_{OL} = 16 \text{ mA}, \quad I_{IL} = -1.6 \text{ mA}$$

$$I_{OH} = 0.4 \text{ mA}, \quad I_{IH} = 0.04 \text{ mA}$$

数据前的负号表示电流的流向, 对于灌电流负载取负号, 计算时只取绝对值。

(2) 根据式 (3.3.4), 可计算低电平输出时的扇出系数:

$$N_{OL} = \frac{16 \text{ mA}}{1.6 \text{ mA}} = 10$$

(3) 根据式 (3.3.5), 可计算高电平输出时的扇出系数:

$$N_{OH} = \frac{0.4 \text{ mA}}{0.04 \text{ mA}} = 10$$

可见, 这时 $N_{OL} = N_{OH}$ 。如前所述, 若 $N_{OL} \neq N_{OH}$, 则取较小的作为电路的扇出系数。

3.3.2 其他功能的 TTL 门电路

对图 3.3.1 所示的基本 TTL 与非门电路的结构做些改进,就可以得到其他逻辑功能的 TTL 逻辑门电路,如 TTL 非门、或非门、与或非门、集电极开路门和三态输出门电路等,下面分别加以讨论。

1. TTL 非门

将图 3.3.1 的 TTL 与非门电路中的 T_1 管改为一个发射极的三极管就成为形如图 3.3.17 的非门电路。很容易验证,该电路实现的非逻辑关系为 $L = \bar{A}$ 。

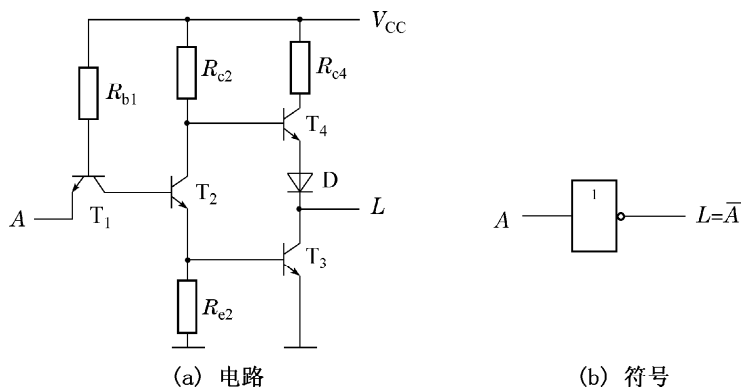


图 3.3.17 TTL 非门电路

2. TTL 或非门

TTL 或非门电路如图 3.3.18 所示。图中 T_{1B} 、 T_{2B} 、 R_{1B} 组成的部分与 T_{1A} 、 T_{2A} 、 R_{1A} 组成的部分完全相同。 A 、 B 两输入端中只要有一个为高电平,则 T_{2A} 或 T_{2B} 饱和导通,使 T_3 也饱和导通, L 输出低电平;只有当 A 、 B 两输入端都为低电平时, T_{2A} 和 T_{2B} 才都截止, T_3 也截止,输出 L 为高电平。电路实现了或非功能,即 $L = \overline{A+B}$ 。

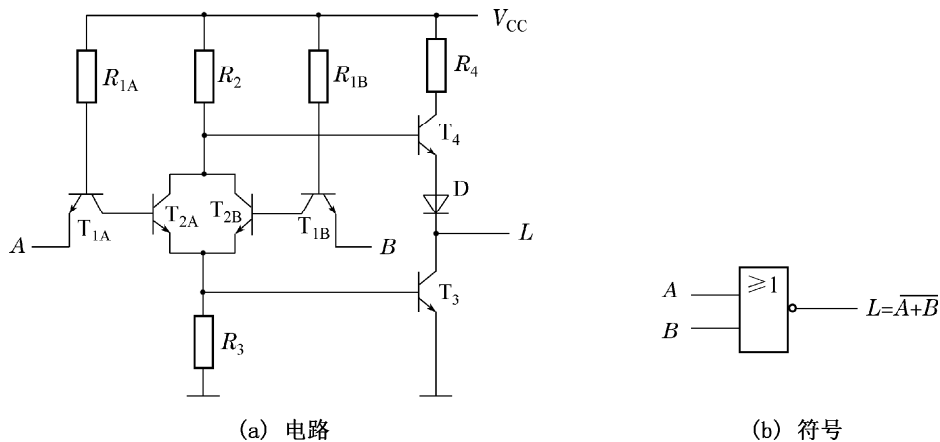


图 3.3.18 TTL 或非门电路

3. TTL 与或非门

将图 3.3.18 所示的 TTL 或非门电路中 T_{1A} 、 T_{1B} 都换成多发射极三极管, 就成为 TTL 与或非门电路, 如图 3.3.19 所示。用前面介绍的与非门和或非门的分析方法, 很容易分析得出, 该电路实现的是与或非逻辑功能, 即 $L = \overline{A_1 A_2 + B_1 B_2}$ 。

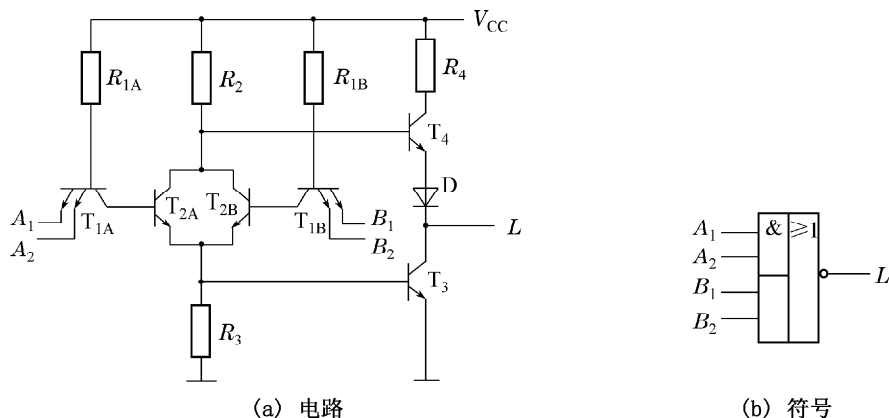


图 3.3.19 TTL 与或非门电路

4. 集电极开路门(OC 门)

在工程实践中, 有时需要将几个门的输出端直接并联使用, 以实现与逻辑, 这种接法称为“线与”。TTL 门电路的输出结构决定了它不能进行线与。

如果将 G_1 、 G_2 两个 TTL 与非门的输出直接连接起来, 如图 3.3.20 所示, 当 G_1 输出为高电平, G_2 输出为低电平时, 从 G_1 的电源 V_{CC} 通过 G_1 的 T_4 、 D 到 G_2 的 T_3 , 形成一个低阻通路, 产生很大的电流, 致使输出既不是高电平也不是低电平的不确定状态而导致电路出错, 逻辑功能将被破坏, 而且极有可能烧毁器件, 使门电路不能正常工作。所以, 普通的 TTL 门电路是不能进行线与的。

为满足实际应用中实现线与的要求, 专门生产了一种可以进行线与的门电路——集电极开路门, 简称 OC 门 (Open Collector)。图 3.3.21(a) 为 OC 门的内部电路结构, 它是在 TTL 门的输出极电路中, 取消了 R_4 、 T_4 和二极管 D 三个元件, 使 T_3 的集电极处在开路状态直接作为输出端。图 3.3.21(b) 是 OC 门的逻辑符号。

OC 门主要有以下几方面的应用:

(1) 实现线与

2 个 OC 门实现线与时的电路如图 3.3.22 所示。此时的逻辑关系为

$$L = L_1 \cdot L_2 = \overline{AB} \cdot \overline{CD} = \overline{AB + CD} \quad (3.3.6)$$

即在输出线上实现了与运算。由式 3.3.6 可知, 利用图

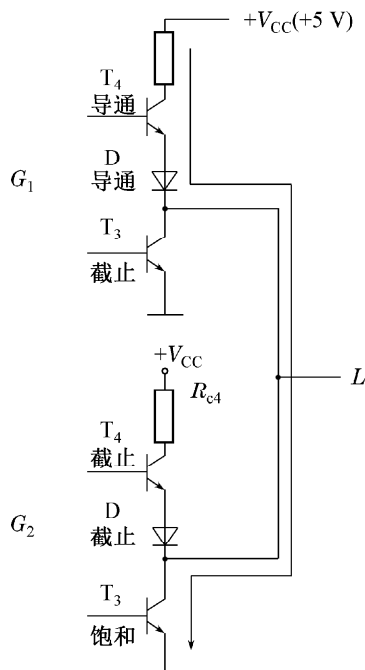


图 3.3.20 普通 TTL 门电路输出的并联

3.3.22 类型的线与电路可以很方便地实现与或非运算。

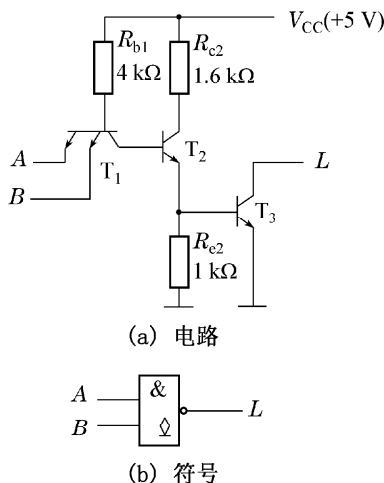


图 3.3.21 集电极开路(OC)门

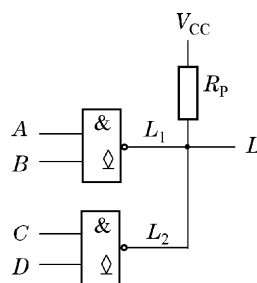


图 3.3.22 实现线与

在使用 OC 门进行线与时,外接上拉电阻 R_P 的选择非常重要,只有 R_P 选择得当,才能保证 OC 门输出满足要求的高电平和低电平。

假定有 n 个 OC 门的输出端并联,后面接 m 个普通的 TTL 与非门作为负载,如图 3.3.23 所示,则 R_P 的选择按以下两种最坏情况考虑。

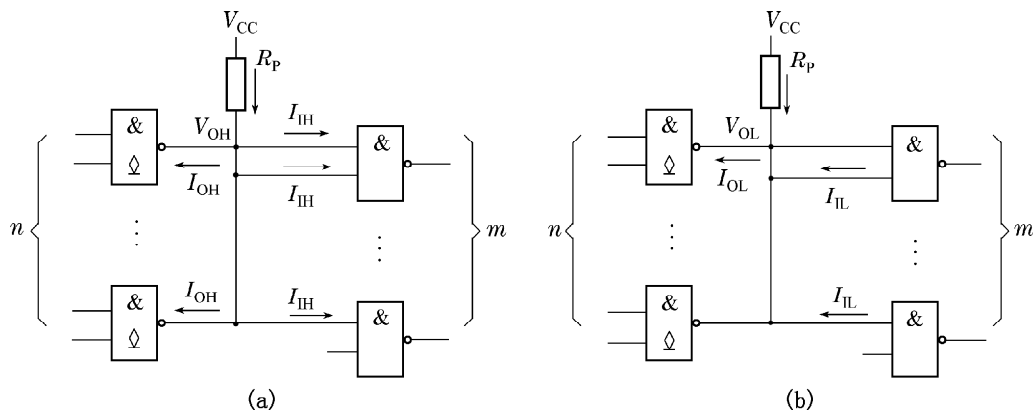


图 3.3.23 外接上拉电阻 R_P 的选择

当所有的 OC 门都截止时,输出 V_o 应为高电平,如图 3.3.23(a)所示。这时 R_P 不能太大,如果 R_P 太大,则其上压降太大,输出高电平就会太低。因此,当 R_P 为最大值时要保证输出电压为 $V_{OH(\min)}$,由

$$V_{CC} - (nI_{OH} + m'I_{IH})R_{P(\max)} = V_{OH(\min)} \quad (3.3.7)$$

得

$$R_{P(\max)} = \frac{V_{CC} - V_{OH(\min)}}{nI_{OH} + m'I_{IH}} \quad (3.3.8)$$

式中: $V_{OH(\min)}$ 为 OC 门输出高电平的下限值; I_{OH} 为 OC 门输出管截止时的灌电流; I_{IH} 为负载门的输入高电平电流; n 为 OC 门的个数; m' 为负载门输入端的个数(注意不是负载

门的个数)。

当 OC 门中至少有一个导通时,输出 V_o 应为低电平。我们考虑最坏情况,即只有一个 OC 门导通,如图 3.3.23(b)所示。这时 R_P 不能太小,如果 R_P 太小,则灌入导通的那个 OC 门的负载电流超过 $I_{OL(max)}$,就会使 OC 门的 T_3 管脱离饱和,导致输出低电平电压上升。因此,当 R_P 为最小值时要保证输出电压为 $V_{OL(max)}$ 。由

$$I_{OL(max)} = \frac{V_{CC} - V_{OL(max)}}{R_{P(min)}} + m' I_{IL} \quad (3.3.9)$$

得

$$R_{P(min)} = \frac{V_{CC} - V_{OL(max)}}{I_{OL(max)} - m' I_{IL}} \quad (3.3.10)$$

式中: $V_{OL(max)}$ 为 OC 门输出低电平的上限值; $I_{OL(max)}$ 为 OC 门输出低电平时的灌电流的最大值; I_{IL} 为负载门的输入低电平电流; m' 为负载门输入端的个数。

综合以上两种情况, R_P 可由式(3.3.11)确定。

$$R_{P(min)} \leq R_P \leq R_{P(max)} \quad (3.3.11)$$

【例 3.3.2】 如图 3.3.24 所示电路中,OC 门是 3 个 74LS 系列 OC 门,负载门为 3 个 74LS 系列与非门。已知 $I_{OH} = 100 \mu A$, $V_{OH(max)} = 0.4 V$, $V_{OH(min)} = 3.2 V$, $I_{IL} = 0.4 mA$, $I_{IH} = 20 \mu A$, 试计算 R_P 的阻值范围。

解

$$R_{P(max)} = \frac{V_{CC} - V_{OH(min)}}{n I_{OH} + m' I_{IH}} = \frac{5 - 3.2}{3 \times 0.1 + 3 \times 0.02} = 5 (k\Omega)$$

$$R_{P(min)} = \frac{V_{CC} - V_{OH(max)}}{I_{OL(max)} - m' I_{IL}} = \frac{5 - 0.4}{8 - 3 \times 0.4} = 0.68 (k\Omega)$$

故 R_P 应取 $0.68 \sim 5 k\Omega$ 。

(2) 实现电平转换

OC 门可以作为在数字系统的接口电路(与外部设备相连接的地方),能很方便地实现逻辑电平转换。如图 3.3.25 把上拉电阻接到 10 V 电源上,这样在 OC 门输入普通的 TTL 电平,而输出高电平就可以高达 10 V。

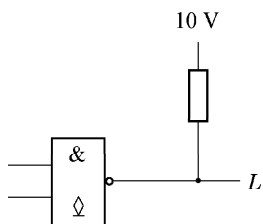


图 3.3.25 实现电平转换

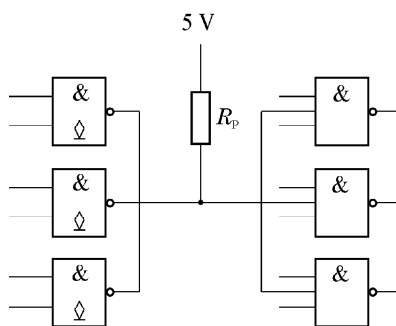


图 3.3.24 OC 门

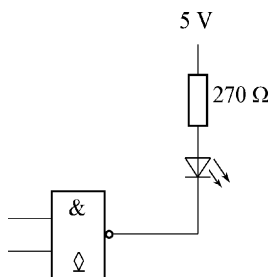


图 3.3.26 驱动发光二极管

(3) 驱动显示器件和执行机构

OC 门可以直接驱动发光二极管 LED 和执行机构,如干簧继电器、脉冲变压器等,如图 3.3.26 是用来驱动发光二极管的电路。应用中 V_{CC} 和限流电阻的值要根据 OC 门和 LED、

执行机构的正常工作电流等来确定。

5. 三态输出门

三态输出门(Tristate Logic,TSL)简称三态门,是一种特殊的门电路,它有三种输出状态,即在原有输出高电平、低电平两种工作状态的基础上,多了一种高阻抗状态(相当于隔离状态)。输出高阻抗状态时,表示其输出端悬浮,相当于与输入电路隔离开来。此时,三态门电路的输出与其输入变量状态没有关系。

(1) 三态输出门的电路结构

TTL 三态门的电路是在 TTL 与非门基础上增加控制电路所构成的,典型电路如图 3. 3. 27(a)所示,其逻辑符号如图 3. 3. 27(b)和图 3. 3. 27(c)所示。

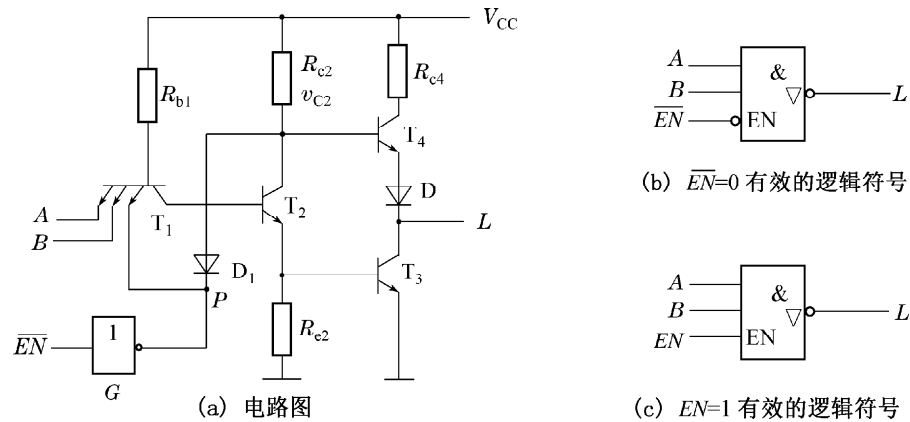


图 3. 3. 27 三态输出与非门

当 $\overline{EN}=0$ 时, G 输出为 1, D_1 截止,与 P 端相连的 T_1 的发射结也截止。三态门相当于一个正常的二输入端与非门,输出 $L=\overline{AB}$,为与非门的正常工作状态。

当 $\overline{EN}=1$ 时,控制门 G 输出为 0,即 $v_P=0.3\text{ V}$,这一方面使 D_1 导通, $v_{C2}=1\text{ V}$, T_4 、 D 截止;另一方面使 $v_{B1}=1\text{ V}$, T_2 、 T_3 也截止。这时从输出端 L 看进去,对地和对电源都相当于开路,呈现出高电阻。所以,称这种状态为高阻态或禁止态。

这种 $\overline{EN}=0$ 时为正常工作状态的三态门称为低电平有效的三态门,其逻辑符号如图 3. 3. 27(b)所示。根据以上分析,可以得到三态门电路的真值见表 3. 3. 2。

表 3. 3. 2 TTL 三态门电路的真值表

输入			输出	功能说明
$\overline{EN}$	A	B	L	
0	0	0	1	与非门 $L=\overline{AB}$
0	0	1	1	
0	1	0	1	
0	1	1	0	
1	×	×	高阻	高阻抗状态

如果将图 3.3.27(a)中的非门 G 去掉,并将 $\overline{EN}$ 改为 EN 时,则在使能端 $EN=1$ 时为正常工作状态, $EN=0$ 时为高阻状态,这种三态门称为高电平有效的三态门,逻辑符号如图 3.3.27(c)。

(2) 三态输出门的应用

三态门的基本应用是在数字系统中构成总线。

① 用三态门构成单向总线

图 3.3.28(a)为用三态门构成的单向数据总线。图中的总线是由多个三态门的输出连接而成。在任何时刻,仅允许其中一个三态门的输入控制端 EN 为 1,使输入数据从经过这个三态门反相后,单向送到总线上,而其他的 EN 都为 0,使它们的三态门都处于高阻态,对传送的数据没有影响,从而实现信号的分时传送。

注意:在某一时刻不允许电路同时有两个或两个以上的 EN 为 1,否则,总线传送的数据就会出错,甚至损坏器件。

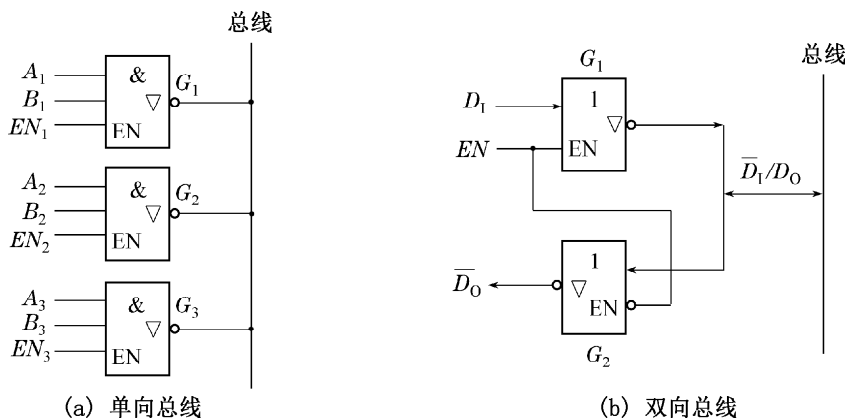


图 3.3.28 三态门组成的总线

② 实现数据的双向传输

图 3.3.28(b)所示的电路为三态门组成的双向总线,能够实现数据的双向传输。图中用了两个三态非门。当 EN 为高电平时,三态门 G_1 处于正常工作,三态门 G_2 为高阻态,输入数据 D_1 经 G_1 反相后为 $\overline{D_1}$ 送到数据总线上;当 EN 为低电平时,三态非门 G_2 处于正常工作, G_1 为高阻态,总线上的数据 D_0 经 G_2 反相后输出 $\overline{D_0}$ 。可见,通过改变控制信号 EN ,就实现了信号的分时双向传送。

3.3.3 其他系列的 TTL 门电路

前面分析的典型 TTL 与非门电路属于 CT54/74 系列,为了提高门电路的工作速度和降低功耗,对 CT54/74 系列 TTL 门电路进行改进,相继研制出了 CT54H/74H 系列、CT54S/74S 系列和 CT54LS/74LS 等系列。

1. CT54H/74H 系列

CT54H/74H 系列又称高速系列,图 3.3.29 所示为 CT54H/74H 系列与非门的典型电

路。为了减小门的传输延迟时间,在电路结构上采取了两条措施:一个是在输出级采用了达林顿结构,用 T_3 、 T_4 复合管代替原来的 T_4 、D 管;另一个是将所有的电阻的阻值比 CT54/74 系列的减小了一半。采用达林顿结构提高了带拉电流的负载能力,同时也加快了对负载电容的充电速度,加上电路中阻值的减小,使得 CT54H/74H 系列与非门的平均传输时间比 74 系列缩短了将近一半,达到 6 ns 左右。所以,CT54H/74H 系列称为高速系列。

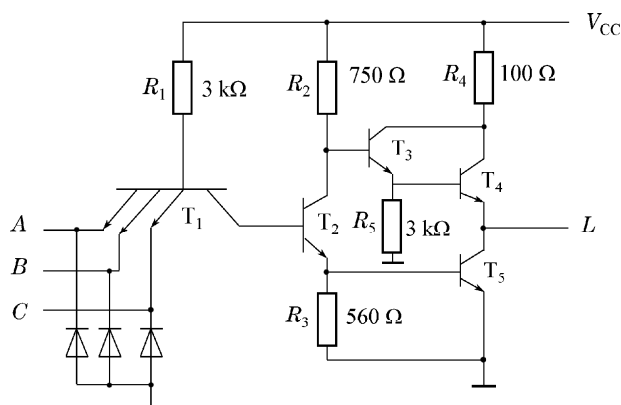


图 3.3.29 CT54H/74H 系列与非门

但是,由于电阻阻值的减小,使电路的静态功耗增加。74H 系列与非门电路的电源平均电流约为 74 与非门系列的两倍。所以,74H 系列工作速度的提高是靠增加功耗换来的。

性能比较好的门电路应该是工作速度快、功耗又小的门电路,通常用延迟-功耗积,即传输延迟时间和功耗的乘积(delay-power product,简称 dp 积)来评价门电路的性能优劣。74H 系列与非门和 74 系列与非门的 dp 积相差不多,所以 74H 系列的改进效果不够理想。

2. CT54S/74S 系列

CT54S/74S 系列又称肖特基系列。通过对前面介绍的 TTC 与非门电路的分析可以看到,限制与非门速度提高的主要因素是几只三极管在导通时几乎都处于饱和状态。因此,当三极管由饱和状态转换为截止状态时,需要消除在三极管基区内的存储电荷 Q_{BS} ,这要经过一段较长的存储时间 t_s ,从而限制了速度的提高。为解决这个问题,在 CT54S/74S 系列与非门电路中采用了抗饱和三极管(或称为肖特基三极管)。

所谓抗饱和三极管,是由双极型三极管和肖特基二极管(Schottky Barrier Diode, SBD)组成,如图 3.3.30 所示。SBD 管是借助于金属铝和 N 型硅的接触势垒产生整流作用,它具有正向导通压降小(约为 0.1~0.3 V)、本身无电荷存储作用及开关速度快等特点。所以,将肖特基二极管并接在三极管的基极 b 和集电极 c 之间,如图 3.3.30(a)所示。当三极管进入正向偏置时,SBD 管导通,将三极管的集电结电压钳制在

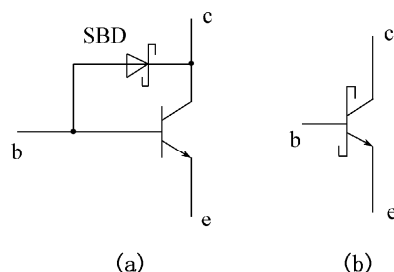


图 3.3.30 抗饱和三极管

0.3 V 左右,同时 SBD 管还将三极管基极的过驱动电流分流到集电极,从而使三极管工作在临界饱和状态,因此大大提高了工作速度。通常在电路图中用图 3.3.30(b)表示。

图 3.3.31 所示为 CT54S/74S 系列的典型与非门电路,它除了采用抗饱和三极管 T_1 、

T_2 、 T_3 、 T_4 、 T_5 、 T_6 外,还以 T_6 、 R_3 、 R_6 组成的有源网络代替图 3.3.29 中的电阻 R_3 ,这为 T_3 管的基极提供了有源泄放回路。另外,由于 T_6 管的作用,输入电压必须达到 1.4 V, T_2 管才开始导通,输出电压才开始下降,不存在 T_2 导通、 T_3 仍截止的情况,所以有较好的电压传输特性,如图 3.3.32 所示。显然,74S 系列门电路在低电平输入时的抗干扰能力得到提高。

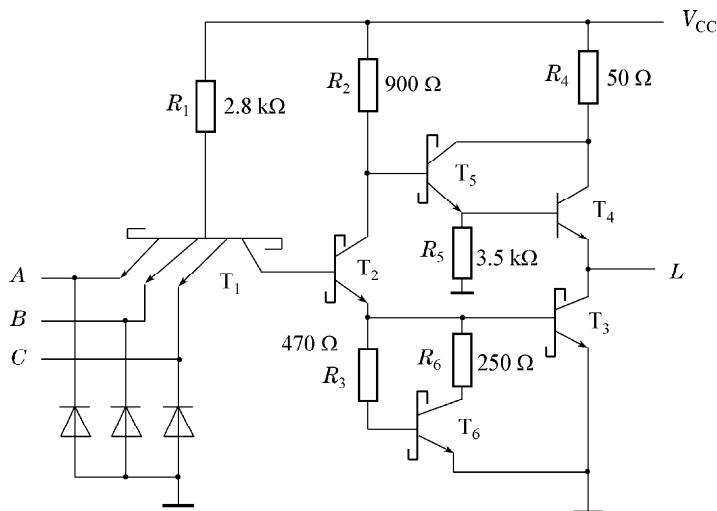


图 3.3.31 CT54S/74S 系列

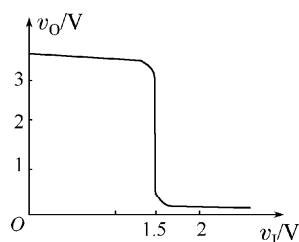


图 3.3.32 CT54S/74S 系列
电压传输特性

但是,由于 T_3 管工作在临界饱和状态,使输出低电平数值有所增加,最大值可达 0.5 V 左右。另外,由于 74S 系列门电路的电阻阻值减小,使静态功耗有所增加,它的平均传输延迟时间大约为 3 ns,静态平均功耗约为 19 mW,其 dp 积较 54/74、54H/74H 有所改善。

3. CT54LS/74LS 系列

74LS (Low-power Schottky TTL) 系列,也称为低功耗肖特基系列,是为了得到更小的延时-功耗积,在兼顾功耗和速度两方面的基础上,进一步开发出来的。

图 3.3.33 所示为 CT54LS/74LS 系列与非门的典型电路。与 74S 系列相比,为了降低功耗,首先是将电路中各个电阻的阻值增大了,同时将 R_5 接地的一端改接到输出端,以减小 T_3 管导通时 R_5 上的功耗。采取这些措施后,74LS 系列的功耗约为 74 系列的 1/5 或 74H 系列的 1/7。由于电阻阻值的加大,势必会影响电路的工作速度。但 74LS 系列电路中将多极管 T_1 用 SBD 管代替,因为这种二极管无电荷存储效应,有利于提高工作速度。同时为了进一步加速电路开关状态的转换过程,又接入了 D_3 、 D_4 这

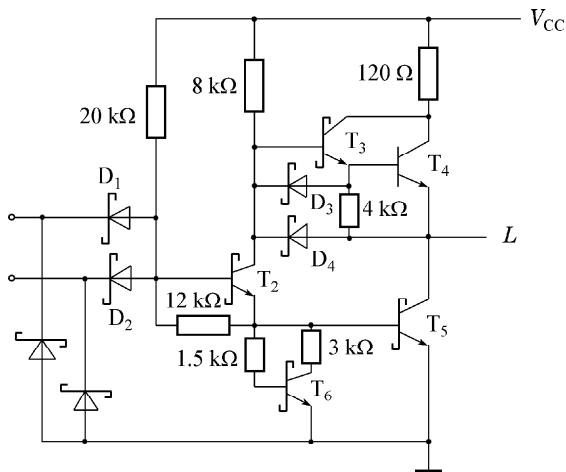


图 3.3.33 CT54LS/74LS 系列

两个 SBD 管。当输出端由高电平转换为低电平时, D_4 管经 T_2 管的集电极、 T_5 管的基极为输出端的负载电容提供了另一条放电回路, 这既加快了负载电容的放电过程, 又增加了 T_5 管的基极驱动电流, 故加快了 T_5 管的导通过程。同时 D_3 管也通过 T_2 管的集电极为 T_4 管的基极提供了一个泄放回路, 加快了 T_4 管的截止速度, 从而大大缩短了传输延迟时间。故解决了前面几种系列中为提高工作速度和降低功耗所采取措施的矛盾, 所以 CT54S/74LS 系列在四种系列中 d_p 积最小。

表 3.3.3 给出了四种 TTL 系列主要性能的比较。

表 3.3.3 TTL 系列器件主要性能比较

性能	CT54/74	CT54H/74H	CT54S/74S	CT54LS/74LS
$(t_{pd}/f_J)/ns$	10	6	3	10
$(p/f_J)/mW$	10	22	19	2
$d_p/(ns \cdot mW)$	100	135	80	20
最高工作频率/mHz	35	50	125	45

TTL 集成逻辑门电路是一种使用广泛的双极型集成门电路。双极型集成门电路除了 TTL 以外, 还有射极耦合逻辑(Emitter Coupled Logic, ECL)和集成注入逻辑(Integrated Injection Logic, I^2L 或 IIL)等几种逻辑门电路。有关 ECL 和 I^2L 门电路请扫描二维码学习。



ECL 和 I^2L 门电路

3.3.4 TTL 集成逻辑门电路的使用注意事项

1. 电源电压及电源干扰的消除

电源电压的变化对 54 系列应满足 $5V(\pm 10\%)$ 、对 74 系列应满足 $5V(\pm 5\%)$ 的要求, 电源的极性和地线不可接错。为了防止外来干扰通过电源串入电路, 需要对电源进行滤波, 通常在印刷电路板的电源输入端接入 $10 \sim 100 \mu F$ 成的电容进行滤波, 在印刷电路板上, 每隔 $6 \sim 8$ 个门加接一个 $0.01 \sim 0.1 \mu F$ 的电容对高频进行滤波。

2. 输出端的连接

具有推拉输出结构的 TTL 门电路的输出不允许直接并联使用。输出端不允许直接接电源 V_{CC} 或直接接地。使用时, 输出电流应小于产品手册上规定的最大值。三态输出门的输出可并联使用, 但在同一时刻只能有一个门工作, 其他门输出处于高阻状态。集电极开路门出端可并联使用, 但公共输出端和电源 V_{CC} 之间应接负载电阻 R_L 。

3. 闲置输入端的处理

TTL 集成门电路使用时, 对于闲置输入端(不用的输入端)一般不悬空, 主要是防止干扰信号从悬空输入端上引入电路。对于闲置输入的处理以不改变电路逻辑功能及工作稳定性为原则。常用的方法有以下几种。

(1) 对于与非门的闲置输入端, 可直接接电源电压 V_{CC} , 或通过 $1 k\Omega \sim 10 k\Omega$ 的电阻接电源 V_{CC} , 如图 3.3.34(a) 和 3.3.34(b) 所示。

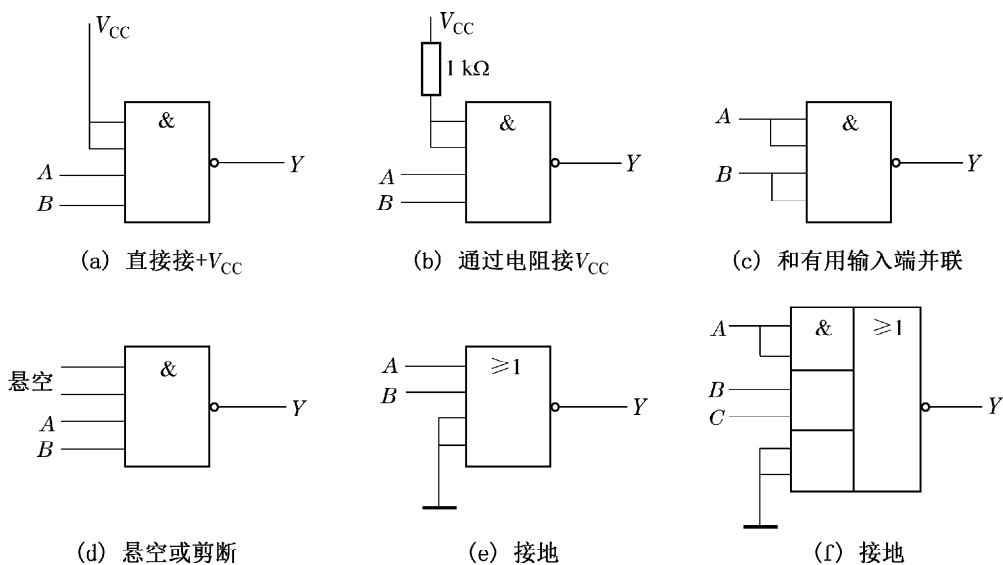


图 3.3.34 与非门和或非门闲置输入端的处理

(2) 如前级驱动能力允许时,可将闲置输入端与有用输入端并联使用,如图 3.3.34(c)所示。

(3) 在外界干扰很小时,与非门的闲置输入端可以剪断或悬空,如图 3.3.34(d)所示。但不许接开路长线,以免引入干扰而产生逻辑错误。

(4) 或非门不使用的闲置输入端应接地,对与或非门中不使用的与门至少有一个输入端接地,如图 3.3.34(e)和图 3.3.34(f)所示。

4. 电路安装接线和焊接应注意的问题

- (1) 连线要尽量短,最好用绞合线。
- (2) 整体接地要好,地线要粗、短。
- (3) 焊接用的烙铁最好不大于 25 W ,使用中性焊剂,如松香酒精溶液,不可使用焊油。
- (4) 由于集成电路外引线间距离很近,焊接时焊点要小,不得将相邻引线短路,焊接时间要短。

(5) 印刷电路板焊接完毕后,不得浸泡在有机溶液中清洗,只能用少量酒精擦去外引线上的助焊剂和污垢。

5. 调试中应注意的问题

(1) 对 CT54/CT74 和 CT54H/CT74H 系列 TTL 电路,输出的高电平不小于 2.4 V ,输出低电平不大于 0.4 V 。对 CT54S/CT74S 和 CT54LS/CT74LS 系列的 TTL 电路,输出的高电平不小于 2.7 V ,输出的低电平不大于 0.5 V 。上述 4 个系列输入的高电平不小于 2.4 V ,低电平不大于 0.8 V 。

(2) 当输出高电平时,输出端不能碰地,不然会使 T_4 因电流过大而烧坏。输出低电平时,输出端不能碰电源 $V_{CC}=5\text{ V}$,否则 T_3 同样会烧坏。

3.4 CMOS 集成逻辑门电路

CMOS 逻辑门是互补-金属-氧化物-半导体场效应管门电路的简称,它由增强型 PMOS 管和增强型 NMOS 管组成,是继 TTL 电路之后开发出来的数字集成器件。CMOS 数字集成电路分为 4000 系列和高速系列,我国生产的 CC4000 系列和国际上 4000 系列同序号产品可互换使用。高速 CMOS (HCMOS) 数字集成电路主要有 54/74HC 和 54/74HCT 两个系列,后者可与同序号的 TTL 产品互换使用。由于 CMOS 数字集成电路具有低功耗和高抗干扰能力等突出优点,因此,在中、大规模数字集成电路中有着广泛的应用,目前已超越 TTL 成为占据市场统治地位的逻辑器件。下面先讨论 CMOS 反相器,然后介绍其他 CMOS 逻辑门电路。

3.4.1 CMOS 反相器

CMOS 逻辑门电路是由 N 沟道 MOSFET 和 P 沟道 MOSFET 互补而成,通常称为互补型 MOS 逻辑电路,简称 CMOS 逻辑电路。

图 3.4.1(a)表示 CMOS 反相器电路,由两只增强型 MOSFET 组成,其中一个为 N 沟道结构,另一个为 P 沟道结构。它们的栅极相连作为反相器的输入端,漏极相连作为反相器的输出端。 T_P 的源极接正电源 V_{DD} , T_N 的源极接地。图 3.4.1(b)是在本书中使用的简化电路。为了电路能正常工作,要求电源 V_{DD} 大于两管开启电压绝对值之和,即 $V_{DD} > (V_{TN} + |V_{TP}|)$, 且 $V_{TN} = |V_{TP}|$ 。

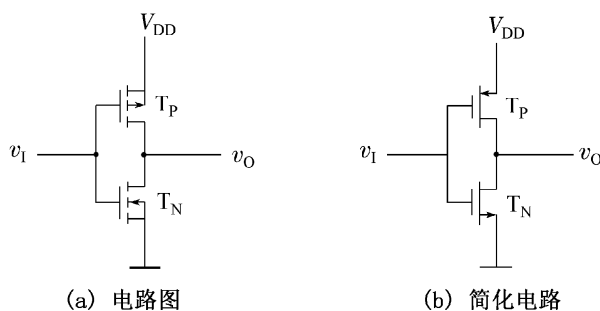


图 3.4.1 CMOS 非门电路

1. 工作原理

首先考虑两种极限情况:当 v_1 处于逻辑 0 时,相应的电压近似为 0 V;而当 v_1 处于逻辑 1 时,相应的电压近似为 V_{DD} 。假设在两种情况下,N 沟道管 T_N 为工作管,P 沟道管 T_P 为负载管。但是,由于电路是互补对称的,这种假设可以是任意的,相反的情况亦将导致相同的结果。

(1) 当输入为低电平,即 $v_1 = 0$ V 时, T_N 截止, T_P 导通, T_N 的截止电阻约为 $500\text{ M}\Omega$, T_P 的导通电阻约为 $750\text{ }\Omega$ 。所以输出 $v_O \approx V_{DD}$, 即 v_O 为高电平。

(2) 当输入为高电平,即 $v_1 = V_{DD}$ 时, T_N 导通, T_P 截止, T_N 的导通电阻约为 $750\text{ }\Omega$, T_P 的

截止电阻约为 $500\text{ M}\Omega$ 。所以,输出 $v_O \approx 0\text{ V}$,即 v_O 为低电平。该电路实现了非逻辑 $L = \overline{A}$ 。

通过以上分析可以看出,在 CMOS 非门电路中,无论电路处于何种状态, T_N 、 T_P 中总有一个截止,所以它的静态功耗极低,有微功耗电路之称。

2. 电压传输特性

设 CMOS 非门的电源电压 $V_{DD} = 10\text{ V}$, 两管的开启电压为 $V_{TN} = |V_{TP}| = 2\text{ V}$ 。

(1) 当 $v_I < 2\text{ V}$, T_N 截止, T_P 导通, 输出 $v_O \approx V_{DD} = 10\text{ V}$ 。

(2) 当 $2\text{ V} < v_I < 5\text{ V}$, T_N 和 T_P 都导通, 但 T_N 的栅源电压 $< T_P$ 栅源电压绝对值, 即 T_N 工作在饱和区, T_P 工作在可变电阻区, T_N 的导通电阻 $> T_P$ 的导通电阻。所以, 这时 v_O 开始下降, 但下降不多, 输出仍为高电平。

(3) 当 $v_I = 5\text{ V}$, T_N 的栅源电压 $= T_P$ 栅源电压绝对值, 两管都工作在饱和区, 且导通电阻相等。所以, $v_O = (V_{DD}/2) = 5\text{ V}$ 。

(4) 当 $5\text{ V} < v_I < 8\text{ V}$, 情况与 (2) 相反, T_P 工作在饱和区, T_N 工作在可变电阻区, T_P 的导通电阻 $> T_N$ 的导通电阻, 所以 v_O 变为低电平。

(5) 当 $v_I > 8\text{ V}$, T_P 截止, T_N 导通, 输出 $v_O = 0\text{ V}$ 。可见, 两管在 $v_I = V_{DD}/2$ 处转换状态, 所以 CMOS 门电路的阈值电压 (或称门槛电压) $V_{th} = V_{DD}/2$ 。

3. 工作速度

因为 CMOS 反相器电路具有互补对称的性质, 所以它的开通时间与关闭时间是相等的。电路工作时总有一个管子导通, 且管子的导通电阻做得较小, 充放电的时间常数就较小, 当带电容负载时, 给电容充电和放电过程都比较快。图 3.4.3(a) 所示为 $v_I = 0$ 时, T_N 截止, T_P 导通。这时由 V_{DD} 通过导通的 T_P 向负载电容 C_L 充电; 图 3.4.3(b) 所示为 $v_I = 1$ 时, T_P 截止, T_N 导通, 这时负载电容 C_L 通过导通的 T_N 放电。可见, 充电、放电都是低电阻回路, 所以速度较快。CMOS 非门的平均传输延迟时间约为 10 ns 。

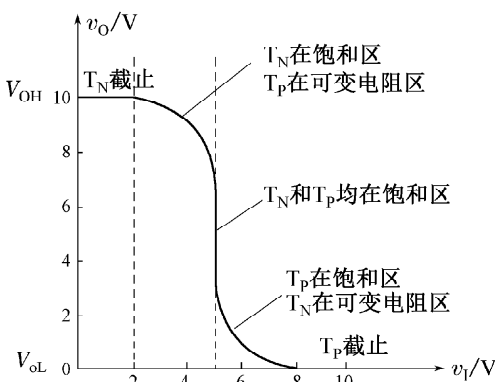


图 3.4.2 CMOS 反相器的传输特性

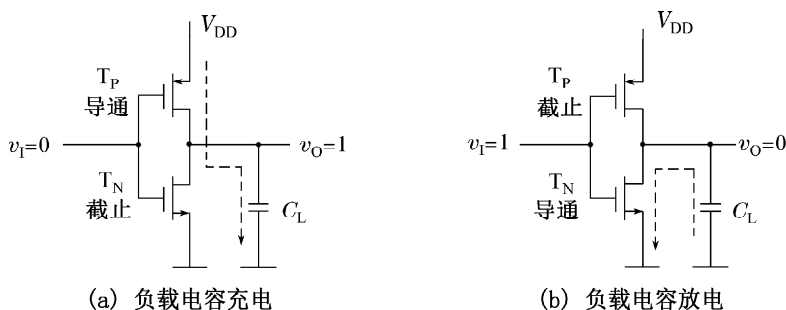


图 3.4.3 CMOS 非门带电容负载的情况

3.4.2 其他功能的 CMOS 门电路

CMOS 系列逻辑门电路中,除上述介绍的反相器(非门)外,还有与非门、或非门、与或非门、异或门、开路门、三态门和传输门等电路,并且实际的 CMOS 逻辑电路,许多都带有输入、输出保护电路和缓冲电路。

1. CMOS 与非门和或非门电路

(1) CMOS 与非门电路

图 3.4.4 所示为二输入端 CMOS 与非门电路,其中包括两个串联的 N 沟道增强型 MOS 管和两个并联的 P 沟道增强型 MOS 管。一个 NMOS 管和一个 PMOS 管的栅极连接在一起,与一个输入端相接。当输入端 A、B 中只要有一个为低电平时,就会使与它相连的 NMOS 管截止,与它相连的 PMOS 管导通,输出为高电平;仅当 A、B 全为高电平时,才会使两个串联的 NMOS 管都导通,使两个并联的 PMOS 管都截止,输出为低电平。

这样,就实现了与非的逻辑功能,即

$$L = \overline{A \cdot B}$$

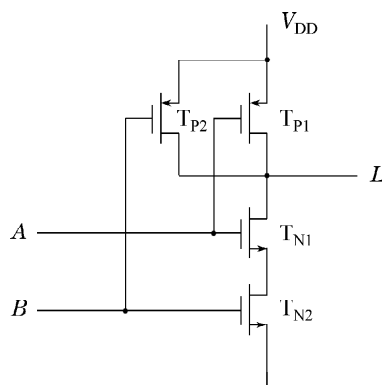


图 3.4.4 CMOS 与非门电路

几个输入端的与非门就必须有几个 NMOS 管串联和几个 PMOS 管并联。相应地,其逻辑表达式为

$$L = \overline{A \cdot B \cdot C \cdots}$$

(2) CMOS 或非门电路

图 3.4.5 所示是二输入端 CMOS 或非门电路,其中包括两个并联的 NMOS 管和两个串联的 PMOS 管。一个 NMOS 管和一个 PMOS 管的栅极连接在一起,引出一个输入端。

当输入端 A、B 中只要有一个为高电平时,就会使与它相连的 NMOS 管导通,与它相连的 PMOS 管截止,输出为低电平;仅当 A、B 全为低电平时,两个并联 NMOS 管都截止,两个串联的 PMOS 管都导通,输出为高电平。

因此,这种电路具有或非的逻辑功能,其逻辑表达式为

$$L = \overline{A + B}$$

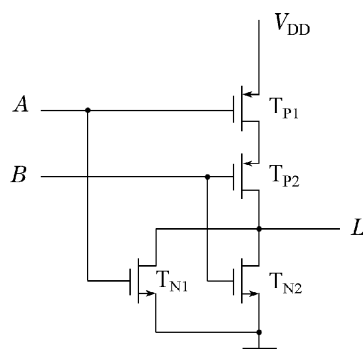


图 3.4.5 CMOS 或非门电路

当需要增加输入端个数到 n 时,只要有 n 个 NMOS 管并联和 n 个 PMOS 管串联,仿照图 3.4.5 的连接方式,就可以做成 n 个输入端的或非门,从而实现或非逻辑功能:

$$L = \overline{A + B + C + \cdots}$$

根据以上 CMOS 与非门和或非门电路的讨论可知,如果输入端的数目越多,则串联的管子也越多。若串联的管子全部导通时,其总的导通电阻会增加,以致影响输出电平,使与非门的低电平升高,而使或非门的高电平降低。所以,CMOS 逻辑门电路的输入端不宜过多,并且在 CMOS 电路的输入和输出端增加缓冲电路,即 CMOS 反相器,以规范电路的输入和输出逻辑电平。

2. CMOS 与或非门电路

CMOS 与或非门电路如图 3.4.6 所示。当输入 A 、 B 都为高电平时, T_2 、 T_4 导通同时 T_3 、 T_7 截止,输出低电平;或者当 C 、 D 都为高电平时, T_6 、 T_8 导通同时 T_1 、 T_5 截止,输出为低电平;否则,输出为高电平。实现了与或非逻辑, $L = \overline{AB + CD}$ 。

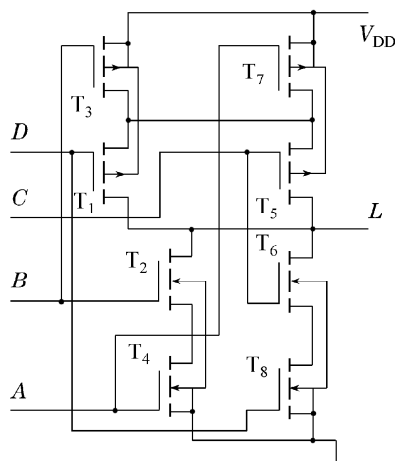


图 3.4.6 CMOS 与或非门电路

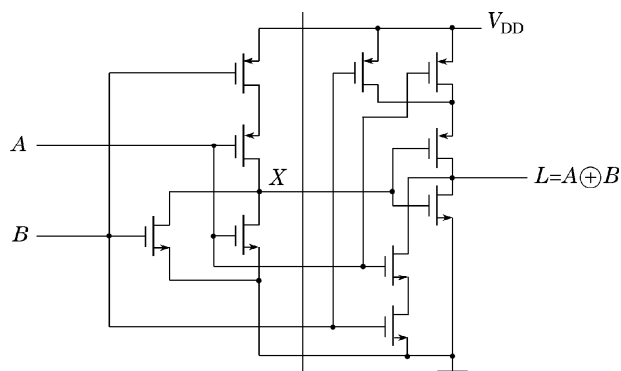


图 3.4.7 CMOS 异或门电路

3. CMOS 异或门电路

CMOS 异或门电路如图 3.4.7 所示,它是由两级组成,前级为或非门,输出为 $X = \overline{A+B}$ 。后级为与或非门,经过逻辑变换,可得

$$L = \overline{A \cdot B + X} = \overline{A \cdot B + \overline{A+B}} = \overline{A \cdot B + \overline{A} \cdot \overline{B}} = (\overline{A+B})(A+B) = \overline{A} \overline{B} + AB = A \oplus B。$$

即输出 L 为输入 A 、 B 的异或。

在异或门后面加接一级反相器就成为异或非门,便具有 $\overline{L} = \overline{AB + \overline{A} \overline{B}} = \overline{A} \overline{B} + AB = A \odot B$ 逻辑的功能,所以异或非门也称为同或门。

4. CMOS 漏极开路门和三态(TSL)输出门电路

如果从 CMOS 集成电路的输出端看,CMOS 漏极开路门和三态输出门又具有和前面几种门电路不同的输出结构。

(1) CMOS 漏极开路(OD 门)

OD 门与 TTL 集电极开路门(OC 门)对应,其特点是可以实现线与。可以用来进行逻辑电平变换,具有较强的带负载(指的是 CMOS 门电路作负载)能力等。OD 门有多种形式。图 3.4.8 所示是漏极开路的 CMOS 与非门的输出级电路图、逻辑符号及输出线与连接图。

注意使用时必须外接电阻 R_P , R_P 的选择原则等同于 OC 门中的 R_P 的选择原则。

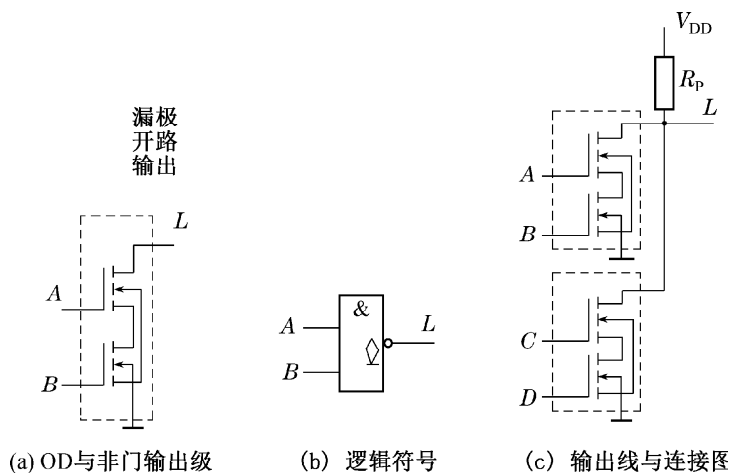


图 3.4.8 漏极开路(OD)与非门电路

(2) 三态(TSL)输出门电路

利用 OD 门虽然可以实现线与的功能,但外接电阻 R_P 的选择要受到一定的限制而不能取得太小,因此影响了工作速度。同时它省去了有源负载,使得带负载能力下降。为保持推拉式输出级的优点,又能作线与连接,跟 TTL 三态门电路相似,人们也开发了 CMOS 三态输出门电路,它的输出也具有高电平、低电平和高阻态(禁止态)这三种状态。这种电路也称为三态输出缓冲器。

图 3.4.9(a)所示为低电平有效的 CMOS 三态输出门。它是在 CMOS 反相器基础上增加一个 P 沟道和一个 N 沟道 MOS 管 T_{P2} 、 T_{N2} 并分别与反相器的 T_{P1} 、 T_{N1} 串联而构成。其工作原理是:

当 $\overline{EN}=0$ 时, T_{P2} 和 T_{N2} 同时导通, T_{N1} 和 T_{P1} 组成的非门正常工作,输出 $L=\overline{A}$ 。

当 $\overline{EN}=1$ 时, T_{P2} 和 T_{N2} 同时截止,输出 L 对地和对电源都相当于开路,为高阻状态。

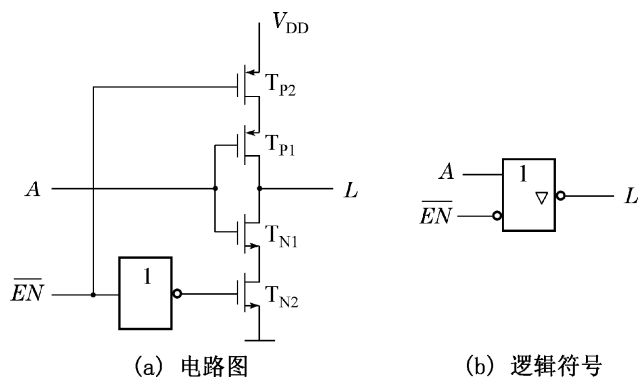


图 3.4.9 低电平使能的 CMOS 三态非门

所以,这是一个使能端 $\overline{EN}$ 为低电平有效的三态门,逻辑符号如图 3.4.9(b)所示。

图 3.4.10(a)所示为高电平使能的三态输出缓冲电路,其中 A 是输入端, L 为输出端, EN 是控制信号输入端,即使能端,图 3.4.10(b)是它的逻辑符号。

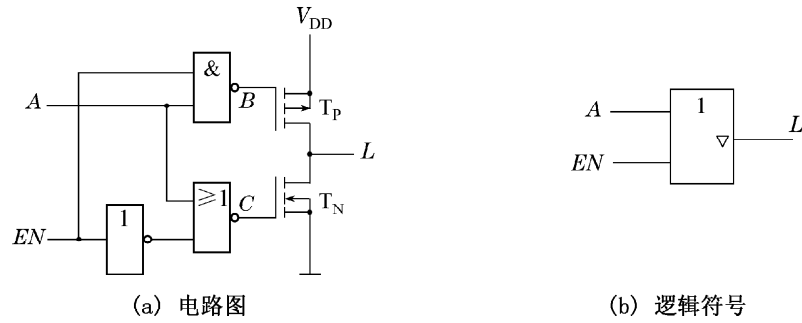


图 3.4.10 高电平使能的三态输出门电路

当使能端 $EN=1$ 时,如果 $A=0$,则 $B=1,C=1$,使得 T_N 导通,同时 T_P 截止,输出端 $L=0$;如果 $A=1$,则 $B=0,C=0$,使得 T_N 截止, T_P 导通,输出端 $L=1$ 。当使能端 $EN=0$ 时,不论 A 的取值为何,都使得 $B=1,C=0$,则 T_N 和 T_P 均截止,电路的输出端出现开路,既不是低电平,又不是高电平,是高阻工作状态。

这就是说,当 EN 为有效的高电平时,电路处于正常逻辑工作状态, $L=A$ 。而当 EN 为低电平时,电路处于高阻状态。图 3.4.10(a)所示三态门电路的真值表如表 3.4.1 所示,其中 $\times$ 表示 A 可以是 0 或 1。

表 3.4.1 高电平使能三态门电路的真值表

使能 EN	输入 A	输出 L
1	0	0
1	1	1
0	$\times$	高阻

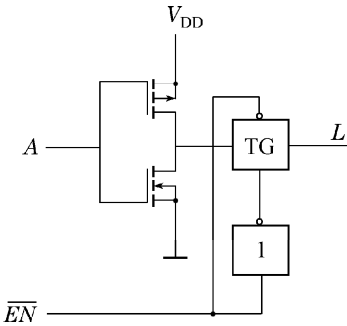


图 3.4.11 传输门控制的 CMOS 三态非门电路

图 3.4.11 所示的是低电平使能、传输门控 CMOS 三态非门电路,它是在反相器基础上增加一级 CMOS 传输门(下面再作介绍)作为反相器的控制开关。当 $\overline{EN}=0$ 时,传输门导通,输出 $L=\overline{A}$;当 $\overline{EN}=1$ 时,传输门截止,输出呈现高阻态。

5. CMOS 传输门

传输门(Transmission Gate, TG)的应用比较广泛,可以作为基本单元电路构成各种逻辑电路,用于数字信号的传输;也可以在取样——保持电路、斩波电路、模数和数模转换等电路中传输模拟信号,作为模拟开关。

传输门电路由一个 N 沟道增强型和一个 P 沟道增强型 MOSFET 并联组成,如图 3.4.12(a)所示,逻辑符号如图 3.4.12(b)所示。 C 和 $\overline{C}$ 为控制端,使用时总是加互补的信号。下面简要分析它的工作原理。

设两管的开启电压 $V_{TN}=|V_{TP}|$ 。如果要传输的信号 v_i 的变化范围为 $0\sim V_{DD}$,则将控制端 C 和 $\overline{C}$ 的高电平设置为 V_{DD} ,低电平设置为 0,并将 T_N 的衬底接低电平 0 V, T_P 的衬底

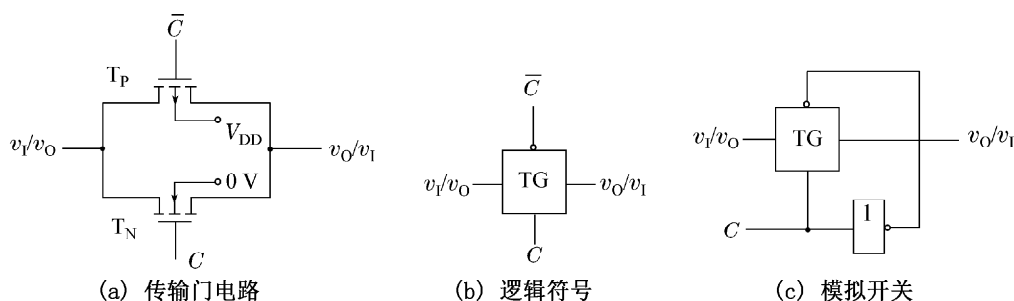


图 3.4.12 CMOS 传输门及模拟开关

接高电平 V_{DD} 。

当 C 接高电平 V_{DD} , $\bar{C}$ 接低电平 0 V 时, 若 $0 < v_I < (V_{DD} - V_{TN})$, T_N 导通; 若 $|V_{TP}| \leq v_I \leq V_{DD}$, T_P 导通。即 v_I 在 $0 \sim V_{DD}$ 的范围变化时, 至少有一管导通, 输出与输入之间呈低电阻, 将输入电压传到输出端, $v_O = v_I$, 相当于开关闭合。

当 C 接低电平 0 V , $\bar{C}$ 接高电平 V_{DD} , v_I 在 $0 \sim V_{DD}$ 的范围变化时, T_N 和 T_P 都截止, 输出呈高阻状态, 输入电压不能传到输出端, 相当于开关断开。

可见, CMOS 传输门实现了信号的可控传输。由于 T_N 和 T_P 的源极和漏极可以互换, 所以 CMOS 传输门是双向器件, 即输入和输出端允许互换使用。

模拟开关就是把 CMOS 传输门和一个非门组合起来, 由非门产生互补的控制信号来控制的开关, 如图 3.4.12(c) 所示。

3.4.3 高速 CMOS 门电路

自 CMOS 门电路问世以来, 便以其低功耗、高抗干扰能力等突出的优点引起了用户和生产厂商的普遍重视。然而早期生产的 CMOS 门电路是 CMOS 4000 系列, 其工作速度较低, 使它的应用范围受到了一定的限制。高速 CMOS 与 4000 系列相比, 具有较高的工作速度、较强的负载能力。

高速 CMOS 门电路主要有 54 系列和 74 系列产品。这里介绍两个 74 系列高速 CMOS 门电路: HC(High-speed CMOS, 高速 CMOS) 和 HCT(High-speed CMOS TTL Compatible, 与 TTL 兼容的高速 CMOS)。

HC 系列只用于 CMOS 逻辑的系统中, 并可用 $2 \sim 6\text{ V}$ 的电源, 即使采用 5 V 电源, HC 门电路也不能与 TTL 门电路兼容。HC 门电路使用 CMOS 输入电平。当用 5 V 电源时, HC 门电路的最小输入高电平 $V_{IH(\min)} = 3.5\text{ V}$, 最大输入低电平 $V_{IL(\max)} = 1.5\text{ V}$; HC 门电路的最小输出高电平 $V_{OH(\min)} = 3.84\text{ V}$, 最大输出低电平 $V_{OL(\max)} = 0.33\text{ V}$, 而 TTL 器件的输出高电平为 $2.4 \sim 3.5\text{ V}$ 。所以, HC 门电路不能与 TTL 门电路兼容。

HCT 系列门电路可直接与 TTL 门电路互换。HCT 系列门电路也使用 CMOS 门电路输入电平。当用 5 V 电源时, HCT 系列门电路的最小输入高电平 $V_{IH(\min)} = 2.0\text{ V}$, 最大输入低电平 $V_{IL(\max)} = 0.8\text{ V}$; HCT 门电路的最小输出高电平 $V_{OH(\min)} = 3.84\text{ V}$, 最大输出低电平 $V_{OL(\max)} = 0.33\text{ V}$, 它与 TTL 门电路的输出电平匹配, 故可直接与 TTL 门电路互换。

3.4.4 CMOS 数字集成逻辑门电路使用注意事项

1. 防止静电

CMOS 电路的栅极与沟道之间有一层绝缘的二氧化硅薄层,厚度仅为 $0.1\sim 0.2\ \mu\text{m}$ 。由于 CMOS 电路的输入阻抗很高,而输入电容又很小,所以当不太强的静电加在栅极上时,其电场强度将超过 $10^5\ \text{V/cm}$ 。这样强的电场极易造成栅极击穿,导致永久损坏。所以,在使用时需注意以下几点:

(1) 人体能感应出几十伏的交流电压,人们衣服的摩擦也会产生上千伏的静电(尤其在冬天),故尽量不要用手接触 CMOS 电路的引脚。

(2) 焊接时宜使用 20 W 的内热式电烙铁,电烙铁外壳应接地。为安全起见,也可先拔下电烙铁电源插头,利用电烙铁的余热进行焊接。焊接时间不要超过 5 s。操作时,应避免穿戴尼龙、纯涤纶等易生静电的衣裤及手套等。

(3) 长期不使用的 CMOS 集成电路,应用锡纸将全部引脚短路后包装存放,待使用时再拆除包装。

(4) 更换集成电路时应先切断电源。

(5) 在存储、携带或运输 CMOS 器件和焊装有 MOS 器件的半成品印制板的过程中,应将集成电路和印制板放置于金属容器内,也可用铝箔将器件包封后放入普通容器内,但不要使用易产生静电的尼龙及塑料盒等容器,采用抗静电的塑料盒当然也可以。

(6) 装配工作台上不宜铺设塑料或有机玻璃板,最好铺上一块平整铝板或铁板,如没有则什么都不要铺。

(7) 在进行装配或实验时,电烙铁、示波器、稳压源等工具及仪器仪表都应良好接地,并要经常检查,发现问题应及时处理。一种简易检查接地是否良好的方法是,在电烙铁及仪器通电时,用电笔测试其外壳,若电笔发亮,说明接地不好;反之,则说明接地良好。

2. 正确选择电源

由于不同系列 CMOS 集成逻辑门电路的电源电压范围不同,故选择电源电压时应考虑不要超过电源极限电压。

(1) 4000 系列的电源电压可在 $3\sim 18\ \text{V}$ 范围内选择。

(2) HC 和 HCT 的电源电压分别可在 $2\sim 6\ \text{V}$ 和 $4.5\sim 5.5\ \text{V}$ 范围内选择。

3. 对输入端的处理

在使用 CMOS 逻辑门电路时,一般对输入端有以下要求:

(1) 应保证输入信号幅值不超过 CMOS 电路的电源电压,即满足 $V_{\text{SS}} \leq v_i \leq V_{\text{DD}}$,一般 $V_{\text{SS}} = 0\ \text{V}$ 。

(2) 所有不用的输入端不能悬空,应根据实际要求接入适当的电压(V_{DD} 、高电平或低电平)。由于 CMOS 集成电路输入阻抗极高,所以一旦输入端悬空,极易受外界噪声影响,从而破坏了电路的正常逻辑关系,也可能感应静电,造成栅极被击穿。

4. 对输出端的处理

(1) CMOS 逻辑门电路的输出端不能直接连到一起,否则导通的 P 沟道 MOS 场效应

管和导通的 N 沟道 MOS 场效应管形成低阻通路,造成电源短路。

(2) 在 CMOS 逻辑系统设计中,应尽量减少电容负载。因为电容负载会降低 CMOS 集成电路的工作速度和增加功耗。

3.5 TTL 门电路与 CMOS 门电路的接口

在设计数字电路或数字系统时,设计者从工作速度或功耗指标等方面的要求出发,合理选择不同类型的数字集成电路。在目前 TTL 和 CMOS 两种门电路并存的情况下,经常会遇到 TTL 门电路与 CMOS 门电路的接口问题。在图 3.5.1 中,无论是用 TTL 门电路驱动 CMOS 门电路还是用 CMOS 门电路驱动 TTL 门电路,驱动门必须能为负载门提供符合标准的高、低电平和足够的驱动电流,必须要满足下列条件:

驱动门 负载门

$$V_{OH(\min)} \geq V_{IH(\min)} \quad (3.5.1)$$

$$V_{OL(\max)} \leq V_{IL(\max)} \quad (3.5.2)$$

$$I_{OH(\max)} \geq I_{IH(\text{总})} \quad (3.5.3)$$

$$I_{OL(\max)} \geq I_{IL(\text{总})} \quad (3.5.4)$$

表 3.5.1 中列出了 TTL、CMOS4000 和 HCMOS 门电路的输出电压、输出电流、输入电压、输入电流等参数,供选择接口电路参考。

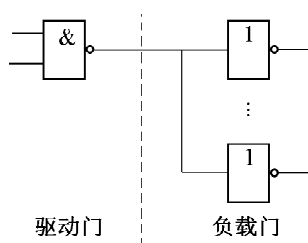


图 3.5.1 驱动门和负载门的连接

表 3.5.1 TTL 和 COMS 逻辑器件主要参数比较

参数名称	TTL					COMS		
	74	74S	74LS	74AS	74ALS	4000	74HC	74HCT
输入低电平电流 $I_{IL(\max)}/\text{mA}$	1.6	2.0	0.4	0.5	0.1	0.001	0.001	0.001
输入高电平电流 $I_{IH(\max)}/\mu\text{A}$	40	50	20	20	20	0.1	0.1	0.1
输出低电平电流 $I_{OL(\max)}/\text{mA}$	16	20	8	20	8	0.51	4	4
输出高电平电流 $I_{OH(\max)}/\text{mA}$	0.4	1	0.4	2	0.4	0.51	4	4
输入低电平电压 $V_{IL(\max)}/\text{V}$	0.8	0.5	0.8	0.8	0.8	1.5	1.0	0.8
输入高电平电压 $V_{IH(\min)}/\text{V}$	2.0	2.0	2.0	2.0	2.0	3.5	3.5	2.0
输出低电平电压 $V_{OL(\max)}/\text{V}$	0.4	0.5	0.5	0.5	0.5	0.05	0.1	0.1
输出高电平电压 $V_{OH(\min)}/\text{V}$	2.4	2.7	2.7	2.7	2.7	4.95	4.9	4.9
平均传输时间 t_{pd}/ns	9.5	3	8	3	2.5	45	10	13
功耗(每门) P_n/mW	10	19	4	8	1.2	0.005	0.005	0.005
电源电压 V_{CC} 或 V_{DD}/V	4.75~5.25					3~18	2~6	4.5~5.5

注:上述参数均是在电源电压 V_{CC} 或 $V_{DD}=5\text{V}$ 时测得。

3.5.1 TTL 门电路驱动 CMOS 门电路

由于 TTL 门的 $I_{OH(max)}$ 和 $I_{OL(max)}$ 远远大于 CMOS 门的 I_{IH} 和 I_{IL} , 所以 TTL 门驱动 CMOS 门时, 主要考虑 TTL 门的输出电平是否满足 CMOS 输入电平的要求。

(1) TTL 门驱动 4000 系列和 74HC 系列

从表 3.5.1 中看出, 当都采用 5 V 电源时, TTL 的 $V_{OH(min)}$ 为 2.4 V 或 2.7 V, 而 CMOS4000 系列和 74HC 系列电路的 $V_{IH(min)}$ 为 3.5 V, 显然不满足要求。这时可在 TTL 电路的输出端和电源之间接一上拉电阻 R_P , 如图 3.5.2(a) 所示。 R_P 的阻值取决于负载器件的数目及 TTL 和 CMOS 器件的电流参数, 可以用 OC 门外接上拉电阻的计算方法进行计算。但须注意, 此时 $V_{OH(min)} < V_{IH(min)}$ 。为保证负载门输入高电平的要求, 当 TTL 门输出高电平时, 使

$$V_{OH} = V_{DD} - R_P(I_{OH} + I_{IH(总)}) \quad (3.5.5)$$

式中: I_{OH} 为 TTL 门输出高电平时输出管截止时的漏电流; $I_{IH(总)}$ 为流入全部 CMOS 负载电路的电流。这两个电流的数值都很小, 如果 R_P 取值不太大, TTL 门的输出电压可以提升到接近 V_{DD} 。一般 R_P 在几百欧至几千欧之间。

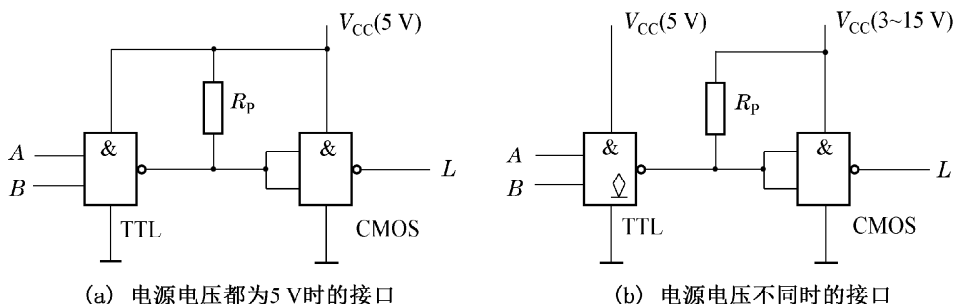


图 3.5.2 TTL 驱动 CMOS 门电路

如果 TTL 和 CMOS 器件采用的电源电压不同, 则宜使用 OC 门, 同时使用上拉电阻 R_P , 如图 3.5.2(b) 所示。

(2) TTL 门驱动 74HCT 系列

前面提到 74HCT 系列与 TTL 器件电压兼容, 它的输入电压参数为 $V_{IH(min)} = 2.0\text{ V}$, 而 TTL 的输出电压参数为 $V_{OH(min)}$ 为 2.4 V 或 2.7 V。因此, 两者可以直接相连, 不需外加其他器件。

3.5.2 CMOS 门驱动 TTL 门

根据表 3.5.1 中数据, 当都采用 5 V 电源时, CMOS 门的 $V_{OH(min)} > V_{IH(min)}$, CMOS 的 $V_{OL(max)} < V_{IL(max)}$, 两者电压参数相容。但是 CMOS 门的 I_{OH} 、 I_{OL} 参数较小, 所以, 这时主要考虑 CMOS 门的输出电流是否满足 TTL 输入电流的要求。

【例 3.5.1】 一个 74HC00 与非门电路能否驱动 4 个 7400 与非门? 能否驱动 4 个 74LS00 与非门?

解 从表 3.5.1 中查出,74 系列门的 $I_{IL}=1.6\text{ mA}$,74LS 系列门的 $I_{IL}=0.4\text{ mA}$,4 个 74 门的 $I_{IL(\text{总})}=4\times 1.6=6.4(\text{mA})$,4 个 74LS 门的 $I_{IL(\text{总})}=4\times 0.4=1.6(\text{mA})$ 。而 74HC 系列门的 $I_{OL}=4\text{ mA}$,所以不能驱动 4 个 7400 与非门,可以驱动 4 个 74LS00 与非门。

要提高 CMOS 门的驱动能力,可将同一芯片上的多个门并联使用,如图 3.5.3(a)所示;也可在 CMOS 门的输出端与 TTL 门的输入端之间加一 CMOS 驱动器,如图 3.5.3(b)所示。

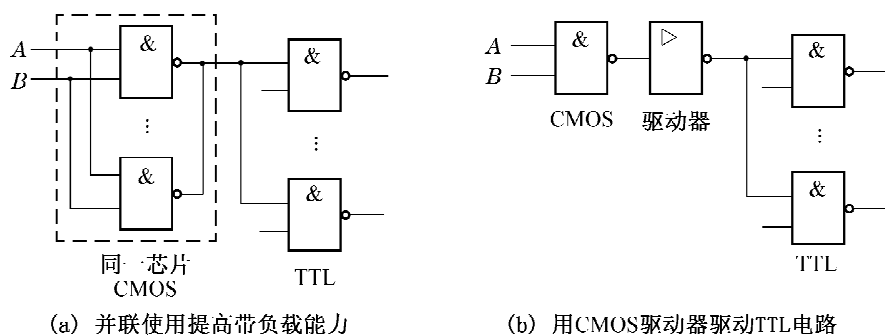


图 3.5.3 CMOS 驱动 TTL 门电路

3.5.3 TTL 和 CMOS 门电路带负载时的接口电路

在工程实践中,常常需要用 TTL 或 CMOS 门电路去驱动指示灯、发光二极管 LED、继电器等负载。

对于电流较小、电平能够匹配的负载可以直接驱动,图 3.5.4(a)所示为用 TTL 门电路驱动发光二极管 LED,这时只要在电路中串接一个约几百欧姆的限流电阻即可。如果 LED 的额定电流为 I_F ,额定电压为 V_D ,则限流电阻 $R=(V_{CC}-V_D-V_{OL})/I_F$ 。实际应用中,往往是根据计算结果选取靠近系列产品规格的电阻元件。

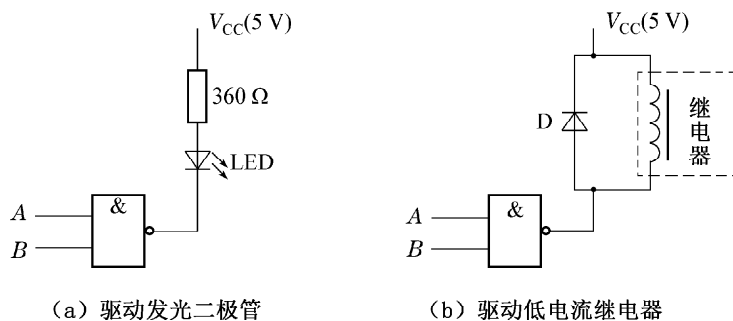


图 3.5.4 门电路带小电流负载

图 3.5.4(b)所示为用 TTL 门电路驱动 5 V 低电流继电器,其中阻尼二极管 D 作阻尼保护,在与非门关断时,继电器的初级电感线圈会产生较大的下面正极性上面负极性的感应电压,这时二极管 D 正好导通,可以释放电感中的电流能量。由于 D 的阻尼作用,把感应电压限制在 0.7 V 以内,用以防止感应过电压击穿与非门。

如果负载电流较大,可将同一芯片上的多个门并联作为驱动器,如图 3.5.5(a)所示。原则上只用少数几个并联使用,总的驱动电流略大于负载电流(留有余地)即可。当单个门驱动电流比负载电流小得多时,一般不使用多个门并联的方法,而是在门电路输出端接一只三极管,以提高负载能力,如图 3.5.5(b)所示。如果负载电流很大,则适宜选用功率三极管或者复合功率管。

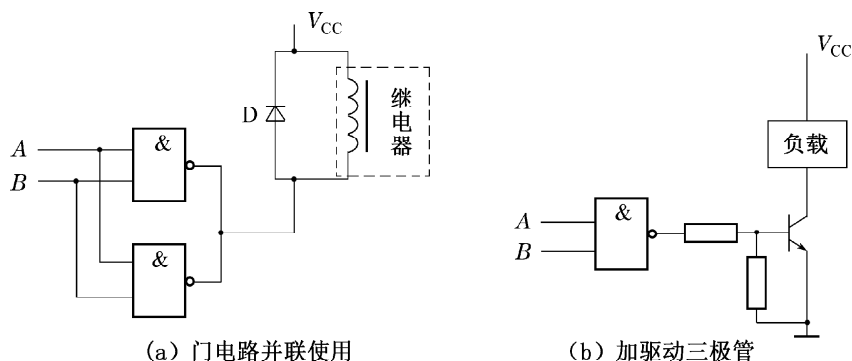


图 3.5.5 门电路带大电流负载

本章小结

在数字电路中,不论哪一种逻辑门电路,其中的关键器件是二极管、三极管和 MOS 管,它们均可以作为开关器件。影响它们开关速度的主要因素是器件内部各电极之间的电容。

逻辑门电路的主要技术参数有输入和输出高、低电平的最大值或最小值、噪声容限,传输延迟时间,功耗,延迟-功耗积,扇入数和扇出数等。

TTL 逻辑门电路是应用较广泛的门电路之一,电路由若干 BJT 和电阻组成。TTL 与非门采用多发射极三极管的输入级,输出级采用推拉式结构,其目的是为提高开关速度和增强带负载的能力。常用的还有 TTL 非门、或非门、异或门以及可以线与接法的集电极开路门和三态输出门。

CMOS 逻辑门电路是目前已经超越 TTL 门且应用最广泛的逻辑门电路。其突出优点是集成度高、功耗低、抗干扰功能强、电源适应范围宽。CMOS 逻辑门电路中,也可采用漏极开路门和三态门。

在逻辑门电路的实际应用中,有可能遇到不同类型门电路之间、门电路与负载之间的接口问题。正确分析与解决这些问题,是数字电路设计工作者应当具备的基本能力。

自我检测题

一、填空题

1. 和 TTL 集成逻辑门电路相比,CMOS 集成逻辑门电路的功耗_____。
2. TTL 集成逻辑门在带负载时,有_____电流负载和_____电流负载。
3. 三态门的 3 种输出状态是_____、_____、_____。
4. 在 TTL 集成逻辑门电路中,能实现线与功能的门称_____门。
5. TTL 集成逻辑或非门多余的输入端在使用时,应该接_____电平。
6. CMOS 集成逻辑与非门多余的输入端在使用时,应该接_____电平。
7. 欲使漏极开路的 CMOS 门电路实现“线与”功能,则需在输出端与电源之间外接_____。
8. CMOS 集成逻辑门驱动同类逻辑门的数量比 TTL 集成逻辑门_____。

二、选择题

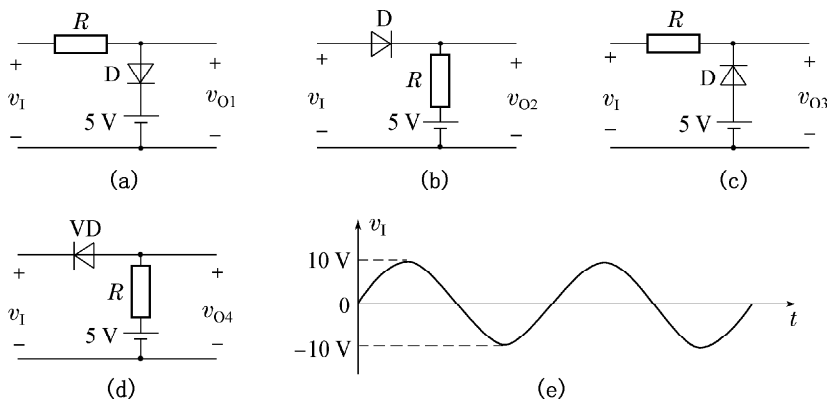
1. 在 CMOS 集成逻辑门电路中,能实现“线与”逻辑功能的是_____。
A. 与非门 B. 异或门 C. 三态门 D. OD 门
2. 对于 TTL 集成逻辑门电路,以下英国当作低电平“0”来看待的电压值是_____。
A. 2.4 V B. 3.6 V C. 5 V D. 0.3 V
3. 已知 CMOS 集成逻辑门电路的电源电压为 5V,则其阈值电压为_____。
A. 0 V B. 1.4 V C. 2.5 V D. 5 V
4. 在 TTL 集成逻辑门电路中,以下能实现总线连接方式的是_____。
A. 一般与非门 B. OC 门 C. 三态门 D. 一般异或门
5. 对于三态门,以下说法中正确的是_____。
A. 需要外接上拉电阻 B. 可以实现“线与”
C. 控制端有效时输出高阻抗状态 D. 控制端无效时输出高阻状态
6. 对于 OC 门,以下说法中正确的是_____。
A. 可以实现“线或” B. 可以实现“线与”
C. 可以输出高阻状态 D. 具有控制端 EN
7. 三态输出的门电路其输出端_____。
A. 可以并联且实现“线与” B. 不能并联也不能实现“线与”
C. 可以并联但不能实现“线与” D. 无需并联但可以实现“线与”
8. 当 TTL 集成逻辑门电路的输出端输出高电平时,随着输出电流绝对值的增加,输出高电平的变化趋势是_____。
A. 增加 B. 减小 C. 不变 D. 不能确定

练习题

- 3.1 半导体二极管的开、关条件是什么? 导通和截止时各有什么特点? 和理想开关比

较,它的主要缺点是什么?

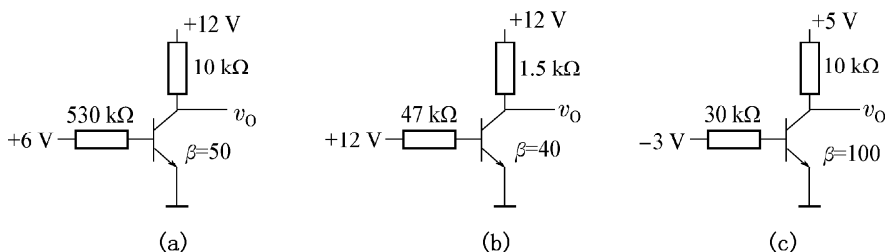
3.2 如题图 3.2(a~d)所示的二极管电路,其输入电压 v_1 的波形如题图 3.2(e)所示。试对应画出各电路输出电压 $v_{O1} \sim v_{O4}$ 的波形。设二极管正向压降为 $V_D = 0.7 \text{ V}$ 。



题图 3.2

3.3 半导体三极管的开、关条件是什么? 饱和导通和截止时各有什么特点? 和半导体二极管比较,它的主要优点是什么?

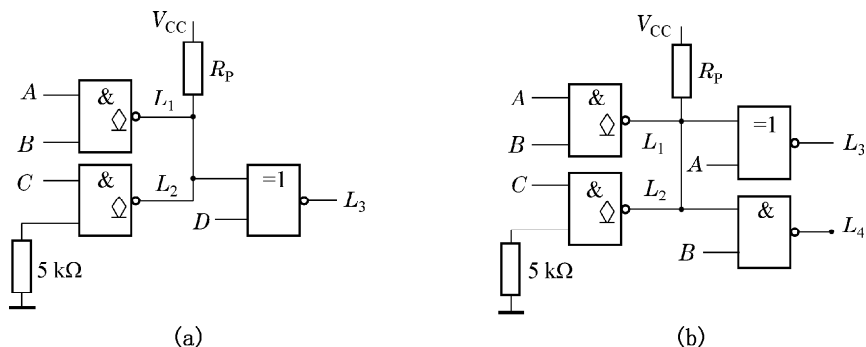
3.4 判断如题图 3.4 所示各电路中三极管的工作状态,并计算输出电压 v_O 的值。设三极管导通后 $V_{BE} = 0.7 \text{ V}$ 。



题图 3.4

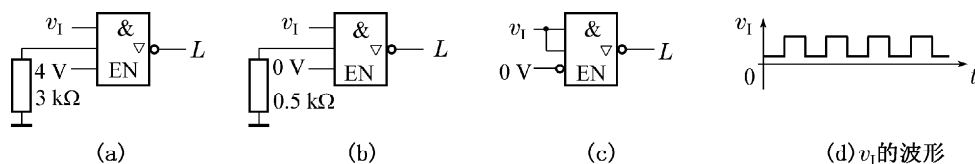
3.5 N 沟道增强型 MOS 管的开、关条件是什么? 导通和截止时各有什么特点? 和 P 沟道增强型 MOS 管比较,两者的主要区别是什么?

3.6 有 TTL 集电极开路门组成的逻辑电路如题图 3.6 所示,试写出各个输出端的逻辑函数表达式。



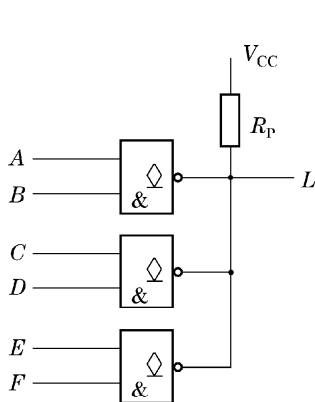
题图 3.6

3.7 逻辑电路如题图 3.7 所示,三态门电路的电源电压为 5 V,图中各个三态门电路的输入信号电压 v_i 的波形如图(d)所示,试画出图中(a~c)各个电路连接的情况下输出电压的波形。

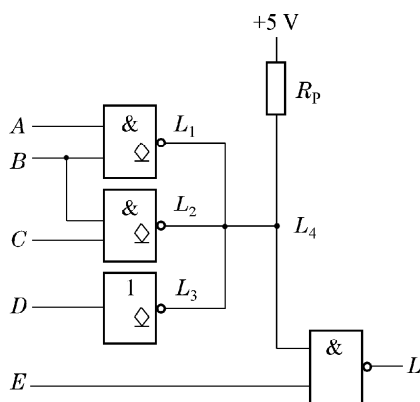


题图 3.7

3.8 TTL 集电极开路门电路组成如题图 3.8 所示电路,写出输出 Y 的逻辑表达式。



题图 3.8

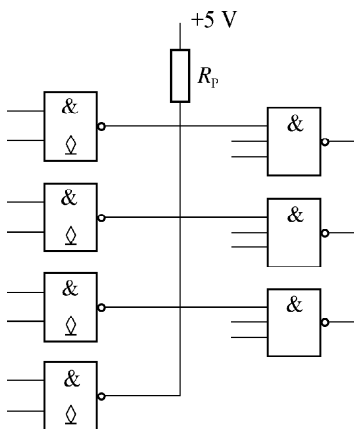


题图 3.9

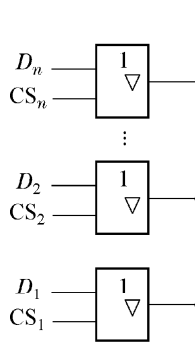
3.9 求如题图 3.9 所示电路输出 L 的逻辑表达式。

3.10 已知与非门 54LS00 的 $I_{OH(max)} = 0.4\text{ mA}$, $I_{OL(max)} = 4\text{ mA}$, $I_{IL(max)} = -0.4\text{ mA}$, $I_{IH(max)} = 20\text{ }\mu\text{A}$,试求该与非门的扇出系数。

3.11 试用 74LS03 中的 2 输入集电极开路 4 个与非门作“线与”连接去驱动 74LS10 中的 3 输入 3 个与非门,电路如题图 3.11 所示。已知 74LS03 的 $I_{OH} = 100\text{ }\mu\text{A}$, $I_{OL(max)} = 8\text{ mA}$, 74LS10 的 $I_{IH} = 20\text{ }\mu\text{A}$, $I_{IL} = -0.4\text{ mA}$, $V_{OH(min)} = 2.4\text{ V}$, $V_{OL(max)} = 0.3\text{ V}$,试求上拉电阻 R_P 的范围。



题图 3.11

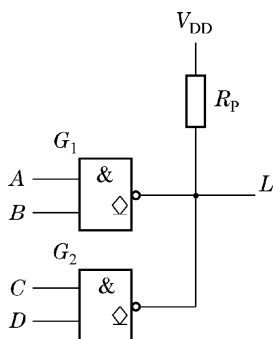


题图 3.12

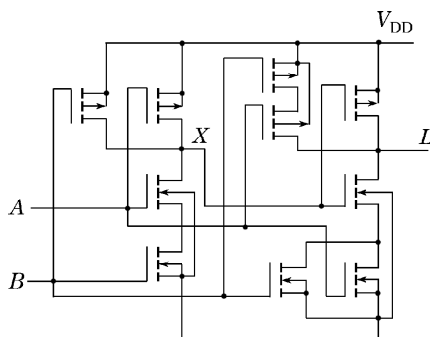
3.12 题图 3.12 表示三态门做总线传输的示意图,图中 n 个三态门的输出接到数据传输总线, $D_1, D_2, \dots, D_n$ 为数据输入端, $CS_1, CS_2, \dots, CS_n$ 为片选信号输入端。试问:

- (1) CS 信号如何控制,以便数据 $D_1, D_2, \dots, D_n$ 通过该总线进行正常传输?
- (2) CS 信号能否有两个或两个以上同时有效?
- (3) 如果所有 CS 信号均无效,总线处在什么状态?

3.13 在题图 3.13 的电路中,已知 OD 门 G_1, G_2 输出高电平时输出端 MOS 管的漏电流 $I_{OH(max)} = 5 \mu A$; 输出电流 $I_{OL(max)} = 10 \text{ mA}$ 时,输出低电平 $V_{OL} \leq 0.3 \text{ V}$ 。若取 $V_{DD} = 5 \text{ V}$, 试计算在保证 $V_{OH} \geq 3.5 \text{ V}, V_{OL} \leq 0.3 \text{ V}$ 的条件下,外接电阻 R_p 的取值范围。



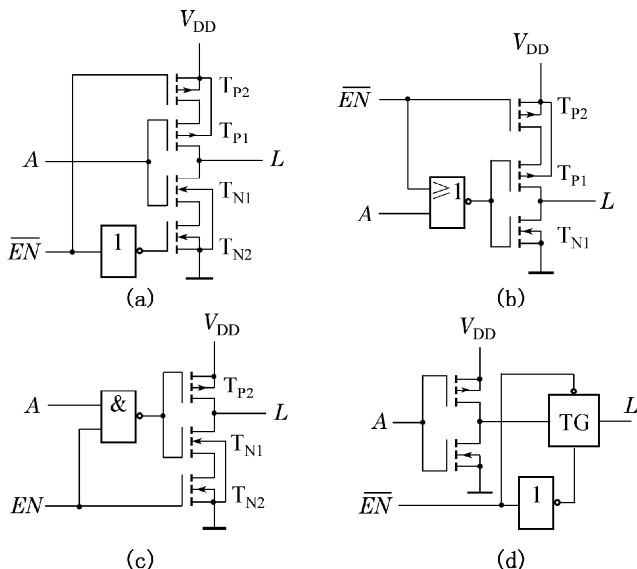
题图 3.13



题图 3.14

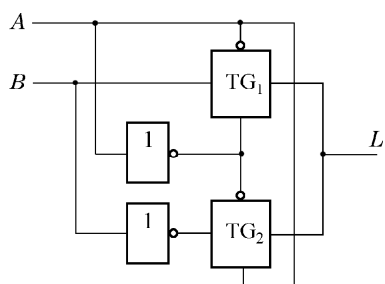
3.14 试分析题图 3.14 所示的电路,写出其逻辑表达式,说明它是什么逻辑电路。

3.15 试分析题图 3.15 所示的电路,说明它们的逻辑功能。



题图 3.15

3.16 试分析题图 3.16 所示传输门构成的电路,写出其逻辑表达式,说明它是什么逻辑电路。



题图 3.16

3.17 设计一发光二极管(LED)驱动电路,设LED的参数为 $V_F = 2.5\text{ V}$, $I_D = 4.5\text{ mA}$;若 $V_{CC} = 5\text{ V}$,当LED发亮时,电路的输出为低电平,选用集成门电路的型号,并画出逻辑电路图。